KGS
OF
81-2

Evaluation of Leaky Aquifer Pumping Test Data:

An Automated Numerical Solution Using Sensitivity Analysis

by

P.M. Cobb, C.D. McElwee, and M.A. Butt

Kansas Geological Survey, 1981

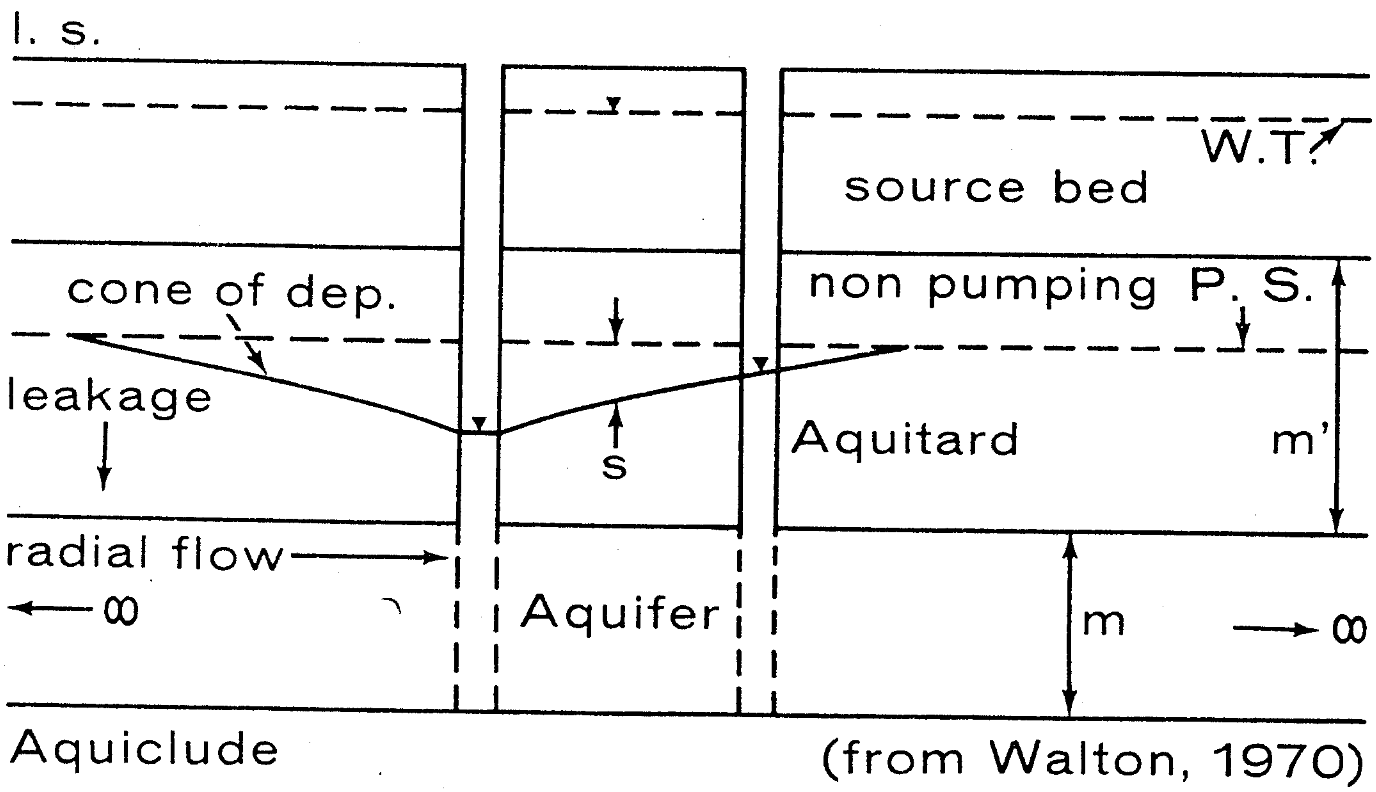


Table of Contents

Cover Plate : Definition of Problem
i Table of Contents
ii List of Figures
iii List of Appendices
1 Abstract
2 Introduction
3 Theory of Analytical Solution to the Leaky Confined Aquifer Problem
5 Numerical Solution of the Leaky Confinedi Aquifer Problem
7 Sensitivity Analysis
11 Discussion of the Leaky Aquifer Sensitivity Coefficients
21 The Least Squares Fitting Procedure
24 Convergence Properties of the Fitting Algorithm
30 Using Program LEAKYFIT
33 Example of Typical Run with LEAKYFIT
37 Program TSSLEAK and Its Use
38 Example of Typical Run with TSSLEAK
44 Program HANTUSH and Its Use
47 Example of Typical Run with HANTUSH
49 Discussion and Summary
50 List of References

List of Figures

12 Figure 1 - Radial Dependence of U_T

14 Figure 2 - Effect of Radius and Transmissivity on the Time Dependence of U_T.

15 Figure 3 - Effects of Changes in S on the Radial Dependence of U_S.

16 Figure 4 - Effects of Radius on the Time Dependence of U_S.

17 Figure 5 - Effect of Changes in L on the Radial Dependence of U_L.

18 Figure 6 - Effect of Changes in L and Radius on the Time Dependence of U_L.

26 Figure 7 - "Best" Fit Parameters for Various Combinations of Over-Estimation and Under-Estimation of Initial Values.

27 Figure 8 - "Best" Fit Leaky Aquifer Parameters Compared with Leaky Type Curve Analysis of Data.

29 Figure 9 - "Best Fit Leaky Aquifer parameters compared with Confined Type Curve Analysis of Data.

46 Figure 10a - Tables of Drawdowns for Interpolated Values of U, r/B, and W(U,r/B).

46 Figure 10b - Tables of Drawdowns for Exact Values of U, r/B, and W(U,r/B).

List of Appendices

52 Appendix I - Program LEAKYFIT.
56 Appendix II - Program TSSLEAK.
61 Appendix III - Program HANTUSH.
63 Appendix IV - FILE WURB, A List of Explicit Functions for Solution of W(U,RB).
67 Appendix V - List of Data Sources.
68 Appendix VI - Comments on Program Notation.

ABSTRACT

The Kansas Geological Survey is pursuing an effort to automate some of the more common aquifer test type curve solutions. This document discusses the results of the work done on the leaky artesian aquifer as defined by Jacob and Hantush (1955). the text covers the basic theory of the aquifer type, the numerical solution of the leaky artesian well functin, W(U,r/B), and the methodology of achieving the "best fit" parameters. In keeping with its attempt to be a user guide as well, listings of all programs developed in this effort and examples of their use are included. Several figures are included which show examples of "best fit" solutions and their corresponding type curve values. These comparisons indicate the generally satisfactory results produced by the regression algorithms documented here. The leaky artesian aquifer drawdown program documented here functions in an acceptable fashion and could serve as the core for an analytical well field simulator capable of handling that type of aquifer.

Introduction

The Kansas Geological Survey is in the process of fabricating a series of computer programs designed to solve the inverse problem of pumptest analysis. The program discussed in this article solves the inverse problem for a leaky artesian aquifer system proposed by Hantush and Jacob (1955). While it is not the most general configuration of the leaky artesian problem (see Hantush, 1960), the limited number of data sets available for analysis tend to be for the simple case. The technique for analysis of the confined problem has been previously published by the Survey (McElwee, 1980a). The methodology to be used involves the formalism of sensitivity analysis and a least squared error fitting technique to fit the time-drawdown data set while satisfying the equations developed by Hantush and Jacob (1955). These techniques will be outlined in the text. More information may be found in McElwee (1980a, 1980b), McElwee and Yukler (1978), and Cobb, McElwee and Butt (1978).

This program is being published after extensive but not exhaustive testing. This is principally due to the limited number of available data sets for this aquifer configuration. However, we have tested it for three synthetic data sets and for seven real data sets readily available to us. At this point, we feel very confident in the algorithm's capabilities. It is hoped that by setting this model out for public scrutiny, new data sets will be tested, and the program more thoroughly verified. Using the available data sets, we have been able to establish that for fairly smooth data sets (those that conform generally to the shape of the leaky type curves), the model has excellent convergence properties. Initial guesses of the storage coefficient, transmissivity, and leakage coefficient may be in the range of plus or minus three orders of magnitude of the correct value while still obtaining successful convengence.

This method of solution of the inverse problem does not remove the requirement of having an experienced hydrologist evaluate the pumptest data for general features. However, once the decision is made as to what aquifer configuration is being observed, this program will, in a quick and unbiased fashion, give an accurate assessment of the leaky aquifer parameters within the limits of the theoretical approximations. After using this model for the pumptest analysis, the hydrologist should always look at the root-mean-square (rms) deviation in drawdown and the "best fit" drawdowns calculated by the program. The experimental and theoretical drawdowns should not differ greatly anywhere and the rms deviation should be less than a few tenths of a foot in order to have confidence in the analysis. If this is not the case, one is probably not dealing with a simple leaky aquifer.

Theory and Analytical Solution to the Leaky Confined Aquifer Problem

The aquifer system defined by Hantush and Jacob (1955), as depicted on the cover, is composed of a level, isotropic, homogeneous, porous medium of infinite areal extent. The lower aquifer boundary is assumed to be impervious, while the upper boundary is assumed to be a leaky confining bed. A source bed overlies the leaky confining bed. Water is derived from the aquifer by elastic expansion of the water and compression of the aquifer matrix as pumping occurs. Leakage through the semiconfining bed is assumed to be proportional to the drawdown in the semiconfined aquifer. No water is removed from storage in the semiconfining unit and no drawdown occurs in the source bed.

These assumptions lead to the following differential equation (Jacob, 1946)

$$\frac{\partial^2 s}{\partial r^2} + \frac{1}{r}\frac{\partial s}{\partial r} - \frac{s}{B^2} = \left(\frac{S}{T}\right)\frac{\partial s}{\partial t} \qquad (1)$$

, where

$s(r,t)$ is the drawdown at any distance from the well at any time.

r is the radial distance measured from the well

S is the storage coefficient of the artesian aquifer

K and K' are the respective permeabilities of the artesian aquifer and the semiconfining bed

b and b' are the respective thicknesses of the artesian aquifer and the semiconfining bed

$T = Kb$ is the transmissivity of the artesian aquifer

K'/b' is the leakance of specific leakage of the semiconfining bed (Hantush, 1949)

$B^2 = T/(K'/b')$

Q is the well discharge

With appropriate boundary conditions, an analytical solution is obtainable.

Hantush and Jacob give several solutions to equation (1) for different ranges of u and r/b. In this program three equations are solved numerically in order to cover the broadest possible range of u and r/B. The equations are listed here, along with the appropriate ranges of u and r/B.

$$s = Q/4\pi T \bullet \int_u^{\infty} \exp(-y-a)\,dy/y \qquad (2)$$

$u = r^2S/4Tt$, $z = r^2/4B^2y$, $u > 1.0$, any value of r/B

3) $$s = Q/4\pi \cdot \left[2K_o(r/B) - \int_p^{\infty} \exp(-y-z)\,dy/y\right]$$

$$p = Tt/SB^2 \quad , \quad r^2/B^2 > u < 1.0$$

4) $$s = Q/4\pi \cdot \{2K_o(r/B) - I_o(r/B) \cdot -Ei(-r^2/4B^2u) + \exp(-r^2/rB^2u)\,[0.5772 + \ln u + Ei(-u) - u + u\,(I_o(r/B)-1)/(r^2/4B^2)$$

$$- u^2 \sum_{n=1}^{\infty} \sum_{m=1}^{\infty} \frac{(-1)^{n+m}(n-m+1)!}{((n+2)!)^2} (r^2/4B^2)^m u^{n-m}]\}$$

$$r/B < u < 1$$

where Ei(x) is the exponential integral, I_o and K_o are the modified Bessel functions of the first and second kind, zero order. The numerical computational routines involving these functions were checked by generating the table published in Walton (1970), page 146. This table could be produced accurately to the fourth decimal place.

Numerical Solution of the Leaky Confined Aquifer Problem

Integral functions of the form

$$\int_0^{\infty} f(x)e^{-x}dx$$

may be approximated by the method of Laguerre integration:

$$\int_0^{\infty} f(x)e^{-x}dx \approx \sum_{i=1}^{n} w_i f(x_i)$$

where the w_i's are weighting factors and the x_i's are the abscissas and correspond to the zeros of Laguerre polynomials. The method of solution and values of w_i and x_i are catalogued in Abramowitz and Stegun (1968).

In order to perform the integrations in equations 2 and 3, a transformation of variables must occur in order to make the limits of integration compatible with the Laguerre Polynomial method. This transformation is a straight-forward substitution of the form y=x+n, where n takes the value of u for equation 2 and p for equation 3. The integrals take the form

$$G(r/B,n) = \int_0^\infty \exp\{-n-r^2/[4B^2(x+n)]\}/(x+n) \bullet \exp(-x) \bullet dx$$

and are solved numerically by the appropriate substitutions in the Laguerre integration formula.

The function G(r/B,n) was evaluated by Laguerre integration of order 15. It was able to successfully duplicate the table of W(u,r/B) published in Walton (1970). We considered the possibility that G(r/B,n) might not be accurately approximated for all possible values of r/B and n. The roots of the Laguerre polynomial should sample the function to be integrated properly for desired accuracy. A scaling transformation was incorporated to change the range of abscissas over which the evaluation occured. The transformation was of the form x=ay such that:

$$f(y) = f(x/a)$$

and

$$\int_0^\infty h(x) \bullet \exp(-x) \bullet dx/a \approx \frac{1}{a} \sum_i w_i h(x_i) \quad ,$$

where $h(x) = f(x/a)\exp(1/a)$.

Results of numerical experiments indicate that the value of the scaling transformation ranging from 1.0 to 10.0 has no effect upon the integral when it is used simply to solve the drawdown equation. However, in the regression algorithm, a slight economy in number of iterations accrues when the scaling transformation is set equal to 5.0 (parameter AA in the function list, Appendix IV). This is due to improvement in the evaluation of the higher order decimal places.

The numerical solution of the exponential integral, Ei(x), is described in detail in McElwee (1980, p. 3). Solutions for the modified Bessels functions of the first and second kinds, zero order, $I_o(x)$ and $K_o(x)$, were carried out in the manner of polynomial approximations. Abramowitz and Stegun (1968) catalog several forms for each function. Each form is suitable for a particular range of x. The solutions appear in the function list (Appendix IV) and are titled Function AIO(RB) and AKO(RB).

The double summation in equation (4) is solved numerically by Function SUM(U,RB) and Function IFACT(L) (see Appendix IV). A truncated summation performed, since only a finite number of terms are required to approximate a convergent function. Numerical experiments showed that n=5 (LIMIT=5 in SUM(U,RB)) yields a very stable value for the summation.

Sensitivity Analysis

Parametric sensitivity analysis is a method of examining the stability of a mathematical representation of a dynamic system with respect to variations in the values of the system's physical parameters. The theoretical basis of this technique is outlined by Tomovic (1962), while the application to hydrologic problems has been examined by Vemuri, et al, 1969; McCuen, 1973; and Yukler, 1976.

In formulating the sensitivity analysis of the leaky confined aquifer problem, the following mathematical model is useful:

$$F(h_{xx}, h_{yy}, h_t;\ S, T, L, Q) = 0$$

where $h_{xx} = \frac{\partial^2 h}{\partial x^2}$, $h_{yy} = \frac{\partial^2 h}{\partial y^2}$, $h_t = \frac{\partial h}{\partial t}$

h = hydraulic head

S = storage coefficient

T = transmissivity

L = inverse leakage coefficient (L = 1/B)

Q = pumpage

and whose solution may be written as $h=h(x,y,t;S,T,L,Q)$. Variations of any single parameter such as T produces a new formulation

$$F(h^*_{xx}, h^*_{yy}, h^*_t; S, T+\Delta T, L, Q) = 0$$

where ΔT is the incremental change in T and h* is the perturbed head. The solution to this expression is of the form $h^*=h^*(x,y,t;S,T+\Delta T,L,Q)$. The stability of the system to small changes in the parameter T may be expressed by

$$\frac{\Delta h}{\Delta T} = \frac{h^*-h}{\Delta T}$$

If the limit to this expression exists as ΔT approaches zero, it may be written as

$$U_T(x,y,t;S,T,L,Q) = \frac{\partial h}{\partial T} = \lim_{\Delta T \to 0} \frac{\Delta h}{\Delta S}$$

and

$$U_L(x,y,t;S,T,L,Q) = \frac{\partial h}{\partial L} = \lim_{\Delta L \to 0} \frac{\Delta h}{\Delta L}$$

which are respectively the sensitivity coefficient with respect to changes in S and the sensitivity coefficieint with respect to changes in L.

The solution to the flow equation is assumed to depend analytically upon the parameters S, T, and L; and that S, T, L, and Q are independent of each other. Because of these assumptions, the function, $h^*(x,y,t;S,T+\Delta T,L,Q)$, which is perturbed in the parameter T, may be expanded in a Taylor's series (Tomovic, 1962), and if ΔT is small, all non-linear terms can be neglected

$$h^*(x,y,t;S,T+\Delta T,L,Q) = h(x,y,t;S,T,L,Q) + U_T \Delta T$$

where $U_T = \frac{\partial h}{\partial T}$. Thus, new hydraulic heads, resulting from incremental changes in T, can be computed directly if the unperturbed head is known and U_T can be computed. Similar expressions may be derived for perturbation with respect to S and L

$$h^*(x,y,t;S+\Delta S,T,L,Q) = h(x,y,t;S,T,L,Q) + U_S \Delta S$$

$$h^*(x,y,t;S,T,L+\Delta L,Q) = h(x,y,t;S,T,L,Q) + U_L \Delta L$$

where both are correct to first order in ΔS and ΔL respectively.

In order for this technique to be useful, it is only necessary to be able to compute U_S, U_T and U_L, since $h(x,y,t;S,T,L,Q)$ may be computed by previously discussed techniques. This requirement may be satisfied by analytical or numerical techniques. In this work, it was found to be convenient to obtain U_S and U_T by direct analytical means and U_L by a numerical method.

Recall that the basic equation describing the solution to the leaky confined aquifer is

5) $$s = \frac{Q}{4\pi T} \int_u^\infty \frac{1}{y} \exp(-y - \frac{L^2 r^2}{4y})\, dy, \quad u = \frac{r^2 S}{4Tt}, \quad L = B^{-1}$$

By applying Leibnitz's rule for differentiating an integral (Hildebrand, 1962) it is easy to obtain the sensitivity coefficients with respect to S and T:

$$U_S = \frac{\partial s}{\partial S} = -\frac{Qr^2}{4\pi T^2 t} \left(\frac{1}{u} \exp(-u - \frac{L^2 r^2}{4u})\right)$$

$$U_T = \frac{\partial s}{\partial T} = -\frac{Q}{4\pi T^2} \int_u^\infty \frac{1}{y} \exp(-y - \frac{L^2 r^2}{4y}) dy + \frac{Qr^2 S}{16\pi T^3 t} \left(\frac{1}{u} \exp(-u - \frac{L^2 r^2}{4u})\right)$$
$$= -\frac{S}{T} + \ldots.$$

These equations may be easily evaluated by standard numerical techniques on a high speed computer once $\underset{\sim}{S}$ is known.

The reasoning for computing U_L by a direct numerical technique, rather than by formulating an analytical solution, was based on a desire to conserve program simplicity while retaining computational accuracy. Note that the argument of the exponential within the integral of equation (5) contains the parameter L. Hence, differentiation would transform the entire function within the integral and would define

$$U_L = \int_u^\infty \frac{\partial}{\partial L} \{\exp(-y - L^2r^2/4y)/y\} dy$$

$$= \int_u^\infty \{-Lr^2/2y^2\} ex(-y - L^2r^2/4y) dy$$

Note that both U_S and U_T can be expressed in such a manner that after the drawdown s is computed, no further numerical integration is required. The sensitivity with respect at leakage, U_L, however can only be computed by numerical integration which would involve the formulation of a more complex function of Function SS(U,RB) (see Appendix IV). Since a numerical integration of W(u,r/B) was already appearable (see Function W(u,RB), Appendix IV), the decision was made to generate U_L by finite difference approximation. The approximation

$$\partial s/\partial L \approx \{ s(L+\Delta L) - s(L-\Delta L)\} /2\Delta L$$

where

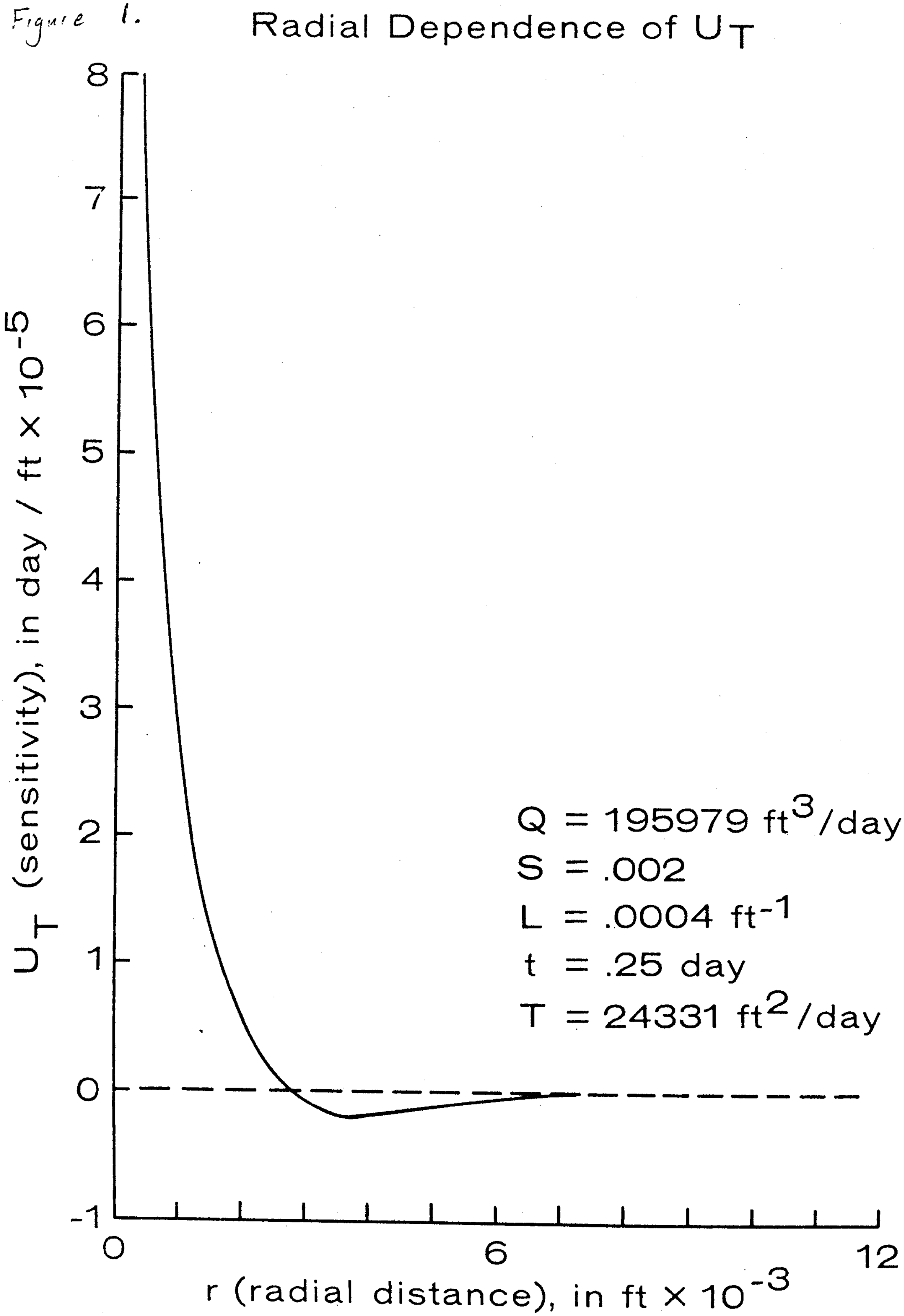
$$s(L\pm\Delta L) = Q/4\pi T \int_u^\infty \exp\{-y - r^2 (L\pm\Delta L)^2 /4y\} /y \bullet dy$$

becomes increasingly accurate as ΔL approaches zero. Satisfactory evaluation of U_L occured for ΔL set equal to .01 L. The methodology for computing the sensitivity coefficients is now complete.

Discussion of the Leaky Aquifer Sensitivity Coefficients

The radial dependence of U_T is shown in Figure 1 . The function

Figure 1.



Radial Dependence of U_T
8
7
6
5
4
3
2
1
0
-1
0
6
12
U_T (sensitivity), in day / ft × 10^-5
r (radial distance), in ft × 10^-3
Q = 195979 ft^3/day
S = .002
L = .0004 ft^-1
t = .25 day
T = 24331 ft^2/day

diverges logarithmically near the well. U_T changes sign at some finite value of radius. This demonstrates the fact that when T is changed, the cone of depression deepens in some areas and shallows in others.

Figure 2 depicts the time dependence of positive values of U_T on variations in r and T. Note that U_T is inversely proportional to T. The curves represent a transmissivity of 24331 ft^2/day and ±20% of that value at a radius of 100 feet and for a T of 24331 ft^2/day at a radius of 1000 feet. Note that all curves flatten after 3 to 4 days. This describes the steady condition caused by deriving the discharge Q totally from leakage.

The radial dependence of U_S is shown in Figure 3 . This coefficient does not diverge at the well, nor does its sign change. It is inversely proportional to S. The constancy of algebraic sign indicates that as S changes there is a general raising and lowering of the cone of depression.

The time dependence of U_S is presented in Figure 4 . Radial variation is represented by the presence of 3 curves. Each curve reaches its maximum value for U_S at a time directly proportional to its radial value. At some finite value of time each curve approaches zero in value, indicating that a steady state is achieved. The differing nature of the curves is related to the fact that until steady state is attained, there is a dual source supplying the pumpage, namely water released from storage and leakage. The curves role over as leakage starts to dominate the source mechanism. U_S is zero outside the cone of depression and at any time t after steady state is attained.

Figure 5 shows the radial dependence of U_L. The sensitivity coefficient U_L does not diverge at the well and approaches zero for large values of r.

The time dependence of U_L is shown in Figure 6 for two values of r. all curves grow with time until a steady state is achieved where leakage is

Figure 2.

Effect of Radius and Transmissivity on the Time Dependence of U_T

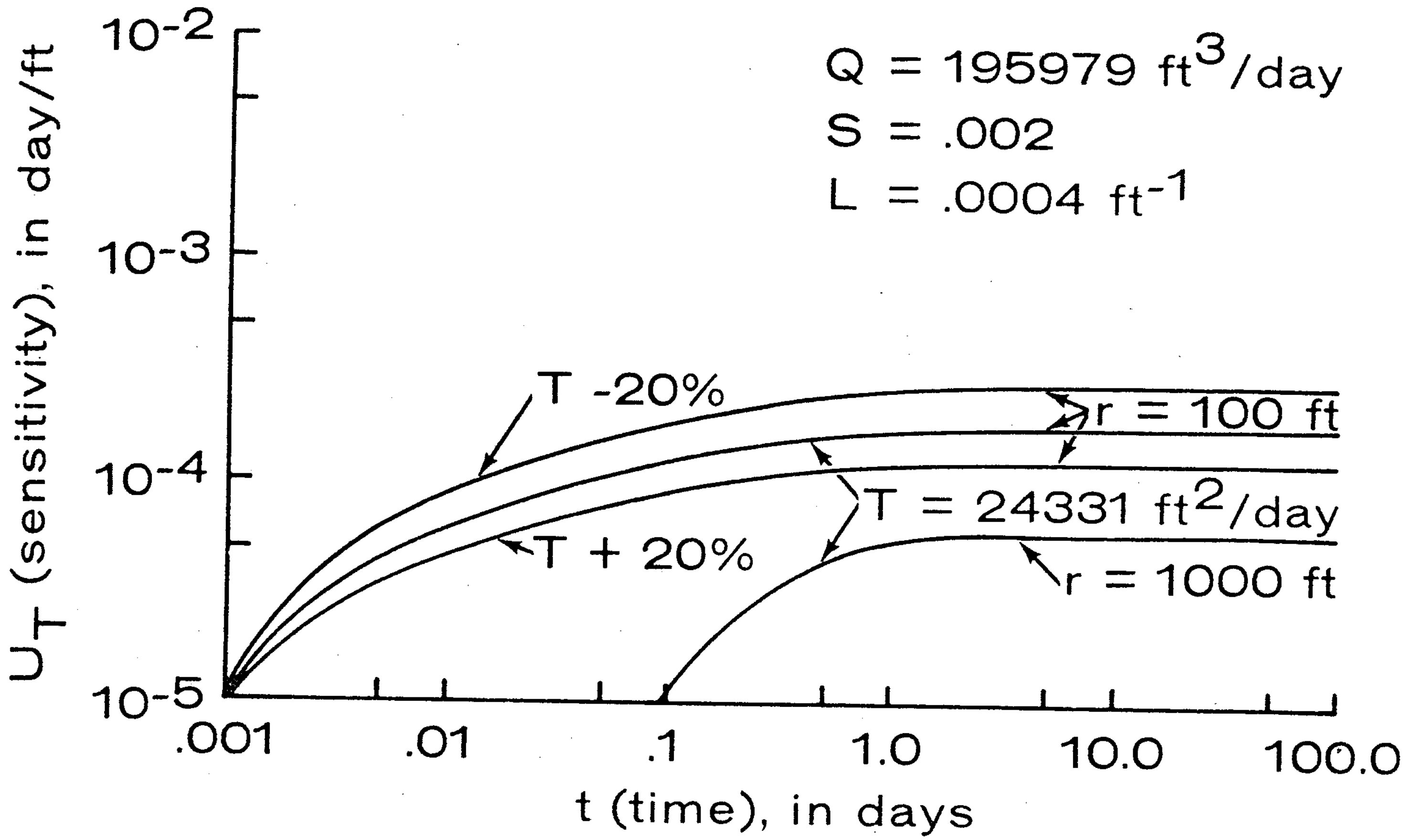


Figure 3.

Effects of Changes in S on the Radial Dependence of U_S

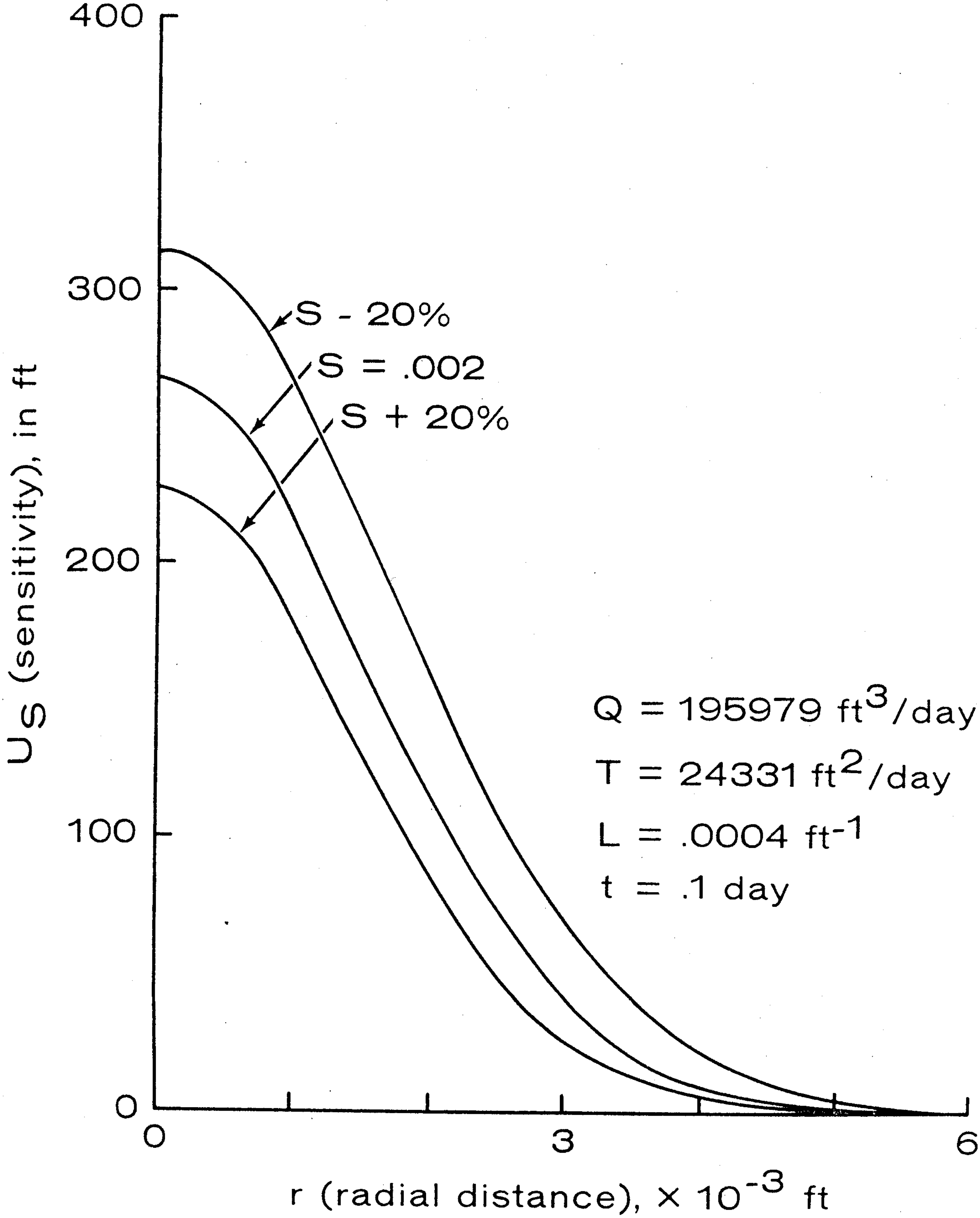
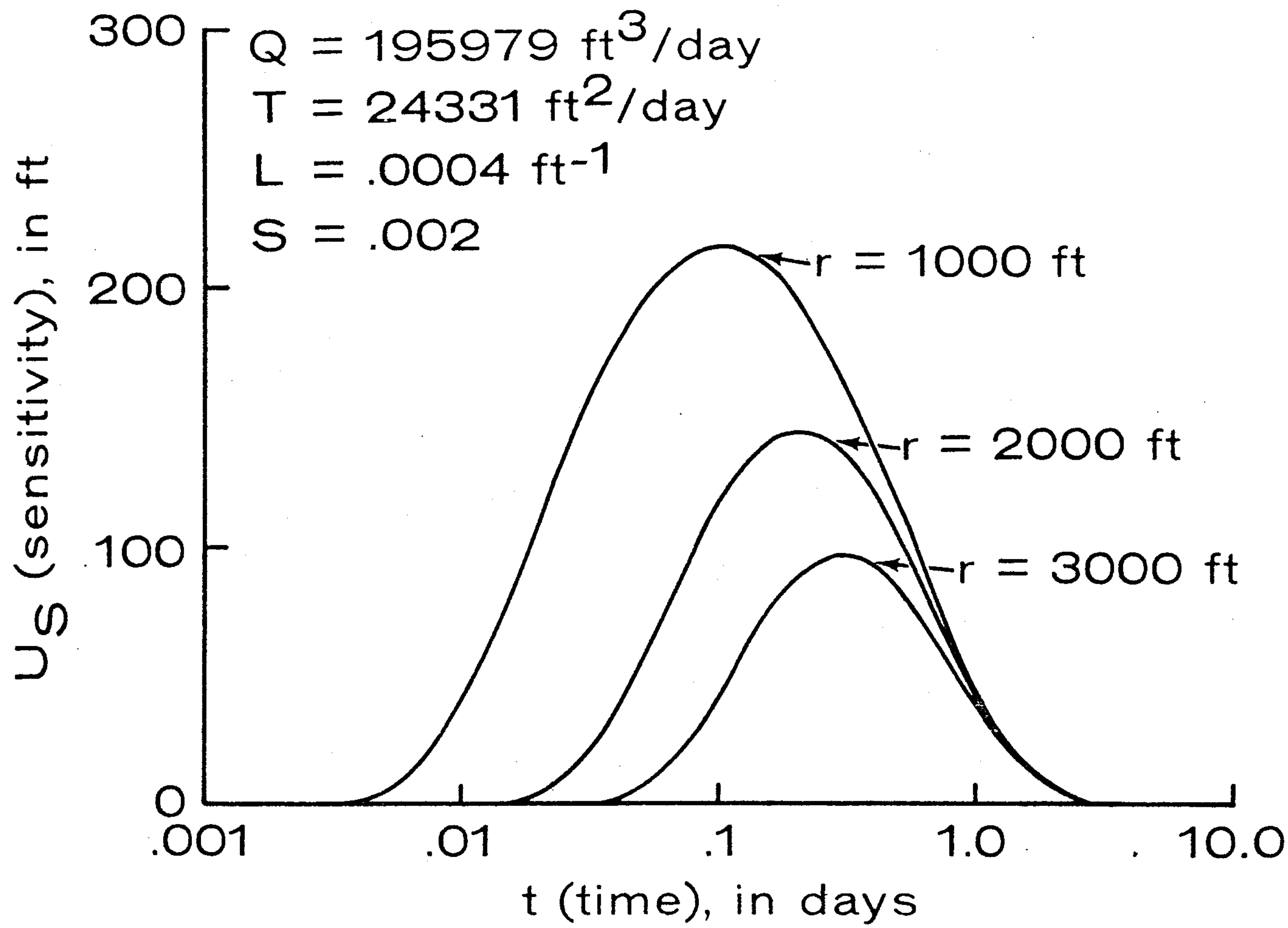


Figure 4

Effects of Radius on the Time Dependence of U_S



10

Figure 5. Effect of Changes in L on the Radial Dependence of U_L

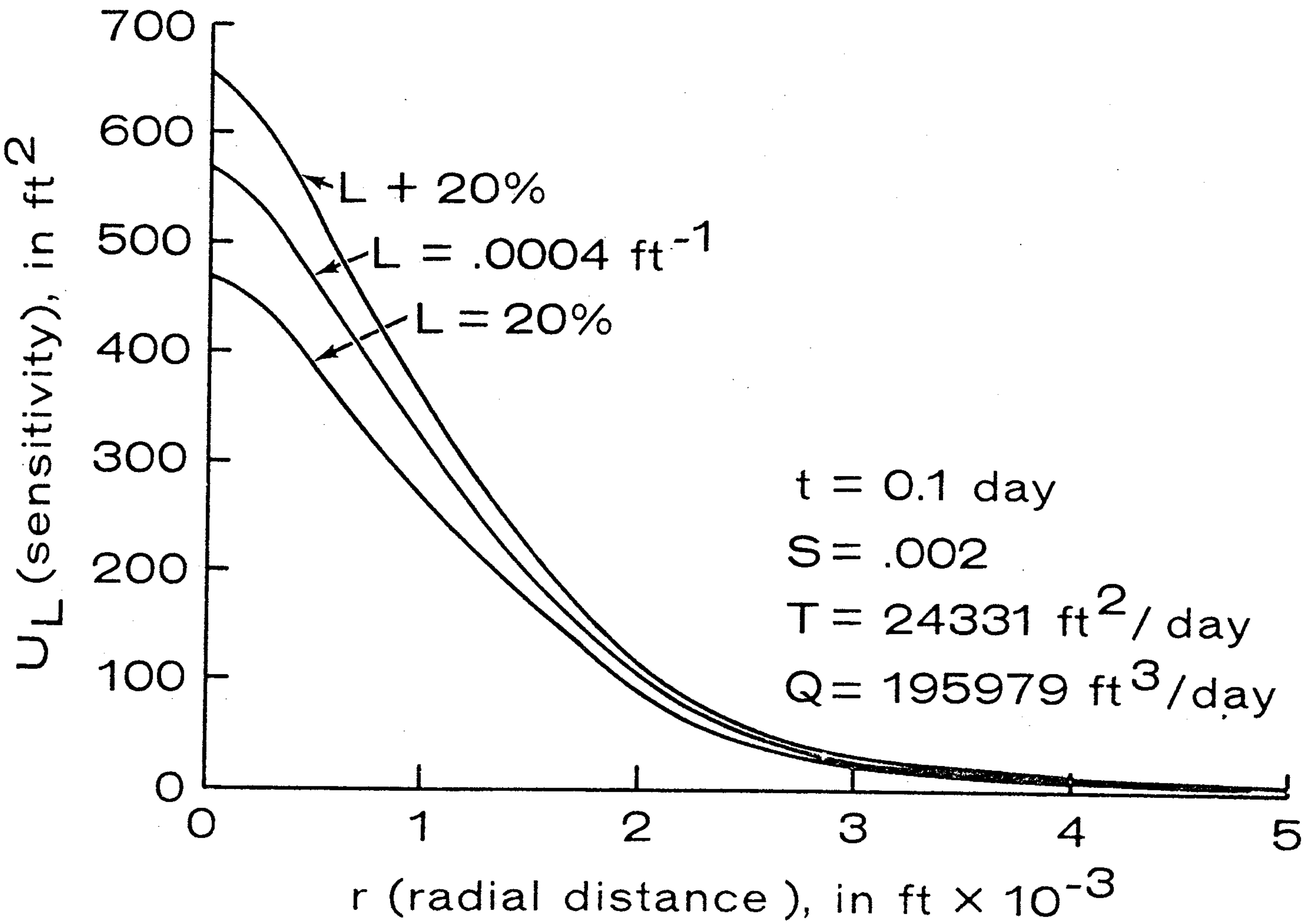
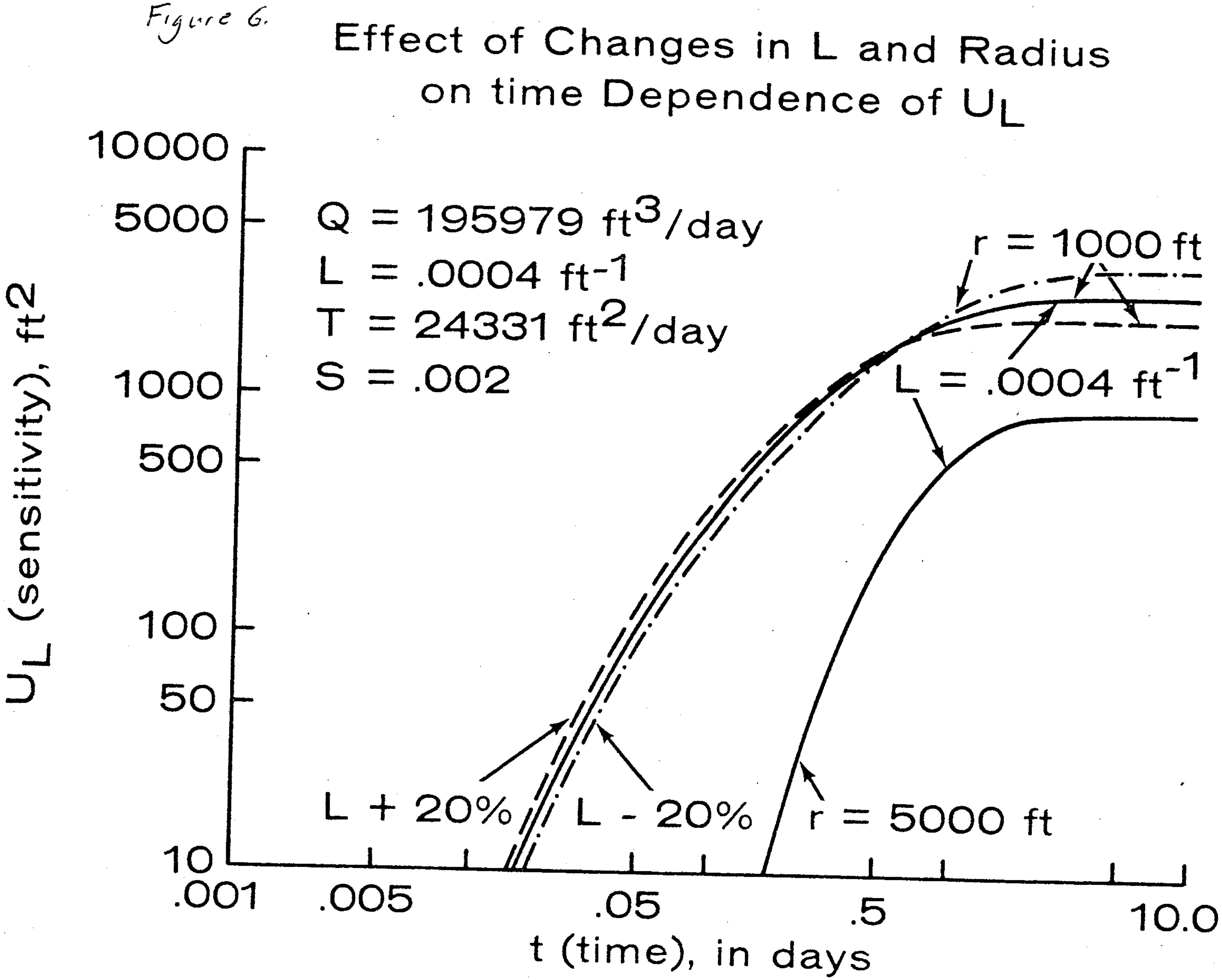


Figure 6.


Effect of Changes in L and Radius
on time Dependence of U_L
$Q = 195979$ ft^3/day
$L = .0004$ ft^{-1}
$T = 24331$ ft^2/day
$S = .002$
r = 1000 ft
L = .0004 ft^{-1}
L + 20%
L - 20%
r = 5000 ft
U_L (sensitivity), ft^2
10000
5000
1000
500
100
50
10
.001
.005
.05
.5
10.0
t (time), in days

supplying the entire discharge Q. The set of curves labled $L = .0004 \text{ ft}^{-1}$ and $\pm 20\%$ of that value are of interest. Observe that at t less than .6 days U_L is directly proportional to L, while for t greater than .6 days, U_L is inversely proportional to L. As indicated before, Q is supplied by a dual source in the leaky artesian aquifer: water taken from storage in the aquifer and water supplied by leakage through the aquitard. This dual source mechanism results in the changeing dependence on L.

In order to better understand the behavior of the sensitivity coefficient U_L, it is necessary to examine the phenomenon in a little closer detail. The volume of water derived by a pumping well in a leaky artesian aquifer before a steady state is achieved, as stated previously, is due both to water taken from storage by elastic expansion and leakage through a semi-confining bed from a source bed. Both effects depend upon the drawdown in the confined aquifer.

Equation (1) is the equation governing the flow in the leaky confined system and its solution yields the drawdown in the semi-confined aquifer in terms of r and t, s = s(r,t). Recasting the flow equation into a more general form

$$T \nabla^2 s = S \frac{\partial s}{\partial t} \pm Q^*$$

where Q* is a source or sink per unit area, it can be argued in a heuristic fashion that the total volume of water removed from the leaky confined aquifer at any time t may be repressented as

6) $$Qt = SA\overline{s} + CAL^2\overline{s} \quad ,$$

where Q is volume per unit time, A is the area of the plan view of the cone of depression at time t, L^2 is B^{-2} and C is equal to Tt. The quantity $\bar{s}$ is an average drawdown which satisfies the equation.

$$V(t) = \int_A s(t)dA = \bar{s}A$$

The quantity s is guaranteed to exist by the mean value theorem which states that for suitably continuous functions given over a closed bounded region R

$$\iint_R f(x,y)dxdy = f(x_o,y_o)A$$

where A is the area of R (Kreysig, 1972).

Equation (6) may be recast and, taking the partial derivative with respect to L, yields

$$7) \qquad \partial\bar{s}/\partial L = -2CtQL/A(S+CL^2)^2$$

At early times storage contributions greatly exceed contributions by leakage, so that $s > CL^2$, implying that

$$8) \qquad U_L = \partial\bar{s}/\partial L \simeq -2CtQL/AS^2$$

This indicates that U_L is directly proportional to L, in agreement with Figure 6 . For large values of t, leakage becomes the dominant source so that $s < CL^2$. Equation (7) now becomes

$$U_L = \partial\bar{s}/\partial L \simeq -2tQ/ACL^3 \simeq -2\bar{s}/L$$

indicating that U_L is indeed inversely proportional L for large values of time.

The Least Squares Fitting Procedure

The objective of any curve fitting technique, whether performed manually or by machine, is to fit a theoretical type curve to an experimental data set to as high a degree as possible, evaluating in the process a corresponding set of physical parameters. In order to perform this task successfully, a mechanism is required for judging the error in the fit. Classical manual lcurve fitting relies basically on the best "eye ball" fit. The machine method described here allows the error in fitting to be accurately and meaningfully determined as the rms error.

In order to apply the parametric senstivity method to the fitting problem, it is necessary to define the squared error function

$$E = \sum_i [S_e(t_i) - S^*(t_i)]^2$$

where E is the summation over i discrete samples, of the squared difference between the experimental drawdown S_e and the updated drawdown S* which is computed from the truncated Taylor's Series

$$S^* = S + U_T \Delta T + U_S \Delta S + U_L \Delta L$$

The arguement t_i represents the i^{th} value of time. Expansion of the squared error function, taking partial derivatives with respect to the perturbed parameters, and setting the partial derivatives equal to zero, yields a set of three simultaneous linear equations which must be satisfied to obtain the best fit. More specifically, for minimizing E, it is required that

$$\frac{\partial E}{\partial \Delta T} = \frac{\partial E}{\partial \Delta S} = \frac{\partial E}{\partial \Delta L} = 0$$

The linear system of equations which results is

$$\begin{bmatrix} \sum_i U^2_L & \sum_i U_L U_S & \sum_i U_L U_T \\ \sum_i U_S U_L & \sum_i U_S^2 & \sum_i U_S U_T \\ \sum_i U_T U_L & \sum_i U_T U_S & \sum_i U_T^2 \end{bmatrix} \begin{bmatrix} \Delta L \\ \Delta S \\ \Delta T \end{bmatrix} = \begin{bmatrix} \sum_i U_L (S-S_e) \\ \sum_i U_S (S-S_e) \\ \sum_i U_T (S-S_e) \end{bmatrix}$$

and can be solved explicitly for ΔL, ΔS, and ΔT. The term S is the theoretical drawdown at time t calculated from the previously updated values of L, S, and T. The new values of the parameters are simply

$$L_{i+1} = L_i + \Delta L_i$$

$$S_{i+1} = S_i + \Delta S_i$$

$$T_{i+1} = T_i + \Delta T_i$$

This process continues until the values of ΔL_i, ΔS_i, and ΔT_i simultaneously satisfy a specified convergence criteria. The goodness of fit obtained at the termination of the last iteration is indicated by the value of the root mean squared error

$$\frac{\sum_i (S - S_e)^2}{n}$$

where n is the number of discrete samples of S.

The success of this methodology is dependent to a degree upon the initial guesses of the parameters S, T, and L. However, numerical experiments conducted with the most recent version of the program indicate that the initial guesses may be as much as three orders of magnitude above or blow the converged solution values and still obtain convergence.

In order to maintain conformitiy to physical reality and improve numerical stabiity, the algorithm requires that the parameters S, T, and L must always be positive. Furthermore, the increments T, S, and L are never allowed to exceed 0.5 or be less than -0.2. This subterfuge insures that the algorithm executes in a convergent fashion to the local and possibly global minimum. ~~is obtained efficiently.~~

Convergence Properties of the Fitting Algorithm

In order to achieve a converged solution, the typical regression model involving sensitivity analysis requires initial estimates of the aquifer parameters. If these estimates vary greatly from the actual values, convergence may not occur. A desirable property of this type of algorithm would be to have a large "window" in which estimates can be made and achieve convergence in a reasonable number of iterations.

The algorithm presented in this discussion (see ~~Appendix~~ Appendices I, II, IV) has consistently proven its ability to converge to the correct set of aquifer parameters for a given data set. These values are comparable to values of the parameters obrtained by curve matching methods. These "best" fit values are achieved over a range of initial estimates ranging from 3 orders or magnitude above to 3 orders of magnitude below the converged values (see Fig. 7). The sources for this data is tabulated in Appendix V . Typical data sets, with initial estiamtes in this range, converge in less than 40 iterations. convergence tends to occur more rapidly for underestimated parameter sets. Iterations are reduced as the estimated parameter values approach the true values. For typical data sets the rms error tends to be only a few tenths of a foot, while for fairly idealized sets of data, the rms error is a few hundredths of a foot (see ~~Table~~ Figure 8). Iterations can be reduced by increasing the size of the acceptable error criteria, but only at the cost of increased rms error.

The parameter AA, which is a scaling factor in the numerical integration routine, has a small effect on rate of convergence. The statistics quoted above are for AA = 1.0. Numerical experiments indicate that a small economy in iteration steps is available by setting AA equal to 5.0.

We have extensively tested the algorithm on some synthetic data formulated by Cooper (1963) for an infinite leaky confined aquifer. Some typical results appear in Figure 7 . Note that for all permutations of 3 orders of magnitude above and below the correct values, the regression results are virtually identical, showing that the algorithm's convergence properties are not parameter specific. The rms error for this data is shown in Figure 8 (data set 2a) to be .038 feet.

If the data diverges to much from ideal data, convergence may not occur. If convergence does occur the rms error may be unacceptable. Although this algorithm gives a unique solution to any data set for which it can achieve a converged set of values, it cannot distinguish absolutely between different types of aquifers. Since the three analytical degrees of freedom give the algorithm considerable latitude in achieving convergence, an imperfect data set from a confined aquifer may be successfully run and a set of values for transmissivity, storage, and leakage produced. This fact points to several cautions. First, only the best data available should be anlayzed. Second, the geohydrology should be carefully examined by experirenced personnel to aid in classifying the aquifer type. Thirdly, if doubt exists about the validity of the converged values, the rms error value should be noted, but the individual "best fit" drawdowns should be compared to the field data for gross divergences. The following discussin will aid in understanding these points.

Figure 7.
"Best" Fit Parameters for Various Combinations of Over-Estimation and Under-Estimation of Initial Values.

Values from Cooper (1963)
Type Curve Values: SC = .0001 KB = 13,300 ft/day LC = .0005 ft^{-1}

Initial Guess Values			Converged Values r = 100 ft		
Sc	KB (ft/day)	Lc (1/ft)	Sc	KB (ft/day)	Lc (1/ft)
1×10^{-7}	13.3	4.98×10^{-7}	9.789×10^{-5}	13338.	4.9401×10^{-4}
1×10^{-1}	13.3×10^{6}	4.98×10^{-1}	9.789×10^{-5}	13338.	4.9404×10^{-4}
1×10^{-7}	13.3	4.98×10^{-1}	9.789×10^{-5}	13338.	4.9404×10^{-4}
1×10^{-7}	13.3×10^{6}	4.98×10^{-1}	9.789×10^{-5}	13338.	4.9404×10^{-4}
1×10^{-7}	13.3×10^{6}	4.98×10^{-7}	9.789×10^{-5}	13338.	4.9403×10^{-4}
1×10^{-1}	13.3	4.98×10^{-7}	9.789×10^{-5}	13338.	4.9402×10^{-4}
1×10^{-1}	13.3	4.98×10^{-1}	9.789×10^{-5}	13338.	4.9403×10^{-4}
1×10^{-1}	13.3×10^{6}	4.98×10^{-7}	9.789×10^{-5}	13338.	4.9403×10^{-4}

Figure 8.

"Best" Fit Leaky Aquifer Parameters Compared with Leaky Type Curve Analysis of Data

Data Source Code		Type Curve* Values		Converged Values for Initial Guesses 10^{+3} and 10^{-3} Type Curve Values	rms† error
1		T = 182000 gpd/ft S = .002 B = 2500 ft		202000 gpd.ft .002 3300 ft	.007 ft
2	a	T = 99400 S = .001 B = 20000	a	99000 .00097 20000	.038
	b		b	100000 .00097 19800	.016
	c		c	97800 .001 19500	.010
3		T = 1500 S = .00020 B = 430		1800 .00017 650	.125
4	a	T = 49000 S = .000090 B = 4100		44000 .000086 3900	.378
	b	T = 41000 S = .000080 B = 4000		46000 .000084 4800	.030

*T = Transmissivity
S = Storage coefficient
B = Leakage coefficient

†rms = root mean square

The information tabulated in Figure 8 is "best fit" values for data sets analyzed by leaky artesian type curve fitting. The sources are listed by number in Appendix V . The lower case letters indicate that data was taken for different observation wells at the same pumptest, or that several independent pumping tests were listed in the same source. The principle feature of this figure is the quite good agreement between the "best fit" algorithm values and the type curve values. All values are well within the same order ot magnitude of each other and in fact the differences are not over 35%; most of them being in the 10-20% range. The largest rms error is about .4 feet for set 4a. The smallest rms error is .007 feet for data set 1. Note that data sets with the lowest rms error do not necessarily have the closest agreement between sets of parameters. This feature is related to the sensitivities of the various parameters.

Figure 9 is a comparison of parameter values derived from data sets evaluated by the confined artesian type curve method and the leaky artesian regression algorithm. Although the rms errors are satisfactory, there are discrepancies of several orders of magnitude in the storage values. This is especially true for example 6a,b. Data set 6b has two sets of S and T produced from the same set and the "best fit" S and T are the approximate averages of these values. These examples demonstrate the fact that non-perfect data can still achieve convergence in this algorithm. This points to the need to examine the geology of a site as well as the drawdown curve from an aquifer test. It also illustrates the ~~impression~~ inexactness of real data, versus theoretical curves.

Figure 9.

"Best" Fit Leaky Aquifer Parameters Compared with Confined Type Curve Analysis of Data

Data Source Code		Confined Aquifer Type Curve Values	Leaky Aquifer Values Obtained From Regression Fit	rms error
5		T = 44000 gpd/ft S = .00046 B = 0 ft	T = 42000 S = .00044 B = 8600	0.240
6	a	T = 42000 S = .000004 B = 0	T = 9800 S = .0045 B = 65	.036
	b	T_1 = 48000, T_2 = 19500 S_1 = .0000065, S_2 = .002 B = 0	T = 25500 S = .00055 B = 1180	.032

Using Program LEAKYFIT

LEAKYFIT is an algorithm which is run in batch mode. Data may be entered by cards or from a permfile when initiating the job from a remote iintreactive terminal. These files are in free field format, where the data are separated by blanks or commas. In general, the algorithm is not in machine dependent code. Only one statement (see Appendix I), CALL FXOPT (...), is specific to the Honeywell 66/60 now operating at Kansas University. This statement can be removed by a user implementing this algorithm on a "foreign" machine. The purpose of this statement is to circumvent program termination caused by exponential overflows, exponential underflows, and division checks.

There are some pre-assigned parameter values in the algorithm which the user may need to redefine from time to time. The Parameter value ND controls the size of arrays SE, T, and SGS, which store the discrete valuesof experimental drawdown, time, and computed drawdown respectively. The value of ND must always be equal to the number of time-drawdown pairs read into the algorithm. if this restriction is not adhered to the standard deviation will not be correctly computed and problems will arise in reading data. The parameter ITMAX controls the maximum number of iterations performed by the program. It is set at 50, since our experiments show that all the data sets which have ben tested converge in less than 50 iterations. However, if convergence does not occur, this parameter may be increased. The value of ERROR check for convergence of the parameter updating quantities ΔU_T, ΔU_S, and ΔU_L. This value gives good fits with acceptable numbers of iterations. Changing this parameter is not recommended. making it larger reduces number of iterations while decreasing the accuracy of the fit. Reducing its value has the opposite effect.

Data input is straight forward. The first free field format entry consists of the parameters

SC = Elastic storage coefficient of confined aquifer

KB = Transmissivity of confined aquifer

LC = Inverse leakage coefficient of the confining bed

R = Radial distance from the pumping well to the observation well

Q = pumping rate

TCL = Thickness of the confining bed if known

These parameters may have any consistent set of units associated with them. If TCL is known, the program can compute K', the vertical hydarulic conductivity of the confining bed. If TCL is unknown, entering zero will cause this calculation to be skipped.

The second and succeeding free format fields enter the time-drawdown pairs of which there must be ND. Entering one pair at a time reduces possibilities of mistakes in punching or keying in data. When all data is read, it is echoed as the first part of the output.

As iterations of the fitting algorithm are completed, the standard deviation (rms error) is printed along with the values of LC, Sc, and KB. If the routine completes ITMAX iterations without converging, a message appears announcing the fact and the program terminates. If convergence is successful, the convergence is announced. The best fit time-drawdown pairs are printed, and the values for LC, Sc, and KB are printed. If TCL has been given a positive value, the value of K' is also printed. The program then terminates.

One other feature which may prove to be of occasional use is the option IGENDATA (see Appendix I for a listing of LEAKYFIT). When this statement is set equal to zero (/0/), LEAKYFIT operates in a normal regression type mode. Setting IGENDATA equal to one (/1/) causes the program to terminate after the first iteration and produces a set of values which define a portion of the theoretical Leaky Aquifer type curve. This is useful if the user wishes to check point by point the deviation of a converged set of values against the theoretical values.

A Typical Batch Run of Program LEAKYFIT

```
 SC=          0.00002000
KB=          0.17000000
LC=          0.00000400
R=        2100.00000000
Q=         136.09625626
ERROR=          0.00100000
ITMAX=         50
```

T	SE
50.00000000	0.02000000
60.00000000	0.03000000
70.00000000	0.04000000
80.00000000	0.06000000
90.00000000	0.07000000
100.00000000	0.09000000
200.00000000	0.23000000
300.00000000	0.36000000
400.00000000	0.43000000
500.00000000	0.50000000
600.00000000	0.56000000
700.00000000	0.60000000
800.00000000	0.65000000
900.00000000	0.68000000
1000.00000000	0.70000000
1440.00000000	0.75000000

```
THE STANDARD DEVIATION FOR ITERATION NUMBER,     1IS          62.85103226FEET.

 LC=       0.40025644E-05
SC=       0.30000000E-04
KB=       0.25500000E 00


THE STANDARD DEVIATION FOR ITERATION NUMBER,     2IS          41.74836826FEET.

 LC=       0.40058561E-05
SC=       0.45000000E-04
KB=       0.38250000E 00
```

THE STANDARD DEVIATION FOR ITERATION NUMBER, 3IS 27.68291068FEET.

LC= 0.40082158E-05
SC= 0.67500000E-04
KB= 0.57375000E 00

THE STANDARD DEVIATION FOR ITERATION NUMBER, 4IS 18.30720854FEET.

LC= 0.40231423E-05
SC= 0.10125000E-03
KB= 0.86062501E 00

THE STANDARD DEVIATION FOR ITERATION NUMBER, 5IS 12.05551541FEET.

LC= 0.45295292E-05
SC= 0.15187500E-03
KB= 0.12909375E 01

THE STANDARD DEVIATION FOR ITERATION NUMBER, 6IS 7.88695598FEET.

LC= 0.45440256E-05
SC= 0.22781250E-03
KB= 0.19364063E 01

THE STANDARD DEVIATION FOR ITERATION NUMBER, 7IS 5.10868293FEET.

LC= 0.65234190E-05
SC= 0.34171875E-03
KB= 0.29046094E 01

THE STANDARD DEVIATION FOR ITERATION NUMBER, 8IS 3.25611785FEET.

LC= 0.97851286E-05
SC= 0.51257813E-03
KB= 0.43569141E 01

THE STANDARD DEVIATION FOR ITERATION NUMBER, 9IS 2.02096519FEET.

LC= 0.14677693E-04
SC= 0.76886720E-03
KB= 0.65353712E 01

THE STANDARD DEVIATION FOR ITERATION NUMBER, 10IS 1.19757660FEET.

LC= 0.22016539E-04
SC= 0.11533008E-02
KB= 0.98030568E 01

THE STANDARD DEVIATION FOR ITERATION NUMBER, 11IS 0.64873422FEET.

LC= 0.33024809E-04
SC= 0.16955480E-02
KB= 0.14704585E 02

THE STANDARD DEVIATION FOR ITERATION NUMBER, 12IS 0.29228624FEET.

LC= 0.49537214E-04
SC= 0.20554769E-02
KB= 0.18988866E 02

THE STANDARD DEVIATION FOR ITERATION NUMBER, 13IS 0.14641501FEET.

LC= 0.74305821E-04
SC= 0.21458429E-02
KB= 0.20350819E 02

THE STANDARD DEVIATION FOR ITERATION NUMBER, 14IS 0.10980574FEET.

LC= 0.11145873E-03
SC= 0.21363545E-02
KB= 0.20254100E 02

THE STANDARD DEVIATION FOR ITERATION NUMBER, 15IS 0.09851251FEET.

LC= 0.16718810E-03
SC= 0.21212719E-02
KB= 0.19969440E 02

THE STANDARD DEVIATION FOR ITERATION NUMBER, 16IS 0.07496800FEET.

LC= 0.25078214E-03
SC= 0.20963287E-02
KB= 0.19571687E 02

THE STANDARD DEVIATION FOR ITERATION NUMBER, 17IS 0.02587029FEET.

LC= 0.30246229E-03
SC= 0.20498505E-02
KB= 0.18829561E 02

THE STANDARD DEVIATION FOR ITERATION NUMBER, 18IS 0.00728759FEET.

LC= 0.30211786E-03
SC= 0.20454614E-02
KB= 0.18767505E 02

THE STANDARD DEVIATION FOR ITERATION NUMBER, 19IS 0.00711824FEET.

THE PARAMETERS CONVERGED IN 19 ITERATIONS.

THE BEST FIT TIME-DRAWDOWN PAIRS FOR THE CONVERGED VALUES OF S,T, AND L ARE

T	SE
50.00000000	0.01581064
60.00000000	0.02705157
70.00000000	0.04026270
80.00000000	0.05479956
90.00000000	0.07016420
100.00000000	0.08598887
200.00000000	0.23661267
300.00000000	0.35285641
400.00000000	0.44046858
500.00000000	0.50804146
600.00000000	0.56138865
700.00000000	0.60430527
800.00000000	0.63935507
900.00000000	0.66833135
1000.00000000	0.69252767
1440.00000000	0.76210560

LEAKAGE COEFFICIENT = 0.00030241

STORAGE COEFFICIENT = 0.00204471

TRANSMISSIVITY = 18.75390863

Program TSSLEAK and Its Use

This program is simply an interactive time-sharing version of LEAKYFIT. A listing appears in Appendix II . Prompting statements query the user for values of storage (SC), transmissivity (KB), leakage (LC), pumprate (Q), radial observation distance (R), number of time-drawdown pairs (NOTP). All data is echo printed, as well as ITMAX and ERROR, and the user is given an opportunity to correct the input data. next, the algorithm asks for time-drawdown pairs. Only NDTP pairs may be entered. After entry is complete, all pairs are echo printed and the user is again asked to examine for errors and correct any that are found. correction is done by entering the serial number of the erroneous time-drawdown pair and the correct values. After reading the corrections, the terminal again asks if there are any errors in the time-drawdown data. As long as affirmative responses are given by the user, the program will ask for data corrections. As soon as a negative resonse is given, the program proceeds.

At this point, output is indentical to that printed by LEAKYFIT (as may be seen by comparing the sample runs). Values of the rms error, iteration number and LC, Sc, and KB are printed at the end of each iterative step. When and if convergence occurs, this fact is announced. The "best fit" time-drawdown pairs are printed and the converged values of LC, SC, and KB are printed. The program then asks if the thickness of the semi-confining bed is known. if an affirmative response is made, the value of that parameter is asked, and the value of the leaky permeability $K^{(1)\prime}$ is printed. The program then terminates.

PROGRAM TSSLEAK (EXAMPLE RUN)

```
<W>1470 EQUALITY OR NON-EQUALITY COMPARISON MAY NOT BE MEANINGFUL I
N LOGICAL IF EXPRESSIONS
ESTIMATE FOR STORAGE ?
=.00002
ESTIMATE FOR TRANSMISSIVITY? L**2/T
=.17
ESTIMATE FOR LEAKAGE COEFFICIENT ? 1/L
=.000004
CONSTANT PUMPAGE RATE? L**3/T
=136.0963
OBSERVATION DISTANCE FROM PUMPING WELL? L
=2100
NUMBER OF DRAWDOWN-TIME PAIRS TO BE READ?
=16
ECHO THE INITIAL DATA
SC=          0.00002000
KB=          0.17000000
LC=          0.00000400
Q=         136.09630013
R=        2100.00000000
NDTP=         16
ITMAX=         50
ERROR=           0.00100000
ARE THERE ANY ERRORS IN DATA INPUT?
ANSWER YES IF ANY ERROR, OTHERWISE NO
=NO
TYPE IN DRAWDOWN-TIME PAIRS IN ORDER OF INCREASING TIME.
=.02 50
=.03 60
=.04 70
=.06 80
=.07 90
=.09 100
=.23 200
=.36 300
=.43 400
=.50 500
=.56 600
=.60 700

 .60 700


FILE CODE 41 ILLEGAL CHAR; CORRECTION =
***PROG.  L#   (ERR #32)
......   610
ILLEGAL CHAR IN DATA BELOW
ERROR IN COLUMN          8 OF
.60 700

TREAT ILLEGAL CHAR AS ZERO
=.6 700
```

this is not done by TSSLEAK

```
=.65 800
=.68 900
=.70 1000
THE PUMP TEST DATA IN DRAW-DOWN TIME PAIRS
.20000000E-01       0.50000000E 02
.30000000E-01       0.60000000E 02
.40000000E-01       0.70000000E 02
.60000000E-01       0.80000000E 02
.70000000E-01       0.90000000E 02
.90000000E-01       0.10000000E 03
.23000000E 00       0.20000000E 03
.36000000E 00       0.30000000E 03
.43000000E 00       0.40000000E 03
.50000000E 00       0.50000000E 03
.56000000E 00       0.60000000E 03
.60000000E 00       0.70000000E 04
.60000000E 00       0.70000000E 03
.65000000E 00       0.80000000E 03
.68000000E 00       0.90000000E 03
.70000000E 00       0.10000000E 04
ARE THERE ANY ERRORS IN DRAWDOWN-TIME PAIRS?
ANSWER YES OR NO
=YES
ENTER THE LINE NUMBER, CORRECT DRAWDOWN AND TIME
=12 .6 700
ARE THERE ANY ERRORS IN DRAWDOWN-TIME PAIRS?
ANSWER YES OR NO
=YES
ENTER THE LINE NUMBER, CORRECT DRAWDOWN AND TIME
=13 .65 800
ARE THERE ANY ERRORS IN DRAWDOWN-TIME PAIRS?
ANSWER YES OR NO
=YES
ENTER THE LINE NUMBER, CORRECT DRAWDOWN AND TIME
=14 .68 900
ARE THERE ANY ERRORS IN DRAWDOWN-TIME PAIRS?
ANSWER YES OR NO
=YES
ENTER THE LINE NUMBER, CORRECT DRAWDOWN AND TIME
=15 .7 1000
ARE THERE ANY ERRORS IN DRAWDOWN-TIME PAIRS?
ANSWER YES OR NO
=YES
ENTER THE LINE NUMBER, CORRECT DRAWDOWN AND TIME
=16 .75 1440
ARE THERE ANY ERRORS IN DRAWDOWN-TIME PAIRS?
ANSWER YES OR NO
=NO
THE PUMP TEST DATA IN DRAW-DOWN TIME PAIRS
.20000000E-01       0.50000000E 02
.30000000E-01       0.60000000E 02
.40000000E-01       0.70000000E 02
.60000000E-01       0.80000000E 02
.70000000E-01       0.90000000E 02
```

.90000000E-01	0.10000000E 03
.23000000E 00	0.20000000E 03
.36000000E 00	0.30000000E 03
.43000000E 00	0.40000000E 03
.50000000E 00	0.50000000E 03
.56000000E 00	0.60000000E 03
.60000000E 00	0.70000000E 03
.65000000E 00	0.80000000E 03
.68000000E 00	0.90000000E 03
.70000000E 00	0.10000000E 04
.75000000E 00	0.14400000E 04

THE STANDARD DEVIATION FOR ITERATION NUMBER, 1IS 62.85105276FEET.

LC= 0.40025728E-05
SC= 0.30000000E-04
KB= 0.25500000E 00

THE STANDARD DEVIATION FOR ITERATION NUMBER, 2IS 41.74835539FEET.

LC= 0.40058683E-05
SC= 0.45000000E-04
KB= 0.38250000E 00

THE STANDARD DEVIATION FOR ITERATION NUMBER, 3IS 27.68289542FEET.

LC= 0.40085048E-05
SC= 0.67500000E-04
KB= 0.57375000E 00

THE STANDARD DEVIATION FOR ITERATION NUMBER, 4IS 18.30680537FEET.

LC= 0.40249487E-05
SC= 0.10125000E-03
KB= 0.86062501E 00

THE STANDARD DEVIATION FOR ITERATION NUMBER, 5IS 12.05539763FEET.

LC= 0.58241606E-05
SC= 0.15187500E-03
KB= 0.12909375E 01

THE STANDARD DEVIATION FOR ITERATION NUMBER, 6IS 7.88670623FEET.

LC= 0.62520667E-05
SC= 0.22781250E-03
KB= 0.19364063E 01

THE STANDARD DEVIATION FOR ITERATION NUMBER, 7IS 5.10816783FEET.

LC= 0.79332423E-05
SC= 0.34171875E-03
KB= 0.29046094E 01

THE STANDARD DEVIATION FOR ITERATION NUMBER, 8IS 3.25584453FEET.

LC= 0.11899863E-04
SC= 0.51257813E-03
KB= 0.43569141E 01

THE STANDARD DEVIATION FOR ITERATION NUMBER, 9IS 2.02059296FEET.

LC= 0.17849795E-04
SC= 0.76886720E-03
KB= 0.65353712E 01

THE STANDARD DEVIATION FOR ITERATION NUMBER, 10IS 1.19702405FEET.

LC= 0.26774693E-04
SC= 0.11533008E-02
KB= 0.98030568E 01

THE STANDARD DEVIATION FOR ITERATION NUMBER, 11IS 0.64788027FEET.

LC= 0.40162040E-04
SC= 0.17007875E-02
KB= 0.14704585E 02

THE STANDARD DEVIATION FOR ITERATION NUMBER, 12IS 0.28963649FEET.

LC= 0.60243060E-04
SC= 0.20656104E-02
KB= 0.18959719E 02

THE STANDARD DEVIATION FOR ITERATION NUMBER, 13IS 0.14274371FEET.

LC= 0.90364590E-04
SC= 0.21426278E-02
KB= 0.20315696E 02

THE STANDARD DEVIATION FOR ITERATION NUMBER, 14IS 0.10534521FEET.

LC= 0.13554688E-03
SC= 0.21307199E-02
KB= 0.20141783E 02

THE STANDARD DEVIATION FOR ITERATION NUMBER, 15IS 0.08934250FEET.

LC= 0.20332033E-03
SC= 0.21240056E-02
KB= 0.22379864E 02

THE STANDARD DEVIATION FOR ITERATION NUMBER, 16IS 0.02354923FEET.

LC= 0.30343557E-03
SC= 0.20659742E-02
KB= 0.18751437E 02

THE STANDARD DEVIATION FOR ITERATION NUMBER, 17IS 0.00768745FEET.

LC= 0.30388260E-03
SC= 0.20429861E-02
KB= 0.18676353E 02

THE STANDARD DEVIATION FOR ITERATION NUMBER, 18IS 0.00712054FEET.

LC= 0.30264961E-03
SC= 0.20447055E-02
KB= 0.18740105E 02

THE STANDARD DEVIATION FOR ITERATION NUMBER, 19IS 0.00711852FEET.
THE PARAMETERS CONVERGED IN 19 ITERATIONS.

THE BEST FIT TIME-DRAWDOWN PAIRS FOR THE CONVERGED VALUES OF S,T, AND L ARE

THE PUMP TEST DATA IN DRAW-DOWN TIME PAIRS

T	SE
.50000000E 02	0.15777296E-01
.60000000E 02	0.27006592E-01
.70000000E 02	0.40208600E-01
.80000000E 02	0.54739043E-01
.90000000E 02	0.70099745E-01
.10000000E 03	0.85922604E-01
.20000000E 03	0.23658266E 00
.30000000E 03	0.35287328E 00
.40000000E 03	0.44051933E 00
.50000000E 03	0.50810999E 00
.60000000E 03	0.56146308E 00
.70000000E 03	0.60437723E 00
.80000000E 03	0.63941880E 00
.90000000E 03	0.66838298E 00
.10000000E 04	0.69256462E 00
.14400000E 04	0.76206753E 00

LEAKAGE COEFFICIENT = 0.00030248

```
STORAGE COEFFICIENT =            0.00204471

TRANSMISSIVITY =           18.74988437
DO YOU WANT TO COMPUTE AQUITARD PERMEABILITY ?
ANSWER YES IF TCL IS KNOWN OTHERWISE NO
=YES
THICKNESS OF CONFINING LAYER ?
=30
AQUITARD PERMEABILITY =          0.00005146
```

Program Hantush and Its Use

The algorithm titled HANTUSH is an interactive time sharing program which solves the drawdown equation for a leaky confined aquifer as formulated by Jacob and Hantush in 1955. The cover of this publication shows the definition of the appropriate aquifer system. A complete listing of this program appears in Appendix III. The input parameters for this program are the discharge Q, the inverse leakage coefficient LC, the confined aquifer transmissivity KB, the elastic storage coefficient SC, the unit length designation LU, the unit time designator TU, the observation radial distance R, and the pumping period T. The parameter LC is simply the inverse of B as defined by Jacob and Hantush, 1955.

The program introduces itself with a short explanation of its function and queries the user for the data indicated above. At critical points in the program all important data is echo printed and the user is given an opportunity to correct errors if they are found to exist. Corrections may be made in several ways. At points of direct questioning, appropriate responses (YES or NO) will determine whether or not corrections will be made. In other locations, data may simply be re-entered when the user recognizes an error. The program may be formally exited by entering zeros for R and T at any read statement, or simply by using the BREAK key.

An example run is included here as a tutorial. This run was made in TSS FORTRAN as compiled on the Honeywell 66/60 at the University of Kansas Computation Center. Access and run commands shown here may or may not be meaningful on other systems. All queries (data reading locations) are indicated by "equals" signs (=). The user enters the appropriate data string in free format following the prompt symbol. If the user fails to enter all the neccessary data in the first string, the prompting symbol will continue to appear until all requested data has been read. Free format means that data fields are not of specific size or location, but are entered in a serial fashion, each field separated by a blank space or a comma. The responses of the user are underlined in the example.

The obvious advantage of this program is the accuracy of the value of W(U,r/B) as compared to interpolating values from tables. This is especially true if the table is sparce in its value range of U and r/B. Figure 10a shows drawdowns computed from exact and interpolated values of W(U,r/B). These values are compared to drawdowns computed by the program. Note that for the exact table values (Figure 10b) the hand computed and algorithm computed values are virtually exact, while there are considerable differences in the interpolated values. The table used is from Walton (1970) and a linear interpolation scheme is used. A more sophistocated interpolation scheme might reduce the discrepancy in computed drawdown.

Another application of this algorithm, which has not been initiated as of this writing, is its use as the core of a leaky confined well field simulator. Well field simulators for confined aquifers are common, but no such simulator is known to exist for the leaky artesian case.

10 b. Tables of Drawdowns for Exact Table Values of U, r/B, and W(U,r/B).

Drawdown From Exact Table Values				Drawdown From Program			
U	r/B	W(U,r/B)	s	U	r/B	W(U/r/B)	s
0.05	0.2	2.3110	69.883	.05	.20	2.3110	69.884
0.01	0.6	1.5550	9.405	.01	.5999	1.5551	9.405
0.0001	0.03	7.2122	0.436	.00010	.0300	7.2123	0.435

10 a. Tables of Drawdowns for Interpolated Values U, r/B, and W(U, r/B).

U	r/B	W(U,r/B)	s	U	r/B	W(U,r/B)	s
4.41	0.9392	0.0288	0.1146	4.4100	0.9392	.0022	0.009
0.7350	0.9392	0.3194	1.2708	.7350	0.9392	0.2896	1.152
0.5625	0.3354	0.5007	1.9922	0.5625	0.3354	0.4760	1.894
1.3496	0.3354	0.1965	0.7818	1.3496	0.3354	0.1236	0.492

(Drawdown interpolated from table) (Drawdown computed by program)

A Typical Interactive Timesharing Session with HANTUSH

```
THIS PROGRAM CALCULATES THE DRAWDOWN IN A
LEAKY ARTESIAN AQUIFER.  THE RADIAL DISTANCE FOR THE
OBSERVATION WELL AND THE PUMPING PERIOD MAY BE CONSTANT
OR VARIABLE.


ENTER THE FOLLOWING DATA IN A FREE FORMAT FIELD:
Q = PUMPING RATE (L**3/T)
LC = INVERSE LEAKAGE COEFFICIENT (1/L)
KB = AQUIFER TRANSMISSIVITY (L**2/T)
SC = STORAGE COEFFICIENT (UNITLESS)
UNIT = MIN OR DAY (UNIT OF TIME USED IN THE INPUT DATA)
=5.E4
=750
=8.E3
=.003
=DAY

ARE THERE ANY ERRORS IN THE ABOVE ENTRIES?
IF NOT, ANSWER NO.
=YES

ENTER THE FOLLOWING DATA IN A FREE FORMAT FIELD:
Q = PUMPING RATE (L**3/T)
LC = INVERSE LEAKAGE COEFFICIENT (1/L)
KB = AQUIFER TRANSMISSIVITY (L**2/T)
SC = STORAGE COEFFICIENT (UNITLESS)
UNIT = MIN OR DAY (UNIT OF TIME USED IN THE INPUT DATA)
=5.E4
=.00133333
=8.E3
=.003
=DAY

ARE THERE ANY ERRORS IN THE ABOVE ENTRIES?
IF NOT, ANSWER NO.
=NO

THE FOLLOWING PARAMETERS ARE USED IN THIS SOLUTION:

Q=  50000.00

LC=0.00133333

KB=   8000.00

SC=0.00300000

UNIT =DAY

ENTER THE FOLLOWING DATA:
R,T
R = DISTANCE TO OBSERVATION WELL (L)
T = LENGTH OF PUMPING PERIOD (T)
```

```
TERMINATE BY  RETURNING BLANK FIELDS.

ENTER R,T
=100.,.05

U=  0.018750

R/B=   0.13333

W(U,R/B)=  3.213409

THE DRAWDOWN   100. FEET FROM THE PUMPING WELL
IS       1.598 FEET AFTER      0.050 DAYS OF PUMPING.

ENTER R,T
=50.,.5

U=  0.000469

R/B=   0.06667

W(U,R/B)=  5.626895

THE DRAWDOWN    50. FEET FROM THE PUMPING WELL
IS       2.799 FEET AFTER      0.500 DAYS OF PUMPING.

ENTER R,T
=100.,.5

U=  0.001875

R/B=   0.13333

W(U,R/B)=  4.259999

THE DRAWDOWN   100. FEET FROM THE PUMPING WELL
IS       2.119 FEET AFTER      0.500 DAYS OF PUMPING.

ENTER R,T
=0.,0.

YOU HAVE TERMINATED THE PROGRAM.
IS THIS CORRECT?
=YES
>>STOP JOURNALIZING
```

Discussion and Summary

This paper has examined the methodlogy of solving the leaky artesian aquifer pumping test analysis by using a numerical regression algorithm motivated by sensitivity analysis. A byproduct is the solution to the drawdown equation. Results show that the drawdown solutions are unique and unambiguous. The results of the regression program are also unique, but only unambiguous to the extent of the data being analyzed. This points to the fact that while this type of programming can ease the life of the hydrologist, it does not appear that it will reduce his role; for the results of a pumping test must be viewed as a complex interaction of the fluid and the formations in which they occur. Regression analysis can make the hydrologist more productive, but it should not be an excuse to become lazy.

Bibliography

Abramowitz, M., and Stegun, I.A., 1968, Handbook of mathematical functions: New York, Dover Publications, p. 231.

Cobb, P.M., McElwee, C.D., and Butt, M.A., 1978, Leaky aquifer parameter identification by sensitivity analysis: Transactions of the American Geophysical Union, v. , p. .

Hantush, M.S., 1949, Plain potential flow of ground water with linear leakage, doctoral dissertation, University of Utah.

Hantush, M.S., 1960, Modification of the theory of leaky aquifers: Journal of Geophysical Research, v. 65, no. 11, p. 3713-3725.

Hantush, M.S., and Jacob, C.E., 1955, Non-steady radial flow in an infinite leaky aquifer: Transactions of the American Geophysical Union, v. 36, pp. 95-100.

Hildebrand, F.B., 1962, Advanced calculus for applications: Englewood Cliffs, Prentice-Hall, p. 360.

Kreysig, E., 1972, Advanced engineering mathematics: New York, John Wiley and Sons, Inc., pp. 329-331.

McCuen, R.H., 1973, component sensitivity: a tool for the analysis of complex water resources systems: Water Resources Research, v. 9, no. 1, p. 243-246.

McElwee, C.D., and Yukler, M.A., 1978, Sensitivity of groundwater models with respect to variations in transmissivity and storage: Water Resources Research, v. 14, no. 3, p. 451.

McElwee, C.D., 1980a, Theis papameter evaluation from pumping tests by sensitivity analysis: Ground Water, v. 18, pp. 56-60.

McElwee, C.D., 1980b, The Theis equation: evaluation, sensitivity to storage and transmissivity, and automated fit of pump test data: Kansas Geological Survey, Groundwater Series 3.

Tomovic, R., 1962, Sensitivity analysis of dynamic systems: New York, McGraw-Hill, 141 p.

Vemuri, V., Dracup, J.A., Erdmann, R.C., and Vemuri, N., 1969, Sensitivity analysis method of system identification and its potential in hydrologic research: Water Resources Research, v. 5, no. 2, p. 341-349.

Walton, W.C., 1970, Groundwater resource evaluation: New York, McGraw-Hill, 664 p.

Yukler, M.A., 1976, Analysis of error in groundwater modeling, Ph.D. dissertation, The University of Kansas, 182 p.

APPENDIX I.   PROGRAM LEAKYFIT

```
C        PROGRAM  LEAKYFIT
        PARAMETER ND = 16
       PARAMETER DECRES=-.2
       PARAMETER INCRES=.5
        COMMON LC,ITERAT,LCOUNT,KB,KBCOUNT
        DIMENSION  X(3), SE(ND), T(ND), SGS(ND)
       REAL KB,LC,INCRES
       PI=3.1415926
       DATA IGENDAT/0/
C      Q=PUMPAGE (CUFT/DAY)
C      SC=STORAGE COEFF. (UNITLESS)
C      KB= TRANSMISSIVITY   (FT2/DAY)
C      LC=MODIFIED COEFFICIENT OF LEAKAGE (1/FT.)
C      ERROR= CONVERGENCE CRITERIA FOR MAIN DO LOOPE (UNITLESS)
C      R= RADIAL DISTANCE FROM PUMPING WELL TO OBSERVATION WELL(FT.)
C      ITMAX= MAX NUMBER OF ITERATIONS (UNITLESS)
C      T=  TIME (DAY)
C      SE=  EXPERIMENTAL DRAWDOWN (FT.)
C       TCL=THICKNESS OF CONFINING LAYER (FT)
C      INITIALIZE PROGRAM
       N=3
       ERROR=0.001
       ITMAX=50
       ITERAT=0
       CALL FXOPT(89,1,1,0)
       READ(5,1) SC,KB,LC,R,Q,TCL
        READ(5,1)(T(I),SE(I),I=1,ND)
  1     FORMAT(V)
       WRITE(6,2) SC,KB,LC,R,Q,ERROR,    ITMAX
     2 FORMAT(1H1,'SC=',F20.10/'KB=',F20.10/'LC=',F20.10/'R=',F20.10/
      1'Q=',F20.10/'ERROR=',F20.10/              'ITMAX=',I10)
       IF(TCL .GT.0)WRITE(6,7)TCL
  7    FORMAT('TCL=',F20.10)
C      ECCO PRINT DRAWDOWN - TIME DATA PAIRS
       WRITE (6,3)
     3 FORMAT(1H0,T15,1HT ,T35,2HSE)
       WRITE(6,4) (T(I),SE(I),  I=1,ND)
     4 FORMAT(1H0,  (F20.10,5X,F20.10/ ))
       IF(IGENDAT.EQ.1) GO TO 101
        LCOUNT=0
        KBCOUNT=0
 777    CONTINUE
 508    CONTINUE
        IF(KBCOUNT.GT.0.OR.LCOUNT.GT.0)ITERAT=ITERAT-1
        LCOUNT=0
        KBCOUNT=0
C   INITIALIZE ITERATIONS 8888888888888888************************
 100    ITERAT=ITERAT+1
C      ZERO OUT SUMMATIONS
       SDELS2 =0.0
       SUSCDS=0.0
       SURCDS=0.0
```

```
      SUKBDS=0.0
      SUKBUS=0.0
      SURCUK=0.0
      SUSCUR=0.0
      SURC2=0.0
      SUSC2=0.0
      SUKB2=0.0
C     ZERO OUT  X MATRIX**********************************************************
      DO 10 K=1,N
       X(K)=0.0
   10 CONTINUE
C     COMPUTE RB
  101 RB=R*LC
      DO 1001  I=1,ND
C
C     COMPUTE U
      U=(R*R*SC)/(4*KB*T(I))
C     COMPUTE THEORETICAL DRAWDOWN
      SG=(Q/(4*PI*KB))*W(U,RB)
       IF(LCOUNT .EQ. 0 .AND. KBCOUNT .EQ.0) GO TO 333
       GO TO 777
  333 SGS(I)=SG
      IF(IGENDAT.EQ.1) GO TO 1001
C     COMPUTE DIFFERENCE BETWEEN THEORETICAL AND EXPERIMENTAL DRAWDOWN
      DELS=SE(I)-SG
      IF(ABS(DELS) .LT. 1.0E-3) DELS=0.0
      SDELS2=SDELS2+DELS*DELS
C     COMPUTE DUMMY COEFFICIENT 'Z'
      Z= (U+(RB*RB)/(4.*U))
C     COMPUTE SENSITIVITY COEFFICIENT AND SUMMATIONS
      USC=-(Q/(4.*PI*KB)*(1/U)*((R*R)/(4.*KB*T(I)))*EXP(-Z))
      UKB=-SG/KB+(Q/(4.*PI*KB))*((R*R*SC)/
     1(4.*KB*KB*T(I)))*(1/U)*EXP(-Z)
       IF(USC.EQ.0.OR.UKB.EQ.0)ITERAT=ITERAT-1
       IF(USC.EQ.0.OR.UKB.EQ.0)GO TO 987
      RBM=R*.99*LC
      RBP=R*1.01*LC
      WPLUS=W(U,RBP)
       IF(LCOUNT .EQ. 0 .AND. KBCOUNT .EQ.0)GO TO 334
       GO TO 777
  334 WMINUS=W(U,RBM)
       IF(LCOUNT .EQ. 0 .AND. KBCOUNT .EQ.0)GO TO 335
       GO TO 777
  335 URC=(Q/(4.0*PI*KB))*(WPLUS-WMINUS)/(.02*LC)
       IF(URC.EQ.0)ITERAT=ITERAT-1
       IF(URC.EQ.0)GO TO 987
      SUKB2=SUKB2+UKB*UKB
      SUSC2=SUSC2+USC*USC
      SURC2=SURC2+URC*URC
      SUSCUR=SUSCUR+USC*URC
      SUKBUS=SUKBUS+UKB*USC
      SURCUK=SURCUK+URC*UKB
      SUSCDS=SUSCDS+USC*DELS
      SURCDS=SURCDS+URC*DELS
```

```
      SUKBDS=SUKBDS+UKB*DELS
 1001 CONTINUE
      IF(IGENDAT.EQ.0) GO TO 102
      WRITE(6,3)
      WRITE(6,4) (T(I),SGS(I),I=1,ND)
      IF(IGENDAT.EQ.1) GO TO 133
C     COMPUTE MATRIX TO BE SOLVED FOR SENSITIVITY DELTAS
  102  U11= SURC2*LC
       U12=SUSCUR*SC
       U13=SURCUK*KB
       U14=SURCDS
       U21=SUSCUR*LC
       U22=SUSC2*SC
       U23=SUKBUS*KB
       U24=SUSCDS
       U31=SURCUK*LC
       U32=SUKBUS*SC
       U33=SUKB2*KB
       U34=SUKBDS
C      SOLVE MATRIX BY DIRECT GAUSS ELIMINATION
      X(3)=((U14*U21-U24*U11)*(U12*U31-U32*U11)
     1-(U14*U31-U34*U11)*(U12*U21-U22*U11))/
     2((U13*U21-U23*U11)*(U12*U31-U32*U11)
     3-(U13*U31-U33*U11)*(U12*U21-U22*U11))
      X(2)=((U14*U21-U24*U11)-(U13*U21-U23*U11)*X(3))/(U12*U21-U22*U11)
      X(1)=(U14-U13*X(3)-U12*X(2))/U11
       GO TO 876
 987   CONTINUE
C     UPDATE INITIAL GUESS VALUES OF  SC, KB, LC
       X(1)=INCRES
       LC=LC*(1.0+X(1))
       X(2)=INCRES
       SC=SC*(1.0+X(2))
       IF(SC.GE.1)SC=1.0
       X(3)=INCRES
       KB=KB*(1.0+X(3))
       GO TO 765
 876   CONTINUE
C      COMPUTE STANDARD DEVIATION FOR EACH ITERATION
       SIGMA=SQRT(SDELS2/ND)
      WRITE(6,202) ITERAT,SIGMA
  202 FORMAT(1H0,'THE STANDARD DEVIATION FOR ITERATION NUMBER,'I5,
     1'IS',F20.10,'FEET.')
C     UP-DATE COEFFICIENTS
 765   CONTINUE
      IF(X(1).LT.DECRES) X(1)=DECRES
      IF(X(1).GT.INCRES) X(1)=INCRES
      LC=LC*(1.0+X(1))
      IF(X(2).LT.DECRES) X(2)=DECRES
      IF(X(2).GT.INCRES) X(2)=INCRES
      SC=SC*(1.0+X(2))
        IF(SC .GE.1)SC=1
      IF(X(3).LT.DECRES) X(3)=DECRES
      IF(X(3).GT.INCRES) X(3)=INCRES
```

```
      KB=KB*(1.0+X(3))
C     CHECK FOR DELTA CONVERGENCE
      IF(ABS( X(1)    ).GT.ERROR.
     1 OR.ABS( X(2)    ).GT.ERROR.
     2  OR.ABS( X(3)    ).GT.ERROR) GOTO 500
      GO TO 600
  500 IF (ITERAT.GE.ITMAX) GO TO 999
       IF(URC.EQ.0.OR.USC.EQ.0.OR.UKB.EQ.0)GO TO 100
      WRITE(6,501) LC,SC,KB
  501 FORMAT(1H0,3HLC=,E20.10/3HSC=,E20.10/3HKB=,E20.10//)
      GO TO 100
C      WRITE OUT FINAL PROGRAM STATUS
 600  WRITE(6,601) ITERAT
  601 FORMAT(1H1,'THE PARAMETERS CONVERGED IN',I5,1X,'ITERATIONS.')
      WRITE (6,203)
  203 FORMAT(1H0,37HTHE BEST FIT TIME-DRAWDOWN PAIRS FOR ,
     139HTHE CONVERGED VALUES OF S,T, AND L ARE )
      WRITE(6,3)
      WRITE(6,4) (T(I),SGS(I),I=1,ND)
      WRITE (6,201) LC,SC,KB
  201 FORMAT(1H0,22HLEAKAGE COEFFICIENT = ,F20.10,
     1//22HSTORAGE COEFFICIENT = ,F20.10,
     2//17HTRANSMISSIVITY = ,F20.10)
      IF(TCL.GT.0)GO TO 1010
      STOP
 1010 AK=KB*TCL*LC**2
      WRITE(6,1111)AK
 1111 FORMAT(1H0,'AQUITARD PERMEABILITY=',F20.10)
      STOP
  999 WRITE (6,998) ITMAX
  998 FORMAT(1H0,'THE PROGRAM DID NOT CONVERGE IN ',I5,1X,'ITERATIONS.')
  133 STOP
      END
C      FILE WURB
C      FILE WURB IS A LIST OF FUNCTIONS REQUIRED IN THE
C      SOLUTION OF THE LEAKY ARTESIAN WELL FUNCTION W(U,R/B).
       FUNCTION W(U,RB)
C      THIS FUNCTION DEFINES THE LEAKY ARTESIAN WELL FUNCTION.
C      THE THREE FORMS CORRESPOND TO THOSE OUTLINED IN
C      HANTUSH AND JACOB, 1955.
       COMMON LC,ITERAT,LCOUNT,KB,KBCOUNT
       REAL LC,KB
      IF(U.GE.1.0) GO TO 1000
      IF(U.LT.1.0.AND.(RB*RB).GT.U) GO TO 2000
      IF(U.LT.1.0.AND.(RB*RB).LE.U) GO TO 3000
 1000 W = SS(U,RB)
      GO TO 5000
 2000 W = (2*AK0(RB)-SS(U,RB))
      GO TO 5000
 3000 F1 = (RB*RB*.25/U)
       W=2*AK0(RB)-AI0(RB)*(EI(F1))+EXP(-F1)*
     1 (0.5772+ALOG(U)+EI(U)-U+U*
     2 ((AI0(RB)-1)/(RB*RB*.25))-U*U*SUM(U,RB))
 5000 RETURN
```

APPENDIX II.   PROGRAM TSSLEAK

```
C         PROGRAM  TSSLEAK
       PARAMETER DECRES=-.2
       PARAMETER INCRES=.5
        COMMON LC,ITERAT,LCOUNT,KB,KBCOUNT
        DIMENSION  X(3),  SE(100), T(100),  SGS(100)
        REAL KB,LC
        CHARACTER*3  CHEKDATA, WRITAQP
       PI=3.1415926
       DATA IGENDAT/0/
C      Q=PUMPAGE (L**3/T)
C      SC=STORAGE COEFF. (UNITLESS)
C      KB= TRANSMISSIVITY   (L**2/T)
C      LC=MODIFIED COEFFICIENT OF LEAKAGE (1/L)
C      ERROR= CONVERGENCE CRITERIA FOR MAIN DO LOOPE (UNITLESS)
C      R= RADIAL DISTANCE FROM PUMPING WELL TO OBSERVATION WELL(L)
C      EPS=  CONVERGENCE CRITERIA FOR SUBROUTINE SIMUL (UNITLESS)
C      ITMAX= MAX NUMBER OF ITERATIONS (UNITLESS)
C      T=  TIME
C      SE=  EXPERIMENTAL DRAWDOWN (L)
C       TCL=THICKNESS OF CONFINING LAYER (L)
C      INITIALIZE PROGRAM
       N=3
       ERROR=0.001
       ITMAX=50
       ITERAT=0
       CALL FXOPT(89,1,1,0)
C        READ IN THE INITIAL  DATA
  1     PRINT  112
  112   FORMAT('ESTIMATE FOR STORAGE ?')
        READ:SC
        PRINT 113
  113   FORMAT('ESTIMATE FOR TRANSMISSIVITY? L**2/T')
        READ:KB
        PRINT 114
  114   FORMAT('ESTIMATE FOR LEAKAGE COEFFICIENT ? 1/L')
        READ:LC
        PRINT 115
  115   FORMAT('CONSTANT PUMPAGE RATE? L**3/T ')
        READ:Q
        PRINT 116
 116    FORMAT('OBSERVATION DISTANCE FROM PUMPING WELL? L')
        READ:R
        PRINT 117
  117   FORMAT('NUMBER OF DRAWDOWN-TIME PAIRS TO BE READ?')
        READ:NDTP
C       ECHO PRINT THE INITIAL DATA
        WRITE(6,2)SC,KB,LC,Q,R,NDTP,ITMAX,ERROR
   2    FORMAT(' ECHO THE INITIAL DATA '/'SC=',F20.10/'KB='F20.10/'LC=',F20.10/
&       'Q=',F20.10/'R=',F20.10/'NDTP=',I10/'ITMAX=',I10/
&       'ERROR=',F20.10)
C      CHECK FOR THE DATA INPUT
        PRINT 212
```

```
  212   FORMAT('ARE THERE ANY ERRORS IN DATA INPUT?',/,
&         'ANSWER YES IF ANY ERROR, OTHERWISE NO')
        READ:CHEKDATA
        IF(CHEKDATA .EQ. 3HYES)GO TO 1
C      TYPE IN DRAWDOWN-TIME PAIRS IN ORDER OF INCREASING TIME
        PRINT 213
  213   FORMAT('TYPE IN DRAWDOWN-TIME PAIRS IN ORDER OF INCREASING TIME.')
        DO 214 I=1,NDTP
  214   READ:SE(I),T(I)
C      ECHO PRINT THE DRAWDOWN-TIME PAIRS
        WRITE(6,4)(SE(I),T(I),I=1,NDTP)
  4     FORMAT('THE PUMP TEST DATA IN DRAW-DOWN TIME PAIRS',/(2E20.8))
C        CHECK FOR ANY ERRORS IN THE DATA INPUT
        NEWDATA=0
  216   PRINT 215
  215   FORMAT('ARE THERE ANY ERRORS IN DRAWDOWN-TIME PAIRS?',/,
&       'ANSWER YES OR NO')
        READ:CHEKDATA
        IF(CHEKDATA.EQ.2HNO.AND.NEWDATA.EQ.0)GO TO 13
        IF(CHEKDATA.EQ.2HNO.AND.NEWDATA.GE.1)GO TO 515
        IF(CHEKDATA.EQ.3HYES)GO TO 218
        IF(CHEKDATA.NE.3HYES.OR.CHEKDATA.NE.2HNO)PRINT 219
  219   FORMAT('YOU HAVE AN ERROR IN DATA ENTRY')
  218   PRINT 217
  217   FORMAT('ENTER THE LINE NUMBER, CORRECT DRAWDOWN AND TIME')
        READ:M,SE(M),T(M)
        NEWDATA=NEWDATA+1
        GO TO 216
C        ECHO PRINT THE CORRECTED DRAWDOWN TIME PAIRS.
  515   WRITE(6,4)(SE(I),T(I),I=1,NDTP)
  13    IF(IGENDAT.EQ.1) GO TO 101
        LCOUNT=0
        KBCOUNT=0
 777    CONTINUE
 508    CONTINUE
        IF(KBCOUNT.GT.0.OR.LCOUNT.GT.0)ITERAT=ITERAT-1
        LCOUNT=0
        KBCOUNT=0
C    INITIALIZE ITERATIONS 8888888888888888************************
 100    ITERAT=ITERAT+1
C      ZERO OUT SUMMATIONS
       SDELS2 =0.0
       SUSCDS=0.0
       SURCDS=0.0
       SUKBDS=0.0
       SUKBUS=0.0
       SURCUK=0.0
       SUSCUR=0.0
       SURC2=0.0
       SUSC2=0.0
       SUKB2=0.0
C      ZERO OUT  X MATRIX*************************************************
       DO 10 K=1,N
        X(K)=0.0
```

```
   10 CONTINUE
C     COMPUTE RB
  101 RB=R*LC
      DO 1001  I=1,NDTP
C
C     COMPUTE U
      U=(R*R*SC)/(4*KB*T(I))
C     COMPUTE THEORETICAL DRAWDOWN
      SG=(Q/(4*PI*KB))*W(U,RB)
       IF(LCOUNT .EQ. 0 .AND. KBCOUNT .EQ.0) GO TO 333
       GO TO 777
  333 SGS(I)=SG
      IF(IGENDAT.EQ.1) GO TO 1001
C     COMPUTE DIFFERENCE BETWEEN THEORETICAL AND EXPERIMENTAL DRAWDOWN
      DELS=SE(I)-SG
      IF(ABS(DELS) .LT. 1.0E-3) DELS=0.0
      SDELS2=SDELS2+DELS*DELS
C     COMPUTE DUMMY COEFFICIENT 'Z'
      Z= (U+(RB*RB)/(4.*U))
C     COMPUTE SENSITIVITY COEFFICIENT AND SUMMATIONS
      USC=-(Q/(4.*PI*KB)*(1/U)*((R*R)/(4.*KB*T(I)))*EXP(-Z))
      UKB=-SG/KB+(Q/(4.*PI*KB))*((R*R*SC)/
&     (4.*KB*KB*T(I)))*(1/U)*EXP(-Z)
       IF(USC.EQ.0.OR.UKB.EQ.0)ITERAT=ITERAT-1
       IF(USC.EQ.0.OR.UKB.EQ.0)GO TO 987
      RBM=R*.99*LC
      RBP=R*1.01*LC
      WPLUS=W(U,RBP)
       IF(LCOUNT .EQ. 0 .AND. KBCOUNT .EQ.0)GO TO 334
       GO TO 777
  334 WMINUS=W(U,RBM)
       IF(LCOUNT .EQ. 0 .AND. KBCOUNT .EQ.0)GO TO 335
       GO TO 777
  335 URC=(Q/(4.0*PI*KB))*(WPLUS-WMINUS)/(.02*LC)
       IF(URC.EQ.0)ITERAT=ITERAT-1
       IF(URC.EQ.0)GO TO 987
      SUKB2=SUKB2+UKB*UKB
      SUSC2=SUSC2+USC*USC
      SURC2=SURC2+URC*URC
      SUSCUR=SUSCUR+USC*URC
      SUKBUS=SUKBUS+UKB*USC
      SURCUK=SURCUK+URC*UKB
      SUSCDS=SUSCDS+USC*DELS
      SURCDS=SURCDS+URC*DELS
      SUKBDS=SUKBDS+UKB*DELS
 1001 CONTINUE
      IF(IGENDAT.EQ.0) GO TO 102
      WRITE(6,3)
    3  FORMAT(1H0,T15,1HT ,T35,2HSE)
      WRITE(6,4) (T(I),SGS(I),I=1,NDTP)
      IF(IGENDAT.EQ.1) GO TO 133
C     COMPUTE MATRIX TO BE SOLVED FOR SENSITIVITY DELTAS
  102  U11= SURC2*LC
       U12=SUSCUR*SC
```

```
      U13=SURCUK*KB
      U14=SURCDS
      U21=SUSCUR*LC
      U22=SUSC2*SC
      U23=SUKBUS*KB
      U24=SUSCDS
      U31=SURCUK*LC
      U32=SUKBUS*SC
      U33=SUKB2*KB
      U34=SUKBDS
C     SOLVE MATRIX BY DIRECT GAUSS ELIMINATION
     X(3)=((U14*U21-U24*U11)*(U12*U31-U32*U11)
&     -(U14*U31-U34*U11)*(U12*U21-U22*U11))/
&     ((U13*U21-U23*U11)*(U12*U31-U32*U11)
&     -(U13*U31-U33*U11)*(U12*U21-U22*U11))
     X(2)=((U14*U21-U24*U11)-(U13*U21-U23*U11)*X(3))/(U12*U21-U22*U11)
     X(1)=(U14-U13*X(3)-U12*X(2))/U11
      GO TO 876
 987  CONTINUE
C   UPDATE INITIAL GUESS VALUES OF  SC, KB, LC
      X(1)=INCRES
      LC=LC*(1.0+X(1))
      X(2)=INCRES
      SC=SC*(1.0+X(2))
      IF(SC.GE.1)SC=1.0
      X(3)=INCRES
      KB=KB*(1.0+X(3))
      GO TO 765
 876  CONTINUE
C     COMPUTE STANDARD DEVIATION FOR EACH ITERATION
      SIGMA=SQRT(SDELS2/NDTP)
     WRITE(6,202) ITERAT,SIGMA
  202 FORMAT(1H0,'THE STANDARD DEVIATION FOR ITERATION NUMBER,'I5,
&     'IS',F20.10,'FEET.')
C    UP-DATE COEFFICIENTS
 765  CONTINUE
     IF(X(1).LT.DECRES) X(1)=DECRES
     IF(X(1).GT.INCRES) X(1)=INCRES
     LC=LC*(1.0+X(1))
     IF(X(2).LT.DECRES) X(2)=DECRES
     IF(X(2).GT.INCRES) X(2)=INCRES
     SC=SC*(1.0+X(2))
       IF(SC .GE.1)SC=1
     IF(X(3).LT.DECRES) X(3)=DECRES
     IF(X(3).GT.INCRES) X(3)=INCRES
     KB=KB*(1.0+X(3))
C    CHECK FOR DELTA CONVERGENCE
     IF(ABS( X(1)   ).GT.ERROR.
&      OR.ABS( X(2)   ).GT.ERROR.
&       OR.ABS( X(3)   ).GT.ERROR) GOTO 500
     GO TO 600
  500 IF (ITERAT.GE.ITMAX) GO TO 999
      IF(URC.EQ.0.OR.USC.EQ.0.OR.UKB.EQ.0)GO TO 100
     WRITE(6,501) LC,SC,KB
```

```
  501 FORMAT(1H0,3HLC=,E20.10/3HSC=,E20.10/3HKB=,E20.10//)
      GO TO 100
C      WRITE OUT FINAL PROGRAM STATUS
 600  WRITE(6,601) ITERAT
  601 FORMAT(1H1,'THE PARAMETERS CONVERGED IN',I5,1X,'ITERATIONS.')
      WRITE (6,203)
  203 FORMAT(1H0,37HTHE BEST FIT TIME-DRAWDOWN PAIRS FOR ,
&     39HTHE CONVERGED VALUES OF S,T, AND L ARE )
      WRITE(6,3)
      WRITE(6,4) (T(I),SGS(I),I=1,NDTP)
      WRITE (6,201) LC,SC,KB
  201 FORMAT(1H0,22HLEAKAGE COEFFICIENT = ,F20.10,
&     //22HSTORAGE COEFFICIENT = ,F20.10,
&     //17HTRANSMISSIVITY = ,F20.10)
       PRINT   118
  118  FORMAT(' DO YOU WANT TO COMPUTE AQUITARD PERMEABILITY ?',/,
&      'ANSWER YES IF TCL IS KNOWN OTHERWISE NO')
       READ:WRITAQP
       IF(WRITAQP.NE.2HNO)GO TO 1111
       IF(WRITAQP.NE.3HYES)STOP
 1111  PRINT 119
  119  FORMAT('THICKNESS OF CONFINING LAYER ?')
       READ:TCL
       IF(TCL.LE.0)STOP
       AQP=KB*TCL*LC**2
       WRITE(6,120)AQP
  120  FORMAT('AQUITARD PERMEABILITY =',F20.10)
       STOP
  999 WRITE (6,998) ITMAX
  998 FORMAT(1H0,'THE PROGRAM DID NOT CONVERGE IN ',I5,1X,'ITERATIONS.')
  133 STOP
      END
C      FILE WURB
```

APPENDIX III. PROGRAM HANTUSH

```
C      PROGRAM HANTUSH
       CHARACTER CHECK*3,LU*3,TU*3
       REAL KB,LC
       AA = 1.0
C      THIS PROGRAM COMPUTES THE DRAWDOWN IN A LEAKY AQUIFER AS
C      DEFINED BY JACOB AND HANTUSH, 1955. ALL INPUT IS IN CONSISTENT
C      UNITS.  A FULLY PENETRATING WELL IN AN ARTESIAN AQUIFER
C      AND NO WATER RELEASED FROM STORAGE IN THE AQUITARD,
C      WITH CONSTANT DISCHARGE CONDITIONS ARE THE PRINCIPLE
C      ASSUMPTIONS.
C      R= RADIUS OF OBSERVATION WELL FROM PUMPED WELL (L)
C      S= DRAWDOWN (L)
C      T= TIME (T)
C      Q= PUMPING RATE (L**3/T)
C      LC= INVERSE LEAKAGE COEFFICIENT OF SEMICONFINING BED (1/L)
C      LC= 1/B
C      KB= TRANSMISSIVITY OF AQUIFER (L**2/T)
C      SC= STORAGE COEFFICIENT OF AQUIFER (UNITLESS)
       PI= 3.1415926
       WRITE(6,1)
     1 FORMAT(1H0,'THIS PROGRAM CALCULATES THE DRAWDOWN IN A',
&      /,'LEAKY ARTESIAN AQUIFER.  THE RADIAL DISTANCE FOR THE',/,
&      'OBSERVATION WELL AND THE PUMPING PERIOD MAY BE CONSTANT',/,
&      'OR VARIABLE.  ANY CONSISTENT SET OF UNITS MAY BE USED.'//)
   100 WRITE (6,2)
     2 FORMAT(1H0,'ENTER THE FOLLOWING DATA IN A FREE FORMAT FIELD:',/
&      'Q = PUMPING RATE (L**3/T)',/,
&      'LC = INVERSE LEAKAGE COEFFICIENT (1/L)',/,
&      'KB = AQUIFER TRANSMISSIVITY (L**2/T)',/,
&      'SC = STORAGE COEFFICIENT (UNITLESS)',/,
&      'LU = UNIT OF LENGTH (3 CHARACTERS MAX)',/,
&      'TU = UNIT OF TIME (3 CHARACTERS MAX)'/)
       READ(5,3)Q,LC,KB,SC,LU,TU
     3 FORMAT(V)
       WRITE(6,4)
     4 FORMAT(1H0,'ARE THERE ANY ERRORS IN THE ABOVE ENTRIES?',/,
&      'IF NOT, ANSWER NO.')
       READ(5,3)CHECK
       IF(CHECK.NE.2HNO)GOTO100
       WRITE(6,10)
    10 FORMAT(1H0 ,'THE FOLLOWING PARAMETERS ARE USED IN THIS SOLUTION:')
       WRITE(6,11)Q,LC,KB,SC,LU,TU
    11 FORMAT(1H0,'Q=',F10.2,//,'LC=',F10.8,//,'KB=',F10.2,//,'SC=',F10.8,//
&      'LU =',A3,//,'TU =',A3,//)
       WRITE(6,7)
     7 FORMAT(1H0,'ENTER THE FOLLOWING DATA:',/,
&      'R,T',/,
&      'R = DISTANCE TO OBSERVATION WELL (L)',/,
&      'T = LENGTH OF PUMPING PERIOD (T)',/,
&      'TERMINATE BY  RETURNING BLANK FIELDS.')
   200 WRITE(6,13)
    13 FORMAT(1H0,'ENTER R,T')
```

```
      READ(5,3)R,T
      IF(R.EQ.0..OR.T.EQ.0.)GOTO300
      RB=R*LC
      U=(R*R*SC)/(4.*KB*T)
      WRITE(6,12) U,RB
   12 FORMAT(1H0,'U=',F10.6,1X,//,'R/B=',F10.5)
      S=(Q/(4.*PI*KB))*W(U,RB)
      WRITE(6,8)R,LU,S,LU,T,TU
    8 FORMAT(1H0,'THE DRAWDOWN',1X,F6.0,1X,A3,1X,/,
&     'FROM THE PUMPING WELL IS',1X,F10.5,1X,A3,1X,/,
&     'AFTER',1X,F10.2,1X,A3,1X,'OF PUMPING.',/)
      GOTO200
  300 WRITE(6,14)
   14 FORMAT(1H0,'YOU HAVE TERMINATED THE PROGRAM.',/,
&     'IS THIS CORRECT?')
      READ(5,3)CHECK
      IF(CHECK.NE.'YES')GOTO200
      STOP
      END
C       FILE WURB
```

APPENDIX IV. FILE WURB, A LIST OF EXPLICIT FUNCTIONS FOR SOLUTION OF W(U,RB).

```
C       FILE WURB
C       FILE WURB IS A LIST OF FUNCTIONS REQUIRED IN THE
C       SOLUTION OF THE LEAKY ARTESIAN WELL FUNCTION W(U,R/B).
        FUNCTION W(U,RB)
C       THIS FUNCTION DEFINES THE LEAKY ARTESIAN WELL FUNCTION.
C       THE THREE FORMS CORRESPOND TO THOSE OUTLINED IN
C       HANTUSH AND JACOB, 1955.
        COMMON LC,ITERAT,LCOUNT,KB,KBCOUNT
        REAL LC,KB
      IF(U.GE.1.0) GO TO 1000
      IF(U.LT.1.0.AND.(RB*RB).GT.U) GO TO 2000
      IF(U.LT.1.0.AND.(RB*RB).LE.U) GO TO 3000
 1000 W = SS(U,RB)
      GO TO 5000
 2000 W = (2*AK0(RB)-SS(U,RB))
      GO TO 5000
 3000 F1 = (RB*RB*.25/U)
       W=2*AK0(RB)-AI0(RB)*(EI(F1))+EXP(-F1)*
&        (0.5772+ALOG(U)+EI(U)-U+U*
&        ((AI0(RB)-1)/(RB*RB*.25))-U*U*SUM(U,RB))
 5000 RETURN
      END
      FUNCTION SS(U,RB)
C       THIS FUNCTION SOLVES THE INDEFINITE INTEGRAL OUTLINED
C       ON PAGE -- OF THE TEXT.   THE METHOD USED IS
C       LAGURRE INTEGRATION AS DISCUSSED IN A & S
C       (ABRAMOWITZ AND SEGUN, 19--), PAGE 923.
C       FOR AN EXPLANATION OF THE PARAMETER AA
C       IN THE CODE BELOW, SEE THE TEXT.
        COMMON LC,ITERAT,LCOUNT,KB,KBCOUNT
        REAL LC,KB
      DOUBLE PRECISION  Y(15), WF(15)
       DATA AA/1./
      DATA Y(1)/0.093307812017/
      DATA Y(2) /0.492691740302/
      DATA Y(3) /1.215595412071/
      DATA Y(4) /2.269949526204/
      DATA Y(5) /3.667622721751/
      DATA Y(6)/5.425336627414/
      DATA Y(7) /7.565916226613/
      DATA Y(8) /10.120228568019/
      DATA Y(9) /13.130282482176/
      DATA Y(10) /16.654407708330/
      DATA Y(11) /20.776478899449/
      DATA Y(12) /25.623894226729/
      DATA Y(13) /31.407519169754/
      DATA Y(14) /38.530683306486/
      DATA Y(15) /48.026085572686/
      DATA WF(1)   /0.239578170311/
      DATA WF(2)   /0.560100842793/
      DATA WF(3)   /0.887008262919/
```

```
      DATA WF(4)   /1.22366440215/
      DATA WF(5)   /1.57444872163/
      DATA WF(6)   /1.94475197653/
      DATA WF(7)   /2.34150205664/
      DATA WF(8)   /2.77404192683/
      DATA WF(9)   /3.25564334640/
      DATA WF(10)  /3.80631171423/
      DATA WF(11)  /4.45847775384/
      DATA WF(12)  /5.27001778443/
      DATA WF(13)  /6.35956346973/
      DATA WF(14)  /8.03178763212/
      DATA WF(15)  /11.5277721009/
          B=(RB)**2
          WU=0
      DO 30 I=1,15
      IF(U.LT.1.0) GO TO 31
          A=1/(U+Y(I)/AA)
      F= A*EXP(-(U+B*A*.25+Y(I)/AA))
        FEW=F
       IF(FEW .LE. 0.)GO TO 888
      GO TO 32
   31    UM=0.25*RB*RB*(1/U)
         A= 1/(UM+Y(I)/AA)
         F=A*EXP(-(UM+B*A*0.25+Y(I)/AA))
       FEW2=F
       IF(FEW2 .LE. 0) GO TO 888
   32     FW=F*WF(I)
          WU=WU+FW
   30 CONTINUE
          SS=(1/AA)*WU
       RETURN
 888   CONTINUE
       TESTB=EXP(-(B*A*.25+Y(I)/AA))
       TESTU=EXP(-U)
       TESTUM=EXP(-UM)
       IF(TESTB.LE.0.OR.U.LT.1.AND.TESTUM.LE.0)GO TO 222
       IF(U .GT. 1 .AND. TESTU .LE. 0)GO TO 223
 222   CONTINUE
       LC=LC*.05
       LCOUNT=LCOUNT+1
       RETURN
 223   CONTINUE
       KB=KB*10
       KBCOUNT=KBCOUNT+1
       RETURN
       END
       FUNCTION AKO(RB)
C      THIS FUNCTION SOLVES THE MODIFIED BESSELS FUNCTION
C      OF THE SECOND KIND, ZERO ORDER, A & S, PAGE 379.
      IF(RB.GT.2.)GO TO 7000
      TJ=(RB/2.)
      AKO   =-ALOG(TJ)*AI0(RB)-.57721566
&       +0.42278420*(TJ)**2+0.23069756*(TJ)**4
&       +0.03488590*(TJ)**6+0.00262698*(TJ)**8
```

```
&        +0.00010750*(TJ)**10+0.00000740*(TJ)**12
      GO TO 20000
 7000  TM=(2./RB)
      AK0=(1./SQRT(RB))*EXP(-RB)*(1.25331414
&        -0.07832358*TM+0.02189568*TM**2
&        -0.01062446*TM**3+0.00587872*TM**4
&        -0.00251540*TM**5+0.00053208*TM**6)
#20000 RETURN
      END
      FUNCTION AI0(RB)
C      THIS FUNCTION SOLVES THE MODIFIED BESSELS FUNCTION
C      OF THE FIRST KIND, ZERO ORDER, A & S, PAGE 378.
      TF=RB/3.75
      IF(RB.LE.3.75) GO TO 5000
      AI0=(1/SQRT(RB))*EXP(RB)*(.39894228
&        +.01328592*TF**(-1)+.00225319*TF**(-2)
&        -.00157565*TF**(-3)+.00916281*TF**(-4)
&        -.02057706*TF**(-5)+.02635537*TF**(-6)
&        -.01647633*TF**(-7)+.00392377*TF**(-8))
      GO TO 10000
 5000 AI0     = 1.0+3.5156229*TF**2+3.0899424*TF**4
&        +1.2067492*TF**6+0.2659732*TF**8
&        +0.0360768*TF**10+0.0045813*TF**12
#10000 RETURN
      END
C      FUNCTIONS SUM AND IFACT ARE SPECIAL FUNCTIONS
C      WHICH ARE USED TO SOLVE THE EQUATION BEGINNING
C      AT STATEMENT 3000 IN FUNCTION W.
      FUNCTION SUM(U,RB)
      DATA LIMIT/5/
      SUM=0.0
      RBF=RB*RB*0.25
      DO 1100 N=1,LIMIT
      DO 1200 M=1,N
      LF=(N-M+1)
      LH=(N+2)
      BFACT=IFACT(LH)*1.0
      AFACT=IFACT(LF)*1.0
      PSUM=((-1)**(N+M))*(AFACT/(BFACT*BFACT))*( RBF**M)*(U**(N-M))
      SUM=SUM+PSUM
      IF(PSUM.LT.1.0E-8) GO TO 1000
 1200 CONTINUE
 1100 CONTINUE
 1000 RETURN
      END
      FUNCTION IFACT(L)
      IFACT=1
      IF(L.EQ.1) GO TO 1301
      DO 1300 I=2,L
      MA=I
      IFACT=IFACT*MA
 1300 CONTINUE
 1301 RETURN
      END
```

```
      FUNCTION EI(U)
C     THIS FUNCTION SOLVES THE EXPONENTIAL INTEGRAL
C     DEFINED ON PAGE 231 OF A & S.
      IF(U.GT.1.0)GO TO 186
      EI=-ALOG(U)-.5772156+.99999139*U-.24991055*U*U
&       +.05519968*U**3-.0097004*U**4+.00107857*U**5
      GO TO 187
  186 EI=(EXP(-U)/U)*(U*U+2.334733*U+.250621)/
&       (U*U+3.330657*U+1.681534)
  187 RETURN
      END
```

Appendix V

List of Test Data Sources

1) Groundwater Resources Evaluation, Walton, W.C., 1970.

   (page 286, Problem 4.5)

2) Cooper, H.H., 1963, in Ground Water hydarulics, lohman, S.W., 1972, USGS Prof. Paper 708.

   (page 31, Table II)

3) Illinois State Water Supply, Bulletin 49, Walton W.C., 19 . Department of Registration and Education, Urbana (page 32, Table 5)

4) Aquifer test in the Ogallala Formation (26-37-21ddd), Gutentag, E.D., 1965, USGS, Garden City, Kansas.

   (data file)

5) Results of aquifer tests in the Wellington Aquifer near Salina, Kansas, Gillespie, J.B., 1979.

   (unpublished data)

6) Pump test data at Test Well #1 (in Groundwater Resources Investigation of the Spratt Site for Sunflower Electric Cooperative), 1977, Burns and McDonnel Consultants, Kansas City.

Appendix VI

Comments on Program Notation

The variables appearing in the matrix equation are defined in the program by the following notation:

T : KB U_T : UKB

S : SC U_S : USC

L : LC U_L : URC

$\sum_i U_L (Sg-S_e)$: SURCDS

$\sum_i U_S (Sg-S_e)$: SUSCDS

$\sum_i U_T (Sg-S_e)$: SUKBDS

$\sum_i U_L^2$ ← raise : SURC2

$\sum_i U_L U_S$: SUSCUR

$\sum_i U_L U_T$: SURCUK

$\sum_i U_S U_L$: SUSCUR

$\sum_i U_S^2$ ← raise : SUSC2

$\sum_i U_S U_T$: SUKBUS

$\sum_i U_T U_L$: SURCUK

$\sum_i U_T U_S$: SUKBUS

$\sum_i U_T^2$ ← raise : SUKB2