Aquifer Simulation Model for Use on Disk Supported Small Computer Systems

by T. A. PRICKETT and C. G. LONNQUIST

CONTENTS

AQUIFER SIMULATION MODEL FOR USE
ON DISK SUPPORTED SMALL COMPUTER SYSTEMS

by T. A. Prickett and C. G. Lonnquist

ABSTRACT

A generalized digital computer program listing is given that can simulate two-dimensional flow of groundwater in heterogeneous aquifers under nonleaky and/or leaky artesian conditions. A special feature of the program is that a large simulation (up to an estimated 10,000 node problem) can be accomplished on a disk supported small computer with only 4000 words (32 bits per word) of core storage.

The computer program can handle time varying pumpage from wells, natural or artificial recharge rates, the relationships of water exchange between surface waters and the groundwater reservoir, the process of groundwater evapotranspiration, and the mechanism of flow from springs.

A finite difference approach is used to formulate the equations of groundwater flow. A modified alternating direction implicit method is used to solve the set of resulting difference equations.

The discussion of the digital technique includes the necessary mathematical background, the documented program listing, sample computer input data, and explanations of job setup procedures. The program is written in FORTRAN and will operate with any consistent set of units.

INTRODUCTION

The use of digital computers in groundwater resource evaluation has been firmly established for several years. Computers are available that allow solution of large sets of simultaneous equations that are involved in studying cause and effect relationships in heterogeneous aquifer systems with a wide variety of source

and sink functions and boundary conditions. In addition, listings of computer programs are readily available for solution of groundwater problems (see for example Tyson and Weber, 1964; Prickett and Lonnquist, 1968a, 1968b, and 1971; and Pinder, 1970). However, these programs require a relatively large computer in terms of core storage (in excess of 64,000 words for a typical regional problem).

When a groundwater hydrologist needs a computer analysis, his first problem is to make arrangements for the use of computer facilities. In general, large computers are available through arrangements made with universities and commercial data processing firms. On the other hand, there are substantial numbers of small computers (less than 32,000 words of core storage) to be found in local private firms, universities, and state agencies. Quite often, the small computer facilities are cheap to rent and have time available, and thus present another possibility for getting the job processed.

The purpose of this report is to present a computer program for solving a large groundwater problem on a small computer and thus eliminate the absolute need of a large computer. This is accomplished by writing a program (which requires only a small amount of core storage) to be used on a disk supported small computer. The disk unit serves as the main storage device as opposed to core storage on large computers.

A generalized computer program is presented that will simulate two-dimensional nonsteady-state flow problems in heterogeneous aquifers under nonleaky and/or leaky artesian conditions. This program includes time varying pumpage from wells, natural or artificial recharge rates, the relationships of water exchange between surface waters and the groundwater reservoir, the process of groundwater evapotranspiration, and the mechanism of flow from springs. The program is written to operate with any consistent set of units.

A finite difference approach that is identical to that taken by Prickett and Lonnquist (1971) has been adopted in formulating the equations of groundwater flow. A brief mathematical background is given that describes the modified iterative alternating direction implicit method of solving the equations. This is followed by the documented program listing, sample computer input data, and an explanation of the job setup procedures.

2

Parts of this report are abstracted from Bulletin 55, written by Prickett and Lonnquist (1971), since the mathematics and job setup are basically the same. For a full understanding of the validity of the method described here, the reader should refer to the more detailed explanation in Bulletin 55.

The computer program in this report is written in FORTRAN for use on an IBM 1130 disk supported computer with only 4000 words (32 bits per word) of core storage. However, the program will operate, with modifications, on other computers such as a CDC model SC 1700, Datacraft model 602V5, Digital Equipment Corporation model PDP-11/10, Microdata Corporation model 810, and Varian Data Machines model 620/L-100.

The program operates most efficiently on computers which have between 4000 and 32,000 words of core storage. Beyond this range other programs should be considered to take advantage of additional core storage.

Acknowledgments. This report has been prepared under the general supervision of Dr. William C. Ackermann, Chief of the Illinois State Water Survey, H. F. Smith, Head of the Hydrology Section, and R. J. Schicht, Engineer in charge of groundwater research. D. J. Beaver of the Hydrology Section reviewed and criticized the material and made useful suggestions. Many individuals made helpful comments on the computer programming and spent considerable time running checkout programs on their own computers. Special thanks are due R. W. Davis, Department of Geology, Southern Illinois University; M. K. Householder, Department of Civil Engineering, Youngstown State University; T. G. Gebhard, Jr., Department of Civil Engineering, New Mexico State University; and L. W. Gelhar, Department of Civil Engineering, Massachusetts Institute of Technology. Thanks are also due Mrs. J. Loreena Ivens for final editing, and J. W. Brother, Jr., for preparing the illustrations.

BRIEF MATHEMATICAL BACKGROUND

The partial differential equation (Jacob, 1950) governing the nonsteady-state, two-dimensional flow of groundwater may be expressed as

$$\partial/\partial x(T_x \partial h/\partial x) + \partial/\partial y(T_y \partial h/\partial y) = S\partial h/\partial t + Q \qquad (1)$$

where

T_x = aquifer transmissivity in the x direction

T_y = aquifer transmissivity in the y direction

S = aquifer storage coefficient

h = head

t = time

Q = source or sink functions expressed as net flow rates per unit area

A numerical solution of equation 1 can be obtained through a finite difference approach. The finite difference approach first involves replacing the continuous aquifer system parameters with an equivalent set of discrete elements. Secondly, the equations governing the flow of groundwater in the discretized model are written in finite difference form. Finally, the resulting set of finite difference equations is solved numerically with the aid of a digital computer.

A finite difference grid is superposed over a map of an aquifer as illustrated in figure 1. The aquifer is thus subdivided into volumes having dimensions $m\Delta x\Delta y$, where m is the thickness of the aquifer. The differentials ∂x and ∂y are approximated by the finite lengths Δx and Δy, respectively. The area $\Delta x\Delta y$ should be small compared with the total area of the aquifer to the extent that the discrete model is a reasonable representation of the continuous system. The intersections of grid lines are called nodes and are referenced with a column *(i)* and row (*j*) coordinate system colinear with the x and y directions, respectively.
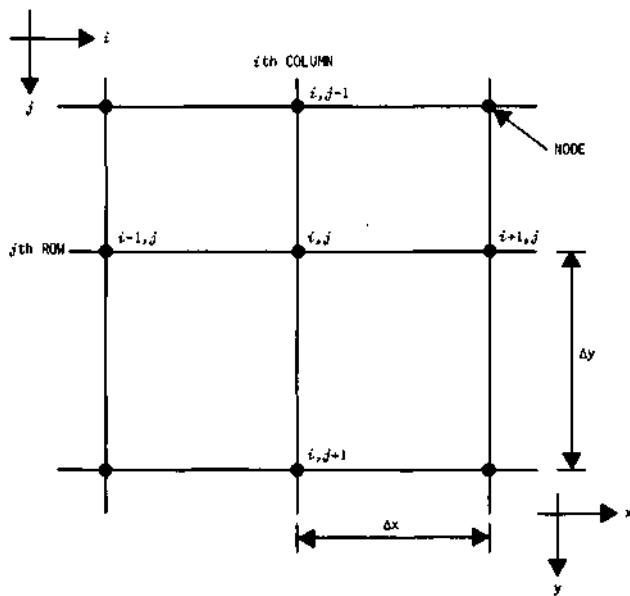
Figure 1. Finite difference grid

The general form of the finite difference equation governing the flow of groundwater in the discretized model is then given by

4

$$T_{i-1,j,2}(h_{i-1,j}-h_{i,j})/\Delta x^2 + T_{i,j,2}(h_{i+1,j}-h_{i,j})/\Delta x^2$$
$$+T_{i,j,1}(h_{i,j+1}-h_{i,j})/\Delta y^2 + T_{i,j-1,1}(h_{i,j-1}-h_{i,j})/\Delta y^2$$
$$= S_{i,j}(h_{i,j}-h\phi_{i,j})/\Delta t + Q_{i,j}/\Delta x\Delta y \qquad (2)$$

where

$T_{i,j,1}$ = aquifer transmissivity between nodes i,j and $i,j+1$

$T_{i,j,2}$ = aquifer transmissivity between nodes i,j and $i+1,j$

$h_{i,j}$ = calculated heads at nodes i,j at the end of a time increment measured from an arbitrary reference level

$h\phi_{i,j}$ = calculated heads at nodes i,j at the end of the previous time increment measured from the same reference level defining $h_{i,j}$

Δt = time increment elapsed since last calculation of heads

$S_{i,j}$ = aquifer storage coefficient at node i,j

$Q_{i,j}$ = net withdrawal rate if positive, or net accretion rate if negative at node i,j

Since there is an equation of the same form as equation 2 for every node of the digital model, a large set of simultaneous equations must be solved for the principal unknowns $h_{i, j}$. A modified form of the iterative alternating direction implicit method given by Prickett and Lonnquist (1971) is used to solve the set of simultaneous equations.

Modified Iterative Alternating Direction Implicit Method (MIADI)

Briefly, the MIADI method involves first, for a given time increment, reducing the large set of simultaneous equations down to a number of small sets. This is done by solving the node equations, by Gauss elimination, of an individual row of the model while all terms related to the nodes in the two adjacent row6 are held constant. After all row equations have been processed row by row, attention is focused on solving the node equations, again by Gauss elimination, of an individual column while all terms related to the two adjacent columns are held constant. Finally, after all equations have been solved column by column, an 'iteration' has been completed. The above process is repeated a sufficient number of times to

achieve convergence, and this completes the calculations for the given time incre-
ment. The calculated heads are then used as initial conditions for the next time
increment. This total process is repeated for successive time increments until the
desired simulation is completed.

Equation 2 may be rewritten to illustrate the general form for calculations by
rows. As a first simplification it is assumed that the finite difference grid is
made up of squares such that $\Delta y = \Delta x$. (The case where Δy does not equal Δx can be
treated by methods outlined by Prickett and Lonnquist, 1971.) Equation 2 is then
expanded, the signs reversed, and terms of $h_{i,j}$ grouped together to yield

$$
-T_{i-1,j,2}h_{i-1,j} + h_{i,j}(T_{i-1,j,2} + T_{i,j,2} + T_{i,j,1} + T_{i,j-1,1}
$$
$$
+ S_{i,j}\Delta x^2/\Delta t) - T_{i,j,2}h_{i+1,j} = (S_{i,j}\Delta x^2/\Delta t)h\phi_{i,j}
$$
$$
+ T_{i,j-1,1}h_{i,j-1} + T_{i,j,1}h_{i,j+1} - Q_{i,j} \tag{3}
$$

Equation 3 is of the form

$$
AAh_{i-1,j} + BBh_{i,j} + CCh_{i+1,j} = DD \tag{3'}
$$

where

$$
AA = -T_{i-1,j,2} \tag{3a}
$$
$$
BB = T_{i-1,j,2} + T_{i,j,2} + T_{i,j,1} + T_{i,j-1,1} + S_{i,j}\Delta x^2/\Delta t \tag{3b}
$$
$$
CC = -T_{i,j,2} \tag{3c}
$$
$$
DD = (S_{i,j}\Delta x^2/\Delta t)h\phi_{i,j} + T_{i,j-1,1}h_{i,j-1} + T_{i,j,1}h_{i,j+1} - Q_{i,j} \tag{3d}
$$

A similar set of equations can be written for calculations by *columns.*

The key to the manner in which the small storage computer is used lies in the
fact that data from only three rows or columns are needed in core at any one time
to solve the above mentioned equations. The logic of the following computer pro-
gram is such that data for all equations of the model are stored on a disk. Then
special instructions are used to transfer appropriate data between the disk and
core storage for individual row or column calculations as the simulation proceeds.
An explanation of the mechanics of the transfers between disk and core storage is
included in a later section.

```
C   ILLINOIS STATE WATER SURVEY
C   AQUIFER SIMULATION MODEL FOR DISK
C   SUPPORTED COMPUTERS WITH LESS
C   THAN 8K WORDS OF CORE STORAGE
C
C
C   DEFINITION OF VARIABLES
C
C   AA,BB,CC,DD--COEFFICIENTS IN WATER
C                BALANCE EQUATIONS
C   G------------PEACEMAN-RACHFORD G ARRAY
C   B------------PEACEMAN-RACHFORD B ARRAY
C   I------------FOR ROW CALCULATIONS THIS
C                IS THE COLUMN NUMBER.  FOR
C                COLUMN CALCULATIONS THIS IS
C                THE ROW NUMBER.
C   J------------FOR COLUMN CALCULATIONS THIS
C                IS THE ROW NUMBER.
C                FOR ROW CALCULATIONS THIS
C                IS THE COLUMN NUMBER.
C   HH,SI,QQ,TT--EXAMPLE MODEL
C                DEFAULT VALUES
C   NC-----------NUMBER OF COLUMNS OF MODEL
C   NR-----------NUMBER OF ROWS OF MODEL
C   DELTA--------TIME INCREMENTS, IN DAYS
C   NSTEP--------NUMBER OF TIME INCREMENTS
C   ITER---------NUMBER OF PROGRAM ITERATIONS
C   ERROR--------CRITERIA FOR SOLUTION
C                CONVERGENCE, IN FT
C   COLM---------ROW CALCULATIONS WHEN
C                COLM=0, COLUMN CALCULATIONS
C                WHEN COLM=1
C   S(10,50,3)---CORE STORAGE ARRAY FOR
C   *  *  *      NODE DATA OF THREE ROWS
C   *  *  *      OR COLUMNS OF MODEL
C   *  *  *****FOR ROW CALCULATIONS,
C   *  *         J1=NODE DATA FROM ADJACENT ROW.
C   *  *         J2=NODE DATA FROM ROW ALONG
C   *  *         WHICH HEADS WILL BE CALCULATED.
C   *  *         J3=NODE DATA FROM OTHER
C   *  *         ADJACENT ROW.
C   *  *         FOR COLUMN CALCULATIONS,
C   *  *         J1=NODE DATA FROM ADJACENT COLUMN.
C   *  *         J2=NODE DATA FROM COLUMN ALONG
C   *  *         WHICH HEADS WILL BE CALCULATED.
C   *  ******FOR ROW CALCULATIONS,
C   *          THIS IS THE COLUMN NUMBER.
C   *          FOR COLUMN CALCULATIONS,
C   *          THIS IS A ROW NUMBER.
C   ********CORE STORAGE LOCATIONS
C            FOR NODE DATA AS FOLLOWS:
C            1---HEAD, H, AT END OF
C                TIME INCREMENT, IN FT
C            2---HEAD, HO, AT BEGINNING
C                OF TIME INCREMENT, IN FT
C            3---DIFFERENCE IN HEADS, D,
C                AS DEFINED IN PREDICTOR SECTION
C            4---AQUIFER TRANSMISSIVITY, T1
C                BETWEEN I,J AND I,J+1
C                IN GPD/FT
C            5---AQUIFER TRANSMISSIVITY, T2
C                BETWEEN I,J AND I+1,J
C                IN GPD/FT
C            6---STORAGE FACTOR, SF1,
C                IN GAL/FT
C            7---NET WITHDRAWAL RATE,
C                Q, IN GPD
C            8---RECHARGE FACTOR, R,
C                IN GPD/FT
C            9---ELEVATION OF STREAM OR
C                LAND SURFACE, RH, IN FT
C            10--ELEVATION OF STREAMBED
C                BOTTOM, ELEVATION BELOW
C                WHICH EVAPOTRANSPIRATION
C                CEASES, OR ELEVATION OF
C                TOP OF AQUIFER, RO, IN FT
C   ISP---------TIME STEPS BEFORE NET
C               WITHDRAWAL RATES ARE CHANGED
C   IP----------I COORDINATE WHERE NET
C               WITHDRAWAL TAKES PLACE
C   JP----------J COORDINATE WHERE NET
C               WITHDRAWAL TAKES PLACE
C
C
      DIMENSION S(10,50,3),G(50),B(50)
C
C   DEFINE STRUCTURE OF DISK STORAGE
C
      DEFINE FILE 1(2500,20,U,ID1)
C
C
C   DEFINE INPUT-OUTPUT DEVICE NUMBERS.
C   FOR SOME IBM 1130 COMPUTERS IN=2
C   AND OUT=3.
C
      IN=5
      IOUT=6
C


C   SET VARIABLE PUMPAGE INDEX
      ISP=-1
C
C   CALL SUBROUTINE TO READ
C   THE PARAMETER AND DEFAULT
C   VALUE CARDS AND FILL
C   DISK STORAGE ARRAYS
C
      CALL PARM(NSTEP,DELTA,ERROR,NC,NR,IN)
C
C   CALL SUBROUTINE TO READ NODE CARDS
C   AND REPLACE VALUES IN APPROPRIATE
C   LOCATIONS ON DISK STORAGE
C
      CALL NODE(NC,NR,IN)
C
C   START OF SIMULATION
C
40    TIME=0.0
      DEL=DELTA
50    DO 370 ISTEP=1,NSTEP
      ITER=0
      TIME=TIME+DELTA
60    E=0.0
      ITER=ITER+1
C
C   TRANSFER FIRST THREE ROWS OF DATA FROM
C   DISK PACK TO CORE STORAGE FOR ROW
C   CALCULATIONS
C
70    ID1=1
      READ(1'ID1)(((S(L,I,K),L=1,10),
     1I=1,NC),K=1,3)
      M1=4
      M2=5
      COLM=0.
      N=NC-1
      NL=NR-1
      GO TO 100
C
C   TRANSFER FIRST THREE COLUMNS OF DATA
C   FROM DISK PACK TO CORE STORAGE FOR
C   COLUMN CALCULATIONS
C
80    DO 90 I=1,NR
      ID1=(I-1)*NC+1
90    READ(1'ID1)((S(L,I,K),L=1,10),K=1,3)
      M1=5
      M2=4
      COLM=1.0
      N=NR-1
      NL=NC-1
C
C   SET CORE STORAGE INDICES
C
100   J1=1
      J2=2
      J3=3
      J=2
C
C   ZERO OUT FIRST G AND B VALUES
C
      G(1)=0.0
      B(1)=0.0
C
C   CALCULATE HEADS BY MIADI METHOD
C
110   DO 150 I=2,N
      IF(S(1,I,J2)-S(10,I,J2))130,130,120
120   RE=S(9,I,J2)*S(8,I,J2)
      RB=1.0
      GO TO 140
130   RE=(S(9,I,J2)-S(10,I,J2))*S(8,I,J2)
      RB=0.0
140   DD=S(2,I,J2)*S(6,I,J2)/DELTA-S(7,I,J2)+
     1RE+S(1,I,J1)*S(M1,I,J1)+S(1,I,J3)*S(M1,I,J2)
      BB=S(6,I,J2)/DELTA+S(8,I,J2)*RB+S(M1,I,J1)+
     1S(M1,I,J2)+S(M2,I-1,J2)+S(M2,I,J2)
      AA=-S(M2,I-1,J2)
      CC=-S(M2,I,J2)
      W=BB-AA*B(I-1)
      B(I)=CC/W
      G(I)=(DD-AA*G(I-1))/W
150   CONTINUE
      K=N
160   HA=G(K)-B(K)*S(1,K+1,J2)
      E=E+ABS(S(1,K,J2)-HA)
      S(1,K,J2)=HA
      K=K-1
      IF(K-1)170,170,160
170   CONTINUE
      IF(J-NL)180,230,230
C
C   PUT BACK ON DISK THE
C   UNNECESSARY ROW OF DATA
C
180   IF(COLM)200,190,200
190   ID1=(J-2)*NC+1
      WRITE(1'ID1)(((S(K,I,J1),K=1,10),I=1,NC)
```

Figure 2.   Aquifer simulation model for disk supported computer

```
C     READ INTO CORE THE NEXT
C     ROW OF DATA
C
      ID1=(J+1)*NC+1
      READ(1'ID1)((S(K,I,J1),K=1,10),I=1,NC)
      GO TO 220
C
C     PUT BACK ON DISK THE
C     UNNECESSARY COLUMN OF DATA
C
200   DO 210 I=1,NR
      ID1=(I-1)*NC+J-1
      WRITE(1'ID1)(S(K,I,J1),K=1,10)
C
C     READ INTO CORE THE NEXT
C     COLUMN OF DATA
      ID1=ID1+2
210   READ(1'ID1)(S(K,I,J1),K=1,10)
C
C     SYNCHRONOUS REVOLVING
C     SUBSCRIPT SPECIFICATION
C
220   K=J1
      J1=J2
      J2=J3
      J3=K
      J=J+1
      GO TO 110
C
C     ROW OR COLUMN CALCULATIONS
C     COMPLETED AND CORE STORAGE PURGED
C
230   IF(COLM)250,240,250
240   ID1=(J-2)*NC+1
      WRITE(1'ID1)((S(K,I,J1),K=1,10),I=1,NC),
     1((S(K,I,J2),K=1,10),I=1,NC)
      GO TO 80
250   DO 260 I=1,NR
      ID1=I*NC-2
260   WRITE(1'ID1)(S(K,I,J1),K=1,10),
     1(S(K,I,J2),K=1,10)
      COLM=0.0
      N=NC-1
      NL=NR-1
C
C     TEST FOR CONVERGENCE
C     AND LIMIT ITERATIONS TO 20
C
270   IF(E-ERROR)280,280,275
275   IF(ITER-20)60,280,280
C
C     CALL SUBROUTINE TO PRINT
C     RESULTS AND PREPARE FOR
C     NEXT TIME STEP
C
280   CALL OUT(S,ISP,ISTEP,TIME,E,ITER,NC,
     1NR,IN,IOUT,RESET)
C
      DELTA=DELTA*1.2
      IF(RESET)370,370,385
385   DELTA=DEL
370   CONTINUE
      END
C     ***********************************
C     THIS SUBROUTINE READS THE PARAMETER AND
C     DEFAULT VALUE CARD AND FILLS THE
C     DISK STORAGE ARRAYS
C     ***********************************
C
      SUBROUTINE PARM(NSTEP,DELTA,ERROR,NC,NR,IN)
C
      DEFINE FILE 1(2500,20,U,ID1)
C
C     READ DEFAULT VALUE CARD
C
5     READ(IN,10)NSTEP,DELTA,ERROR,NC,NR,
     1TT,S1,HH,QQ,RR,RRH,RRO
10    FORMAT(I6,2F6.0/2I6,7F6.0)
C
C     CONSTRUCT FRAME AROUND MODEL
C
      NC=NC+2
      NR=NR+2
C
C     STORE DEFAULT VALUES ON DISK PACK
C
      NA=(NC+1)*10
      Z=0.0
      ID1=1
      WRITE(1'ID1)(Z,I=1,NA),((HH,HH,Z,TT,
     1TT,S1,QQ,RR,RRH,RRO,I=4,NC),HH,HH,
     2Z,TT,Z,S1,QQ,RR,RRH,RRO,(Z,I=1,20),
     3J=4,NR),(HH,HH,Z,Z,TT,S1,QQ,RR,RRH,RRO,
     4I=4,NC),(HH,HH,Z,Z,S1,QQ,RR,RRH,RRO,
     5(Z,I=1,NA)
      RETURN
      END

C     ***********************************
C     THIS SUBROUTINE READS THE NODE CARDS
C     AND PLACES THEIR VALUES IN THE
C     APPROPRIATE LOCATIONS ON THE DISK
C     ***********************************
C
      SUBROUTINE NODE(NC,NR,IN)
      DEFINE FILE 1(2500,20,U,ID1)
C
C     READ NODE CARDS
C
20    READ(IN,30)I,J,T1,T2,SF1,H,
     1Q,R,RH,RO
30    FORMAT(2I3,2F6.0,2F4.0,4F6.0)
C
C     SET ZERO TRANSMISSIVITIES
C     IN MODEL FRAME
      IF(I)29,40,29
29    IF(I-NC+2)32,31,32
31    T2=0.0
32    IF(J-NR+2)34,33,34
33    T1=0.0
C
C     STORE NODE DATA ON DISK PACK
C
34    ID1=J*NC+I+1
      Z=0.0
      WRITE(1'ID1)H,H,Z,T1,T2,SF1,Q,
     1R,RH,RO
      GO TO 20
40    RETURN
      END
C
C     ***********************************
C     THIS SUBROUTINE PRINTS HEADS, PREDICTS
C     FUTURE HEADS, AND CHANGES NET
C     WITHDRAWAL RATES IF NECESSARY.
C     ***********************************
C
      SUBROUTINE OUT(S,ISP,ISTEP,TIME,E,ITER,
     1NC,NR,IN,IOUT,RESET)
C
      DEFINE FILE 1(2500,20,U,ID1)
      DIMENSION S(10,50,3)
C
C     MAIN OUTPUT SECTION, UPDATE CALCULATED
C     HEADS, AND PREDICT HEADS FOR NEXT
C     TIME INCREMENT
C
      RESET=0
280   WRITE(IOUT,285)TIME,E,ITER
285   FORMAT(6H2TIME=,F6.2///,E20.7,I5)
      JJ=NR-1
      II=NC-1
      DO 360 J=2,JJ
      ID1=(J-1)*NC+1
      READ(1'ID1)((S(K,I,1),K=1,10),I=1,NC)
C
C     PREDICT HEADS FOR NEXT
C     TIME INCREMENT (SEE PAGES
C     11 AND 12 IN BULLETIN 55).
C
      DO 350 I=2,II
      D=S(1,I,1)-S(2,I,1)
      S(2,I,1)=S(1,I,1)
      F=1.0
      IF(S(3,I,1))290,340,290
290   F=D/S(3,I,1)
300   IF(F-5.0)320,320,310
310   F=5.0
320   IF(F)330,340,340
330   F=0.0
340   S(3,I,1)=0
350   S(1,I,1)=S(1,I,1)+D*F
C
C     CHANGE NET WITHDRAWAL RATES
C
      K=J-1
349   IF(ISP)351,356,353
351   READ(IN,352)ISP,IP,JP,QQ
352   FORMAT(3I3,F11.0)
      GO TO 349
353   IF(ISP-ISTEP)351,354,356
354   IF(JP-J)356,357,356
357   S(7,IP+1,1)=QQ
      RESET=1
      GO TO 351
356   CONTINUE
C
C     PRINT RESULTS
C
      WRITE(IOUT,355)K,(S(2,I,1),I=2,II)
355   FORMAT(I5,5X,10F10.4/(12X,10F10.4))
C
C     RETURN INFORMATION TO DISK PACK
C
      ID1=(J-1)*NC+1
360   WRITE(1'ID1)((S(K,I,1),K=1,10),I=1,NC)
      RETURN
      END
```

Figure 2. (Concluded)

8

AQUIFER SIMULATION PROGRAM
FOR DISK SUPPORTED SMALL COMPUTERS

The aquifer simulation program listing given in figure 2 was coded in FORTRAN to solve the sets of row node equations 3"3d.

Figure 3 illustrates the parameters included in the simulation program. Briefly, the program is intended for use when analyzing cause and effect relationships involving drawdowns or heads in a nonsteady, heterogeneous and/or homogeneous aquifer system under nonleaky and leaky artesian conditions. Programming techniques involving time varying pumpage from wells, natural or artificial recharge rates, the relationships of water exchange between surface waters and the groundwater reservoir, the process of groundwater evapotranspiration, and the mechanism of flow from springs
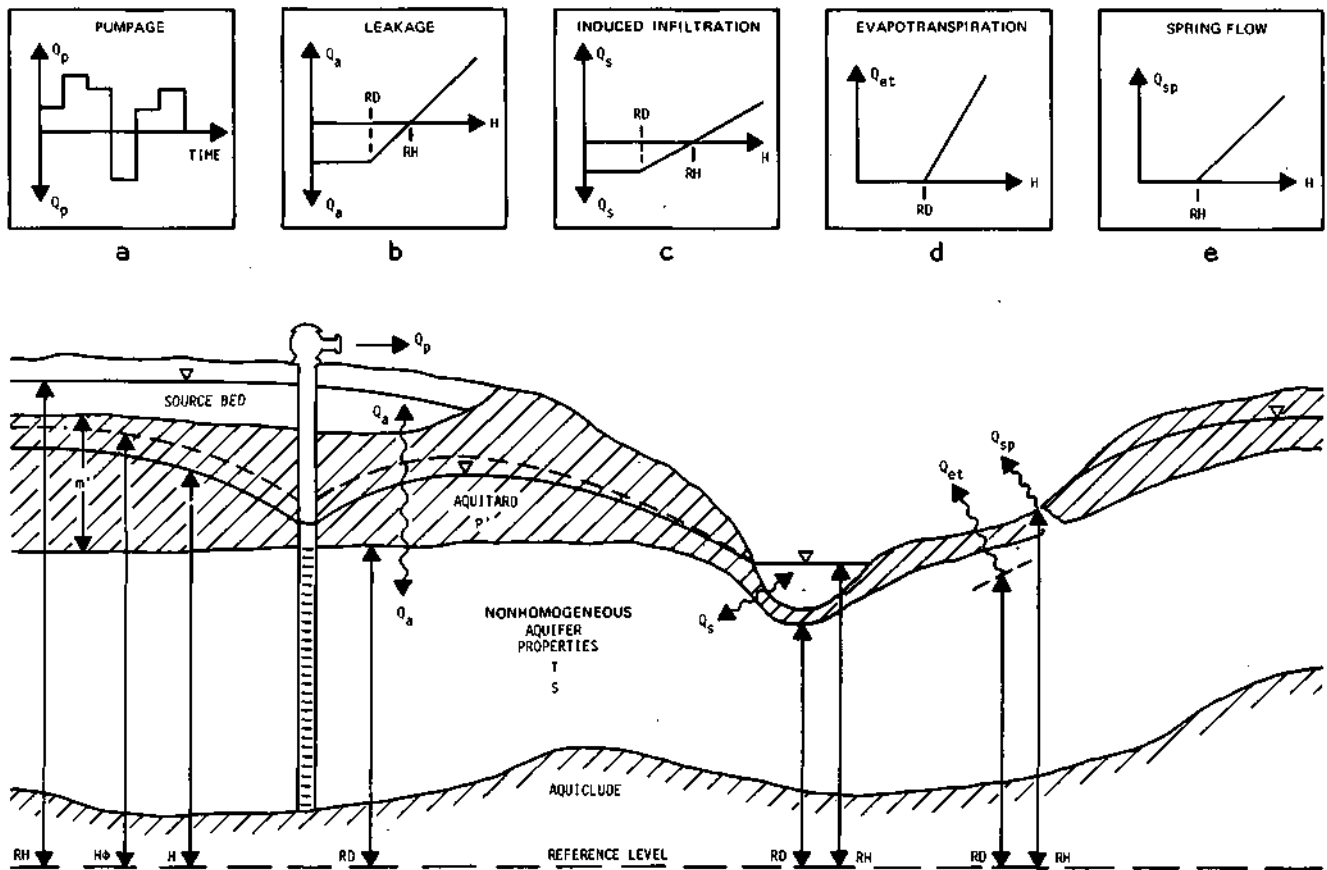
Figure 3. Generalized aquifer cross section showing simulation program parameters

9

are also included. Under these conditions it is possible to include special
boundaries such as irregular barrier or recharge boundaries (constant head) and
constant withdrawal or recharge rates.

The program listing of figure 2 is written in such a way that it will operate
with any consistent set of units. However, the comment cards preceding the actual
program list the variables in the gallon-day-foot system of units. This system of
units is made consistent (exterior to the program) by working with a so-called
storage factor defined as

$$SF_{i,j} = 7.48S_{i,j}\Delta x\Delta y \tag{4}$$

where

SF$_{i, j}$ = storage factor for node located at model coordinates i, j, in
     gallons per foot (gal/ft)

S = the aquifer storage coefficient for artesian conditions, a fraction

7.48 = number of gallons in a cubic foot of water, in gallons per cubic
     feet (gal/ft^3)

Δx,Δy = finite difference grid intervals, in feet

The storage factor of equation 4 is substituted for the terms $S_{i,j}\Delta x^2$ in
equations 3, 3b, and 3d, which transforms these equations so that all combinations
of terms have consistent units of gallons per day (gpd). To work with a consistent
set of units such as in the metric system the user merely drops the constant 7.48
from equation 4.

Job Setup

The computer job setup will first be explained in general and then in more
detail as necessary.

The aquifer system properties are discretized by superposing a finite differ-
ence grid over maps of the aquifer system as shown in figure 4. The total dimen-
sions of the grid are defined by NC, the number of columns of the model, and by NR,
the number of rows of the model. Next, a parameter card and default value card are
prepared according to the formats illustrated in figures 5a and b. The default
value card provides data for simulating an NC by NR aquifer system model having
homogeneous properties with identical initial heads and net withdrawal rates.
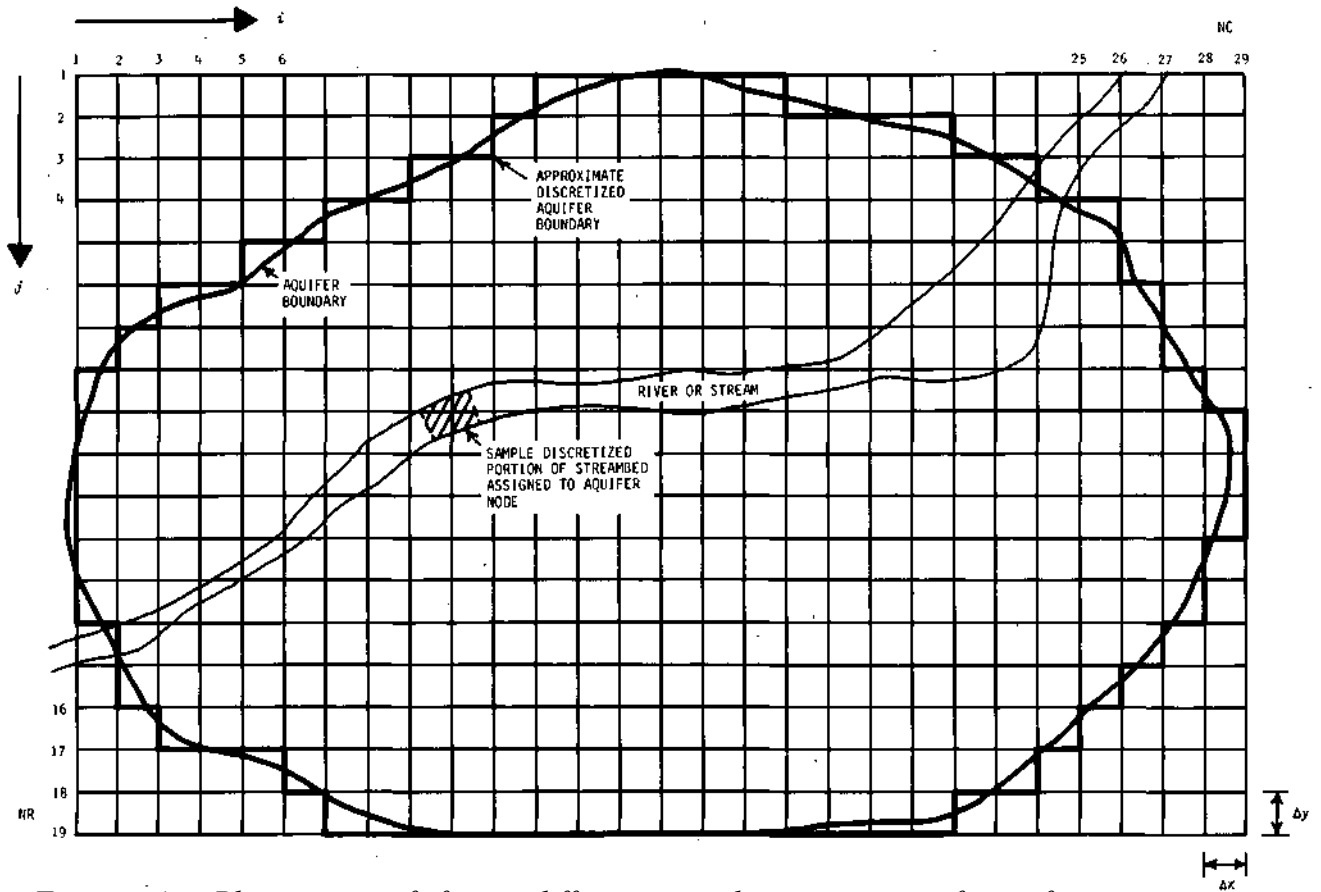
10

Figure   4.   Plan   view   of   finite   difference   grid   over   map   of   aquifer   system
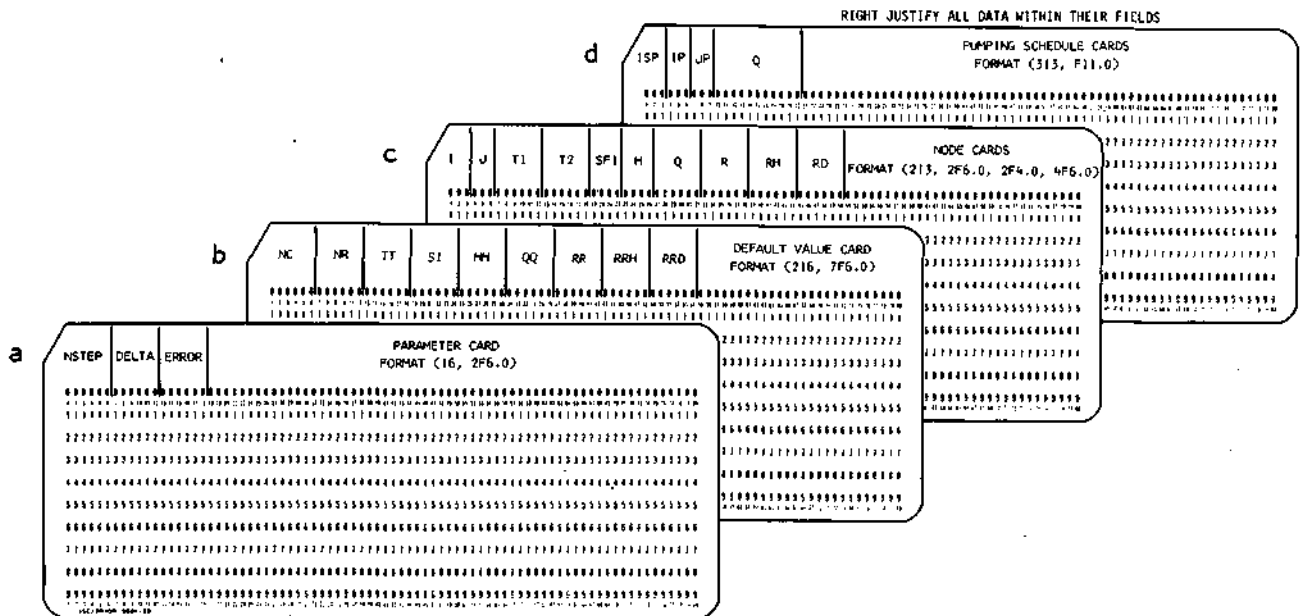

Figure   5.   Input   data   formats   for   computer   program

A node card deck is then prepared according to the format illustrated in figure 5c. The node card deck contains one card for each node that has any aquifer system properties differing from those defined on the default value card. Finally, pumping schedule cards are prepared according to the card format illustrated in figure 5d.
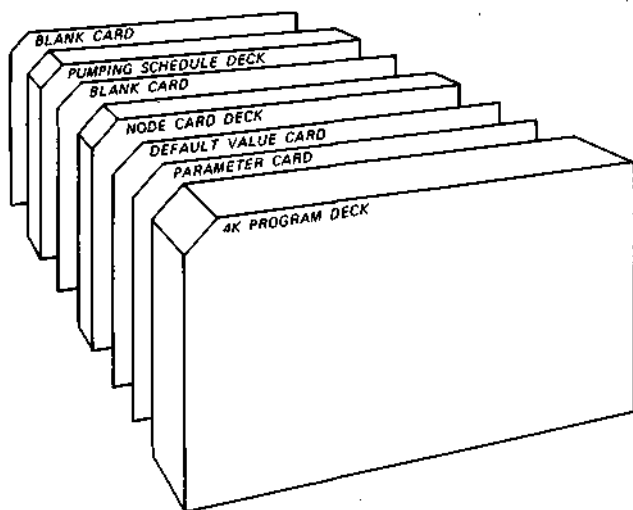
Figure 6, Typical 30b setup

The program deck, parameter card, default value card, node card deck, and pumping schedule deck are assembled in the order illustrated in figure 6. Appropriate computer installation job control cards are included and the program is ready to run. The computer output will be in the form of printed numerical values of heads for all nodes at the end of every time increment according to FORMAT statement 355 of the program shown in figure 2.

Preparation of Data Cards

Parameter Card. Enter numerical values for NSTEP, DELTA, and ERROR according to the format given in figure 5a. A general rule for choosing an initial DELTA is to decide at what minimum TIME drawdowns or heads of interest occur and then precede this time by at least six time increments. Choose an initial value of ERROR from the following formula

$$ERROR = Q \times DELTA/(10 \times SF1) \qquad (5)$$

where

Q = total net withdrawal rate of model, in gpd

DELTA = initial time increment, in days

SF1 = average storage factor of model, in gpd/ft

The computer program is then 'tested' by making a few preliminary runs with different values of ERROR, with a final value being chosen at the point where reduction in that term does not significantly change the solution.

12

Default Value Card. After entering values of NC and NR on the card, enter the most commonly occurring values of the aquifer system parameters.

Node Card Deck. It should be emphasized that the node deck contains one card for each node that has any aquifer system properties differing from those defined on the default value card. If a node card is included, all values must be punched on it even if some of the values are equal to the default values.

Leaky Artesian Conditions. Calculate recharge factors $R_{i,j}$ in gallons per day per foot (gpd/ft) from the following formula

$$R_{i,j} = (P'/m')\Delta x\Delta y \tag{6}$$

where

    P' = vertical hydraulic conductivity of confining bed, in gallons per day per square foot (gpd/ft^2)

    m' = thickness of confining bed, in feet

The recharge factor defines the slope of the line given in figure 3b.

Appropriate source bed heads RH, elevations of the top of the aquifer RD, and recharge factors are entered on the default value card and any differing values on the node cards.

Induced Infiltration. Calculate recharge factors (in gpd/ft) from the following formula

$$R_{i,j} = (P'/m')A_s \tag{7}$$

where

    P' = hydraulic conductivity of the streambed, in gpd/ft^2

    m' = thickness of streambed, in feet

    A_s = area of the streambed assigned to node, in square feet

The recharge factor defines the slope of the line shown in figure 3c. Appropriate values of stream water surface elevations RH, elevations of the bottom of the streambed RD, and recharge factors calculated from equation 7 are entered on node cards.

Evapotranspiration. Find the slope of the line shown in figure 3d by calculations with field data. The value of the slope of that line is then entered on node cards in the space reserved for the recharge factor. Set values of both RD and RH equal to one another. RD is defined as the elevation of the water table below which the effects of evapotranspiration cease.

Flow from Springs. Determine a recharge factor (slope of the line shown in figure 3e) from field data of flow versus head changes in the vicinity of the spring. A recharge factor may also be found empirically by matching simulated with observed spring flows. The elevation at which water flows from the spring is recorded in both RH and RD.
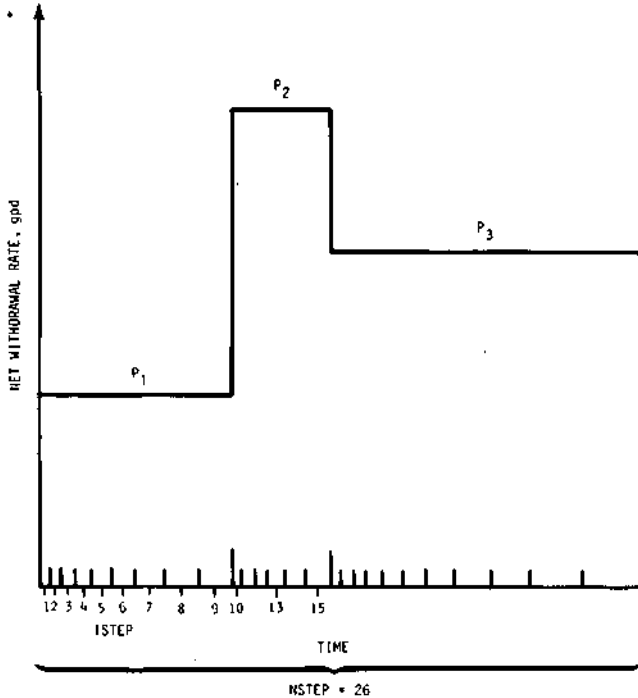

Figure 7. Example variable net withdrawal rate schedule

Variable Pumping Rates. Set up a stepwise pumping schedule, as exemplified in figure 7, for each well of the model. The computer program can manipulate positive (pumping), negative (recharge), or zero withdrawal rates. Between changes in pumping rates, the computer program operates with nonuniform time steps within each period of pumping. However, DELTA is reset to its initial value at each change in pumping rate.

Once the pumping schedules have been set up, enter the rates and well location coordinates on node cards and on pumping schedule cards according to the formats of figure 5c and d. The first pumping rate of the schedule is entered into Q by including a node card. Subsequent pumping rates are entered on pumping schedule cards, their rates becoming effective after the time step number, ISTEP, equals the value entered as ISP. With the schedule of figure 7, for example, the pumping rate P. would be included on a node card and two pumping schedule cards would be prepared with ISP equaling 9 for rate P_2 and ISP equaling 15 for rate P_3. The pumping schedule deck must be in order of increasing ISP and row number, and preceded and followed by a blank card.

The basic operation of the computer program is explained in the following discussion according to the listing of figure 2 and the flow chart given in figure 8.

The computer program consists of a MAIN section and the three subroutines, PARM, NODE, and OUT. The storage requirements, including arrays, are only 2520 words for MAIN, 313 words for PARM, 202 words for NODE, and 392 words for OUT. For clarity the total program will be explained in the order of the subroutines first and the main section last.
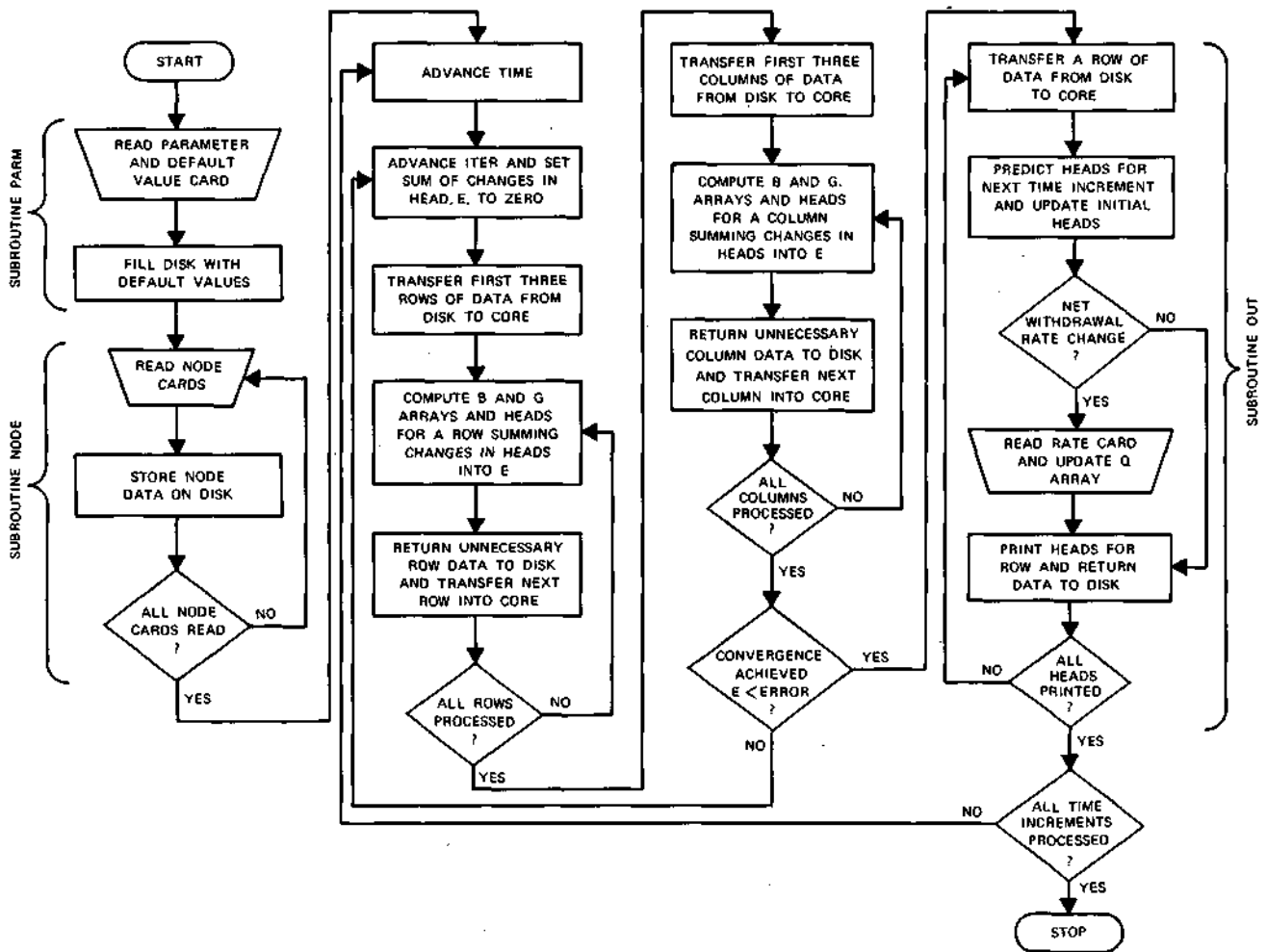

Figure 8. Flow chart for computer program

Subroutine PARM

The first statement of this subroutine defines the structure of the disk storage. Storage space is allocated for a 50 by 50 model or 2500 nodes. Each node has 10 variables for storage, denoted as a 'record.' Since each variable requires 2 words when using an IBM 1130 computer, 20 words per record must be specified on the disk to store the 10 variables per node. The U indicates that the disk input/output functions are unformated. The term ID1 is used as the 'pointer' for reading and writing on the disk.

Figure 9 illustrates how the disk storage configuration is set up from data specified on the parameter and default value cards. A simple 4 by 5 finite difference grid is used as an example. One should note that a 'frame' of extra rows and columns has been set up around the simulated model. In storing the default values on the disk pack, the transmissivities within the frame are set equal to zero. This has been done so that boundary transmissivities are automatically set to zero at the beginning or end of a row or column.

Subroutine NODE

This subroutine reads the node cards and places the data in the appropriate locations on the disk. In addition, if any of the node cards contain data which are in contact with the frame, then the aquifer transmissivities are reset to zero in the frame.

Subroutine OUT

This subroutine prints the simulation results, predicts future heads, and reads new net withdrawal rate values in preparation for the next time step. To save core storage, the pumping schedule cards are read only as they are needed. The computer operator should be made aware of this feature since he might, out of curiosity, remove the cards during the simulation and cause the computation to stop.

MAIN Program

The dimension statement reserves core storage sufficient for three rows or three columns of data from models that contain not more than 50 rows and 50 columns.
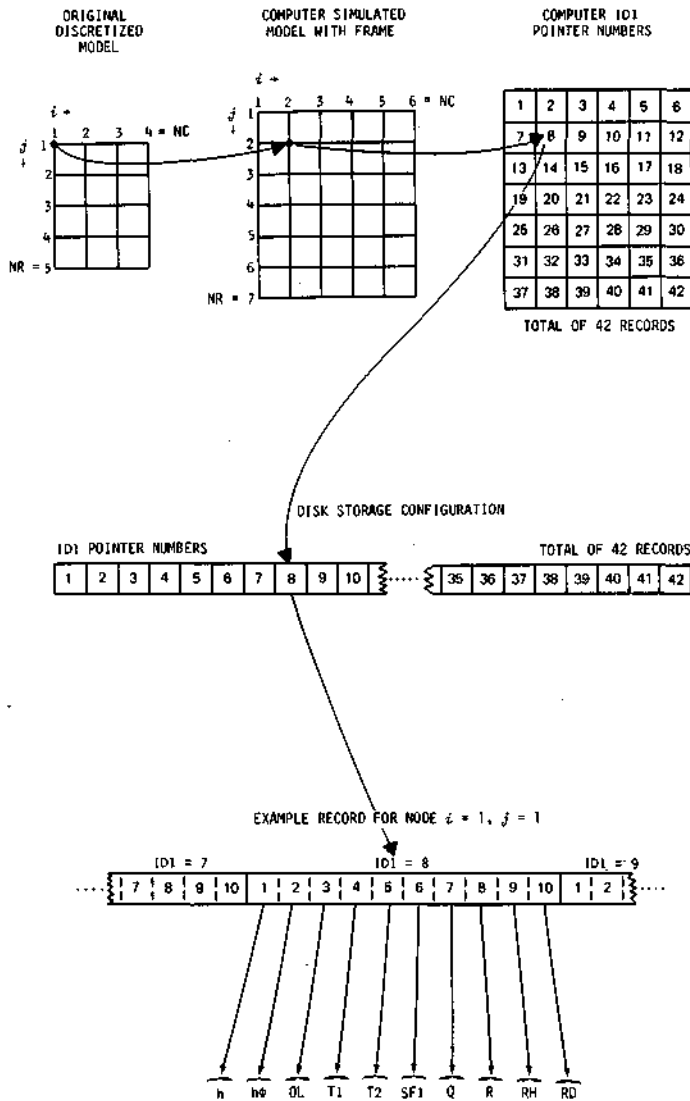
16

Figure 9 (left):

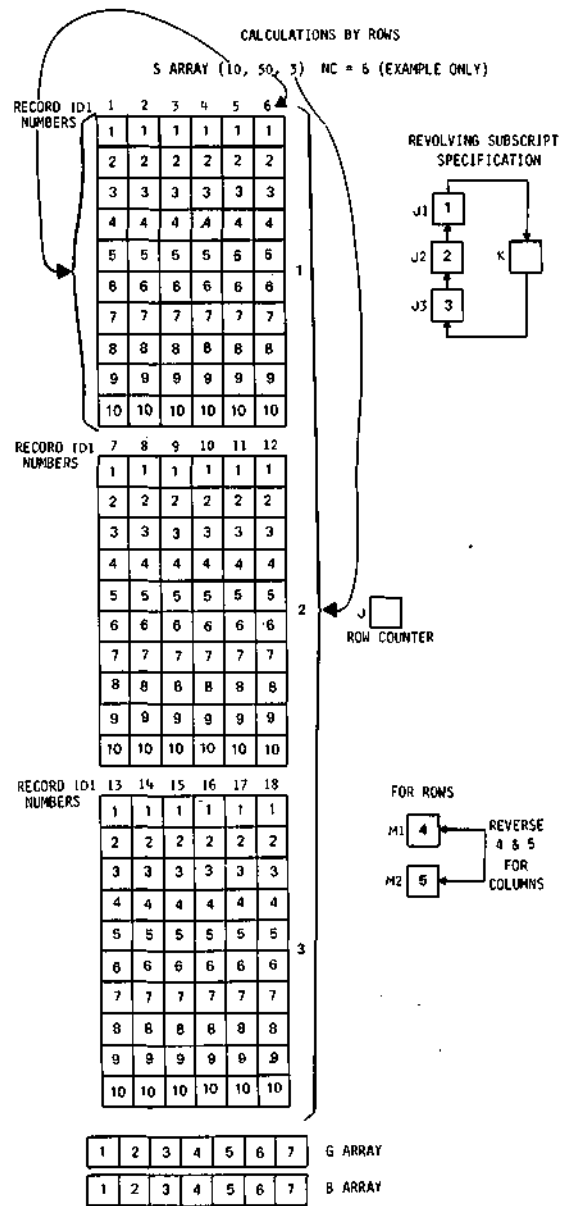ORIGINAL DISCRETIZED MODEL
COMPUTER SIMULATED MODEL WITH FRAME
COMPUTER IDI POINTER NUMBERS

TOTAL OF 42 RECORDS

DISK STORAGE CONFIGURATION

IDI POINTER NUMBERS   TOTAL OF 42 RECORDS

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | ... | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42 |

EXAMPLE RECORD FOR NODE $i = 1$, $j = 1$

IDI = 7   IDI = 8   IDI = 9

h   hφ   ΦL   T1   T2   SF1   Q   R   RH   RD

Figure 9. Disk storage configuration

Figure 10 (right):

CALCULATIONS BY ROWS
S ARRAY (10, 50, 5)   NC = 6 (EXAMPLE ONLY)

RECORD IDI NUMBERS   1 2 3 4 5 6

REVOLVING SUBSCRIPT SPECIFICATION
J1 1
J2 2   K
J3 3

RECORD IDI NUMBERS   7 8 9 10 11 12

ROW COUNTER
J

RECORD IDI NUMBERS   13 14 15 16 17 18

FOR ROWS
M1 4   REVERSE 4 & 5
M2 5   FOR COLUMNS

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | G ARRAY |

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | B ARRAY |

Figure 10. Core storage configuration

The G and B arrays (Prickett and Lonnquist, 1971) are used in both the row and column calculations and must be dimensioned to the maximum row or column limits.

Figure 10 illustrates the computer core storage configuration with the first three rows of data in core for the example model of figure 9. The largest portions of the MAIN program are devoted first to statements transferring the proper rows or

columns of data back and forth between the disk and core. Secondly, a series of statements are included to determine the core storage location of the row or column along which heads are to be calculated. This second series of statements has to do with what is termed 'revolving subscript specification' to be explained as follows.

The third subscript in the DIMENSION statement S array reserves three block locations of row or column data with numbered subscripts 1, 2, and 3. At the beginning of the simulation the first three rows of data are transferred from the disk into these 3 block locations. The data from row one goes in location 1, that from the second row in 2, and that from the third row in 3. As calculations of heads progress row by row, data are transferred between the disk and available core storage. The new row data are transferred into the block location vacated by unnecessary row data. The sequence of the row data in the three S array block locations is as follows. The data for a row along which heads are to be calculated follows a core storage sequence (J2) of 2, 3, 1, 2, 3, 1, etc. The adjacent row data needed for calculation of heads follow a sequence which is one step behind or ahead of the above sequence; namely, for one adjacent row the sequence (J1) is 1, 2, 3, 1, 2, 3, etc., and for the other adjacent row the sequence (J3) becomes 3, 1, 2, 3, 1, 2, etc. Therefore, the formulation of the MIADI program section can be completely generalized by revolving the S array subscripts J1, J2, and J3 from 1 through 3 as described above. A similar description can be given for calculations along columns.

EXAMPLE PROGRAM CHECKOUT PROBLEM

A small test problem is included to check out the program of this report for use on various small core storage disk supported computers.

As illustrated in figure 11, two wells are pumping in a finite aquifer near a river. Initial heads in the aquifer and river are set equal to zero, the chosen arbitrary head reference level. It is assumed that the two wells are to be pumped at the rates shown in the schedules of figure 12. The river shown in figure 11 is 50 feet wide, has a streambed thickness of 0.3 feet, hydraulic conductivity of 1.2 gpd/ft^2, and a streambed bottom elevation of 0.8 feet below the reference level.

18

A square ($\Delta y = \Delta x$) finite difference grid interval of 1000 feet is assumed, and the computer model then is 5 columns by 7 rows in size. Data from figure 11 indicate that the coordinates of well 1 are $i = 3$, $j = 3$, and the coordinates of well 2 are $i = 4$, $j = 5$.

The recharge factors for the streambed of the river are, by equation 7

$$R_{i,j} = (P'/m')A_s = (1.2/0.3)\ 1000 \times 50$$
$$R_{i,j} = 200,000\ \text{gpd/ft}$$

where A_s is the area of the streambed simulated at each node discretized into 50-foot widths by 1000-foot lengths along row 6 of the model.

The storage factor for the entire model is by equation 4

$$SFI_{i,j} = 7.48\ S_{i,j}\ \Delta x \Delta y$$
$$SFI_{i,j} = 7.48 \times 1.003 \times 10^{-3} \times 1000 \times 1000$$
$$SFI_{i,j} = 7500\ \text{gal/ft}$$

The data deck setup for the above hypothetical test model is given in figure 13. A sample time-drawdown curve for an observation point located at model grid point $i = 3$, $j = 7$ is given in figure 14. A sample printout of all model heads for TIME = 2.20 days is given in figure 15.

COMPUTATION TIME COMPARISONS

The example program checkout problem of this report, and other equivalent ones, have been run on IBM 1130 disk supported computers with core storage of 8000 and 16,000 words. In addition, the same problems have been run, all in core storage, with the programs given by Prickett and Lonnquist (1971) on IBM 360 models 75 and 40 computers. Comparisons for needed central processor times are as follows.

Computer	Time (seconds)
360/75 (in core)	6
360/40 (in core)	25
1130 (16,000 words)	150
1130 (8000 words)	460
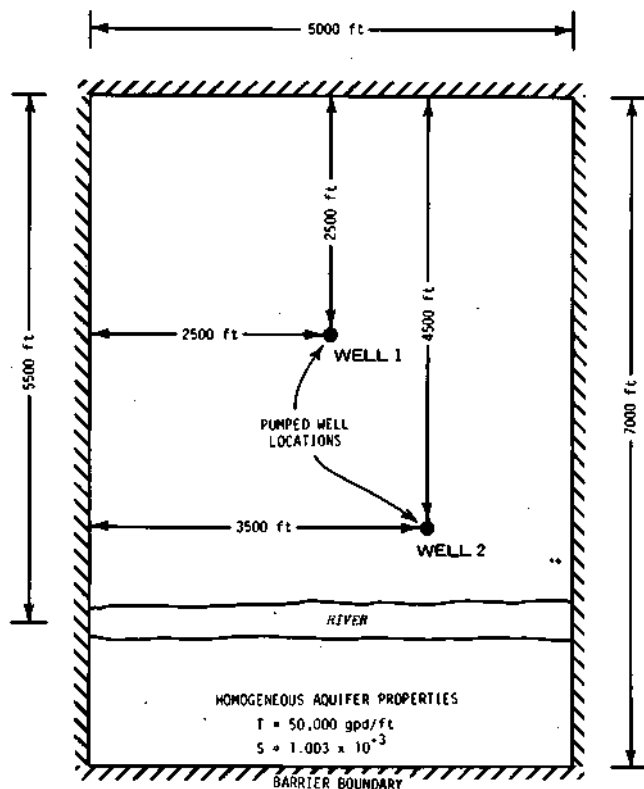
Figure 11. Example aquifer problem
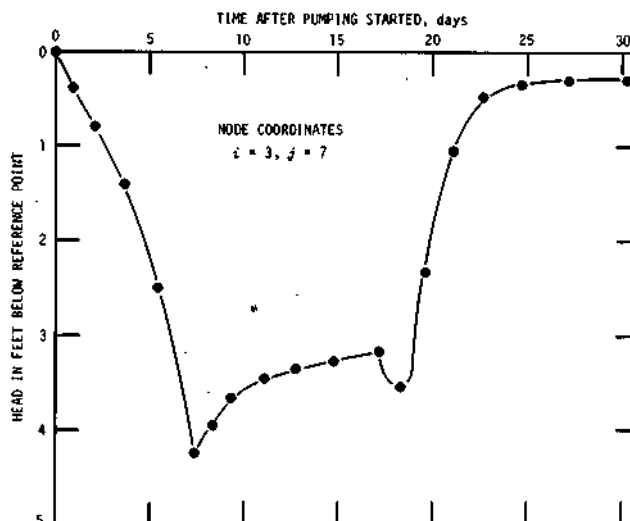

Figure 13. Data deck setup for example problem

Figure 14. Time drawdown curve for example problem
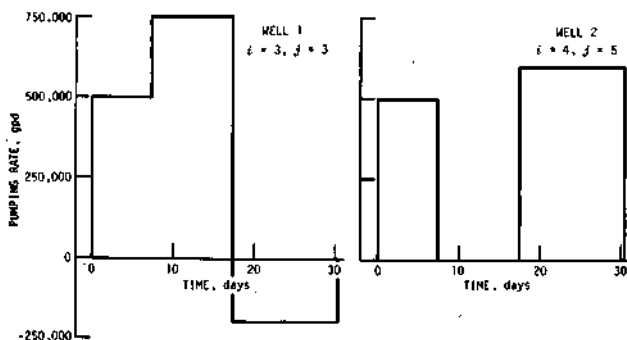
Figure 12. Example pumping schedules for program checkout problem

Figure 15. Numerical printout for example problem

20

It should be pointed out that the above times do not allow for reading cards or printing the results. Also from a user's standpoint, a factor which does not show up in the above table is actual turnaround time, which is the time between handing the cards to the operator and receiving the printed output. Turnaround time can vary considerably from one computer installation to another. For instance, turnaround time for the 360/75, operated through a remote job entry terminal, at best was about 10 minutes. This time can be compared with about 5 minutes turn-around for the 1130 with 16,000 words of storage. The main cause of the difference here is that the 360/75 computer is processing several jobs simultaneously whereas the 1130 is working on only one job.

It is obvious that the small computer consumes large amounts of central processor time, but it does allow problem solving with small core storage.

REFERENCES

Jacob, C. E. 1950. *Flow of groundwater. In* Engineering Hydraulics, Hunter. Rouse (ed.), John Wiley & Sons, Inc., New York.

Pinder, G. F. 1970. *A digital model for aquifer evaluation.* U. S. Geological Survey, Techniques of Water-Resources Investigations, Book 7, Chapter Cl.

Prickett, T. A., and C. G. Lonnquist. 1968a. *Comparison between analog and digital simulation techniques for aquifer evaluation.* Illinois State Water Survey Reprint Series 114.

Prickett, T. A., and C. G. Lonnquist. 1968b. *Aquifer simulation program listing using alternating direction implicit method.* Illinois State Water Survey mimeo-graphed report presented at International Association of Scientific Hydrology Symposium on Use of Computers in Hydrology, Tucson, Arizona.

Prickett, T. A., and C. G. Lonnquist. 1971. *Selected digital computer techniques for groundwater resource evaluation.* Illinois State Water Survey Bulletin 55.

Tyson, H. N., and E. M. Weber. 1964. *Ground-water management for the nation's future: Computer simulation of ground-water basins.* Proceedings American Society of Civil Engineers v. 90(HY4):59-77.