



矩形含水层井流模型求解的软件实现^{*}

刘志涛, 彭磊, 张明德, 杨宝林, 王忠民, 吕金

(山东省鲁北地质工程勘察院, 山东 德州 253015)

摘要: 矩形含水层井流模型的求解过程比较繁琐, 计算工作量较大, 而且容易出错, 而利用计算机编程技术可使求解变得极为简单、快捷, 从而极大地提高工作效率。该文以矩形隔水边界承压井流模型为例, 研究了实现软件求解该类问题的思路。

关键词: 矩形含水层; 井流模型; 计算机; 编程技术

中图分类号: F062.2; F129.1 **文献标识码:** B

在进行矩形含水层井流模型的水位计算时, 发现其计算过程繁琐复杂, 运算量也较大, 于是便考虑利用计算机编程技术实现该类问题的求解, 并经过研究简单制作了一个计算软件。下面以矩形隔水边界承压井流模型为例来论述其实现思路。

1 矩形隔水边界承压井流模型

假定矩形含水层边界长度和宽度分别为 A 和 B , 边界条件为隔水边界, 其他条件与泰斯井流模型相同, 这时关于水位降深 s 的数学模型为^[1]:

$$\frac{\partial^2 s}{\partial x^2} + \frac{\partial^2 s}{\partial y^2} = \frac{S}{T} \frac{\partial s}{\partial t} \quad (0 < x < A, 0 < y < B, t > 0)$$

$$s(x, y, 0) = 0 \quad (0 < x < A, 0 < y < B)$$

$$\lim_{r_1 \rightarrow 0} (r_1 \int_0^{2\pi} \frac{\partial s}{\partial r_1} d\theta_i) = -\frac{Q}{T} \quad (t > 0) \quad (式 1)$$

$$\frac{\partial s}{\partial x} \Big|_{x=0} = \frac{\partial s}{\partial x} \Big|_{x=A} = 0 \quad (0 < y < B, t > 0)$$

$$\frac{\partial s}{\partial y} \Big|_{y=0} = \frac{\partial s}{\partial y} \Big|_{y=B} = 0 \quad (0 < x < A, t > 0)$$

式中: s —降深; S —释水系数; T —导水系数; Q —抽水井流量; t —抽水时间; (x, y) —直角坐标系坐标; (r, θ) —局部辅助坐标系坐标。

2 问题求解

该类问题一般是通过映射将原问题转化为一个等价的无界含水层群井井流问题, 然后利用无界含

水层群井的叠加原理求得问题的解^[2], 经映射后其模型如图 1 所示^[3]。

映射后所对应的无界含水层数学模型为:

$$\frac{\partial^2 s}{\partial x^2} + \frac{\partial^2 s}{\partial y^2} = \frac{S}{T} \frac{\partial s}{\partial t} \quad (-\infty < x < \infty, -\infty < y < \infty, t > 0)$$

$$s(x, y, 0) = 0 \quad (-\infty < x < \infty, -\infty < y < \infty) \quad (式 2)$$

$$\lim_{(x^2+y^2) \rightarrow \infty} s = 0 \quad (-\infty < x < \infty, -\infty < y < \infty, t > 0)$$

$$\lim_{r_i \rightarrow 0} (r_i \int_0^{2\pi} \frac{\partial s}{\partial r_i} d\theta_i) = -\frac{Q}{T} \quad (t > 0, i = 1, 2, \dots, n)$$

式中: n —映射后的井点数; 其他各字符代号含义同上。

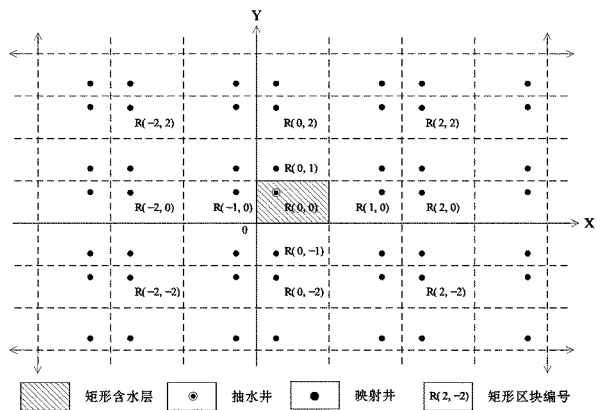


图 1 矩形隔水边界映射图

根据叠加原理, 该无界含水层群井问题的解为:

$$s = \sum_{i=1}^n s_i = \sum_{i=1}^n \frac{Q}{4\pi T} W(u_i) \quad (式 3)$$

* 收稿日期: 2010-01-14; 修订日期: 2010-10-28; 编辑: 曹丽丽

作者简介: 刘志涛 (1982—), 男, 河南巩义人, 工程师, 主要从事水工环地质工作; E-mail: strawer@126.com。

式中： $W(u_i)$ —泰斯井函数

$$u_i = \frac{r_i^2 S}{4Tt} \quad (式 4)$$

式中各字符代号含义同上。

当 $u_i \leq 0.01 \sim 0.05$ 时,根据雅可布公式,其解可近似表示为:

$$s = \sum_{i=1}^n \frac{0.183Q_i}{T} \lg \frac{2.25Tt}{r_i^2 S} \quad (式 5)$$

式中各字符代号含义同上。

3 软件实现思路

程序编写的总体思路为:通过一个循环,分别求出各映射井在计算点的降深 s , 然后进行累加便可得到解。其中所需参数如涌水量 Q 、导水系数 T 、弹性释水系数 S 、开采时间 t 在建立模型之时应该都已确定,映射井与计算点距离 r_i 可根据两点坐标求出。但在具体编写时会发现,映射井点坐标的表示是一个难点,因为映射井并不是均匀分布的,如何用有限的数学表达式表示出映射井点坐标是软件实现的关键。

从图 1 映射后的结果可以看出,在抽水井所在矩形区块周围连续分布着相同的矩形区块,每一区块里各分布着一个映射井。将各矩形块进行编号,假定抽水井坐标为 (a, b) , 通过一些试运算后,会发现其中的规律,矩形块 $R_{i,j}$ 内映射井坐标 (x, y) 可表示为:

$$\begin{cases} x = m_1 \times A + n_1 \times a \\ y = m_2 \times B + n_2 \times b \end{cases} \quad (式 6)$$

$$其中: m_1 = \begin{cases} [\frac{i}{2}] \times 2 (i < 0) \\ [\frac{i+1}{2}] \times 2 (i > 0) \end{cases}$$

$$m_2 = \begin{cases} [\frac{j}{2}] \times 2 (j < 0) \\ [\frac{j+1}{2}] \times 2 (j > 0) \end{cases}$$

$$n_1 = \begin{cases} 1 (i \text{ 为偶数}) \\ -1 (i \text{ 为奇数}) \end{cases} \quad n_2 = \begin{cases} 1 (j \text{ 为偶数}) \\ -1 (j \text{ 为奇数}) \end{cases}$$

($[k]$ 表示对 k 取整)。

这样,程序编写过程中的所有难点基本都已解决,然后就可采用 VC、VB 等各种软件开发工具进行程序设计编写。

软件主界面主要是设置一些输入框用于输入矩形边界大小、水文地质参数等基本计算数据,具体计

算过程中主要是通过一个循环求出所有映射井点坐标,进而求得映射井点与计算点距离,并最终求得问题的解。下面是根据式 6 计算映射井点坐标的源代码(C++)^[4]:

```
int tem_a, tem_b;
int i, j, k;
for(i = -m_well_num; i <= m_well_num; i++)
{
    tem_a = (i < 0) ? (i/2) * 2 : (i+1)/2 * 2;
    tem_b = (i%2) ? -1 : 1;
    p_x[i + m_well_num] = tem_a * m_rect_l + tem_b * m_well_x;
}
for(i = -m_well_num; i <= m_well_num; i++)
{
    tem_a = (i < 0) ? (i/2) * 2 : (i+1)/2 * 2;
    tem_b = (i%2) ? -1 : 1;
    p_y[i + m_well_num] = tem_a * m_rect_b + tem_b * m_well_y;
}
```

其他计算都相对较简单,且由于篇幅关系,这里不再给出源代码。另外,可将计算过程中的一些重要中间数据如各映射井点坐标、各映射井与计算点距离 r_i 、各映射井在计算点产生的水位降深 s 等进行保存,以便最后进行验证查看。

图 2 是使用 VC6.0 做的一个程序界面^[5],该程序主要实现了求解功能,另外也简单实现了一个保存功能,可将各映射井与计算点距离 r_i 、各映射井在计算点产生的水位降深 s 保存在一个 txt 文件中。



图 2 软件界面

4 计算实例

假设矩形隔水边界的长、宽分别为 5 000 m 和 3 700 m,含水层厚度 100 m,其渗透系数为 4.8

m/d,弹性释水系数为 2×10^{-4} ,在坐标(1000,900)处有一井做定流量抽水,抽水量 $1\ 176\ \text{m}^3/\text{d}$,通过软件计算第10年末在坐标(2500,1850)处的降深。取单侧映射井数为5时,很快即可计算出各映射井在计算点的水位降深(表1)。

表1 各映射井降深计算结果(m)

降深	-5	-4	...	2	3	4	5
-5	0.61	0.63	...	0.77	0.69	0.66	0.58
-4	0.62	0.65	...	0.80	0.71	0.68	0.59
-3	0.66	0.69	...	0.92	0.77	0.74	0.62
-2	0.67	0.70	...	0.96	0.79	0.75	0.63
-1	0.69	0.73	...	1.07	0.83	0.78	0.65
0	0.70	0.73	...	1.09	0.83	0.79	0.65
1	0.69	0.72	...	1.04	0.82	0.78	0.64
2	0.68	0.71	...	1.00	0.81	0.77	0.64
3	0.65	0.68	...	0.88	0.75	0.72	0.61
4	0.64	0.66	...	0.84	0.73	0.70	0.60
5	0.59	0.61	...	0.74	0.66	0.64	0.57
合计	100.58						

5 注意问题

(1)式5的适用条件为: $u_i \leq 0.01 \sim 0.05$,实际上这个条件是很容易满足的,但严格来讲,程序中应对 u_i 进行判断,当 u_i 不能满足条件时,按式3进行计算。

(2)映射时应在矩形边界四周对抽水井进行无限次映射,直至映射井在计算点产生的降深可以忽略为止,上述计算实例中单侧映射井数取5仅仅是一个测试,实际上可以看到其最远处映射井在计算点产生的降深还不足以忽略,因此还应继续映射。实际计算过程中,可以根据试算结果人工对其进行

判别。或者在程序中加入相关代码实现自动判别,从而实现全自动求解功能。

(3)当抽水井与计算点重合时,所对应的降深 s 是根据抽水试验确定的,应是一个已知量作为程序设计中的一个参数,而不是根据求解公式计算的,也不是公式所能计算出来的。

6 结语

(1)矩形边界井流问题求解过程繁琐复杂,但通过计算机编程技术可使其变得轻而易举,从而极大地提高工作效率。

(2)矩形边界问题中不仅有隔水边界,也有常水头边界,其组合情况也多种多样,但映射过程都是一样的,只是在有常水头边界的矩形边界模型中,一些映射井应视为注水井。对程序进行一些改动,判断各种情况下映射井是抽水井还是注水井,就可做出功能更加强大更加完善的程序。

参考文献:

- [1] 郭东屏,张石峰. 渗流理论基础[M]. 西安:陕西科学技术出版社,1994:30-65.
- [2] 陈崇希,林敏. 地下水动力学[M]. 武汉:中国地质大学出版社,1999:80-85.
- [3] 郭东屏. 地下水动力学[M]. 西安:陕西科学技术出版社,1994:255-258.
- [4] 谭浩强. C++程序设计[M]. 北京:清华大学出版社,2004:50-70.
- [5] 黄维通,姚瑞霞. VisualC++程序设计教程[M]. 北京:机械工业出版社,2003:208-214.

Implementation Idea of the Software Program for Solving Rectangular Aquifer Well Flow Model

LIU Zhitao, PENG Lei, ZHANG Mingde, YANG Baolin, WANG Zhongmin, LV Jin

(Lubei Geo-engineering Exploration Institute, Shandong Dezhou 253015, China)

Abstract: The process of solving rectangular aquifer well flow model is relatively complicated. Due to large amount of calculation work, errors are easily caused. By using computer programming technology, the process becomes extremely simple and fast, and work efficiency can be improved as well. In this paper, regarding impermeable boundary rectangular aquifer confined well flow model as an example, the implementation idea of software program for solving such problems are introduced.

Key words: Rectangular aquifer; well flow model; computer; programming technology