APPLICATION OF THE CONJUGATE-GRADIENT METHOD TO GROUND-WATER MODELS

By T. A. Manteuffel, D. B. Grove, and L. F. Konikow

U.S. GEOLOGICAL SURVEY

Water-Resources Investigations Report 83-4009

Denver, Colorado

1983

U.S. DEPARTMENT OF THE INTERIOR
March 3, 1849

UNITED STATES DEPARTMENT OF THE INTERIOR

JAMES G. WATT, Secretary

GEOLOGICAL SURVEY

Dallas L. Peck, Director

For additional information
write to:

D. B. Grove, Project Chief
Water Resources Division, Central Region
U.S. Geological Survey
Box 25046, Mail Stop 413
Denver Federal Center
Lakewood, Colorado 80225

Copies of this report can be
purchased from:

Open-File Services Section
Western Distribution Branch
U.S. Geological Survey
Box 25425, Federal Center
Lakewood, Colorado 80225
(Telephone: [303] 234-5888)

CONTENTS

ILLUSTRATIONS

TABLES

APPLICATION OF THE CONJUGATE-GRADIENT METHOD TO GROUND-WATER MODELS

By

T. A. Manteuffel, D. B. Grove, and L. F. Konikow

ABSTRACT

The conjugate-gradient method can solve efficiently and accurately finite-difference approximations to the ground-water flow equation. An aquifer-simulation model using the conjugate-gradient method was applied to a problem of ground-water flow in an alluvial aquifer at the Rocky Mountain Arsenal, Denver, Colorado. For this application, the accuracy and efficiency of the conjugate-gradient method compared favorably with other available methods for steady-state flow. However, its efficiency relative to other available methods depends on the nature of the specific problem. The main advantage of the conjugate-gradient method is that it does not require the use of iteration parameters, thereby eliminating this partly subjective procedure.

INTRODUCTION

Aquifer-simulation models commonly use finite-difference methods to solve the partial differential equation that describes ground-water flow. Several numerical procedures are available and commonly used to solve the system of finite-difference equations generated by approximating the flow equation. Trescott and others (1976) include the option of using LSOR (line successive overrelaxation), ADIP (alternating direction implicit procedure), or SIP (strongly implicit procedure) in their well-documented program for aquifer simulation. However, the efficiency of these numerical methods is sensitive to the choice of the iteration (or acceleration) parameters. The optimum values for these parameters depend on the problem and are not estimated easily. The method of conjugate gradients offers an alternative procedure for solving iteratively large systems of simultaneous linear equations (Beckman, 1960). This method requires no iteration parameters, and it is relatively easy to implement.

The purposes of this report are to: (1) Describe the derivation and application of the method, including the use of matrix scaling to increase the efficiency of the solution; (2) demonstrate the application of the method to a complex field problem; and (3) compare the results with other available methods.

GROUND-WATER FLOW EQUATION

The partial differential equation describing ground-water flow in a confined aquifer in two dimensions may be written as:

$$\frac{\partial}{\partial x}\left(T_{xx}\frac{\partial h}{\partial x}\right) + \frac{\partial}{\partial y}\left(T_{yy}\frac{\partial h}{\partial y}\right) = S\frac{\partial h}{\partial t} + W(x,y,t) \qquad (1)$$

where

x and y = the coordinate axes, L;

T_{xx} and T_{yy} = the components of transmissivity tensor, L^2/T;

h = the hydraulic head, L;

s = the storage coefficient (dimensionless);

t = the time, T; and

$W(x,y,t)$ = the volume flux per unit area (positive for outflow and negative for inflow), L/T.

Equation 1 assumes that the prinicipal components of the transmissivity tensor are alined with the coordinate axes x and y.

A standard approach to solving equation 1 is to approximate h at a discrete set of points arranged in a rectangular grid. The continuous derivatives then are replaced by finite-difference approximations.

Suppose we approximate h at N evenly spaced points (or nodes)

$$(x_i, y_j) \qquad i=1,\ldots,n; \quad j=1,\ldots,m$$

where N=nm. The grid spacing is given by

$$\Delta x = x_i - x_{i-1} \tag{2a}$$

$$\Delta y = y_i - y_{j-1} \tag{2b}$$

ordered from left to right and bottom to top. Letting $h_{ij}(t) = h(x_i, y_j, t)$, we can replace equation 1 by

$$\frac{1}{\Delta x}\left[T_{xx}(i+\tfrac{1}{2},j)\,\frac{h_{i+1,j} - h_{ij}}{\Delta x} - T_{xx}(i-\tfrac{1}{2},j)\,\frac{h_{ij} - h_{i-1,j}}{\Delta x}\right]$$

$$+ \frac{1}{\Delta y}\left[T_{yy}(i,j+\tfrac{1}{2})\,\frac{h_{i,j+1} - h_{ij}}{\Delta y} - T_{yy}(i,j-\tfrac{1}{2})\,\frac{h_{ij} - h_{i,j-1}}{\Delta y}\right]$$

$$= S_{ij}\frac{\partial h_{ij}}{\partial t} + W_{ij} \tag{3}$$

for each $i=1,\ldots,n$; $j=1,\ldots,m$.

If the transmissivity midway between nodes is computed to be the harmonic mean of the transmissivities at the adjacent nodes, we can write equation 3 as:

$$b_{ij}\,h_{i,j-1} + c_{ij}\,h_{i-1,j} - (b_{ij} + c_{ij} + c_{i+1,j} + b_{i,j+1})\,h_{ij}$$

$$+ c_{i+1,j}\,h_{i+1,j} + b_{i,j+1}\,h_{i,j+1} = S_{ij}\frac{\partial h_{ij}}{\partial t} + W_{ij} \tag{4}$$

where

$$b_{ij} = \frac{\left[\dfrac{2T_{yy(ij)}\ T_{yy(i,j-1)}}{T_{yy(ij)} + T_{yy(i,j-1)}}\right]}{\Delta y^2}$$

and

$$c_{ij} = \frac{\left[\dfrac{2T_{xx(ij)}\ T_{xx(i-1,j)}}{T_{xx(ij)} + T_{xx(i-1,j)}}\right]}{\Delta x^2} ,$$

for each $i=1,...,n$; $j=1...,m$. The use of the harmonic mean makes the appropriate coefficients equal to zero at no-flow boundaries.

If we discretize the time derivative by letting

$$\frac{\partial h_{ij}}{\partial t} = \frac{h_{ijk} - h_{ij,k-1}}{\Delta t_k} ,$$

where k is the index of the time discrete, and let

$$d_{ij} = (b_{ij} + c_{ij} + c_{i+1,j} + b_{i,,j+1}) ,$$

then equation 4 can be written:

$$b_{ij}\ h_{i,j-1,k} + c_{ij}\ h_{i-1,jk} - d_{ij}\ h_{ijk} + c_{i+1,j}\ h_{i+1,jk} + b_{i,j+1}\ h_{i,j+1,k} = \frac{S_{ij}}{\Delta t_k}(h_{ijk} - h_{ij,k-1}) + W_{ijk} \tag{5}$$

for $i=1...,n$; $j=1,...,m$. Equation 5 is an implicit equation with respect to the time discretization; that is, the hydraulic-head values on the left-hand side are at the new time step.

If we arrange the above equations in the same order as the finite-difference grid (left to right and bottom to top), the N equations represented by equation 5 can be conveniently written in matrix form. Let

$$\overline{h}_k = \begin{pmatrix} h_{11k} \\ h_{21k} \\ \vdots \\ h_{n1k} \\ h_{12k} \\ \vdots \\ h_{nmk} \end{pmatrix}_{\mathrm{Nx1}} , \qquad \overline{W}_k = \begin{pmatrix} W_{11k} \\ W_{21k} \\ \vdots \\ W_{n1k} \\ W_{12k} \\ \vdots \\ W_{nmk} \end{pmatrix}_{\mathrm{Nx1}} ,$$

and

$$S_k = \frac{1}{\Delta t_k} \begin{pmatrix} s_{11} & & & & & \\ & s_{21} & & & & \\ & & \ddots & & & \\ & & & s_{n1} & & \\ & & & & s_{12} & & \\ & & & & & \ddots & \\ & & & & & & s_{nm} \end{pmatrix}_{\mathrm{NxN}} ,$$

where the flux boundry conditions are incorporated in $\overline{W}_k$. Let

$$T = \begin{pmatrix} C_1 & B_2 & & & \\ B_2 & C_2 & B_3 & & \\ & B_3 & C_3 & \ddots & \\ & & \ddots & \ddots & B_m \\ & & & B_m & C_m \end{pmatrix}_{\mathrm{NxN}}$$

where the B's and C's are nxn blocks

$$C_j = \begin{pmatrix} d_{1j} & c_{2j} & & & \\ c_{2j} & d_{2j} & c_{3j} & & \\ & c_{3j} & \ddots & \ddots & \\ & & \ddots & \ddots & c_{nj} \\ & & & c_{nj} & d_{nj} \end{pmatrix}_{n\mathrm{x}n} ,$$

and

$$B_j = \begin{pmatrix} b_{1_j} & & & & \\ & b_{2_j} & & & \\ & & \ddots & & \\ & & & & b_{nj} \end{pmatrix}_{nxn}$$

Now we can write the equations represented by equation 5 as:

$$(T + S_k)\ \bar{h}_k = S_k \bar{h}_{k-1} - \bar{W}_k \tag{6}$$

where a bar over a symbol indicates a vector.

Computing the hydraulic head at the new time step is equivalent to solving a linear system of equations. The steady-state solution is found by either solving the system (equation 6) repeatedly or by setting

$$S_k \equiv 0$$

and solving the system. It is clear by its construction that T is a symmetric matrix. It can be shown easily that if $S_{ij} \geq 0$ for all ij, then the matrix

$$A_k = T + S_k$$

is positive definite; that is, it has positive real eigenvalues (Varga, 1962; Forsythe and Wasow, 1960).

METHOD OF CONJUGATE GRADIENTS

Basis for Application

Suppose we wish to solve the linear system

$$A\bar{h} = \bar{b} \tag{7}$$

where A is a positive definite matrix of dimension N, and $\bar{h}$ and $\bar{b}$ are vectors of length N. Suppose we have an initial guess $(\bar{h}_o)$ for the solution so that

$$\bar{e}_o = \bar{h} - \bar{h}_o$$

is the initial error. Suppose, further, that we have a set of mutually

A-orthogonal vectors, $\left\{\bar{p}_i\right\}_{i=1,n}$, such that $\bar{e}_o$ is a linear combination of the $\bar{p}$'s; that is, suppose

$$<A\bar{p}_i,\bar{p}_j> = 0 \qquad i \neq j \tag{8}$$

where $<\bar{x}, \bar{y}>$ denotes the inner product between the vectors $\bar{x}$ and $\bar{y}$, and

$$\bar{e}_o = \alpha_1\bar{p}_1 + \alpha_2\,\bar{p}_2 + . . . + \alpha_n\bar{p}_n\ . \tag{9}$$

therefore,

$$\bar{h} = \bar{h}_o + \alpha_1\bar{p}_1 + . . . + \alpha_n\bar{p}_n\ . \tag{10}$$

Because we have the set $\left\{\bar{p}_i\right\}_{i=1,n}$ all we need are the constants $\left\{\alpha_i\right\}_{i=1,n}$ in order to construct the solution $\bar{h}$. If we multiply both sides of equation 10 by A, we have

$$A\bar{h} = A\bar{h}_o + \alpha_1\,A\bar{p}_1 + . . . + \alpha_n\,A\bar{p}_n \tag{11}$$

Because the $\bar{p}$'s are A-orthogonal, if we take the inner product of both sides of equation 11 with the vector $\bar{p}_1$, we have

$$<A\bar{h}, \bar{p}_1> = <A\bar{h}_o, \bar{p}_1> + \alpha_1\,<A\bar{p}_1, \bar{p}_1>\ . \tag{12}$$

By letting

$$\bar{r}_o = \bar{b} - A\bar{h}_o$$

and using equation 7, we have

$$<\bar{r}_o, \bar{p}_1> = \alpha_1\,<A\bar{p}_1, \bar{p}_1>\ . \tag{13}$$

The property that A is positive definite guarantees that $<A\bar{y}, \bar{y}> >0$ for any vector $\bar{y} \neq 0$ (Varga, 1962) so we can write

$$\alpha_1 = \frac{<\bar{r}_o, \bar{p}_1>}{<A\bar{p}_1, \bar{p}_1>}\ .$$

By defining

$$\bar{h}_1 = \bar{h}_o + \alpha_1\,\bar{p}_1$$

we have from equation 10

$$\bar{h} = \bar{h}_1 + \alpha_2\bar{p}_2 + . . . + \alpha_n\bar{p}_n\ .$$

By repeating the above process, where

$$\bar{r}_1 = \bar{b} - A\bar{h}_1$$

we obtain

$$\alpha_2 = \frac{<\bar{r}_1, \bar{p}_2>}{<A\bar{p}_2, \bar{p}_2>} \quad .$$

Then, we let

$$\bar{h}_2 = \bar{h}_1 + \alpha_2\bar{p}_2$$

and so forth. In this manner, we construct the solution h in n steps. The task that remains is to construct a complete set of A-orthogonal vectors. In the algorithm outlined below, the vectors $\left\{ \bar{p}_i \right\}_{i=1,n}$ are constructed recursively from the sequence of residuals

$$\bar{r}_i = \bar{b} - A\bar{h}_i \tag{14}$$

The derivation may be found in Hestenes and Stiefel (1952).

Development of Algorithm

Given the system

$$A\bar{h} = \bar{b}$$

and the initial guess $\bar{h}_o$, let

$$\bar{r}_o = \bar{b} - A\bar{h}_o \ ,$$

$$\bar{p}_1 = \bar{r}_o \ ,$$

with general step

$$\alpha_i = \frac{<\bar{r}_{i-1}, \bar{p}_i>}{<A\bar{p}_i,\bar{p}_i>} \ , \tag{15a}$$

$$\bar{h}_i = \bar{h}_{i-1} + \alpha_i\bar{p}_i \ , \tag{15b}$$

$$\bar{r}_i = \bar{r}_{i-1} - \alpha_i A\bar{p}_i \qquad (= \bar{b} - A\bar{h}_i) \ , \tag{15c}$$

$$\beta_i = - \frac{<\bar{r}_i,A\bar{p}_i>}{<A\bar{p}_i,\bar{p}_i>} \ , \tag{15d}$$

$$\bar{p}_{i+1} = \bar{r}_i + \beta_i\bar{p}_i \ . \tag{15e}$$

In place of the expressions in equations 15a and 15b for α_i and β_i, we may use

$$\alpha_i = \frac{<\bar{r}_{i-1}, \bar{r}_{i-1}>}{<A\bar{p}_i, \bar{p}_i>} , \tag{16a}$$

and

$$\beta_i = \frac{<\bar{r}_i, \bar{r}_i>}{<\bar{r}_{i-1}, \bar{r}_{i-1}>} . \tag{16b}$$

These expressions for α_i and β_i are equivalent because of the relations that hold among the various vectors:

$$\begin{aligned} <\bar{r}_i, \bar{r}_j> &= 0 \qquad i \neq j , \\ <A\bar{p}_i, \bar{p}_j> &= 0 \qquad i \neq j , \\ <\bar{p}_i, \bar{r}_j> &= 0 \qquad i < j\text{-}1 , \\ <\bar{r}_i, A p_j> &= 0 \qquad i \neq j , j\text{-}1 . \end{aligned}$$

The equations 16a and 16b for α_i and β_i require fewer calculations than do equations 15a and 15d (Hestenes and Stiefel, 1952). It can be shown that the sequence $\left\{ \bar{h}_i \right\}$ generated in this manner will converge to the solution $\bar{h}$ in fewer or equal to N steps (Hestenes and Stiefel, 1952). In fact, it can be shown that this method is theoretically faster than any polynomial-based method (Schwarz and others, 1973). However, in practice orthogonality deteriorates due to round-off errors, and the theoretical rate of convergence seldom is realized.

Scaling of the System

The presence of non-zero elements in the matrix A that vary significantly in magnitude can cause orthogonality to break down quickly. Therefore, it may be necessary that the system be scaled whenever the conjugate-gradient method is used.

Scaling is accomplished by pre- and post-multiplication of the matrix A by diagonal matrices. Suppose P and Q are diagonal matrices, then the system

$$A\overline{h} = \overline{b} \tag{17}$$

can be written

$$P\ A\ Q\ Q^{-1}\ \overline{h} = P\overline{b}\ . \tag{18}$$

If we let

$$S = P\ A\ Q\ , \tag{19a}$$

$$\overline{y} = Q^{-1}\overline{h}\ , \tag{19b}$$

$$\overline{g} = P\overline{b}\ , \tag{19c}$$

then we can write

$$S\ \overline{y} = \overline{g}\ . \tag{20}$$

The solution to the original system (equation 17) can be found from the solution of the scaled system (equation 20) as

$$\overline{h} = Q\overline{y}\ . \tag{21}$$

The scaled system may be much easier to solve than the original system. The rate of convergence of most iterative methods is affected by the condition number of the matrix; that is, the ratio of the largest eigenvalue to the smallest eigenvalue. With the conjugate-gradient method, the condition number directly affects the breakdown in orthogonality (Hestenes and Stiefel, 1952). Scaling may be used to decrease the condition number. The condition number of the matrix S of equation 20 may be much smaller than the condition number of the matrix A of equation 17.

If the matrix A is a symmetric five-point difference matrix, as is the matrix of equation 6, then the smallest condition number is achieved by using the scaling where

$$P = Q = D \tag{22}$$

and

$$D = \begin{pmatrix} \frac{1}{\sqrt{a_{11}}} & & \\ & \ddots & \\ & & \frac{1}{\sqrt{a_{nn}}} \end{pmatrix} \tag{23}$$

(Forsythe and Strauss, 1955). (Here a_{ii} are the diagonal elements of A.) Equation 19a becomes

$$S = D\,A\,D\ , \tag{24}$$

where

$$S_{ij} = \frac{a_{ij}}{\sqrt{a_{ii}a_{jj}}} \qquad i=1,\ .\ .\ .,\ N\ ; \quad j=1,\ .\ .\ .,\ N\ ,$$

and equation 19c becomes

$$\overline{g} = D\overline{b} \tag{25}$$

where

$$g_i = \frac{b_i}{\sqrt{a_{ii}}} \qquad i=1,\ .\ .\ .,\ N\ .$$

The solution of equation 17 can be found from the solution of equation 20 as

$$\overline{h} = D\overline{y} \tag{26}$$

or

$$h_i = \frac{y_i}{\sqrt{a_{ii}}} \qquad i=1,\ .\ .\ .,\ N\ .$$

It follows from equation 24 that all the diagonal elements of the scaled matrix S will equal 1. This will decrease the number of multiplications required to perform matrix multiplications.

APPLICATION TO FIELD PROBLEM

Selection of Study Area

The applicability of the conjugate-gradient method to ground-water problems can be demonstrated through the solution of a complex field problem. The study area selected for this demonstration is located in and adjacent to the Rocky Mountain Arsenal near Denver, Colorado (fig. 1). This area was selected for an evaluation of the numerical model because: (1) Detailed hydrogeologic data were available from a previous study (Konikow, 1974), (2) a flow model that uses an alternative but well-documented numerical method

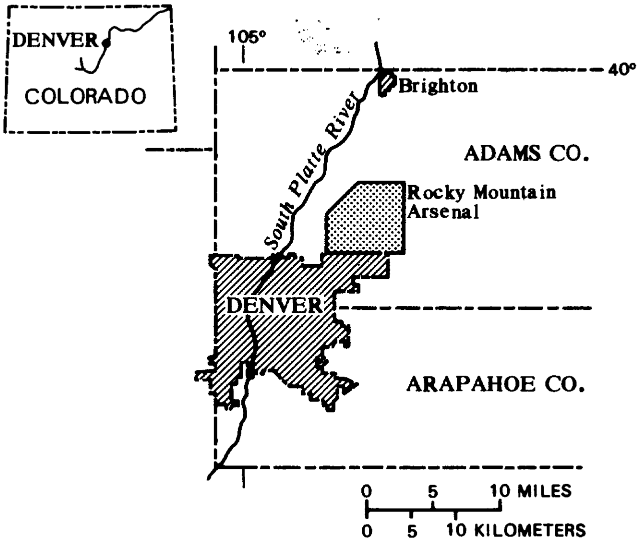

Figure 1.--Location of study area.

had been calibrated previously for this flow system (Konikow, 1976), and (3) the flow system is typical of many field problems but sufficiently complex to offer a rigorous test of the general applicability of the model to field problems.

Description of Study Area

Shallow alluvium forms the primary aquifer in this area. The hydrogeologic characteristics of the alluvium indicate that this aquifer is sloping, discontinuous, heterogeneous, and nonuniform in thickness (Konikow, 1974).

The major hydrologic features in the study area are shown in figure 2. Much of the area north of the arsenal is irrigated, both with surface water diverted from one of the irrigation canals, which are unlined, and with ground water pumped from irrigation wells. Infiltration from irrigated fields is the main source of recharge to the alluvial aquifer. However, leakage (or seepage losses) from the unlined irrigation canals, unlined disposal ponds, and freshwater reservoirs also are significant. Most ground-water outflow occurs as seepage into the South Platte River, withdrawals by irrigation wells, and underflow through the aquifer out of the study area.

A map showing the general water-table configuration for 1943-56 is presented in figure 3. The assumptions and limitations of figure 3 are discussed in detail by Konikow (1974). Note that approximately 20 percent of the study area is underlain by areas in which the alluvium either is absent or unsaturated most of the time. These areas form internal barriers that significantly affect ground-water flow patterns within the aquifer.

Calibration of Model

The limits of the modeled area were selected to coincide as closely as possible with natural boundaries and divides in the ground-water flow system. The modeled area was subdivided into a finite-difference grid of uniformly spaced squares having 305 meters between nodes.

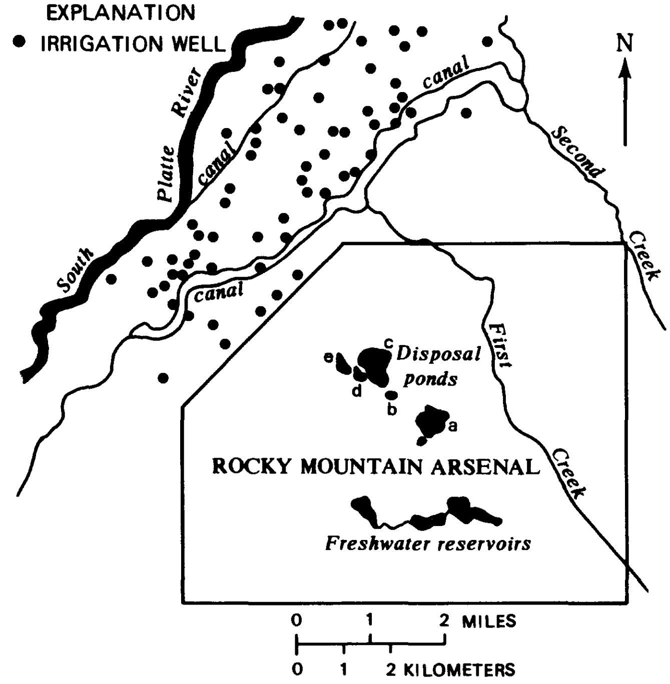

Figure 2.--Major hydrologic features.

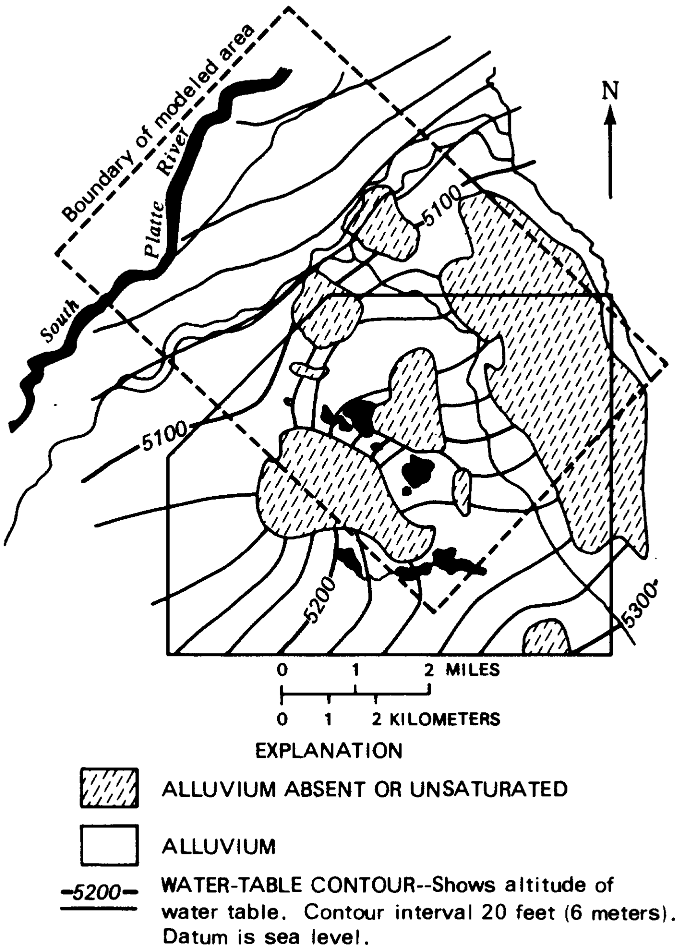
Boundary of modeled area
South Platte River
N
5100
5100
5200
5300
0 1 2 MILES
0 1 2 KILOMETERS
EXPLANATION
ALLUVIUM ABSENT OR UNSATURATED
ALLUVIUM
5200
WATER-TABLE CONTOUR--Shows altitude of water table. Contour interval 20 feet (6 meters). Datum is sea level.

Figure 3.--General water-table configuration in the alluvial aquifer in and adjacent to the Rocky Mountain Arsenal, 1943-56.

All aquifer properties and stresses were defined at each node of the grid. The boundaries were represented as either no-flow or constant-head conditions, as appropriate. The values specified for aquifer properties and stresses were the same as those specified for the previously calibrated model (Konikow, 1976).

Insufficient field data were available to calibrate accurately the transient-flow model. Hence, a steady-state model was calibrated to minimize the differences between the measured and computed water-table altitudes.

Implementation

The conjugate-gradient method described here was used in conjunction with the aquifer-simulation model documented by Trescott and others (1976). The algorithm for the numerical solution using the conjugate-gradient method was programmed into several subroutines and substituted for solution subroutines contained in the Trescott model.

The symmetric matrix A, requiring (N,3) words of storage, and the target vector $\bar{b}$, requiring N words of storage, were constructed from vectors obtained from the Trescott model. As described earlier, scaling of the system was accomplished, and the diagonal scaling matrix D was stored in the diagonal of the scaled matrix A, which was known to have a value of 1.0.

The subroutine CG, which appears in the Supplemental Information section of this report, was used to make the conjugate-gradient iteration. This subroutine is self-contained and only requires a matrix-vector multiplication subroutine, MATMUL, which is specific to the storage scheme of the matrix. The subroutine in this implementation, which also appears in the Supplemental Information section, used the fact that the diagonal of the scaled system had a value of 1.0.

Results

The solution computed by the model using the conjugate-gradient method was very accurate numerically. The numerical error was measured in terms of the residual, r_i,

$$\overline{r}_i = \overline{b} - A\overline{h}_i \ , \tag{27}$$

and of the ℓ_2-norm (or Euclidean norm)

$$\ell_2\text{-norm} = \sqrt{\sum r_i^2} \ , \tag{28}$$

for the scaled and the unscaled system. The max-norm also was computed and differed uniformly from the ℓ_2-norm by a factor of less than 10. The computed errors in the mass balance using tolerance criteria of a scaled system for the residual error are shown in table 1. These results indicate that a tolerance of 10^{-6} is acceptable as a bound on the ℓ_2-norm residual of the scaled system.

Decreasing the tolerance on the ℓ_2-norm of the residual to decrease the mass-balance error increases the number of iterations required to converge to a solution. This relationship of the scaled system, for the steady-state problem described in this report, is shown in figure 4. Each iteration requires $7(m-1)(n-1)$ multiplications and an equal number of additions. For the steady-state flow problem at the Rocky Mountain Arsenal, the conjugate-gradient method required 35 iterations to assure an acceptable mass-balance error. A comparison, with available codes using ADIP and SIP to solve the equations resulting from the finite-difference approximation, was feasible only for unscaled systems. Although written by different people, this probably is reasonable, providing that a similar basis of comparison is available. In this instance, we have compared the ℓ_2-norm for an unscaled system versus the work measured in number of conjugate-gradient iterations. The latter also could be expressed in terms of CPU time. These data are plotted in figure 4. Data points for the CG and SIP algorithms are joined by lines indicating a steady decrease in error with additional expended work. Data points for ADIP do not necessarily show this trend for a short work cycle. However, for a longer cycle, a similar downward trend in error with expended work is evident. This cyclic nature of the alternating-direction

Table 1.--*Mass-balance error in steady-state solution using conjugate-gradient method*

ℓ_2-norm of residual error for scaled system	Mass-balance error (percent)
10^{-5}	0.143
10^{-6}	$-.29 \times 10^{-2}$
10^{-8}	$-.164 \times 10^{-3}$
10^{-12}	$.119 \times 10^{-4}$

method is considered normal. For this particular problem, the SIP is shown to be clearly superior with CG, and ADIP is quite similar depending on the error criteria chosen.

The model computed a steady-state water-table configuration that was acceptably similar to the configuration based on water-level measurements. However, with respect to an evaluation of the numerical method, it is more significant that the solution using the conjugate-gradient method was identical (to at least five significant figures) with the solution obtained using ADIP and SIP.

SUMMARY AND CONCLUSION

The conjugate-gradient method can solve efficiently and accurately the ground-water flow equation for steady-state problems. When applied to a typical problem of ground-water flow in an alluvial aquifer, the method compared favorably with the alternating direction implicit procedure and less satisfactorily with the strongly implicit procedure. The main advantage of

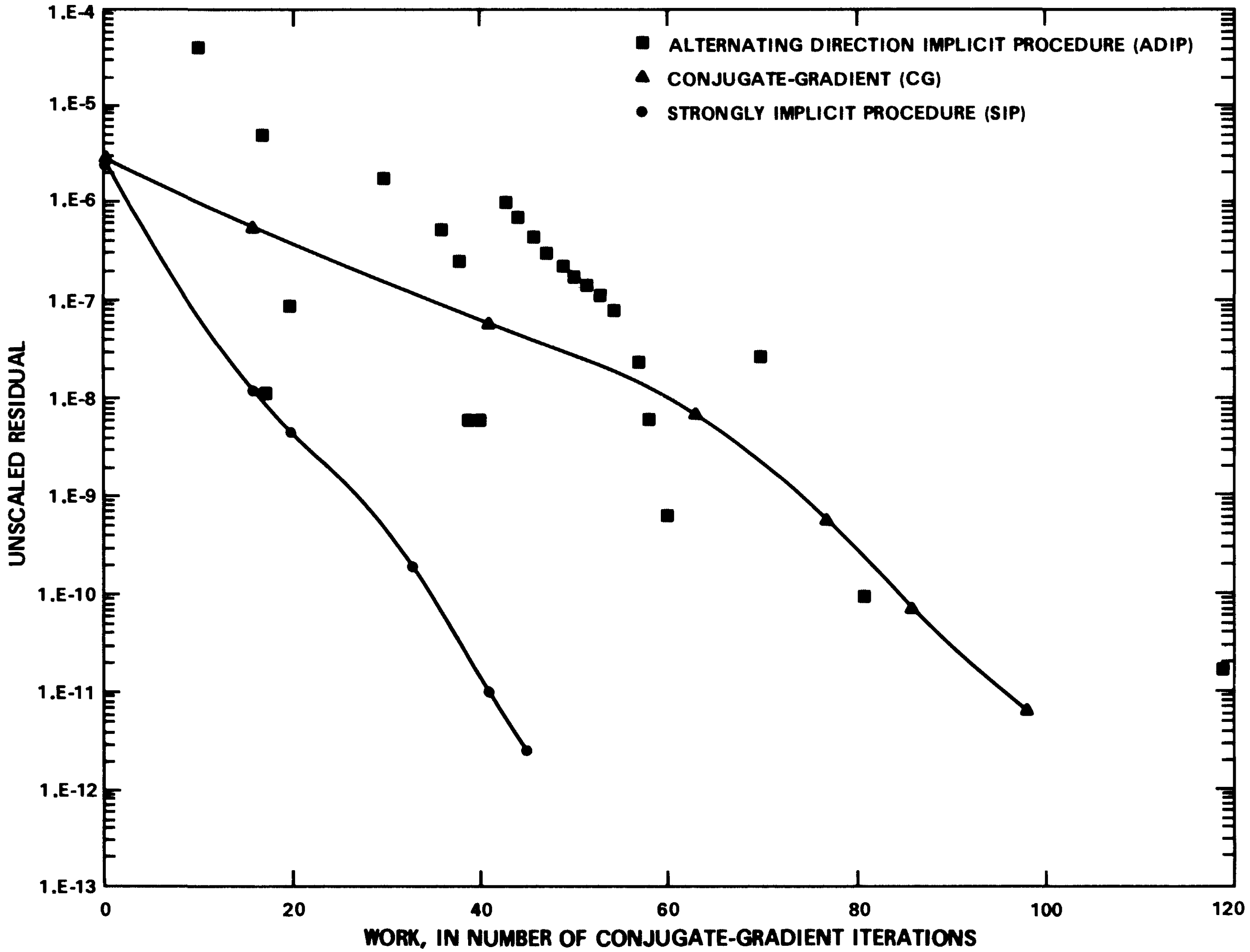

Figure 4.--Relationship between unscaled residual error and work, in number of conjugate-gradient iterations, for several matrix-solution methods.

the conjugate-gradient method is that it does not require the use of iteration parameters, therby eliminating this partly subjective procedure. The authors are aware that there have been recent theoretical advances in the use of this method that are not commented upon in this paper (Manteuffel, 1980). However, the use of concepts presented herein can be used to help explain conjugate-gradient application and to stimulate further study by interested readers.

REFERENCES CITED

Beckman, F. S., 1960, The solution of linear equations by the conjugate gradient method, *in* Ralston, A., and Wilf, H. S., eds., Mathematical methods for digital computers: New York, John Wiley, p. 62-72.

Forsythe, G. E., and Strauss, E. G., 1955, On best conditioned matrices: American Mathematical Society Proceedings, v. 6, no. 3, p. 340-345.

Forsythe, G. E., and Wasow, W. R., 1960, Finite-difference methods for partial differential equations: New York, John Wiley, 444 p.

Hestenes, M. R., and Stiefel, E., 1952, Methods of conjugate gradients for solving linear systems: National Bureau of Standards Journal of Research, v. 49, no. 6, p. 409-436.

Konikow, L. F., 1974, Hydrogeologic maps of the alluvial aquifer in and adjacent to the Rocky Mountain Arsenal, Colorado: U.S. Geological Survey Open-File Report 74-342, 17 p.

______ 1976, Modeling solute transport in ground water: International Conference on Environmental Sensing and Assessment, Las Vegas, Nev., 1976, Article 20-3, 6 p.

Manteuffel, T. A., 1980, An incomplete factorization technique for positive definite linear systems: Journal of Mathematics and Computations, v. 34, no. 150, p. 473-497.

Schwarz, H. R., Rutishauser, H., and Stiefel, E., 1973, Numerical analysis of symmetric matrices: Englewood Cliffs, N. J., Prentice Hall, Inc., 276 p.

Trescott, P. 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.

Varga, R. S., 1962, Matrix iterative analysis: Englewood Cliffs, N. J., Prentice Hall, 322 p.

SUPPLEMENTAL INFORMATION

LISTING OF FORTRAN SUBROUTINES FOR CONJUGATE-GRADIENT PROCEDURE

SUBROUTINE CG

```
      SUBROUTINE CG(A,X,R,P,AP,NX,NY,ITMAX,ERBND)
C*******************************************************************
C                 PROGRAMMED BY TOM MANTEUFFEL                     *
C  THIS SUBROUTINE SOLVES THE SYSTEM: A*X = P, RETURNING THE       *
C  SOLUTION IN X. THE TARGET VECTOR P IS DESTROYED DURING          *
C  THE COMPUTATION.                                                *
C*******************************************************************
      DIMENSION A(NX,NY,3),X(1),R(1),P(1),AP(1)
      DOUBLE PRECISION SUM
C
C     SET DIMENSION
          N = NX*NY
C     INITIALIZE ISTEP
          ISTEP = 0
C     COMPUTE THE FIRST RESIDUAL AND RESIDUAL ERROR
          CALL MATMUL(A,X,R,NX,NY)
          SUM = 0.0
          DO 10 I=1,N
              R(I)=P(I)-R(I)
              SUM = SUM+R(I)*R(I)
   10     CONTINUE
          RSD2 = SUM
          RSD=SQRT(RSD2)
          WRITE(6,300) ISTEP,RSD
C     INITIALIZE P
          DO 15 I=1,N
              P(I)=R(I)
   15     CONTINUE
C     MAIN LOOP
C         UPDATE ISTEP
   20         ISTEP=ISTEP+1
C         COMPUTE NEW X
              CALL MATMUL(A,P,AP,NX,NY)
              SUM = 0.0
              DO 25 I=1,N
                  SUM = SUM+P(I)*AP(I)
   25         CONTINUE
              PAP = SUM
              ALPHA=RSD2/PAP
              DO 30 I=1,N
                  X(I)=X(I)+ALPHA*P(I)
   30         CONTINUE
```

```
C           COMPUTE NEW RESIDUAL AND RESIDUAL ERROR
                SUM = 0.0
                DO 35 I=1,N
                    R(I)=R(I)-ALPHA*AP(I)
                    SUM = SUM+R(I)*R(I)
   35           CONTINUE
                ORSD2=RSD2
                RSD2 = SUM
                RSD=SQRT(RSD2)
            WRITE(6,300) ISTEP,RSD
C           TEST TO HALT
                IF(RSD.LT.ERBND) GO TO 55
                IF(ISTEP.GT.ITMAX) GO TO 40
                    GO TO 45
C               THEN PRINT WARNING
   40               WRITE(6,200)
                    GO TO 55
C               ELSE CONTINUE
C           COMPUTE NEW P




   45           BETA=RSD2/ORSD2
                DO 50 I=1,N
                    P(I)=R(I)+BETA*P(I)
   50           CONTINUE
C           REPEAT LOOP
                GO TO 20
C       END OF LOOP
C
C       OUTPUT FINAL RESIDUAL ERROR
  55        WRITE(6,100) ISTEP,RSD
C
  100   FORMAT(3X,'AFTER',I5,' ITERATIONS THE RESIDUAL ERROR IS ',
     1          E12.5)
  200   FORMAT('   WARNING: ITMAX EXCEEDED.')
  300   FORMAT(3X,'ISTEP=',I3,3X,'RSD=',E12.5)
C
        RETURN
        END
```

LISTING OF FORTRAN SUBROUTINES FOR CONJUGATE-GRADIENT PROCEDURE--Continued

SUBROUTINE MATMUL

```
        SUBROUTINE MATMUL(A,X,Y,NX,NY)
C****************************************************************
C  THIS SUBROUTINE DOES THE MATRIX VECTOR MULTIPLICATION:        *
C  A*X = Y, WHERE A IS A FIVE POINT DIFFERENCE MATRIX,WITH       *
C  DIAGONAL ELEMENTS EQUAL TO 1.0.                               *
C****************************************************************
        DIMENSION A(NX,NY,3),X(NX,NY),Y(NX,NY)
C
C       SET DIMENSION PARAMETERS
            NXM1 = NX - 1
            NYM1 = NY - 1
C
        J=1
            I=1
                Y(I,J)=X(I,J)+A(I,J,2)*X(I+1,J)
     1                  +A(I,J,3)*X(I,J+1)
            DO 10 I=1,NXM1
                Y(I,J)=X(I,J)+A(I,J,2)*X(I+1,J)
     1                  +A(I,J,3)*X(I,J+1)+A(I-1,J,2)*X(I-1,J)
   10       CONTINUE
            I=NX
                Y(I,J)=X(I,J)+A(I,J,3)*X(I,J+1)
     1                  +A(I-1,J,2)*X(I-1,J)
        DO 30 J=2,NYM1
            I=1
                Y(I,J)=X(I,J)+A(I,J,2)*X(I+1,J)
     1                  +A(I,J,3)*X(I,J+1)+A(I,J-1,3)*X(I,J-1)
            DO 20 I=2,NXM1
                Y(I,J)=X(I,J)+A(I,J,2)*X(I+1,J)
     1                  +A(I,J,3)*X(I,J+1)+A(I-1,J,2)*X(I-1,J)
     2                  +A(I,J-1,3)*X(I,J-1)
   20       CONTINUE
            I=NX
                Y(I,J)=X(I,J)+A(I,J,3)*X(I,J+1)
     1                  +A(I-1,J,2)*X(I-1,J)+A(I,J-1,3)*X(I,J-1)
   30   CONTINUE

        J=NY
            I=1
                Y(I,J)=X(I,J)+A(I,J,2)*X(I+1,J)
     1                  +A(I,J-1,3)*X(I,J-1)
            DO 40 I=2,NXM1
                Y(I,J)=X(I,J)+A(I,J,2)*X(I+1,J)
     1                  +A(I-1,J,2)*X(I-1,J)+A(I,J-1,3)*X(I,J-1)
   40       CONTINUE
            I=NX
                Y(I,J)=X(I,J)+A(I-1,J,2)*X(I-1,J)
     1                  +A(I,J-1,3)*X(I,J-1)
C
        RETURN
        END
```