

美国地质调查局水资源调查技术丛书之六

MODFLOW: 模块化三维有限差分地下水流动模型

原著: Michael G. McDonald 和 Arlen W. Harbaugh

编译: 郭卫星 卢国平

审校: 朱学愚 刘建立

一九九九年六月

## 附 图

图 号		页 数
图 1	含水层的空间离散。	2-3
图 2	计算单元的划分: 单元中心系统和结点中心系统。	2-4
图 3	计算单元(i, j, k)和其六个相邻的计算单元。	2-7
图 4	从计算单元(i, j-1, k)至计算单元(i, j, k)的流量。	2-8
图 5	河流与含水层之间水力联系的理想模式。	2-11
图 6	计算单元(i, j, k)水头随时间的变化。	2-13
图 7	水头的迭代计算过程示意图。	2-16
图 8	含水层边界在模型中的表示。	2-20
图 9	垂向剖分示意图。	2-22
图 10	在砂层—粘土—砂层体系中地下水的可能流态。	2-23
图 11	模型剖面图。粘土层由六个模型层表示。	2-24
图 12	模型剖面图。粘土层的影响由垂向水力传导系数来反映。	2-25
图 13	MODFLOW总框图。	3-2
图 14	MODFLOW主要子程序包分类以及执行步骤。	3-4
图 15	MODFLOW主要子程序包以及执行步骤。	3-6
图 16	MODFLOW总体结构框图。	3-9
图 17	各子程序包在数组IUNIT中的位置以及使用方法示例。	3-17
图 18	IUNIT数组的作用示例。	3-18
图 19	利用IBOUND数组表示含水层的边界条件。	4-2
图 20	水头的计算过程示意图。	4-4
图 21	模拟时间与应力期及时间段的划分。	4-5
图 22	MODFLOW水均衡表实例。	4-7
图 23	达西流量计算示意图。	5-3
图 24	串联条件下的总水力传导系数。	5-4
图 25	平面上两相邻计算单元之间的水力传导系数的计算。	5-7
图 26	位于同一水文地质单元内上下两计算单元之间的垂向水力传导率的计算。	5-11
图 27	模型按水文地质单元分层时垂向水力传导率的计算。	5-12
图 28	两含水层中夹有弱透水层时垂向水力传导率的计算。	5-13
图 29	上层滞水示意图。	5-16
图 30	贮水系数/给水度转换示意图。	5-20

图 号		页 数
图 31	BCF子程序包流程示意图[略]。	
图 32	河流离散为河段的示意图。一些小河段被忽略。	6-2
图 33	(a)有河流出现的含水层横剖面图和(b)河流-含水层相互连接概念的示意图。	6-3
图 34	单个计算单元中河床水力传导系数的理想化模型示意图。	6-4
图 35	河床底积层底面之水头与计算单元水头关系的剖面示意图。 计算单元之水头等于潜水面高程。	6-6
图 36	河流流向某计算单元的流量, $QRIV$ , 与该单元之水头变量的关系示意图。 $RBOT$ 为河床底积层底面高程, $HRIV$ 为河水位。	6-8
图 37	单位水力梯度下的河流极限渗流示意图。	6-9
图 38	补给子程序包中三种选择下接收补给的计算单元。	7-3
图 39	计算单元 $(i, j, k)$ 的汇聚水流之水头损失横剖面示意图。	9-2
图 40	沟渠的直接相邻区域内水头损失的影响因素: (a)回填沟中埋设的排水管道和(b)明渠。	9-3
图 41	流入沟渠的流量 $QD$ 对计算单元之水头 $h$ 的图解。 其中, $d$ 为沟渠之设定水位, $CD$ 为水力传导系数。	9-5
图 42	一个计算单元中体积蒸发蒸腾量, $Q_{ET}$ , 作为水头, $h$ , 的函数图解。 其中 $d$ 为截止深度, $h_s$ 为 $ET$ 界面高程。	10-3
图 43	蒸发蒸腾子程序包中两种选择时发生蒸发蒸腾的计算单元。	10-5
图 44	通用水头边界子程序包原理图。	11-2
图 45	从通用水头边界水源流入某计算单元的流量 $Q_b$ 与该计算单元水头 $h$ 的函数关系示意图。 $h_b$ 为水源之水头。	11-3
图 46	表示一个含有二行, 三列, 两层的网格的有限差分方程组与其矩阵方程的对应关系。	12-3
图 47	系数矩阵的结构。斜线表示非零对角线。	12-4
图 48	一个含有二行, 三列, 两层的网格的对称系数矩阵。	12-5
图 49	系数矩阵与由此分解出的上, 下三角矩阵。	12-6
图 50	矩阵 $[A+B]$ 之结构。斜线表示非零对角线。	12-8
图 51	由矩阵 $[A+B]$ 分解出的下三角矩阵 $[L]$ 和上三角矩阵 $[U]$ 之结构,斜线表示非零对角线。	12-9
图 52	根据某函数在一个矩形三个顶点的已知值, 估计该函数在另一个顶点的值。	12-12
图 53	使用三元下标法和单一下标法分别表示计算单元在网格中的位置。	12-16

图号		页数
图 54	在SSOR子程序包中, 将三维模型分解为垂向分层进行处理。	13-2
图 55	图55. 分层方程组的系数矩阵及其在计算机中存贮形式。	13-7
图 56	工具子程序ULAPRW和ULAPRS的输出示例。该模型共含有七行十七列。	14-2

## 附 表

表 1	MODFLOW子程序包列表。	3-7
表 2	工具子程序ULAPRS和ULAPRW使用的打印格式代码。	14-3

目 录

译者的话	
摘要	
第一章 概述 .....	1-1
目的	1-1
本书内容的安排	1-1
致谢	1-3
第二章 多孔介质中地下水流动的有限差分公式数学模型 .....	2-1
离散化的概念	2-2
有限差分公式	2-5
迭代求解	2-15
差分方程的求解形式	2-18
计算单元的类型和边界条件的处理	2-19
有关垂向离散的讨论	2-19
第三章 MODFLOW程序设计 .....	3-1
总体结构	3-1
模型边界和含水层边界 .....	3-10
水均衡计算	3-12
内存分配	3-15
三维数组下标的定义	3-15
数据输入	3-16
结果输出控制	3-19
主程序	3-19
第四章 基本子程序包 (BAS) .....	4-1
概念及程序化	4-1
子程序包的选择及其输入文件	4-1
边界数组 (IBOUND)	4-1
初始条件	4-3
时间离散	4-3
输出	4-3
水均衡计算	4-6
基本子程序包输入数据及格式	4-8
输入数据说明	4-9
BAS子程序包输入样单	4-12
输出控制部分的输入数据	4-14
输入数据说明	4-15
输入控制选择输入样单	4-17
第五章 计算单元间渗流子程序包 (BCF) .....	5-1
概念及程序化	5-1
水力传导系数的计算	5-1
承压含水层中水平方向上水力传导系数的计算	5-5

潜水含水层中水平方向上的水力传导系数的计算	5-6
垂向水力传导率的计算 .....	5-8
非饱和带垂向渗流的计算	5-14
贮水系数的处理	5-17
贮水系数的转换	5-18
特殊处理方式的使用及其局限性	5-21
输入数据	5-21
BCF子程序包输入数据及格式	5-25
输入数据说明	5-27
BCF子程序包输入样单	5-29
第六章 河流子程序包 (RIV) .....	6-1
概念及程序化	6-1
河流子程序包输入数据及格式	6-11
输入数据说明	6-11
河流子程序包输入样单	6-13
第七章 补给子程序包 (RCH).....	7-1
概念及程序化	7-1
补给子程序包输入数据及格式.....	7-5
输入数据说明	7-5
补给子程序包输入样单	7-7
第八章 井流子程序包 (WEL).....	8-1
概念及程序化	8-1
井流子程序包输入数据及格式	8-2
输入数据说明	8-2
井流子程序包输入样单	8-4
第九章 沟渠子程序包 (DRN).....	9-1
概念及程序化	9-1
沟渠子程序包输入数据及格式	9-6
输入数据说明	9-6
沟渠子程序包输入样单	9-8
第十章 蒸发蒸腾子程序包 (EVT).....	10-1
概念及程序化	10-1
蒸发蒸腾子程序包输入数据及格式	10-6
输入数据说明	10-6
蒸发蒸腾子程序包输入样单	10-8
第十一章 通用水头边界子程序包 (GHB).....	11-1
概念及程序化	11-1
通用水头边界子程序包输入数据及格式	11-4
输入数据说明	11-4
通用水头边界子程序包输入样单	11-6
第十二章 强隐式法子程序包 (SIP) .....	12-1
概念及程序化	12-1
数组转换	12-17

计算顺序	12-17
迭代参数	12-19
SIP子程序包输入数据及格式	12-23
输入数据说明	12-23
SIP子程序包输入样单	12-24
第十三章 分层逐次超松弛法子程序包 (SSOR) .....	13-1
概念及程序化	13-1
SSOR子程序包输入数据及格式	13-8
输入数据说明	13-8
SSOR子程序包输入样单	13-9
第十四章 工具子程序和数据读入 .....	14-1
数组输入说明	14-1
参考文献	
附录A 干枯计算单元的几种处理方法 .....	A-1
附录B PCG2子程序包 (PCG) .....	B-1
PCG2子程序包输入数据及格式	B-3
输入数据说明	B-3
PCG2子程序包输入样单	B-5
附录C MODFLOW计算结果的处理 .....	C-1
MODFLOW的输出种类	C-1
无格式输出	C-1
无格式水头和降深文件的解读	C-2
无格式流量文件的解读	C-4
附录D 一些新开发的子程序包 .....	D-1

## 译者的话

### 一. MODFLOW简介

MODFLOW是英文Modular Three-dimensional Finite-difference Ground-water Flow Model (模块化三维有限差分地下水流动模型)的简称。这套计算机程序由美国地质调查局(U.S. Geological Survey)的McDonald和Harbaugh于八十年代开发出来。它是一套用于孔隙介质中地下水流动数值模拟的软件。自从它的问世以来, MODFLOW已经在全世界范围内, 在科研、生产、工业、司法、环境保护、城乡发展规划、水资源利用等许多行业和部门得到了广泛的应用。它已经成为目前世界上最为普及的地下水运动数值模拟的计算机程序。

不需对源程序进行任何修改, MODFLOW就可直接用来解决大多数地下水模拟问题。这为我们提供了一个相对来说标准化的软件, 也为同行之间的交流提供了方便。自从它问世以来, 人们已经对MODFLOW进行了多种测试, 至今尚未发现错误。所以, 它已经被世界上许多官方和司法机构所认可。

MODFLOW的一个最显著的特点是它采用了模块化的结构。它一方面将许多具有类似功能的子程序组合成为子程序包, 另一方面是用户可以按实际工作需要选用其中某些子程序包对地下水运动进行数值模拟。此外, 这种模块化结构使程序易于理解、修改、以至添加新的子程序包。事实上, 自从MODFLOW问世以来, 已经有许多新的子程序包被开发出来, 用来解决一些MODFLOW本身不能解决的问题。例如用于模拟河流与含水层之间水力联系的河流子程序包(Prudic, 1989), 用于模拟由于抽水引起地面沉降的子程序包(Leake和Prudic, 1988), 用于模拟水平流动障碍(Horizontal flow-barrier)的子程序包(Hsieh和Freckleton, 1993)等。这些新子程序包的加入, 大大增加了MODFLOW的功能和应用范围。

虽然MODFLOW本身仅限于模拟地下水在孔隙介质中的流动, 但大量实际工作经验表明, 只要合理使用, 它也可以用来解决许多地下水在裂隙介质中的流动问题。经过合理的线性化处理, MODFLOW还可以用于解决空气在土壤中的运动问题(Guo和Neville, 1998)。如果将它与其它用于溶质运移模拟的程序结合起来, 它还可以用来模拟诸如海水入侵等地下水密度为变量的问题(Guo和Bennett, 1998)。

MODFLOW之所以能得到如此广泛的应用, 一个很重要的因素就在于它在设计上的合



理性。很多参与MODFLOW设计、编程的人员，都是具有丰富实际经验的著名水文地质学家。这些都不难从MODFLOW对很多问题的处理方式中看出。因此，MODFLOW的使用说明书本身就是一部关于地下水数值模拟的教科书，值得广大水文地质工作者阅读参考。当然，MODFLOW也有它的缺点。例如对于干枯计算单元的处理，还很不合理。

MODFLOW最初由FORTRAN66版本写成 (McDonald和Harbaugh, 1984)。原作者又在1988年将其移植为FORTRAN77语言 (McDonald和Harbaugh, 1988)。虽然最近Harbaugh等人 (Harbaugh和McDonald, 1996) 将一些新开发的子程序包加入了MODFLOW，并将数据的输入输出方法做了改进；修改后的MODFLOW被称为MODFLOW96。但在主要内容上，MODFLOW96与MODFLOW88并没有太大的差别。大多数水文地质工作者仍采用MODFLOW88的使用说明书。本书的编译也以美国地质调查局于1988年出版的MODFLOW使用说明书为基础进行的。故在本书中，除非特别申明，MODFLOW均指1988年版的MODFLOW。

## 二. 关于MODFLOW说明书的编译

鉴于MODFLOW的种种优点及其应用的广泛性，我们深感有必要将MODFLOW程序的使用说明书翻译成中文以便中国广大的水文地质工作者学习使用MODFLOW。随着经济建设的发展和人口的不断增长，地下水资源的开发、利用和保护已显得日益重要。同时，随着电子计算机的普及，地下水运动的数值模拟也成为水文地质工作中一种经济、迅速、定量程度高的工作手段。在中国推广使用MODFLOW，无疑将有助于中国的水文地质工作，同时也能促进中国水文地质的科研工作、及时吸收国外的先进技术和经验，并促进对外科技文化交流。此外，中国的水文地质工作者也会在MODFLOW使用过程中不断总结提高，并根据实际情况，对MODFLOW加以改进和提高。

在翻译过程中，我们对MODFLOW使用说明书的原文作了适量的增删。一方面，原文中所附有的MODFLOW源程序及各子程序的文字说明均被略去。读者可以从INTERNET上下载MODFLOW的源程序。原文中的附录内容多已过时，故略去未译。另一方面，我们在本书的附录中，增加了有关干枯计算单元的讨论、计算结果的处理等内容。考虑到共轭梯度法 (PCG2, Hill, 1990a)应用的广泛性，我们对这一方法的基本原理和使用方法也作了简单的介绍。另外，原书中的第十三章未附有输入格式和输入样单。为一致起见，我们在译本中增加了这一内容。原文中所有的公式均为保留。除了图31外（该图出现于BCF子程序包子程序说明之中），原文中的其它图件均为保留。除了原文的参考文献之外，我们还加入

了其它一些文献供读者参考。为帮助中国读者了解MODFLOW的发展情况，我们在附录D中列出了一些由美国地质调查局开发的新子程序包的资料。在译文中，我们根据工作实践经验和体会，加入了一些解释，并置于方括弧中。

经过一年多的时间、多方面的努力，我们终于完成了MODFLOW使用说明书的翻译工作。本书的初稿基本上为编译，于一九九八年六月完成，由南京大学地球科学系出版并作为同年七月初在南京大学举办的MODFLOW学习班的教材。之后，我们按照美国地质调查局的意见，将原文中第十二、十三章全文译出，并将主要添加内容移至附录部分。

本书的翻译工作由美国Missimer International (Camp Dresser & Mckee, Inc.) 郭卫星博士主持。第一章至第五章及第十二章和第十四章由郭卫星翻译。第六章至第十一章和第十三章由美国阿拉巴马大学地质系博士研究生卢国平翻译。附录各部分由郭卫星编写。大部分中文输入工作由南京大学地球科学系研究生班完成，并由南京大学地球科学系博士研究生刘建立负责校对工作。全书由南京大学地球科学系朱学愚教授审阅。美国地质调查局的简小东博士和王文振博士对译文提供了宝贵的修改意见。

### 三. 鸣谢

在全书的翻译过程中，我们得到了下列部门和个人的鼎力协助，在此一致鸣谢：美国国家地质调查局 (U. S. Geological Survey)，美国Missimer International, Inc.，美国S. S. Papadopoulos & Associates, Inc.，中国南京大学地球科学系，朱学愚教授和刘建立等同学（中国南京大学地球科学系），郑春苗博士 (The University of Alabama)，卫迦教授（成都理工大学），Dr. Charles Andrews 和Mr. Gordon D. Bennett (S. S. Papadopoulos & Associates, Inc.)，Dr. Thomas E. Reilly 和Mr. Arlen W. Harbaugh (U. S. Geological Survey)。

我们在此也对在本书的翻译工作过程中，我们的太太和孩子们对这项志愿性工作的理解和支持深表谢意。

本书的翻译工作，完全是利用业余时间完成的。作为译者，我们唯一的愿望就是希望这本书能够帮助中国的广大水文地质工作者了解并能使用MODFLOW这一世界范围已得到广泛应用的地下水运动模拟计算机程序，以促进数值模拟技术在中国水资源利用和环境保护工作中的应用。对译文中可能出现的不正之处，我们热忱欢迎读者提出批评建议。

## 模块化三维有限差分地下水流动模型

Michael G. McDonald 和 Arlen W. Harbaugh

### 摘要

本书描述一个有限差分模型及其相关的模块化计算机程序。这个模型用于三维地下水运动模拟。本书介绍了建立地下水流动模型的物理和数学概念以及如何将这些概念应用于计算机程序。这里谈的模块化结构由一个主程序和一系列具有高度独立性的子程序（模块）组成。这些模块又组合成为子程序包。每个子程序包用于代表所模拟水文体系中的某一特征，例如来自河流的渗流补给、排泄于沟渠的地下水等；或者对描述地下水运动的线性方程组进行求解，例如强隐式法和超松弛因子法等。

程序的模块化结构有利于用户分别研究整个模型中某一水文地质特征。这种结构也有利于在增加新的子程序包时，不用修改现有的子程序包。MODFLOW的输入和输出格式具有很高的灵活性。

本程序采用单元中心有限差分法模拟地下水在含水层中的运动。模型层可以用来代表承压含水层、潜水含水层、以及二者的组合。模型还可模拟各种外应力，例如井流、面状补给、蒸发蒸腾、沟渠和河流等对地下水流动的影响。有限差分方程组可以用强隐式法或超松弛因子法进行求解。

本程序用FORTRAN 77 语言写成，可直接在大多数装有FORTRAN 77 编译系统的计算机上运行。本书对程序中的每一个子程序都给予详细描述，并附有各个子程序的计算框图、变量表和源程序。[译注：原文中对各子程序的描述，框图，变量表和源程序略去未译。]

## 第一章 概述

### 目的

一系列用于模拟地下水二维和三维流动的计算机程序 (Trescott,1975;Trescott和Larson, 1976;Trescott,Pinder和Larson,1976) 已经问世, 并已在美国地质调查局内部以及其它部门的地下水模拟工作中得到广泛使用。MODFLOW就是在这些早期工作的基础之上发展起来的。其主要目的就是发展一套容易修改、实用、稍加修改就能在多种计算机上运行、能有效地利用计算机的内存并提高运算速度的计算机软件供水文地质工作者使用。

本书所描述的程序采用了模块化的结构。它一方面将许多具有类似功能的子程序组合成为子程序包, 另一方面用户可以按实际工作需要选用其中某些子程序包对地下水运动进行数值模拟。此外, 这种模块化结构使程序易于理解、修改、以至添加新的子程序包。此外, 不被选用的子程序可以从程序中删除以减小程序的长度。这套程序可用于二维或三维地下水的数值模拟。用户将输入数据按一定格式存入一系列输入文件并由程序读入。灵活多样的输入格式使程序可以直接输入数据而不用对程序进行修改。另外, 程序还提供了多种输出格式供用户选用。这套程序最初用FORTRAN66语言写成 (McDonald和Harbaugh,1984)。本书则介绍MODFLOW的FORTRAN77语言版本。这套程序具有高度的移植性, 可在多种电子计算机上运行。有些计算机则可能要求对源程序稍加修改。

本书介绍的MODFLOW程序[译注: 即1988年版]可以用来模拟水井、补给、河流、沟渠、蒸发蒸腾和“通用水头边界”。这套程序现包括两种迭代求解方法: 强隐式法 (SIP) 和超松弛因子法 (SSOR)。

### 本书内容的安排

本书的主要内容包括地下水运动数值模拟中的数学、物理概念, MODFLOW的程序结构以及使用方法、输入数据的类型及格式。程序可分为一个主程序和若干个高度独立的子程序 (称为模块, modules)。这些模块反过来结合成为子程序包 (packages)。每个子程序包含有若干个模块, 每个模块用以完成整个模拟计算工作的一部分。例如, 井流子程序包模拟抽水井、注水井, 河流子程序包模拟河流渗流, SIP子程序包用强隐式法对线性方程组进行迭代求解。用户可根据实际需要选用这些子程序包。每个子程序包将在本书中分章进行描述。第

二、三章则对整个程序加以描述。第二章介绍MODFLOW中使用的有限差分法；了解有限差分法的概念及数学推导，对了解MODFLOW的功能及局限性有很大的益处，同时对了解有限差分法在地下水模拟中的应用也很有帮助。第三章介绍整个程序的整体设计。在第十四章，我们将介绍一些具有特殊用途的子程序。

第四章至第十三章对各个子程序包逐一进行说明。对每个子程序包的介绍包括（1）概念和程序化；（2）输入数据的种类及格式。在第一部分中，主要对该子程序包所代表的水文地质过程、概念以及数学表达加以介绍。例如在河流子程序包中，首先介绍河流与含水层之间的水力联系，然后讨论如何定量表示这种水力联系，推导出通过河床的地下水渗流近似计算公式。最后介绍如何将这个计算公式应用于有限差分公式。第二部分的内容包括输入数据的种类、定义、输入的方法及格式。在每章的最后部分还列出了各项输入数据的定义，供用户参考。同时还给出了输入文件的例子。第十二章和第十三章分别介绍程序所使用的迭代求解方法。

在第四至十三章中介绍输入时，我们用了输入项目（items）的名称。一个输入项目可以是单独的一行记录(record)，或是多行记录，也可以是一个或多个数组。（MODFLOW中三维数组的输入总是由一系列二维数组依次进行。每个二维数组对应于模型的一层。）在每章的输入格式部分，列出了该子程序包所要求的输入项目的定义，以供用户参考。这些输入项按输入顺序编号。一般来说，每一输入项的描述包括两行（有时还带有注解）。对于那些只有一行或一组记录的输入项，第一行列出该项的名称，第二行给出由FORTRAN语言规定的输入格式。如果输入项为数组，则由第一行给出该数组的名称，而由第二行给出用以读入该数组的子程序名。有关这些子程序，将在第十四章中加以介绍。

对于多数子程序包而言，输入数据可以分为两大类：（1）一次性输入数据，它们在整个模拟过程中保持不变；（2）多次性输入数据，它们可能随时间发生变化（例如抽水量等），所以每个应力期（stress period）均要求一组输入数据。各个子程序包的输入数据又进一步按读入数据的子程序分组。每组数据之前为读入子程序的名称。输入项目名称用大写字母表示，同时这些名称也作为程序中该项数据的变量名。由大小写字母混合表示的输入数据名仅用于输入说明，而非程序中该项之变量名。第四章为基本子程序包。它含有两张输入数据表（即两个输入文件）。一张表列出了必需的输入数据；而另一张表可以省略，它包含进行输出控制(output control)的各项数据。

第四章至第十三章中列举的各输入项将在相应的章节中加以介绍。多数情况下，我们将举例说明输入方法。在第四章，用以输出控制的各项输入数据也将给予解释，并给出例子。

在进行模拟时，用户必须告诉程序选用哪些子程序包，同时也必须告诉程序输入文件的名称及路径。用户可通过一个一维数组，IUNIT，将这些信息传递给MODFLOW。输入到这个一维数组中的整数值也将成为程序读入相应输入文件的设备号。IUNIT数组的输入方法，将在第五章至第十三章中输入数据开始的部分作以介绍。输出控制的输入将在第四章中介绍。对于选用的子程序包，用户必须在IUNIT中指定的位置上输入一个正整数加以表示，而这个正整数也就是程序读入该子程序包输入文件的设备号。对不选用的子程序包，则在相应位置以零表示。有关IUNIT的用法，我们在第三、四章中还要详细说明。

总的来说，本书的第二、三章和第四章中的“概念及程序化”部分对程序的总体设计以及功能进行讨论。某些参数的计算方法将在第五章中介绍。第六章至第十一章则讨论各个外部源汇子程序包的应用。第十二章和第十三章讲解程序中所使用的两种迭代求解方法。每个子程序包所要求的输入数据将在相应的章节中加以介绍。

### 致谢

本文作者特别感谢Gordon Bennett。他不仅在程序的开发过程中提供了行政管理上的协助，并不断给予我们鼓励和指导。他的审稿意见在很大程度上提高了本报告的易读性。

## 第二章 多孔介质中地下水流动的有限差分公式数学模型

在不考虑水的密度变化的条件下，孔隙介质中地下水在三维空间的流动可以用下面的偏微分方程来表示：

$$\frac{\partial}{\partial x} \left( K_{xx} \frac{\partial h}{\partial x} \right) + \frac{\partial}{\partial y} \left( K_{yy} \frac{\partial h}{\partial y} \right) + \frac{\partial}{\partial z} \left( K_{zz} \frac{\partial h}{\partial z} \right) - W = S_s \frac{\partial h}{\partial t} \quad (1)$$

其中：

$K_{xx}$ ,  $K_{yy}$ 和 $K_{zz}$ 为渗透系数在 $x$ 、 $y$ 和 $z$ 方向上的分量。在这里，我们假定渗透系数的主轴方向与坐标轴的方向一致，量纲为 $(LT^{-1})$ ；

$h$ : 水头(L)；

$W$ : 单位体积流量( $T^{-1}$ )，用以代表流进汇或来自源的水量；

$S_s$ : 孔隙介质的贮水率( $L^{-1}$ )；

$t$ : 时间 (T)。

一般来说， $S_s$ ,  $K_{xx}$ ,  $K_{yy}$ 和 $K_{zz}$ 都可能为空间的函数，而 $W$ 可能不仅随空间变化，还可能随时间发生变化。公式(1)描述了在三维空间中，当渗透系数主轴和坐标系方向一致时，地下水在孔隙介质中的流动。有关这个公式的导出，读者可以在许多教科书中找到(如Rushton和Redshaw, 1979)。

公式(1)加上相应的初始条件和边界条件，便构成了一个描述地下水流动体系的数学模型。从解析解的角度上说，该数学模型的解就是一个描述水头值分布的代数表达式。在所定义的空间和时间范围内，所求得的水头 $h$ 应满足边界条件和初始条件。但除了某些简单的情况，公式(1)的解析解一般很难求得。因此，各种各样的数值法被用来求得公式(1)的近似解。其中一种就是有限差分法。

在有限差分法求解过程中，连续的时间和空间被划分成为一系列离散的点。在这些点上，连续的偏导数也由水头差分公式来取代。将所求的未知点联合起来，这些有限差分式构成了一个线性方程组；然后对这个线性方程组进行联立求解。这样获得的解就是水头在各个离散点上的近似解。数值解虽然不能给出描述水头随时间和空间变化的代数表达式，但它可以用来解决大量的实际问题。

我们可以运用有限差分计算的方法对公式(1)以及相应的边界和初始条件求解。为了便

于讨论，我们通过物理概念对有限差分法进行介绍。

### 离散化的概念

如图1所示，我们将一个三维含水层系统划分为一个三维的网格系统，整个含水层被剖分为若干层，每一层又剖分为若干行和若干列。这样，含水层就可以由许多剖分成的小长方体所表示。这些剖分出来的小长方体称为计算单元(Cell)。有时这些计算单元也称为元素或格点。[译注：在本书中，我们统一使用计算单元这个名称。] 每个计算单元的位置可以用该计算单元所在的行号(i)，列号(j)和层号(k)来表示。假定一个含水层可以剖分为“nlay”层，而每一层又剖分为“nrow”行和“ncol”列，则i称为行下标，j称为列下标，k称为层下标。显然，我们有

$$i = 1, 2, \dots, nrow$$

$$j = 1, 2, \dots, ncol$$

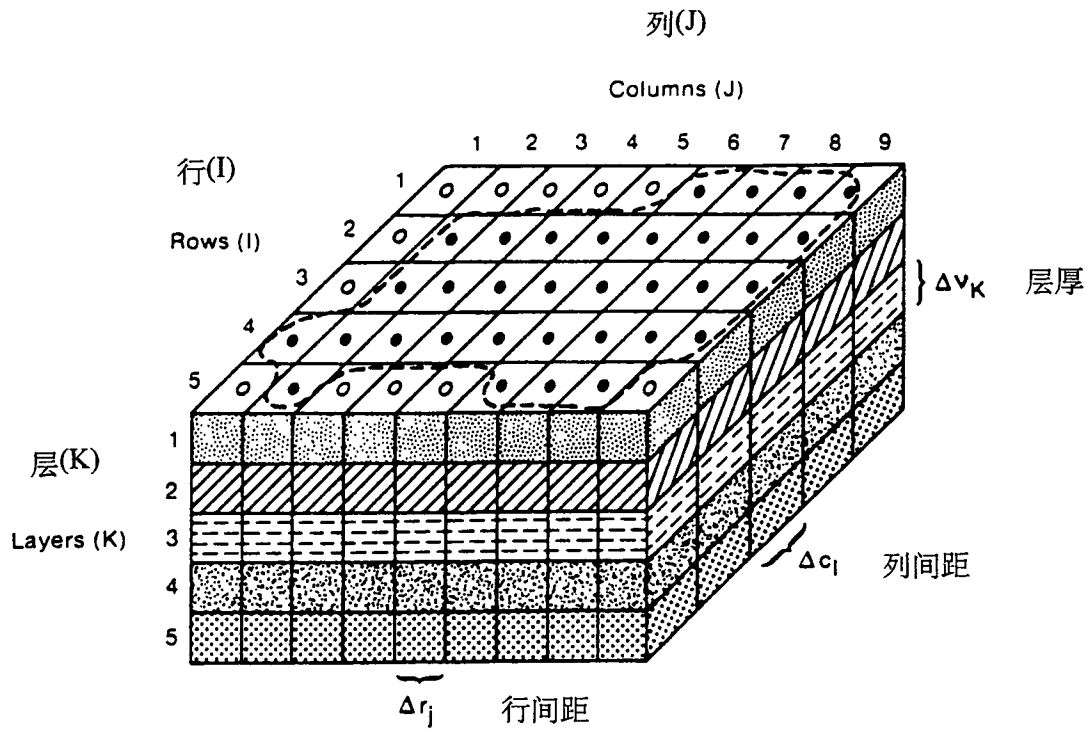
$$k = 1, 2, \dots, nlay$$

例如，在图1中所示的含水层系统被剖分为5层，5行和9列。在剖分过程中，我们一般尽量使模型的分层与实际含水层的水平分层一致。因此，在一个卡笛尔坐标系中，下标k通常沿垂直方向z而变化。在MODFLOW中，一个模型的第一层（即k=1）规定为最顶层，所以k的值随着高程降低而增加。MODFLOW还规定行的延伸方向与x轴平行，列的延伸方向与y轴平行。因此列下标(j)随x值增加而增加，而行下标(i)随y值降低而增加。图1所示的标号就是依据这些规定所绘出的。值得注意的是，MODFLOW仅要求一个模型的行与列相互正交，而并不要求用户定义实际的x-y-z坐标系。事实上，MODFLOW中的所有计算并不需要知道计算单元(i,j,k)在空间中的绝对坐标(x, y, z)。

按照图1所示的规定，某列j中的一个计算单元沿行方向上的宽度由 $\Delta r_j$ 所表示。某行i中的一个计算单元沿列方向上的宽度由 $\Delta c_i$ 所表示。层k中的计算单元的厚度则由 $\Delta v_k$ 所表示。在理论上，位于同一层中的所有计算单元应具有相同的厚度。但在实际工作中，同层计算单元的厚度也常常不一样，以反映含水层厚度在空间中的变化。按上述定义，计算单元(4,8,3)的体积应为 $\Delta r_8 \Delta c_4 \Delta v_3$ 。

每个剖分出来的小长方体的中心位置称为节点。一个计算单元的水头实际上由水头在该节点的值所表示。正如图2所示，在一个二维平面上，有限差分法的计算可以分为两大类：单元中心法和结点中心法。不论采用哪种方法，整个含水层均由两组正交的平行线剖

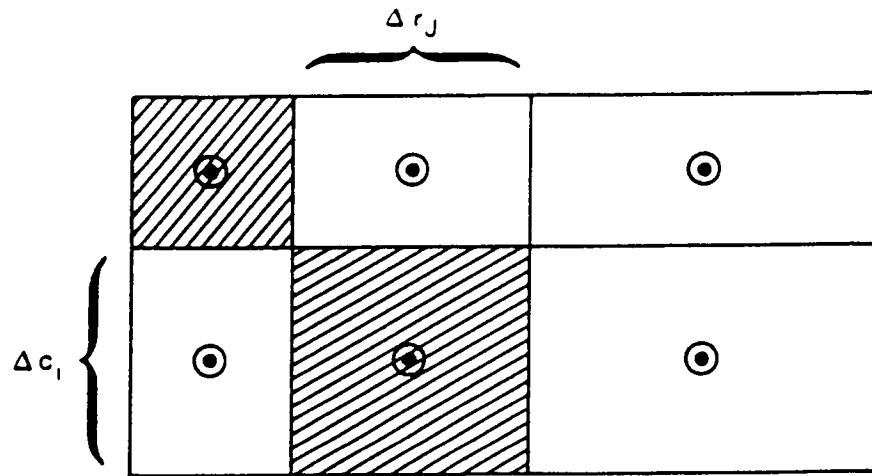




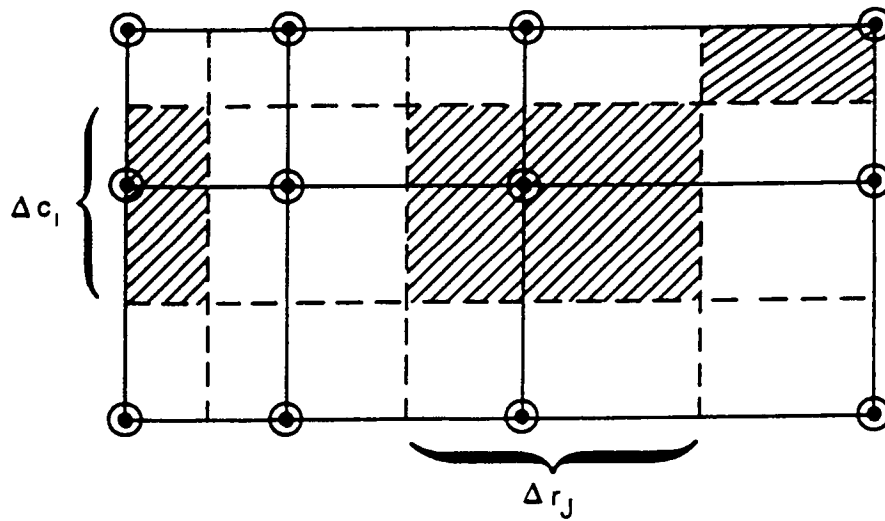
说明

- 含水层边界       $\Delta r_j$  行间距
- 有效计算单元       $\Delta c_i$  列间距
- 无效计算单元       $\Delta v_k$  层厚

图1. 含水层的空间离散。



单元中心系统



结点中心系统

说明

- ⊙ 计算单元的中心位置
- 网格
- - - 结点中心系统中计算单元的边界
- //// 计算单元的面积

图2. 计算单元的划分: 单元中心系统和结点中心系统。

分为一系列单元。在单元中心法中，这些平行线构成了计算单元的边界。而在结点中心法中，结点位于这两组正交直线的交点上，计算单元的边界则位于两个相邻结点中线的位置上。无论采用哪种方法，在计算单元的划分过程中，应尽量保持每个计算单元中的水文地质参数为常数。

这两种剖分方法各有优缺点。二者之间的差别主要在于渗流边界的处理上。在单元中心法中，渗流边界总是位于计算单元的边线上。而在结点中心法中，渗流边界则与计算单元的中心位置相吻合。一般来说，在比较复杂的计算机程序中，由于对边界的数学处理比较容易，故多采用单元中心法。在MODFLOW中，也采用了单元中心法。然而，下面有关有限差分公式的推导，则适用于这两种方法。

由于所计算的水头值，既是空间的函数，也是时间的函数。因此，我们不仅得将含水层进行空间上的离散，同时也得对时间进行离散。

### 有限差分公式

地下水运动的有限差分公式实际上是根据地下水流动的连续性方程进行的。按照连续性方程，流入和流出某个计算单元的水流之差应等于该单元中贮水量的变化。当地下水的密度不变时，连续性方程可以简单地表示为：

$$\sum Q_i = SS \cdot \frac{\Delta h}{\Delta t} \cdot \Delta v \quad (2)$$

式中：

$Q_i$ ：单位时间内流进或流出该计算单元的水量( $L^3T^{-1}$ )；

$SS$ ：含水层的贮水率( $L^{-1}$ )。它表示当水头变化为一个单位时，该含水层单位体积中所吸收或释放的水量；

$\Delta v$ ：计算单元的体积( $L^3$ )；

$\Delta h$ ：某一时间段内水头的变化( $L$ )；

$\Delta t$ ：时间变化量 ( $T$ )。

公式(2)中的右侧项表示在单位时间内，当水头变化为 $\Delta h$ 时含水层的贮水量变化。在这个公式中，仅表示了由于地下水的流入而引起的贮水量的增加。同理，如果流出的地下水量大于流入的地下水量，则含水层的贮水量应为减少。

图3表示计算单元(i,j,k)和其相邻的六个计算单元。这六个相邻的计算单元的下标分别由(i-1, j, k), (i+1, j, k), (i, j-1, k), (i, j+1, k), (i, j, k-1)和(i, j, k+1)来表示。

为简化下面的推导, 我们用正号来表示流入某一计算单元(i,j,k)的流量, 用负号来表示流出计算单元(i,j,k)的流量。而在一般情况下, 这样的负号与达西公式本身所包含的负号相抵消。由达西公式, 我们可以得到在行方向上由计算单元(i,j-1,k)流入单元(i,j,k)的流量(见图4)为:

$$q_{i,j-\frac{1}{2},k} = KR_{i,j-\frac{1}{2},k} \Delta c_i \Delta v_k \frac{(h_{i,j-1,k} - h_{i,j,k})}{\Delta r_{j-\frac{1}{2}}} \quad (3)$$

其中:

$h_{i,j,k}$ : 水头在计算单元(i, j, k)处的值;

$h_{i,j-1,k}$ : 水头在计算单元(i, j-1, k)处的值;

$q_{i,j-\frac{1}{2},k}$ : 通过计算单元(i, j, k)和计算单元(i, j-1, k)之间界面的流量(L<sup>3</sup>T<sup>-1</sup>);

$KR_{i,j-\frac{1}{2},k}$ : 计算单元(i, j, k)和(i, j-1, k)之间的渗透系数(LT<sup>-1</sup>);

$\Delta c_i \Delta v_k$ : 横断面面积(L<sup>2</sup>);

$\Delta r_{j-\frac{1}{2}}$ : 计算单元(i, j, k)和计算单元(i, j-1, k)之间的距离(L)。

值得注意的是, 下标中的 $j-\frac{1}{2}$ 并不表示几何坐标, 而表示该变量在两计算单元之间的相当值, 例如两点间的平均值。在MODFLOW中, 这样的相当值多由调和平均值所代替。有兴趣的读者可参看Collins (1961)。在一维稳定流的情况下, (3)式可以用来求得自计算单元(i,j-1,k)到(i,j,k), 通过横断面积 $\Delta c_i \Delta v_k$ 的地下水流量。

通过其它五个界面的地下水流量, 均可按公式(3)类推。例如, 沿行方向从计算单元(i,j+1,k)至(i, j, k)的地下水流量为:

$$q_{i,j+\frac{1}{2},k} = KR_{i,j+\frac{1}{2},k} \Delta c_i \Delta v_k \frac{(h_{i,j+1,k} - h_{i,j,k})}{\Delta r_{j+\frac{1}{2}}} \quad (4)$$

同理, 我们可以写出:

$$q_{i+\frac{1}{2},j,k} = KC_{i+\frac{1}{2},j,k} \Delta r_j \Delta v_k \frac{(h_{i+1,j,k} - h_{i,j,k})}{\Delta c_{i+\frac{1}{2}}} \quad (5)$$

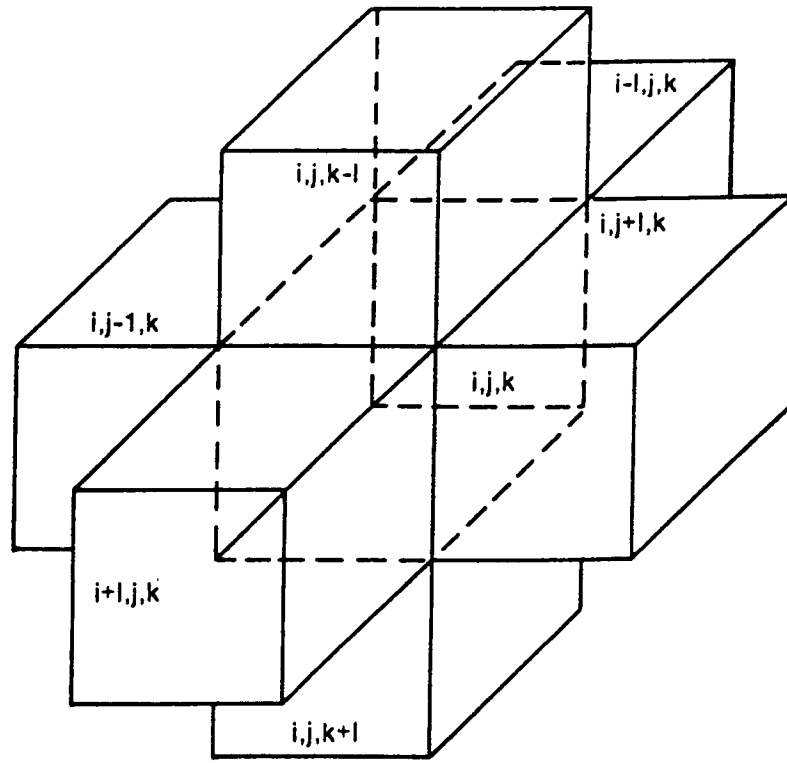


图3. 计算单元 $(i, j, k)$ 和其六个相邻的计算单元。

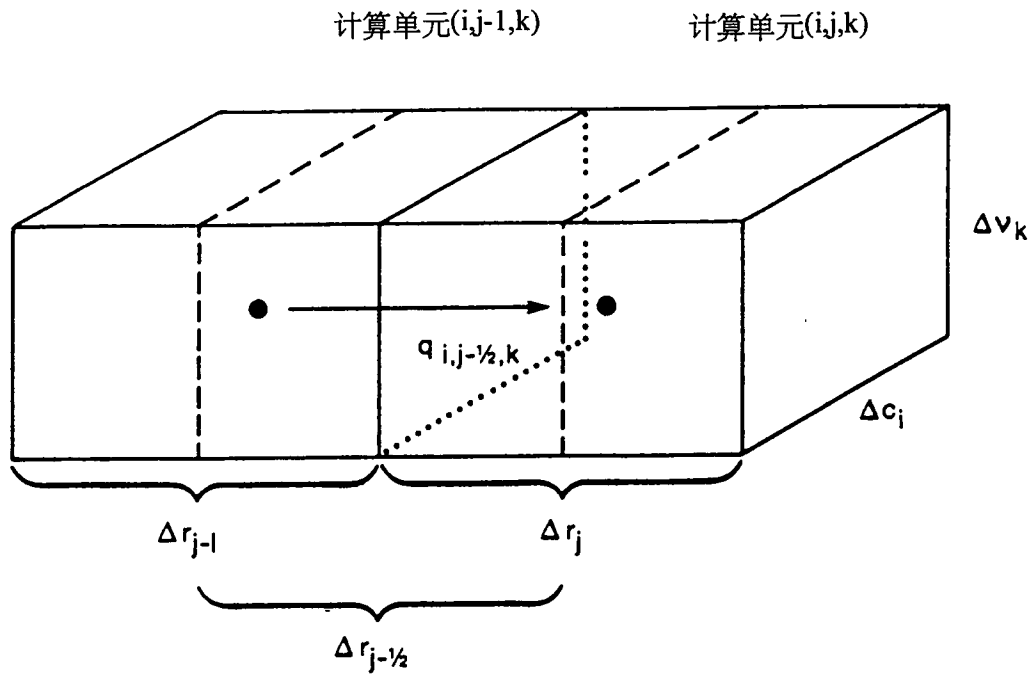


图4. 从计算单元(i, j-1, k)至计算单元(i, j, k)的流量。

$$q_{i-\frac{1}{2},j,k} = KC_{i-\frac{1}{2},j,k} \Delta r_j \Delta v_k \frac{(h_{i-1,j,k} - h_{i,j,k})}{\Delta c_{i-\frac{1}{2}}} \quad (6)$$

$$q_{i,j,k+\frac{1}{2}} = KV_{i,j,k+\frac{1}{2}} \Delta r_j \Delta c_i \frac{(h_{i,j,k+1} - h_{i,j,k})}{\Delta v_{k+\frac{1}{2}}} \quad (7)$$

$$q_{i,j,k-\frac{1}{2}} = KV_{i,j,k-\frac{1}{2}} \Delta r_j \Delta c_i \frac{(h_{i,j,k-1} - h_{i,j,k})}{\Delta v_{k-\frac{1}{2}}} \quad (8)$$

公式(3)至(8)表示通过计算单元(i,j,k)的六个界面上的地下水渗流量。这些公式中包括了水头、计算单元中心之间的距离和渗透系数。我们还可以简化这些公式，将计算单元间距和渗透系数合并为同一个变量，并称之为水力传导系数(hydraulic conductivity):

$$CR_{i,j-\frac{1}{2},k} = KR_{i,j-\frac{1}{2},k} \Delta c_i \Delta v_k / \Delta r_{j-\frac{1}{2}} \quad (9)$$

式中 $CR_{i,j-\frac{1}{2},k}$ 定义为位于层k行i中计算单元(i,j-1,k)和计算单元(i,j,k)之间的水力传导系数( $L^2T^{-1}$ )。它等于渗透系数和横断面积的乘积除以计算单元间距。

将水力传导系数的概念应用于公式(3)至公式(8)，我们可以得到:

$$q_{i,j-\frac{1}{2},k} = CR_{i,j-\frac{1}{2},k} (h_{i,j-1,k} - h_{i,j,k}) \quad (10)$$

$$q_{i,j+\frac{1}{2},k} = CR_{i,j+\frac{1}{2},k} (h_{i,j+1,k} - h_{i,j,k}) \quad (11)$$

$$q_{i-\frac{1}{2},j,k} = CC_{i-\frac{1}{2},j,k} (h_{i-1,j,k} - h_{i,j,k}) \quad (12)$$

$$q_{i+\frac{1}{2},j,k} = CC_{i+\frac{1}{2},j,k} (h_{i+1,j,k} - h_{i,j,k}) \quad (13)$$

$$q_{i,j,k-\frac{1}{2}} = CV_{i,j,k-\frac{1}{2}} (h_{i,j,k-1} - h_{i,j,k}) \quad (14)$$

$$q_{i,j,k+\frac{1}{2}} = CV_{i,j,k+\frac{1}{2}} (h_{i,j,k+1} - h_{i,j,k}) \quad (15)$$

公式(10)至公式(15)中的水力传导系数项均可按公式(9)的方式计算。这些公式可以用来计算从计算单元(i,j,k)的六个界面流入的地下水流量。除此以外，我们还得考虑其他各种外部源和汇对计算单元(i,j,k)的影响:例如河流、沟渠、生产井、注水井、蒸发蒸腾等等。这些外部源、汇对计算单元的水头 $h_{i,j,k}$ 的影响，有些与该水头有关，有的则与所求水头无关。自外部源汇流入计算单元(i,j,k)的流量可以用一个通式来表达:

$$a_{i,j,k,n} = p_{i,j,k,n} h_{i,j,k} + q_{i,j,k,n} \quad (16)$$

式中： $a_{i,j,k,n}$ 第n项外部源对计算单元(i,j,k)的补给量 ( $L^3T^{-1}$ )。 $q_{i,j,k,n}$ 和 $p_{i,j,k,n}$ 均为常数，它们的单位分别为( $L^3T^{-1}$ )和( $L^2T^{-1}$ )。

例如，计算单元(i,j,k)从两种外部源得到补给：注水井和河床的渗漏补给。对第一种源而言， $n=1$ 。因为假定注水量与该计算单元的水头无关，则 $p_{i,j,k,1}=0$ ；而 $q_{i,j,k,1}$ 等于该注水井的注水量。这时我们有

$$a_{i,j,k,1} = q_{i,j,k,1} \quad (17)$$

对于第二种源( $n=2$ )来说，由于通过河床的渗漏补给量与河水位与该计算单元的水头之差成正比，这个比例系数称为河床的水力传导系数。河床的水力传导系数直接反映了河床堆积物对渗流的阻力。据此可得：

$$a_{i,j,k,2} = CRIV_{i,j,k,2} (R_{i,j,k} - h_{i,j,k}) \quad (18)$$

式中 $R_{i,j,k}$ 为河水位， $CRIV_{i,j,k,2}$ 为河床的水力传导系数( $L^2T^{-1}$ )。

如图5所示，河床的水力传导系数等于河床堆积物的垂向渗透系数与河床的面积之乘积，除以河床堆积物的厚度。公式(18)也可以写为：

$$a_{i,j,k,2} = -CRIV_{i,j,k,2} h_{i,j,k} + CRIV_{i,j,k,2} R_{i,j,k} \quad (19)$$

其中 $-CRIV_{i,j,k,2}$ 一项相当于公式(16)中的 $p_{i,j,k,2}$ ，而 $CRIV_{i,j,k,2} R_{i,j,k}$ 相当于公式(16)中 $q_{i,j,k,2}$ 项。与此类似，所有其它外部源和汇的计算都可以参照公式(16)的形式表达。一般来说，如果一个计算单元受到N种外部源汇的影响，这些影响的总和可以归纳为：

$$QS_{i,j,k} = \sum_{n=1}^N a_{i,j,k,n} = \sum_{n=1}^N p_{i,j,k,n} h_{i,j,k} + \sum_{n=1}^N q_{i,j,k,n} \quad (20)$$

如果将式中的 $p_{i,j,k}$ 和 $q_{i,j,k}$ 分别表达为：

$$P_{i,j,k} = \sum_{n=1}^N p_{i,j,k,n}$$

$$Q_{i,j,k} = \sum_{n=1}^N q_{i,j,k,n}$$

则外部源和汇的一般计算公式为：

$$QS_{i,j,k} = P_{i,j,k} h_{i,j,k} + Q_{i,j,k} \quad (21)$$



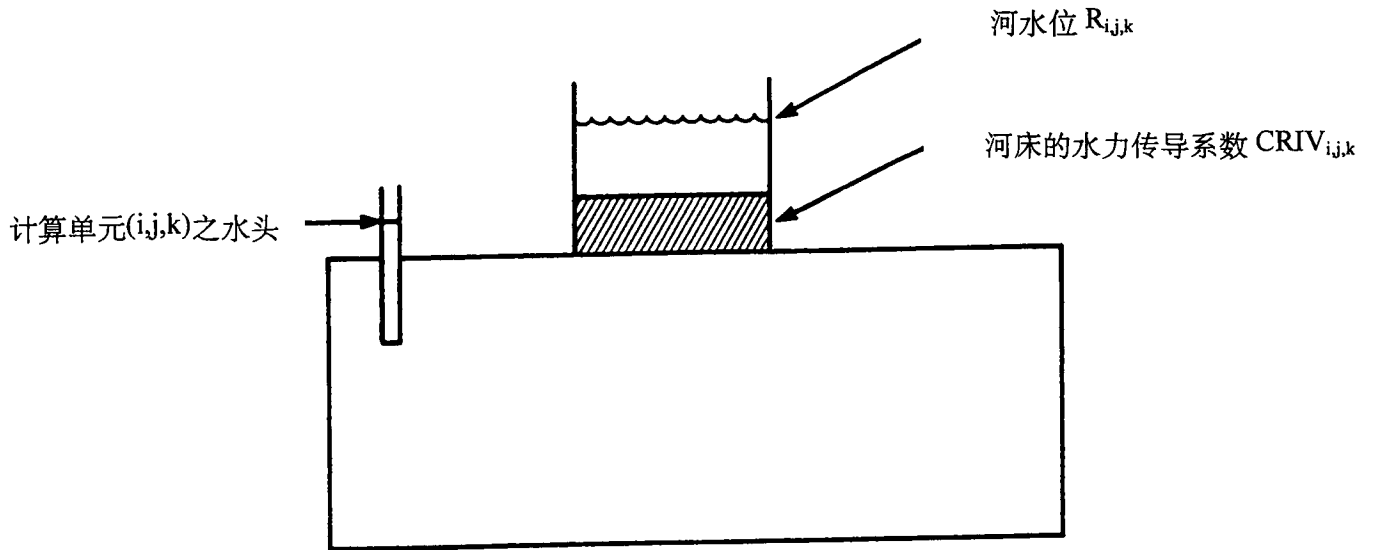


图5. 河流与含水层之间水力联系的理想模式。

考虑到某计算单元六个相邻的计算单元以及该单元所包含的所有源与汇，连续性方程，公式(2)可以表述为：

$$\begin{aligned} & q_{i,j-\frac{1}{2},k} + q_{i,j+\frac{1}{2},k} + q_{i-\frac{1}{2},j,k} + q_{i+\frac{1}{2},j,k} + q_{i,j,k-\frac{1}{2}} + q_{i,j,k+\frac{1}{2}} + QS_{i,j,k} \\ & = SS_{i,j,k} \frac{\Delta h_{i,j,k}}{\Delta t} \Delta r_j \Delta c_i \Delta v_k \end{aligned} \quad (22)$$

式中  $\frac{\Delta h_{i,j,k}}{\Delta t}$  为水头对于时间的偏导数之差分近似表达式 ( $LT^{-1}$ );

$SS_{i,j,k}$  表示该计算单元的贮水率 ( $L^{-1}$ );

$\Delta r_j \Delta c_i \Delta v_k$  为该计算单元的体积 ( $L^3$ )。

将公式(10)至(15)以及公式(21)代入公式(22)，则可获得计算单元(i,j,k)的地下水渗流计算的有限差分公式：

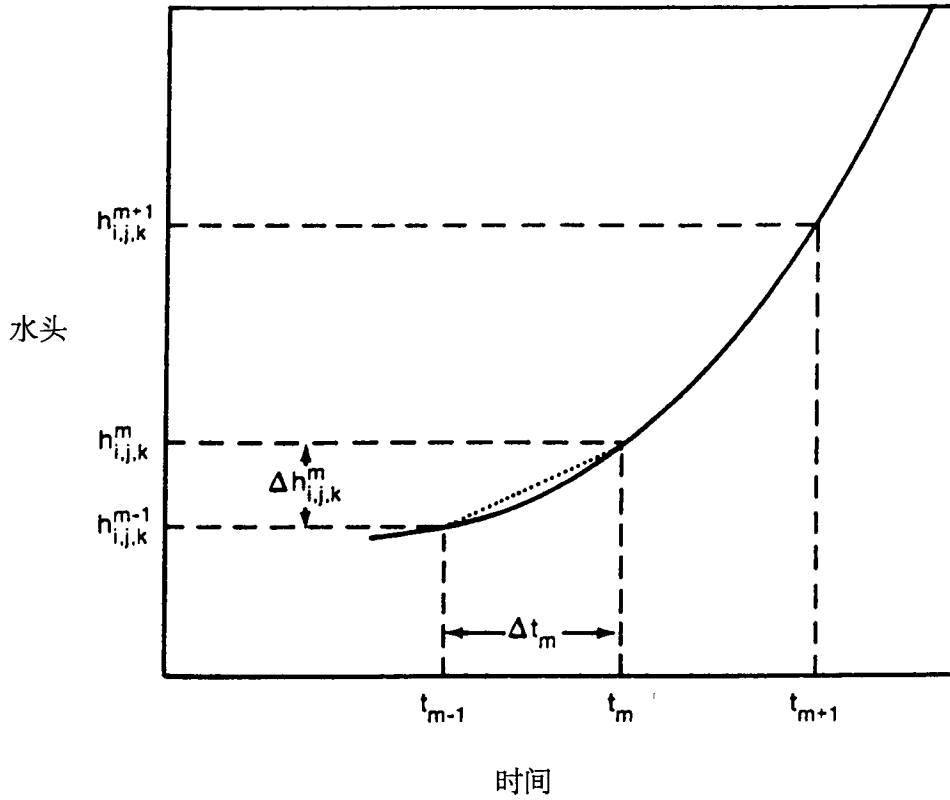
$$\begin{aligned} & CR_{i,j-\frac{1}{2},k} (h_{i,j-1,k} - h_{i,j,k}) + CR_{i,j+\frac{1}{2},k} (h_{i,j+1,k} - h_{i,j,k}) + CC_{i-\frac{1}{2},j,k} (h_{i-1,j,k} - h_{i,j,k}) \\ & + CC_{i+\frac{1}{2},j,k} (h_{i+1,j,k} - h_{i,j,k}) + CV_{i,j,k-\frac{1}{2}} (h_{i,j,k-1} - h_{i,j,k}) + CV_{i,j,k+\frac{1}{2}} (h_{i,j,k+1} - h_{i,j,k}) \\ & + P_{i,j,k} h_{i,j,k} + Q_{i,j,k} = SS_{i,j,k} (\Delta r_j \Delta c_i \Delta v_k) \frac{\Delta h_{i,j,k}}{\Delta t} \end{aligned} \quad (23)$$

下面我们来讨论水头的变化对时间的差分。图6表示了计算单元水头值  $h_{i,j,k}$  随时间的变化曲线。在横坐标轴上任选两个点， $t_{m-1}$  和  $t_m$ 。 $t_m$  表示当前的时刻， $t_{m-1}$  表示在此之前的某一时刻。与这两个时刻相对应的水头值不妨用  $h_{i,j,k}^m$  和  $h_{i,j,k}^{m-1}$  来分别表示。根据有限差分计算的方法，我们可以将水头对时间的偏导数用差商来近似表示：

$$\left( \frac{\Delta h_{i,j,k}}{\Delta t} \right)_m \approx \frac{h_{i,j,k}^m - h_{i,j,k}^{m-1}}{t_m - t_{m-1}}$$

这样的差分公式也称为向后差分法，因为在计算  $t_m$  时的水头值时，我们利用了该时间之前某一时刻的水头值。除了向后差分法，还有其它不同的差分方式。例如，我们也可以用当前时刻和其之后的某一时刻的水头值来计算  $\Delta h / \Delta t$ （即向前差分法），或者利用当前时刻之前与当前时刻之后的水头值来计算  $\Delta h / \Delta t$ （即中心差分法）。值得注意的是，向前差分与中心差分法都可能引起数值计算的非稳定性问题，即截断误差可能会随时间而增加，最终导致错误的计算结果或程序运行的中断。

当解的非稳定问题出现后，计算过程中的误差项将随着计算过程的推移而增长。由此



说明

$t_m$  时间段 $m$ 结束的时间

$h_{i,j,k}^m$  时间 $t_m$ 时计算单元 $(i,j,k)$ 之水头

..... 时刻 $t_m$ 水头变化以向后差分近似表达

图6. 计算单元 $(i, j, k)$ 水头随时间的变化。

而获得的计算结果毫无价值。相反，在数值计算中，向后差分是无条件稳定的。这也就是说，计算中引起的误差（如截断误差）将会随着计算过程的推移而减小。出于这种考虑，在利用有限差分法对地下水流方程求解时应当尽可能采用向后差分法。利用向后差分法求解时，要求对模型中所包含的所有有效计算单元的水头进行同步求解，因此计算量和内存需要量都会比较大。在MODFLOW中，采用了向后差分法。

下面我们就根据向后差分法将公式(23)重写。如果所有的流量项均以某一时间段长的结束时间 $t_m$ 为准，则有

$$\begin{aligned} & CR_{i,j-\frac{1}{2},k}(h_{i,j-1,k}^m - h_{i,j,k}^m) + CR_{i,j+\frac{1}{2},k}(h_{i,j+1,k}^m - h_{i,j,k}^m) + CC_{i-\frac{1}{2},j,k}(h_{i-1,j,k}^m - h_{i,j,k}^m) \\ & + CC_{i+\frac{1}{2},j,k}(h_{i+1,j,k}^m - h_{i,j,k}^m) + CV_{i,j,k-\frac{1}{2}}(h_{i,j,k-1}^m - h_{i,j,k}^m) + CV_{i,j,k+\frac{1}{2}}(h_{i,j,k+1}^m - h_{i,j,k}^m) \\ & + P_{i,j,k}h_{i,j,k}^m + Q_{i,j,k} = SS_{i,j,k}(\Delta r_j \Delta c_i \Delta v_k) \frac{h_{i,j,k}^m - h_{i,j,k}^{m-1}}{t_m - t_{m-1}} \end{aligned} \quad (24)$$

公式(24)即称为向后差分公式。该式可以用来对描述地下水在三维空间流动的偏微分方程（公式（1））进行数值求解。公式(24)中的 $Q_{i,j,k}$ 与初始水头均应为已知，应预先输入给程序。该公式中还包含有七个未知水头项，也就是我们要求的解。另外，公式(24)不能独立求解，因为它不仅含有计算单元(i,j,k)的水头值，还包含有与其相邻的另外六个计算单元的水头值。我们可以按公式(24)的方式，对整个模型所包含的所有计算单元写出类似于公式(24)的有限差分方程。如果模型中共有n个计算单元，我们就可以写出n个这样的方程式，并由这几个方程式构成一个n元方程组。最后通过对这个n元方程组联立同步求解，获得每一个计算单元的水头值。

进行非稳定流数值模拟的目的是预测在不同时刻的水头变化。这种水头变化取决于初始水头分布、边界条件、各种水文地质参数的分布以及各种外界源和汇的分布与强度。非稳定流模拟总是从初始水头开始，每一步求出每个时间段(time step)结束时的水头值，并用该值作为下一时间段的初始值。并重复这样的过程，直至所要求的时间结束。在第一个时间段的计算中，我们有 $m=2$ 以及 $m-1=1$ 。这时公式(24)的形式为：

$$\begin{aligned} & CR_{i,j-\frac{1}{2},k}(h_{i,j-1,k}^2 - h_{i,j,k}^2) + CR_{i,j+\frac{1}{2},k}(h_{i,j+1,k}^2 - h_{i,j,k}^2) + CC_{i-\frac{1}{2},j,k}(h_{i-1,j,k}^2 - h_{i,j,k}^2) \\ & + CC_{i+\frac{1}{2},j,k}(h_{i+1,j,k}^2 - h_{i,j,k}^2) + CV_{i,j,k-\frac{1}{2}}(h_{i,j,k-1}^2 - h_{i,j,k}^2) + CV_{i,j,k+\frac{1}{2}}(h_{i,j,k+1}^2 - h_{i,j,k}^2) \\ & + P_{i,j,k}h_{i,j,k}^2 + Q_{i,j,k} = SS_{i,j,k}(\Delta r_j \Delta c_i \Delta v_k) \frac{h_{i,j,k}^2 - h_{i,j,k}^1}{t_2 - t_1} \end{aligned} \quad (25)$$

其中上标1和2分别表示第一个时间段的起始值和终结值。我们可以参照上式对模型中的每一个计算单元写出这样的一个差分方程。当然无效计算单元单元或常水头计算单元可以除外。然后将这些差分方程联立求解，就可以获得第一个时间段结束时  $t=t_2$  时的水头分布。应当注意到，这组差分方程应当在每个时间段结束后重新定义，因为其中所包含的未知项已经成为已知项了。

### 迭代求解

有很多方法可以用来对线性方程组进行求解。当线性方程组中的方程个数比较少时，可采用直接求解的方法（如高斯消元法等）。但在解决地下水的流动问题中，我们常常碰到包含有上百万个未知数的线性方程组。对于这样的问题，我们通常采用迭代的方法进行求解。[译注：标准的MODFLOW都是采用迭代法求解的。在1996年的MODFLOW中Harbaugh加进去了一个直接求解的程序包。但这个程序包仅当所求的未知水头数目较少时才有实用价值。]

按照迭代求解的方法，求解过程开始时，每个水头未知的计算单元都应赋给初始水头或估计水头。对于非稳定流计算，这些初始值应为已知的初始条件。初始水头通常由野外实际观测资料获得。由于初始水头会影响计算结果，所以在准备输入数据时应当仔细。而对于稳定流计算而言，初始水头仅用于第一次迭代计算。故初始水头值可以由估计获得。一个好的数值模型，不论什么样的估计值，最终都应收敛于同样的结果。但初始值估计的好坏，和迭代的次数有关。如果初始水头的估计值与最终计算的结果比较接近，则可减少程序运行的时间。[译注：干枯计算单元的出现有时与水头的估计值有关。]

在迭代过程中，每次迭代的结果都将经过处理后用于下一次的计算。不同的算法有不同的处理方法。在正常情况下，每次迭代后的水头变化逐渐减小，最终达到收敛。这样就完成了一个时间段的水头计算。是否收敛，通常由一个预先定义的收敛指标来确定。当两次迭代计算的最大水头差值小于该收敛指标时，我们就称之为收敛。所以计算结果的精度与收敛指标的选定有很大的关系。

因此，在每个时间段的计算过程中，计算机内存贮水头的数组也不断更新。我们可以设想有许多数组，每个数组存贮一组计算结果。如图7所示，这些数组均由  $\bar{h}$  表示。 $\bar{h}$  有两个上标，第一个上标表示当前的时间段，第二个上标表示当前的迭代次数。例如  $\bar{h}^{m,l}$  表

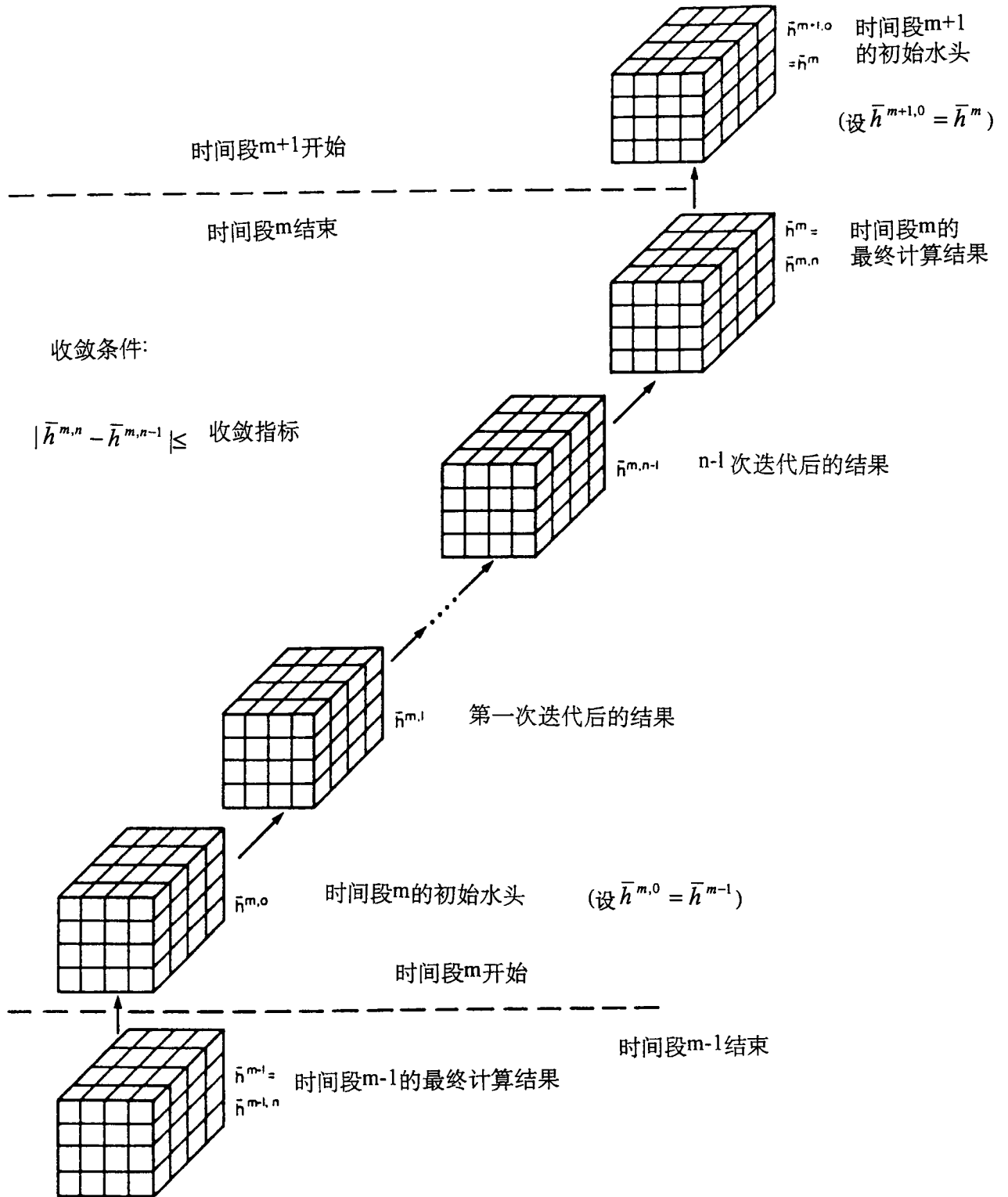


图7. 水头的迭代计算过程示意图。

示时间段 $m$ 中第一次迭代步骤之后的水头值； $\bar{h}^{m,0}$ 则表示第 $m$ 个时间段开始时的初始值。经过 $n$ 次迭代，才完成时间段 $m$ 的计算。最后一次迭代后的计算结果，既作为本时间段的结果，也将用于下一个时间段( $m+1$ )计算的初始水头。

当计算的水头值与实际水头非常接近时，我们总是希望结束运算。但问题在于我们无法预测需要多少次迭代才能得到满意的结果。为了防止程序无休止地迭代下去，我们常可采用一种间接的方法来结束迭代过程，即预计一个收敛指标和一个最大迭代次数。当相邻两次迭代计算出的水头变化的最大值小于该收敛指标时，程序就自动终止迭代而执行下一步运算指令。注意收敛指标指的是计算水头的变化量而非水头本身。通常收敛指标应比期望的计算精度至少小一个数量级。最大迭代次数是用来防止在不收敛的情况下，程序无休止地进行计算。如果实际的迭代次数已达到所规定的最大迭代次数而尚未收敛时，用户可以适当调整最大迭代次数以得到结果。但如果迭代过程中已显示出不可能收敛的趋势时，则应检查模型的结构和输入数据是否合理，并对不合理的部分进行修正后再重新计算。

当实际迭代次数达到最大迭代次数仍未收敛时，MODFLOW将会自动终止运行并打印出相应的信息供用户参考。收敛指标和最大迭代次数均为输入数据，在MODFLOW中分别由实型变量HCLOSE和整型变量MXITER表示。

对于稳定流计算，起始水头虽不可少，但它对最终结果的影响很小。从理论上来说，经过多次迭代之后，最终结果都应收敛于同样的水头值。当然，迭代次数与起始水头的选择有关。在进行非稳定流计算时，初始水头的选择就很重要了。在MODFLOW中，每个时间段结束时的水头计算值，也将用做下一时间段计算的初始值。正如图7所示，数组 $\bar{h}^{m-1,n}$ 为时间段 $m-1$ ，迭代次数为 $n$ 时的计算水头值。当时间段 $m-1$ 结束后，这些水头值将存入数组 $\bar{h}^{m,0}$ ，作为时间段 $m$ 的初始水头。每次非稳定流模拟只需要用户在模拟开始时输入初始水头。另外，图7中所示的数组系列仅为了说明水头数组所存内容随时间的变化。在MODFLOW的程序设计时，并不需要对每次迭代开辟一个三维数组，而是使用同一个三维数组并不断更新其内容。

有关各种各样对线性方程组迭代求解的方法，用户不难在许多参考书中找到。例如：Peaceman(1977)，Crichlow (1977)和Remson, Hornberger and Molz (1971)等书。通过阅读这些参考文献，读者可以加深对有关概念的理解，并熟悉矩阵计算、迭代参数的作用以及

收敛速度的各种影响因素。尤其是在阅读本书第十二、十三章之前，读这些参考书会很有帮助。

应当说明的是，通过迭代法所得到的解，仅仅是差分方程的近似解。其精度受很多因素的影响，如选定的收敛指标以及所用的迭代方法本身。即使每个时间段所得到的解是精确的，也仅仅是相对于在该时间段内所建立的差分方程组而言。对于偏微分方程公式(1)来说，它的数值解也是近似解而已。因为通过有限差分法所得到的解与对应的解析解相比，总带有截断误差。一般来说，这种截断误差会随网格间距和时间段的增加而增加。最后值得说明的是，即使是获得了对基本偏微分方程的精确解，对于野外条件而言，这种解仍然是近似解而已。因为野外观测的水文地质参数总带有误差。另外，水文地质边界的确定，也常常带有很大的的人为因素和不确定因素。

### 差分方程的求解形式

MODFLOW中采用了两种不同的迭代求解方法。为了使程序结构满足任何迭代解的格式，差分方程的形式需要改变。为建立起线性方程组的矩阵形式，我们将所有包括未知水头的项移到方程的左侧，而将所有的已知项移到方程的右侧，则有：

$$\begin{aligned}
 & CV_{i,j,k-\frac{1}{2}} h_{i,j,k-1}^m + CC_{i-\frac{1}{2},j,k} h_{i-1,j,k}^m + CR_{i,j-\frac{1}{2},k} h_{i,j-1,k}^m \\
 & + (-CV_{i,j,k-\frac{1}{2}} - CC_{i-\frac{1}{2},j,k} - CR_{i,j-\frac{1}{2},k} - CR_{i,j+\frac{1}{2},k} - CC_{i+\frac{1}{2},j,k} - CV_{i,j,k+\frac{1}{2}} \\
 & + HCOF_{i,j,k}) h_{i,j,k}^m + CR_{i,j+\frac{1}{2},k} h_{i,j+1,k}^m + CC_{i+\frac{1}{2},j,k} h_{i+1,j,k}^m \\
 & + CV_{i,j,k+\frac{1}{2}} h_{i,j,k+1}^m = RHS_{i,j,k}
 \end{aligned} \tag{26}$$

其中：

$$HCOF_{i,j,k} = P_{i,j,k} - SC1_{i,j,k} / (t_m - t_{m-1}) \quad (L^2 T^{-1})$$

$$RHS_{i,j,k} = -Q_{i,j,k} - SC1_{i,j,k} h_{i,j,k}^{m-1} / (t_m - t_{m-1}) \quad (L^3 T^{-1})$$

$$SC1_{i,j,k} = SS_{i,j,k} \Delta r_j \Delta c_i \Delta v_k \quad (L^2)$$

如果对模型所包含的计算单元逐个写出类似于公式(26)那样的差分方程，则可得一个线性方程组。这个方程组可用矩阵的形式表示为：

$$[A]\{h\}=\{q\} \tag{27}$$



其中 $[A]$ 为水头的系数矩阵;  $\{h\}$ 为所求的水头矩阵; 而 $\{q\}$ 表示各个方程中所包含的所有常数项和已知项。 $\{q\}$ 也称为右侧项。在MODFLOW中, 系数矩阵和右侧项是通过各个软件包来逐步建立起来的。最后MODFLOW根据这两个矩阵, 通过迭代法对 $\{h\}$ 求解。

### 计算单元的类型和边界条件的处理

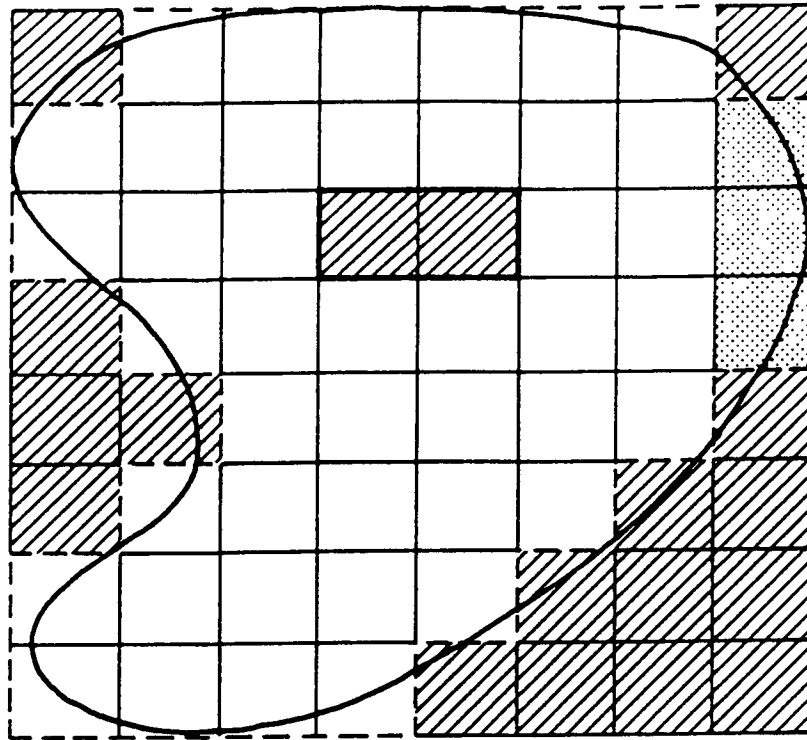
在实际工作中, 我们不必对模型中所包含的每个单元都写出类如公式(24)那样的差分公式, 因为有些单元用于代表相应的水文地质边界。它们的水头值是已知的。还有些单元可能位于所研究问题的边界之外, 它们的水头与问题本身无关。根据计算单元的性质, 可将它们划分为三大类: 定水头计算单元, 无效计算单元(inactive)和变水头计算单元。我们只须对变水头计算单元写出有限差分公式并计算其水头值。定水头计算单元的水头值是由用户事先规定, 并在计算过程中保持不变。无效计算单元相当于该单元的渗透系数为零, 不允许地下水通过, 故也可称为无渗流或不透水计算单元(no-flow cells)。除此之外其它所有的计算单元称为变水头单元(active cells); 它们的水头值随时间和空间发生变化, 是通过计算得到的。

定水头计算单元和无渗流计算单元可以用来描述边界上的水文地质条件。如图8中所示的含水层, 具有不规则的几何形状。假定将该含水层剖分为8行8列的等间距网格, 则有些单元落于边界之内, 也有不少单元落于边界之外。这些位于边界之外的单元则可称为无效计算单元, 它们可以从计算中排除。该含水层中含有一段定水头边界, 它可能表示与含水层相连的地表水体(如河流、湖泊等)。它们的水头值是已知并固定不变的。其它类型的水文地质边界, 如定流量边界, 则可利用外部源汇与无效计算单元的设置进行模拟。

[译注: 虽然无效计算单元和无渗流计算单元的作用在模拟效果上相同, 但它们所代表的意义并不相同。无效计算单元通常指那些位于水文地质边界之外的单元, 而无渗流计算单元可能位于边界之内, 但其渗透系数为零或忽略不计。]

### 有关垂向离散的讨论

一个数值模型在水平方向上的离散, 是将所模拟的区域在平面上剖分为相互垂直的行和列, 并定义这些行和列的间距。垂向上的离散则是将所模拟的岩层在垂向上进行剖分分层。这些层的厚度由用户在输入数据中定义, 或并入其它水文地质参数, 如导水系数的计



说明




- 含水层边界
- - - 模型中不透水边界
-  无效计算单元或不透水计算单元
-  定水头计算单元
-  变水头计算单元

图8. 含水层边界在模型中的表示。

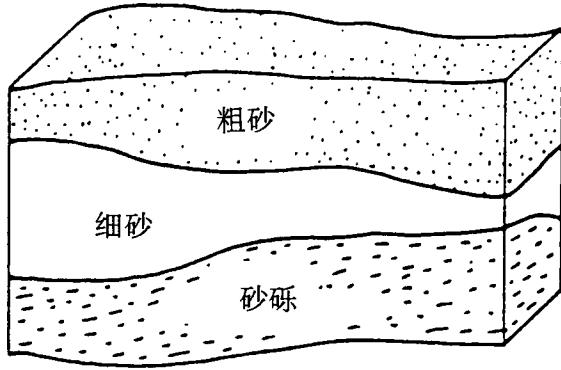
算。

在一种极端的情况下，垂向离散与平面离散相似，或多或少地将模型在垂直方向上划分为若干层。在另一种极端的情况下，则是将整个含水层看作一层来处理。在图9中，我们可以看到如何按这两种方法将一个典型的水文地质系统进行垂直离散。按照第一种情况（如图9b）将整个模型在垂向上剖分成若干层，而在第二种情况下，模型的垂直分层与含水层在空间上的分布一致（参见图9c）。在第一种情况下，模型被划分为三维正交网格，而不考虑地层的变化，但在同一层的所有计算元素具有相同的厚度。而若按第二种方法分层，模型的分层与地层相一致，但同一层中所含的计算单元将有不同的厚度。

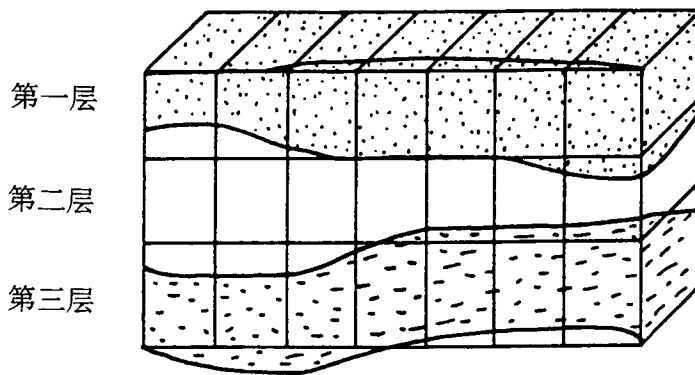
这两种方法各有其优缺点。在有限差分方程中，我们假定同一个计算单元内部的各项水文地质参数为常数，或至少可以用某种平均值来代替。这种假定实际上正符合如图9c所示的垂向离散法。此外，若不考虑水头在垂向上的损失，这种方法能提高计算结果的精确度。但是，按这种方法分层也违反了其它一些建立有限差分方程的假定。例如，某些计算单元不再具有规则的矩形形状。渗透系数的主轴方向也可能与模型的主轴方向产生不一致的情况。一般来说，违反这些假定将会在计算中导致某种程度的误差。

在实际工作中，多采用这两种方法相结合的方法。例如，当分层的边界与水文地质单元的边界一致时，为提高计算精度，常将同一层再细分为若干层来代表同一水文地质单元体。图10表示一个包括两层砂岩和一层粘土的水文地质系统。假定砂岩和粘土均具有同样的产状和厚度，则每一层都可以用一模型层来代表，而不必使用非矩形的计算单元。在这种情况下，地下水的流动既不水平又不垂直，而是以一定的夹角穿过粘土层。如果欲知道地下水在每个地层中的具体流向，则应将每个地层再细分为若干层，同样，对于图11中所示的砂岩与粘土互层的地质体系，不能简单地把模型在垂向上分为三层。如果在砂岩中抽水，其水量的一部分来自于粘土层贮水量的变化。如果建立模型的目的之一是研究粘土层的贮水量的变化情况，则应将该粘土层用数层模型层来表示。但如果该粘土层的释水量很小，水量基本来自砂岩，则地下水在砂岩中的流动基本上是水平的，而在粘土层中的流动为垂直的。在这种情况下，上下砂岩可分别由一模型层来代表，而粘土层则通过垂向水力传导系数来反映（图12）。这种方法有时称为“准三维”离散法。对此，我们将在第四章中进一步讨论。

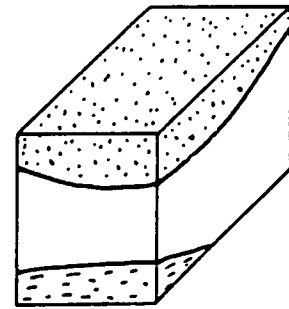
无论采用哪种方法进行离散，最终都可以得到类似于公式(26)那样的差分方程组。这



(a)含水层剖面图。

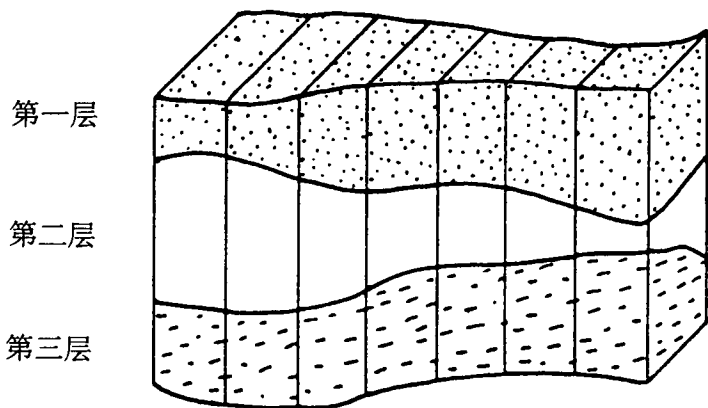


(b)长方体剖分网格。

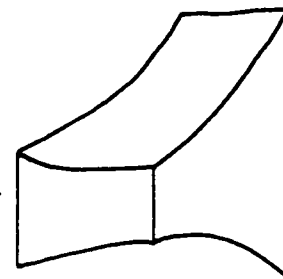


长方体计算单元。

同一单元内含有三种不同的地质体。



(c)曲面体剖分网格。



曲面六面体计算单元。

同一单元内仅包含一种地质体。

图9. 垂向剖分示意图。

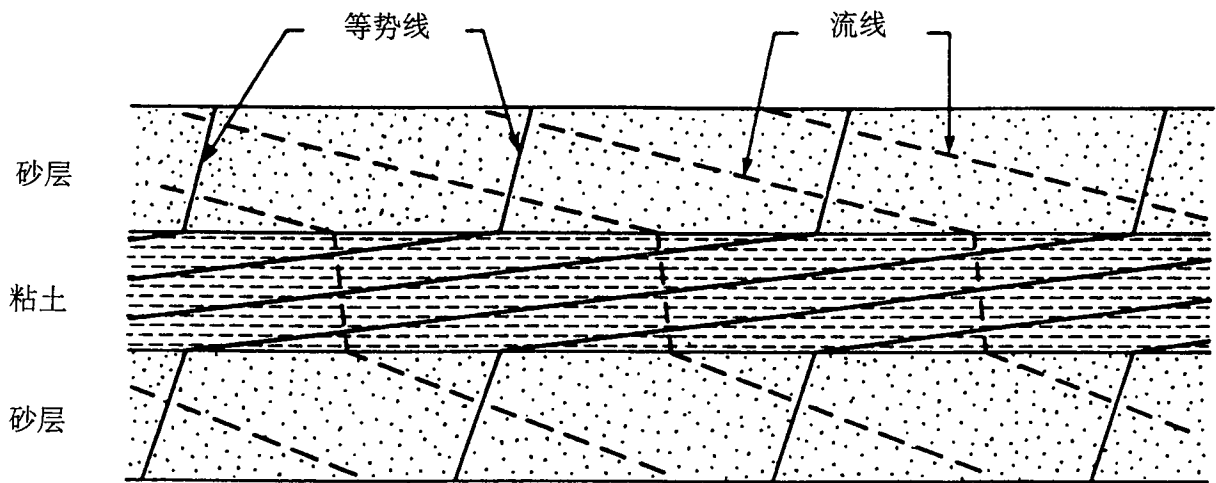


图10. 在砂层—粘土—砂层体系中地下水的可能流态。

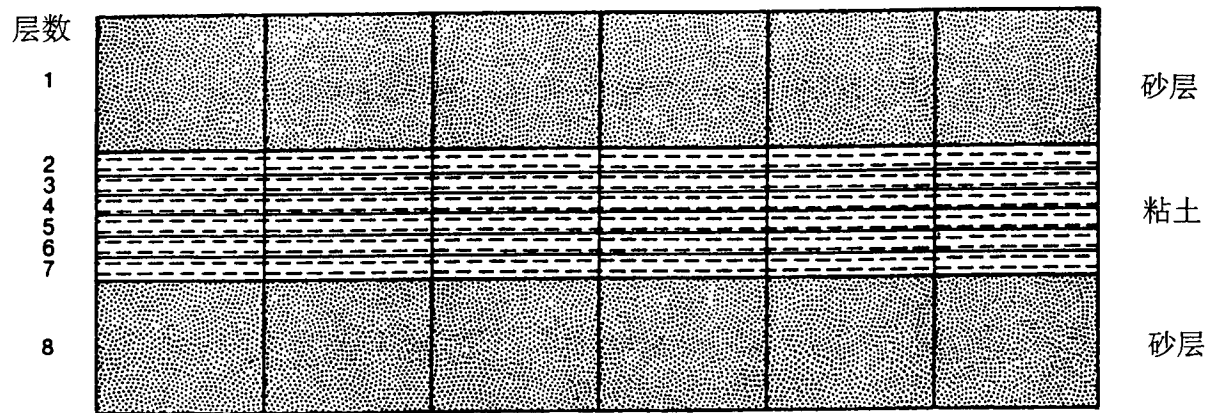


图11. 模型剖面图。粘土层由六个模型层来表示。

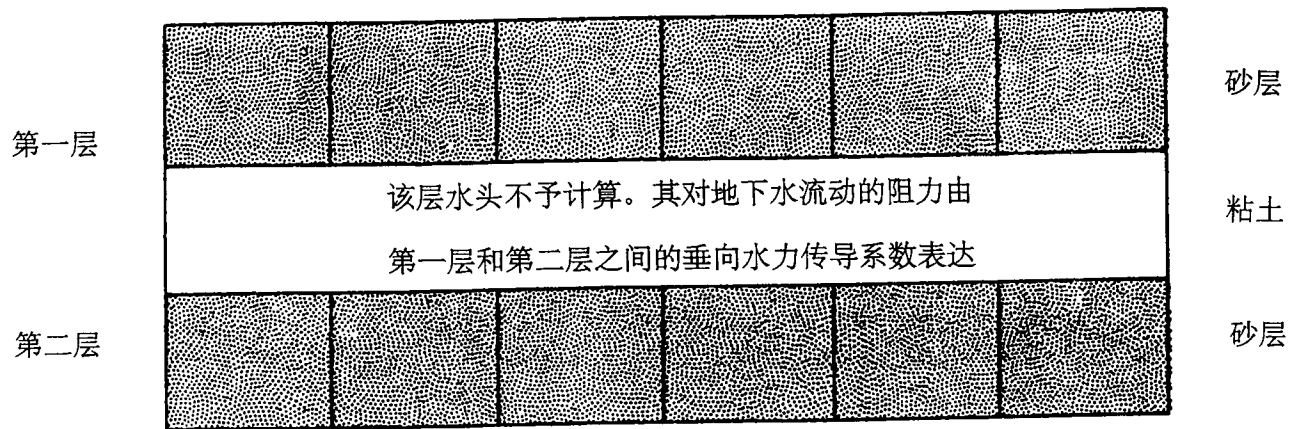


图12. 模型剖面图。粘土层的影响由垂向水力传导系数来反映。

个方程组必须在每个时间段内联立求解。前面所讨论的两种垂向离散方法的差别在于如何定义垂向水力传导系数、贮水项、方程组中方程的个数和计算结果的精度。MODFLOW允许用户采用所有这些方法。值得注意的是，程序的输入参数中并没有层厚这样的数值。但它们已包含在其它水文地质参数的计算之中，例如，在承压含水层的情况下，导水系数定义为渗透系数与厚度的乘积，贮水系数为贮水率与厚度的乘积。对于潜水含水层来说，输入数据包括含水层底板的标高和渗透系数，在程序运行过程中，MODFLOW根据水头和含水层底板标高来计算饱和厚度，然后由饱和厚度和渗透系数计算导水系数。因此，计算单元的厚度可能随底板标高和水头而变化。在第五章中讨论有关水力传导系数和贮水项的计算方法时，我们将进一步讨论垂向离散化的方法以及在程序中的处理。



## 第三章 MODFLOW程序设计

### 总体结构

本章中，我们介绍MODFLOW程序的总体设计。与任何一个FORTRAN程序一样，MODFLOW包括一个主程序和一系列相对独立的子程序包。每个子程序包又包括有多个模块和子程序。在这章中，我们将介绍主程序的功能以及子程序包的构成和它们的功能。

图13中列出了MODFLOW的基本程序结构，以及各个主要部分的功能。整个模拟过程可分为一系列应力期(stress periods)。在每一个应力期内，所有的外应力(stress)，如抽水量、蒸发量等保持不变。每个应力期又可再分为若干个时间段(time steps)。通过对有限差分方程组的迭代求解，我们可以得到每个时间段结束时的水头值。所以每个模拟应包括三大循环：应力期循环，时间段循环以及迭代求解循环。

图13中所示的每个矩形表示一个步骤(procedure)。每个步骤完成一定的任务。例如，在进入应力期循环之前，程序要首先完成三项与整个模拟过程有关的步骤。在“模型定义”步骤中，模型的大小，类型（稳定流或非稳定流），应力期数目，子程序包的选择，以及求解方法的选择等都在这个步骤中加以确定。在“存贮分配”步骤中，程序按所选用的子程序包，以及各个数组的大小，按一定的顺序进行内存的分配。在“输入处理”步骤中，程序读入所有不随时间变化的输入数据并且按要求进行适当的处理。这些数据包括：边界条件，初始水头，导水系数，渗透系数，给水度和贮水系数，顶面标高及底面标高以及用于控制迭代运算的有关参数等。在执行这个步骤的操作时，MODFLOW将对某些数据进行处理换算为后面程序运算所需要的数据类型。

在应力期循环的过程中，MODFLOW首先进入一个称为“应力”的步骤(Stress)。在该步骤中，MODFLOW读入时间段的数目并计算各个时间段的长度。在“输入处理”步骤中，程序将读入所有与当前应力期有关的数据，如抽水量、补给量等。在此以后，程序进入时间段循环。接下来，程序将执行另一个输入操作(Advance)，计算当前时间段的步长，并准备水头计算的初始值。然后MODFLOW开始用迭代的方法对水头进行求解。迭代循环的过程中包括建立方程(Formulate)步骤，其任务是计算公式(27)中的系数矩阵。在此以后，程序执行“近似求解”(Approximate)步骤，即对有限差分方程组进行一次迭代计算。迭代求解循环将持续进行直到达到收敛或达到预定的最大循环次数。如果迭代次数达到最大循环次数时

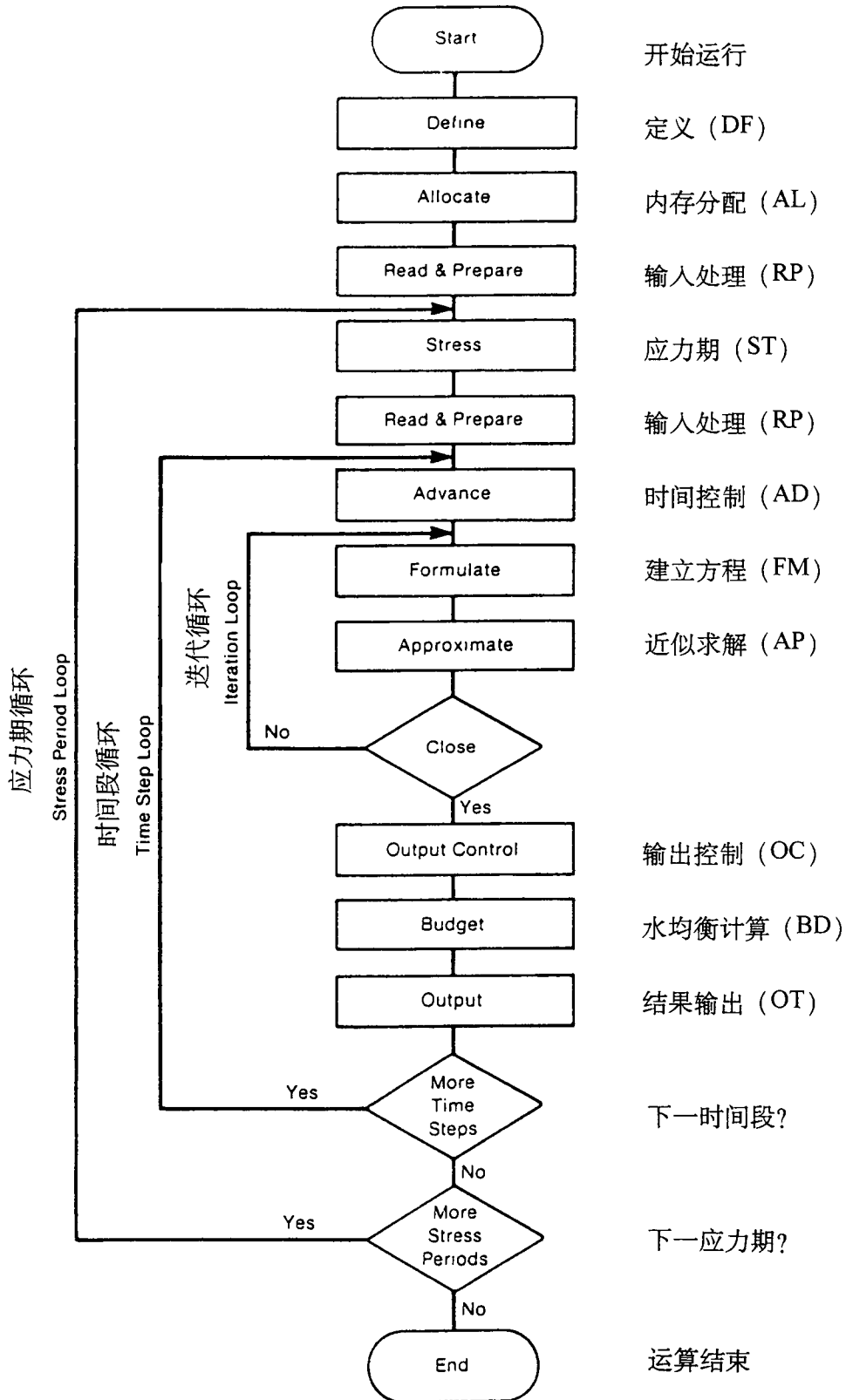


图13. MODFLOW 总框图。

仍未收敛，程序将自动停止运算。如果在达到最大迭代次数之前已经收敛，MODFLOW将执行输出（Output control）步骤和水均衡（Budget）步骤：将计算得到的水头值以及计算单元之间的流量按要求输出，并打印出有关信息供用户参考。程序还将计算出水均衡各项，并打印出水均衡计算得到的相对误差。

上面所叙述的各个步骤，可以用图13所示的框图来表示。图13也描述了MODFLOW的程序结构。该图也是主程序的程序框图。图中的各个步骤均由相应的子程序来实现。在主程序中使用了許多条件判断语句，根据具体情况由主程序调动有关的子程序来完成模拟计算。在MODFLOW中，子程序可分为两种类型：直接由主程序调用的基本子程序和被子程序所调用的次级子程序。

图13中所示的各个步骤由不同的子程序来完成。这些子程序组合在一起共同完成某些特别的功能。正如在第一章中所指出的，某些具有特定功能的子程序结合起来组成子程序包，如河流子程序包等。MODFLOW通过调用某些特定的子程序包来实现对地下水运动的模拟计算。了解每个子程序包中所包含的子程序以及它们的功能对了解整个程序的结构很有帮助。例如，从子程序包的分类可以看出该子程序包的功能以及执行过程。有些子程序包在每次模拟中都不不可少（如基本子程序包），而有些子程序包则由用户根据研究的问题进行选用。

步骤的划分，则是按其所包含的子程序的功能来进行的。同一步骤可能需调用不同子程序包中的某些功能相似的子程序来完成某一特定的操作。例如，在内存分配的步骤中，MODFLOW需调用不同子程序包中用于内存分配的子程序分别对该子程序包运算时所需要的内存空间进行分配。若某个模拟所涉及的子程序包数目较少，则在内存分配步骤中调用的子程序数目也较少。所以，引用子程序的数目与问题的复杂程度有直接的关系。

图14列出了MODFLOW所定义的子程序包所包含的步骤。在图14中，横列为子程序包的名称，纵列为步骤名称。表格中“X”表示该子程序包具有执行相应步骤的功能。反之，若表格中为空白，则表示该子程序包不具有执行对应步骤的功能。下标“U”表示对应的子程序为工具（Utility）类子程序；而下标“S”表示该子程序仅在相对应的子程序包内起作用。子程序包使用各种工具类子程序来完成某些特定的功能。

对子程序包的命名是根据该子程序包所包含的子程序以及它所完成的任务来进行的。前三个字母表示子程序包的名称；然后是版本号；最后两位字母表示该子程序所具有的功能。

步 骤	地下水运动子程序包								求解子程序包	
	应力子程序包									
	B A S	B C F	W E L	R C H	R I V	D R N	E V T	G H B	S I P	S O R
定义 (DF)	X									
内存分配 (AL)	X	X	X	X	X	X	X	X	X	X
输入处理 (RP)	$X_U$	$X_{US}$							X	X
应力期 (ST)	X									
输入处理 (RP)			X	$X_U$	X	X	$X_U$	X		
时间控制 (AD)	X									
建立方程 (FM)	X	$X_S$	X	X	X	X	X	X		
近似求解 (AP)									$X_S$	$X_S$
输出控制 (OC)	X									
水均衡计算 (BD)		$X_{US}$	$X_U$	$X_U$	$X_U$	$X_U$	$X_U$	$X_U$		
结果输出 (OT)	$X_U$									

图14. MODFLOW主要子程序包分类及其执行步骤。

[译注：由于1988年标准MODFLOW是根据FORTRAN77发展的，故所有子程序的名称限用六个字符。]例如，图14中的井流子程序包中，在内存分配步骤中应用了一个名为WEL1AL的子程序。这个子程序名的前三个字母WEL表示该子程序属于井流子程序包；“1”表示该子程序的版本序号；“AL”是单词“Allocate”的前两个字母，表示该子程序的功能是为井流子程序包所使用的数组和变量分配内存。

图15中列出了MODFLOW中所包括的所有主要子程序的名称。和图14一样，下标中字母“S”表示该子程序为次级子程序；下标“U”表示该子程序为工具类子程序。

次级子程序的名字也为六个字符。第一个字母总为“S”；然后的三个字母表示该子程序所属的子程序包名称；随后一个字符为一数字，用以表示该子程序的版本号；最后一个字母用来区分该子程序。例如，一个名为“SBCF1S”的次级子程序，通过其名称可以知道这个子程序属于BCF子程序包；其版本号为1。工具类子程序的名字以“U”开始，而其余的5个字符用来表示该子程序所具有的功能。例如，“U2DREL”是一个工具类子程序，该子程序的功能是读入一个二维实数数组。

表1中列出了MODFLOW所包括的所有子程序包，以及它们名称的缩写形式（由三个字母表示）。表中对这些子程序包的作用也作了简单的介绍。这些子程序包可以分为两大类：水文地质子程序包和求解子程序包。水文地质子程序包中又包括了一些与外应力有关的子程序包。在这里，水文地质子程序包均用于计算有限差分方程组的系数矩阵。这一类子程序包也包括BCF子程序包；该子程序包用于计算各计算单元之间的地下水渗流量。这一类子程序包还包括外应力子程序包，分别用于模拟不同的外应力对地下水运动的影响。例如，河流子程序包可用来计算地表水体与含水层之间的水力交换。另一大类子程序称为求解子程序包，用于对线性方程组求解。[译注：88年版本MODFLOW中仅含有两种迭代求解方法：超松弛因子法（SOR）和强隐式法（SIP）。后来，由美国地质调查局的Hill发展的PCG2法也成为常用的迭代求解方法，而SOR法较少使用。]除了这两大类子程序包之外，MODFLOW还有一个称为基本子程序包的子程序包，其作用是完成准备整个模拟的基本任务，如模拟时间的划分等。这个基本子程序包也常常简称为“BAS”[译注：BASIC的简称。]子程序包。

在计算各单元之间地下水渗流时，目前版本MODFLOW采用BCF子程序包。这里，BCF为Block-Centered-Flow（计算单元间地下水渗流）的缩写。如用户希望采用其它的计算方法，则可自行编写一些类似的程序来代替本书中的BCF子程序包。例如，使用结点中心法等。同

子程序包

步 骤	BAS	BCF	WEL	RCH	RIV	DRN	EVT	GHB	SIP	SOR
定义 (DF)	BAS1DF									
内存分配 (AL)	BAS1AL	BCF1AL	WEL1AL	RCH1AL	RIV1AL	DRN1AL	EVT1AL	GHB1AL	SIP1AL	SOR1AL
输入处理 (RP)	BAS1RP <sub>U</sub>	BCF1RP <sub>US</sub>							SIP1RP	SOR1RP
应力期 (ST)	BAS1ST									
输入处理 (RP)			WEL1RP	RCH1RP <sub>U</sub>	RIV1RP	DRN1RP	EVT1RP <sub>U</sub>	GHB1RP		
时间控制 (AD)	BAS1AD									
建立方程 (FM)	BAS1FM	BCF1FM <sub>S</sub>	WEL1FM	RCH1FM	RIV1FM	DRN1FM	EVT1FM	GHB1FM		
近似求解 (AP)									SIP1AP <sub>S</sub>	SOR1AP <sub>S</sub>
输出控制 (OC)	BAS1OC									
水均衡计算 (BD)		BCF1BD <sub>US</sub>	WEL1BD <sub>U</sub>	RCH1BD <sub>U</sub>	RIV1BD <sub>U</sub>	DRN1BD <sub>U</sub>	EVT1BD <sub>U</sub>	GHB1BD <sub>U</sub>		
结果输出 (OT)	BAS1OT <sub>U</sub>									

图15. MODFLOW主要子程序包及执行步骤。

表1. MODFLOW子程序包列表

子程序包名称	英文缩写	子程序包功能
基本子程序包	BAS	指定边界条件、时间段长度、初始条件及结果打印方式
计算单元间渗流子程序包	BCF	计算多孔介质中地下水流有限差分方程组各项, 即, 单元间流量和进入贮存的流量
井流子程序包	WEL	将流向水井的流量项加进有限差分方程组
补给子程序包	RCH	将代表面状补给的流量项加进有限差分方程组
河流子程序包	RIV	将流向河流的流量项加进有限差分方程组
沟渠子程序包	DRN	将流向沟渠的流量项加进有限差分方程组
蒸发蒸腾子程序包	EVT	将代表蒸发蒸腾作用的流量项加进有限差分方程组
通用水头边界子程序包	GHB	将流向通用水头边界的流量项加进有限差分方程组
SIP求解子程序包	SIP	采用强隐式方法通过迭代求解有限差分方程组
SSOR求解子程序包	SOR	采用分层逐次超松弛迭代方法求解有限差分方程组

样，用户也可以在程序中添加新的求解子程序包，或其它外应力子程序包。但任何一个数值模型，至少要包括基本子程序包、BCF子程序包和一个求解子程序包。除此之外，用户还应根据实际情况和需要选用相应的外应力子程序包。除了基本子程序包、BCF子程序包和求解子程序包必须使用之外，其它所有的外应力子程序包均为选用。用户也可根据需要开发新的子程序包，并加入MODFLOW之中。本程序各模块在设计时已充分考虑到了个子程序包的独立性。除了上面提到的必需的子程序包之外，是否选用某个子程序包对其它子程序包没有任何影响。用户可以根据自己的实际工作需要改写或重写某些子程序包。如果要希望开发一个新的子程序包，用户可以按各个步骤编写新的模块，并在主程序中适当的位置加入相应的调用语句。

图16给出了MODFLOW主程序的框图。其中包括一些条件判断语句，用来选择适当的子程序包进行模拟计算。读者应同时参考图13、图15和表1，对MODFLOW的总体程序结构有比较深刻的理解。这对使用MODFLOW、改进MODFLOW以及错误排除等都很有帮助。

在程序开始运行时，首先需要计算公式(26)中的各水力传导系数项(CC、CR、CV)。这些将用于计算各计算单元之间的地下水渗流量。它们可能在运算中保持不变，也可能需要重新计算。对于非承压含水层来说，由于这些参数取决于含水层的饱和厚度，而饱和厚度可能在迭代过程中不断发生变化。在目前使用的MODFLOW中，水力传导系数的计算由BCF子程序包完成。当然用户也可以根据需要用新的程序来计算。公式(26)中的CR和CC项，均按相邻两计算单元的水文地质参数和它们的尺寸进行调和平均(harmonic mean)。而公式(26)中垂向水力传导系数则由用户在MODFLOW之外计算，并作为输入数据输入MODFLOW。所有这些参数将存入相应的数组并传送给求解子程序包进行求解运算。

公式(26)中的另外两项， $HCOF_{i,j,k}$ 和 $RHS_{i,j,k}$ 则在每次迭代运算后重新计算。这两项的计算是根据所选用的子程序包逐步进行的。在每次迭代开始前，它们都首先清零。然后调用BCF子程序包在 $HCOF_{i,j,k}$ 中加入一项

$$-SS_{i,j,k} \frac{\Delta r_j \Delta c_i \Delta v_k}{t_m - t_{m-1}}$$

并在 $RHS_{i,j,k}$ 中加入

$$-SS_{i,j,k} \frac{\Delta r_j \Delta c_i \Delta v_k}{t_m - t_{m-1}} h_{i,j,k}^{m-1}$$



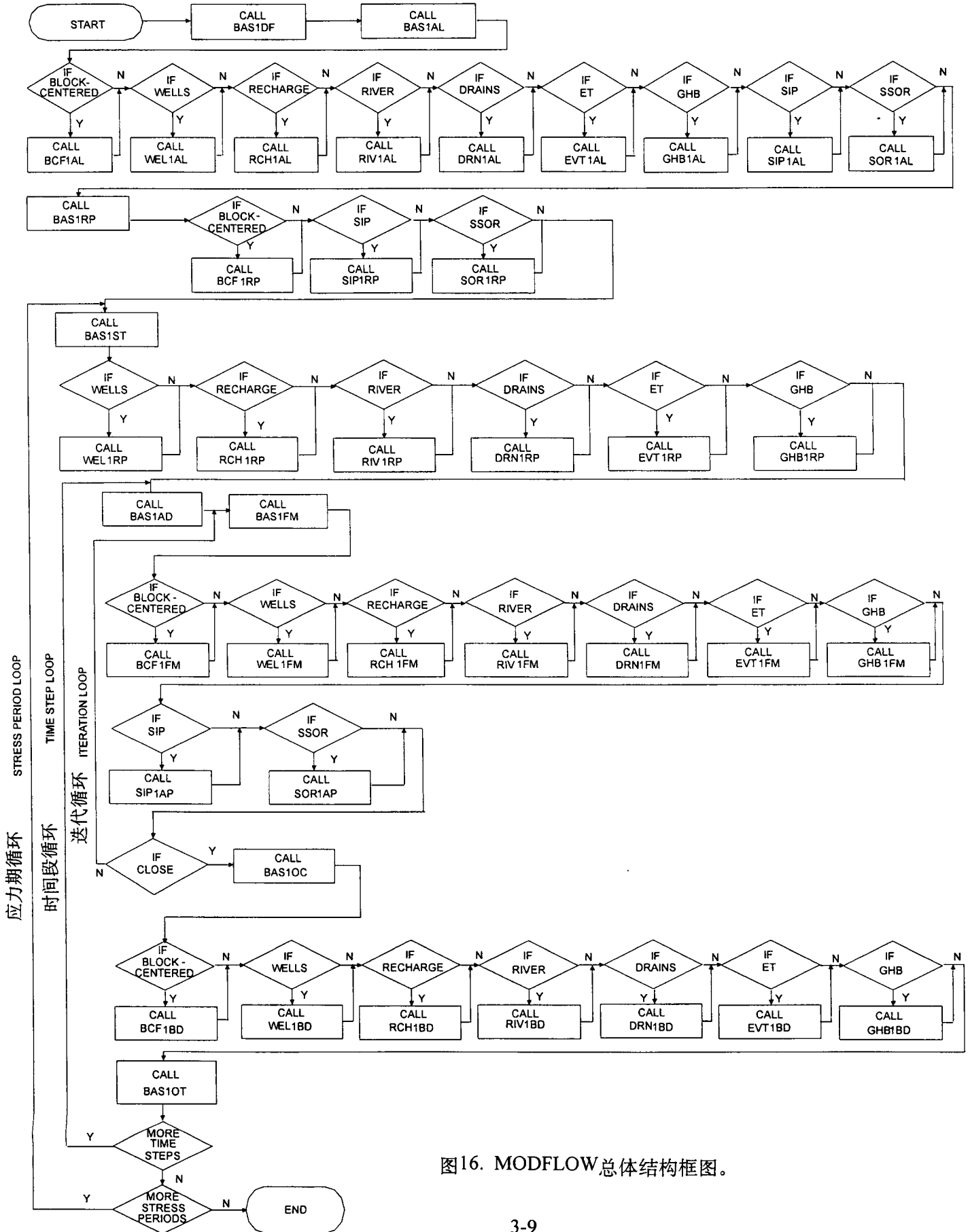


图16. MODFLOW总体结构框图。

若某个计算单元中含有河流段，则由河流子程序包在 $HCOF_{i,j,k}$ 中加上常数 $-P_{i,j,k}$ ；同时在 $RHS_{i,j,k}$ 中加上另一个常数 $-P_{i,j,k}h_s$ 。这里 $h_s$ 为河水水位（同一应力期内为一常数）。对用户所选用的其它外应力子程序包，MODFLOW也依次进行类似的处理。然后，这两个系数数组与三个存贮水力传导系数的数组一同传递给求解子程序包，对整个线性方程组求解。此外，求解时还需要另外两个数组：初始水头数组和边界定义数组。求解子程序包利用 $HCOF$ 的值和三个水力传导系数数组构成公式(26)中的系数矩阵。这样就可以进行迭代运算了。

在MODFLOW中，为便于数据的传递和内存量的估计，这些数组在主程序中均换算为一个一维数组，并称为“X”数组。MODFLOW在读入数据后，将对每个子程序包所使用的内存空间进行计算。X的数组大小预先在程序中设定。当数据量超过X数组的容量时，MODFLOW将打印出相关信息并自动停止运行。此时用户应根据计算机的容量适当增加X数组的容量，并在修改源程序后重新进行编译。[译注：对于FORTRAN77版本，这一步骤有时是相当麻烦的：X数组过大，计算机的物理存贮量不够；如果X数组太小，MODFLOW又无法运行。在FORTRAN90，或FORTRAN77的扩充版中，增加了动态内存分配的功能。用户不必事先定义X数组的大小，而由程序根据计算机的实际物理存贮量进行内存分配。当然，如果模型所包含的数据量太大，超过了计算机的物理存贮量，程序还是无法运行。有兴趣的读者，可以参阅有关编程书籍，对MODFLOW进行修改。]

在这一章的其余部分，我们将介绍边界条件、水均衡计算、内存分配和数据的输入和输出。另外，我们还将对MODFLOW的主程序加以介绍。

### 模型边界和含水层边界

正如上一章中谈过的，一个模型的结构可以看作是一个由许多计算单元组成并按行、列和层排列成的三维网格。这些计算单元的大小与排列方式由用户定义。在设计模型时，用户首先要确定该模型的行数、列数和层数，也就基本上确定了这个模型的大小。当然，随着对问题的认识逐渐深入，有时我们不得不对模型的设计进行必要的修定，例如增加或减少层数等等。

MODFLOW规定，一个模型的六个侧面均为不透水边界，即所模拟的水文地质体系与外界没有任何水力联系。所模拟的体系与外界的水力联系应通过定水头边界、井、降水及蒸发

等加以体现。如果含水层的不透水边界与模型的外边界一致，则模型的边界本身就可以用作含水层的不透水边界。但如果含水层的不透水边界与模型的外边界不一致，用户则必须利用无效计算单元的设置来表示含水层不透水边界的实际位置。对于定水头边界，则可利用定水头计算单元的设置来表示。但地表补给、蒸发蒸腾则应调用相关的子程序包来模拟。

前面说过，对模型中的每个计算单元，我们都可以写出一个类似于公式(26)那样的有限差分方程。对于定水头计算单元，当然不用如此。但对于与定水头计算单元相邻的变水头计算单元，我们需要考虑该单元与定水头计算单元之间的水力联系。对于无效计算单元和不透水计算单元，我们既不用构造相应的差分方程，也不用考虑流进或流出它们的地下水流量。值得注意的是，不透水计算单元与无效计算单元在运算中的效果虽然一样，但它们所代表的意义是不相同的。

出于程序编写的考虑，一个模型总是具有三维立方体的形状。如果含水层的形状为不规则，则模型中可能包含有许多位于含水层边界之外的计算单元。在模型的设计时，可以利用无效计算单元的设置将边界之外的计算单元排除于计算之外。在第二章中，我们已通过例子说明了这一点。在那个例子中，我们用定水头计算单元来代表第一类水文地质边界，利用不透水计算单元与注水井来表示第二类水文地质边界。如果边界条件是水头的函数，可以使用在第11章中介绍的通用水头边界子程序包(GHB)或第6章中介绍的河流子程序包(RIV)来处理。使用GHB和RIV子程序包均要求预先确定水位以及水力传导系数，它们应由实际工作情况来确定。

在MODFLOW中，一个三维整型数组 (IBOUND) 用来定义计算单元的属性：定水头计算单元、无效计算单元（不透水计算单元）和变水头计算单元。该数组中的每一元素与模型中的一个计算单元相对应。对计算单元的规定如下：

定水头计算单元	IBOUND( i, j, k)<0
不透水和无效计算单元	IBOUND( i, j, k)=0
变水头计算单元	IBOUND( i, j, k)>0

这个数组的内容由用户预先准备。在迭代运算过程中，程序仅对变水头计算单元进行求解。值得注意的是，在这三类计算单元中，变水头计算单元的性质可能在计算过程中发生变化。当一个变水头计算单元的导水系数在迭代过程中变为零时，MODFLOW将会把这个单元的属性重新定义为无效计算单元。此外，当一个计算单元中的水头低于该单元底面标高时，

该单元则处于非饱和状态。MODFLOW也会重新定义这个单元为不透水计算单元。这种现象常称为“干枯单元”(dry cell)。它是一个比较难以解决的技术问题。

## 水均衡计算

对于一个确定的空间区域，水均衡指流入和流出该区域的各项地下水渗流量之和。MODFLOW中，水均衡各项的计算均是以体积为单位进行的。严格地讲，这里讲的水均衡并不等于物质平衡，尽管人们常常将物质平衡与体积平衡看作是一回事。由于在我们所讨论的范围内，地下水的密度变化很小，所以质量的均衡可以近似看作为体积的均衡。

MODFLOW在每一个应力期结束时都要进行水均衡的计算，并将其结果打印出来供用户参考。用户可以根据水均衡的情况来检查模拟计算的质量，同时对整个模型的设计、数据输入、程序运行情况有所了解。

对有限差分方程组联立求解所获得的结果并不能保证其正确性。特别是在使用迭代法求解时，迭代过程可能在得到足够接近于收敛指标时就停止了。水均衡是对计算结果的可信度的一个重要测量指标。有限差分方程是依据水流的连续性方程建立起来的，所以流进和流出一个地下水系统的水量总和应能满足连续性原则：总流入量和总流出量之差等于贮水量的变化量。

在MODFLOW中，水均衡的计算与求解无关。所以水均衡可以用作对水头计算结果质量的一种独立检验。在MODFLOW的每个子程序包中，都有一个子程序专门用来计算该子程序包对水均衡的贡献。在输出文件中所列出的水均衡收支表是将模型作为一个整体来计算的。例如，流入和流出所有河流计算单元的总流量，流入和流出所有定水头计算单元的流量等。对于诸如水井流量这样由用户定义的常数，MODFLOW则直接根据输入的数据进行水均衡计算。如果模拟是非稳定流，贮水量的变化也将在水均衡收支表中列出。当贮水量增加时，表明地下水由运动形式转化为存贮形式；而当贮水量减少时，表明地下水由贮存形式转化为运动形式。无论如何变化，贮水量的变化直接反映了地下水不同形式之间的数量转换。

当一个时间段结束时，每个所选用的与地下水渗流有关的子程序包都将计算在这个时间段内该子程序包对整个系统的影响。由各个子程序包计算的各项流入量和流出量，均存入一个称为VBVL的数组。多数子程序包仅仅计算全部水均衡中的一部分，但BCF子程序包需要计

算出定水头计算单元和贮水量变化对水均衡的贡献。除了流量计算之外，在每个时间段结束时，MODFLOW还将由流量和时间的乘积求出流入和流出整个体系的地下水总量。从模拟开始时刻到当前时刻的累积总量也存入数组VBVL之中。

基本子程序包(BAS)中的子程序SBAS1V根据要求，利用VBVL中的流入量、流出量和累积量，将结果输出打印为水均衡收支表。相对于地下水水流系统，贮水量的增加意味着地下水离开水流系统，故被看作为流出量。同理，贮水量的减少标志着地下水水流的增加，因而列为流入量。此外，MODFLOW还将根据总流入量和流出量计算水均衡相对误差。用百分数表示，其计算公式如下：

$$D = \frac{100(IN - OUT)}{(IN + OUT)/2}$$

式中IN为总流入量，OUT为总流出量，D为相对误差(%)。当计算结果比较理想时，相对误差的值应当很小。[译注：虽然无法给出一个绝对指标来衡量计算结果的好坏，但一般来说D的绝对值应当小于1%，而且应当随着模拟时间的增加呈现出下降的趋势。]通常每个应力期结束时流量水均衡情况可以对该应力期内模型的表现进行评估；而累积水均衡量可以作为对整个模拟过程的质量评估的指标。无论用户是否要求，MODFLOW都将在每个应力期结束时打印出水均衡收支表。

在很多情况下，我们希望知道模型中一部分的地下水流入流出量。为了达到这个目的，MODFLOW可以按照要求将流入、流出每个计算单元的地下水渗流量存入一个特别的输出文件。用户可按需要对这个文件进行处理，计算出局部水均衡。在MODFLOW中，计算单元之间的地下水渗流量称为cell-by-cell flow。计算单元间的流量可以有四种形式：(1)由外应力引起的计算单元间流量，即由于外部源汇所引起的流量，如水井注水量，降水补给量等；(2)由于贮水量的变化而引起的计算单元间流量；(3)由于定水头计算单元引起的计算单元间流量；(4)计算单元之间的流量。如果想要MODFLOW将这些信息存入文件，用户必须在输入文件中预先说明。在基本子程序包的输出控制部分中有一个称为ICBCFL的选择项，它控制着整个时间段长结束时是否将计算单元间的流量存盘。另外，在每个与地下水流动有关的子程序包的输入文件中，也有一个类似的选择项，用以决定该子程序包计算的单元间流量是否存盘。例如，在蒸发蒸腾子程序包中的该项如果被选用，则每个时间段内由于蒸发蒸腾所引起的计算单元间流量将输出存盘。当然，与此同时，基本子程序包中的ICBCFL也必须正确选择。是否输出存入文件，则由BCF子程序包输入文件中的IBCFB项单独控制。

值得注意的是, MODFLOW 仅将计算单元间流量和各个外应力引起的流量存入文件。这并不包括水均衡计算所用的水体积或累积体积等信息。另外, MODFLOW 总是假定这些流量的单位与其它输入数据的单位一致。单元间流量以无格式形式存入指定文件, 以节省存贮空间。

某计算单元中的单元源汇渗流指由于外部源汇的存在所引起的流入或流出量。例如, 在计算单元(i,j,k)中, 单元间蒸发流量即为由于蒸发作用所致、从该单元丧失的水量。流入某计算单元的流量为正, 反之为负。如果用户要求, MODFLOW 将把各个外应力子程序包所产生的地下水渗流量写入文件。在这个文件中, 每项外应力子程序包的输出都占有同样的空间, 并按一定的顺序排列。各个外应力子程序的输出都以三维数组的形式写入文件。若某一外应力子程序在计算单元(i,j,k)处的流量为零或不存在, MODFLOW 均以“零”写入输出文件。当模型包含单元较多时, 这个无格式文件可能会占相当多的存贮空间。

在非稳定流的情况下, 计算单元由于贮水量变化而引起的流量记录了模型中各个计算单元的贮水量的变化。由存贮量释出的流量为正, 反之为负。

定水头计算单元中的定水头流量是指进入或离开该单元的流量。这一项总出现于定水头计算单元内部, 而与其它因素无关。若某一个定水头计算单元与六个变水头计算单元相邻, 则定水头流量为该单元与其相邻之变水头计算单元之间的流量之和。当净流量为正数时, 表明地下水从该定水头计算单元流出, 进入模型。而当净流量为负时, 表明地下水由其相邻之变水头计算单元进入该定水头计算单元。例如一个定水头计算单元表示一个具有固定水位的地表水体, 当该单元的流量为正时, 说明含水层获得补给。注意, 当河流、沟渠等外部源汇与一个定水头计算单元相重叠时, 这些源汇的存在并不能改变定水头计算单元的水头值, 故其流量不列入计算。

计算单元之间的流量指通过一个计算单元之六个界面上的流量。按用户要求, 程序还可由BCF子程序包将通过每个变水头计算单元之三个界面上的流量输出存入文件。通过这三个界面上的流量分别称为: 正面流量 (front, 即计算单元i,j,k和i+1,j,k之间的流量), 右面流量 (right, 即单元i,j,k和i,j+1,k之间的流量) 和底面流量 (bottom, 即单元i,j,k和i,j,k+1之间的流量)。另外三个界面上的流量很容易由这三个界面上的流量获得。当地下水沿行、列和层数增加的方向流动时, 其流量为正, 反之为负。在实际工作中, 我们可利用这些单元间流量计算流入或流出模型中一部分区域的地下水流量或用它们建立流场矢量。

理论上, 我们可以根据计算单元之间的流量计算水均衡, 但实际上这并不一定可行。因为在某些情况下, MODFLOW计算水均衡的求和方法有所不同。在计算单元间流量计算时, 同一单元可能包含有同一类型的数项流量。MODFLOW仅将它们累加求和并输出。但在水均衡计算中, 我们则希望将流入量和流出量分别列出。所以用逐个计算单元的流量求和所得到的总流入量和总流出量可能会与水均衡计算所得到的结果有所不同。但由这两种方法得到的总流入量和总流出量应当相同。

### 内存分配

前面已经提到, MODFLOW中使用了一个一维实型数组“X”来存贮程序运行时所使用的实型变量数据。MODFLOW将所有三维实型数组换算为一维数组的形式, 依次存入该X数组。在每个子程序包中, 都有一个专门的子程序为该子程序包所使用的数据进行内存分配。注意, 这里所说的内存分配并不是指计算机存贮器按地址定义的内存分配, 而是指各个数组在X数组中的位置和长度。一维数组X的大小预先在MODFLOW之源程序中确定。实际使用X数组的情况则取决于一个模型的大小和复杂程度。一般来说, X数组至少应为模型中所包含的计算单元数目的10~20倍。通常X数组按计算机的实际存贮能力来确定, 这样用户就不必对每个模型进行源程序修正。在主程序的起始部分, 有两个语句可用来定义或修正X数组的长短。首先定义一个正整数LENX (如令LENX=300,000); 然后定义X数组:

```
COMMON X(LENX)
```

当一个模型所要求的内存量大于LENX时, MODFLOW会打印出错误信息并自动停止运行。此时用户应根据所使用的计算机的容量 (主要指计算机的随机存贮器RAM) 的大小重新定义LENX并重新对MODFLOW进行编译。[译注: 如果用户使用FORTRAN77的扩充版或者FORTRAN90, 则可以将X数组按动态内存分配 (dynamically allocable memory) 的方式进行定义。这在很大程度上解决了估计内存容量的问题。]

### 三维数组下标的定义

在讨论过程中, 我们采用通常的数组下标顺序, 用i,j,k来表示一个计算单元的位置。但从计算单元的角度来看, 这种下标记法并非最优的选择。数组下标的顺序决定了数组在计算机

内存贮的方式。为了提高运算速度，MODFLOW中采用了列—行—层的顺序（即j,i,k）来确定一个计算单元的下标。所以在源程序中，一个计算单元的下标表示为(J,I,K)而不是(I,J,K)。[译注：在阅读MODFLOW之源程序时应记住这一点。特别是在自己编写程序对MODFLOW的结果进行处理时，当心不要把行和列的顺序弄反。]

### 数据输入

MODFLOW在运行时，需要读取输入数据和运行操作指令。实际上，MODFLOW所需要的所有数据和指令均通过输入文件（files）读入。按照FORTRAN的规定，程序通过规定的设备号对所指定的文件进行读和写操作。

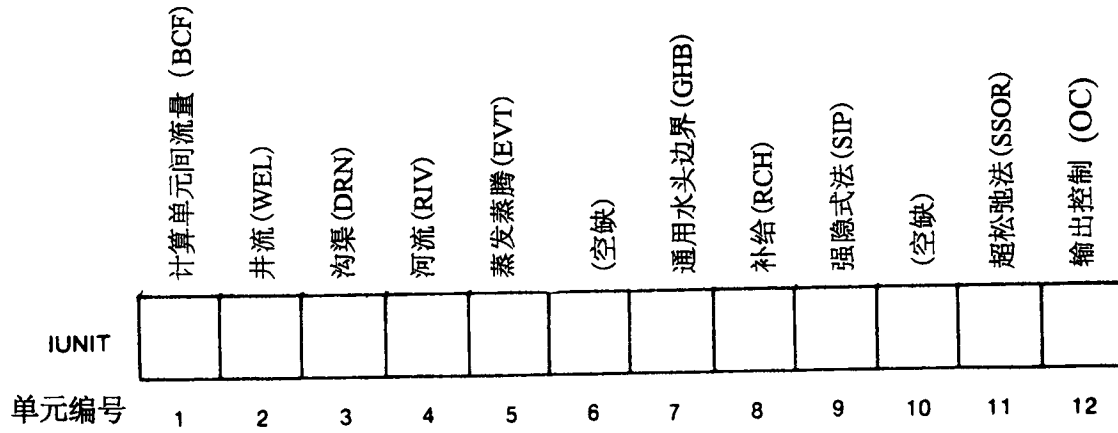
正如前面所介绍的，MODFLOW具有模块结构。用户根据需要选用某些模块或子程序包进行地下水模拟计算。除了基本子程序包之外，所有其它子程序包均可视为可选子程序包。MODFLOW还需要一个输入文件对程序的输出内容进行控制。虽然输出控制部分不是一个独立的子程序包，但它是程序运行不可缺少的一部分。BCF子程序包虽然被看作为可选子程序包，但它也是模拟计算过程中不可缺少的一个子程序包。MODFLOW将它划分为可选子程序包是基于它与地下水渗流计算有关的属性。

数据输入的第一步是定义选用的子程序包。在基本子程序的输入部分有一个一维整型数组，称为IUNIT（参见图17）。该数组中每三位表示一个单元，共有24个单元。[译注：图17仅显示了IUNIT中前12个单元。]用户可以在这个数组的指定位置输入一个正整数表示该单元所对应的子程序包被选用，而且这个正整数表示所规定的输入文件的设备号。MODFLOW将通过这个设备号由所选用的子程序包读取它所需的输入数据和指令。对不选用的子程序包，则在相对应的单元位置上输入“零”。重申一次，每个单元占三位。通过对IUNIT数组的定义，MODFLOW就可知道哪些子程序包被选用了。例如在图18中，用户没有选用沟渠子程序包。故IUNIT中第3单元（第7~9位）的位置上的输入为零。而在第二单元（第4~6位）的输入为23，表明井流子程序包被选用。程序将通过设备号23读入使用井流子程序包所需要的输入文件。

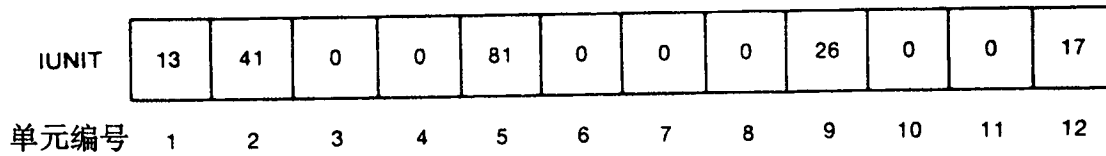
由于基本子程序包不可缺省，故用户必须准备该子程序包所需要的输入文件。在MODFLOW中，读入基本子程序包的输入文件的设备号规定为1。当然，用户可以通过修改源程序对这个设备号重新定义。



各子程序包在数组IUNIT中的位置



示例



- |    |                |               |
|----|----------------|---------------|
| 1  | BCF            | 由第13号设备读入输入文件 |
| 2  | WEL            | 由第41号设备读入输入文件 |
| 3  | DRN            | 未选用           |
| 4  | RIV            | 未选用           |
| 5  | EVT            | 由第81号设备读入输入文件 |
| 7  | GHB            | 未选用           |
| 8  | RCH            | 未选用           |
| 9  | SIP            | 由第26号设备读入输入文件 |
| 11 | SOR            | 未选用           |
| 12 | Output Control | 由第17号设备读入输入文件 |

图17. 各子程序包在数组IUNIT中的位置及用法示例。

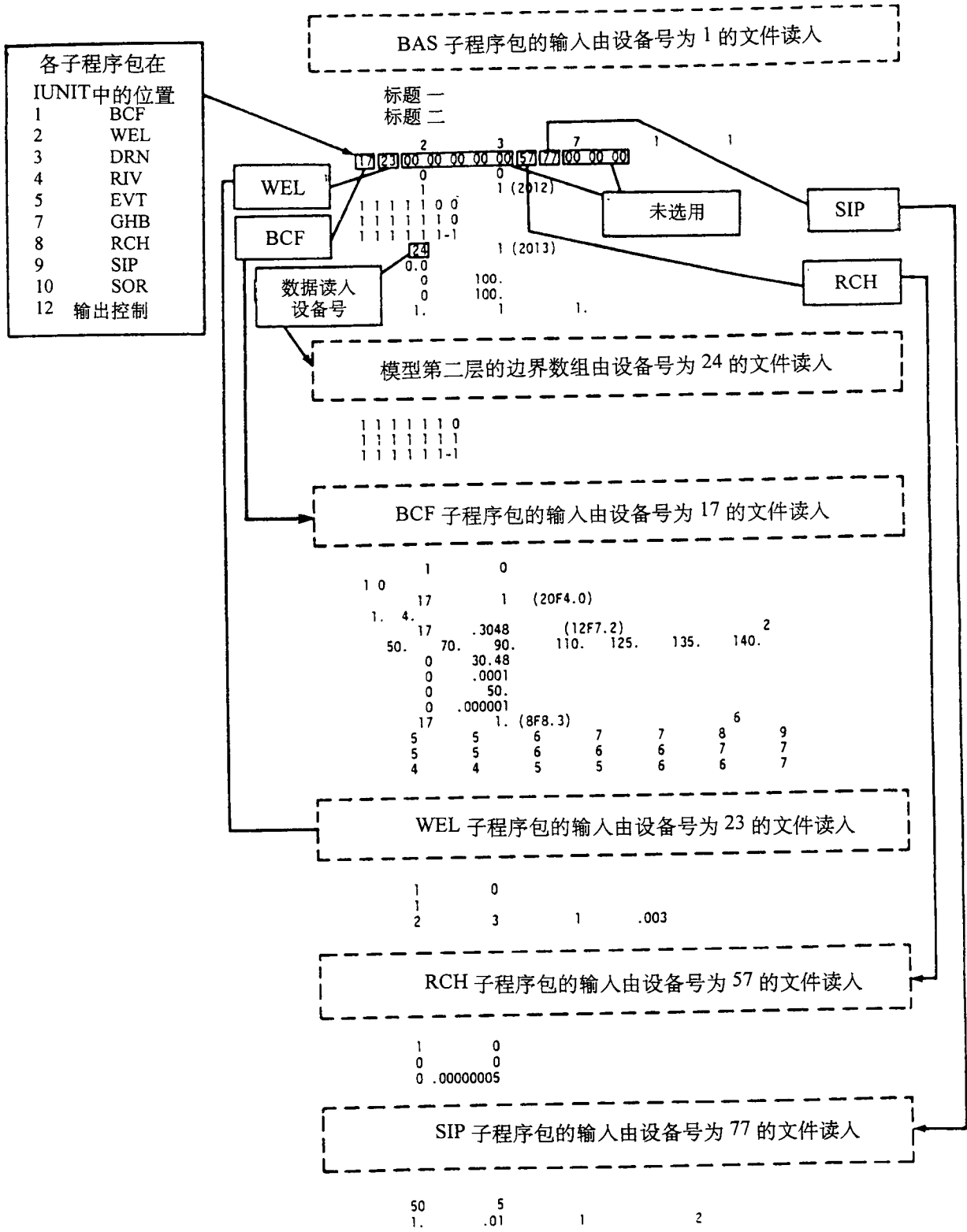


图18. IUNIT的作用示例。

IUNIT数组中的第一单元（第1~3位）必须是用于读入BCF子程序包所需数据的设备号。在现有的MODFLOW中，BCF子程序包用来建立有限差分方程并求计算单元间的地下水渗流量。它是MODFLOW中的一个重要子程序包，实际上也是不可缺省的子程序包之一。

MODFLOW所需要的数据多由一维和二维数组的形式输入。每个数据数组之前必须有一个说明行，用来说明数据的类型、格式等。如果该组数据为常数，则可在这样的说明行中说明，而不必按数组的形式输入。该说明行除表明文件设备号、格式之外，还可能含有是否在输出文件中打印输入数据的指令。在说明行中的文件设备号与IUNIT中的设备号可以相同，也可以不相同。MODFLOW是以IUNIT中的设备号进行文件打开、读入操作的。

MODFLOW的计算可以采用任意时间单位和距离单位。但是，所使用的数据必须在单位上保持一致。例如，当时间单位为天、距离单位为米时，导水系数的单位则必须是米<sup>2</sup>/天。如果将数据的单位搞错，所得到的结果也将毫无意义。

#### 结果输出控制

MODFLOW需要读入一个用来控制结果输出的输入文件。这部分输入数据主要用来控制结果打印及存盘方式、类型和频率。若欲在某个时间段结束时打印或输出计算的水头值，或打印水均衡各项收支情况，则应在这个输入文件中声明。用户还可以指示程序将水头、降深以及计算单元间的流量等结果存盘以用于其它用途。

输出控制是基本子程序的一部分。用户必须在IUNIT数组中相应的位置上（第12个单元）指定读入输出控制文件的设备号。MODFLOW将在每个时间段结束时通过这个文件读入相应的输出要求。如果IUNIT中第12个单元的位置上的输入为零，MODFLOW则会用一个预先设定的输出方式进行结果输出。依照这种预定的输出方式，所计算的水头和水均衡各项将在每个应力期结束时打印出来并存盘。这里所说的打印，并非由打印设备打印结果，而是将所打印内容写入设备号为6的一个标准输出文件（文本文件）。

#### 主程序

主程序是每个FORTRAN程序的主要部分。MODFLOW的主程序有两个功能：(1)对各个子程序包的执行顺序进行控制；(2)作为各个子程序包之间数据的交换场所。当主程序调用各

子程序包时，同时将所需数据以参数形式送入子程序包。

主程序调用子程序包的顺序已在本章图13中所示的框图中标明。在各个步骤中对子程序的调用可以为任意顺序。但如果所调用的子程序是基本子程序中的一部分，则优先调用该子程序。

主程序按顺序完成下列任务：

1. 确定用于存贮数据的X数组的大小，即定义LENX的值；
2. 设定基本子程序包数据文件输入设备号为1，打印文件（标准输出文件）的设备号为6；
3. 定义模型的行数、列数和层数，应力期数和其它一些可选项；
4. 为各个数组在X数组中分配存贮空间；
5. 如果X数组不足以容纳所有数据，则停止运算并打印错误信息；
6. 读入并处理在模拟过程中与时间变量无关的数据；
7. 在每个应力期中：
  - (a) 读入该应力期的长度等信息；
  - (b) 读入并处理随时间变化的有关数据；
  - (c) 在每个时间段内：
    - (1) 计算当前时间段的长度，并将前一个时间段末的水头值存入初始水头数组；
    - (2) 对有限差分方程组进行迭代求解：
      - a. 构造有差分方程组的各项；
      - b. 计算该方程组的近似数值解；
      - c. 如果达到收敛指标，则停止迭代。
    - (3) 确定输出的内容及方式；
    - (4) 计算水均衡各项，并按要求计算通过三个界面的地下水渗流量；
    - (5) 按要求打印和（或）输出计算的水头和降深值。打印水均衡各项以及当前模拟时间；
    - (6) 如果迭代过程达到最大迭代次数尚未收敛，则停止运算。
8. 程序运行结束。

## 第四章 基本子程序包

### 概念及程序化

基本子程序包是MODFLOW的一个必选子程序包。该子程序包读入模型的行数、列数和层数、应力期数和其它一些选择项。该子程序包为一些数组分配存贮空间，并读入初始水头和边界条件。时间控制数据也在此输入。BAS子程序包根据所输入的时间控制参数进行时间离散。BAS子程序包也将计算总水均衡以及输出结果。

### 子程序包的选择以及其输入文件

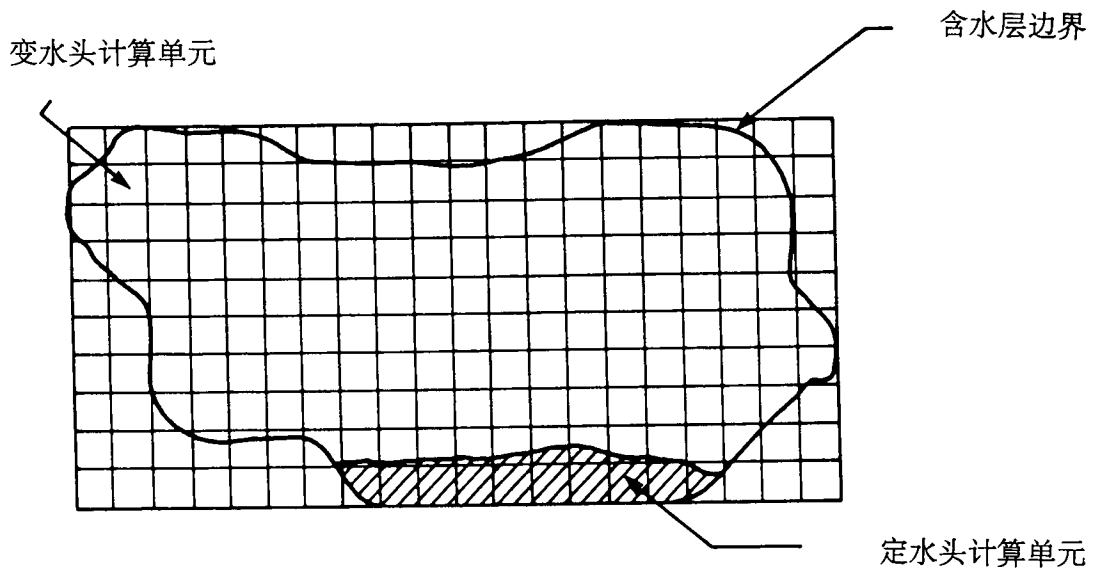
如前面讨论的，MODFLOW通过IUNIT数组来确定在模拟过程调用哪些子程序包，并根据对应的设备号读入这些子程序包运行时所需要的输入文件。MODFLOW在设计上的一大优点就是用户可以根据需要增加新的子程序包。当然，在增加新的子程序包时，应定义该子程序包在IUNIT中的位置，并对主程序进行修改。到目前为止，IUNIT中的单元还没有被全部占用，可供未来增加新的子程序包之用。

### 边界数组

对于模型中每一个变水头计算单元，我们都可以写出下列的有限差分方程：

$$\begin{aligned}
 & CR_{i,j-\frac{1}{2},k}(h_{i,j-1,k}^m - h_{i,j,k}^m) + CR_{i,j+\frac{1}{2},k}(h_{i,j+1,k}^m - h_{i,j,k}^m) + CC_{i-\frac{1}{2},j,k}(h_{i-1,j,k}^m - h_{i,j,k}^m) \\
 & + CC_{i+\frac{1}{2},j,k}(h_{i+1,j,k}^m - h_{i,j,k}^m) + CV_{i,j,k-\frac{1}{2}}(h_{i,j,k-1}^m - h_{i,j,k}^m) + CV_{i,j,k+\frac{1}{2}}(h_{i,j,k+1}^m - h_{i,j,k}^m) \quad (28) \\
 & + P_{i,j,k}h_{i,j,k}^m + Q_{i,j,k} = SC1_{i,j,k}(h_{i,j,k}^m - h_{i,j,k}^{m-1}) / \Delta t_m
 \end{aligned}$$

我们定义一个三维整型数组IBOUND，来区分哪些计算单元为（1）水头随时间发生变化的计算单元（变水头计算单元）；（2）水头为常数（定水头计算单元）；（3）无渗流的计算单元（无效计算单元和不透水计算单元）。这个数组由用户输入，由BAS子程序包读入。值得注意的是，在程序运行过程中，其它子程序可能根据情况重新定义计算单元的属性。在图19所示的例子中，我们可以看到如何利用IBOUND数组来定义水文地质边界条件。



0	1	1	1	1	1	0	0	0	0	0	0	0	1	1	1	1	1	0	0
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0
0	0	0	0	0	0	0	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	0	0	0

IBOUND定义

- <0: 定水头计算单元
- =0: 不透水计算单元
- >0: 变水头计算单元

图19. 利用IBOUND数组表示含水层的边界条件。

### 初始条件

由于公式(28)为隐式差分, 在计算某时间段结束时的水头值时, 需要知道该时间段开始时的水头值。即使是稳定流计算, 程序也需要各个计算单元的起始水头。第一个时间段的初始水头, 或在稳定流计算时的起始水头, 由用户输入。初始水头仅用来计算第一个时间段结束时的水头(参看图20)。如果要求计算降深, 则应在输入的数据中声明, 程序便将初始水头存放在一个称为STRT的数组之中, 并在水头计算完成之后求出各点的降深。

对于稳定流计算来说, 计算结果的精度并不依赖于起始水头。但对于非稳定流计算, 初始水头是描述地下水运动的数学方程的定解条件之一。所以初始条件的精度将直接影响到解的精确度。初始水头应能准确地反映模拟过程起始时的水头分布。但在实际工作中, 要获得过去某个时期的水头分布有时是很困难的。一个变通的方法是先进行稳定流计算, 再利用稳定流计算的结果充当非稳定流计算的初始条件。当然, 要注意哪些外应力(如抽水井等)应当在稳定流计算中予以考虑。

### 时间离散

整个模拟时间可以按要求分为若干个应力期。在每个应力期中, 所有外部源汇的强度应保持不变。在每个应力期内又可再分为若干个时间段(图21), 在同一应力期中, 时间段的长度可以相等, 也可以按一个规定的几何序列逐渐增长。为获得较高的计算精度, 一个应力期初期时的时间段一般较短, 随着系统对外应力变化的逐步调整适应, 故在后期可以使用较长的时间段长。在输入数据中, 用户应当说明每个应力期的长度, 时间段的数目以及时间段长增加因子。时间段长增加因子定义为某个时间段的长度与前一个时间段长度之比, 时间段长增加因子为一个大于1的实数, 如1.2。如果时间段长增加因子为1, 则应力期内各时间段长为常数。根据这三个参数, MODFLOW就可以算出每个时间段的长度, 也就是时间步长。

### 输出

结果输出是任何一个程序的一个重要组成部分。在MODFLOW中, 最重要的输出内容是计算得到的水头值。用户可以在输出控制中说明水头值输出的方式(打印或存盘)、格

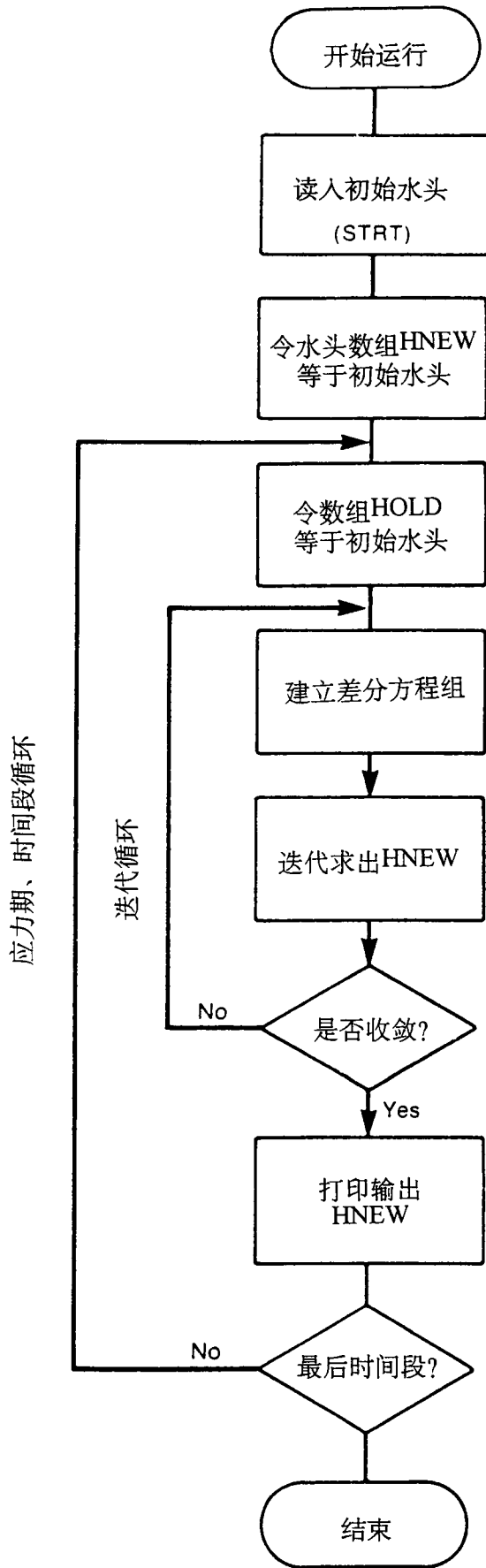


图20. 水头的计算过程示意图。



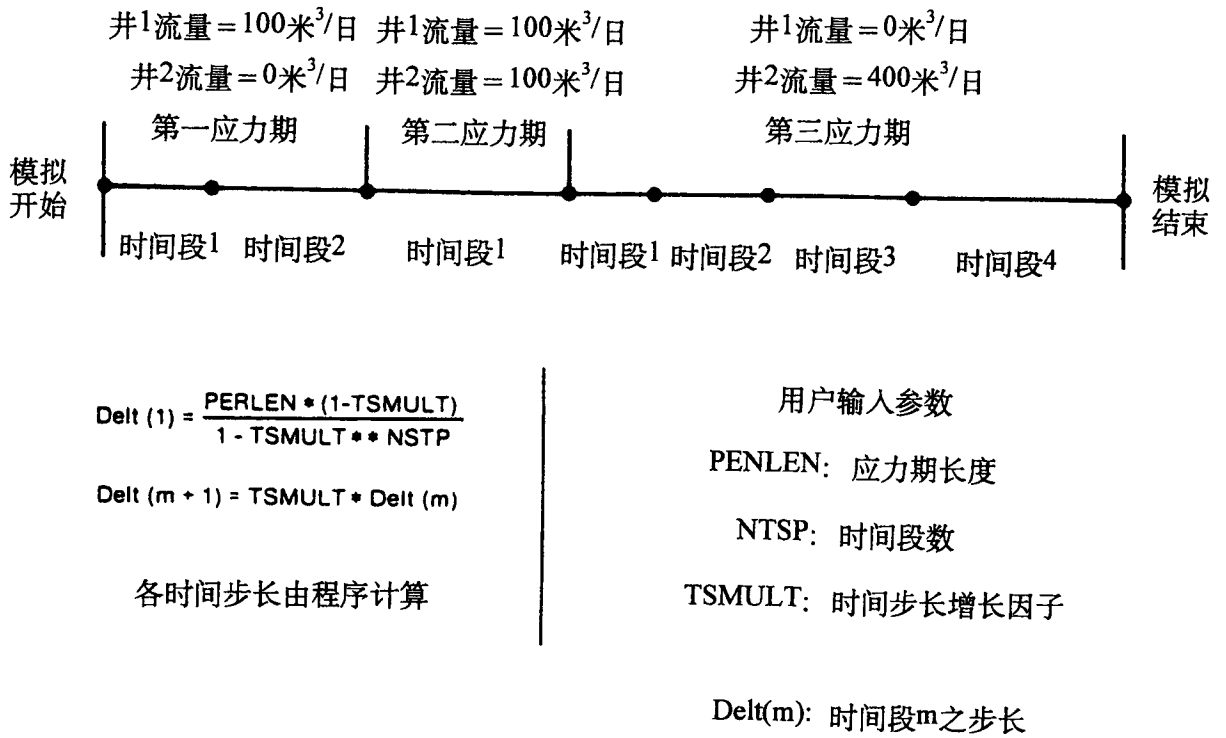


图21. 模拟时间与应力期及时间段的划分。

式以及频率。类似于水头的输出，MODFLOW也可以按要求计算并输出降深和计算单元间的流量、水均衡项等。用户对输出的要求通过一个输出控制文件说明。如果用户忘记定义该输出控制文件，MODFLOW则按预定的规定进行输出：即在每个应力期结束时打印水均衡各项以及水头分布。如果初始水头值已被保存的话，MODFLOW还将打印各点的降深。图22显示了一个MODFLOW水均衡表的例子。

### 水均衡计算

均衡的计算包括两部分：各收支项的计算以及累计收支。每一收支项分别由相应的与地下水流动有关的子程序进行，并存入一个一维数组VBVL中。最后由BAS子程序包根据VBVL进行整体水均衡计算并输出计算结果。

```

VOLUMETRIC BUDGET FOR ENTIRE MODEL AT END OF TIME STEP 1 IN STRESS PERIOD 1
-----
CUMULATIVE VOLUMES      L**3
-----
IN:
---
STORAGE = .0
CONSTANT HEAD = .0
WELLS = .0
DRAINS = .0
RECHARGE = .13608E+08
TOTAL IN = .13608E+08
OUT:
----
STORAGE = .0
CONSTANT HEAD = .43265E+07
WELLS = .64800E+07
DRAINS = .28010E+07
RECHARGE = .0
TOTAL OUT = .13607E+08
IN - OUT = 303.00
PERCENT DISCREPANCY = 0.00

-----
RATES FOR THIS TIME STEP      L**3/T
-----
IN:
---
STORAGE = .0
CONSTANT HEAD = .0
WELLS = .0
DRAINS = .0
RECHARGE = 157.50
TOTAL IN = 157.50
OUT:
----
STORAGE = .0
CONSTANT HEAD = 50.075
WELLS = 75.000
DRAINS = 32.419
RECHARGE = .0
TOTAL OUT = 157.49
IN - OUT = .34943E-02
PERCENT DISCREPANCY = 0.00
    
```

图22. MODFLOW水均衡表实例。

### 基本子程序包输入数据及格式

基本子程序包 (BAS) 的输入数据包括两部分: 输出控制和其它输入项。输出控制文件按规定从 IUNIT (12) 所定义的设备号<sup>12</sup>读入, 而其它输入项则由设备号为<sup>1</sup>的文件中读入。如有必要, 用户可以改变这一设备号以满足主程序中的定义。

基本子程序包要求的输入数据及格式按读入顺序定义如下:

对整个模拟过程:

由子程序BAS1DF读入的数据包括:

1. 数据名称: HEADNG(32)  
输入格式: 20A4
2. 数据名称: HEADNG  
输入格式: 12A4
3. 数据名称: NLAY NROW NCOL NPER ITMUNI  
输入格式: I10 I10 I10 I10 I10
4. 数据名称: IUNIT(24)  
输入格式: 24I3

由子程序BAS1AL读入的数据包括

5. 数据名称: IAPART ISTRT  
输入格式: I10 I10

由子程序BAS1RP读入的数据包括

6. 数据名称: IBOUND (NCOL, NROW)  
读入子程序: U2DINT  
(每层为一个二维实型数组, 重复NLAY次)
7. 数据名称: HNOFLO

输入格式: F10.0

8. 数据名称: SHEAD(NCOL, NROW)

读入子程序: U2DREL

(每层为一个二维实型数组, 重复NLAY次)

注意: IBOUND和SHEAD虽在程序中为三维数组, 但输入时均应按一系列二维数组逐层进行。

对每个应力期:

由子程序BAS1ST读入的数据包括

9. 数据名称: PERLEN NSTP TSMULT

输入格式: F10.0 I10 F10.0

### 输入数据说明

HEADNG: 为模型的名称。可用来说明模型的项目、目的、时间、作者等内容。该名  
称共占两行, 最多可记录128个字符。第一行80个, 第二行48个。  
如果没有定义名称, 则应留出两空白行。

NLAY: 模型的层数。

NROW: 模型的行数。

NCOL: 模型的列数。

NPER: 应力期数。

ITMUNI: 时间单位的标示符。时间单位仅供打印输出时使用, 与计算过程无关。时  
间单位的定义如下:

0: 无定义	3: 小时
1: 秒	4: 天
2: 分钟	5: 年

值得注意的是, 时间单位的定义虽然与计算结果无关, 但用户在准备  
各项含有时间的数据(如导水系数, 抽水量等)时, 应注意使用一致的时间  
单位, 并在此定义所使用的时间单位。例如, 如果渗透系数的单位为

米/天，则抽水量的单位为米<sup>3</sup>/天。在这里也应选用“4”（天）作为时间单位。

IUNIT: 为一个一维整型数组。该数组按每三位为一单元，共含有24个单元。其中10个单元已有了定义（1~5，7~9，和11~12）。其余的14个单元留供以后增加新子程序包使用。

单元序号	子程序包名称
1	BCF子程序包(BCF)
2	井流子程序包(WEL)
3	沟渠子程序包(DRN)
4	河流子程序包(RIV)
5	蒸发蒸腾子程序包(EVT)
6	(未用)
7	通用水头边界子程序包(GHB)
8	补给子程序包(RCH)
9	SIP求解子程序包(SIP)
10	(未用)
11	超松弛因子法求解子程序包(SSOR)
12	输出控制(OPC或OC)

对子程序包进行选用的规定如下：

如果 IUNIT(n)≤0，该子程序包不被选用；

如果

IUNIT(n)>0，选用第n项子程序包。在IUNIT数组中输入的文件设备号应在1—99之间。MODFLOW将用输入的正整数作为设备号打开并读入输入文件。虽然设备号可以重复但最好选用不同的数。数字6已用作MODFLOW标准打印输出文件之通道号，故不能再次选用。基本子程序按规定使用“1”作为设备号，所以不应选用“1”作为其它子程序的设备号。

IAPART: 为标示符。用于说明操作数组BUFF是否与数组RHS共享内存。

IAPART=0: BUFF和RHS两数组共享内存；

IAPART≠0: MODFLOW为数组BUFF和RHS分别开辟内存空间。

ISTRTR: 用以说明初始水头是否保存的标识符。如果希望程序计算降深, 用户则应当指示程序保存初始水头。

ISTRTR=0: 不保存初始水头;

ISTRTR≠0: 保存初始水头。

IBOUND: 边界数组。

IBOUND(i, j, k)<0: 定水头计算单元;

IBOUND(i, j, k)=0: 无效或不透水计算单元;

IBOUND(i, j, k)>0: 变水头计算单元。

注意三维数组的输入方式。在MODFLOW中, 一个三维数组是作为一系列二维数组输入的。每个二维数组代表一层的输入值。在每个二维数组之前, 应有一个格式行用来说明数据的格式等内容。

HNOFLO: 无效或不透水计算单元的水头值。尽管MODFLOW并不计算无效或不透水计算单元的水头, 但仍需要一个数值来表示该点的水头值。一般HNOFLO的值应与其它的水头值有明显的区别。如-999.0等。在初始水头的输入数据中, 用户并不将无效或不透水计算单元处的水头值用HNOFLO来代替。MODFLOW将自动进行这种替代, 并在结果输出时使用HNOFLO来表示无效或不透水计算单元。

SHEAD: 初始水头数组。不论模拟是稳定流还是非稳定流计算, 用户都必须提供这个数组。输入方式与边界数组的输入方式相同。

PERLEN: 应力期长度。

NSTP: 该应力期内时间段的个数。

TSMULT: 时间步长增加因子。如果欲使同一应力期内各时间段的长度相等, 则此处的输入值应为1.0。否则, 第一个时间段的长度按下式计算:

$$\text{DELTA}(1) = \text{PERLEN} / (1 - \text{TSMULT}^{\text{NSTP}})$$

其它的时间段长则为:

$$\text{DELTA}(n) = \text{DELTA}(1) * \text{TSMULT}^{n-1}$$

注意, PERLEN, NSTP和TSMULT三个数据在同一行中输入。每一行对应于一个应力期。若整个模拟过程分为M个应力期, 则应有M行的输入数据。每一行中的PERLEN, NSTP和TSMULT可以相同, 也可以不同, 取决于模型的设计。

BAS子程序包输入样单

数据项	说明	输入记录
1	{HEADNG} 标题1	BAS子程序包输入样单
2	{HEADNG} 标题2	选用RECHARGE, DRAINS, WELLS
3	{NALY,NROW,NCOL,NPER,ITMUNI}	3 10 12 2 1
4	{IUNIT}	31 12 14 00 00 00 00 28 19 00 00 22
5	{IAPART,ISTRRT}	0 1
6	{IBOUND数组控制记录}	1 1 (12I2) 3
		-11111111111111
		-11111111111111
		-11111111111111
		-11111111111111
	边界数组第一层	-11111111111111
		-11111111111111
		-11111111111111
		-11111111111111
		-11111111111111
		-11111111111111
		-11111111111111
6	{IBOUND数组控制记录}	1 1 (12I2) 3
		-22222222222222
		-22222222222222
		-22222222222222
		-22222222222222
	边界数组第二层	-22222222222222
		-22222222222222
		-22222222222222
		-22222222222222
		-22222222222222
		-22222222222222
		-22222222222222
6	{IBOUND数组控制记录}	0 3
7	{HNOFLO} 无效单元水头	999.9



8	{SHEAD数组控制记录}	1	1 (12F4.1)	3
		1.0	1.7 2.4 3.1 3.8 4.5 5.2 5.9 6.6 7.3 8.0 8.7	
		1.1	1.9 2.7 3.5 4.3 5.1 5.9 6.7 7.5 8.3 9.1 9.9	
		1.1	1.9 2.7 3.5 4.3 5.1 5.9 6.7 7.5 8.3 9.1 9.9	
		1.1	1.9 2.7 3.5 4.3 5.1 5.9 6.7 7.5 8.3 9.1 9.9	
	初始水头数组第一层	1.3	2.3 3.3 4.3 5.3 6.3 7.3 8.3 8.4 8.6 9.3 9.9	
		1.3	2.3 3.3 4.3 5.3 6.3 7.3 8.3 8.4 8.6 9.3 9.9	
		1.3	2.3 3.3 4.3 5.3 6.3 7.3 8.3 8.4 8.6 9.3 9.9	
		1.1	1.9 2.7 3.5 4.3 5.1 5.9 6.7 7.5 8.3 9.1 9.9	
		1.0	1.7 2.4 3.1 3.8 4.5 5.2 5.9 6.6 7.3 8.0 8.7	
		1.0	1.7 2.4 3.1 3.8 4.5 5.2 5.9 6.6 7.3 8.0 8.7	
8	{SHEAD数组控制记录}	23	1 (12F6.2)	5
8	{SHEAD数组控制记录}	0	0.0	
9	{PERLEN,NSTP,TSMULT}应力期1	86400	5	1.5
9	{PERLEN,NSTP,TSMULT}应力期2	172800	7	1.3

[译注：样单中的数据可能不符合格式要求，仅供参考。]

### 输出控制部分的输入数据

在MODFLOW中，虽然输出控制被当做BAS子程序的一部分，但用于进行输出控制的参数由一个单独的输入文件，通过IUNIT (12) 所定义的设备号12输给程序。在这个文件中，用户应根据需要对MODFLOW的结果输出的内容、方式及频率进行说明。如果用户不提供这一文件，MODFLOW就将按程序的预先设置进行输出操作。程序的预先设置为：在每个应力期结束时打印水头值及水均衡收支表。如果初始水头已按用户的要求存入了数组STRT，MODFLOW则同时打印出降深。另外，MODFLOW还规定了13种输出格式。程序预定输出的格式为10G11.4。注意，此处所讲的打印并非由打印设备打印，而是将打印内容存入MODFLOW运行时产生的标准输出文件之中。下面我们来介绍MODFLOW输出控制的要求。

对于整个模拟过程：

由子程序BAS1RP读入的数据包括

1. 数据名称:	IHEDFM	IDDNFM	IHEDUN	IDDNUN
输入格式:	I10	I10	I10	I10

对于每个时段：

由子程序BAS1OC读入的数据包括

2. 数据名称:	INCODE	IHDDFL	IBUDFL	ICBCFL
输入格式:	I10	I10	I10	I10
3. 数据名称:	Hdpr	Ddpr	Hdsv	Ddsv
输入格式:	I10	I10	I10	I10

第三行输入的情况由第二行中的INCODE决定。请参看注解中INCODE一项。

## 输入数据说明

IHEDFM: 水头打印格式。MODFLOW规定了13种输出格式,并由数字0-12分别表示。它们是:

0—(10G11.4)	7—(20F5.0)
1—(11G10.3)	8—(20F5.1)
2—(9G13.6)	9—(20F5.2)
3—(15F7.1)	10—(20F5.3)
4—(15F7.2)	11—(20F5.4)
5—(15F7.3)	12—(10G11.4)
6—(15F7.4)	

关于这些格式的意义,读者可以参阅有关FORTRAN语言的书籍。

IDDNFM: 降深打印格式。

IHEDUN: 将水头存入无格式输出文件的设备号。

IDDNUN: 将降深存入无格式输出文件的设备号。

INCODE: 水头/降深输出方式。它的值决定着第三行输入数据的数量:

NCODE<0: 略去第三行的输入数据,并按前一时间段中所定义的方式进行输出操作;

INCODE=0: 输出时,各个层数据的输出方式相同,第三行输入数据仅输入一次;

INCODE>0: 逐层输入第三行输入数据。

IHDDFL: 水头和降深输出标识符:

IHDDFL=0: 不打印存盘;

IHDDFL≠0: 打印水头和降深,并依据第三行输入数据的情况,决定如何将水头和降深存盘。

IBUDFL: 水均衡收支表打印指示符:

IBUDFL=0: 不打印总水均衡收支表;

IBUDFL≠0: 打印总水均衡收支表。

(注意,这里仅指是否在每个时间段结束时打印水均衡收支表。MODFLOW在每个应力期结束时会自动打印总水均衡收支表。)

ICBCFL: 计算单元间流量输出指示符:

ICBCFL=0: 不进行计算单元间流量项的打印和存写;

ICBCFL≠0: 打印或存写计算单元间流量项。对于各应力子程序包所计算的水均衡各项分量是否打印存盘, 还要视这些子程序包中的流量输出指示符而定, 如IWELCB、IRCHCB等。

Hdpr: 打印水头指示符:

Hdpr=0: 不打印该层水头;

Hdpr≠0: 打印该层水头。

Ddpr: 打印降深指示符:

Ddpr=0: 不打印该层降深;

Ddpr≠0: 打印该层降深。

Hdsv: 水头存盘指示符:

Hdsv=0: 不将该层水头存盘;

Hdsv≠0: 将该层水头存盘。

Ddsv: 降深存盘指示符:

Ddsv=0: 不将该层降深存盘;

Ddsv≠0: 将该层降深存盘。

输出控制选择输入样单

数据项	说明	输入记录			
1	{IHEDFM,IDDNFM,IHEDUN,IDDNUN}	4	0	76	77
2	{INCODE,IHDDFL,IBUDFL,ICBCFL}时间段1	1	1	0	0
3	{HDPR,DDPR,HDSV,DDSV}第1层	1	1	1	1
3	{HDPR,DDPR,HDSV,DDSV}第2层	1	1	0	0
3	{HDPR,DDPR,HDSV,DDSV}第3层	1	1	0	0
2	{INCODE,IHDDFL,IBUDFL,ICBCFL}时间段2	-1	0	1	0
2	{INCODE,IHDDFL,IBUDFL,ICBCFL}时间段3	-1	1	1	0
2	{INCODE,IHDDFL,IBUDFL,ICBCFL}时间段4	-1	0	1	0
2	{INCODE,IHDDFL,IBUDFL,ICBCFL}时间段5	-1	0	1	0
2	{INCODE,IHDDFL,IBUDFL,ICBCFL}时间段6	0	1	1	0
3	{HDPR,DDPR,HDSV,DDSV}所有层	1	1	0	0
2	{INCODE,IHDDFL,IBUDFL,ICBCFL}时间段7	-1	0	1	0
2	{INCODE,IHDDFL,IBUDFL,ICBCFL}时间段8	-1	1	1	0
2	{INCODE,IHDDFL,IBUDFL,ICBCFL}时间段9	-1	0	1	0
2	{INCODE,IHDDFL,IBUDFL,ICBCFL}时间段10	-1	0	1	0
2	{INCODE,IHDDFL,IBUDFL,ICBCFL}时间段11	-1	0	1	0
2	{INCODE,IHDDFL,IBUDFL,ICBCFL}时间段12	-1	1	1	0
2	{INCODE,IHDDFL,IBUDFL,ICBCFL}时间段13	-1	0	1	0

[译注：样单中的数据可能不符合格式要求，仅供参考。]

## 第五章 计算单元间渗流子程序包

## 概念及程序化

在MODFLOW中, 计算单元间渗流(简称BCF)子程序包用于计算相邻计算单元之间的水力传导系数以及计算单元之间的地下水渗流量。它也用于计算含水层由于贮水量的变化所吸收或释放的水量。在MODFLOW中, 我们总是用计算单元的中心点来表示该单元的空间位置, 故计算单元间的流量事实上相当于两相邻计算单元中心点之间的流量。

在第二章中, 我们已经根据地下水流运动的连续性方程推出了对计算单元(i,j,k)水头求解的有限差分公式:

$$\begin{aligned}
 & CV_{i,j,k-\frac{1}{2}} h_{i,j,k-1} + CC_{i-\frac{1}{2},j,k} h_{i-1,j,k} + CR_{i,j-\frac{1}{2},k} h_{i,j-1,k} + (-CV_{i,j,k-\frac{1}{2}} - CC_{i-\frac{1}{2},j,k} \\
 & - CR_{i,j-\frac{1}{2},k} - CR_{i,j+\frac{1}{2},k} - CC_{i+\frac{1}{2},j,k} - CV_{i,j,k+\frac{1}{2}} + HCOF_{i,j,k}) h_{i,j,k} \\
 & + CR_{i,j+\frac{1}{2},k} h_{i,j+1,k} + CC_{i+\frac{1}{2},j,k} h_{i+1,j,k} + CV_{i,j,k+\frac{1}{2}} h_{i,j,k+1} = RHS_{i,j,k}
 \end{aligned} \quad (29)$$

其中 $CV$ ,  $CR$ 和 $CC$ 均为计算单元之间的水力传导系数。 $HCOF$ 和 $RHS$ 分别包含源汇项以及贮水项。除了计算这些水力传导系数和贮水项之外, 这个子程序包还对由于悬挂含水层所引起的非饱和带渗流计算进行纠正。当悬挂含水层下面含水层的水头低于悬挂含水层之底板高程时, 其下伏的含水层的上部则处于非饱和状态。这时垂向上的渗流量不再与两个含水层之间的水头差成正比, 而是一个常数。在这种情况下, 用公式(29)所得到的垂向流量必须进行纠正才能更为真实地反映两个含水层之间的地下水流动。[译注: 应当说明, 在这里所讲的非饱和带流量与著名的Richard方程无关。我们知道, 地下水通过非饱和带或包气带的流动远比地下水通过饱和的孔隙介质的流动复杂。在非饱和带中, 介质的渗透系数为土壤的含水率的函数, 而这种非线性函数一般只能通过大量的实验分析才能确定。此外, 对滞后现象的处理, 也是非饱和带数值计算中相当难以解决的问题。在MODFLOW中, 非饱和带水流的计算实际上假定介质的非饱和带渗透系数与饱和带相同。此外, 滞后现象也忽略不计。]

## 水力传导系数的计算

在第二章中, 我们已经介绍了水力传导系数的概念(参见公式(9))。在这一章中, 我们将具体导出不同条件下水力传导系数的计算公式。

实际上,水力传导系数不过是达西(Darcy)公式中几个常数项的合并。例如,在一维流的情况下,通过如图23所示的孔隙介质的地下水渗流量为:

$$Q = KA \frac{(h_2 - h_1)}{L} \quad (30)$$

式中:  $Q$ : 流量 ( $L^3T^{-1}$ );  
 $K$ : 沿地下水流动方向的渗透系数 ( $LT^{-1}$ );  
 $A$ : 与水流方向垂直的横断面面积 ( $L^2$ );  
 $h_2-h_1$ : 流场两端的水头差 ( $L$ );  
 $L$ : 所研究流场两端间的距离 ( $L$ )。

如果我们将(30)式中几个常数项合并在一起,并称之为水力传导系数,  $C$ , 则有

$$C = \frac{KA}{L} \quad (31)$$

这样的话,公式(30)就可以简化为:

$$Q = C(h_2 - h_1) \quad (32)$$

如果使用导水系数(transmissivity)来表示水力传导系数,则有:

$$C = \frac{TW}{L} \quad (33)$$

其中,  $T$  ( $L^2T^{-1}$ )为导水系数,  $W$ 为所研究流场断面的宽度( $L$ )。

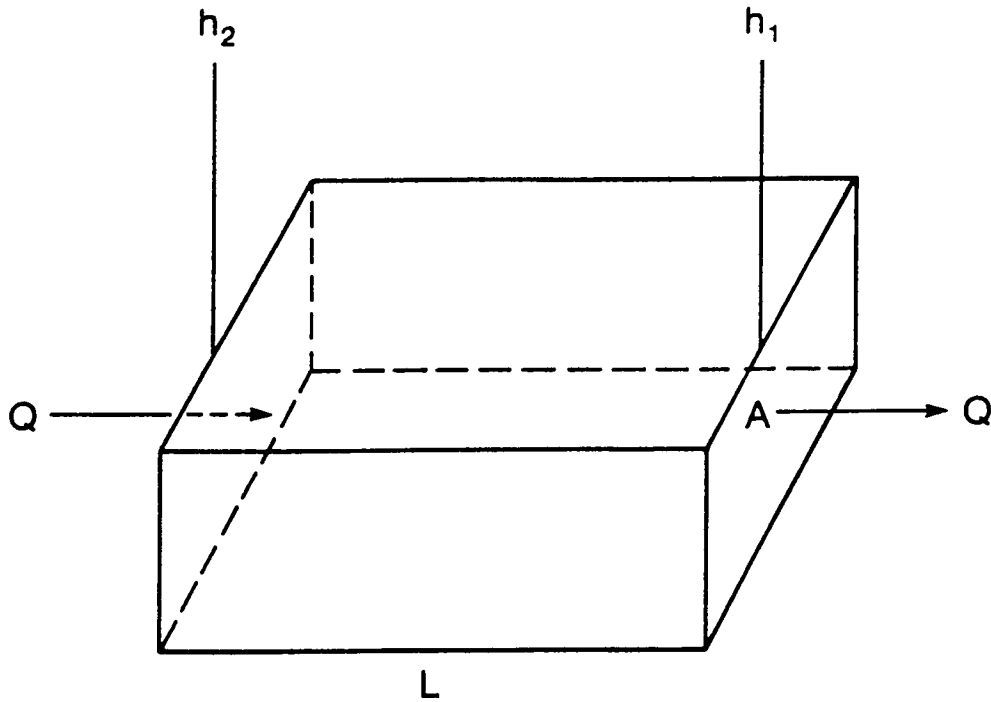
水力传导系数与所考虑的物质透水性以及流动的方向有关。对于各向异性的孔隙介质,在各个方向上的水力传导系数通常也不相同。严格来说,水力传导系数应用二阶张量(tensor)来表示。此外,水力传导系数与过水断面面积成正比,与流线长度成反比。

在实际工作中,我们常可以见到地下水沿水流方向垂直穿过多层具有不同透水能力的岩层的情况。我们也可以用串联计算的方法,求出这些岩层的综合水力传导系数(参见图24)。而流量则可根据综合水力传导系数以及两端之间的总水头差求出:

$$C = \frac{Q}{h_A - h_B} \quad (34)$$

显然,各单层引起的水头损失之和等于总体水头损失:

$$Q = \frac{KA (h_2 - h_1)}{L}$$

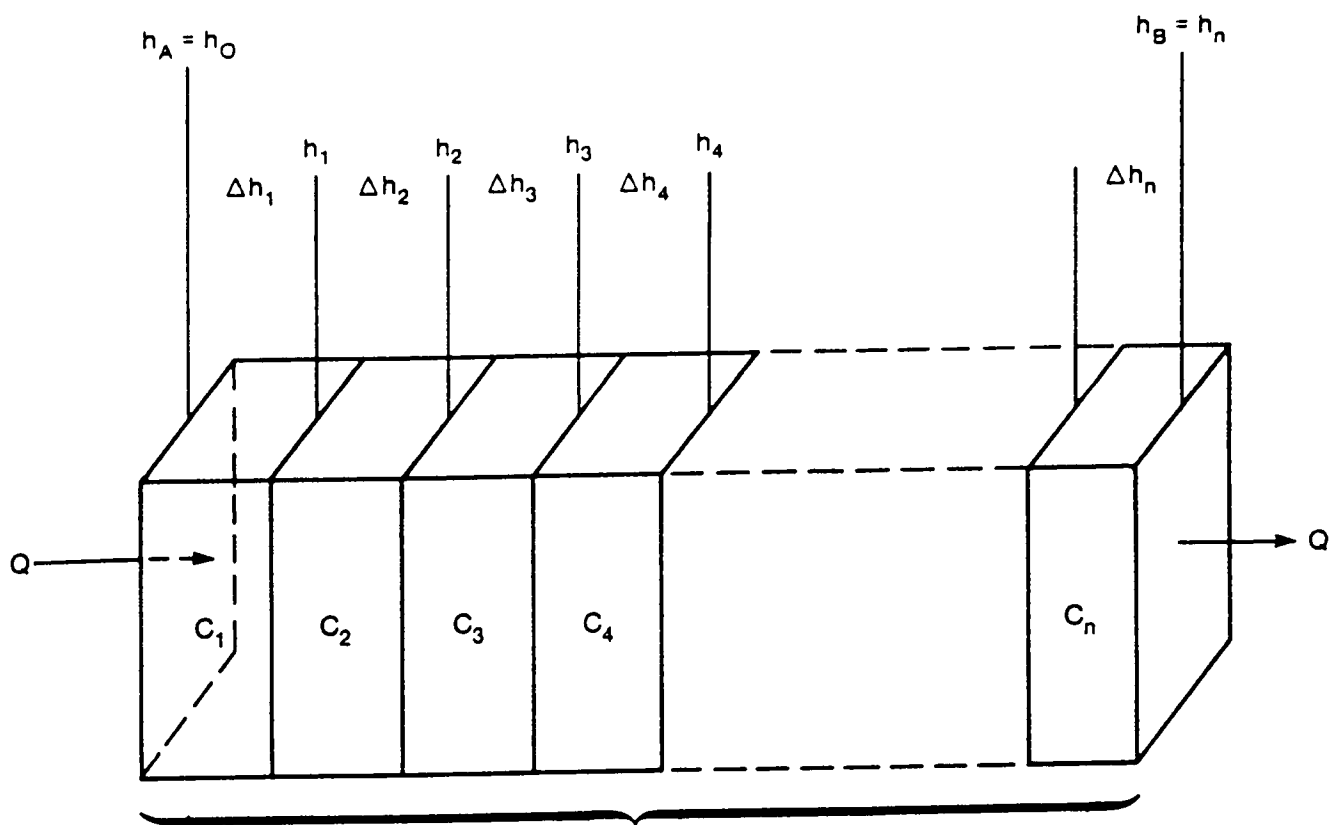


说 明

- K: 渗透系数
- $h_1$ : 右端水头
- $h_2$ : 左端水头
- Q: 自左至右渗流量
- L: 长度
- A: 过水断面面积

图23. 达西流量计算示意图。





$$\frac{1}{C} = \frac{1}{C_1} + \frac{1}{C_2} + \frac{1}{C_3} + \dots + \frac{1}{C_n}$$

### 说明

- Q: 流量  
 \$C\_m\$: 第\$m\$层物质之水力传导系数  
 \$h\_m\$: 第\$m\$层之水头  
 \$\Delta h\_m\$: 第\$m\$层之水头损失  
 C: 总水力传导系数

图24. 串联条件下的总水力传导系数。

$$\sum_{i=1}^n \Delta h_i = h_A - h_B \quad (35)$$

将(35)式代入(32)式, 则有

$$\sum_{i=1}^n \frac{q_i}{C_i} = h_A - h_B \quad (36)$$

在这里, 我们考虑地下水为一维流动。如果不考虑贮水量的变化, 则通过各个单层的渗流量与总流量 $Q$ 相等, 则:

$$Q \sum_{i=1}^n \frac{1}{C_i} = h_A - h_B \quad \text{或}$$

$$\sum_{i=1}^n \frac{1}{C_i} = \frac{h_A - h_B}{Q} \quad (37)$$

将(37)式与(34)式相比, 我们可以看出:

$$\frac{1}{C} = \sum_{i=1}^n \frac{1}{C_i} \quad (38)$$

所以, 对于一系列以串联方式排列的透水岩层来说, 综合水力传导系数的倒数等于各单层之水力传导系数的倒数之和。在只有两个单层相串联排列时, 其综合水力传导系数的计算公式为:

$$C = \frac{C_1 C_2}{C_1 + C_2} \quad (39)$$

### 承压含水层中水平方向上水力传导系数的计算

当地下水从一个计算单元流向一个相邻的计算单元时, 如果这两个计算单元的透水性不同, 地下水将穿过两层串联排列的岩层。所以两个计算单元间的总水力传导系数可以由(39)式计算。当然, 由于水流将流经这两个相邻单元的各一半距离, 而且这两个单元的大小也许不同, 故实际计算公式要比(39)式复杂一些。公式(29)中的 $CR$ 和 $CC$ 的计算公式由图25示意。 $CC$ 和 $CR$ 分别指沿行方向和列方向上相邻两计算单元之间的水力传导系数。为了明确起见, 这些水力传导系数的下标用 $\frac{1}{2}$ 表示。例如,  $CR_{i,j+\frac{1}{2},k}$ 表示计算单元 $(i,j,k)$ 和计

算单元(i, j+1, k)之间的水力传导系数。

在计算水力传导系数时, 我们规定: (1)计算单元的位置以该计算单元的中心点表示; (2)每个计算单元内部的导水系数相同。两个单元间的水力传导系数应包括这两个单元相邻侧的各一半。根据公式(39), 可得:

$$CR_{i,j+\frac{1}{2},k} = \frac{C_1 C_2}{C_1 + C_2} \quad (40)$$

根据公式(33)则可得到:

$$CR_{i,j+\frac{1}{2},k} = \frac{\frac{TR_{i,j,k} DELC_i}{0.5 DELR_j} \frac{TR_{i,j+1,k} DELC_i}{0.5 DELR_{j+1}}}{\frac{TR_{i,j,k} DELC_i}{0.5 DELR_j} + \frac{TR_{i,j+1,k} DELC_i}{0.5 DELR_{j+1}}}$$

式中:  $TR$ : 沿行方向上的导水系数 ( $L^2 T^{-1}$ );

$DEL R$  (或 $\Delta r$ ): 沿行方向上的计算单元长度 ( $L$ );

$DEL C$  (或 $\Delta c$ ): 沿列方向上的计算单元长度 ( $L$ )。

将公式(40)化简后可得:

$$CR_{i,j+\frac{1}{2},k} = 2 DELC_i \frac{TR_{i,j,k} TR_{i,j+1,k}}{TR_{i,j,k} DELR_{j+1} + TR_{i,j+1,k} DELR_j} \quad (41)$$

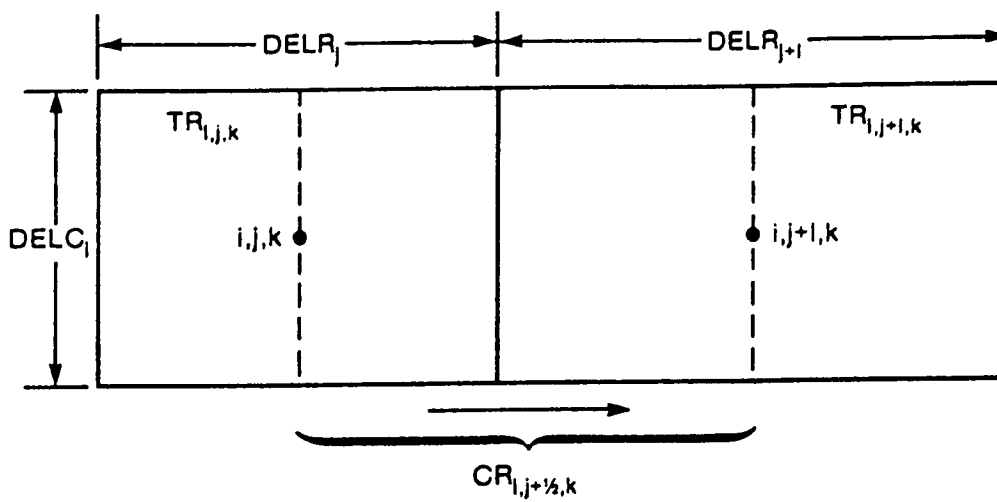
同理可得沿列方向上两相邻计算单元之间的水力传导系数:

$$CC_{i+\frac{1}{2},j,k} = 2 DELR_j \frac{TC_{i,j,k} TC_{i+1,j,k}}{TC_{i,j,k} DELC_{i+1} + TC_{i+1,j,k} DELC_i} \quad (42)$$

其中 $TC$ 为沿列方向上的导水系数( $L^2 T^{-1}$ )。MODFLOW正是用公式(41)和(42)分别计算位于同一层中各相邻计算单元之间的水力传导系数的。当两个相邻计算单元的导水系数均为零时, MODFLOW将直接跳过公式(41)和(42), 并将它们之间的水力传导系数赋为零。

### 潜水含水层中水平方向上水力传导系数的计算

当含水层处于承压状态时, 平面上的水力传导系数( $CR$ 和 $CC$ )在模拟计算过程中保持不变。但当含水层处于非承压的条件下时, 平面上的水力传导系数成为水头的函数。MODFLOW在每次迭代开始之前将重新计算平面上的水力传导系数 $CC$ 和 $CR$ 。在计算 $CC$ 和 $CR$ 时, 首先将渗透系数与饱和厚度相乘得到导水系数, 然后再按公式(41)和(42)求出水力



$$\frac{1}{CR_{i,j+1/2,k}} = \frac{1}{\left( \frac{TR_{i,j,k} DELC_i}{\left( \frac{DELR_j}{2} \right)} \right)} + \frac{1}{\left( \frac{TR_{i,j+1,k} DELC_i}{\left( \frac{DELR_{j+1}}{2} \right)} \right)}$$

$$CR_{i,j+1/2,k} = 2 DELC_i \times \frac{TR_{i,j,k} TR_{i,j+1,k}}{TR_{i,j,k} DELR_{j+1} + TR_{i,j+1,k} DELR_j}$$

说明

$TR_{i,j,k}$ : 计算单元沿行方向上的导水系数

$CR_{i,j,k}$ : 计算单元(i, j, k)和(i, j+1, k)之间的水力传导系数

图25. 平面上两相邻计算单元之间的水力传导系数的计算。

传导系数。

在潜水含水层的条件下，计算单元(i, j, k)的导水系数的计算分下列三种情况进行：

(1) 如果  $HNEW_{i,j,k} \geq TOP_{i,j,k}$

$$\text{则 } TR_{i,j,k} = (TOP_{i,j,k} - BOT_{i,j,k})HYR_{i,j,k} \quad (43)$$

(2) 如果  $TOP_{i,j,k} > HNEW_{i,j,k} > BOT_{i,j,k}$

$$\text{则 } TR_{i,j,k} = (HNEW_{i,j,k} - BOT_{i,j,k})HYR_{i,j,k} \quad (44)$$

(3) 如果  $HNEW_{i,j,k} \leq BOT_{i,j,k}$

$$\text{则 } TR_{i,j,k} = 0$$

这里,  $HYR_{i,j,k}$ 为沿行方向上的渗透系数( $LT^{-1}$ );  $TOP_{i,j,k}$ 为计算单元(i,j,k)的顶面标高(L);  $BOT_{i,j,k}$ 为计算单元(i,j,k)的底面标高(L);  $HNEW_{i,j,k}$ 为某时刻计算单元(i, j, k)的水头(L)。

值得注意的是, MODFLOW仅要求用户输入各种计算单元沿行方向上的渗透系数。而沿列方向的渗透系数则由这些值与平面方向上各向异性率的乘积来计算。在输入时, 用户应给每一层输入一个各向异性率。如果同一层内的岩层有明显的非均质性, 用户则应该对MODFLOW的源程序进行修改, 并加入一个类似于 $HYR$ 那样的数组用于贮存沿列方向上的渗透系数。这种情况在MODFLOW的应用中并不常见。

在计算过程中, 如果计算水头低于该计算单元之底面标高, 该计算单元则处于非饱和状态。这时, MODFLOW并不考虑该计算单元在非饱和态下的渗透系数, 而是将这个计算单元列为“干枯计算单元”(dry cell), 并将其导水系数赋值为零。一旦一个计算单元变为干枯计算单元, 它将被重新定义为不透水或无效计算单元, 从而被摒除出以后的模拟计算。计算结果也可能因为干枯计算单元的出现而受到影响。[译注: 我们将在附录中专门讨论干枯计算单元的一些处理方法。]

### 垂向水力传导率的计算

平面上的水力传导系数 $CC$ 和 $CR$ 均由MODFLOW根据输入数据计算得到, 而垂向水力传导系数则由用户自己根据含水层厚度和垂向渗透系数计算出垂向水力传导率(vertical conductance, 简称为 $VC$ )后再输入给MODFLOW。MODFLOW读入垂向水力传导率后, 乘以计算单元的面积以得到垂向水力传导系数。下面我们来介绍如何计算垂向水力传导率。

一般来说, 在垂向上相邻两个计算单元(i,j,k)和(i,j,k+1)之间还可能包括有几个水文地质亚层。这些亚层的渗透系数和厚度分别由 $K_1, K_2, \dots, K_n$ 和 $\Delta Z_1, \Delta Z_2, \dots, \Delta Z_n$ 来表示。假定这两个计算单元的平面断面积为 $DEL R_j * DEL C_i$ , 则一个亚层(g)的垂向水力传导系数为

$$C_g = \frac{K_g DEL R_j * DEL C_i}{\Delta Z_g} \quad (46)$$

按照前面讲过的计算方法, 这些亚层的综合水力传导系数,  $C_{i,j,k+1/2}$ 为:

$$\frac{1}{C_{i,j,k+1/2}} = \sum_{g=1}^n \frac{1}{C_g} = \sum_{g=1}^n \frac{1}{\frac{K_g * DEL R_j * DEL C_i}{\Delta Z_g}} = \frac{1}{DEL R_j * DEL C_i} \sum_{g=1}^n \frac{\Delta Z_g}{K_g} \quad (47)$$

或写为

$$\frac{C_{i,j,k+1/2}}{DEL R_j * DEL C_i} = \frac{1}{\sum_{g=1}^n \frac{\Delta Z_g}{K_g}} \quad (48)$$

其中:  $\frac{C_{i,j,k+1/2}}{DEL R_j * DEL C_i}$  也可称为垂向水力传导率 (或称越流系数, vertical leakance, 单位为 $T^{-1}$ )。在这里记作 $VC_{i,j,k+1/2}$ :

$$VC_{i,j,k+1/2} = \frac{1}{\sum_{g=1}^n \frac{\Delta Z_g}{K_g}} \quad (49)$$

$VC$ 的值由用户计算并输入。这一点很重要。在MODFLOW的输入数据中并没有垂向渗透系数和层厚。对于一个承压含水层来说, 含水层的厚度仅由用户用来计算与厚度有关的水文地质参数, 如导水系数、贮水系数和垂向水力传导率; MODFLOW并不需要读入含水层的厚度资料。这样一来, 对一个数值模型的空间几何形态的定义就产生了多义性。仅从MODFLOW的输入输出数据, 我们无法得知含水层的厚度和模型的纵深。

一般情况下, 垂向水力传导率可按公式(49)进行计算, 并将结果以二维数组的形式输入MODFLOW。值得注意的是, 垂向水力传导率是指某计算单元(i,j,k)和其下伏的计算单元(i,j,k+1)之间的水力传导率。正因为如此, 对模型最底层不需要输入垂向水力传导率。下面我们来讨论几种特殊情况下垂向水力传导率的计算方法。当然,  $VC_{i,j,k+1/2}$ 表示计算单元(i,j,k)和(i,j,k+1)之间的垂向水力传导率。

当计算单元(i,j,k)和(i,j,k+1)均属于同一水文地质单元(参见图26)时,这两个单元之间的垂向水力传导率可由公式(49)直接得到:

$$VC_{i,j,k+\frac{1}{2}} = \frac{K_{z,i,j}}{\Delta Z_{k+\frac{1}{2}}} \quad (50)$$

其中 $\Delta Z_{k+\frac{1}{2}}$ 为这两个计算单元中心点之间的距离,即 $\frac{\Delta V_k}{2}$ 和 $\frac{\Delta V_{k+1}}{2}$ 之和。在模型设计过程中,为提高计算精度常将同一水文地质单元在垂向上再分为若干亚层。

图27表示另外一种情况,两个上下相邻的计算单元分别代表两个相邻但性质有所不同的水文地质单元。两个计算单元的中心分别落于这两个水文地质层的中心位置。当计算单元(i,j,k)和(i,j,k+1)的渗透系数不等时,相对应的垂向水力传导率可由下式求出:

$$VC_{i,j,k+\frac{1}{2}} = \frac{1}{\frac{\Delta V_k / 2}{K_{z,i,j,k}} + \frac{\Delta V_{k+1} / 2}{K_{z,i,j,k+1}}} \quad (51)$$

式中:  $\Delta V_k$ : 层k的厚度(L);

$\Delta V_{k+1}$ : 层k+1的厚度(L);

$K_{z,i,j,k}$ : 层k的垂向渗透系数 ( $LT^{-1}$ );

$K_{z,i,j,k+1}$ : 层k+1的垂向渗透系数 ( $LT^{-1}$ )。

第三种情况(参见图28)是在计算单元(i,j,k)和(i,j,k+1)之间含有一弱透水层,而该弱透水层又没有在模型之中作为一层明确地表示出来。但它的存在又应通过公式(49)反映在垂向水力传导率之中,即:

$$VC_{i,j,k+\frac{1}{2}} = \frac{1}{\frac{\Delta Z_u / 2}{K_{zu}} + \frac{\Delta Z_c}{K_{zc}} + \frac{\Delta Z_l / 2}{K_{zl}}} \quad (52)$$

其中:  $\Delta Z_u$ 为上层的厚度(L);

$\Delta Z_c$ 为弱透水层的厚度(L);

$\Delta Z_l$ 为下层的层厚(L);

$K_{zu}$ 为上层的垂向渗透系数( $LT^{-1}$ );

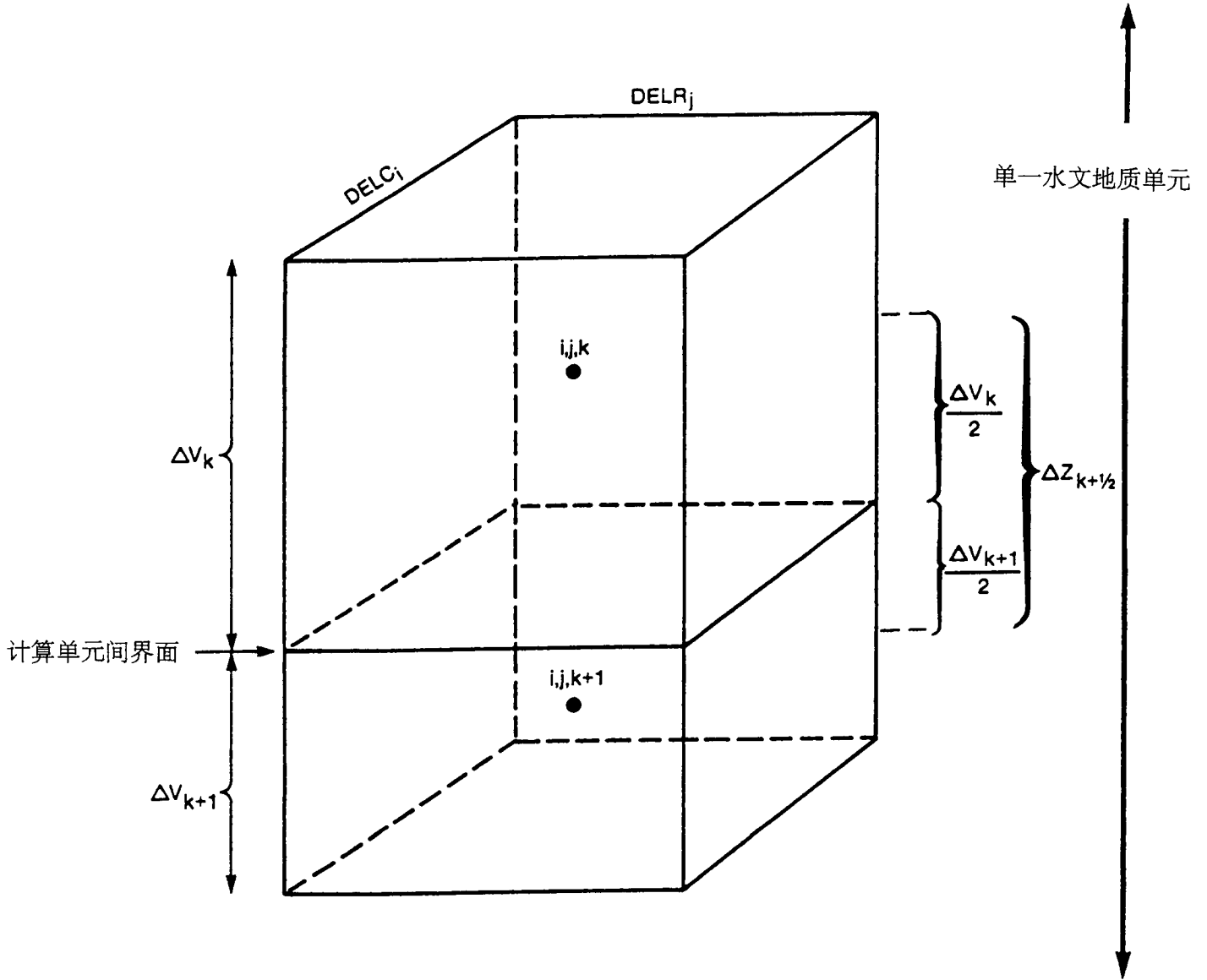


图26. 位于同一水文地质单元内上下两计算单元之间的垂向水力传导率的计算。



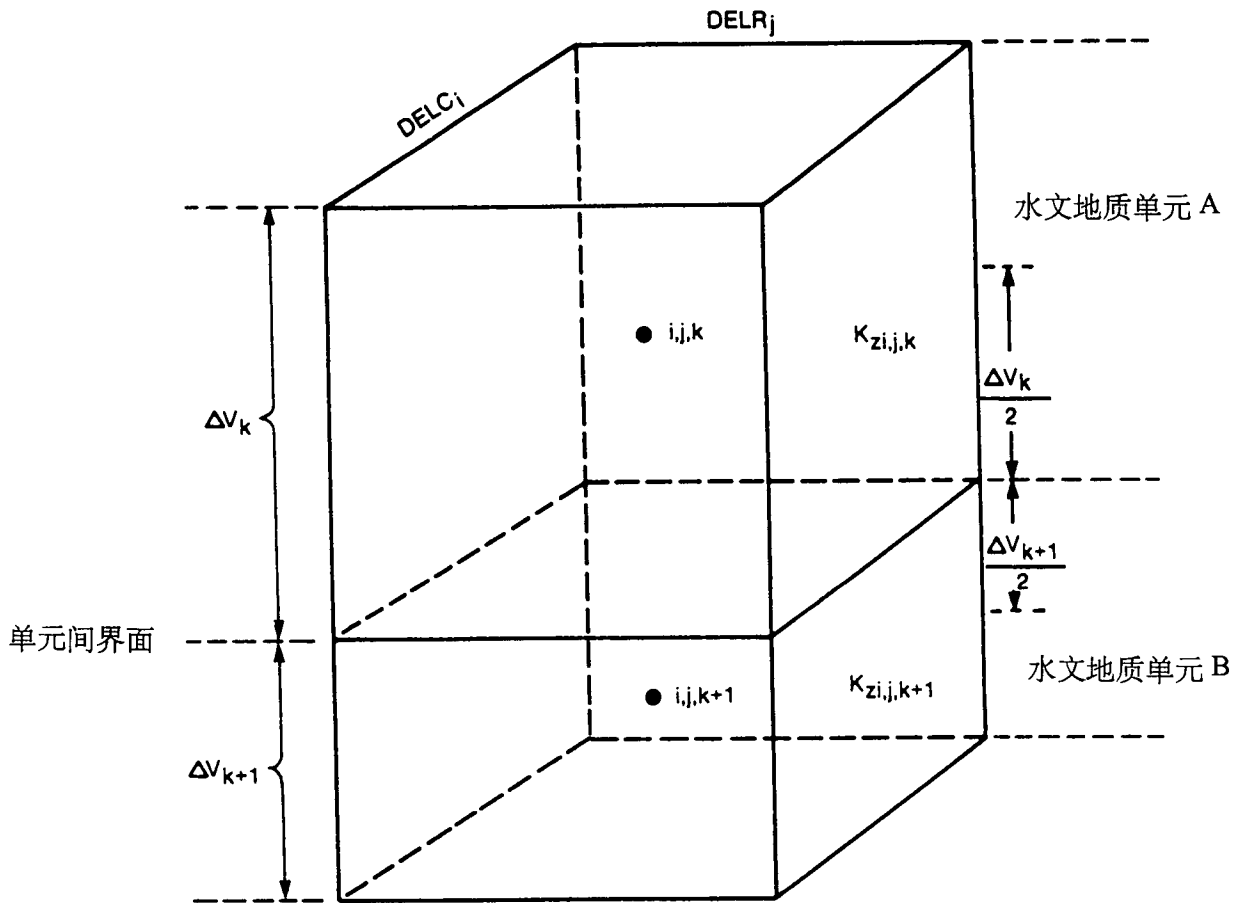
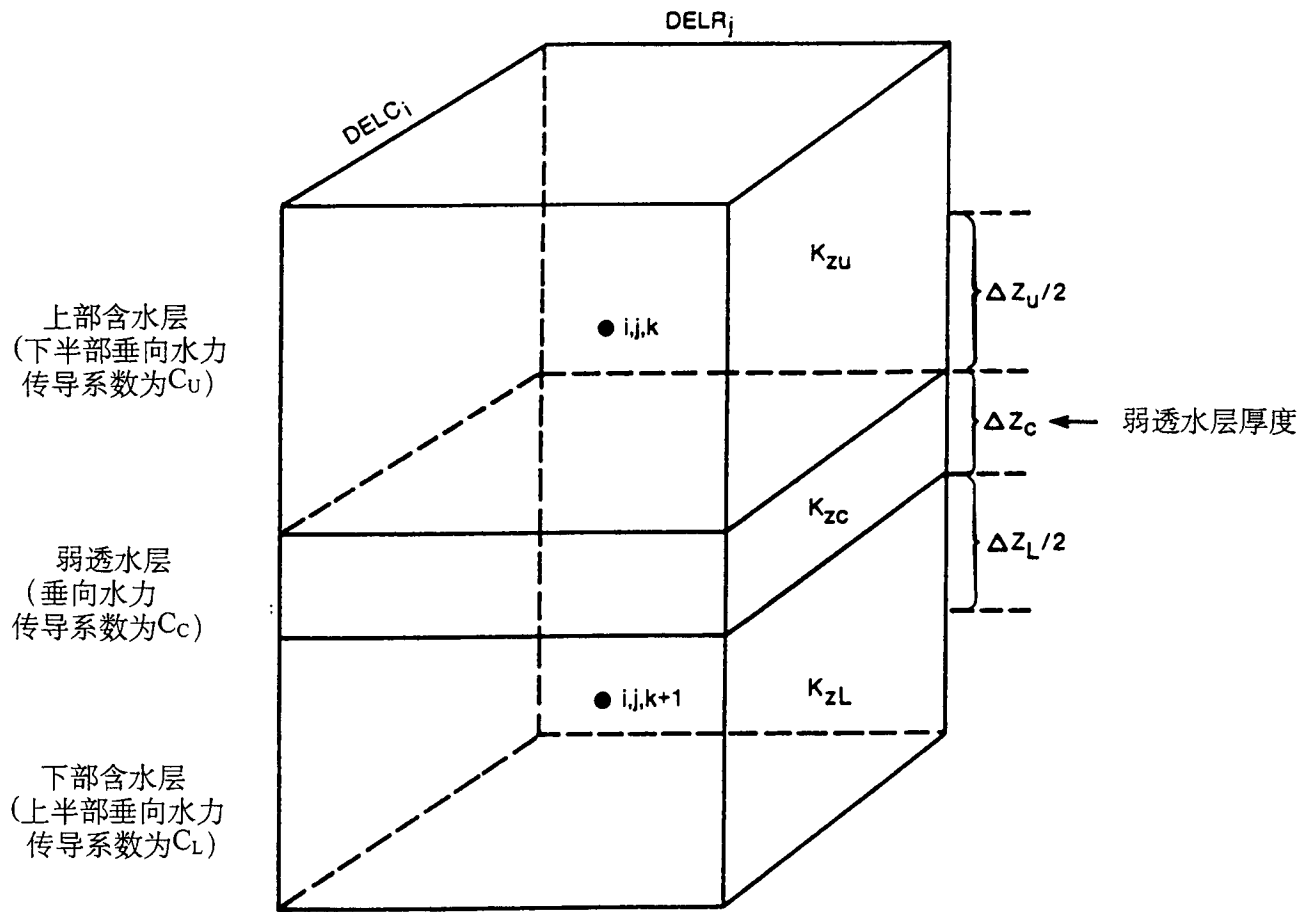


图27. 模型按水文地质单元分层时垂向水力传导率的计算。



$$\frac{1}{C_{eq}} = \frac{1}{C_u} + \frac{1}{C_c} + \frac{1}{C_L} =$$

$$\frac{1}{DELC_i \cdot DELR_j} \left\{ \frac{\Delta Z_u/2}{K_{zu}} + \frac{\Delta Z_c}{K_{zc}} + \frac{\Delta Z_L/2}{K_{zL}} \right\}$$

$$VCONT_{i,j,k+1/2} = \frac{1}{\frac{\Delta Z_u/2}{K_{zu}} + \frac{\Delta Z_c}{K_{zc}} + \frac{\Delta Z_L/2}{K_{zL}}}$$

图28. 两含水层中夹有弱透水层时垂向水力传导率的计算。

$K_{ZC}$ 为弱透水层的垂向渗透系数( $LT^{-1}$ );

$K_{ZL}$ 为下层的垂向渗透系数( $LT^{-1}$ )。

以上这些参数都可能随空间的位置而发生变化。根据实际工作的经验,  $K_{ZC}$ 一般比 $K_{ZU}$ 和 $K_{ZL}$ 要小得多。从计算结果上来看, 含有 $K_{ZU}$ 和 $K_{ZL}$ 的两项可以忽略不计。这种情况下的垂向水力传导率可按下式计算:

$$VC_{i,j,k+\frac{1}{2}} = \frac{K_{ZC}}{\Delta Z_C} \quad (53)$$

如果我们进一步假定夹层的水平渗透系数和储水能力可以忽略不计, 则图(28)所示的上中下三层含水层结构可仅用模型层k和层k+1来分别表示; 而介于它们之间的弱透水夹层的存在则由这两模型层之间的垂向水力传导系数来反映。这种方法在MODFLOW中称为“准三维”(quasi-three-dimensional)方法。

前面已经提到, 在MODFLOW中用户输入的数据为垂向水力传导率而非垂向渗透系数和厚度。MODFLOW根据读入的垂向水力传导率和计算单元的面积来计算垂向水力传导系数。用户应根据实际情况和模型的设计选择适当的公式计算水力传导率。这样做, 虽然会增加用户的工作量, 但也提高了MODFLOW解决问题的灵活性。用户可选用正交网格(图9b), 非正交网格(图9c), 也可以按准三维的办法设计模型。

### 非饱和带垂向渗流的计算

与计算单元相对应的有限差分公式可以写为:

$$\begin{aligned} & CR_{i,j-\frac{1}{2},k}(h_{i,j-1,k}^m - h_{i,j,k}^m) + CR_{i,j+\frac{1}{2},k}(h_{i,j+1,k}^m - h_{i,j,k}^m) + CC_{i-\frac{1}{2},j,k}(h_{i-1,j,k}^m - h_{i,j,k}^m) \\ & + CC_{i+\frac{1}{2},j,k}(h_{i+1,j,k}^m - h_{i,j,k}^m) + CV_{i,j,k-\frac{1}{2}}(h_{i,j,k-1}^m - h_{i,j,k}^m) + CV_{i,j,k+\frac{1}{2}}(h_{i,j,k+1}^m - h_{i,j,k}^m) \quad (54) \\ & + P_{i,j,k}h_{i,j,k}^m + Q_{i,j,k} = SS_{i,j,k}(\Delta r_j \Delta c_i \Delta v_k) \frac{h_{i,j,k}^m - h_{i,j,k}^{m-1}}{t_m - t_{m-1}} \end{aligned}$$

在这个公式中,  $CV_{i,j,k+\frac{1}{2}}(h_{i,j,k+1}^m - h_{i,j,k}^m)$ 一项给出了通过计算单元(i,j,k)底面的地下水渗流量:

$$q_{i,j,k+\frac{1}{2}} = CV_{i,j,k+\frac{1}{2}}(h_{i,j,k+1}^m - h_{i,j,k}^m) \quad (55)$$

按照规定, 当  $q_{i,j,k+\frac{1}{2}}$  为正时, 地下水由计算单元  $(i,j,k+1)$  流入计算单元  $(i,j,k)$ ; 反之, 地下水从计算单元  $(i,j,k)$  通过底面流入计算单元  $(i,j,k+1)$ 。公式(54)和(55)均要求计算单元  $(i,j,k)$  和  $(i,j,k+1)$  为饱和, 即它们的水头高于它们的顶面标高。但在实际模拟过程中, 有时承压含水层会转变为非承压含水层。这样, 处于非承压态的计算单元的一部分就可能转成非饱和态。下面讨论对这种情况的处理办法。

图29中的含水层系统包括两个含水层和一个弱透水夹层。对该弱透水层的处理可以按前面讨论的准三维办法处理。下面我们讨论相关的渗流计算问题。

假定在下层含水层  $(k+1)$  中抽水, 导致该单元的水头低于该单元的顶面标高。这时该计算单元就从承压态变为非承压态, 同时上层含水层成了上层滞水含水层。而弱透水层仍处于饱和态。在该弱透水层顶面处的水头为上层含水层的水头  $(h_{i,j,k})$ , 而其底面的压力水头为零 (或一个大气压)。所以, 在弱透水层底面处的水头事实上是该点的标高,  $TOP_{i,j,k+1}$  而通过弱透水层的流量为:

$$q_{i,j,k+\frac{1}{2}} = CV_{i,j,k+\frac{1}{2}}(TOP_{i,j,k+1} - h_{i,j,k}^m) \quad (56)$$

由于已经假设  $h_{i,j,k+1}$  低于其顶面标高  $TOP_{i,j,k+1}$ , 故此时地下水总是从计算单元  $(i,j,k)$  向下流入单元  $(i,j,k+1)$ 。但值得注意的是, 此时越流补给量的大小与下层计算单元  $(i,j,k+1)$  的水位无关。由于公式(56)与MODFLOW中其它流量计算公式的形式有所不同, 在程序中需要特别的处理。如果将公式(56)直接代入公式(54), 公式(27)中的系数矩阵则会变成不对称矩阵。这对求解过程会造成困难。为避免这种情况出现, MODFLOW中采用了另一种处理方法。将公式(55)代入公式(54)后, 计算单元  $(i,j,k)$  和  $(i,j,k+1)$  之间的流量为:

$$CV_{i,j,k+\frac{1}{2}}(h_{i,j,k+1}^m - h_{i,j,k}^m)$$

由于  $h_{i,j,k}$  大于  $h_{i,j,k+1}$ , 故地下水从计算单元  $(i,j,k)$  流向单元  $(i,j,k+1)$ 。而实际流量应由公式(56)来计算。为获得正确的流量计算, 我们将公式(56)和公式(55)之差用作纠正项,  $q_c$ :

$$q_c = CV_{i,j,k+\frac{1}{2}}(h_{i,j,k+1}^m - TOP_{i,j,k+1}) \quad (57)$$

然后将这个纠正项加在公式(54)的右侧以抵消由于引用公式(55)所引起的误差。由于公式(54)与公式(24)等价, 所以这样做实际上相当于在公式(26)的右侧项加上这个纠正项。但

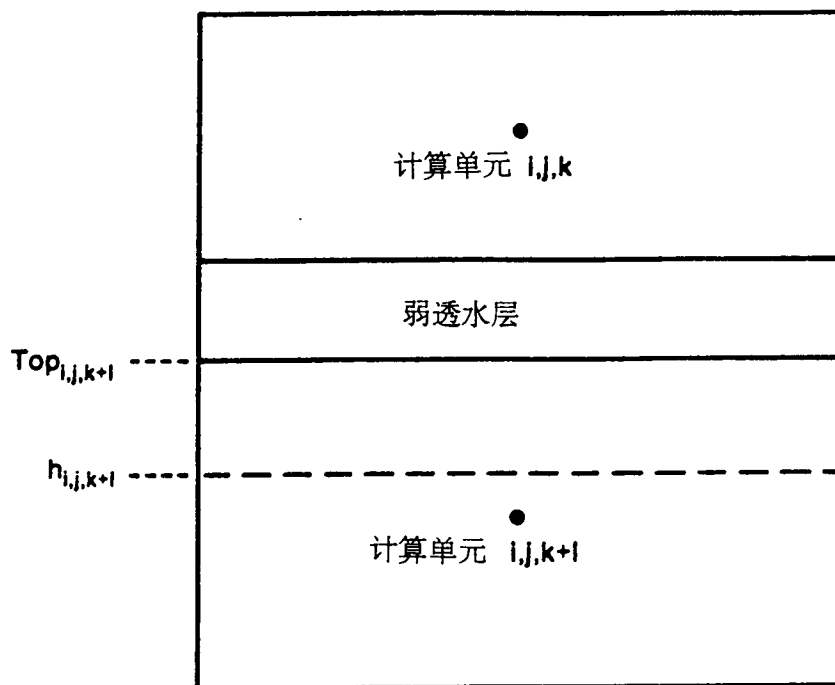


图29. 上层滞水示意图。

这样一来, 公式(26)的右侧项就会出现含有未知水头的项。为解决这一问题, 程序在计算纠正项时用前一次迭代计算得到的水头值 ( $h_{i,j,k+1}^{m,n-1}$ ) 来代替公式(57)中的未知水头项

$h_{i,j,k+1}^m$ , 即有:

$$q_{C,n} = CV_{i,j,k+\frac{1}{2}}(h_{i,j,k+1}^{m,n-1} - TOP_{i,j,k+1}) \quad (58)$$

其中  $q_{C,n}$  为第  $n$  次迭代时使用的纠正项;

$h_{i,j,k+1}^{m,n-1}$  为前一次迭代求得的水头值。

随着迭代过程趋于收敛,  $h_{i,j,k+1}^{m,n-1}$  与  $h_{i,j,k+1}^{m,n}$  之间的差别也会逐渐减小。当  $n=1$  时 (即第一次迭代时),  $h_{i,j,k+1}^{m,n-1}$  的值可用初始水头来代替。注意由公式(58)所表示的纠正项是针对计算单元  $(i,j,k)$  而言的, 而该单元本身并没有出现非饱和带的情况。那么对于出现非饱和现象的计算单元又该怎么办呢?

为便于讨论, 我们将计算单元  $(i,j,k)$  作为出现非饱和情况的单元, 而假定其上面相邻的计算单元  $(i,j,k-1)$  的状态没有发生变化。从计算单元  $(i,j,k-1)$  流向计算单元  $(i,j,k)$  的流量则为  $CV_{i,j,k-\frac{1}{2}}(h_{i,j,k-1}^m - h_{i,j,k}^m)$ 。但实际流量为  $CV_{i,j,k-\frac{1}{2}}(h_{i,j,k-1}^m - TOP_{i,j,k})$ 。二者之差为

$$q_c' = CV_{i,j,k-\frac{1}{2}}(TOP_{i,j,k} - h_{i,j,k}^m)$$

其中  $q_c'$  为纠正项。为编程方便, MODFLOW 将  $q_c'$  分为两部分并分别加至(26)式的两侧: 将  $CV_{i,j,k-\frac{1}{2}}$  加至(26)式左侧的  $HCOF$  中, 同时将  $CV_{i,j,k-\frac{1}{2}}TOP_{i,j,k}$  加在(26)式的右侧项  $RHS$  中。因为  $HCOF$  是系数矩阵中主对角线上的元素, 所以加上纠正项后不会改变系数矩阵的对称性。

在模拟过程中, 是否要求 MODFLOW 处理上层滞水的情况, 由用户在定义层类型的变量(LAYCON)中说明。有关 LAYCON 的使用方法, 后面将进一步介绍。

### 贮水系数的处理

根据贮水系数的计算, MODFLOW 将模型层分为两类: 贮水系数为常数和贮水系数可能进行类型转换的模型层。在前一种情况中, 贮水系数仅需要在运算开始时计算一次, 其

数值在整个模拟过程中保持不变；在后面一种情况中，贮水系数的选用将依据含水层的性质，承压含水层或非承压含水层而定。因为在实际工作中，一个承压含水层可能由于抽水而转变为非承压含水层，或一个非承压含水层由于注水而转变成为承压含水层。含水层是否承压，含水层物质释放或吸收地下水的方式及数量都是不相同的。用户可以通过数组 LAYCON 定义模型中各层的性质。

当贮水系数为常数时，公式(24)或(54)中贮水项的计算比较简单：直接使用由用户输入的贮水系数，并按下式求得贮水项：

$$\frac{\Delta V}{\Delta t} = SS_{i,j,k} (\Delta r_j \Delta c_i \Delta v_k) \frac{h_{i,j,k}^m - h_{i,j,k}^{m-1}}{t_m - t_{m-1}} \quad (60)$$

式中  $\frac{\Delta V}{\Delta t}$  表示计算单元(i,j,k)中地下水在单位时间内的变化量； $SS_{i,j,k}$  为贮水率 ( $L^{-1}$ )；

$\Delta r_j \Delta c_i \Delta v_k$  为计算单元的体积； $h_{i,j,k}^m$  为时间段 $t_m$ 结束时的水头， $h_{i,j,k}^{m-1}$  为时间段 $t_m$ 开始时的起始水头。在(26)式中，用了另一种形式来表示贮水项： $SC1_{i,j,k} = SS_{i,j,k} (\Delta r_j \Delta c_i \Delta v_k)$ 。

在MODFLOW中，这一项称之为计算单元(i,j,k)的贮水量或第一类贮水量(primary storage capacity)。后面在讨论贮水系数的转换时，我们还将引入第二类贮水系数 (secondary storage capacity)的概念。根据贮水量的定义，计算单元(i,j,k)中贮水量的变化率为：

$$SC1_{i,j,k} (h_{i,j,k}^m - h_{i,j,k}^{m-1}) / (t_m - t_{m-1})$$

上式可以分解为两部分， $SC1_{i,j,k} h_{i,j,k}^m / (t_m - t_{m-1})$  加入公式(26)左侧的HCOF中，而将其余部分加到(26)式的右侧项RHS之中。

在BCF子程序包的输入数据中，应包括模型各层的贮水系数（无量纲数）。对于一个承压含水层而言，贮水系数为贮水率与该层厚度的乘积。而对于一个潜水含水层来说，用户应输入潜水含水层的给水度。这里应当注意，给水度是无量纲数，与含水层饱和厚度无关。MODFLOW逐层读入由用户提供的给水度或贮水系数，并存入数组SC1之中。

### 贮水系数的转换

如果在模拟过程中，某计算单元的水头要么保持高于顶板标高，要么保持低于顶板标高，则无须进行贮水系数的转换，计算中可直接使用SC1的值。但当含水层的性质由于水

头的变化而发生变化, 即由承压态变为非承压态或由非承压态转变为承压态时, 则应当进行贮水系数的转换并选用正确的贮水系数或给水度。是否进行这种转换, 则要看用户如何定义含水层的类型。如果用户估计这种转换发生的可能性较大, 则应要求MODFLOW在运算过程中, 检查含水层的类型。同时用户还应当提供第二类贮水系数。在这种情况下, 第一类贮水量将用于含水层处于承压态时的计算。 $SC1_{i,j,k}$ 为承压含水层的贮水系数与计算单元面积的乘积; 而第二类贮水量,  $SC2_{i,j,k}$ , 应当为给水度与计算单元面积的乘积, 用于含水层处于非承压态的计算。贮水系数和给水度的值均由用户按二维数组逐层输入, 由程序读入并与计算单元面积相乘, 然后分别存入SC1和SC2两个数组中。用户应当仔细, 不要将这两组贮水系数弄混了。

下面介绍MODFLOW如何进行贮水系数的转换。对于允许类型转换的含水层, 其贮水量的变化可表示为:

$$\frac{\Delta V}{\Delta t} = \frac{SCB(h_{i,j,k}^m - TOP_{i,j,k}) + SCA(TOP_{i,j,k} - h_{i,j,k}^{m-1})}{t_m - t_{m-1}} \quad (61)$$

式中  $\frac{\Delta V}{\Delta t}$  为单位时间内计算单元(i,j,k)内由于贮水量的变化而引起的水量变化。 $SCA$ 表示计算单元(i,j,k)在时间段 $m$ 开始时的贮水能力, 而 $SCB$ 表示该单元“目前”的贮水能力。如果在时间 $t_{m-1}$ 时水头 $h_{i,j,k}$ 高于或等于该单元顶面标高, 则含水层处于承压态。这时 $SCA$ 的值等于 $SC1_{i,j,k}$ 。如果经过数次迭代后, 计算的水头仍高于或等于顶板标高, 则 $SCB$ 的值也等于 $SC1_{i,j,k}$ , 即含水层仍处于承压态。这种情况下, 公式(61)与公式(60)等价。

但如果经过若干次迭代后, 计算的水头低于顶面标高, 含水层由承压态转变为非承压态(图30)。这时 $SCB$ 使用 $SC2_{i,j,k}$ 的值(即给水度与单元面积的乘积)。在这种情况下, 含水层释放的水量实际上包括两部分:

$$(1) \text{弹性释放: } SC1_{i,j,k} (TOP_{i,j,k} - h_{i,j,k}^{m-1}) / (t_m - t_{m-1})$$

$$(2) \text{重力释放: } SC2_{i,j,k} (h_{i,j,k}^m - TOP_{i,j,k}) / (t_m - t_{m-1})$$

如果在某时间段 $m$ 开始时含水层就处于非承压态, 则(61)式中 $SCA$ 等于给水度 $SC2_{i,j,k}$ 。在迭代过程中, 只要计算的水头低于该单元之顶面标高, 含水层仍处于非承压态,  $SCB$ 的值也将等于给水度 $SC2_{i,j,k}$ 。这时, 公式(61)与(60)等价。如果经过若干次迭代计算后, 发现水头高于该单元的顶面标高, 则含水层的类型发生了转变: 由非承压态变为承压态。这时



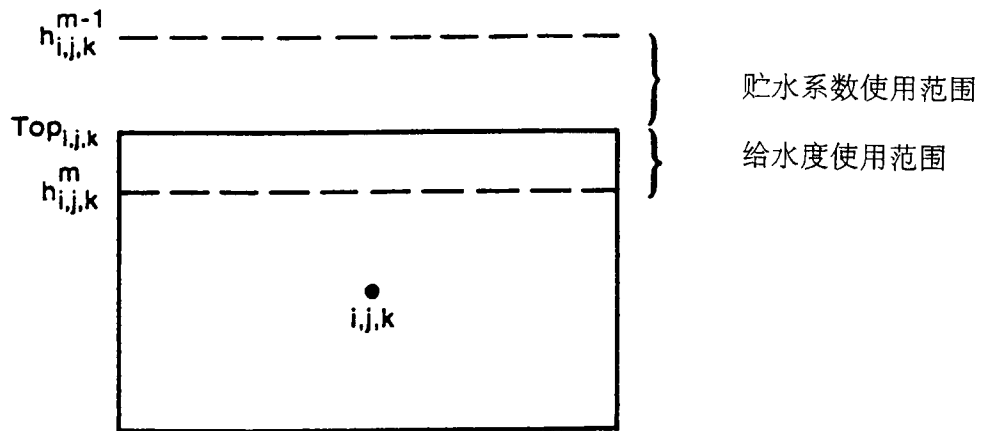


图30. 贮水系数/给水度转换示意图。

$SCB$ 的值则等于 $SC1_{i,j,k}$  (贮水系数与单元面积之乘积) 同样, 这时由计算单元所吸收的水量也包括由弹性贮存和重力贮存两部分。

对于有可能发生含水层类型转变的计算单元,  $MODFLOW$ 将在每次迭代开始之前检查上次迭代所计算的水头与该单元的顶面标高以确定是否进行贮水系数/给水度的转换。这样做, 从理论上来说更为符合实际。但同时也产生了非线性问题, 故可能造成求解上的困难。当然, 由于计算量的增加, 也会增加程序的运行时间。

为计算和程序编写方便起见, 现将公式(61)重写为:

$$\frac{\Delta V}{\Delta t} = \frac{SCB}{t_m - t_{m-1}} h_{i,j,k}^m + \frac{SCA(TOP_{i,j,k} - h_{i,j,k}^{m-1}) - SCB * TOP_{i,j,k}}{t_m - t_{m-1}} \quad (62)$$

在计算过程中, 将含有未知项 $h_{i,j,k}^m$ 的部分并入公式(26)中的左侧系数项 $HCOF$ , 同时将常数项加入(26)式的右侧项 $RHS_{i,j,k}$ 之中。

### 特殊处理方式的使用及其局限性

以上共讨论了三种特殊情况的处理方式: 潜水含水层和承压含水层在平面上的水力传导系数 $CC$ 和 $CR$ 的计算; 非饱和带垂向渗流以及贮水系数/给水度的转换。在这些问题的讨论过程中, 我们均假定每个模型分层代表某种含水层或透水体。含水层和透水体之间由弱透水层分开。如果这些假定不成立, 上述的处理方式有可能会造成许多问题, 计算结果的可靠性也将大打折扣。例如, 如果将一个独立的潜水含水层剖分为若干个模型层, 当潜水面下降时可能会切割入数个模型层。这时不仅难于定义贮水系数/给水度, 同时由于可能出现干枯计算单元而影响模拟结果 (参见附录A)。因此, 在使用这些特殊处理方式时, 应当慎重。

### 输入数据

BCF子程序包需要大量与含水层物质以及透水性有关的数据。这些数据包括: 导水系数、渗透系数、给水度和贮水系数、垂向水力传导率、顶面及底面标高等。所有这些数据均应以二维数组的形式逐层输入。除了这些以外, 计算单元的大小、含水层类型等数据也

由BCF子程序包读入。

MODFLOW根据用户定义的含水层类型(LAYCON)来决定是否以及如何使用前述的特殊处理方法,如水力传导系数是否为常数、是否进行贮水系数/给水度的转换等。由于对这些特殊情况的处理方式不同,MODFLOW所需要的输入数据也不尽相同。

在数据输入过程中,MODFLOW按层从上至下(即从第一层至最后一层)读入数据。这种逐层输入方法比使用三维数组的方式更为简单,且易于检查和修改输入数据。

不论计算单元是有效还是无效计算单元,输入数据都应包括模型中的所有单元。只是对于无效计算单元而言,所赋的值并不用于实际运算。

由于含水层渗透性的各向异性,平面上的导水系数和渗透系数均应按行方向和列方向分别定义。为了减少输入数据量,MODFLOW要求用户对模型每个计算单元在行方向上的导水系数或渗透系数进行定义,同时定义出各层的各向异性比率。MODFLOW则按各层的各向异性比率求出各个计算单元在列方向上的导水系数和渗透系数。每层的各向异性比率只有一个。所以用户在输入数据中以一个一维实型数组定义各层的各向异性比率MODFLOW将这些数据存入数组TRPY。

在垂向上,用户应计算出各计算单元的垂向水力传导率(VCONT)。由于垂向水力传导率( $V_{cont}$ 或 $k_z/\Delta z$ )定义为层 $k$ 与其下伏层 $k+1$ 之间的透水能力,故不用给出模型最底层的垂向水力传导率。例如,第一层的 $V_{cont}$ 输入数据实际上是第一层和第二层之间的垂向渗透率。

除了上述数据之外,BCF子程序包读入的数据还包括模型的行距和列距(即:DELX和DELR),稳定流或非稳定流运算标示符(ISS),以及是否存写计算单元间渗流量的标示符(IBCFCB)。如果模拟为稳定流计算( $ISS \neq 0$ ),则MODFLOW不需要读入有关贮水系数/给水度等数据项。故在准备稳定流模拟的数据时,输入文件中不应列含有贮水系数或给水度,否则程序会误读数据。同样,如果含水层类型的定义与实际输入不符,也可能造成程序的输入错误。

MODFLOW将含水层按处理方式分为四种类型。这四种类型的标示符与相应的参数使

用方式列举如下:

含水层类型0 (LAYCON=0): 这一类型多用于模拟承压含水层。当降深与含水层厚度相比不大, 且从顶层越流补给量可以忽略不计时, 它也可用于模拟非承压含水层。遇到这种类型, 程序将不进行任何前面提到的特殊处理。贮水系数、导水系数和垂向水力传导率均在运算过程中保持不变。使用这种类型的含水层时, 输入数据包括导水系数、垂向水力传导率(若不是模型之最底层的话)。对于非稳定流计算, 还应输入有关给水度的资料。

含水层类型1 (LAYCON=1): 这种类型仅用于单层模型或多层模型之最上层, 并且该含水层具有潜水含水层的特征。对于这种含水层, MODFLOW不进行贮水系数/给水度的转换, 给水度保持不变。由于这种类型的含水层仅出现于模型的最上层, 故不需要进行非饱和带垂向渗流处理。由于潜水含水层的饱和厚度可能随时间发生变化, 故MODFLOW需在每次迭代之前重新计算饱和厚度以及导水系数。使用这种类型的含水层时, 输入数据包括渗透系数、底面标高和垂向水力传导率(VCONT)。对于非稳定流计算, 还应输入有关给水度的资料。这一类型仅用于模型之最顶层, 否则将导致运行错误。

含水层类型2 (LAYCON=2): 当被模拟的含水层既具有承压含水层的性质又具有非承压含水层的性质时, 常使用这一类型。对于这一类型含水层, MODFLOW将根据需要进行贮水系数/给水度的转换, 和非饱和带垂向渗流的处理, 但不重新计算水力传导系数。所以只有当整个含水层厚度的大部分始终处于饱和态时才宜于使用这种类型。如果水头变化较大, 或饱和厚度仅占含水层厚度一小部分时, 使用这一类型将会导致过高估计导水系数 (MODFLOW用整个含水层厚度进行导水系数的计算)。由于这种含水层可能发生贮水系数/给水度的转换, 故输入数据应包括贮水系数和给水度。同时, 为处理非饱和带垂向渗流问题, 输入数据中还应包括计算单元的顶面高程。当然, 如果不是模型的最底层, 输入数据还应包括垂向水力传导率。

[译注: 由于这一类型常会过高估计导水系数, 故在实际工作中较为少用。]

含水层类型3 (LAYCON=3): 这种类型的含水层用于模拟典型的潜水含水层。与含水层类型1不同的是, 这一类型不必是模型的最顶层。对于这一类型MODFLOW将在每次迭代之后重新计算导水系数, 并根据水头的变化进行贮水系数/给水度的转换, 以及非饱和带垂向渗流的计算。所要求的数据包括渗透系数、底面标高、贮水系数、给水度和顶面标高。如果不是模型的最底层, 输入数据还应包括垂向水力传导率。

BCF子程序包是MODFLOW的核心部分。这个子程序包所要求的数据比较多，结构变化也比较大。故用户在准备输入数据时应当格外仔细。

### BCF子程序包的输入数据及格式

BCF子程序包所需要的各项数据从由IUNIT(1)所定义的设备号读入。

对每次模拟

由子程序BCF1AL读入的数据包括

1. 数据名称: ISS      IBCFCB  
输入格式: I10      I10
2. 数据名称: LAYCON (NLAY) (最多80层)  
输入格式: 40I2  
(如果少于40层, 使用一个记录行; 否则使用两个记录行。)

由子程序BCF1RP读入的数据包括

3. 数据名称: TRPY (NLAY)  
输入形式: 一维实型数组  
读入子程序: U1DREL
4. 数据名称: DELR (NCOL)  
输入形式: 一维实型数组  
读入子程序: U1DREL
5. 数据名称: DELC (NROW)  
输入形式: 一维实型数组  
读入子程序: U1DREL

下列的各项数据按层输入。每层数据均以二维数组的形式输入。数据的种类则取决于含水层的类型 (LAYCON) 以及模拟的性质 (稳定流还是非稳定流)。数据的输入应严格按照规定进行。输入从第一层开始, 向下逐层进行。

对于非稳定流模拟 (ISS=0)

6. 数据名称: sf1 (NCOL, NROW) (贮水系数或给水度)

输入形式: 二维实型数组

读入子程序: U2DREL

如果含水层类型为0或2 (LAYCON=0, 2)

7. 数据名称: Tran (NCOL, NROW) (导水系数)

输入形式: 二维实型数组

读入子程序: U2DREL

如果含水层类型为1或3 (LAYCON=1, 3)

8. 数据名称: HY (NCOL, NROW) (渗透系数)

输入形式: 二维实型数组

读入子程序: U2DREL

9. 数据名称: BOT (NCOL, NROW) (底面标高)

输入形式: 二维实型数组

读入子程序: U2DREL

如果不是最底层

10. 数据名称: VCONT (NCOL, NROW) (垂向水力传导率)

输入形式: 二维实型数组

读入子程序: U2DREL

非稳定流模拟并且含水层类型为2或3 (ISS=0, LAYCON =2, 3)

11. 数据名称: sf2 (NCOL, NROW) (给水度)

输入形式: 二维实型数组

读入子程序: U2DREL

如果含水层类型为2或3

12. 数据名称: TOP (NCOL, NROW) (顶面标高)

输入形式: 二维实型数组

读入子程序: U2DREL

### 输入数据说明

ISS: 稳定流模拟标示符:

ISS=0 非稳定流模拟

ISS≠0 稳定流模拟

IBCFCB: 流量输出标示符:

IBCFCB>0: MODFLOW使用其值作为流量输出之设备号。输出的内容包括各个计算单元由于贮水量变化而引起的流量, 流入和流出各个定水头计算单元的水量以及计算单元间的渗流量。

IBCFCB=0: 不打印存写输出各流量项。

IBCFCB<0: 打印输出但不存写各流量项。贮水量变化引起的流量和计算单元间渗流量亦不打印。

LAYCON: 含水层类型标识码。每个输入数字表示一个模型层。从第一层开始逐层输入。最多可有80层。对没有使用的层则留为空白。

LAYCON=0: 承压含水层。导水系数和贮水系数均为常数。

LAYCON=1: 非承压含水层。仅用于模型的最上层。导水系数随饱和厚度和渗透系数而变化。贮水系数为常数。

LAYCON=2: 承压/非承压含水层。导水系数为常数。贮水系数和给水度的选用取决于含水层是否处于承压态。当非饱和带出现时, 垂向渗流量的计算需要特别处理。

LAYCON=3: 承压/非承压含水层。导水系数随饱和厚度而变化。MODFLOW通过渗透系数与饱和厚度的乘积来计算导水系数。贮水系数和给水度的选用取决于含水层状态。当非饱和带出现时, 垂向渗流量的计算需要特别处理。

TRPY: 各层之平面各向异性比率。由一维数组形式输入。每层的各向异性比率为一常数。MODFLOW根据各层之各向异性比率求出渗透系数或导水系数沿列方



向的值。对各向同性之含水层，则各向异性比率为1。这个数组中每个值对应模型的一个分层。

- DELR: 沿行方向的列间距。每列一个值。共有NCOL个值。
- DELC: 沿列方向的行间距。每行一个值。共有NROW个值。
- sf1: 第一类贮水系数。仅用于非稳定流计算 (ISS=0)。当LAYCON=1时，其值为给水度或释水率。当LAYCON等于2或3时，其输入值应为承压条件下的贮水系数。LAYCON=0时，sf1的值一般为贮水系数。注意，当降深很小、没有来自上部的越流补给时，含水层类型0也可以用来近似模拟潜水含水层。在这种条件下，sf1的值应该为该层的给水度。
- Tran: 沿行方向的导水系数。仅当LAYCON等于0或等于2时才输入导水系数。
- HY: 沿行方向的渗透系数。仅当LAYCON等于1或等于3时才输入此数据。
- BOT: 含水层底板标高。仅当LAYCON等于1或等于3时才输入此数据。
- VCONT: 垂向水力传导率。其值等于垂向渗透系数与含水层厚度之比。垂向水力传导率指本层与其下伏层之间的水力联系。模型之最底层不需要定义垂向水力传导率。
- sf2: 第二类贮水系数。仅当LAYCON等于2或3，并且非稳定流计算时，才输入此数据。sf2的值为非承压含水层的给水度。仅用于贮水系数与给水度的转换。
- TOP: 含水层之顶板标高。仅当LAYCON等于2或3，才输入此数据。

## BCF子程序包输入样单

数据项	说明	输入记录
1	{ISS, IBCFCB}	0 0
2	{LAYCON}	1 3 0
3	{TPRY数组控制记录}	31 1.0 (3F5.0)
	{第1、2、3层TRPY的值}	1.0 1.0 4.0
4	{DELR数组控制记录}	0 1000.0
5	{DELC数组控制记录}	31 1000.0 (10F5.0)
	{各行的DELC的值}	2.0 2.0 1.5 1.5 1.5 1.0 1.0 1.0 0.7 0.5
6	{sfl数组控制记录,第一层}	0 0.2
8	{HY数组控制记录,第一层}	0 10.0
9	{BOT数组控制记录,第一层}	0 -100.
10	{VCONT数组控制记录,第一层}	0 0.005
6	{sfl数组控制记录,第二层}	0 0.00007
8	{HY数组控制记录,第二层}	0 40.0
9	{BOT数组控制记录, 第二层}	31 1.0 (12F5.0)
		- 201 - 205 - 210 - 217 - 223 - 230 - 237 - 245 - 253 - 264 - 273 - 284
		- 201 - 204 - 211 - 216 - 224 - 231 - 237 - 244 - 252 - 265 - 273 - 285
		- 200 - 204 - 211 - 216 - 222 - 231 - 236 - 244 - 253 - 265 - 272 - 285
		- 199 - 204 - 212 - 218 - 222 - 232 - 238 - 244 - 252 - 263 - 273 - 283
		- 202 - 205 - 212 - 216 - 222 - 233 - 237 - 243 - 252 - 263 - 272 - 282
		- 201 - 206 - 212 - 218 - 224 - 231 - 237 - 245 - 252 - 265 - 274 - 283
	第二层底板标高	

-202	-206	-211	-216	-222	-231	-236	-244	-252	-263	-272	-285	
-202	-204	-211	-217	-222	-231	-235	-244	-252	-263	-272	-282	
-198	-204	-210	-215	-222	-233	-236	-242	-253	-263	-273	-283	
-203	-206	-214	-217	-224	-233	-235	-245	-255	-265	-275	-286	
10	{VCONT数组控制记录,第二层}											
11	{sf2数组控制记录,第二层}											
12	{TOP数组控制记录,第二层}											
6	{sf1数组控制记录,第三层}											
7	{Tran数组控制记录, 第三层}											
	0	0.001										
	0	0.15										
	32	1.0 (15F7.2)									0	
	0	0.00002										
	31	10 (12F6.0)									4	
	50	65	80	95	110	125	140	155	170	180	200	215
	70	85	100	115	125	150	160	180	190	203	223	234
	85	100	115	132	140	165	176	201	209	220	243	252
	101	113	124	139	148	176	190	213	231	239	260	270
	122	133	142	158	176	195	210	234	250	260	282	300
	137	140	175	175	192	210	224	247	263	275	300	314
	152	160	175	190	208	223	242	261	278	282	312	326
	162	172	187	201	220	230	251	270	291	295	324	333
	170	180	195	209	228	238	259	278	299	303	332	341
	175	186	201	215	234	243	264	283	304	308	338	347

[译注: 样本中的数据可能不符合格式要求, 仅供参考。]

## 第六章 河流子程序包

## 概念及程序化

河流及溪流是向地下水系统提供水源还是排泄地下水，取决于河溪与地下水之间的水力梯度。使用河流子程序包的目的是模拟地面水与地下水系统间的水流。因此，地下水流动方程（公式(26)）中受渗流影响的每个计算单元必须添加能表明地下水渗流流出地表或地面水入渗的参数项。

图32将一条河流分成若干河段，每段完全被包含在一个单独的计算单元之中。河流与含水层之间的水力联系由各河段与其所在的计算单元之间的渗流来模拟。

图33-a的剖面图表示开敞的河流与地下水系统之间夹有一层低渗透性的河床底积物的状况。图33-b将该系统加以理想化，河流—含水层间水力联系用一个简单的水力传导系数表示，并且视之为一维流动。图33所示的系统有利于说明河流—含水层水力联系的模拟并将其概念化。然而在很多场合下，这种低渗透性河床底积层可能并不存在，但只要对所使用的各项条件及参数进行适当的推断分析，由图33的概念所形成的模拟方法仍然能够用来表达这些情形。

图33-b表示一个包含有河段的计算单元之横剖面，其理想化的水力传导系数图解示于图34中。用来计算水力传导系数块体的长度为通过该计算单元的河段长， $L$ ；宽为河床宽度， $W$ ；渗流距离为河床底积层厚度， $M$ ；河床底积物的渗透系数由 $K$ 表示。假定可测定的河流与含水层间水头损失仅产生于河床底积层本身，即在河床底积层的底面和其下面的模型计算单元表示的位置之间无明显的水头损失。并且进一步假定，河床底积层之下对应的计算单元保持完全饱和状态，即河水位不会低于河床底积层底面以下。在这些假定下，河流和地下水系统之间的流量为：

$$QRIV = \frac{KLW}{M}(HRIV - h_{i,j,k}) \quad \text{或} \quad (63-a)$$

$$QRIV = CRIV(HRIV - h_{i,j,k}) \quad (63-b)$$

这里 $QRIV$ 是河流与含水层之间的流量，水流由河流流向含水层时取正值； $HRIV$ 是河流的水位； $CRIV$ 是河流—含水层互相连接的水力传导系数（ $KLW/M$ ）； $h_{i,j,k}$ 是河流河段所在

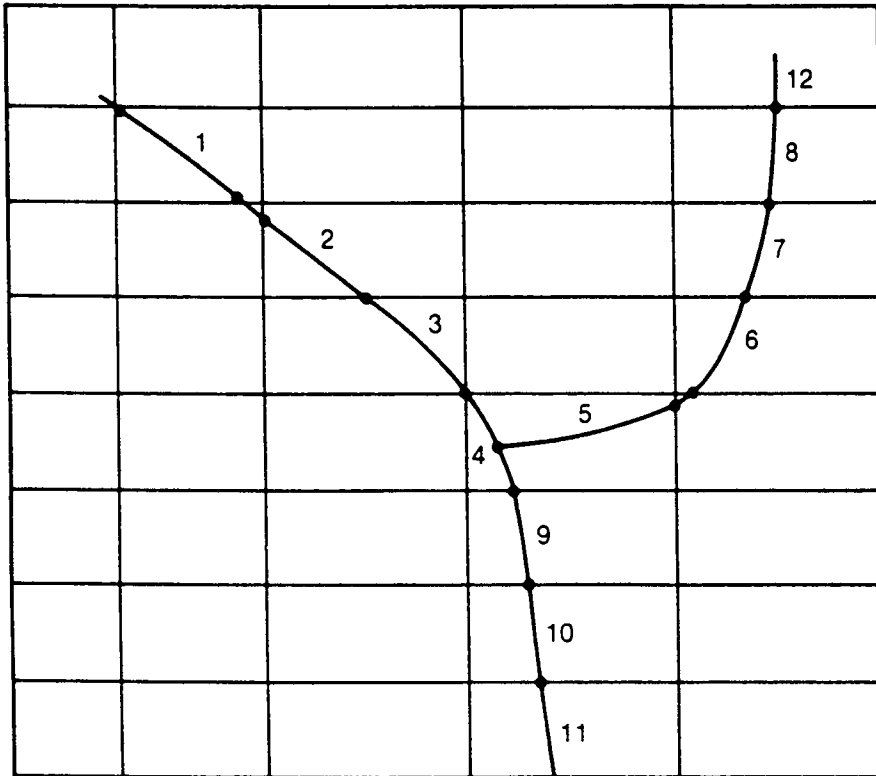
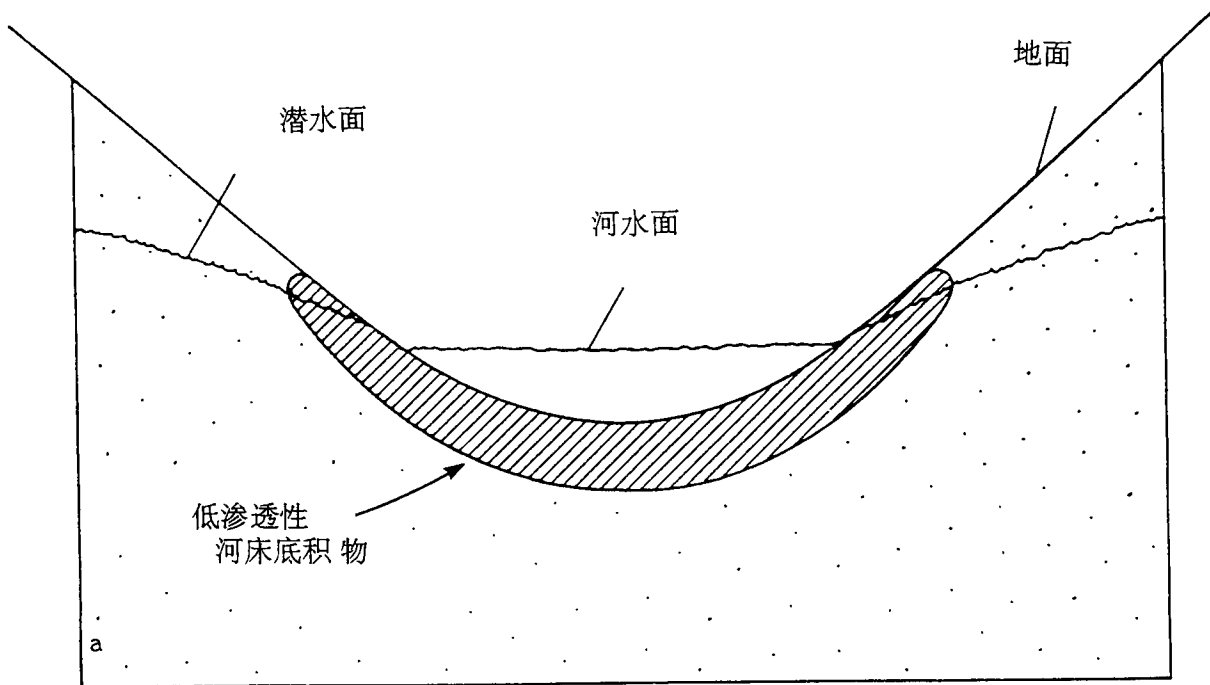
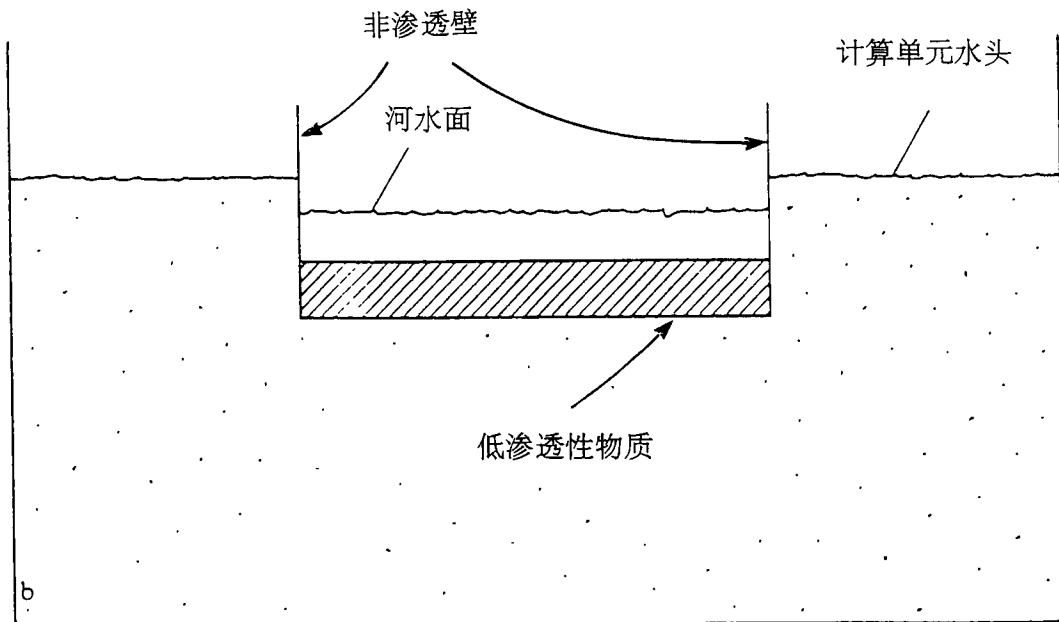


图 32. 河流离散为河段的示意图。一些小河段被忽略。



计算单元边界



计算单元边界

图 33 (a) 有河流出现的含水层横剖面图和 (b) 河流—含水层相互连接概念的示意图。

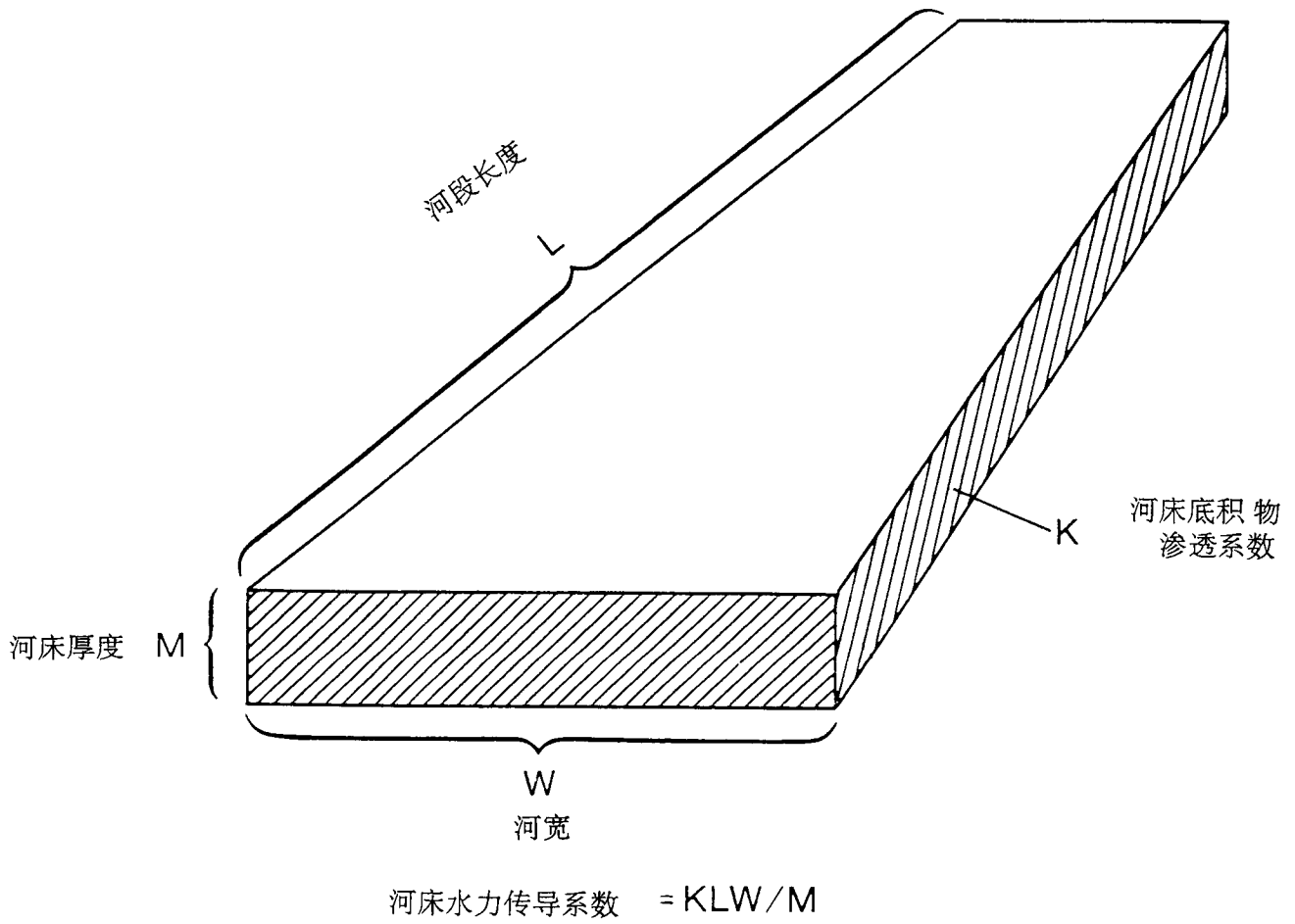


图 34. 单个计算单元中河床水力传导系数的理想化模型示意图。

的计算单元的水头。

若满足假定条件，即所有水头损失都产生于水流通过的河床底积层，那么公式(63)的应用是直观明了的。然而，在更为通常的情况下，公式(63)必须同时适用于河床没有底积物的情形，或水头损失不局限于河床底积层这一层的情况。遇到这些情形时，务必单独列出水力传导系数这一项， $CRIV$ ，将其代入公式(63-b)中，把河流至计算单元 ( $i,j,k$ ) 所在的深度位置间的流量，与相应的水头损失联系起来。这种类型的水流通常为三维流过程。通过一个单独的水力传导系数来表达这一水流，不过是一种近似的做法。如果有可靠的野外河流渗流测量及其相应的水头差资料，则可用于计算有效水力传导系数。不然，则要选择一或多或少随意的水力传导系数值，并在模型校正过程中加以调整。在工作中，选择水力传导系数起始值，可遵循一定的规律。例如，所假定的水流横切面面积通常应该与计算单元内河道宽度和河段长度之乘积为同一数量级；所假定的渗流距离不应超过河床与计算单元 ( $i,j,k$ ) 的垂直间距。另外，如果在这个间距内可辨别区分出不同的层来，这些层通常应当作一系列串联的水力传导系数来处理，以便求得一个等价水力传导系数。可是，总的来说，应当认识到，仅用水力传导系数这一项来表示一个三维流过程，这种方法本身是一个经验做法，在模型校正时几乎总是需要进行调整。

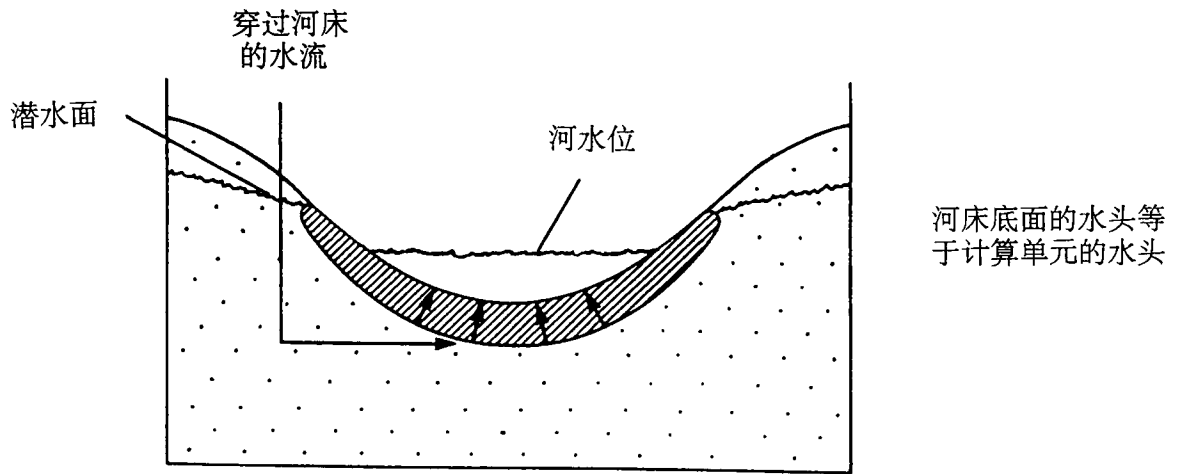
当含水层水头处于一定范围内时，用公式(63)模拟河流—含水层水力联系通常会得到可以接受的近似结果。可是，在大多数情况下，当含水层水头下降到一定位置之下时，河流渗流不再依赖于含水层中的水头。回到河床底积层这一概念，便可想象到这一情形。公式(63)所描述的情形示于图35-a中；含水层水位高于河床底积层的底面时，流经该底积层的流量与河流和含水层的水头差成正比。在图35-b中，含水层水位已经下降至河床底积层底面之下，在其下形成了一个非饱和带。如果假定河床底积层本身保持饱和，河床基底处某点的水头可简单地视为该点的高程。若该高程指定为 $RBOT$ ，穿过河床底积层的流量可由下式给出：

$$QRIV = CRIV(HRIV - RBOT) \quad (64)$$

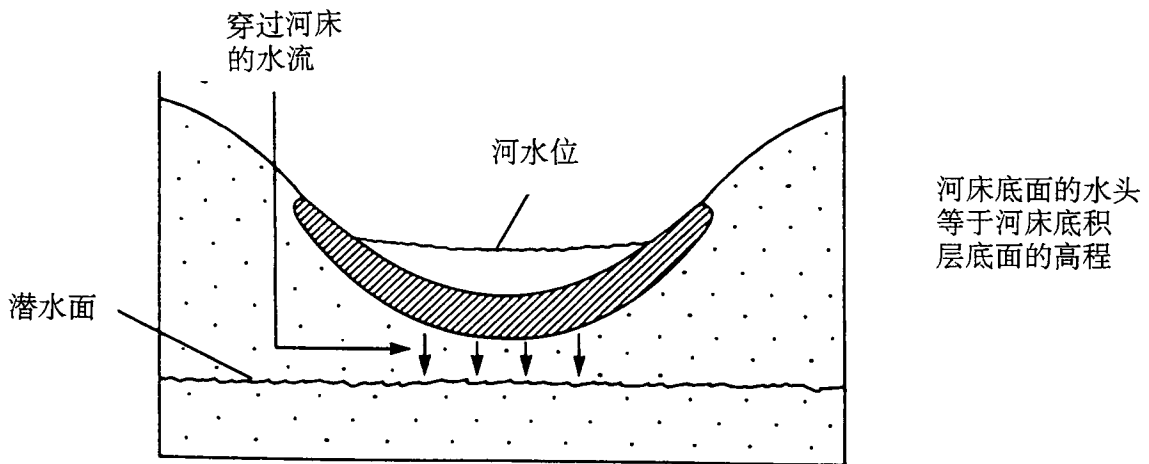
这里 $QRIV$ ， $CRIV$ ，以及 $HRIV$ 已在公式(63-b)中作了定义。显然，水头低于 $RBOT$ 之后，再继续下降不会增加穿过河床底积层的流量，因为我们已经假定水头损失主要由河床底积层产生。只要水头保持在 $RBOT$ 之下，流量会保持由公式(64)所给定的常量值。以下描述的模型，便利用了这些概念来模拟河流—含水层的水力联系：即河流与计算单元 ( $i,j,k$ ) 之间流量的模拟由下面一套方程来实现：

$$QRIV = CRIV(HRIV - h_{i,j,k}), h_{i,j,k} > RBOT$$





A



B

图35. 河床底积层底面之水头与计算单元水头关系的剖面示意图。计算单元之水头等于潜水面高程。

$$QRIV = CRIV(HRIV - RBOT), h_{i,j,k} \leq RBOT \quad (65)$$

河流与计算单元 (i,j,k) 之间的流量, 以水头  $h_{i,j,k}$  作为变量, 按公式(65)计算得到的结果示于图36。当  $h_{i,j,k}$  等于河流水位  $HRIV$  时, 流量为零。当  $h_{i,j,k}$  值变大时, 流量取负值, 也就是说, 地下水流向河流; 当  $h_{i,j,k}$  值变小时, 流量取正值, 也就是说, 河水流向含水层。在  $h_{i,j,k}$  达到  $RBOT$  之前, 这个正流量随  $h_{i,j,k}$  降低而线性增加, 此后, 流量保持为常量值。

无论河床底积层出现与否, 类似公式(65)及图(36)的河流—含水层水力联系通常普遍可见。例如, 一旦在河流与含水层之间出现非饱和的情况, 则河流向含水层的渗流与含水层的水头无关。而且在多数情况下, 甚至在非饱和带出现之前, 这种独立的不相关关系就已经建立起来。图37中含水层水位已经降到河床之下足够低的位置, 在河床与区域地下水之间呈现一个窄小的饱和连接带。细心观察图37, 便会发现该饱和连接带的水头梯度必趋于单位值, 进一步降低区域地下水面将不会增加该梯度。因此, 类似于图37的情况一旦形成, 河流渗流量与含水层的水头进一步下降无关。图37所示的情形本身是野外状况的一种简化。野外也许常常涉及到河流下有饱和与非饱和的复杂组合形式。可见, 在所有这些情况下, 河流的渗流量必然会在某点过渡为与含水层水头无关, 不受水头继续下降的影响。

如果水文状况表明, 河流渗流量会随当地地下水面高程下降而增加, 但当地下水位达到图37所示的高程  $h_l$  时, 河流渗流量会达到极限, 这时  $RBOT$  应取值  $h_l$ 。因为图37中河流之下的垂向水力梯度近似等于1。自河流进入计算单元 (i,j,k) 的渗流量由  $KLW$  的乘积近似给出, 这里  $K$  为河流下面饱和介质的垂向渗透系数, 与前述相同;  $L$  为穿过计算单元 (i,j,k) 的河段长;  $W$  为河宽。与上述极限渗流的估计值及  $RBOT$  选用的值连贯对应的一项, 河流—含水层水力传导系数  $CRIV$ , 可通过  $KLW$  替代  $QRIV$  而求解出来。由此得出:

$$CRIV = \frac{KLW}{HRIV - RBOT} \quad (66)$$

概括地说, 如果可以预料河流渗流的极限条件符合图37的模型, 向极限渗流的过渡预料会在某潜水面高程, 那么应该选择该潜水面高程为  $RBOT$ , 而且从(66)式可计算出  $CRIV$  的值。基于公式(65)的处理方法应该可以合理地近似模拟河流—含水层的水力联系。

在河流—含水层水力联系的简化模型中, 我们假定这种水力联系与河段在计算单元内

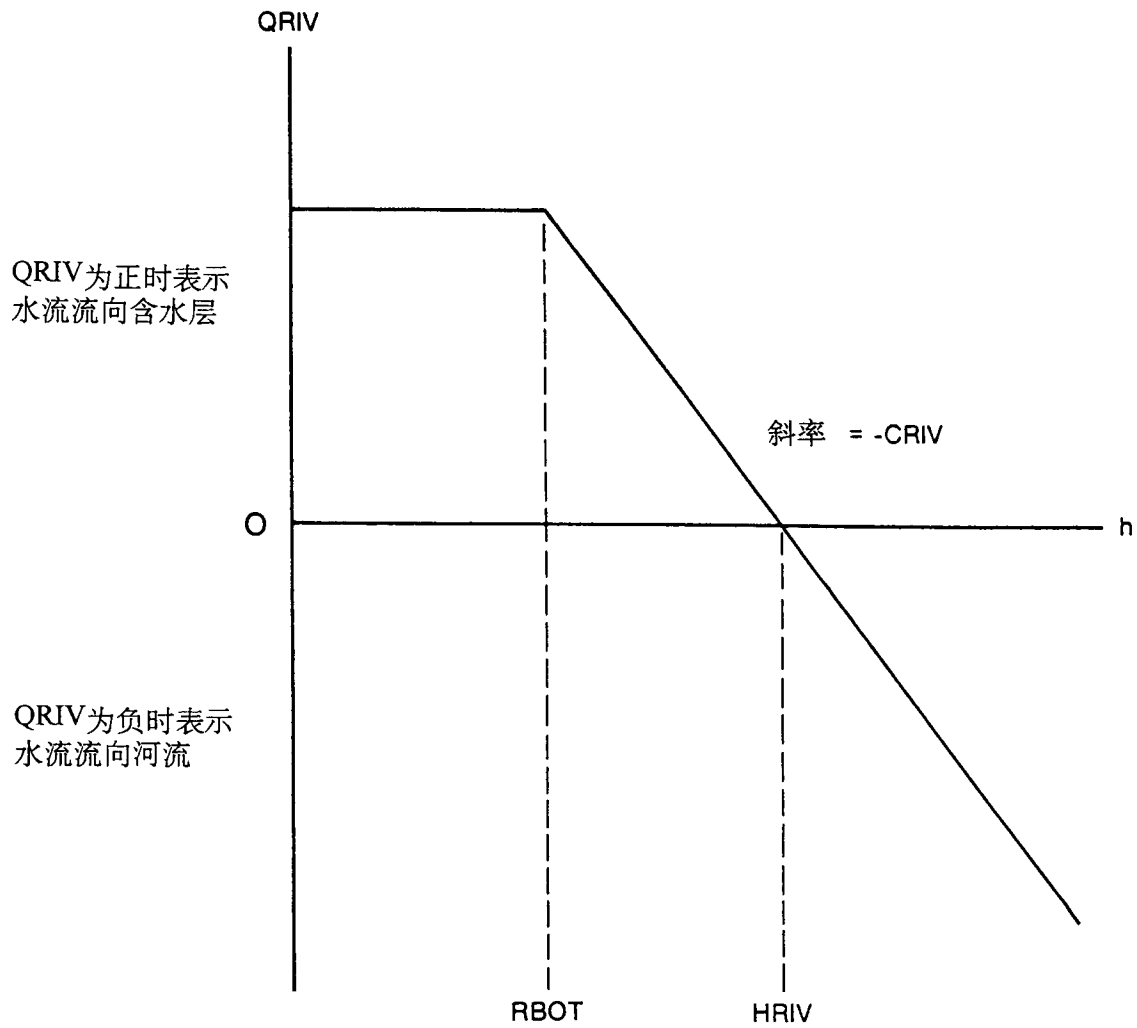
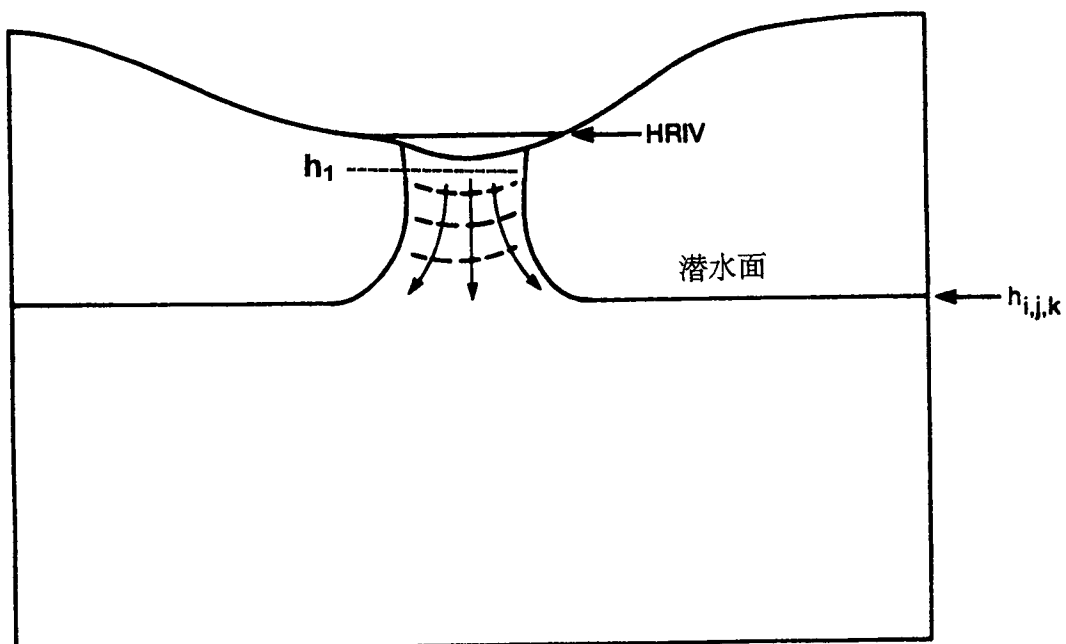


图36. 河流流向某计算单元的流量,  $QRIV$ , 与该单元之水头变量的关系示意图。  
 $RBOT$ 为河床底积层底面高程,  $HRIV$ 为河水位。



----- 等水头线

图 37 单位水力梯度下的河流极限渗流示意图。

的位置无关，每一河段内河流水位是均匀一致的，且在每个应力期内都为常量。后一假定意味着河流的状况在每个应力期内不发生显著变化—例如，河流不会变为干枯状态或漫出河岸，或这样的状况虽然发生但持续时间很短，不致于影响河流—含水层的水力联系。

描述每条河流段的数据由用户在每个应力期内定义。每个河流段的输入数据有六项，包括河段所出现的计算单元的层号、行号和列号，以及计算渗流要用到的三个参数—河水位( $HRIV$ )，河流—含水层相互连接的水力传导系数( $CRIV$ )，和“河床底面高程”，或河流渗流达到极限时的高程位置( $RBOT$ )。

对于有河段出现的每个计算单元，每次迭代开始时，在流动方程中都相应添加河流渗流项。是选用公式(63)还是(64)来计算河流渗流，则通过比较该计算单元的最新水头值和河底标高 $RBOT$ 的值而决定。因为这个选择过程在每次迭代开始时就已完成，最新的水头值( $HNEW$ )是上一次迭代求得的值。因此，检查判断究竟使用哪一个河流渗流方程，比起渗流的计算要滞后一次迭代。如果选用公式(63)则 $CRIV$ 这一项添加到 $HCOF$ 中，而将 $CRIV * HRIV$ 这一项添加到 $RHS$ 中。如果选用公式(64)，则将 $-CRIV(HRIV-RBOT)$ 添加到 $RHS$ 中。

## 河流子程序包输入数据及格式

河流(RIV)子程序包的输入由IUNIT(4)所指定的文件设备号读入。

对每次模拟:

由子程序RIV1AL读入的数据包括

- 1.数据名称: MXRIVR    IRIVCB  
输入格式: I10        I10

对每个应力期:

由子程序RIV1RP读入的数据包括

- 2.数据名称: ITMP  
输入格式: I10
3. 数据名称: 层号    行号    列号    河水位    水力传导系数    河床底面高程  
                  Layer    Row    Column    Stage    Cond                    Rbot  
输入格式: I10    I10    I10    F10.0    F10.0                    F10.0  
(该输入为每一河段为一记录。若ITMP为负或零, 则不予输入本项。)

## 输入数据说明

MXRIVR: 各应力期中有效河段的最大数目。

IRIVCB: 标示符和文件设备号。

若IRIVCB>0, 是文件设备号。当ICBCFL (见输出控制) 被设定时, 将各计算单元的河流渗流量按该设备号输出。

若IRIVCB=0, 将不打印或输出单元间河流渗流量。

若IRIVCB<0, 当ICBCFL被设定时, 在标准输出文件中打印各河流段的流量。

ITMP: 标示符和计数器。

若ITMP<0, 重复使用上一应力期的河流数据。

若ITMP≥0, ITMP是当前应力期中有效河段的数目。

- Layer:** 含有该河段计算单元所在的层号。
- Row:** 含有该河段计算单元所在的行号。
- Column:** 含有该河段计算单元所在的列号。
- Stage:** 河流的水位高程。
- Cond:** 河床水力传导系数。
- Rbot:** 河床底积层底面的高程。

## 第七章 补给子程序包

## 概念及程序化

使用补给子程序包 (RCH) 的目的是模拟地下水系统的面状补给。面状补给的通常是由降水入渗补给地下水系统形成。本模型中的补给定义为:

$$Q_{Ri,j} = I_{i,j} * DELR_j * DELC_i \quad (67)$$

其中,  $Q_{Ri,j}$  是水平面上某一计算单元(i,j)的补给率, 用单位时间水体积量表示。 $I_{i,j}$  是施加在该计算单元面积  $DELR_j * DELC_i$  上的补给通量 (用单位时间长度单位表示)。补给率  $Q_{Ri,j}$  施加到位于(i,j)的柱体上某一单个计算单元。没必要让补给发生在同一垂向柱体的不同深度上, 这是因为自然补给多是从顶部进入地下水系统的。最简单的情形是地下水系统的上界面出现在第一层内; 可是, 当潜水面升高或降低时, 地下水系统的上界面在各点的位置将随时间而变化。稍后将叙述如何通过MODFLOW规定的三种选择, 对接受补给的每个垂向柱体指定接受补给的计算单元。补给(RCH)子程序包可用于模拟降水以外的其它水源补给, 如人工补给。若需要在同一垂向柱体上不止一个计算单元接受补给, 那么可用井流子程序包。井流子程序包允许对模型中任一计算单元进行补给或排泄。

补给子程序包将每个应力期 (除非选择了使用前一应力期的补给通量) 的补给通量  $I_{i,j}$  读入到一个二维数组  $RECH_{i,j}$  中, 然后将补给通量的值乘以计算单元在水平面上的面积  $DELR_j * DELC_i$ , 求得  $Q_{Ri,j}$  的值, 并将其储存在  $RECH$  数组中。如何选择垂向柱体上接受补给的计算单元, 可通过补给方式选择符  $NRCHOP$  和补给层号数组  $IRCH$  来指定。补给方式选择项包括: (1) 补给指定到模拟层第一层; (2) 补给指定到垂向柱体上任一计算单元, 并由二维数组  $IRCH_{i,j}$  中的层号确定; (3) 假若计算单元上方没有定水头计算单元, 补给指定到位于最上面的有效计算单元。对于第一种以及第二种选择, 如果指定接受补给的计算单元是不透水的, 实际上没有接受补给。至于第三种选择, 因为已经假定补给将被定水头水源截获, 如果垂向柱体上有定水头计算单元以及其上方有无效计算单元, 那么该垂向柱体实际上没有采纳补给。模型读取的补给通量, 必须采用与其它模型参数相一致的长度和时间单位, 如米/天。

给定水平面上计算单元位置(i, j)以及由补给选择项所确定的垂向位置k, 相应的补给



量  $Q_{Ri,j}$ ，将会在矩阵方程求解时，从  $RHS_{i,j,k}$  (公式(26)或(29)) 中减去。每次迭代都对所有接受补给的计算单元进行此项计算。按原有定义，补给是不受含水层水头影响的，所以水头系数  $HCOF_{i,j,k}$  不添加任何值。

在实际工作中，应仔细考虑所研究的问题及上述三种选择，再决定选用哪一种选择。例如，图38所示情形，用一个横剖面模型来模拟补给，河流向地下水渗流，以及地下水渗流进入河流（图38-a）。按照第五章描述的潜水水平方向水力传导系数计算的规定，模型中有效计算单元在模拟过程中被逐渐调整以近似表达地下水水面，结果每个垂向柱体中最上方的有效计算单元大致落在潜水面。这个过程给出了有效、定水头、以及无效计算单元的最终分布情况，见图38-b。

图38-c说明采用上述第一种选择时，模型的补给分布情况。该选择只允许对模型最顶层补给。当潜水面低于第一层之底面标高时，含水层将无法获得应有的补给。这样做显然会低估整个潜水面所获得的补给量。

图38-d说明采用第二种选择时的补给分布。这里假定用户在模拟之前已根据预先估计的潜水面的位置而指定了接受补给的计算单元。但由于预先估计的潜水面与模拟所得的最终结果有些出入，用户指定为接受补给的计算单元中，有四个已转换成无效计算单元，因而无法接受补给。

图38-e是采用第三种选择时的模拟结果。这种选择最适合这种特定情形。在这项选择下，除定水头计算单元用来表示河流外，补给进入每个垂向柱体最上面的有效计算单元。从而模拟了补给条件下的潜水面形状。

就典型的降水补给来说，第三种选择最为常用。用户不需判定哪一个是垂向柱体中最上面的有效计算单元。因为程序具备了在模拟过程中对此作自动判断的功能。不过，在第一层中有不透水计算单元的地方，补给不应该穿透至下伏层位时，也用得着第一种选择，否则MODFLOW会将补给加至不透水层之下的含水层之中。当然，第三种选择仍然可用于这种情形，只要将不可透水的计算单元的补给率指定为零即可。用户应选用最方便指定输入数据的选择项。与此类似，当第一层以外的层位露出地表，同时指定层位的补给不应该穿透不透水计算单元进入下伏含水层，或许用得着第二种选择。在衡量所作选择是否最佳时，应考虑的其他因素是，比起第一、三种选择来，第二种选择占用更多的计算机存储空间。

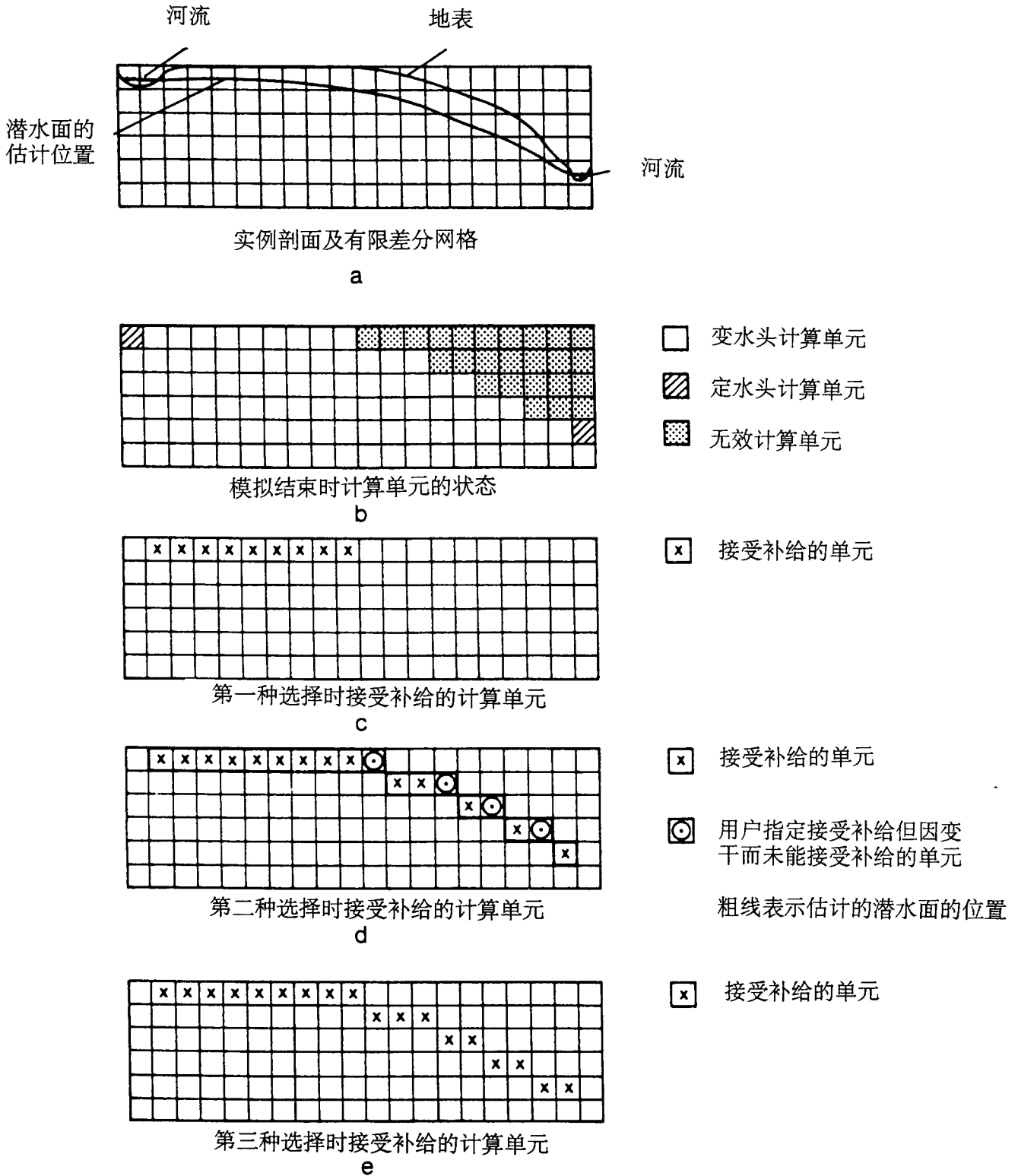


图38. 补给子程序包中三种选择下接受补给的计算单元。

间，而第三种选择比第一、二项选择需用较多计算时间。

### 补给子程序包输入数据及格式

补给 (RCH) 子程序包的输入从IUNIT(8)指定的文件设备号读取。

对每次模拟:

由子程序RCH1AL读取的数据包括

1. 数据名称: NRCHOP      IRCHCB  
输入格式: I10            I10

对每个应力期:

由子程序RCH1RP读取的数据包括

2. 数据名称: INRECH      INIRCH  
输入格式: I10            I10
3. 数据名称: RECH (NCOL, NROW)  
读取模块: U2DREL

若补给方式选择符等于2

4. 数据名称: IRCH (NCOL, NROW)  
读取模块: U2DINT

### 输入数据说明

NRCHOP: 补给方式选择符。补给通量由一个二维数组给出, 每个垂向柱体取一个值。于是, 每个垂向柱体均有某一个计算单元接受补给, 而选择项代号则确定垂向柱体中哪一个计算单元接受补给。

1—补给仅至模型最顶层;

2—补给的垂向分布由数组IRCH指定;

3—由每个垂向柱体中位置最高的有效计算单元接受补给。定水头计算

单元截断补给, 并防止其向更深部位渗透。

IRCHCB: 标示符和设备号。

若IRCHCB>0, 其值为设备号。当ICBCFL (见输出控制) 被设定时, 将各计算单元的补给量按该设备号输出。

若IRCHCB≤0, 将不打印或输出计算单元间补给量。

INRECH: RECH数组的读取标示符。

若INRECH≥0, 读取补给通量数组RECH。

若INRECH<0, 重复使用上一应力期的补给率。

INIRCH: IRCH数组的读取标示符。

当NRCHOP为2时, 若INIRCH≥0, 读取补给层号数组 (IRCH)。

若INIRCH<0, 重复使用上一应力期的补给层号数组IRCH。

注: 当NRCHOP为1或3时, INIRCH则被忽略。

RECH: 补给通量 ( $LT^{-1}$ )。仅在INIRCH大于或等于零时读取。

IRCH: 补给层号数组, 决定垂向柱体中接受补给的层号。仅当NRCHOP为2且INIRCH大于或等于零时读取。

使用第一种选择时的补给子程序包的输入样单

数据项	说明	输入记录
1	{NRCHOP, IRCHCB}	1 0
2	{INRECH}第一应力期	1
3	补给通量数组的控制记录	12 3.17E-8 (10F4.0) 4
		1.0 1.0 1.0 1.0 1.0
		1.0 1.0 1.0 1.0 1.1
		1.0 1.0 1.0 1.0 1.1
	补给通量	1.0 1.0 1.0 1.1 1.1
		1.0 1.0 1.1 1.1 1.1
		1.1 1.1 1.1 1.1 1.1
2	{INRECH}第二应力期	-1
2	{INRECH}第三应力期	-1
2	{INRECH}第四应力期	1
3	补给通量数组的控制记录	12 3.17E-8 (10F4.0) 4
		1.2 1.2 1.2 1.2 1.3
		1.2 1.2 1.2 1.3 1.4
		1.2 1.2 1.3 1.4 1.4
	补给通量	1.0 1.0 1.0 1.1 1.1
		1.2 1.3 1.3 1.4 1.4
		1.3 1.3 1.4 1.4 1.4

使用第二种选择时补给子程序包的输入样单

数据项	说明	输入记录
1	{NRCHOP, IRCHCB}	2 0
2	{INRECH, INIRCH}第一应力期	1 1
3	补给通量数组的控制记录	0 3.17E-8
4	补给层号数组的控制记录	12 1 (20I2) 3
		1 2 2 2 3
		1 2 2 2 2

		1	1	2	2	2
		1	1	1	1	2
		1	1	1	1	1
		1	1	1	1	1
2	{INRECH, INIRCH}第二应力期	1				-1
3	补给通量数组的控制记录	0	1.56E-8			
2	{INRECH, INIRCH}第三应力期	-1				-1

[译注: 样单中的数据可能不符合格式要求, 仅供参考]。

## 第八章 井流子程序包

## 概念及程序化

在MODFLOW中, 使用井流子程序包是为了模拟井流对地下水系统的影响。在一个应力期内, 并以给定流量从含水层中抽水(或向其注水), 其流量不受井所在计算单元的大小及水头影响。在本章的讨论中, 我们假定模拟的对象为实际的抽水井或注水井。

井流子程序包的操作是, 在每个模拟应力期间, 并以指定流量从含水层抽水或向含水层注水。负的流量值 $Q$ 表示抽水井, 而正的流量值则表示注水井。

每个应力期开始时, WEL1RP模块为每个井读取4个值—即井所在计算单元的行号、列号、层号, 以及抽水或注水的流量,  $Q$ 。在每次迭代求解矩阵方程时, MODFLOW都将从每个井所在的计算单元的RHS值(公式(26)或(29))中减去流量 $Q$ 的值。在多口井位于同一个计算单元里的情况下, 在求RHS这项值时, 每口井都分别进行此项计算。因此, 用户给一个计算单元内各井所赋的流量值, 在程序运行时, 实际上被累计起来以获得该计算单元总的井流量。对这样类型的多层井, 可用一组单层井表示, 即每个井只与多层井所穿透的层中的某一层相连通, 每个井在每一应力期中都定义有一单独流量项 $Q$ 。若采用这种方法, 多层井的流量必须以某种方式人为地分配给每个单层, 而不是靠程序本身进行分配。常用的方法是把井流量按每层的导水系数大小分配, 即:

$$\frac{Q_1}{Q_w} = \frac{T_1}{\sum T} \quad (68)$$

其中 $Q_1$ 是给定应力期某特定井自第一层的抽水量,  $Q_w$ 是这一应力期该井的总抽水量,  $T_1$ 是第一层的导水系数,  $\sum T$ 则代表该井穿透的所有层位的导水系数总和。再次提醒注意, 用公式(68), 或其它办法分配抽水量, 应由用户对每应力期每个穿透多层的井人为地进行, 并作为输入数据输给MODFLOW。

在现有的MODFLOW中, 一口多层井由多口不相联的单层井代替, 故不能反映由于井本身引起的层之间的水力联系; 所以不能完全代表实际情况。一个能较好解决多层井问题的子程序包正处于开发过程之中。[译注: 到现在为止, 原文中提到的新子程序包还没有问世, 但美国地质调查局Bennett等人于1982年提出的多层井的解决方法值得读者参考。]



### 井流子程序包输入数据及格式

井流子程序包 (WEL) 的输入从UNIT(2)所指定的文件设备号读取。

对每次模拟:

由子程序WEL1AL读取的数据包括

1. 数据名称: MXWELL IWELCB  
输入格式: I10 I10

对每个应力期:

由子程序WEL1RP读取的数据包括

2. 数据名称: ITMP  
输入格式: I10
3. 数据名称: Layer Row Col Q  
输入格式: I10 I10 I10 F10.0  
(该项输入为每口井一项记录。若ITMP为负或零, 则不输入本项。)

### 输入数据说明

MXWELL: 所有应力期中使用的最大井数。

IWELCB: 标示符和文件设备号。

若IWELCB>0, 是文件设备号。当ICBCFL (见输出控制) 被设定时, 将各井的流量从此设备号输出。

若IWELCB=0, 将不打印或输出井的流量。

若IWELCB<0, 当ICBCFL被设定时, 在标准输出文件中打印出各井的流量。

ITMP: 标示符和计数器。

若ITMP<0, 重复使用上一应力期的井流数据。

若 $ITMP \geq 0$ ,  $ITMP$ 是当前应力期中有效井的数量。

Layer: 该井所在计算单元的层号。

Row: 该井所在计算单元的行号。

Col: 该井所在计算单元的列号。

Q: 抽、注水流量。正值表示注水而负值表示抽水。

## 井流子程序包输入样单

数据项	说明	输入记录			
1	{MXWELL, IWELCB}	6	24		
2	第一应力期 {ITMP}	4			
3	第一口井 {层号, 行号, 列号, 流量}	2	5	6	-0.17
3	第二口井 {层号, 行号, 列号, 流量}	2	6	3	-0.23
3	第三口井 {层号, 行号, 列号, 流量}	2	7	7	-0.77
3	第四口井 {层号, 行号, 列号, 流量}	2	9	2	-0.32
2	第二应力期 {ITMP}	-1			
2	第三应力期 {ITMP}	-1			
2	第四应力期 {ITMP}	6			
3	第一口井 {层号, 行号, 列号, 流量}	2	6	5	-0.73
3	第二口井 {层号, 行号, 列号, 流量}	2	9	4	-0.32
3	第三口井 {层号, 行号, 列号, 流量}	2	5	6	-0.17
3	第四口井 {层号, 行号, 列号, 流量}	2	6	3	-0.43
3	第五口井 {层号, 行号, 列号, 流量}	2	7	7	-0.77
3	第六口井 {层号, 行号, 列号, 流量}	2	9	2	-0.32

[译注: 样单中的数据可能不符合格式要求, 仅供参考。]

## 第九章 沟渠子程序包

## 概念及程序化

使用沟渠子程序包的目的是模拟农用排水沟渠的排水效果。农用排水沟渠从含水层中排泄地下水，其所排水量正比于含水层的水头与某一固定水头或高程之差。其中要求含水层水头高于这一固定的高程，而当含水层水头低于该固定水头高程时，则无排水效果。本章叙述如何模拟实际的沟渠。

图39是某计算单元的横剖面图，用来说明沟渠模拟的概念模型。其中假定沟渠中仅部分有水，因而沟渠的水头大约等于其半高处的高程 $d_{i,j,k}$ 。模型所计算的计算单元 $(i,j,k)$ 的水头 $(h_{i,j,k})$ ，实际上是该计算单元的水头平均值，而且通常假定在沟渠一定距离之外的区域水头值在该计算值上下。沟渠的水头 $d_{i,j,k}$ 只局限于沟渠之内一而不是将其延伸至整个计算单元。在沟渠与含水层水头 $h_{i,j,k}$ 之间的区域，垂向面上存在辐射流或半辐射流，其特点通常为，越接近沟渠，水头梯度越大。在地下水流向沟渠过程中的水头损失是形成水头差 $h_{i,j,k} - d_{i,j,k}$ 的原因之一。另一方面，若沟渠附近的水力渗透系数与计算单元 $(i,j,k)$ 的平均渗透系数不同，那么在含水层与沟渠直接相邻的地方也会造成一部分的水头损失。通常这是因为渠道周围填充材料的透水性与含水层透水性有所不同而引起的（图40）。

最后，水头损失还发生在排水管壁，其值大小取决于管道开口的数目和开口的尺寸，以及这些开口被化学沉淀物，或植物根系阻塞的程度，等等。

上面讨论的三个过程（向沟渠的汇聚流动，流经与沟渠直接相邻的具有不同渗透系数的物质材料，以及流经排水管壁）每种情形引起的水头损失大小可假定与沟渠系统的排水量 $QD$ —即从计算单元 $(i,j,k)$ 排到沟渠的排水量成正比。因为这些水头损失以串联系列形式发生，其总水头损失 $h_{i,j,k} - d_{i,j,k}$ 也可认为与 $QD$ 成正比。沟渠子程序包已将该方法编入计算程序中。也就是说，已经假定排水功能由下述一对公式描述：

$$QD_{i,j,k} = CD_{i,j,k}(h_{i,j,k} - d_{i,j,k}) \quad \text{当 } h_{i,j,k} > d_{i,j,k} \quad (69)$$

$$QD_{i,j,k} = 0 \quad \text{当 } h_{i,j,k} \leq d_{i,j,k} \quad (70)$$

公式(69)中的系数 $CD_{i,j,k}$ 是个综合（或等效）水力传导系数，用来描述沟渠与计算单元 $(i,j,k)$ 的区间总水头损失。其中假定计算单元 $(i,j,k)$ 的水头为 $h_{i,j,k}$ ，其大小取决于流向沟渠

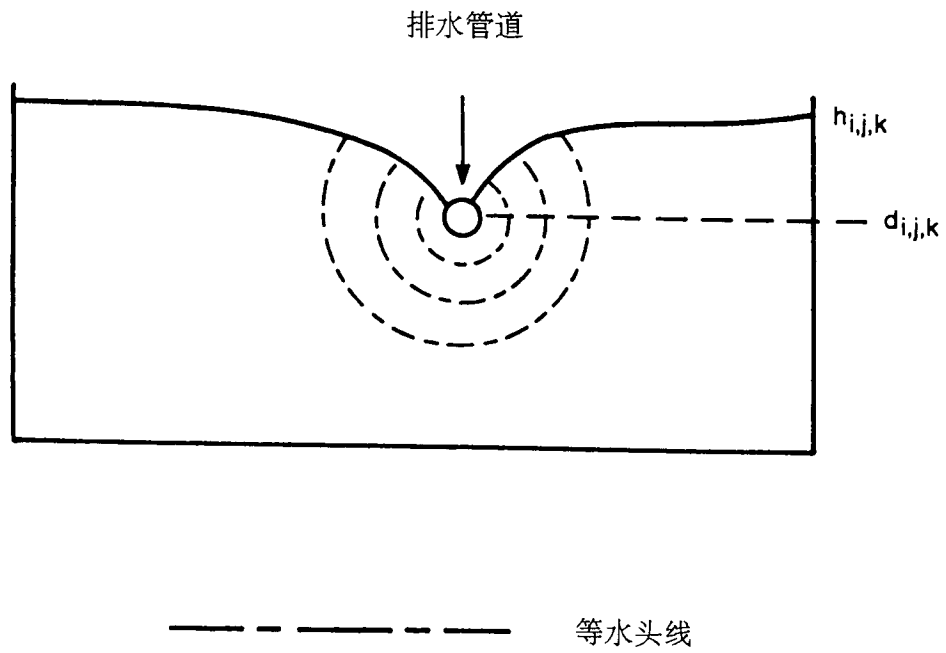


图39. 计算单元(i, j, k)的汇聚水流之水头损失横剖面示意图。

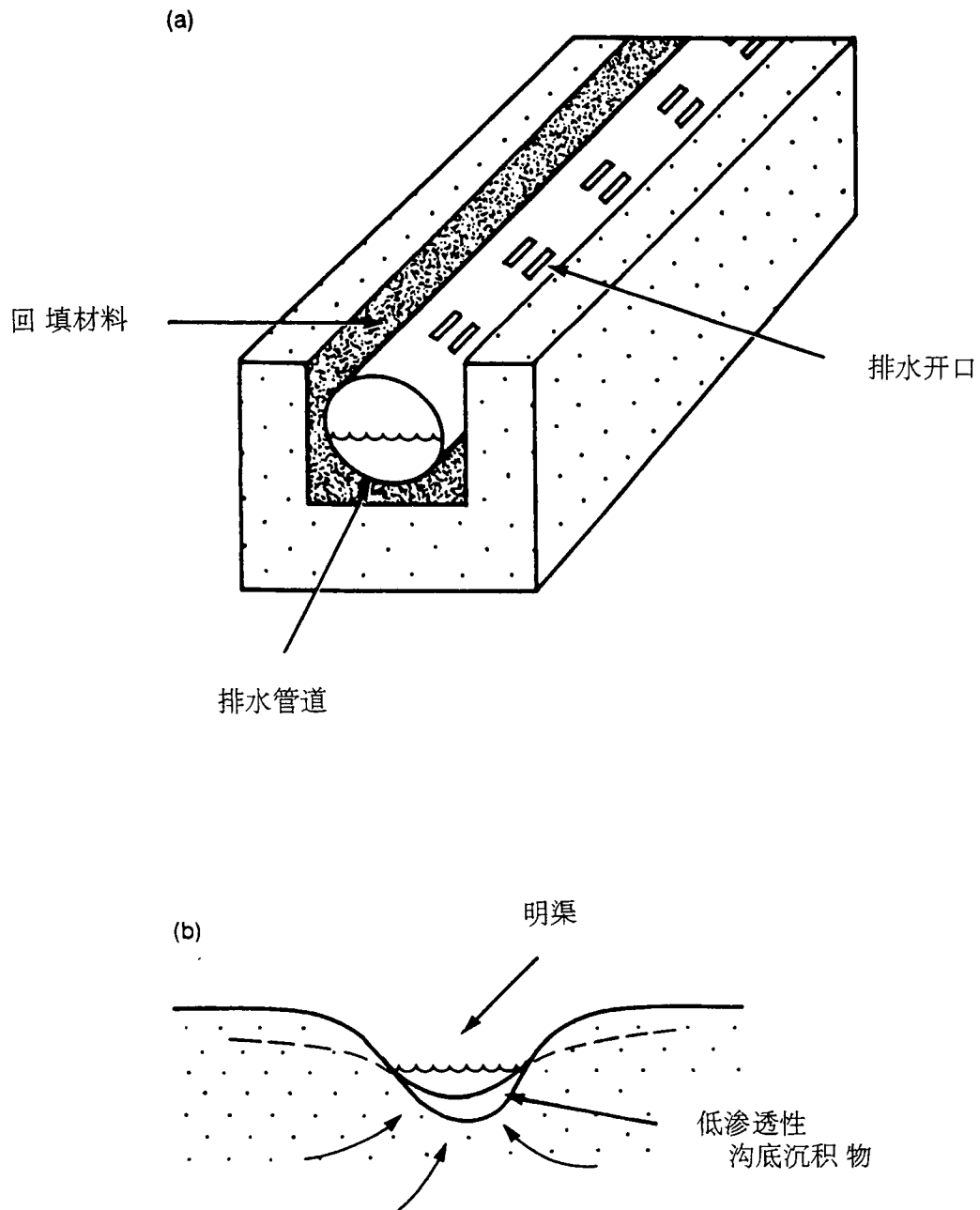


图40. 沟渠的直接相邻区域内水头损失的影响因素:

(a)回填沟中埋设的排水管道和(b)明渠。

的汇聚水流的形式特点、沟渠自身特点及其直接相邻的环境。

我们可以推导出近似表达这三种水流过程的水力传导系数的公式，求得 $CD$ 值，然后计算等价的系列水力传导系数值。每个过程的水力传导系数可用一维水流公式求得。取决于待模拟的沟渠系统的具体特点，所用的公式会很不相同。因此，这里不列出 $CD$ 的通用表达式。这样做的另一个原因是，在大多数情况下，这种类型明确的表达式要求详细但通常却无法得到的数据资料。例如沟渠周围的水头分布的详尽资料，沟渠附近含水层的水力传导系数，回填材料的分布，回填材料的水力渗透系数等。实践中，更为常见的做法是，根据实测的 $QD$ 值及 $h-d$ 值，用公式(69)计算 $CD$ 值。如果不知道 $h-d$ 的准确值，那么在模型校正时对 $CD$ 加以调整，直至 $QD$ 的值吻合实测值。

图41是公式(69)和(70)所定义的 $QD$ 对 $h_{i,j,k}$ 变量的函数曲线图；该函数和地表河流与含水层间流量的函数相似。不同之处是，沟渠只能排水而不允许水流流向含水层。 $QD$ 取正值以表示水流是流向沟渠。实际上，只要对河流子程序包的参数进行适当选择，同样也可用来模拟沟渠的排水效果。

因公式(69)中 $QD_{i,j,k}$ 项被当作是流出计算单元 ( $i,j,k$ ) 的流量，对每个受沟渠影响的计算单元，只要水头 $h_{i,j,k}$ 高于沟渠水位，必须将此项从方程(24)的左边减除。这在沟渠子程序包中体现为，对每个受沟渠影响的计算单元，MODFLOW检验其水头大小，以判断是否超过沟渠内水位高程，若属实，则在联立求解矩阵方程时，把 $-CD_{i,j,k}$ 项加到 $HCOF_{i,j,k}$  (公式(26))而把 $-CD_{i,j,k}d_{i,j,k}$ 项加到 $RHS_{i,j,k}$ 中去。

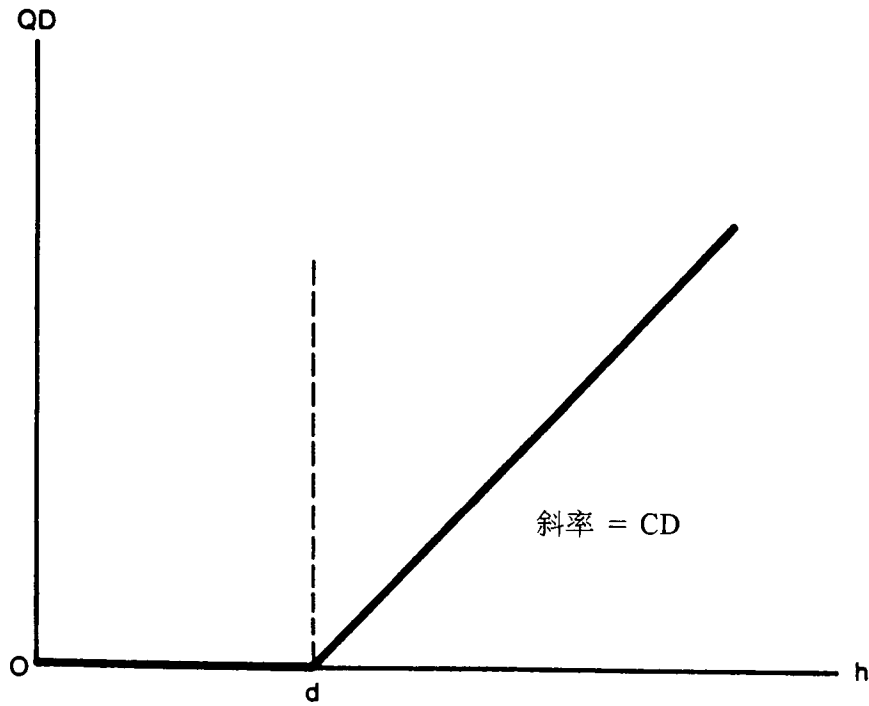


图41. 流入沟渠的流量 $QD$ 对计算单元之水头 $h$ 的图解。  
其中， $d$ 为沟渠之设定水位， $CD$ 为水力传导系数。



### 沟渠子程序包输入数据及格式

沟渠(DRN)子程序包的输入从IUNIT(3)所指定的文件设备号读取。

对每次模拟:

由子程序DRN1AL读取的数据包括

- |          |       |        |
|----------|-------|--------|
| 1. 数据名称: | MXDRN | IDRNCB |
| 输入格式:    | I10   | I10    |

对每个应力期:

由子程序DRN1RP读取的数据包括

- |          |       |     |     |           |       |
|----------|-------|-----|-----|-----------|-------|
| 2. 数据名称: | ITMP  |     |     |           |       |
| 输入格式:    | I10   |     |     |           |       |
| 3. 数据名称: | Layer | Row | Col | Elevation | Cond  |
| 输入格式:    | I10   | I10 | I10 | F10.0     | F10.0 |

(该项输入为每段沟渠一项由记录组成, 若ITMP为负或零, 则不输入本项。)

### 输入数据说明

**MXDRN:** 所有应力期中使用的沟渠段的最大数目。

**IDRNCB:** 标示符和文件设备号。

若IDRNCB>0, 是文件设备号。当ICBCFL (见输出控制) 被设定时, 将在各沟渠段的流量从该设备号输出。

若IDRNCB=0, 将不打印或输出沟渠的流量。

若IDRNCB<0, 当ICBCFL被设定时, 在标准输出文件中打印各沟渠段的流量。

**ITMP:** 标示符及计数器。

若ITMP<0, 重复使用前一应力期的沟渠数据。

若ITMP≥0, ITMP为当前应力期中有效沟渠的数量。

- Layer: 该沟渠段所在计算单元的层号。
- Row: 该沟渠段所在计算单元的行号。
- Col: 该沟渠段所在计算单元的列号。
- Elevation: 沟渠中水位高程。
- Cond: 是含水层与沟渠接触界面的水力传导系数。

## 沟渠子程序包输入样单

数据	说明	输入记录
1	{MXDRN, IDRNCB}	3 55
2	{ITMP} 第一应力期	3
3	{层号,行号,列号,高程,水力传导系数}第一段沟渠	2 6 4 220.0 0.7
3	{层号,行号,列号,高程,水力传导系数}第二段沟渠	2 7 4 225.0 0.9
3	{层号,行号,列号,高程,水力传导系数}第三段沟渠	2 5 4 210.0 0.7
2	{ITMP} 第二应力期	-1
2	{ITMP} 第三应力期	-1
2	{ITMP} 第四应力期	2
3	{层号,行号,列号,高程,水力传导系数}第一段沟渠	2 5 4 210.0 0.8
3	{层号,行号,列号,高程,水力传导系数}第二段沟渠	2 6 4 220.0 0.7
2	{ITMP} 第五应力期	0
2	{ITMP} 第六应力期	-1

[译注：样单中的数据可能不符合格式要求，仅供参考。]

## 第十章 蒸发蒸腾子程序包

## 概念及程序化

蒸发蒸腾子程序包 (EVT) 用于模拟由于植物蒸腾作用以及地下水饱和带直接蒸发的水量。模拟方法假定: (1)当地下水面处于或高于某指定“ET界面”的高程时, 蒸发蒸腾损失在该地下水面位置达到最大值, 而该速率大小则由用户指定; (2)地下水面在ET界面之下的埋深达到某指定的间距, 即本文所指的终止深度或截止深度时, 即停止蒸发蒸腾作用; (3)地下水面介于这两个界限之间时, 蒸发蒸腾量随地下水面高程呈线性变化。

这可用公式表达为:

$$R_{ETi,j} = R_{ETM,j} \quad h_{i,j,k} > h_{si,j} \quad (71)$$

$$R_{ETi,j} = 0 \quad h_{i,j,k} < h_{si,j} - d_{i,j} \quad (72)$$

$$R_{ETi,j} = R_{ETM,j} \left\{ \frac{h_{i,j,k} - (h_{si,j} - d_{i,j})}{d_{i,j}} \right\} \quad h_{si,j} - d_{i,j} \leq h_{i,j,k} \leq h_{si,j} \quad (73)$$

其中  $R_{ETi,j}$  是计算单元面积  $DEL R_j * DEL C_i$  内每单位面积地下水面蒸发蒸腾损失率, 用单位面积单位时间内水量体积表示;  $h_{i,j,k}$  是出现蒸发蒸腾的计算单元的水头, 或地下水面高程;  $R_{ETM,j}$  是  $R_{ETi,j}$  的最大可能值;  $h_{si,j}$  是ET界面高程, 或蒸发蒸腾达到最大值的地下水面标高;  $d_{i,j}$  是截止或终止深度, 当  $h_{si,j}$  至  $h_{i,j,k}$  的距离大于  $d_{i,j}$  时, 蒸发蒸腾作用即告结束。

在进行有限差分计算时, 需要提供给定计算单元的蒸发蒸腾损失的水量。对于有蒸发蒸腾损耗的计算单元, 这可以表达为单位面积的损失率与水平面上计算单元面积,  $DEL R_j * DEL C_i$  的乘积, 即

$$Q_{ETi,j} = R_{ETi,j} * DEL R_j * DEL C_i \quad (74)$$

其中

$Q_{ETi,j}$  是蒸发蒸腾量, 单位为单位时间内通过计算单元面积  $DEL R_j * DEL C_i$  的水的体积量。

若 (与  $R_{ETM,j}$  相对应的)  $Q_{ETi,j}$  最大值指定为  $Q_{ETM,j}$ , 公式(71)—(73)可表达为体积排泄量的形式:

$$Q_{ETi,j} = Q_{ETM,j} \quad h_{i,j,k} > h_{si,j} \quad (75)$$

$$Q_{ETi,j} = 0 \quad h_{i,j,k} < h_{si,j} - d_{i,j} \quad (76)$$

$$Q_{ETi,j} = Q_{ETM,i,j} \left\{ \frac{h_{i,j,k} - (h_{si,j} - d_{i,j})}{d_{i,j}} \right\} \quad h_{si,j} - d_{i,j} \leq h_{i,j,k} \leq h_{si,j} \quad (77)$$

图42是公式(75)—(77)所表达蒸发蒸腾损耗量 $Q_{ETi,j}$ 对其所在计算单元的水头变量的函数关系图解。ET函数与河流函数或沟渠函数的数学形式相似。不同的是，ET函数的线性部分的上下边界均为常量，而不是仅仅由下边界所限。

蒸发蒸腾只从垂向柱体中某一计算单元蒸取水。用户可通过两种选择具体指定该计算单元（即第k层）。第一种选择是，蒸发蒸腾总是发生在模型的最上面的一层中；而第二种选择，用户在垂向柱体ij中指定发生蒸发蒸腾的计算单元所在层位。若这两种选择所指定的计算单元是不透水或定水头计算单元，则蒸发蒸腾作用的影响忽略不计；所输入的蒸发蒸腾率值的大小不会改变模拟结果。

对水平面内每个计算单元(i,j)位置，EVT子程序包将每个应力期（除非欲使用前一应力期的值）的 $R_{ETM}$ 值（单位时间单位面积的最大蒸发蒸腾损失率），读入一个名为EVTR的数组内。这些值随即被乘以计算单元面积 $DEL R_j * DEL C_i$ ，以求得每个计算单元的最大蒸发蒸腾水量 $Q_{ETM}$ ；MODFLOW用这些最大蒸散发水量替换EVTR数组中 $R_{ETM,i,j}$ 的值。因此数组EVTR的输入数据是单位面积最大蒸发蒸腾率，取量纲 $LT^{-1}$ 。然而，程序在执行计算时，EVTR数组的数据项则是最大蒸散发水量，其量纲为 $L^3T^{-1}$ 。

ET界面高程的值即 $h_{si,j}$ （或最大蒸发蒸腾面所在的地下水水面高程），由EVT子程序包读入二维数组SURF内；截止或终止深度值则读入二维数组EXDP内。因为公式(75)—(77)中的 $Q_{ETi,j}$ 项已定义为含水层的排水项，所以必须从公式(24)左边减去此项。至于公式(26)中的 $HCOF$ 和 $RHS$ 项，则在EVT子程序包内以下列方式进行计算：

- 1) 若 $h_{i,j,k} < (h_{si,j} - d_{i,j})$ ，计算单元(i,j,k)的 $HCOF$ 或 $RHS$ 不作调整；
- 2) 若 $h_{i,j,k} > h_{si,j}$ ，将 $Q_{ETM,i,j}$ 加到 $RHS_{i,j,k}$ 中；
- 3) 若 $(h_{si,j} - d_{i,j}) \leq h_{i,j,k} \leq h_{si,j}$ ，将 $-Q_{ETM}/d_{i,j}$ 加到 $HCOF_{i,j,k}$ 上，并将 $-Q_{ETi,j} \left\{ \frac{h_{si,j} - d_{i,j}}{d_{i,j}} \right\}$ 加到 $RHS_{i,j,k}$ 上。

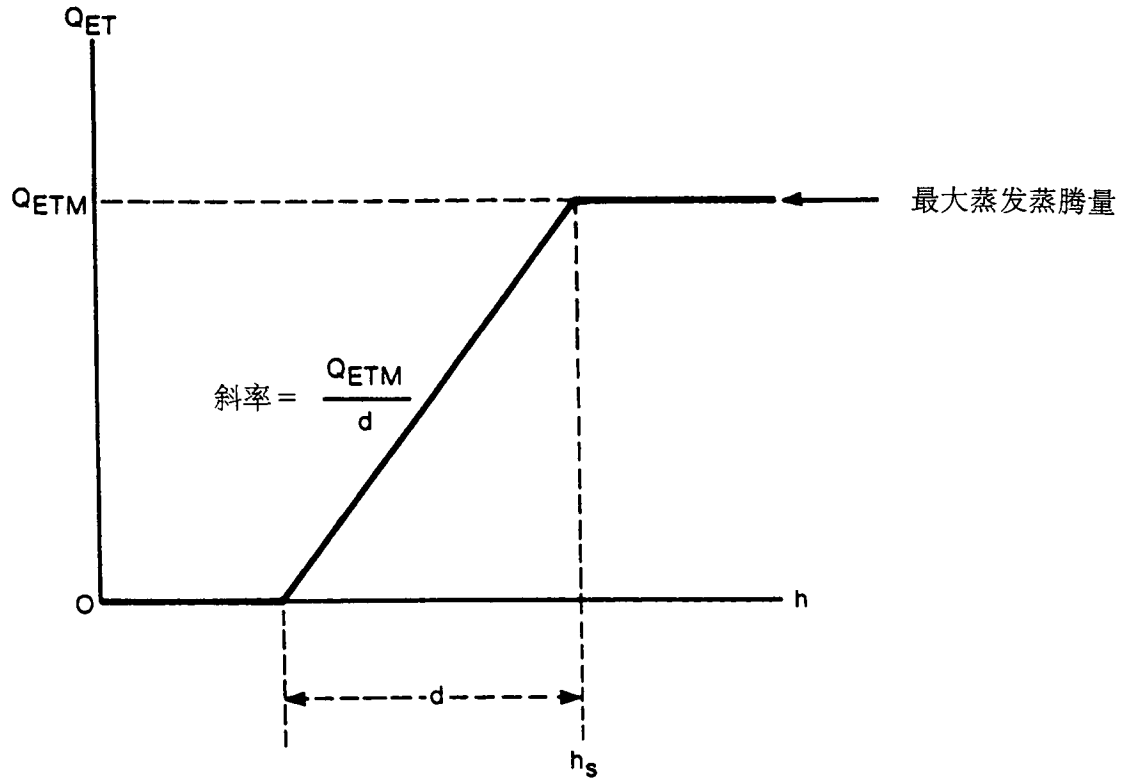


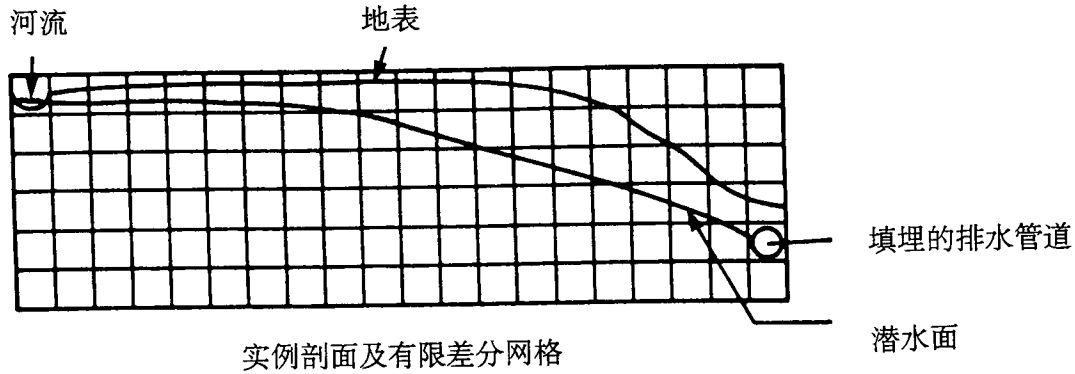
图42. 一个计算单元中体积蒸发蒸腾量,  $Q_{ET}$ , 作为水头,  $h$ , 的函数图解。  
 其中 $d$ 为截止深度,  $h_s$ 为ET界面高程。

蒸发蒸腾作用最大值所对应的地下水面高程 $h_{s,i}$ 值，通常应当作计算单元面积 $DEL R_j^*$   $DEL C_i$ 范围内地表平均高程；截止或终止深度 $d_{i,j}$ 通常假定在地表下6至8英尺的范围内（尽管气候因素，深根湿地植物等等，可能会引起相当可观的变化）。当计算单元面积范围内地表高程到地下水面的深度变化较大时，应当谨慎运用EVT子程序包及选择公式(77)中的各种参数，否则有可能得到错误的结果。

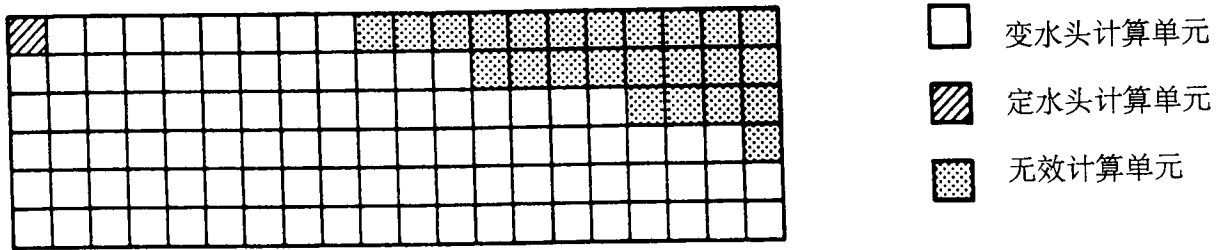
用户可利用该子程序包的设定选择项来指定特殊情况下蒸发蒸腾作用出现的层位。但是运用时也需谨慎。图43所示情形与补给子程序包所讨论的相似，横剖面模型被逐渐调整以近似表达地下水面，其中使用了水平面方向潜水含水层的水力传导系数的求算方法（见第五章）。以图43-a水文剖面图为例，模拟得到的变水头、无效（无渗流）计算单元的最终分布，示于图43-b。

在第一种选择下（图43-c），蒸发蒸腾仅发生于模型的最上面的一层；由于无效单元的出现，模型右半部分的蒸发作用也因此消失。这样一来就无法模拟野外的真实情况。

用第二种选择所得的水文剖面图示于图43-d。这里假定模拟是逐个阶段进行的。根据每一阶段模拟后的潜水面位置，用户应相应调整发生蒸发蒸腾作用的计算单元，以真实地模拟野外的情况。

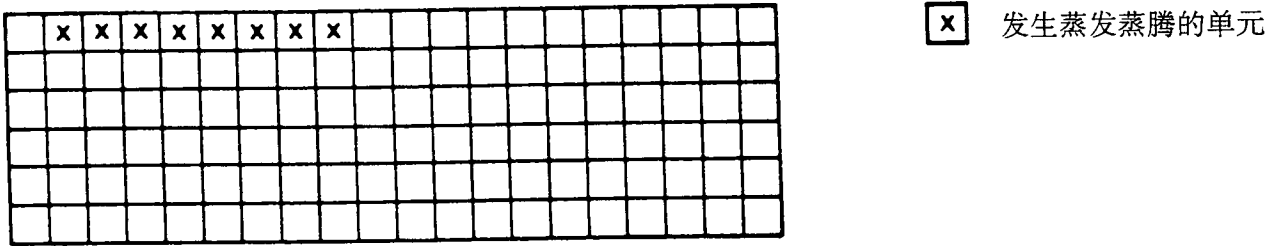


a



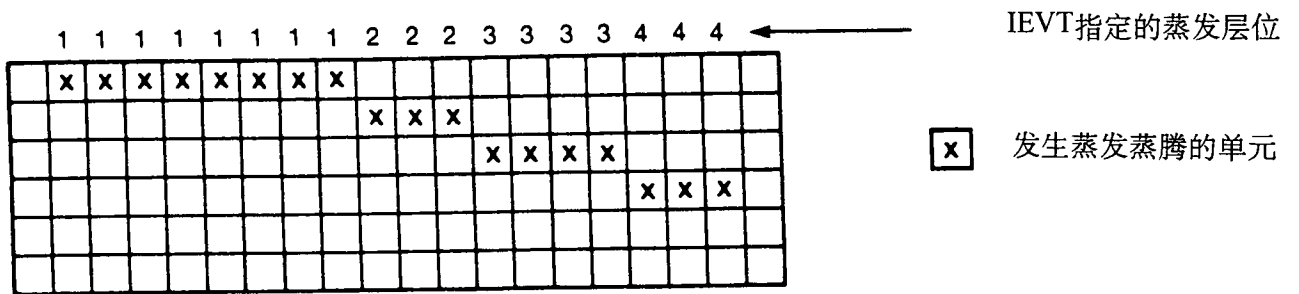
模拟结束时计算单元的状态

b



第一种选择时发生蒸发蒸腾的计算单元

c



第二种选择时发生蒸发蒸腾的计算单元

d

图43. 蒸发蒸腾子程序包中两种选择时发生蒸发蒸腾的计算单元。



### 蒸发蒸腾子程序包输入数据及格式

蒸发蒸腾 (EVT) 子程序包的输入数据从IUNIT(5)所指定的文件设备号读取。

对每次模拟:

由子程序EVT1AL读取的数据包括

- 1. 数据名称: NEVTOP      IEVTCB  
   输入格式:   I10            I10

对每个应力期:

由子程序EVT1RP读取的数据包括

- 2. 数据名称: INSURF    INEVTR      INEXDP      INIEVT  
   输入格式:   I10          I10          I10          I10
- 3. 数据名称: SURF  
   读取模块: U2DREL
- 4. 数据名称: EVTR  
   读取模块: U2DREL
- 5. 数据名称: EXDP  
   读取模块: U2DREL

若ET选择项NEVTOP等于2

- 6. 数据名称:   IEVT  
   读取模块: U2DINT

### 输入数据说明

NEVTOP: 蒸发蒸腾选择指示符。ET参数 (ET界面, 最大ET率, 终止深度) 在二维数组 SURF, EVTR, 以及EXDP中指定。每一垂向柱体对应一个值。于是, 每垂向柱体只对一个计算单元进行计算。选择指示符具体确定将对垂向柱中哪些计算单元进行蒸发计算。

1—仅对模型顶层的计算单元进行蒸发蒸腾计算。

2—用户指定IEVT数组中垂向柱体上发生蒸发蒸腾的计算单元。

IEVTCB: 标示符及文件设备号。

若IEVTCB>0, 表示文件设备号。当ICBCFL (见输出控制) 设定时, 将从该文件设备号输出各计算单元的蒸发蒸腾量。

若IEVTCB≤0, 将不打印或输出蒸发蒸腾量。

INSURF: ET界面 (SURF) 读取标示符。

若INSURF≥0, 将读取ET界面高程的数组。

若INSURF<0, 将重复利用上一应力期的ET界面数据。

INEVTR: 最大蒸发蒸腾率 (EVTR) 读取标示符。

若INEVTR≥0, 将读取最大蒸发蒸腾率数组。

若INEVTR<0, 将重复利用上一应力期的最大蒸发蒸腾率。

INEXDP: 终止深度 (EXDP) 读取标示符。

若INEXDP≥0, 将读取终止深度 (EXDP) 数组。

若INEXDP<0, 将重复利用上一应力期的终止深度。

INIEVT: 蒸发层位数组读取标示符。该数组仅在蒸发蒸腾选择指示符 (NEVTOP) 等于2时使用。

若INIEVT≥0, 将读取蒸发层位数组 (IEVT) 。

若INIEVT<0, 将重复使用上一应力期的蒸发层数组。

SURF: ET界面的高程。

EVTR: 最大蒸发蒸腾率 (单位面积水量(LT<sup>-1</sup>))。

EXDP: ET终止深度。

IEVT: 蒸发层位数组。对每一个水平面位置, 它代表发生蒸发蒸腾的层位。仅当蒸发蒸腾选择指示符等于2时才需使用该数组。

第一种选择下蒸发蒸腾子程序包输入样单

数据项	说明	输入记录
1	{NEVTOP, IEVTCB}	1 0
2	第一应力期{INSURF, INEVTR, INEXDP, INIEVT}	1 1 1
3	ET界面数组控制记录	27 1.(10F5.0) 4
		710 715 720 725 730
		715 720 725 730 735
	ET界面值	720 725 730 735 740
		725 730 735 740 745
		730 735 740 745 750
		735 740 745 750 755
4	最大蒸发蒸腾率数组控制记录	0 9.65E-7
5	终止深度数组控制记录	0 10.0
2	第二应力期{INSURF, INEVTR, INEXDP, INIEVT}	-1 1 -1
4	最大蒸发蒸腾率数组控制记录	0 8.23E-7
2	第三应力期{INSURF, INEVTR, INEXDP, INIEVT}	-1 1 -1
4	最大蒸发蒸腾率数组控制记录	27 9.65E-7 (10F4.0) 4
		1.2 1.2 1.2 1.2 1.3
		1.2 1.2 1.2 1.3 1.4
	蒸发蒸腾率	1.2 1.2 1.3 1.4 1.4
		1.0 1.0 1.0 1.1 1.1
		1.2 1.3 1.3 1.4 1.4
		1.3 1.3 1.4 1.4 1.4

第二种选择下蒸发蒸腾子程序包输入样单

数据项	说明	输入记录
1	{NEVTOP, IEVTCB}	2 45
2	第一应力期 {INSURF, INEVTR, INEXDP, INIEVT}	1 1 1 1
3	ET界面数组控制记录	27 1.(10F5.0) 4
		710 715 720 725 730

		715	720	725	730	735	
	ET界面值	720	725	730	735	740	
		725	730	735	740	745	
		730	735	740	745	750	
		735	740	745	750	755	
4	最大蒸发蒸腾率数组控制记录	0 9.65E-7					
5	消失深度数组控制记录	0	10.0				
6	蒸发层位数组控制记录	12	1 (2012)				3
		1	2	2	2	3	
		1	2	2	2	2	
	蒸发层位数	1	1	2	2	2	
		1	1	1	1	2	
		1	1	1	1	1	
		1	1	1	1	1	
2	第二应力期{ INSURF, INEVTR, INEXDP, INIEVT}	-1	1	-1	-1	-1	
4	最大蒸发蒸腾率数组控制记录	0 8.23E-7					

[译注: 样单中的数据可能不符合格式要求, 仅供参考。]

## 第十一章 通用水头边界子程序包

### 概念及程序化

通用水头边界 (General-Head Boundary, 或简称为GHB) 子程序包的作用在数学形式上与河流、沟渠及蒸发蒸腾子程序包相似之处在于, 从外部水源进入或流出计算单元(i,j,k)的水流量与该计算单元水头 $h_{i,j,k}$ 和外部水源的水头 $h_{b,i,j,k}$ 之差成正比。据此, 可确立计算单元水流量与计算单元水头间的线性关系, 即

$$Q_{b,i,j,k} = C_{b,i,j,k}(h_{b,i,j,k} - h_{i,j,k}) \quad (78)$$

其中 $Q_{b,i,j,k}$ 是从外部水源进入计算单元(i,j,k)的流量;  $C_{b,i,j,k}$ 是外部水源与计算单元(i,j,k)间的水力传导系数;  $h_{b,i,j,k}$ 是外部水源的水头;  $h_{i,j,k}$ 是计算单元(i,j,k)的水头。计算单元(i,j,k)和外部水源的关系见图44。通用水头的水源由图44右边的具有固定水头的水箱表示, 该水箱中的水源不受其它因素的影响, 将水头维持在 $H_b$ 的位置; 水源与计算单元(i,j,k)的联系用多孔材料块体 $C_{b,i,j,k}$ 表示。注意图44中并没有任何机制限制水流因 $h_{i,j,k}$ 上升或下降而向或左或右的方向流动。当 $h_{i,j,k}$ 大于 $h_{b,i,j,k}$ 时, 地下水由计算单元(i,j,k)向右流向水箱; 而反过来, 地下水向左流动进入计算单元(i,j,k)。

图45是公式(78)给出的 $Q_{b,i,j,k}$ 对 $h_{i,j,k}$ 变量的曲线图解。与河流、沟渠和蒸发蒸腾子程序包相比, 通用水头边界与含水层之间的流量仅与二者之间的水头差成正比, 而不受其它条件限制。当计算单元(i,j,k)和外部水源水头之差增加时, 进入或流出计算单元的水流不受限制地持续增加。因而在使用通用水头边界子程序包时必须注意, 以保证不让进入或流出系统的流量随模拟过程变得不切实际。

因为公式(78)中 $Q_{b,i,j,k}$ 是定义为流进含水层的, 所以必须将其加到公式(24)的左边。至于 $HCOF$ 和 $RHS$ 两项, 则在联立求解矩阵方程时, 从公式(26)的 $HCOF_{i,j,k}$ 中减去 $C_{b,i,j,k}$ , 以及从 $RHS_{i,j,k}$ 项中减去 $C_{b,i,j,k}h_{b,i,j,k}$ 。

[译注: 当水力传导系数很大时, GHB子程序包的作用与定水头边界相同。而且由于沿GHB边界的水位可以随时间变化, 故可以在非稳定流计算中用来模拟地表水体水位随时间的变化。]

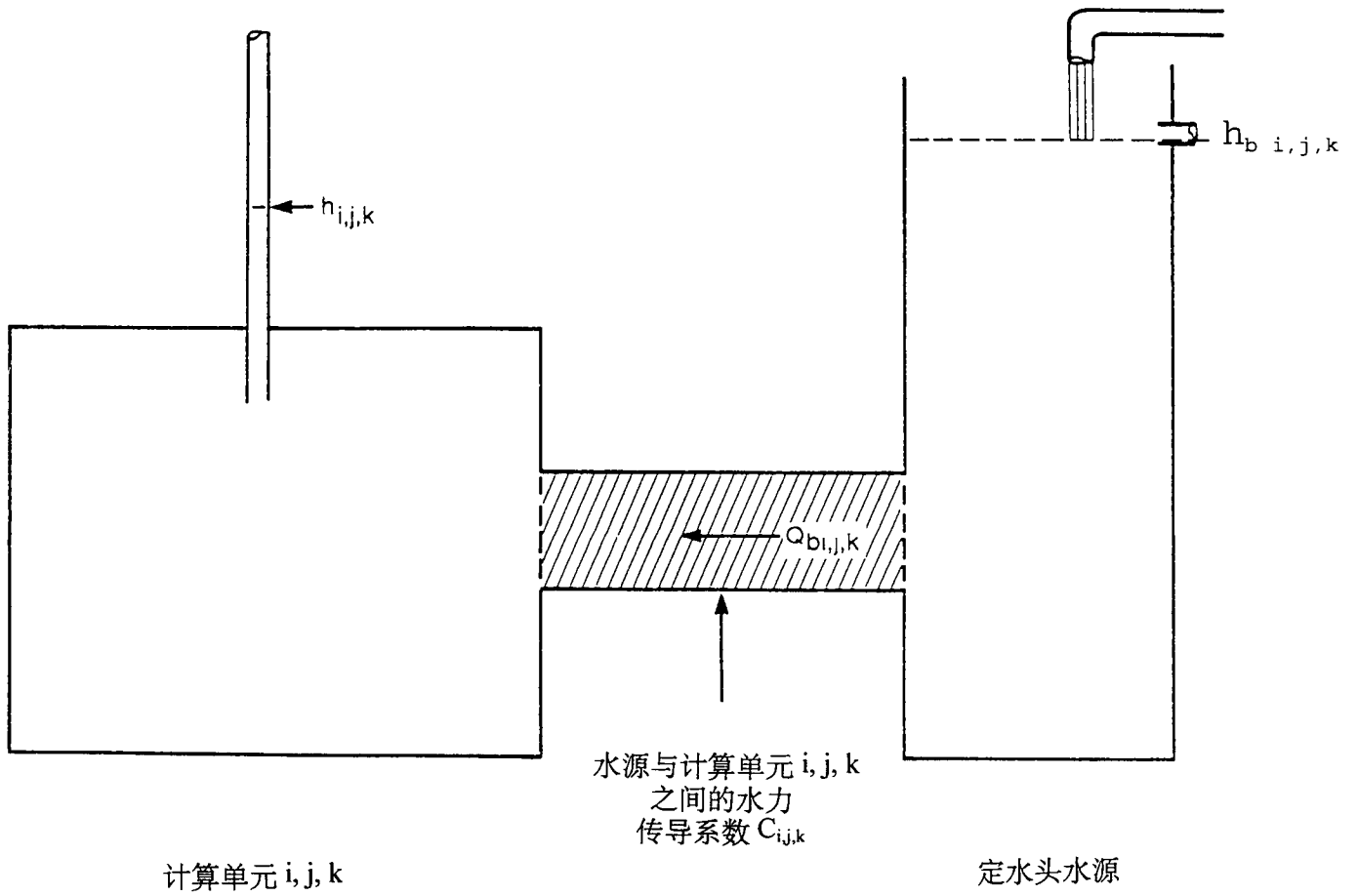


图44. 通用水头边界子程序包原理图。

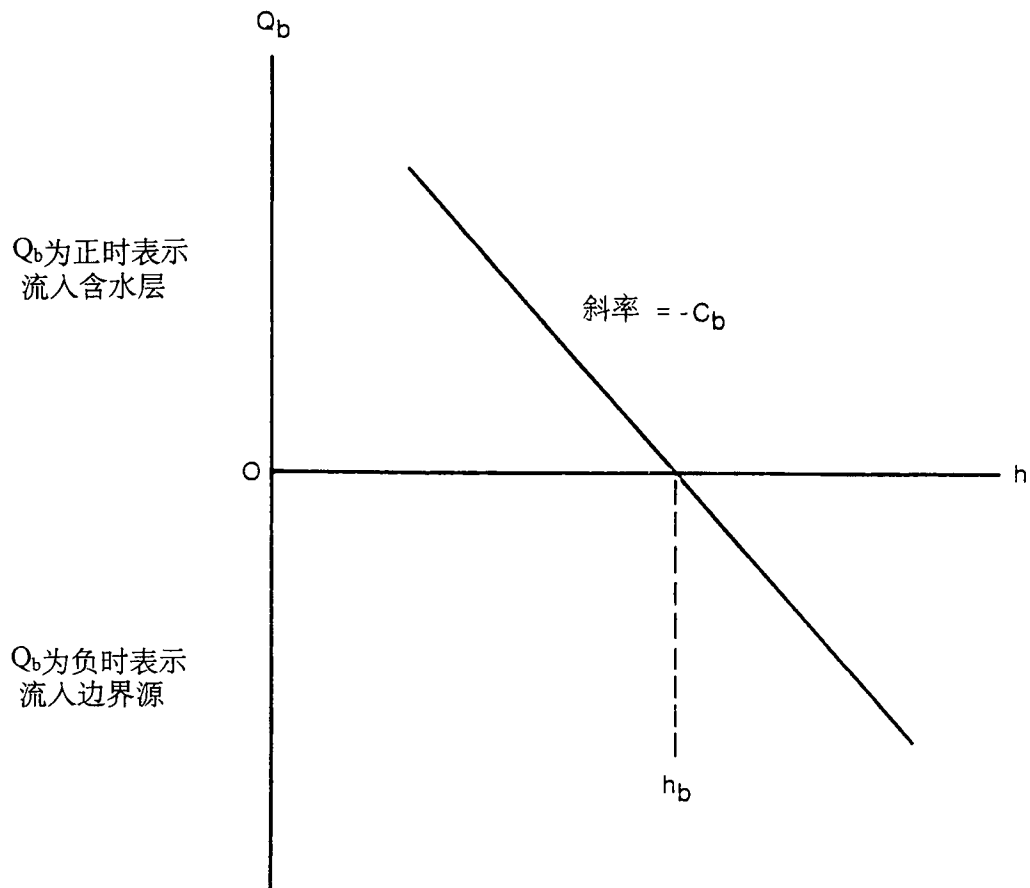


图45. 从通用水头边界水源流入某计算单元的流量 $Q_b$ 与该计算单元水头 $h$ 的函数关系示意图。 $h_b$ 为水源之水头。

### 通用水头边界子程序包输入数据及格式

通用水头边界子程序包的输入从IUNIT(7)所指定的文件设备号读取。

对每次模拟:

由子程序GHB1AL读取的数据包括

1. 数据名称: MXBND IGHBCB  
输入格式: I10 I10

对每个应力期:

由子程序GHB1RP读取的数据包括

2. 数据名称: ITMP  
输入格式: I10
3. 数据名称: Layer Row Col Head Cond  
输入格式: I10 I10 I10 F10.0 F10.0  
(该项输入为每段通用水头边界一行记录。若ITMP是负或零, 则不应输入本项。)

### 输入数据说明

MXBND: 各应力期中通用水头边界所在计算单元的最大数目。

IGHBCB: 标示符和文件设备号。

若IGHBCB>0, 当ICBCFL (见输出控制) 设定时, 是记录计算单元间流量的文件设备号。

若IGHBCB=0, 将不打印或记录计算单元间流量。

若IGHBCB<0, 当ICBCFL设定时, 将打印每个计算单元的边界渗流量。

ITMP: 标示符和计数器。



若 $ITMP < 0$ ，将重复使用上一应力期GHB的数据。

若 $ITMP \geq 0$ ，ITMP是当前应力期通用水头边界单元的总数目。

Layer: 受通用水头边界影响的计算单元所在层号。

Row : 受通用水头边界影响的计算单元所在行号。

Col: 受通用水头边界影响的计算单元所在列号。

Head: 边界上的水头。

Cond: 含水层计算单元与通用水头边界之间界面的水力传导系数。

通用水头边界子程序包输入样单

数据项	说明	输入记录				
1	{MXBND, IGHBCB}	6	24			
2	第一应力期 {ITMP}	4				
3	第一段边界 {层号,行号,列号,水头,水力传导系数}	2	5	6	235.0	.0012
3	第二段边界 {层号,行号,列号,水头,水力传导系数}	2	4	6	230.0	.0012
3	第三段边界 {层号,行号,列号,水头,水力传导系数}	2	5	8	250.0	.0018
3	第四段边界 {层号,行号,列号,水头,水力传导系数}	2	7	6	235.0	.0012
2	第二应力期 {ITMP}	-1				
2	第三应力期 {ITMP}	-1				
2	第四应力期 {ITMP}	6				
3	第一段边界 {层号,行号,列号,水头,水力传导系数}	2	5	6	235.0	.0012
3	第二段边界 {层号,行号,列号,水头,水力传导系数}	2	4	6	230.0	.0012
3	第三段边界 {层号,行号,列号,水头,水力传导系数}	2	5	8	250.0	.0018
3	第四段边界 {层号,行号,列号,水头,水力传导系数}	2	7	6	235.0	.0018
3	第五段边界 {层号,行号,列号,水头,水力传导系数}	2	9	6	235.0	.0012
3	第六段边界 {层号,行号,列号,水头,水力传导系数}	2	10	6	250.0	.0012

[译注: 样单中的数据可能不符合格式要求, 仅供参考。]

## 第十二章 强隐式法子程序包

### 概念及程序化

在本章, 我们讨论强隐式法 (Strongly Implicit Procedure, 简称为SIP)。对于文中涉及到的一些矩阵代数和数值分析中的基本概念, 读者可以阅读有关参考书, 例如前面提到的Peaceman (1977), Crichlow(1977)或Remson, Hornberger和Molz(1971)。这些参考文献不仅可以帮助熟悉有关基本概念, 它们对强隐式法的介绍还可以作为本书内容的补充。

强隐式法是一种对大型线性方程组联立迭代求解的方法。正如在第二章中讨论过的, 一个计算单元 (i, j, k)的有限差分方程具有下面的形式:

$$\begin{aligned}
 & CV_{i,j,k-\frac{1}{2}} h_{i,j,k-1} + CC_{i-\frac{1}{2},j,k} h_{i-1,j,k} + CR_{ij-\frac{1}{2},k} h_{i,j-1,k} \\
 & + (-CV_{i,j,k-\frac{1}{2}} - CC_{i-\frac{1}{2},j,k} - CR_{i,j-\frac{1}{2},k} - CR_{i,j+\frac{1}{2},k} - CC_{i+\frac{1}{2},j,k} - CV_{i,j,k+\frac{1}{2}} + HCOF_{i,j,k}) h_{i,j,k} \\
 & + CR_{i,j+\frac{1}{2},k} h_{i,j+1,k} + CC_{i+\frac{1}{2},j,k} h_{i+1,j,k} + CV_{i,j,k+\frac{1}{2}} h_{i,j,k+1} = RHS_{i,j,k} \quad (79)
 \end{aligned}$$

在每个时间段开始时,我们可以按上式对网格内每个有效计算单元写出一个这样的差分方程。该方程中包含有计算单元 (i,j,k)本身以及与其相邻之六个计算单元的水头。由于每个差分方程中可能出现七个未知项, 而且它们的值可能不完全相同, 故所有计算单元的未知水头应在同一时间段内联立求解。联立的有限差分方程组的同步解, 即为该时间段完成时, 欲求的各个计算单元的水头预测值。

最早提出SIP方法的是 Weinstein, Stone 和 Kwan(1969)。在这里我们仍采用他们的标记法, 将公式 (79) 改写为:

$$\begin{aligned}
 & Z_{i,j,k} h_{i,j,k-1} + B_{i,j,k} h_{i-1,j,k} + D_{i,j,k} h_{i,j-1,k} + E_{i,j,k} h_{i,j,k} \\
 & + F_{i,j,k} h_{i,j+1,k} + H_{i,j,k} h_{i+1,j,k} + S_{i,j,k} h_{i,j,k+1} = Q_{i,j,k} \quad (80)
 \end{aligned}$$

公式 (80) 中各项系数的下标均记为 (i,j,k)。因此, 该式中的 $Z_{i,j,k}$ 相当于(79)中的 $CV_{i,j,k-1/2}$ ;  $E_{i,j,k}$ 则相当公式 (79) 中的

$$(-CV_{i,j,k-\frac{1}{2}} - CC_{i-\frac{1}{2},j,k} - CR_{i,j-\frac{1}{2},k} - CR_{i,j+\frac{1}{2},k} - CC_{i+\frac{1}{2},j,k} - CV_{i,j,k+\frac{1}{2}} + HCOF_{i,j,k})$$

其它项则可依此类推。

正如在第二章中提到的，我们可以按照公式 (80) 的形式，对每个未知计算单元写出一个差分方程，并将它们记为：

$$[A]\{h\} = \{q\} \quad (81)$$

在这里[A]为水头的系数矩阵；{h}为水头矢量矩阵；{q}为(80)式中的右侧项矢量(已知量)。图46表示一个包含有三行、四列、两层的模型所对应的系数矩阵以及两个矢量矩阵。很显然，系数矩阵为一稀疏矩阵。这就是说，只有很少一部分的元素不为零；而且这些非零元素均位于七条对角线上(参见图47)。

从公式(79)和(80)中可以看出，(79)式中的 $CV_{i,j,k-1/2}$ 项同时出现于(80)式中表示计算单元(i,j,k)的系数Z和表示计算单元(i,j,k-1)的系数S之中。即：

$$Z_{i,j,k} = S_{i,j,k-1} \quad (82)$$

同样，

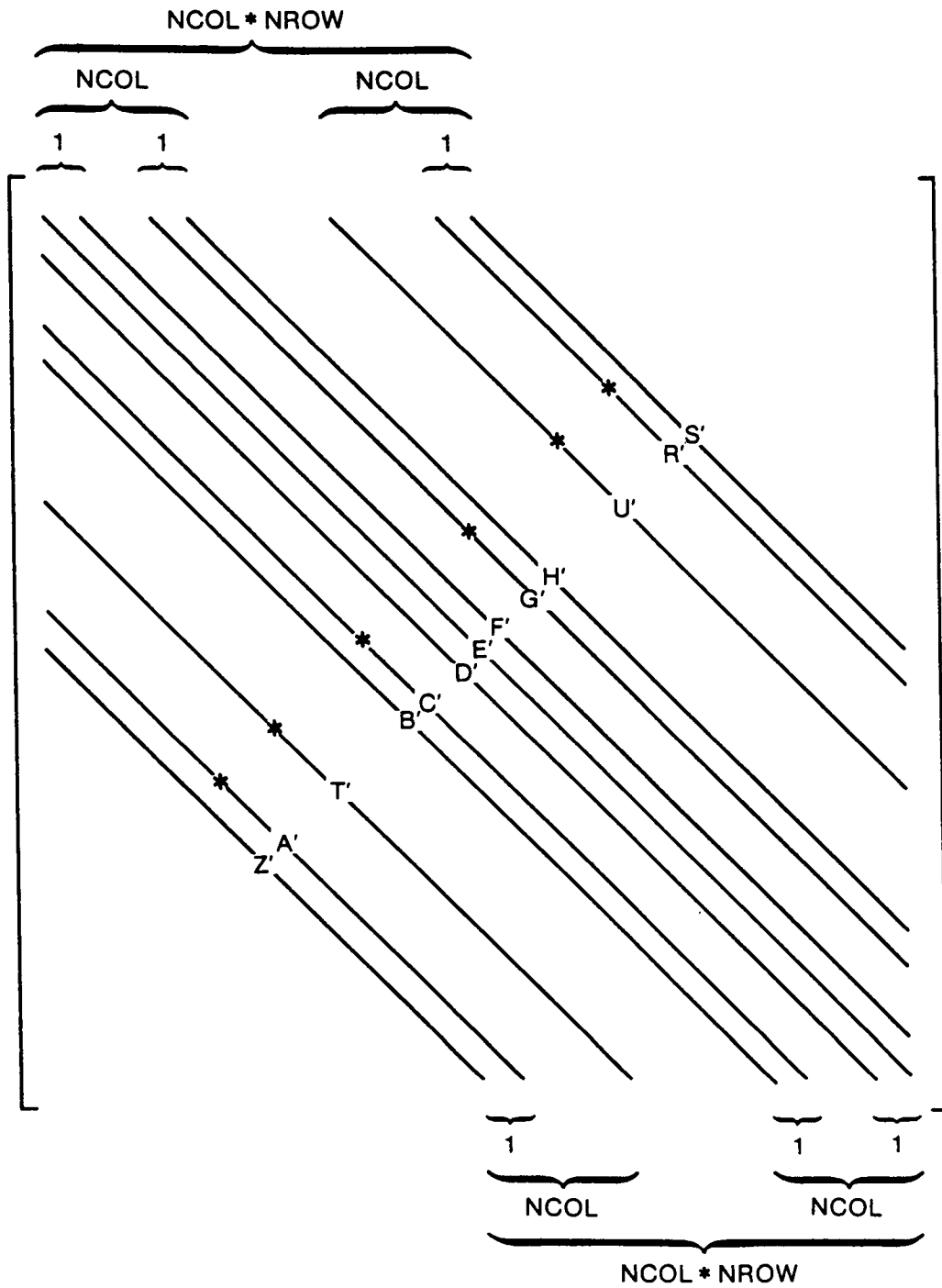
$$B_{i,j,k} = H_{i-1,j,k} \quad (83)$$

$$D_{i,j,k} = F_{i,j-1,k} \quad (84)$$

如果按公式(82)至(84)的定义将图46中所示矩阵中的Z, B, D用与它们相当的S, H和F所替换，就可以得到如图48所示的矩阵形式。其对称性显而易见。所以我们说，公式(81)中系数矩阵为一稀疏对称矩阵。

如果[A]可以分解为两个子矩阵： $[L^*]$ 和 $[U^*]$ ，由(81)所表示的方程组则可用直接的方法求解。这里， $[L^*]$ 为一下三角矩阵(即所有非零元素均出现于主对角线或主对角线以下)； $[U^*]$ 为一上三角矩阵(即所有非零元素均出现于主对角线上或主对角线以上且主对角线上的元素均为1)。图49表示一个3x3的矩阵以及相应的 $[L^*]$ 和 $[U^*]$ 。一旦系数矩阵分解为 $[L^*]$ 和 $[U^*]$ 之后，就可以用一种称为“前后替换”的方法对方程组进行求解。但这样做会有困难：虽然[A]为稀疏矩阵，但一般 $[L^*]$ 和 $[U^*]$ 并非稀疏矩阵。对其中非零元素的计算常需要大量的计算机内存和时间。除此之外，舍入误差可能会很大。





上下两端的花括弧表示非零对角线的水平距离，并以列数来表示（例如，对角线E和F相邻）

图47. 系数矩阵的结构。斜线表示非零对角线。



$$\begin{matrix} & [A] & & \{h\} & = & \{q\} \\ \begin{bmatrix} 1 \\ -1 \\ 3 \end{bmatrix} & \begin{bmatrix} 2 \\ 1 \\ 2 \end{bmatrix} & \begin{bmatrix} 1 \\ 2 \\ -2 \end{bmatrix} & \begin{bmatrix} h_1 \\ h_2 \\ h_3 \end{bmatrix} & & \begin{bmatrix} 1 \\ 2 \\ -3 \end{bmatrix} \end{matrix}$$

$$\begin{matrix} & [L^*] & & [U^*] & & \{h\} & = & \{q\} \\ \begin{bmatrix} 1 \\ -1 \\ 3 \end{bmatrix} & \begin{bmatrix} 0 \\ 3 \\ -4 \end{bmatrix} & \begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix} & \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} & \begin{bmatrix} 2 \\ 1 \\ 0 \end{bmatrix} & \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} & \begin{bmatrix} h_1 \\ h_2 \\ h_3 \end{bmatrix} & & \begin{bmatrix} 1 \\ 2 \\ -3 \end{bmatrix} \end{matrix}$$

图49. 系数矩阵与由此分解出的上, 下三角矩阵。



强隐式法子程序包 (SIP) 的工作原理是寻找一个矩阵[B], 使得[A+B]可以分解为两个子矩阵[L]和[U]。并且[A+B], [L]和[U]满足下列条件:

- (1) [A+B] “接近” 于[A];
- (2) [L]具有下三角矩阵的形式而[U]具有上三角矩阵的形式; [U]之主对角线上的元素均为1;
- (3) [L]和[U]均为稀疏矩阵;
- (4) [L]和[U]均仅含有四条非零对角线。

如果矩阵[B]能满足上述条件, 则将[B]{h}加至公式 (81) 两侧后可得:

$$[A + B]\{h\} = \{q\} + [B]\{h\} \quad (85)$$

如果{h}是公式 (85) 的解, 它也必然是 (81) 的解。但由于{h}出现于 (85) 式的两侧, 使对公式 (85) 的求解并非易事。但如果采用迭代的办法 (参见第二章) 用前一次迭代后获得的值代替 (85) 式右侧的h值, 则可将公式 (85) 写为下面的形式:

$$[A + B]\{h^l\} = \{q\} + [B]\{h^{l-1}\} \quad (86)$$

其中{h<sup>l</sup>}为第l次迭代后的水头值; {h<sup>l-1</sup>}为第l-1次迭代后的水头值。在公式 (86) 中, {h<sup>l-1</sup>}实际上是{h<sup>l</sup>}的一种近似值。如果矩阵[B]已知, 则对公式 (86) 的求解是非常容易的。根据前面讲过的定义, 我们可以将[A+B]分解成为两个稀疏矩阵[L]和[U], 并使用前后替换法得到解。这样一来, 对公式 (86) 的求解问题就转化为如何确定恰当的矩阵[B]的问题。但实际上, 我们可利用矩阵[A]、[A+B]、[L]和[U]进行求解。从公式 (86) 的两侧同时减去[A+B]{h<sup>l-1</sup>}后可得:

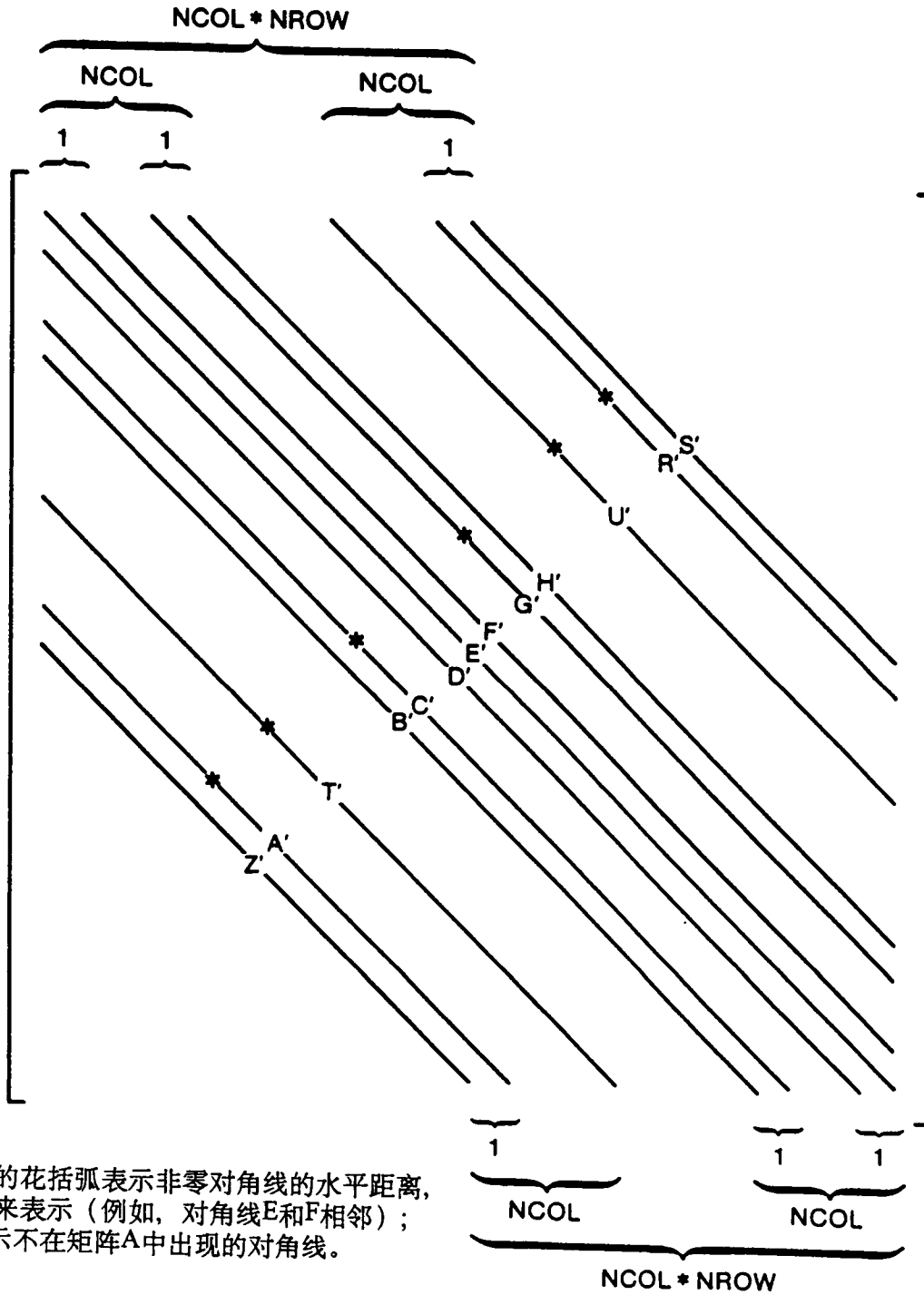
$$[A + B]\{h^l\} - [A + B]\{h^{l-1}\} = \{q\} - [A]\{h^{l-1}\} \quad (87)$$

或

$$[A + B]\{h^l - h^{l-1}\} = \{q\} - [A]\{h^{l-1}\} \quad (88)$$

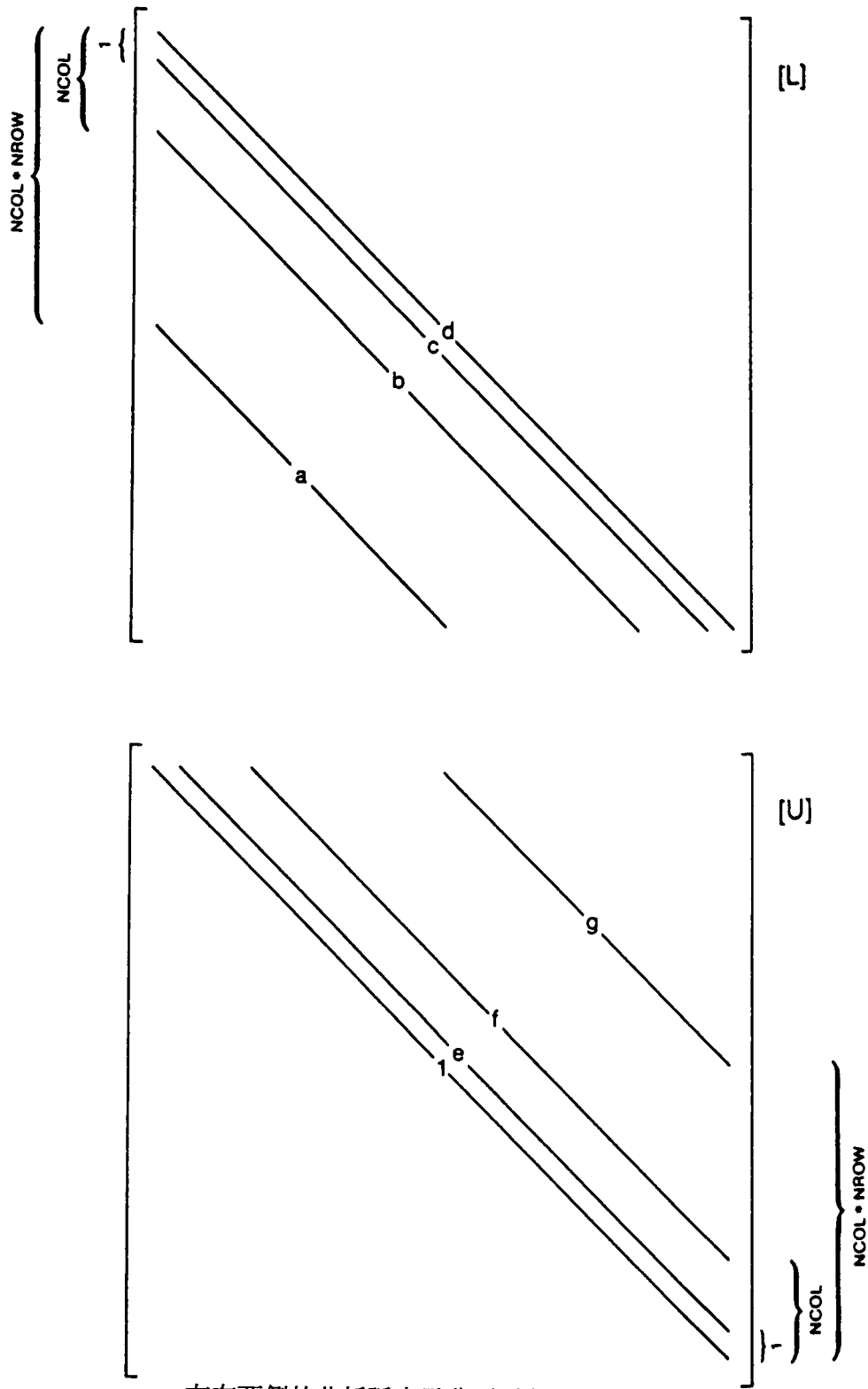
为了[L]和[U]以及[A+B]满足前面提到的条件, [A+B]必须包括矩阵[A]中所没有的六条非零对角线 (见图50)。这些新非零对角线元素的增加, 使得对计算单元 (i,j,k) 的求解公式中出现了与该单元不相邻的计算单元的水头项。矩阵[A+B]的元素与[L]和[U]的元素之间的关系可以由下列公式来定义。在图50和图51中, a、b、c和d为[L]的非零元素, 而e、f和g为[U]中的非零元素。大写字母则表示矩阵[A+B]元素。

$$Z'_{i,j,k} = a_{i,j,k} \quad (89-a)$$



上下两端的花括弧表示非零对角线的水平距离，并以列数来表示（例如，对角线E和F相邻）；星号\*表示不在矩阵A中出现的对角线。

图50. 矩阵 $[A+B]$ 之结构。斜线表示非零对角线。



左右两侧的花括弧表示非零对角线的垂直距离，并以行数来表示（例如，对角线 d 和 e 相邻）；

图51. 由矩阵[A+B]分解出的下三角矩阵[L]和上三角矩阵[U]之结构，斜线表示非零对角线。

$$A'_{i,j,k} = a_{i,j,k} e_{i,j,k-1} \quad (89-b)$$

$$T'_{i,j,k} = a_{i,j,k} f_{i,j,k-1} \quad (89-c)$$

$$B'_{i,j,k} = b_{i,j,k} \quad (89-d)$$

$$C'_{i,j,k} = e_{i-1,j,k} b_{i,j,k} \quad (89-e)$$

$$D'_{i,j,k} = c_{i,j,k} \quad (89-f)$$

$$E'_{i,j,k} = a_{i,j,k} g_{i,j,k-1} + b_{i,j,k} f_{i-1,j,k} + e_{i,j-1,k} c_{i,j,k} + d_{i,j,k} \quad (89-g)$$

$$F'_{i,j,k} = d_{i,j,k} e_{i,j,k} \quad (89-h)$$

$$G'_{i,j,k} = f_{i,j-1,k} c_{i,j,k} \quad (89-i)$$

$$H'_{i,j,k} = f_{i,j,k} d_{i,j,k} \quad (89-j)$$

$$U'_{i,j,k} = b_{i,j,k} g_{i-1,j,k} \quad (89-k)$$

$$R'_{i,j,k} = g_{i,j-1,k} c_{i,j,k} \quad (89-l)$$

$$S'_{i,j,k} = g_{i,j,k} d_{i,j,k} \quad (89-m)$$

如果公式 (89-a...m)中某元素的下标表示该元素位于模型之外, 则其值为零。这十三个方程中共含有二十个未知数, 它们都是矩阵[L], [U]和[A+B]的元素。这表明满足这些条件的矩阵[B]的选择不是唯一的, 很多不同的[B]矩阵与[A]相加后都可以分解为上三角矩阵[U]和下三角矩阵[L]。但[B]的选择还应满足另外一个条件, 那就是[A+B]应当“接近”于[A], 或者说:

$$[A+B]\{h\} \approx [A]\{h\} \quad (90)$$

对于计算单元 (i,j,k)而言, 如果用矩阵[A+B]{h}和[A]{h}的元素来表达的话, (90)式可以看作为:

$$\begin{aligned} & Z'_{i,j,k} h_{i,j,k-1} + A'_{i,j,k} h_{i,j+1,k-1} + T'_{i,j,k} h_{i+1,j,k-1} + B'_{i,j,k} h_{i-1,j,k} + C'_{i,j,k} h_{i-1,j+1,k} \\ & + D'_{i,j,k} h_{i,j-1,k} + E'_{i,j,k} h_{i,j,k} + F'_{i,j,k} h_{i,j+1,k} + G'_{i,j,k} h_{i+1,j-1,k} + H'_{i,j,k} h_{i+1,j,k} \\ & + U'_{i,j,k} h_{i-1,j,k+1} + R'_{i,j,k} h_{i,j-1,k+1} + S'_{i,j,k} h_{i,j,k+1} \quad (91) \\ & \approx Z_{i,j,k} h_{i,j,k-1} + B_{i,j,k} h_{i-1,j,k} + D_{i,j,k} h_{i,j-1,k} + E_{i,j,k} h_{i,j,k} + F_{i,j,k} h_{i,j+1,k} \\ & + H_{i,j,k} h_{i+1,j,k} + S_{i,j,k} h_{i,j,k+1} \end{aligned}$$

我们还可以把 (91) 式的形式改写如下: 将不属于[A]矩阵六条非零对角线的元素移至右侧, 而将矩阵[A]和[A+B]相应元素之差项移至方程的左侧, 则有:

$$\begin{aligned}
 & (Z_{i,j,k} - Z'_{i,j,k})h_{i,j,k-1} + (B_{i,j,k} - B'_{i,j,k})h_{i-1,j,k} \\
 & + (D_{i,j,k} - D'_{i,j,k})h_{i,j-1,k} + (E_{i,j,k} - E'_{i,j,k})h_{i,j,k} \\
 & + (F_{i,j,k} - F'_{i,j,k})h_{i,j+1,k} + (H_{i,j,k} - H'_{i,j,k})h_{i+1,j,k} \\
 & + (S_{i,j,k} - S'_{i,j,k})h_{i,j,k+1} \approx A'_{i,j,k} h_{i,j+1,k-1} \\
 & + T'_{i,j,k} h_{i+1,j,k-1} + C'_{i,j,k} h_{i-1,j+1,k} + G'_{i,j,k} h_{i+1,j-1,k} \\
 & + U'_{i,j,k} h_{i-1,j,k+1} + R'_{i,j,k} h_{i,j-1,k+1}
 \end{aligned} \tag{92}$$

公式 (92) 右侧项为矩阵[A+B]六条非零对角线上的元素, 它们都不在矩阵[A]中出现。这些元素来自于矩阵[B]。与[A]中的非零元素相比较, 这些元素都和与计算单元(i,j,k)直接相连的计算单元的水头有关。公式 (92) 左侧各项则来自于矩阵[A]和[B], 并与计算单元(i,j,k)以及与其直接相连的计算单元的水头有关。

为了减少非相邻计算单元项的影响, 我们引入三个计算参数 (称为 $\alpha$ ,  $\beta$ 和 $\gamma$ )作为公式 (92) 右侧项的乘因子。这些乘因子均取值于0和1之间。在矩阵求解(公式(85)和(86))过程中, 这三个参数作为迭代参数使用。把它们用于 (92) 式后可得:

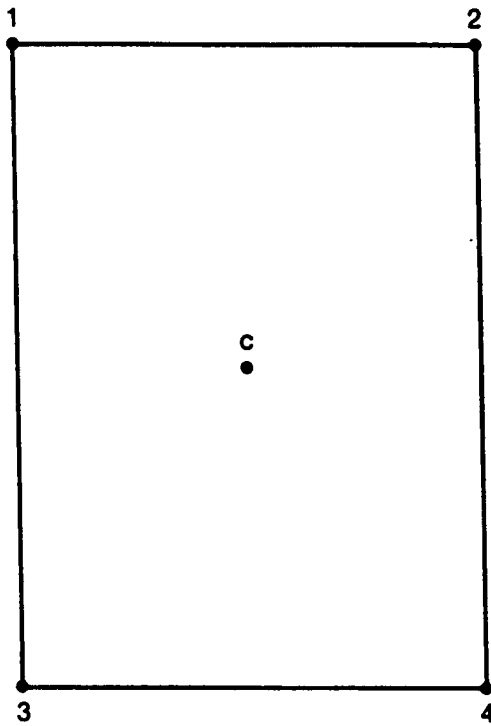
$$\begin{aligned}
 & (Z_{i,j,k} - Z'_{i,j,k})h_{i,j,k-1} + (B_{i,j,k} - B'_{i,j,k})h_{i-1,j,k} \\
 & + (D_{i,j,k} - D'_{i,j,k})h_{i,j-1,k} + (E_{i,j,k} - E'_{i,j,k})h_{i,j,k} \\
 & + (F_{i,j,k} - F'_{i,j,k})h_{i,j+1,k} + (H_{i,j,k} - H'_{i,j,k})h_{i+1,j,k} \\
 & + (S_{i,j,k} - S'_{i,j,k})h_{i,j,k+1} \approx \alpha A'_{i,j,k} h_{i,j+1,k-1} \\
 & + \beta T'_{i,j,k} h_{i+1,j,k-1} + \gamma C'_{i,j,k} h_{i-1,j+1,k} + \gamma G'_{i,j,k} h_{i+1,j-1,k} \\
 & + \beta U'_{i,j,k} h_{i-1,j,k+1} + \alpha R'_{i,j,k} h_{i,j-1,k+1}
 \end{aligned} \tag{93}$$

下一步是将含有非相邻计算单元的水头项 (即公式 (93) 的右侧项) 用与计算单元(i,j,k)相邻的水头来表达。例如, 计算单元 (i,j+1,k-1)位于一个长方形的一个角, 而其它三个角上分别为计算单元(i,j,k-1),(i,j+1,k)和(i,j,k)。用图52所示的插值的办法,  $h_{i,j+1,k-1}$ 可近似表示为:

$$h_{i,j+1,k-1} = h_{i,j+1,k} + h_{i,j,k-1} - h_{i,j,k} \tag{94-a}$$

同理:

假定函数  $f$  在顶角 2, 3, 4 的值已知:



通过插值, 该函数在中心点  $c$  的值可近似表达为:

$$f_1(c) \approx \frac{f(2) + f(3)}{2}$$

或

$$f_2(c) \approx \frac{f(1) + f(4)}{2}$$

假定

$$f_1(c) \approx f_2(c)$$

则

$$\frac{f(2) + f(3)}{2} \approx \frac{f(1) + f(4)}{2}$$

所以

$$f(1) \approx f(2) + f(3) - f(4)$$

图52. 根据某函数在一个矩形三个顶点的已知值, 估计该函数在另一个顶点的值。

$$h_{i+1,j,k-1} = h_{i,j,k-1} + h_{i+1,j,k} - h_{i,j,k} \quad (94-b)$$

$$h_{i-1,j+1,k} = h_{i-1,j,k} + h_{i,j+1,k} - h_{i,j,k} \quad (94-c)$$

$$h_{i+1,j-1,k} = h_{i+1,j,k} + h_{i,j-1,k} - h_{i,j,k} \quad (94-d)$$

$$h_{i-1,j,k+1} = h_{i,j,k+1} + h_{i-1,j,k} - h_{i,j,k} \quad (94-e)$$

$$h_{i,j-1,k+1} = h_{i,j,k+1} + h_{i,j-1,k} - h_{i,j,k} \quad (94-f)$$

将公式 (94 - a … f) 代入 (93) 式并整理后可得:

$$\begin{aligned} & (Z'_{i,j,k} - Z_{i,j,k} + \alpha A'_{i,j,k} + \beta T'_{i,j,k}) h_{i,j,k-1} \\ & + (B'_{i,j,k} - B_{i,j,k} + \gamma C'_{i,j,k} + \beta U'_{i,j,k}) h_{i-1,j,k} \\ & + (D'_{i,j,k} - D_{i,j,k} + \gamma G'_{i,j,k} + \alpha R'_{i,j,k}) h_{i,j-1,k} \\ & + (E'_{i,j,k} - E_{i,j,k} - \alpha A'_{i,j,k} - \beta T'_{i,j,k} - \beta U'_{i,j,k} - \gamma C'_{i,j,k} - \gamma G'_{i,j,k} - \alpha R'_{i,j,k}) h_{i,j,k} \\ & + (F'_{i,j,k} - F_{i,j,k} + \alpha A'_{i,j,k} + \gamma C'_{i,j,k}) h_{i,j+1,k} \\ & + (H'_{i,j,k} - H_{i,j,k} + \beta T'_{i,j,k} + \gamma G'_{i,j,k}) h_{i+1,j,k} \\ & + (S'_{i,j,k} - S_{i,j,k} + \beta U'_{i,j,k} + \alpha R'_{i,j,k}) h_{i,j,k+1} \approx 0 \end{aligned} \quad (95)$$

如果上式中各项系数接近于零, 式 (95) 所表达的关系则可成立。为此, 我们令这些系数等于零:

$$Z'_{i,j,k} - Z_{i,j,k} + \alpha A'_{i,j,k} + \beta T'_{i,j,k} = 0 \quad (96-a)$$

$$B'_{i,j,k} - B_{i,j,k} + \gamma C'_{i,j,k} + \beta U'_{i,j,k} = 0 \quad (96-b)$$

$$D'_{i,j,k} - D_{i,j,k} + \gamma G'_{i,j,k} + \alpha R'_{i,j,k} = 0 \quad (96-c)$$

$$E'_{i,j,k} - E_{i,j,k} - \alpha A'_{i,j,k} - \beta T'_{i,j,k} - \beta U'_{i,j,k} - \gamma C'_{i,j,k} - \gamma G'_{i,j,k} - \alpha R'_{i,j,k} = 0 \quad (96-d)$$

$$F'_{i,j,k} - F_{i,j,k} + \alpha A'_{i,j,k} + \gamma C'_{i,j,k} = 0 \quad (96-e)$$

$$H'_{i,j,k} - H_{i,j,k} + \beta T'_{i,j,k} + \gamma G'_{i,j,k} = 0 \quad (96-f)$$

$$S'_{i,j,k} - S_{i,j,k} + \beta U'_{i,j,k} + \alpha R'_{i,j,k} = 0 \quad (96-g)$$

公式 (96 - a … g) 和 (89 - a … m) 构成一个由20个方程组成的方程组, 其中包含有20个未知数。它们的解就是矩阵[A+B], [L]和[U]的各项元素, 并满足[A+B]“接近”[A], 并可以很容易地分解为[L]和[U]的条件。[L]和[U]均为稀疏矩阵, 并具有所要求的上下三角矩阵的形式。例如将公式 (89 - a, - b, - c) 代入 (96 - a) 后可得:

$$a_{i,j,k} = Z_{i,j,k} / (1 + \alpha e_{i,j,k-1} + \beta f_{i,j,k-1}) \quad (97-a)$$

同理:

$$b_{i,j,k} = B_{i,j,k} / (1 + \gamma e_{i-1,j,k} + \beta g_{i-1,j,k}) \quad (97-b)$$

$$c_{i,j,k} = D_{i,j,k} / (1 + \gamma f_{i,j-1,k} + \alpha g_{i,j-1,k}) \quad (97-c)$$

$$A'_{i,j,k} = a_{i,j,k} e_{i,j,k-1} \quad (97-d)$$

$$C'_{i,j,k} = e_{i-1,j,k} b_{i,j,k} \quad (97-e)$$

$$G'_{i,j,k} = f_{i,j-1,k} c_{i,j,k} \quad (97-f)$$

$$R'_{i,j,k} = g_{i,j-1,k} c_{i,j,k} \quad (97-g)$$

$$T'_{i,j,k} = a_{i,j,k} f_{i,j,k-1} \quad (97-h)$$

$$U'_{i,j,k} = b_{i,j,k} g_{i-1,j,k} \quad (97-i)$$

$$d_{i,j,k} = E_{i,j,k} + \alpha A'_{i,j,k} + \beta T'_{i,j,k} + \gamma C'_{i,j,k} + \gamma G'_{i,j,k} + \beta U'_{i,j,k} \\ + \alpha R'_{i,j,k} - a_{i,j,k} g_{i,j,k-1} - b_{i,j,k} f_{i-1,j,k} - e_{i,j-1,k} c_{i,j,k} \quad (97-j)$$

$$e_{i,j,k} = (F_{i,j,k} - \alpha A'_{i,j,k} - \gamma C'_{i,j,k}) / d_{i,j,k} \quad (97-k)$$

$$f_{i,j,k} = (H_{i,j,k} - \beta T'_{i,j,k} - \gamma G'_{i,j,k}) / d_{i,j,k} \quad (97-l)$$

$$g_{i,j,k} = (S_{i,j,k} - \alpha R'_{i,j,k} - \beta U'_{i,j,k}) / d_{i,j,k} \quad (97-m)$$

当[L]和[U]的元素确定之后, 公式(8)中的[A+B]就可以由[L][U]来代替:

$$[L][U]\{h^l - h^{l-1}\} = \{q\} - [A]\{h^{l-1}\} \quad (98)$$

在这里,  $l$  表示当前的迭代水平;  $l-1$  表示前一次迭代水平。下面我们定义矢量  $\{RES^l\}$ :

$$\{RES^l\} = \{q\} - [A]\{h^{l-1}\} \quad (99)$$

并就此将式(98)重写为:

$$[L][U]\{h^l - h^{l-1}\} = \{RES^l\} \quad (100)$$

到这里, 公式(100)就可以用前后替换法求解了。第一步是利用向前替换法对矢量  $\{v\}$  求解:

$$[L]\{v\} = \{RES^l\} \quad (101)$$

其中  $\{v\} = [U]\{h^l - h^{l-1}\}$ 。  $\{v\}$  确定之后, 再利用向后替换法求出下式中的  $\{h^l - h^{l-1}\}$ :



$$[U]\{h' - h^{l-1}\} = \{v\} \quad (102)$$

在前面的讨论中，我们采用计算单元的下标来表示矩阵的元素或公式中的系数（参见图53 - a）。为了更清楚地演示向前替换的过程，在下面的讨论过程中，我们将采用单一下标来表示矩阵元素（图53 - b）。因为[L]中主对角线以上的元素均为零，所以由公式（101）表示的第一个线性方程为：

$$d_1 v_1 = RES_1' \quad (103)$$

在公式（103）中， $d_1$ 项已由公式（97 - j）确定，并且 $RES_1'$ 已由公式（99）作为矢量（ $RES'$ ）中的一个元素求出。因此， $v_1$ 的值可从公式（103）中获得。由公式（101）表示的第二个方程为：

$$c_2 v_1 + d_2 v_2 = RES_2' \quad (104)$$

同样， $c_2$ 和 $d_2$ 已在公式（97）中解出，而 $RES_2'$ 则可从（99）式中得到。利用从（103）式得到的 $v_1$ 的值，则可从公式（104）中得到 $v_2$ 。

{v}中元素的一般表达式为：

$$v_n = (RES_n' - a_n v_{n-NRC} - b_n v_{n-NCOL} - c_n v_{n-1}) / d_n \quad (105)$$

其中NRC为当前层中计算单元的数目；NCOL为模型的列数，系数 $a_n$ 、 $b_n$ 、 $c_n$ 和 $d_n$ 均可由（97）式求出，而 $RES_n'$ 则通过（99）式得到。在第一和第二个公式（式（103）和（104））中，系数 $a_n$ 和 $b_n$ 等于零。在计算过程中，所需{v}的元素则在前一步运算中求得。这个过程叫做向前替换。在向前替换过程中{v}的元素可依次求得，因为矩阵[L]是一个下三角矩阵，其右上侧的元素均为零。

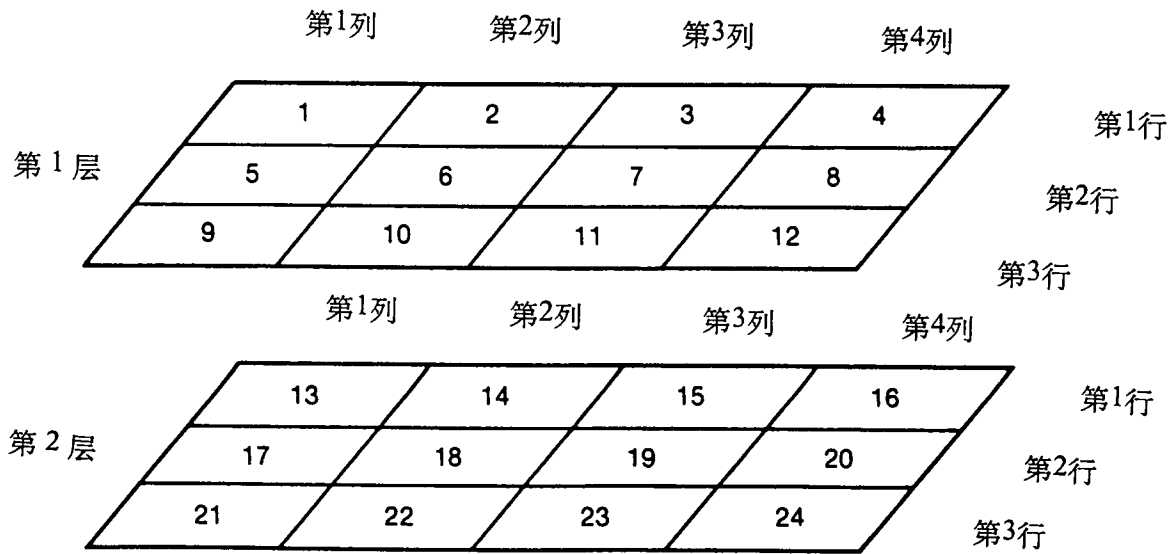
下一步是采用向后替换法利用已知{v}的元素，计算矢量 $\{h' - h^{l-1}\}$ ，也就是对公式（102）求解。向后替换过程与向前替换相似。但在向后替换过程中，由于矩阵[U]为上三角矩阵，故计算顺序与向前替换相反。求出矢量 $\{h' - h^{l-1}\}$ 后，再与矢量 $\{h^{l-1}\}$ 相加，就可以得到 $\{h'\}$ ，即本次迭代欲求的水头矢量。

总之，对下面方程组求解的问题：

$$[A]\{h\} = \{q\} \quad (106)$$



(a) 三元下标法



(b) 单一下标法

图53. 使用三元下标法和单一下标法分别表示计算单元在网格中的位置。

可以转化为下面的迭代过程: (1) 由公式 (97) 确定矩阵[L]和[U]; (2) 利用矢量{q}, 系数矩阵[A]以及前一次迭代得到的水头值, 计算出系统残差{RES<sup>i</sup>}; (3) 对公式 (100) 进行向前向后替换求出水头残差{h<sup>i</sup>-h<sup>i-1</sup>}; (4) 将水头残差{h<sup>i</sup>-h<sup>i-1</sup>}与{h<sup>i-1</sup>}相加从而得到水头矢量{h<sup>i</sup>}。但是, 这些仅仅是SIP方法的一些主要步骤, 还有其它一些问题需要进一步讨论。

### 数组转换

前面谈到, 系数矩阵[A]为一稀疏矩阵, 其中只有七条对角线上的元素不为零。在计算过程中, 我们只需要知道这七条对角线上的元素就行了, 而不必将整个矩阵传递给SIP子程序包。又由于该矩阵的对称性, 只要知道主对角线以及主对角线以下三条非零对角线上的元素就足够了。这三条非零对角线所相对应的水力传导系数数组分别为: CC, CR和CV。主对角线上的元素由这三个数组以及HCOF数组(参见第二章)构成。线性方程组的右侧项, 矢量{q}, 则对应于数组RHS (参见第二章)。前一次迭代后所得到的水头分布数组{h<sup>i-1</sup>}则对应于数组HNEW。每次迭代完成之后, HNEW的内容也更新一次。根据这些, SIP子程序包所要求的数据包括: CC, CR, CV, RHS, HCOF和HNEW, 而该子程序包的输出则包括新计算的结果HNEW。如在第三章中谈到的, 迭代循环中包括“建立方程”的步骤, 故这些输入数组的内容可能在每次迭代后发生变化。

### 计算顺序

实际经验表明, 如果对有限差分方程组的迭代求解过程沿两种不同方向交替进行, 便可以减少迭代次数。在这里, 第一个方向的迭代从第一列, 第一行, 第一层开始, 并沿列, 行和层数增加的方向进行。另一个迭代方向可以选择从最底层的最后一行和第一列开始, 并沿列增加而层数和行数减少的方向进行。按照图51所定义的对角线的名称, 我们可以写出一组与公式 (97 - a ... m) 类似的方程:

$$a_{i,j,k} = Z_{i,j,k} / (1 + \alpha e_{i,j,k+1} + \beta f_{i,j,k+1}) \quad (107-a)$$

$$b_{i,j,k} = B_{i,j,k} / (1 + \gamma e_{i+1,j,k} + \beta g_{i+1,j,k}) \quad (107-b)$$

$$c_{i,j,k} = D_{i,j,k} / (1 + \gamma f_{i,j-1,k} + \alpha g_{i,j-1,k}) \quad (107-c)$$

$$A'_{i,j,k} = a_{i,j,k} e_{i,j,k+1} \quad (107-d)$$

$$C'_{i,j,k} = e_{i+1,j,k} b_{i,j,k} \quad (107-e)$$

$$G'_{i,j,k} = f_{i,j-1,k} c_{i,j,k} \quad (107-f)$$

$$R'_{i,j,k} = g_{i,j-1,k} c_{i,j,k} \quad (107-g)$$

$$T'_{i,j,k} = a_{i,j,k} f_{i,j,k+1} \quad (107-h)$$

$$U'_{i,j,k} = b_{i,j,k} g_{i+1,j,k} \quad (107-i)$$

$$d_{i,j,k} = E_{i,j,k} + \alpha A'_{i,j,k} + \beta T'_{i,j,k} + \gamma C'_{i,j,k} + \gamma G'_{i,j,k} + \beta U'_{i,j,k} \\ + \alpha R'_{i,j,k} - a_{i,j,k} g_{i,j,k+1} - b_{i,j,k} f_{i+1,j,k} - e_{i,j-1,k} c_{i,j,k} \quad (107-j)$$

$$e_{i,j,k} = (F_{i,j,k} - \alpha A'_{i,j,k} - \gamma C'_{i,j,k}) / d_{i,j,k} \quad (107-k)$$

$$f_{i,j,k} = (H_{i,j,k} - \beta T'_{i,j,k} - \gamma G'_{i,j,k}) / d_{i,j,k} \quad (107-l)$$

$$g_{i,j,k} = (S_{i,j,k} - \alpha R'_{i,j,k} - \beta U'_{i,j,k}) / d_{i,j,k} \quad (107-m)$$

在MODFLOW中, 公式(107-a...m)和公式(97-a...m)的计算依次交替进行。事实上, MODFLOW中仅使用一组通用公式进行计算。这些公式中的变量名由单一下标表示。通过将值按顺序赋给该下标所代表的元素, 公式(97)和(107)的求解顺序也就可以确定了。在这组通用公式中, 下标 $n\ l\ l$ 表示对应于刚计算过的层中与第 $n$ 个元素具有相同行号和列号的元素; 下标 $n\ r\ l$ 和 $n\ c\ l$ 也按这种方式定义。另外, 在这些公式中, 迭代参数 $\alpha$ ,  $\beta$ 和 $\gamma$ 均用 $\omega$ 表示。下面我们还将进一步讨论参数 $\omega$ 。注意下列的公式中有一个新的方程用来计算 $v_n$  (矢量 $\{v\}$ 中对应于计算单元 $n$ 的元素)。一旦矩阵 $[L]$ 和 $[U]$ 的第 $n$ 行计算完成之后,  $v_n$ 就可以求得了。这些公式为:

$$a_n = Z_n / (1 + \omega(e_{nll} + f_{nll})) \quad (108-a)$$

$$b_n = B_n / (1 + \omega(e_{nrl} + g_{nrl})) \quad (108-b)$$

$$c_n = D_n / (1 + \omega(f_{ncl} + g_{ncl})) \quad (108-c)$$

$$A'_n = a_n e_{nll} \quad (108-d)$$

$$C'_n = b_n e_{nrl} \quad (108-e)$$

$$G'_n = c_n f_{ncl} \quad (108-f)$$

$$R'_n = c_n g_{ncl} \quad (108-g)$$

$$T'_n = a_n f_{nll} \quad (108-h)$$

$$U'_n = b_n g_{nrl} \quad (108-i)$$

$$d_n = E_n + \omega(A'_n + T'_n + C'_n + G'_n + U'_n + R'_n) - a_n g_{nll} - b_n f_{nrl} - c_n e_{ncl} \quad (108-j)$$

$$e_n = (F_n - \omega(A'_n + C'_n)) / d_n \quad (108-k)$$

$$f_n = (H_n - \omega(T'_n + G'_n)) / d_n \quad (108-l)$$

$$g_n = (S_n - \omega(R'_n + U'_n)) / d_n \quad (108-m)$$

$$v_n = (RES_n - a_n v_{nll} - b_n v_{nrl} - c_n v_{ncl}) / d_n \quad (108-n)$$

由于向后替换时要用到  $e_n$ ,  $f_n$ ,  $g_n$  和  $v_n$  的值, SIP子程序包内专门设置了四个数组来存放它们的值。这四个数组的大小等于模型内所有计算单元的数目。

### 迭代参数

虽然 Weinstein, Stone 和 Kwan (1969) 在他们的文章中定义了三个迭代参数, 但实际上他们仅使用了一个迭代参数。因此, 公式 (93) 中的  $\alpha$ ,  $\beta$ , 和  $\gamma$  可以用一个参数  $\omega$  来代表。这个参数用来与公式 (93) 的右侧各项相乘。但为了达到一定的收敛速度,  $\omega$  应取几个不同的值, 并在迭代过程中循环使用这些值。在 MODFLOW 中,  $\omega$  的值由下式确定:

$$\omega(\lambda) = 1 - (WSEED)^{(\lambda-1)/(NPARM-1)} \quad \lambda=1,2,\dots,NPARM \quad (109)$$

其中 NPARM 为欲使用的  $\omega$  值的数目;  $\lambda$  为指标数, 取值从 1 到 NPARM;  $\omega(\lambda)$  为与  $\lambda$  相对应的迭代参数; WSEED 称为迭代参数之种子 (SEED), 它是确定迭代参数  $\omega$  的基础。其取值按下面的规定进行。

下面我们讨论 WSEED 的确定。首先根据各个计算单元与其相邻单元之间水力传导系数求出三个参数:  $\rho_1$ ,  $\rho_2$ , 和  $\rho_3$ :

$$\rho_1 = \frac{CC_{\max} + CV_{\max}}{CR_{\min}} \quad (110)$$

$$\rho_2 = \frac{CR_{\max} + CV_{\max}}{CC_{\min}} \quad (111)$$

$$\rho_3 = \frac{CR_{\max} + CC_{\max}}{CV_{\min}} \quad (112)$$

其中：相对于计算单元 (i, j, k),  $CC_{\max}$  为沿列方向的水力传导系数  $CC_{i-1/2,j,k}$  和  $CC_{i+1/2,j,k}$  中间较大者,  $CC_{\min}$  两者中间较小者; 同样,  $CR_{\max}$  为沿行方向的水力传导系数  $CR_{i,j-1/2,k}$  和  $CR_{i,j+1/2,k}$  中间较大者,  $CR_{\min}$  为二者中间较小者;  $CR_{\max}$  为  $CV_{i,j,k-1/2}$  和  $CV_{i,j,k+1/2}$  中间较大者,  $CV_{\min}$  为二者中间较小者。利用这些值, 我们可以对每个计算单元计算出下列值:

$$\frac{\pi^2}{2(NCOL)^2(1+\rho_1)}, \quad \frac{\pi^2}{2(NROW)^2(1+\rho_2)}, \quad \frac{\pi^2}{2(NLAY)^2(1+\rho_3)}$$

其中  $NCOL$  为模型的列数,  $NROW$  为行数,  $NLAY$  为层数。这三个数中最小者就是该单元之“种子”, 而  $WSEED$  为所有单元之种子的平均值。对于不同的  $\lambda$ , 可得到不同的  $\omega$  值。迭代参数  $\omega(\lambda)$  依次用于迭代计算, 并循环使用 (每  $NPARM$  次为一周期)。

上述对  $\omega$  值的计算过程与 *Weinstein, Stone* 和 *Kwan* (1969) 所提出的方法略有不同, 但得到的结果类似, 并在许多实际问题中使用效果良好。但是, 在迭代参数的选择过程中, 我们还应当注意以下几点。第一, 上述的计算方法完全是经验性的, 我们还不清楚为什么某些参数的组合会优于另外一些组合。第二, 所选择的参数可能会影响收敛速度, 但并不影响最终计算结果 (如果迭代收敛的话)。第三, 所选参数对收敛速度的影响可能会很大。

*MODFLOW* 中还采用了另外一个迭代参数。为区别于  $\omega$ , 称之为迭代加速因子, 在程序中用  $ACCL$  来表示。它用作  $\{RES^i\}$  的乘积因子。当  $ACCL$  等于 1 时, 它实际不起作用。虽然 *Weinstein, Stone* 和 *Kwan* (1969) 在他们的算法中并没有这样一个参数, 但已有不少人在不同版的程序中引入这样的加速因子 (如 *Peaceman*, 1977, 第 130 页)。迭代加速因子并不象  $\omega$  那样循环使用, 而是由用户定义一个确定的值并预先输入给程序。一般来说,  $ACCL$  的取值在开始时为 1, 并通过调整种子的值来改善

收敛速度。下面我们还将进一步讨论这个问题。如果无法克服不收敛的问题，再通过调整ACCL的值进行试验。

经验表明，加速因子ACCL取1并使用程序计算的种子时，不一定能得到最好的收敛速度（即达到收敛的迭代次数为最少）。当每次迭代求得的水头变化的绝对值太大或太小时，收敛速度一般不会是最优。当水头变化太大时，计算的水头值会大大超过正确的水头值（这种现象称为过量,overshoot），因此水头会不断进行调整以补偿这种偏大的计算水头变化。这样就会出现振荡的情况（oscillation）。过量严重时会造成迭代发散，而中等程度的过量则会降低收敛速度。当水头变化太小时，则可能出现相反的问题：计算的水头会单调地向正确值缓慢趋近。在严重的情况下，迭代过程可能因计算的水头变化小于收敛指标而停止，但事实上，所计算的水头仍远远偏离正确值。在这种情况下，水均衡计算会显示出巨大的误差。

Weinstein, Stone和Kwan (1969) 建议使用试算法来确定种子的取值。实际工作中可以这样做：首先定ACCL为1，然后用程序计算的或根据经验估计的种子值进行初步计算。在同一个时间段内，注意观察每次迭代后水头变化的趋势。由于迭代参数的循环使用，一般情况下先后两次迭代所计算出的水头变化量会有所不同。但从每次迭代后得到的水头变化量仍不难看出其变化的总趋势，即水头变化值随着迭代次数的增加而上升或下降的趋势。这种总趋势正是我们所感兴趣的。这种趋势一般在迭代试验的后期比较明显。某种程度的振荡现象（即正负交替）是正常的。但持续性振荡则表明过量的情况，即计算的水头变化太大。另一方面，当水头变化太小时，则可观察到非常平缓的水头变化趋势。为适当评估所观察到的水头变化趋势，试算过程中的迭代次数至少应为所选用的种子数目的4至5倍，除非，在此之前迭代已经收敛。

初步试算完成之后则可根据水头变化的情况对种子加以调整：如果水头变化太大，则增加其值2至10倍，反之则减少2至10倍。如果水头变化的趋势不明显，则可试将种子乘以一个数或除以一个数。无论哪种情况，都应进行第二次试算。与第一次试算一样，注意观察计算水头变化情况，并与第一次试算的情况加以比较。如果两次试算都达到收敛则应比较各次的收敛次数。如果两次试算都不收敛，则可比较最后一次迭代后得到的水头变化量。试算过程可以持续下去，以对种子值进行微调。一般来说，每次试算时对种子的改变

量应当逐步减小。但应注意，一般不必进行太多次的试算。每次试算时对种子值的改变量不应小于2倍。

多数情况下，利用这种方法得到的种子值可以满足一个具体的数值模型的计算。即使边界条件，外应力，甚至差分网格发生了变化，也不必对已确定的种子值再做调整。但是，如果这些变化导致了模型不收敛，则应当重新选择种子。应当指出，系数矩阵的对角性 (diagonal dominant) 越强，解对种子选择的依赖程度也就越小。所以，源汇项 (如蒸发蒸腾、河流渗流等) 的使用仅影响主对角线上的元素，可增加解的稳定性。因此，源汇项的增加可提高解的稳定性。

每次迭代计算完成后，程序将 $|\Delta h|_{\max}$ 的值记录下来。这里 $|\Delta h|_{\max}$ 指的是在所有计算单元中，经过本次迭代计算后得到的水头变化的最大绝对值。每个应力期结束时，程序将各次迭代后得到的 $|\Delta h|_{\max}$ 打印在标准输出文件中供用户参考。用户也可以要求程序在每个时间段结束时输出 $|\Delta h|_{\max}$ 的值。除了 $|\Delta h|_{\max}$ 之外，MODFLOW还同时列出 $|\Delta h|_{\max}$ 所在的位置 (即  $i, j, k$ )。最大水头变化的方向则由正负号表示。这些信息可供用户在前面讲到的试算过程中参考。当然，我们假定这些 $|\Delta h|_{\max}$  值能够反映所在点的水头变化情况。

通过调整迭代加速因子，ACCL，还可以提高收敛速度。加大ACCL的值可以增加每次迭代计算出的水头变化量，而减小ACCL的值则可减小水头的变化量。以前谈到的试算法同样可以用来对ACCL的取值进行优化。但不能在同一次试算过程中同时调整种子和ACCL的值。

有些情况下，为避免由于计算水头变化太大而导致干枯计算单元的出现，我们希望减慢收敛速度。这时，最优的收敛速度并不能单纯以迭代次数最少来衡量，而是在不造成水头变化过量的前提之下的最小迭代次数。理解了这点差别之后，我们就可以使用前面提到的试算法对种子和ACCL的取值进行优化了。



### SIP子程序包输入数据及格式

SIP子程序包所需要的各项数据从IUNIT(9)所指定的设备号读入。

对每次模拟:

由子程序SIP1AL读入的数据包括

1.数据名称:	MXITER	NPARM
输入格式:	I10	I10

由子程序SIP1RP读入的数据包括

2.数据名称:	ACCL	HCLOSE	IPCALC	WSEED	IPRSIP
输入格式:	F10.0	F10.0	I10	F10.0	I10

#### 输入数据说明

- MXITER: 最大迭代次数, 通常取值50。
- NPARM: 迭代参数的数目, 通常取值5。
- ACCL: 加速因子。大于或等于零。通常取值为1。如ACCL=0, 则无水头变化。
- HCLOSE: 收敛指标。当计算的最大水头变化的绝对值小于此值时, 迭代结束。它的选值将影响解的精度。
- IPCALC: 确定种子方式标识符:  
         IPCALC =0: 种子由用户输入;  
         IPCALC =1: 由程序自行计算种子值。
- WSEED: 计算迭代参数之种子值。仅当IPCALC=0时需具体指定。
- IPRSIP: 打印频率控制。当时间段数为IPRSIP的倍数时, MODFLOW将各次迭代后最大水头变化(正或负)的结果写入标准输出文件。如果IPRSIP=0, MODFLOW将重新赋值为999。无论IPRSIP取什么值, 在每个应力期结束时, MODFLOW总会打印迭代的情况。

SIP子程序包输入样单 (由用户选择种子)

数据项	解释	输入记录
1	{MXITER, NPARAM}	50 5
2	{ACCL,HCLOSE,IPCALC,WSEED,IPRSIP}	1. 0.01 0 .98 10

SIP子程序包输入样单 (由程序计算种子)

数据项	解释	输入记录
1	{MXITER, NPARAM}	100 6
2	{ACCL,HCLOSE,IPCALC,WSEED,IPRSIP}	1. 0.01 1

[译注: 样单中的数据可能不符合格式要求, 仅供参考。]

## 第十三章 分层逐次超松弛法子程序包

## 概念及程序化

迭代法对大型线性方程组求解的另一方法是逐次超松弛法。MODFLOW使用了八分层逐次超松弛法(Slice Successive Overrelaxation) (简称为SSOR)子程序包。有关逐次超松弛法的介绍可从很多参考书中找到, 包括Peaceman(1977), Crichlow (1977), 和Remson, Horberger and Molz(1971)等早期所著的论文。

在SSOR子程序包中, 逐次超松弛法是将有限差分网格分成“垂向”的“分层”, 如图54所示, 将各单元的差分方程分组归类, 每一分层为一组。每次迭代时, 轮流求解这些方程组, 每分层都用估计的水头值形成新的一套方程组。求解时, 每分层方程组首先用相继两次迭代计算所得的水头差表达。然后该分层的方程组用高斯消元法直接求解, 将与其相邻的分层当成已知(即将计算所得的与其相邻的分层的最新水头值作为“已知”值代入正在求解的分层方程组)。然后, 将从高斯消元法所求得的水头变化值乘以加速因子,  $\omega$  (取值范围通常在1~2之间); 该结果被认为是该分层在那一次迭代的最终水头变化值。将这些水头变化值加到上一次迭代所得的各点的水头值上, 便求得该分层的该次迭代的最终估计水头值。每分层都同一方法按顺序重复直至三维数组中所有的分层都已处理一遍, 才完成一次迭代。然后, 用同样的计算顺序逐次算遍各个分层, 直至相继的两次迭代的水头之差均小于截止条件才结束。

应当注意, 虽然各分层的方程组每次迭代都用一个直接的求解法(高斯消元法)进行求解, 但是整个求解不是直接的而是迭代的。基于最新计算求得的与其相邻分层的水头值, 每一直接解仅给出该分层水头变化的临时值或估计值; 当逐个处理各分层时, 计算值继续改变, 直至满足截止条件。

上述求解过程可通过计算单元的差分方程加以更加详细的说明。下面是第二章中推导出的单个计算单元的差分方程, 但其中添加了第二个上标以表示迭代的次序:

$$CV_{i,j,k-\frac{1}{2}} h_{i,j,k-1}^{m,l} + CC_{i-\frac{1}{2},j,k} h_{i-1,j,k}^{m,l} + CR_{i,j-\frac{1}{2},k} h_{i,j-1,k}^{m,l} \\ + (-CV_{i,j,k-\frac{1}{2}} - CC_{i-\frac{1}{2},j,k} - CR_{i,j-\frac{1}{2},k} - CR_{i,j+\frac{1}{2},k} - CC_{i+\frac{1}{2},j,k} - CV_{i,j,k+\frac{1}{2}})$$

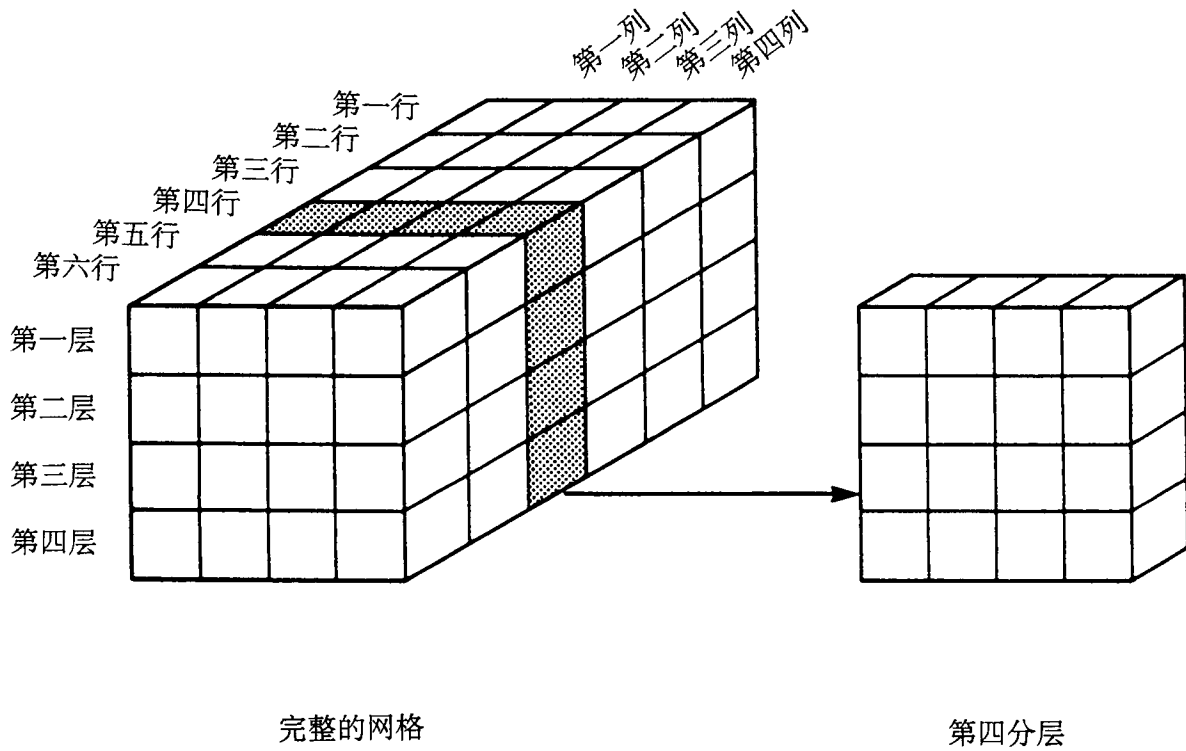


图54. 在SSOR子程序包中，将三维模型分解为垂向分层进行处理。

$$\begin{aligned}
 & + HCOF_{i,j,k})h_{i,j,k}^{m,l} + CR_{i,j+\frac{1}{2},k} h_{i,j+1,k}^{m,l} + CC_{i+\frac{1}{2},j,k} h_{i+1,j,k}^{m,l} \\
 & + CV_{i,j,k+\frac{1}{2}} h_{i,j,k+1}^{m,l} = RHS_{i,j,k} \quad (113)
 \end{aligned}$$

在公式(113)中, 上标 $m$ 为当前时间段, 而上标 $l$ 是指迭代次数。若仿照公式(113)写出下一次迭代,  $l+1$ , 的方程式, 从这一新的公式的两边同时减去方程(113)的左边, 结果可写成:

$$\begin{aligned}
 & CV_{i,j,k-\frac{1}{2}}(h_{i,j,k-1}^{m,l+1} - h_{i,j,k-1}^{m,l}) + CC_{i-\frac{1}{2},j,k}(h_{i-1,j,k}^{m,l+1} - h_{i-1,j,k}^{m,l}) \\
 & + CR_{i,j-\frac{1}{2},k}(h_{i,j-1,k}^{m,l+1} - h_{i,j-1,k}^{m,l}) + (-CV_{i,j,k-\frac{1}{2}} - CC_{i-\frac{1}{2},j,k} - CR_{i,j-\frac{1}{2},k} \\
 & - CV_{i,j,k+\frac{1}{2}} - CC_{i+\frac{1}{2},j,k} - CR_{i,j+\frac{1}{2},k} + HCOF_{i,j,k})(h_{i,j,k}^{m,l+1} - h_{i,j,k}^{m,l}) \\
 & + CR_{i,j+\frac{1}{2},k}(h_{i,j+1,k}^{m,l+1} - h_{i,j+1,k}^{m,l}) + CC_{i+\frac{1}{2},j,k}(h_{i+1,j,k}^{m,l+1} - h_{i+1,j,k}^{m,l}) \\
 & + CV_{i,j,k+\frac{1}{2}}(h_{i,j,k+1}^{m,l+1} - h_{i,j,k+1}^{m,l}) = RHS_{i,j,k} \\
 & - CV_{i,j,k-\frac{1}{2}} h_{i,j,k-1}^{m,l} - CC_{i-\frac{1}{2},j,k} h_{i-1,j,k}^{m,l} - CR_{i,j-\frac{1}{2},k} h_{i,j-1,k}^{m,l} \\
 & - (-CV_{i,j,k-\frac{1}{2}} - CC_{i-\frac{1}{2},j,k} - CR_{i,j-\frac{1}{2},k} - CR_{i,j+\frac{1}{2},k} - CC_{i+\frac{1}{2},j,k} - CV_{i,j,k+\frac{1}{2}} \\
 & + HCOF_{i,j,k})h_{i,j,k}^{m,l} - CR_{i,j+\frac{1}{2},k} h_{i,j+1,k}^{m,l} - CC_{i+\frac{1}{2},j,k} h_{i+1,j,k}^{m,l} - CV_{i,j,k+\frac{1}{2}} h_{i,j,k+1}^{m,l} \quad (114)
 \end{aligned}$$

公式(114)中未知项当成是第 $l$ 次迭代与第 $l+1$ 次迭代之间的计算水头变化值—例如,  $(h_{i,j,k}^{m,l+1} - h_{i,j,k}^{m,l})$ 。注意, 当第 $l$ 次迭代完成时, (114)式的右边全由已知项组成—它包括 RHS 和联立求解得出的水力传导系数, 以及第 $l$ 次迭代后已获得的水头估计值。

现在假定我们按图54所示沿行将模型分成垂向的分层, 将每个单独分层中所有计算单元的方程式分离出来—例如, 从图54中将三维数组的第四行作为第四分层。用公式(114)来说, 当处理第 $i$ 行所对应的第 $i$ 分层时, 我们仍然将该分层各单元的水头变化当作未知项, 但是认为与其相邻的分层的单元的水头变化为已知项。因此, 公式(114)左边项中的两项,

$CC_{i-1/2,j,k}(h_{i-1,j,k}^{m,l+1} - h_{i-1,j,k}^{m,l})$  和  $CC_{i+1/2,j,k}(h_{i+1,j,k}^{m,l+1} - h_{i+1,j,k}^{m,l})$ , 此时已作为已知量处理。如果将这两项移至方程的右边并重新整理, 我们则已经消去  $h_{i-1,j,k}^{m,l}$  和  $h_{i+1,j,k}^{m,l}$  这两项, 仅留下:

$$\begin{aligned}
 & CV_{i,j,k-\frac{1}{2}}(h_{i,j,k-1}^{m,l+1} - h_{i,j,k-1}^{m,l}) + CR_{i,j-\frac{1}{2},k}(h_{i,j-1,k}^{m,l+1} - h_{i,j-1,k}^{m,l}) \\
 & + (-CV_{i,j,k-\frac{1}{2}} - CC_{i-\frac{1}{2},j,k} - CR_{i,j-\frac{1}{2},k} \\
 & - CV_{i,j,k+\frac{1}{2}} - CC_{i+\frac{1}{2},j,k} - CR_{i,j+\frac{1}{2},k} + HCOF_{i,j,k})(h_{i,j,k}^{m,l+1} - h_{i,j,k}^{m,l}) \\
 & + CR_{i,j+\frac{1}{2},k}(h_{i,j+1,k}^{m,l+1} - h_{i,j+1,k}^{m,l}) + CV_{i,j,k+\frac{1}{2}}(h_{i,j,k+1}^{m,l+1} - h_{i,j,k+1}^{m,l}) = RHS_{i,j,k} \\
 & - CV_{i,j,k-\frac{1}{2}}h_{i,j,k-1}^{m,l} - CC_{i-\frac{1}{2},j,k}h_{i-1,j,k}^{m,l+1} - CR_{i,j-\frac{1}{2},k}h_{i,j-1,k}^{m,l} \\
 & - (-CV_{i,j,k-\frac{1}{2}} - CC_{i-\frac{1}{2},j,k} - CR_{i,j-\frac{1}{2},k} - CR_{i,j+\frac{1}{2},k} - CC_{i+\frac{1}{2},j,k} - CV_{i,j,k+\frac{1}{2}} \\
 & + HCOF_{i,j,k})h_{i,j,k}^{m,l} - CR_{i,j+\frac{1}{2},k}h_{i,j+1,k}^{m,l} - CC_{i+\frac{1}{2},j,k}h_{i+1,j,k}^{m,l+1} - CV_{i,j,k+\frac{1}{2}}h_{i,j,k+1}^{m,l} \quad (115)
 \end{aligned}$$

现在假定, 分层的处理是以行数*i*增加的顺序进行的; 对于每次迭代在第*i*分层的计算开始之前, 第*i-1*分层的计算已经完成。这样说来, 当第*i*分层的第*i+1*次迭代处理开始时,  $h_{i-1,j,k}^{m,l+1}$  的值是已知的, 而  $h_{i+1,j,k}^{m,l+1}$  的值则是未知。因此, 处理第*i*分层时,  $CC_{i-1/2,j,k}h_{i-1,j,k}^{m,l+1}$  可直接作为已知项合并, 但  $CC_{i+1/2,j,k}h_{i+1,j,k}^{m,l+1}$  则是未知的。为了克服这一困难, 将上一次迭代得到的  $h_{i+1,j,k}^m$  值, 即  $h_{i+1,j,k}^{m,l}$ , 代替公式(115)右边的  $h_{i+1,j,k}^{m,l+1}$ 。(这样一来, 我们使用的相邻分层的值, 是它们最新计算所得的水头值。)由此可得:

$$\begin{aligned}
 & CV_{i,j,k-\frac{1}{2}}(\tilde{h}_{i,j,k-1}^{m,l+1} - h_{i,j,k-1}^{m,l}) + CR_{i,j-\frac{1}{2},k}(\tilde{h}_{i,j-1,k}^{m,l+1} - h_{i,j-1,k}^{m,l}) \\
 & + (-CV_{i,j,k-\frac{1}{2}} - CC_{i-\frac{1}{2},j,k} - CR_{i,j-\frac{1}{2},k} - CR_{i,j+\frac{1}{2},k} - CC_{i+\frac{1}{2},j,k} - CV_{i,j,k+\frac{1}{2}} \\
 & + HCOF_{i,j,k})(\tilde{h}_{i,j,k}^{m,l+1} - h_{i,j,k}^{m,l}) + CR_{i,j+\frac{1}{2},k}(\tilde{h}_{i,j+1,k}^{m,l+1} - h_{i,j+1,k}^{m,l}) \\
 & + CV_{i,j,k+\frac{1}{2}}(\tilde{h}_{i,j,k+1}^{m,l+1} - h_{i,j,k+1}^{m,l}) = RHS_{i,j,k} - CV_{i,j,k-\frac{1}{2}}h_{i,j,k-1}^{m,l}
 \end{aligned}$$

$$\begin{aligned}
 & -CC_{i-\frac{1}{2},j,k} h_{i-1,j,k}^{m,l+1} - CR_{i,j-\frac{1}{2},k} h_{i,j-1,k}^{m,l} \\
 & -(-CV_{i,j,k-\frac{1}{2}} - CC_{i-\frac{1}{2},j,k} - CR_{i,j-\frac{1}{2},k} - CR_{i,j+\frac{1}{2},k} - CC_{i+\frac{1}{2},j,k} - CV_{i,j,k+\frac{1}{2}} \\
 & + HCOF_{i,j,k}) h_{i,j,k}^{m,l} - CR_{i,j+\frac{1}{2},k} h_{i,j+1,k}^{m,l} - CC_{i+\frac{1}{2},j,k} h_{i+1,j,k}^{m,l} - CV_{i,j,k+\frac{1}{2}} h_{i,j,k+1}^{m,l}
 \end{aligned} \tag{116}$$

方程(116)中,  $\tilde{h}$  表示第( $l+1$ )次迭代时第*i*分层的水头项。使用这样的记法是为了便于清楚描述求解过程。分层中计算单元的数目为NC\*NL, 这里NC是模型中的列数, 而NL是模型中的层数; 每个计算单元都可写出一个与公式(116)类似的方程。因此对NC\*NL个未知数, 则形成一个包含有NC\*NL个方程的方程组。因为层数一般说来是个比较小的数目, 方程组中的方程的总数也比较少, 通常用高斯消元法直接求解的效果也较为理想(注意用高斯消元法这种直接求解的办法对整个三维模型数组这样大的数组求解通常不可行)。

单个分层*i*的整套方程可写成矩阵的形式:

$$[A]_i \{\Delta h\}_i = \{R\}_i \tag{117}$$

这里 $[A]_i$  是第*i*分层的系数矩阵;  $\{\Delta h\}_i$  是第*l*次迭代与第*l+1*次迭代间该分层的计算水头变化值矢量, 用  $\tilde{h}_{i,j,k}^{m,l+1} - h_{i,j,k}^{m,l}$  的估计值表示;  $\{R\}_i$  是“常量”矢量, 用它来表示第*i*分层的公式(116)右侧各项。

高斯消元法解矩阵式(117)便得到该分层的每个计算单元的水头残差  $\tilde{h}_{i,j,k}^{m,l+1} - h_{i,j,k}^{m,l}$ 。将这些项作为从第*l*次迭代到第*l+1*次的计算水头的变化最初估计值。先将每个值乘以加速因子,  $\omega$ , 并将所得的结果与上一迭代所得到的水头值相加, 以求得第*l+1*次迭代后的最终水头估计值; 即:

$$h_{i,j,k}^{m,l+1} = h_{i,j,k}^{m,l} + \omega(\tilde{h}_{i,j,k}^{m,l+1} - h_{i,j,k}^{m,l}) \tag{118}$$

当第*i*分层中每个单元(*j,k*)的  $\tilde{h}_{i,j,k}^{m,l+1}$  值算出后, 便开始下一分层*i+1*的计算处理。所有分层都经过处理后, 这一次迭代便完成了。然后开始下一次迭代计算, 直至精度达到所定要求。

从图55-a可知，公式(117)中的系数矩阵 $[A]_i$  具有对称性并且呈带状(banded)分布，其半带宽度等于模型的层数。因为矩阵对称性的缘故，只需要存储下三角部分就行了；程序中使用了一个二维数组来存储该矩阵，如图55-b所示。这个二维数组的大小为 $NL*NC$ 和 $NL+1$ 。在图示中的例子中， $NL=NC=3$ 。

为达最优收敛速度，SSOR中的加速因子常需调整。用户可使与用本书第十二章中所述的调整SIP“种子”值相似的试错法。

[译注：由于SSOR的计算速度较慢，故实际工作中很少使用。]



$a_{11}$	$a_{12}$		$a_{14}$					
$a_{12}$	$a_{22}$	$a_{23}$		$a_{25}$				
	$a_{23}$	$a_{33}$			$a_{36}$			
$a_{14}$			$a_{44}$	$a_{45}$		$a_{47}$		
	$a_{25}$		$a_{45}$	$a_{55}$	$a_{56}$			
		$a_{36}$		$a_{56}$	$a_{66}$		$a_{68}$	
			$a_{47}$			$a_{77}$		
					$a_{68}$		$a_{88}$	$a_{89}$
							$a_{89}$	$a_{99}$

(a) 单个分层的系数矩阵

$a_{11}$	$a_{22}$	$a_{33}$	$a_{44}$	$a_{55}$	$a_{66}$	$a_{77}$	$a_{88}$	$a_{99}$
$a_{12}$	$a_{23}$		$a_{45}$	$a_{56}$			$a_{89}$	
					$a_{68}$			
$a_{14}$	$a_{25}$	$a_{36}$	$a_{47}$					

(b) 存贮矩阵元素的二维数组

图55. 分层方程组的系数矩阵及其在计算机中存贮形式。

## 第十四章 工具子程序和数据读入

除了主程序以及各个子程序包，MODFLOW还附带有许多工具子程序。这些工具子程序不属于任何一个子程序包，主要用来协助不同的子程序包完成一些输入输出任务。这一类子程序的名称前总冠以“U”以表示工具类（Utility）。MODFLOW共含有八个这样的子程序。它们分别是：

- UBUDSV：将无格式实数数组写入指定的输出文件。该数组包括模型中所有计算单元。
- ULASAV：将无格式实数数组写入指定的输出文件。每次输出模型的一个分层所包含的计算单元。
- ULAPRS和ULAPRW：将一组二维实型数组写入指定的输出文件。每次输出模型的一个分层所包含的计算单元。ULAPRS输出时按列进行（参见图56）：首先输出第一行元素的前N列，然后输出第二行元素的前N列（这里N为一打印行中打印数值的数目）。当各行之前N列都输出之后，再输出第一行的第N+1至2N个元素，然后是第二行的N+1至2N个元素。如此循环下去，直到将该层所有的元素输出完毕为止。而ULAPRW输出时则按行进行：首先输出第一行的所有元素，然后输出第二行的元素，直到该层的所有元素输出完毕为止。数组打印输出的格式列于表2。ULAPRS和ULAPRW仅将数据写入MODFLOW之标准输出文件之中。它们所使用的格式已在第四章输出控制部分列出。
- UCOLNO：专门用来标示由ULAPRS和ULAPRW所输出数据之列号。
- U2DREL：读入二维实型数组。MODFLOW所用的数据多由此子程序读入。
- U2DINT：读入二维整型数组。
- U1DREL：读入一维实型数组。

### 数组输入说明

在输入数组时，MODFLOW要求在数据之前加一输入控制行。用于读入数组的子程序（包括用来读入二维实型数组的U2DREL、读入二维整型数组的U2DINT以及读入一维实型数组的U1DREL）首先读入该控制行，然后再按控制行的规定读入数组。控制行也从相应的设备号读入。例如，补给子程序包利用子程序U2DREL读入补给通量数组RECH。

	1	2	3	4	5	6	7	8	9	10
	11	12	13	14	15	16	17			
1	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79
	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79			
2	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79
	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79			
3	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79
	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79			
4	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79
	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79			
5	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79
	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79			
6	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79
	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79			
7	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79
	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79			

ULAPRW输出格式

	1	2	3	4	5	6	7	8	9	10
1	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79
2	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79
3	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79
4	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79
5	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79
6	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79
7	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79

	11	12	13	14	15	16	17
1	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79
2	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79
3	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79
4	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79
5	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79
6	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79
7	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79	1325.79

ULAPRS输出格式

图56. 工具子程序ULAPRW和ULAPRS的输出示例。该模型层含有七行十七列。

表2. 工具子程序ULAPRS 和 ULAPRW 使用的打印格式代码

<u>IPRN</u>	<u>格 式</u>
1	11G10.3
2	9G13.6
3	15F7.1
4	15F7.2
5	15F7.3
6	15F7.4
7	20F5.0
8	20F5.1
9	20F5.2
10	20F5.3
11	20F5.4
12	10G11.4

用户在IUNIT (8) 的位置的选定整数, 也是读入数组RECH的设备号。

输入控制行的格式如下:

(1) 实型数组 (U2DREL和UIDREL) :

数据名称: LOCAT CNSTNT (FMTIN) IPRN

输入格式: I10 F10.0 5A4 I10

(2) 整型数组 (U2DINT) :

数据名称: LOCAT ICONST (FMTIN) IPRN

输入格式: I10 I10 5A4 I10

LOCAT: 含有输入数据的文件设备号。

如果LOCAT<0: 从与LOCAT符号相反的设备号读入无格式数据;

如果LOCAT=0: 数组各元素为一常数。该常数的值由CNSTNT或ICONST给出。

如果LOCAT>0: 从设备号LOCAT 的输入文件中, 按FMTIN的格式输入数据。

CNSTNT/ICONST: 为一常数。它们的作用取决于LOCAT是否等于零。

如果 LOCAT=0: 则该数组各元素的值均等于CNSTNT/ICONST的值;

如果LOCAT≠0且CNSTNT/ICONST不为零时, CNSTNT/ICONST的值作为一个因数。MODFLOW将读入的数据乘以该因数之后再存入相应的数组。如果CNSTNT/ICONST为零, 则忽略不计。

FMTIN: 输入数据之格式。仅当LOCAT为正数时, FMTIN才有意义。FMTIN的定义完全按照FORTRAN语言的规定, 例如F10.2, I4等。值得注意的是, 格式说明符必须置于一对圆括号之中。例如: (15F5.0), (15I5)等等。忘记将格式说明符放在括号之中, 将会导致程序出错。另外, 同一组数据应具有相同的输入格式。

IPRN: 数据打印标识符。用来表明是否将读入的数据写入标准输出文件, 以供检查, 同时, 也作为打印输出时的格式代码。MODFLOW仅当LOCAT不为零时才读入该值。如果IPRN为一负数, MODFLOW将不打印读入的数据。当LOCAT≠0并且当IPRN大于零时, IPRN的值同时用作输出格式的代码。注意, 各个子程序输出时的格式代码完全不同。此外, 如果IPRN超出下列各代码的最大值时, 程序将会按IPRN=0的格式输出。例如, 如果用户输入的IPRN为15时, 数据由子程序U2DREL读入时, MODFLOW将会以IPRN=0所指定的格式(10G11.4)将读入数据写入

标准输出文件。

下面将IPRN的值与其所表示的输出格式列出：

IPRN	U2DREL	U2DINT	U1DREL
0	10G11.4	10I11	10G12.5
1	11G10.3	60I1	
2	9G13.6	40I2	
3	15F7.1	30I3	
4	15F7.2	25I4	
5	15F7.3	20I5	
6	20F7.4		
7	20F5.0		
8	20F5.1		
9	20F5.2		
10	20F5.3		
11	20F5.4		
12	10G11.4		

## 参考文献-1

- Collins, R. E. 1961, Flow of fluids through porous materials; New York, Reinhold Publishing Corp., 270 p.
- Crichlow, Henry B., 1977, Modern reservoir engineering – A simulation approach; Englewood Cliffs, N. J., Prentice Hall Inc., 354 p.
- McDonald, M. G., and Harbaugh, A. W. 1984, A modular three-dimensional finite-difference ground-water flow model: U. S. Geological Survey Open-File Report 83-875, 528 p.
- Peaceman, Donald W., 1977, Fundamentals of numerical reservoir simulation; New York, Elsevier Scientific Publishing Company, 176 p.
- Remson, Irwin, Hornberger, George M. and Molz, Fred J., 1971, Numerical methods in subsurface hydrology; New York, Wiley-Interscience, 389 p.
- Rushton, K. R., Redshaw S. C., 1979, Seepage and groundwater flow-numerical analysis by analog and digital methods; New York, John Wiley and Sons.
- Trescott, Peter C., 1975, Documentation of finite-difference model for simulation of three-dimensional ground-water flow; U. S. Geological Survey Open-File Report 75-438, 32 p.
- Trescott, Peter C., and Larson, S. P., 1976, Supplement to Open-File Report 75-348, Documentation of finite-difference model for simulation of three-dimensional ground-water flow; U. S. Geological Survey Open-File Report 76-591, 21 p.
- Trescott, Peter C., Pinder, G. F., and Larson, S. P., 1976, Finite-difference model for aquifer simulation in two-dimensions with results of numerical experiments: U. S. Geological Survey Techniques of Water-Resources Investigations, Book 7, Chapter C1, 116 p.
- Weinstein, H. C., Stone, H. L. and Kwan, T. V., 1969, Iterative procedure for solution of systems of parabolic and elliptic equations in three-dimensions: Indus. Engineering Chemistry Fundamentals, v. 8, no. 2, p. 281-287.

## 参考文献-2

- Bennett, G. D., A. L. Kontis, and S. P. Larson, 1982, Representation of multiaquifer well effects in three-dimensional ground-water flow simulation, *Ground Water*, 20(3), pp. 334-344.
- Guo, W. and C. J. Neville, 1998, Adaptation of MODFLOW for transient air flow simulations, in *Proceedings of MODFLOW '98*, Colorado School of Mines, Golden Colorado, pp. 157-164.
- Guo, W. and G. D. Bennett, 1998, Simulation of saltwater/fresh water flows using MODFLOW, in *Proceedings of MODFLOW '98*, Colorado School of Mines, Golden Colorado, pp. 267-274.
- Harbaugh, A. W. and M. G. McDonald, 1996, User's documentation for MODFLOW-96, and update to the U.S. Geological Survey modular finite-difference ground-water flow model, U. S. Geological Survey Open-file Report 96-486, 220 p.
- Hill, M. C., 1990a, Preconditioned conjugate-gradient 2 (PCG2), a computer program for solving ground-water flow equations: U.S. Geological Survey Water-Resources Investigations Report 90-4048, 43 p.
- Hill, M. C., 1990b, Solving groundwater flow problems by conjugate-gradient methods and the strongly implicit procedure, *Water Resources Research*, 26(9), pp. 1961-1969.
- Hsieh, P. A., and J. R. Freckleton, 1993, Documentation of a computer program to simulate horizontal-flow barriers using the U.S. Geological Survey modular three-dimensional finite-difference ground-water flow model: U.S. Geological Survey Open-File Report 92-477, 32 p.
- Kuiper, L. A., 1981, A comparison of incomplete Cholesky conjugate gradient method with strongly implicit method as applied to the solution of two-dimensional groundwater flow equations, *Water Resources Research*, 17(4), pp. 1082-1086.
- Leake, S.A., and D. E. Prudic, 1988, Documentation of a computer program to simulate aquifer-system compaction using the modular finite-difference ground-water flow model: U.S. Geological Survey Open-File Report 88-482, 80 p.
- McDonald, M. G., A. W. Harbaugh, 1988, A modular three-dimensional finite-difference ground-water flow model, *Techniques of Water Resources Investigations*, Book 6, A1, U. S. Geological Survey.
- McDonald, M. G., A. W. Harbaugh, B. R. Orr, and D. J. Ackerman, 1992, A method of converting no-flow cells to variable-head cells for the U.S. Geological Survey



modular finite-difference ground-water flow model: U.S. Geological Survey Open-File Report 91-536, 99 p.

Prudic, D. E., 1989, Documentation of a computer program to simulate stream-aquifer relations using a modular, finite-difference, ground-water flow model: U.S. Geological Survey Open-File Report 88-729, 113 p.

Verga, R. S., 1962, Matrix iterative Analysis, Prentice Hall, Englewood Cliffs, N. J., 322 p.

李荣华, 冯果忱, 1980, 微分方程数值解法, 人民教育出版社, 北京, 418 页。

## 附录A 干枯计算单元的几种处理方法

MODFLOW在迭代计算过程，一旦发现非承压含水层中某个计算单元的计算水头低于该单元的底面标高就将它列为干枯计算单元，并在以后的计算中按无效计算单元处理。如果这个计算单元中含有抽水井、河流等源汇，这些源汇也就随之而被排除出后面的计算过程。对于干枯计算单元的处理比较困难。MODFLOW中对干枯计算单元的处理方法还不够完善。

为避免干枯计算单元的出现，首先应当适当调整用于控制迭代计算的参数，以减小计算水位的震荡范围。例如，减小迭代求解中的加速因子等。其次应注意初始水头的选择。如果初始水头与实际情况比较接近，而且空间变化比较平缓，则出现干枯计算单元的可能性会有所降低。除了初始水头的分布，还应当仔细检查其它输入数据是否合理，例如抽水井的位置、抽水量、透水滤网的深度、含水层底板标高、贮水系数以及降水补给量等。

为克服88年版MODFLOW中计算单元“一旦变干就永远不能恢复”的问题，美国地质调查局的McDonald等人于1992年推出了一个新的子程序包，称为BCF2，并用来取代原有的BCF1子程序包。在BCF2中，MODFLOW允许干枯计算单元重新成为有效计算单元。但从BCF2的使用情况来看，效果并不理想。虽然干枯计算单元的问题有所解决，但同时也造成了计算难以收敛的问题。

在实际应用中，有人经常通过修改MODFLOW源程序(子程序SBCF1H)来保证计算单元不会出现干枯现象。例如，预先确定一个基本饱和厚度，在程序计算水力传导系数时，一旦发现计算水头低于底面标高，就用这个基本饱和厚度来计算导水系数。这样一来，即使计算水头低于底面标高，导水系数也不为零，从而避免了干枯计算单元的出现。这种方法并不严格，故使用时应当慎重。基本饱和厚度的取值并无标准可循，由用户自行决定。一般可取值0.1米左右，或取含水层的厚度的1%至5%。

还有一种理论上较为合理的解决方法是在MODFLOW中加入真正的非饱和流计算。这样做也需要修改MODFLOW源程序。首先定义一个反映非饱和带中水头（此时压力水头为负值）与渗透系数关系的通用函数，然后根据计算水头求出渗透系数和导水系数。这样做可以避免计算单元变干，但无疑会增加流动方程的非线性而造成求解时收敛上的问题。另外，用来反映水头与渗透系数通用的函数也很难确定。这种方法存在的一个严重缺陷是仍无法解决定流量抽水井的问题。从根本上来说，定流量抽水井的设定是许多计算单元变干的原因所在。在MODFLOW中，抽水井的流量是由用户事先确定的，与抽水井所在计算单

元的导水系数无关。在同一应力期中，抽水量保持不变。这与野外工作遇到的情况有所不同。在实际工作中，井的抽水量可能随含水层的水位下降而降低。

总之，干枯计算单元的处理是比较困难的。到目前为止，这个问题还没有令人满意的解决办法。我们在这里介绍的几种处理方法，都属于使用MODFLOW过程中的变通办法。这些方法都缺乏严格的论证，仅供读者参考。

## 附录B PCG2子程序包

有限差分方程组建立起来之后, 下面一个重要步骤就是对该线性方程组联立求解。一般来说, 求解的方法可以分为直接求解方法和迭代求解方法。直接求解不用进行迭代, 可直接求得线性方程组的解。常用的直接求解方法如高斯消元法、逆矩阵法等。但由于直接求解方法需要大量的内存, 故并不常用于工程计算。迭代求解的方法也有许多。它们的共同特点是通过一系列的迭代运算, 使每次迭代得到的近似解逐渐趋于真实解。当解的变化量(有时为残差的变化量)小于一个事先设定的收敛指标时, 则称迭代已经收敛, 而得到的结果则称为原线性方程组的解。

MODFLOW88年版含有两种迭代求解子程序包: SIP和SSOR。最近, Harbaugh(1996)又推出了一种直接求解子程序包。但由于直接求解子程序包仅适用于计算单元数目较少的情况, 故直接求解的方法不太具有实用价值。由Hill (1990a)推出的PCG2子程序包则是另一种迭代求解子程序包, 并得到了相当广泛的应用。因此, 我们对PCG2子程序包的使用法作一介绍。

PCG(即 Preconditioned Conjugate-gradient)也可称为预调共轭梯度法或预调共轭斜量法。它也是一种对大型线性方程组迭代求解的方法。在1988年版本MODFLOW中并不包括这种求解子程序包。PCG2子程序包于1990年由美国地质调查局的Hill推出。由于它的应用比较普遍, 我们在这里也加以介绍。

PCG方法已经广泛地应用于对大型线性方程组的迭代求解。在这方面有许多文献可供参考(Varga, 1962; Remson等, 1971; 李荣华和冯果忱, 1980)。PCG2子程序包中含有两种不同PCG的算法: MICCG (Modified Incomplete Cholesky Conjugate-Gradient)法和POLCG (Least-Squares Polynomial Conjugate-Gradient)法。这两种方法各有其优缺点。MICCG法较适用于标量型电子计算机(Scalar Computers), 而POLCG较适用于向量型电子计算机(Vector Computers)。与SIP方法相比, POLCG方法需要较少计算机内存。在多数情况下, MICCG方法比SIP方法更有效。但对于非线性问题, PCG方法又常不如SIP方法。Hill (1990b)曾通过大量的例子比较了SIP、POLCG和MICCG等解法的优缺点。

对于非线性问题, Kuiper (1981)发现, 如果在Picard法迭代之间进行若干次迭代计算(称内循环迭代)PCG方法会变得更有效。在内循环迭代过程中, 系数矩阵保持不变, 仅

为水头本身求解。内循环完成之后，再进行外循环。每次外循环开始时，MODFLOW需要重新计算导水系数等参数以及系数矩阵。对于内循环次数有以下规定：(1)由用户在输入数据中指定，即ITERI的值；(2)达到收敛指标。内循环结束后，计算转至外循环。PCG2首先根据内循环计算的结果对系统进行调整，然后开始新一轮内循环。如果在新一轮内循环中第一次迭代计算的结果就满足收敛指标，则表示本时间段的求解收敛。否则就继续进行内循环迭代。

另外，值得注意的是，PCG2方法规定的收敛指标不仅仅指水头，还包括计算单元之间的流量。只有当这两个指标同时得到满足时，计算才以收敛而结束。

## PCG2子程序包输入数据及格式

由于PCG2并非标准MODFLOW的子程序包，故MODFLOW本身并没有指定PCG2子程序包中在IUNIT中的位置。一般PCG2子程序包使用IUNIT(13)所定义的输入文件设备号。

对每次模拟过程:

由子程序PCG2AL读入的数据

1. 数据名称: MXITER ITERI NPCOND

输入格式: I10 I10 I10

由子程序PCG2RP读入的数据

2. 数据名称: HCLOSE RCLOSE RELAX NBPOL IPRPCG MUTPCG IPCGCD

输入格式: F10.0 F10.0 F10.0 I10 I10 I10 I10

### 输入数据说明

- MXITER:** 最大外循环迭代次数。对线性问题，MXITER应为1；而对于非线性流动问题，MXITER的值应比较大。
- ITERI:** 最大内循环迭代次数。
- NPCOND:** 预调方法选择指示：  
NPCOND=1: MICCG方法（用于标量计算机）；  
NPCOND=2: POLCG方法（用于矢量或并行计算机）；
- HCLOSE:** 水头收敛指标。
- RCLOSE:** 流量收敛指标。这里的流量指的是迭代计算过程中流量的残差值。当流量残差的最大值低于RCLOSE时，则称解满足流量收敛指标。
- RELAX:** 在MICCG方法中使用的松弛因子。一般RELAX=1，有些情况下使用稍小于1的RELAX可以加快收敛过程。
- NBPOL:** 仅用于POLCG方法中。NBPOL的值即代表特征值，也可能作为操作标识码。如果NBPOL=2，则PCG2按此值求解。如果NBPOL不为2，则PCG2子

程序包会自行计算最大特征值。

IPRPCG: 打印频率控制。用法与前面提到的IPRSIP、IPRSOR相同。

MUTPCG: 打印方式控制:

MUTPCG=1, 仅打印迭代次数; MUTPCG=2, 不打印;

MUTPCG=0, 打印迭代次数、最大水头残差和流量残差。

IPCGCD: 仅用于NPCOND=1的情况。它用来指示程序如何进行Cholesky分解。绝大多数情况下, IPCGCD应为0。在目前的版本中, IPCGCD的值总为零。

PCG2 子程序包输入样单 (线性问题)

数据项	解释	输入记录
1	{MXITER, ITER1, NPCOND}	1 99 2
2	{HCLOSE, RCLOSE, RELAX, NBPOL, IPRPCG, MUTPCG, IPCGCD}	0.001 0.001 1. 2 1

PCG2 子程序包输入样单 (非线性问题)

数据项	解释	输入记录
1	{MXITER, ITER1, NPCOND}	10 5 1
2	{HCLOSE, RCLOSE, RELAX, NBPOL, IPRPCG, MUTPCG, IPCGCD}	0.01 0.01 1. 2 1

[译注: 样单中的数据可能不符合格式要求, 仅供参考。]



## 附录C MODFLOW计算结果的处理

### MODFLOW的输出种类

MODFLOW的输出可分为三类:

- (1)屏幕输出: 主要内容为迭代运算的情况以及程序运行中的错误信息。通过对每次迭代计算得到的最大水头变化以及该计算单元所处位置的观察, 有助于用户检查模型设计的合理性以及输入数据的正确性。而对造成程序运行中断的错误信息的理解, 有助于纠正错误、排除故障。
- (2)标准输出文件: MODFLOW在程序运行过程中, 会将模拟计算的情况、输入数据、结果以及水均衡收支计算的情况写入这个标准输出文件以供参考。有些导致程序中止运行的原因也可以在这个文件中找到: 如“X”数组太小, 运算不收敛等等。此外, MODFLOW还将列出所有干枯计算单元的位置。水均衡收支表则是鉴定一个模型的设计及运算结果质量的重要指标。根据我们的经验, 当一个新建立的模型第一次完成运行后, 用户应当认真检查该次运行时产生的标准输出文件。在模型的调试和校准(calibration)过程中, 也应经常检查水均衡情况。这些对保证计算结果的质量都是非常重要的。
- (3)计算结果: 按用户要求, MODFLOW的计算结果可能分别以无格式的形式写入三个输出文件。它们的内容分别为水头、降深以及流量。由于这些文件均为无格式文件, 以二进制格式记录相应的内容, 故无法直接阅读。一般要通过程序来解读。尽管目前已有不少商业化软件可以直接将计算结果显示出来, 但在实际工作中我们常常需要从这些无格式文件中读取某部分数据进行分析。不幸的是, 对MODFLOW输出的无格式文件的解读并非一件易事。加上各种程序编译系统对无格式输出文件定义的不同, 常使用户感到困难。下面就这个问题专门进行讨论。

### 无格式输出

为减小输出文件的长度以及提高输出操作的速度, MODFLOW在输出结果(水头、降深以及流量)时采用了一种无格式(Unformatted)的方式。所输出的内容并不是ASCII码

的形式，而是以二进制(binary)的形式记录在文件中。这种文件不能以文本文件(TEXT)的方式去解读。

以无格式方式输出数据时，程序将按照数据的长度依次将数据以二进制机器码的形式写入文件。故在解读时，不仅要了解输出文件的内容，还要知道文件中数据的种类（即字长）以及排列的顺序。一般来说，这些情况都可以从源程序中输出部分获得。故在编写解读程序时，应当参考源程序中的输出部分，并按照输出数据的内容、长度以及顺序进行解读。只要有一个数据在解读时错位，其后所有的数据都会变得毫无意义。

值得注意的是，不同的编译程序对无格式文件的定义也可能不同，而且不能混用。例如，由Lahey发行的FORTRAN编译程序所定义的非格式文件就无法用Microsoft发行的FORTRAN编译程序所编译的解读程序直接去读。故在编写解读程序时，应当注意所使用的编译程序与MODFLOW的编译方式的差别。为解决这种矛盾，用户最好将MODFLOW源程序重新编译，这样，用同样FORTRAN编译程序编译的解读程序就可以正确读出MODFLOW的计算结果了。

### 无格式水头和降深文件的解读

MODFLOW计算的水头和降深分别以无格式的形式写入相应的输出文件。这两个文件虽然内容不同，但具有完全相同的形式。下面以无格式水头文件为例介绍如何解读。

无格式水头文件以模型的分层为单位，每个数据块相对于模型的一个分层。每个数据块占有相同的长度。一个文件中含有数据块的多少与模型的层数、输出的频率以及模拟时间的长度有关。这些数据块具有相同的结构。故只要能够对一个数据块进行解读，就很容易通过循环对文件中所有的数据块进行解读。

每个数据块中含有两部分：第一部分为定义行(header)，第二部分为数据。定义行中共由八项，分别为：当前应力期数、当前时间段数、步长、模拟历时、数据名称、总行数、总列数、当前层数。其中当前应力期数、当前时间段数、行数(NROW)、列数(NCOL)和当前层数为整型数，步长和模拟历时为实型数，而数据名称为16位字符串。在水头文件中，名称为“HEAD”；在降深文件中，该字符串表示“DRAWDOWN”（降深）。定义行后面便是输出数据。它包括该层中所存计算单元的水头值。数据的排列方式为：该层第

一行中各列计算单元的水头，然后为第二行的各列，直至最后一行的各列计算单元。

通过下面的解读程序，我们从中很容易看出解读顺序。

```
C  EXAMPLE PROGRAM READING UNFORMATTED HEAD OR DRAWDOWN FILES
C  Weixing Guo, 05/22/1998
      PARAMETER (NROW=10,NCOL=10,NLAY=2)
      INTEGER*4 NC,NR,ILAY, KSTP, KPER
      REAL PERTIM, TOTIM
      Character*16 TEXT
      DIMENSION H(ncol,nrow,nlay)

C  Open unformatted head file
      OPEN(unit=15,file='MODFLOW.UFH',form='unformatted')
      WRITE(*,11)
11  FORMAT(1X,'Reading Unformatted File.....')
      DO 166 it=1,100 !Number of stress periods and time steps
C  Read heads layer by layer
      DO 66 K=1,NLAY
      READ(15,END=1) KSTP,KPER,PERTIM,TOTIM,TEXT,NC,NR,ILAY
      READ(15)((H(J,I,K),J=1,NCOL),I=1,NROW)
66  CONTINUE

C  Codes for output

166  CONTINUE
      1  WRITE(*,*) ' FINISHED '
      STOP
      END
```

## 无格式流量文件的解读

MODFLOW可以按用户的要求，输出一个记录各项流量的无格式文件。与前面讲到的水头文件相似，流量文件内的数据也是按数据块的形式存写的。但不同的是，流量文件中的各个数据模块所代表的内容彼此可能很不相同。它们可能表示河段的流量，单元之间的流量等等。流量文件所包含数据块的数量与模型的层数、输出的项目、输出的频率、模拟的性质（稳定流或非稳定流）以及时间段的数目有关。

与水头文件中的数据模块相同，流量文件中的数据模块也有两部分：定义行和数据。定义行包括六项：当前时间段数、当前应力期数、数据名称、总行数、总列数以及当前层数。其中当前应力期数、当前时间段数、总行数、总列数以及当前层数为整型数。而数据名称为16位字符串。该字符串的内容由数据的性质而定，例如，“Storage”（贮水量），“River”（河流），“Well”（井）等。

定义行后面便是输出数据。它包括该层中所有计算单元的该项（即定义行中字符串所表示的内容）流量。与无格式水头文件不同的是，对一个三维模型来说，其流量数据是以三维数组的形式输出的。如果某一项流量在该单元不存在，其流量以零表示。如果在同一单元中同时有两项以上同一性质的流量，则此处所表示的流量为它们的和。数据排列的方式为：从第一层第一行第一列开始，然后为第一层第一行第二列，……。第一层第一行之后为第一层第二行之第一列，第二列，……。直至最后一行中的各列。然后是第二层的数据，……。直至最后一层的最后一列。

以88年MODFLOW为例，在同一时间段内，各数据块的排列顺序如下（具体情况要视用户的选定）：

1. 贮水量流量 (STORAGE)
2. 定水头计算单元流量 (CONSTANT HEAD)
3. 通过右侧面的流量 (FLOW RIGHT FACE)
4. 通过前侧面的流量 (FLOW FRONT FACE)
5. 通过底面的流量 (FLOW LOWER FACE)
6. 水井流量 (WELLS)
7. 沟渠流量 (DRAINS)
8. 补给流量 (RECHARGE)

9. 蒸发蒸腾流量 (ET)
10. 河流流量 (RIVER LEAKAGE)
11. 通用水头边界流量 (HEAD DEP BOUNDS)

括号中的内容为名称字符串所表示的内容。

为进一步说明该文件的解读, 下面是一个用于解读流量文件的FORTRAN程序。再强调一次, 这些项可能不会都出现。哪些项出现应视输出定义和子程序包的选用而定。对于未出现的数据块, 相应的读入部分则应从程序中删除。

```

C  EXAMPLE PROGRAM READING UNFORMATTED FLOW FILES
C  Weixing Guo, 05/22/1998
      PARAMETER (NROW=10,NCOL=10,NLAY=2)
      DIMENSION CONST(NCOL,NROW,NLAY),RIGHT(NCOL,NROW,NLAY),
& FRONT(NCOL,NROW,NLAY),FLOWER(NCOL,NROW,NLAY),
& RIVER(NCOL,NROW,NLAY),DRN(NCOL,NROW,NLAY),
& RCH(NCOL,NROW,NLAY),WEL(NCOL,NROW,NLAY),
& STG(NCOL,NROW,NLAY),GHB(NCOL,NROW,NLAY),
& EVT(NCOL,NROW,NLAY)
      CHARACTER TEXT*16
      INTEGER KSTP, KPER, NC, NR, NL
      OPEN(11,FILE='MODFLOW.UFF',FORM='UNFORMATTED')
      DO 200 IT=1,100 !Number of stress periods and time steps
C STORAGE  only for transient flow
      READ(11,END=1) KSTP, KPER, TEXT, NC, NR, NL
      PRINT*, 'STORAGE : ',TEXT
      READ(11)((STG(J,I,K),J=1,NCOL),I=1,NROW),K=1,NLAY)
C Flow from Constant Head Cells
      READ(11) KSTP, KPER, TEXT, NC, NR, NL
      PRINT*, 'CONSTANT-HEAD : ',TEXT
      READ(11)((CONST(J,I,K),J=1,NCOL),I=1,NROW),K=1,NLAY)

```

## C Flow from Right Faces

```
READ(11) KSTP, KPER, TEXT, NC, NR, NL
PRINT*, 'RIGHT FACE : ',TEXT
READ(11) ((RIGHT(J,I,K),J=1,NCOL),I=1,NROW),K=1,NLAY)
```

## C Flow from Front Faces

```
READ(11) KSTP, KPER, TEXT, NC, NR, NL
PRINT*, 'FRONT FACE : ',TEXT
READ(11) ((FRONT(J,I,K),J=1,NCOL),I=1,NROW),K=1,NLAY)
```

## C Flow from Lower Faces

```
READ(11) KSTP, KPER, TEXT, NC, NR, NL
PRINT*, 'LOWER FACE : ',TEXT
READ(11) ((FLOWER(J,I,K),J=1,NCOL),I=1,NROW),K=1,NLAY)
```

## C WELLS

```
READ(11) KSTP, KPER, TEXT, NC, NR, NL
PRINT*, 'WELL : ',TEXT
READ(11) ((WEL(J,I,K),J=1,NCOL),I=1,NROW),K=1,NLAY)
```

## C DRAINS

```
READ(11) KSTP, KPER, TEXT, NC, NR, NL
PRINT*, 'DRAIN : ',TEXT
READ(11) ((DRN(J,I,K),J=1,NCOL),I=1,NROW),K=1,NLAY)
```

## C RECHARGE

```
READ(11) KSTP, KPER, TEXT, NC, NR, NL
PRINT*, 'RECHARGE : ',TEXT
READ(11) ((RCH(J,I,K),J=1,NCOL),I=1,NROW),K=1,NLAY)
```

## C EVT

```
READ(11) KSTP, KPER, TEXT, NC, NR, NL
PRINT*, 'EVT : ',TEXT
READ(11) ((EVT(J,I,K),J=1,NCOL),I=1,NROW),K=1,NLAY)
```

## C RIVERS

```
READ(11) KSTP, KPER, TEXT, NC, NR, NL
PRINT*, 'RIVER : ',TEXT
```

```
READ(11)((RIVER(J,I,K),J=1,NCOL),I=1,NROW),K=1,NLAY)
```

C GHB

```
READ(11) KSTP, KPER, TEXT, NC, NR, NL
```

```
PRINT*, 'GENERAL HEAD : ',TEXT
```

```
READ(11)((GHB(J,I,K),J=1,NCOL),I=1,NROW),K=1,NLAY)
```

C Codes for output

```
200 CONTINUE
```

```
1 WRITE(*,*) ' Finished '
```

```
STOP
```

```
END
```

## 附录 D 一些新开发的子程序包

### **The stream package (STR1):**

Prudic, D. E., 1989. Documentation of a computer program to simulate stream-aquifer relations using a modular, finite-difference, groundwater flow model: U. S. Geological Survey Open-File Report 88-729, 113 p.

### **The interbed-storage and time-variant specific-head package:**

Leake, S. A., and Prudic, D. E., 1988. Documentation of a computer program to simulate aquifer-system compaction using the modular finite-difference groundwater flow model, U. S. Geological Survey Open-File Report 88-482, 80 p.

### **The general finite-difference flow package:**

Harbaugh, A. W., 1992. A generalized finite-difference formulation for the U. S. Geological Survey modular three-dimensional finite-difference groundwater flow mode, U. S. Geological Survey Open-file Report 91-494, 60 p.

### **The version 2 of the block-centered flow package (BCF2):**

McDonald, M. G., Harbaugh, A. W., Orr, B. R., and Ackerman, D. J., 1992. A method of converting no-flow cells to variable-head cells for the U. S. Geological Survey modular finite-difference groundwater flow model, U. S. Geological Survey Open-file Report 91-536, 99 p.

### **The version 3 of the block-centered flow package (BCF3):**

Goode, D. J., and Appeal, C. E., 1992. Finite-difference interblock transmissivity for unconfined aquifers and for aquifers having smoothly varying transmissivity, U. S. Geological Survey Water Resources Investigations report 92-4124, 79 p.

### **The horizontal flow barrier package (HFB1):**

Hsieh, P. A., and Freckleton, J. R., 1993. Documentation of a computer program to simulate horizontal-flow barriers using the U.S. Geological Survey modular three-dimensional finite-difference flow model, U. S. Geological Survey Open-File Report 92-477, 32 p.



**The transient-leakage package (TLK1):**

Leake, S. A., Leahy, P. P. and Navoy, A. S., 1994. Documentation of a computer program to simulate transient leakage from confining units using the modular finite-difference groundwater flow model, U. S. Geological Survey Open-File Report 94-59, 70 p.

**The direct solution package (DE45):**

Harbaugh, A. W., 1995. Direct solution package based on alternating diagonal ordering for the U. S. Geological Survey modular finite-difference groundwater flow model, U. S. Geological Survey Open-File Report 95-288, 46 p.

**The reservoir leakage package (RES1):**

Fenske, J. P., Leake, S. A. and Prudic, D. E., 1996. Documentation of a computer program (RES1) to simulate leakage from reservoirs using the modular finite-difference groundwater flow model (MODFLOW), U. S. Geological Survey Open-File Report 96-364, 51 p.