USGS
science for a changing world

Techniques of Water-Resources Investigations
of the United States Geological Survey

Chapter A6

A COUPLED SURFACE-WATER AND GROUND-WATER FLOW MODEL (MODBRANCH) FOR SIMULATION OF STREAM-AQUIFER INTERACTION

By Eric D. Swain and Eliezer J. Wexler

Book 6
MODELING TECHNIQUES

DCFM - Maximum value of the multiplier for the friction term in the momentum equation when the channel is dry.

All of these variables, except for DCFM, are BRANCH array dimensions described by Schaffranek and others (1981). DCFM is described previously in this text.

Another modification to the BRANCH' input data occurs in the second initial condition record for cross section (Schaffranek and others, 1981). The new input format is

Data -	ORIENT (MAXS)	BETVEL (MAXS)	ISTRM (1, MAXS)	ISTRM (2, MAXS),
Format -	F10.3	F10.3	I10	I10
Data -	ISTRM (3, MAXS)	CLK (MAXS)	ZBOT (MAXS)	
Format -	I10	F10.4	F10.4	

The new variables added to these records are

ISTRM (1, MAXS) - Layer number of aquifer model cell containing river reach.

ISTRM (2, MAXS) - Row number of aquifer model cell containing river reach.

ISTRM (3, MAXS) - Column number of aquifer model cell containing river reach.

CLK (MAXS) - Leakage coefficient for reach.

ZBOT (MAXS) - Elevation of riverbed.

To accommodate the additional cross-section data points added by the program to account for drying, the maximum number of cross-sectional data points per cross section (MXPT) should be set at least four more than the actual number of points input by the user.

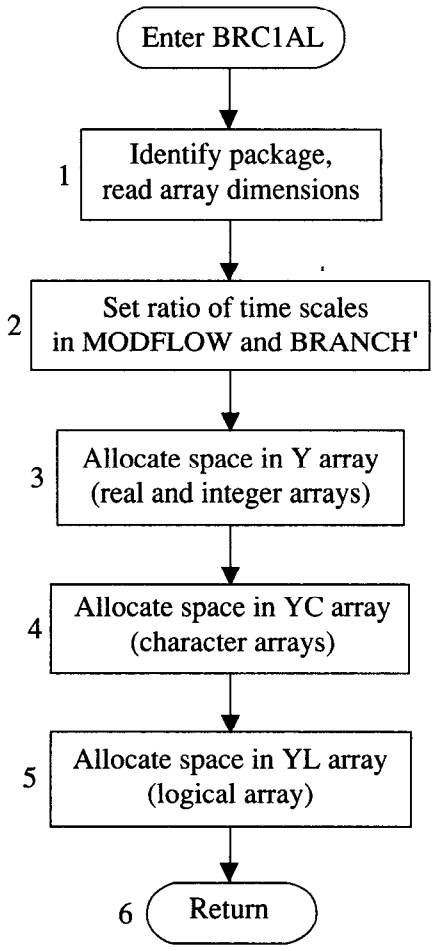

Figure 6. Flowchart of allocation module (BRC1AL).

MODULE BRC1AL

This module allocates space in three large arrays in the main program of MODFLOW for storing the arrays used in BRANCH'. The numbered sections in the narrative correspond to the numbered steps in the flowchart for BRC1AL shown in figure 6.

NARRATIVE

1. Print a message identifying the package. The array dimensions MXBH, NSEC, MAXS, MXTDBC, MAXZBD, MAXCZQ, MXWIND, MXJN, NSEG, MXPT, MXMD, MAXQBD, and MAXMZQ are entered along with a DCFM value. The multiplier DCFM is the factor by which the friction term in the momentum equation is multiplied when the channel is dry. The data entry format is described in the data entry section. The definition of the array dimensions can be found in the report by Schaffranek and others (1981) and in the data entry section.
2. Set the ratio of time scales in MODFLOW and BRANCH'. Although BRANCH' always works in seconds, MODFLOW can work in seconds, minutes, hours, days, and years, or the time units can be undefined. This option, defined as the variable ITMUNI (McDonald and Harbaugh, 1988), is used to calculate the ratio of the time units in the models, the variable TFCTR. An error message is printed if the undefined option is selected in MODFLOW.
3. Allocate space in the Y array. All BRANCH' arrays that contain real or integer numbers are stored in the Y array, with positions and space allocated by this routine.
4. Allocate space in the YC array. All BRANCH' arrays that contain character variables are allocated positions and space in the YC array.
5. Allocate space in the YL array. All BRANCH' arrays that contain logical variables are allocated positions and space in the YL array.
6. Return to MODFLOW.

PROGRAM LISTING FOR MODULE BRC1AL

```
      SUBROUTINE BRC1AL
     & (LENY,LENYC,LENYL,LINDAT,LIZDAT,LIQDAT,LITQMA,LITQMI,
     1 LQMAX,LQMIN,LQSUM,LZQMIN,LZQMAX,LAQMAX,LAQMIN,LA,LZ,LQ,LZP,LQP,
     2 LAP,LBP,LRP,LB,LR,LBT,LBTP,LXSTAT,LDX,LT,LRN,LWANGL,LGDATU,LORIEN
     3 ,LBETVE,LSUMET,LSUMCZ,LSCZQS,LSZQET,LITYPO,LZA,LAA,LBB,LBS,LIPT,
     4 LQA,LTA,LETA,LFUNET,LROW,LAM,LBMX,LBRNAM,LIJF,LIJT,LNSEC,LXSKT,
     5 LPLTBC,LPRTXS,LPRTBC,LPRTSU,LPPLTB,LITYPE,LIBJNC,LNDATA,LIZQBV,
     6 LISTAT,LKTTDB,LZQ,LDTT,LDATUM,LZQBVC,LZQPMI,LLARBP,LZHIGH,
     7 LZLOW,LLINPR,LARBER,LCLK,LZBOT,MXBH,NSEC,MAXS,MXTDBC,
     8 MAXZBD,MAXCZQ,MXWIND,MXJN,NSEG,MXPT,MXMD,MAXQBD,MAXMZQ,ITMUNI,
     9 LWINDS,LWINDD,LIDX,LICT,LW,LU,LUU,LBU,LBUU,LZSAV,LQSAV,LZPSAV,
     & LQPSAV,IN,IOUT,TFCTR,LISTRM,LZPL,LSLKG,IBEGIN,LZN,LQLSUM,NELAP,
     & LITRIA,DCFM)
C                                                                       C
C-----VERSION    1  1JUL1991 BRC1AL                                    C
C     ***************************************************************C
C     ALLOCATE ARRAY STORAGE FOR BRANCH                                 C
C     ***************************************************************C
C     ****************************************************************
C     * THIS SUBROUTINE ALLOCATES SPACE IN THE THREE MAIN ARRAYS IN  *
C     * BRANCH. THE ARRAY Y CONTAINS REAL AND INTEGER VECTORS, THE   *
C     * ARRAY YC CONTAINS CHARACTER VECTORS, AND THE ARRAY YL CON-   *
C     * TAINS LOGICAL VECTORS.                                       *
C     ****************************************************************
C                                                                       C
C     SPECIFICATIONS:                                                   C
C     ------------------------------------------------------------------C
C     ------------------------------------------------------------------C
C                                                                       C
C1------IDENTIFY PACKAGE AND INITIALIZE DIMENSIONS                      C
      IBEGIN=0
      NELAP=1
      WRITE(IOUT,1) IN
C       WRITE(*,*) 'DOWN TO BRC1AL'
    1 FORMAT(1H0,' BRANCH UNSTEADY FLOW MODEL, 7/30/90',
     1' INPUT READ FROM UNIT',I3)
      READ (IN,1000) MXBH,NSEC,MAXS,MXTDBC,MAXZBD,MAXCZQ,MXWIND
      READ (IN,1010) MXJN,NSEG,MXPT,MXMD,MAXQBD,MAXMZQ,DCFM
C
C2 -----SET RATIO OF TIMESCALES IN MODFLOW AND BRANCH = TFCTR, BASED
C       ON THE VALUE OF ITMUNI
C
      TFCTR=0.0
      IF (ITMUNI.EQ.1) TFCTR=1.0
      IF (ITMUNI.EQ.2) TFCTR=60.0
      IF (ITMUNI.EQ.3) TFCTR=3600.0
      IF (ITMUNI.EQ.4) TFCTR=86400.0
      IF (ITMUNI.EQ.5) TFCTR=31536000.0
      IF (TFCTR.EQ.0.0) THEN
      WRITE(*,*) ' PROGRAM ABORTED, PROPER VALUE OF ITMUNI TO
     1 CALCULATE TIMESCALE RATIOS NOT SET'
      STOP
      ENDIF
C
C3 ------ALLOCATE SPACE IN Y ARRAY
C
      ISUM=1
      LINDAT=ISUM
      ISUM=ISUM+MAXQBD
      LIZDAT=ISUM
```

Program listing—Continued

```
ISUM=ISUM+MAXZBD
LIQDAT=ISUM
ISUM=ISUM+MAXQBD
LITQMA=ISUM
ISUM=ISUM+MAXS
LITQMI=ISUM
ISUM=ISUM+MAXS
LQMAX=ISUM
ISUM=ISUM+MAXS
LQMIN=ISUM
ISUM=ISUM+MAXS
LQSUM=ISUM
ISUM=ISUM+MAXS
LZQMIN=ISUM
ISUM=ISUM+MAXS
LZQMAX=ISUM
ISUM=ISUM+MAXS
LAQMAX=ISUM
ISUM=ISUM+MAXS
LAQMIN=ISUM
ISUM=ISUM+MAXS
LA=ISUM
ISUM=ISUM+MAXS
LZ=ISUM
ISUM=ISUM+MAXS
LQ=ISUM
ISUM=ISUM+MAXS
LZP=ISUM
ISUM=ISUM+MAXS
LQP=ISUM
ISUM=ISUM+MAXS
LAP=ISUM
ISUM=ISUM+MAXS
LBP=ISUM
ISUM=ISUM+MAXS
LRP=ISUM
ISUM=ISUM+MAXS
LB=ISUM
ISUM=ISUM+MAXS
LR=ISUM
ISUM=ISUM+MAXS
LBT=ISUM
ISUM=ISUM+MAXS
LBTP=ISUM
ISUM=ISUM+MAXS
LXSTAT=ISUM
ISUM=ISUM+MAXS
LDX=ISUM
ISUM=ISUM+MAXS
LT=ISUM
ISUM=ISUM+MAXS
LRN=ISUM
ISUM=ISUM+MAXS*4
LWANGL=ISUM
ISUM=ISUM+MAXS
LGDATU=ISUM
ISUM=ISUM+MAXS
LORIEN=ISUM
ISUM=ISUM+MAXS
LBETVE=ISUM
```

Program listing—Continued

```
ISUM=ISUM+MAXS
LSUMET=ISUM
ISUM=ISUM+MAXS
LSUMCZ=ISUM
ISUM=ISUM+MAXS
LSCZQS=ISUM
ISUM=ISUM+MAXS
LSZQET=ISUM
ISUM=ISUM+MAXS
LZA=ISUM
ISUM=ISUM+MXPT*MAXS
LAA=ISUM
ISUM=ISUM+MXPT*MAXS
LBB=ISUM
ISUM=ISUM+MXPT*MAXS
LBS=ISUM
ISUM=ISUM+MXPT*MAXS
LIPT=ISUM
ISUM=ISUM+MAXS
LQA=ISUM
ISUM=ISUM+MXPT*MAXS
LTA=ISUM
ISUM=ISUM+MXPT*MAXS
LETA=ISUM
ISUM=ISUM+MXPT*MAXS
LFUNET=ISUM
ISUM=ISUM+MXPT*MAXS
LROW=ISUM
ISUM=ISUM+MXBH*4
LAM=ISUM
ISUM=ISUM+((MXBH*4)**2)
LBMX=ISUM
ISUM=ISUM+MXBH*4
LIJF=ISUM
ISUM=ISUM+MXBH
LIJT=ISUM
ISUM=ISUM+MXBH
LNSEC=ISUM
ISUM=ISUM+MXBH
LXSKT=ISUM
ISUM=ISUM+MXBH
LPLTBC=ISUM
ISUM=ISUM+MXBH
LPRTXS=ISUM
ISUM=ISUM+MXBH
LPRTBC=ISUM
ISUM=ISUM+MXBH
LPRTSU=ISUM
ISUM=ISUM+MXBH
LPPLTB=ISUM
ISUM=ISUM+MXBH
LIBJNC=ISUM
ISUM=ISUM+MXJN
LNDATA=ISUM
ISUM=ISUM+MXJN
LIZQBV=ISUM
ISUM=ISUM+MXJN
LISTAT=ISUM
ISUM=ISUM+MXJN
LKTTDB=ISUM
```

Program listing—Continued

```
      ISUM=ISUM+1
      LZQ=ISUM
      ISUM=ISUM+MAXZBD*MXTDBC
      LDTT=ISUM
      ISUM=ISUM+MXJN
      LDATUM=ISUM
      ISUM=ISUM+MXJN
      LZQBVC=ISUM
      ISUM=ISUM+MXJN*4
      LZQPMI=ISUM
      ISUM=ISUM+MXJN
      LCLK=ISUM
      ISUM=ISUM+MAXS
      LZBOT=ISUM
      ISUM=ISUM+MAXS
      LWINDS=ISUM
      ISUM=ISUM+MXWIND
      LWINDD=ISUM
      ISUM=ISUM+MXWIND
      LIDX=ISUM
      ISUM=ISUM+MXJN*MXBH
      LICT=ISUM
      ISUM=ISUM+MXJN
      LW=ISUM
      ISUM=ISUM+MXJN
      LU=ISUM
      ISUM=ISUM+2*MAXS
      LUU=ISUM
      ISUM=ISUM+4*MAXS
      LBU=ISUM
      ISUM=ISUM+2*MXBH
      LBUU=ISUM
      ISUM=ISUM+4*MXBH
      LZSAV=ISUM
      ISUM=ISUM+MAXS
      LQSAV=ISUM
      ISUM=ISUM+MAXS
      LZPSAV=ISUM
      ISUM=ISUM+MAXS
      LQPSAV=ISUM
      ISUM=ISUM+MAXS
      LISTRM=ISUM
      ISUM=ISUM+3*MAXS
      LZPL=ISUM
      ISUM=ISUM+MAXS
      LSLKG=ISUM
      ISUM=ISUM+MAXS
      LZN=ISUM
      ISUM=ISUM+MAXS
      LQLSUM=ISUM
      ISUM=ISUM+MAXS+MXJN
      LITRIA=ISUM
      ISUM=ISUM+MAXS*MAXCZQ
C                                                                       C
C3A ----PRINT AMOUNT OF SPACE USED BY THE Y VECTOR FOR BRANCH.          C
      ISUM1=ISUM-1
      WRITE(IOUT,9)ISUM1,LENY
    9 FORMAT(1X,I8,' ELEMENTS OF Y ARRAY USED OUT OF',I7)
      IF(ISUM1.GT.LENY) WRITE(IOUT,10)
   10 FORMAT(1X,'   ***Y ARRAY MUST BE DIMENSIONED LARGER***')
```

Program listing—Continued

```
C
C4 -----ALLOCATE SPACE IN YC ARRAY
C
      ISUM=1
      LITYPO=ISUM
      ISUM=ISUM+MAXS*4
      LBRNAM=ISUM
      ISUM=ISUM+MXBH*10
      LITYPE=ISUM
      ISUM=ISUM+MXJN
C                                                                      C
C4A ----PRINT AMOUNT OF SPACE USED BY THE YC VECTOR FOR BRANCH.        C
      ISUM1=ISUM-1
      WRITE(IOUT,19)ISUM1,LENYC
   19 FORMAT(1X,I8,' ELEMENTS OF YC ARRAY USED OUT OF',I7)
      IF(ISUM1.GT.LENYC) WRITE(IOUT,20)
   20 FORMAT(1X,'    ***YC ARRAY MUST BE DIMENSIONED LARGER***')
C
C5 -----ALLOCATE SPACE IN YL ARRAY
C
      ISUM=1
      LLARBP=ISUM
      ISUM=ISUM+MAXS
      LZHIGH=ISUM
      ISUM=ISUM+MAXS
      LZLOW=ISUM
      ISUM=ISUM+MAXS
      LLINPR=ISUM
      ISUM=ISUM+MAXS
      LARBER=ISUM
      ISUM=ISUM+1
C                                                                      C
C5A ----PRINT AMOUNT OF SPACE USED BY THE YL VECTOR FOR BRANCH.        C
      ISUM1=ISUM-1
      WRITE(IOUT,29)ISUM1,LENYL
   29 FORMAT(1X,I8,' ELEMENTS OF YL ARRAY USED OUT OF',I7)
      IF(ISUM1.GT.LENYL) WRITE(IOUT,30)
   30 FORMAT(1X,'    ***YL ARRAY MUST BE DIMENSIONED LARGER***')
      RETURN
 1000 FORMAT(7I10)
 1010 FORMAT(6I10,1F10.3)
      END
```

LIST OF VARIABLES

Variable	Range	Definition
DCFM	Package	Multiplier for the friction term in the momentum equation when the channel runs dry.
IBEGIN	Package	Flag indicating if BRANCH' is being called for the first time (0, first time; 1, not first time).
IN	Package	Primary unit number for all printed output.
IOUT	Global	Primary unit number for all printed output.
ISUM	Module	Index number of lowest element in Y, YC, or YL array that has not yet been allocated.
ISUM1	Module	ISUM minus 1.
ITMUNI	Global	Time units used by MODFLOW (0, undefined; 1, seconds; 2, minutes; 3, hours; 4, days; and 5, years).
L??	Package	Location in the Y, YC, or YL array of the first element of BRANCH variable ??.
MAXCZQ	Package	Maximum number of daily computed results held in storage for plotting purposes.
MAXMZQ	Package	Maximum number of measured data held in storage for plotting purposes.
MAXQBD	Package	Maximum number of discharge boundary value data held in storage for computation.
MAXS	Package	Maximum number of cross sections in the entire channel network.
MAXZBD	Package	Maximum number of stage boundary value data held in storage for computation.
MXBH	Package	Maximum number of branches in the network.
MXJN	Package	Maximum number of junctions in the network.
MXMD	Package	Maximum number of measured data locations accommodated in the network.
MXPT	Package	Maximum number of points used to define a cross section.
MXTDBC	Package	Maximum number of boundaries in the network.
MXWIND	Package	Maximum number of wind data points input.
NELAP	Package	Number of elapsed MODFLOW time steps since beginning of simulation.
NSEC	Package	Maximum cross sections per branch.
NSEG	Package	Maximum segments per branch.
TFCTR	Package	Number of time steps in one time unit in MODFLOW.

MODULE BRC1RP

This module consists of the first five sections of the original BRANCH program, which handle the reading of data used in BRANCH' and is called only once at the beginning of the simulation. The numbered sections in the narrative correspond to the numbered steps in the flowchart for BRC1RP in figure 7. The flowchart is divided with the original code to the left of the dotted line and the modifications made to interface with MODFLOW placed to the right of the dotted line. The modifications are signified with a number and letter (the original code just by number).

NARRATIVE

1. Read program control parameters and assign defaults. This is done by calling the BRANCH subroutine RDCOMP, which reads basic information such as name of network; unit; number of branches, junctions, and boundaries; time-interval lengths; convergence criterion; plot and print options; weighting factors; and wind-drag information. Details are given by Schaffranek and others (1981). Default values are assigned where values are not given.
2. Read branch identification, initial values, and geometry data. This is done by calling the BRANCH subroutine BRICXS. BRICXS reads information for each branch such as inlet and outlet junctions, number of cross sections, branch name, initial conditions, length of segments, water temperature, wind direction, momentum coefficient, and cross-sectional geometry data. Details on input structures are given by Schaffranek and others (1981).

2a. Read aquifer location, riverbed elevation, and leakage coefficient. This addition to the BRICXS subroutine reads the layer, row, and column of the aquifer model cell that contains the river segment. This is followed by the leakage coefficient K'/b', defined by the FORTRAN variable CLK (segment number) and the riverbed elevation Z_{BOT}, defined by the FORTRAN variable ZBOT (cross-section number). The format is described in the data entry section.

2b. Extrapolate cross-sectional geometry. The channel can run dry if cross-sectional geometry is defined in the hypothetical zone below the river bottom. Thus, three additional cross-sectional geometry points are added below the true riverbed to define geometry to a distance below the river-bed equal to the distance from the highest defined cross-sectional geometry point down to the riverbed. The same topwidth is maintained in this extrapolated zone as was used for the lowest entered data point, and the cross-sectional area is reduced to 31 percent of the lowest entered data point. Although this geometry below the riverbed could carry substantial flow, the DCFM factor described previously increases

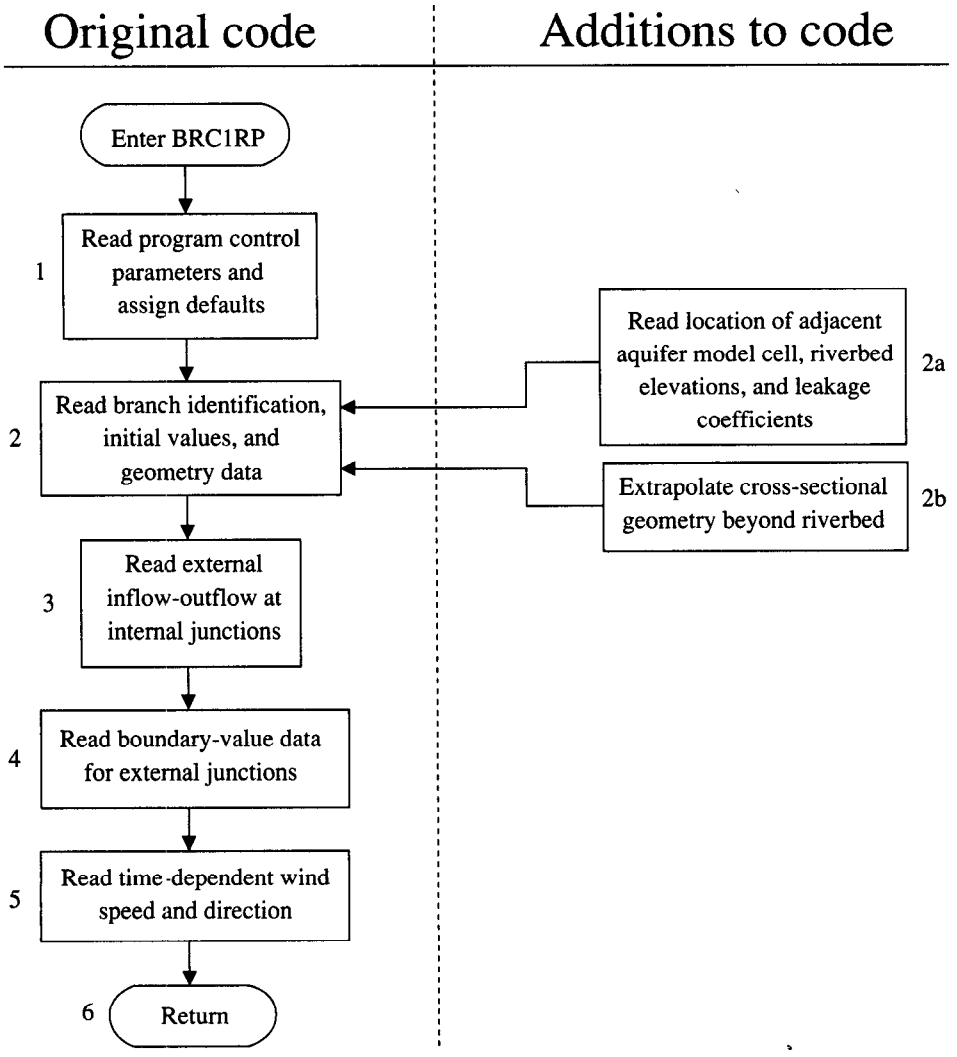

Figure 7. Flowchart of data entry module (BRC1RP).

the resistance in this zone to create flows low enough to approximate a dry condition.

3. Read junction inflows and outflows. External outflows (or inflows) at an internal junction are read according to Schaffranek and others (1981).
4. Read boundary-value data for external junctions. The BRANCH subroutine BVDINP is called to read the type of boundary data, location of boundary, time scale of boundary data, datum corrections, and boundary data points according to Schaffranek and others (1981).
5. Read wind speed and direction. The wind velocities and directions are input the same as the boundary value data (Schaffranek and others, 1981).
6. Return to MODFLOW.

PROGRAM LISTING FOR MODULE BRC1RP

```
      SUBROUTINE BRC1RP(INDATA,IZDATA,IQDATA,ITQMAX,ITQMIN,
     1 QMAX,QMIN,QSUM,ZQMIN,ZQMAX,AQMAX,AQMIN,A,Z,Q,ZP,QP,
     2 AP,BP,RP,B,R,BT,BTP,XSTATN,DX,T,RN,WANGLE,GDATUM,ORIENT
     3 ,BETVEL,SUMETA,SUMCZQ,SCZQSQ,SZQETA,ITYPEO,ZA,AA,BB,BS,IPT,
     4 QA,TA,ETA,FUNETA,BRNAME,IJF,IJT,NSEC,XSKT,
     5 PLTBCH,PRTXSG,PRTBCH,PRTSUM,PPLTBH,ITYPE,IBJNC,NDATA,IZQBVE,
     6 ISTATN,KTTDBC,ZQ,DTT,DATUM,ZQBVCO,ZQPMIN,LARBPR,ZHIGH,
     7 ZLOW,LINPRT,ARBERR,CLK,ZBOT,
     8 MXBH,MXJN,MAXS,MXPT,MXTDBC,MXMD,MAXZBD,
     9 MAXQBD,MAXCZQ,MAXMZQ,MXWIND,MAXBD,MXOTDT,WINDSP,WINDDR,
     & ICT,W,IN,IOUT,ISTRM)
C                                                                       C
C-----VERSION   1 1APR1991  BRC1RP                                      C
C     ***************************************************************
C     * READ INPUT DATA FOR BRANCH PROGRAM AND MODFLOW INTERFACE    *
C     ***************************************************************
C BEGIN COMMON_COMCON ==========================================================
C
      CHARACTER*2 IUNIT,OUNIT
      INTEGER*4 NBCH,NJNC,NBND,NSTEPS,IRDGEO,NIT,IPROPT,IPLOPT,IPLDEV,
     1 IPRMSG,IPLMSG,IEXOPT,INHR,INMN,IDTM,IWRTIC,IRDIC,NUMCOM,INWIND,
     2 TYPETA,OTTDDB,ISMOPT,NTDIOF,IRDNXT,IARDEM
      REAL THETA,QQTOL,ZZTOL,WSPEED,WDIREC,WSDRAG,H2ODEN,CHI,QZCONV,
     1 ZDATUM,DT,G,AIRDEN,GLETA,GLBETA,ETAMIN,ETAMAX,TOLERR
      COMMON /COMCON/ NBCH,NJNC,NBND,NSTEPS,IRDGEO,NIT,IPROPT,
     1 IPLOPT,IPLDEV,IPRMSG,IPLMSG,IEXOPT,INHR,INMN,IDTM,IWRTIC,
     2 IRDIC,NUMCOM,INWIND,THETA,QQTOL,ZZTOL,WSPEED,WDIREC,WSDRAG,
     3 H2ODEN,CHI,ZDATUM,IUNIT,OUNIT,TYPETA,OTTDDB,ISMOPT,G,QZCONV,DT,
     4 AIRDEN,IARDEM,NTDIOF,IRDNXT,GLETA,GLBETA,ETAMIN,ETAMAX,TOLERR
C
C  END  COMMON_COMCON ============================================================
C BEGIN  COMMON_DTYPES ============================================================
C
      CHARACTER*2 DTYPE,ZTYPE,QTYPE,ATYPE,BTYPE,ZPTYPE,QPTYPE,DPTYPE
      COMMON /DTYPES/ DTYPE,ZTYPE,QTYPE,ATYPE,BTYPE,ZPTYPE,QPTYPE,DPTYPE
C
C  END   COMMON_DTYPES ============================================================
C BEGIN COMMON_UNITS ============================================================
C
      CHARACTER*2 IBLK,UNIT,EN,ME,MT,FT,TUNIT,DC
      COMMON /UNITS/ IBLK,UNIT,EN,ME,MT,FT,TUNIT,DC
C
C  END  COMMON_UNITS ============================================================
C BEGIN COMMON_LUNUMS ============================================================
C
      INTEGER*4 READER,PRINTR,PUNCH,DSREF,TDDATA,LUPTRK,LUIFLO,LUIVOL,
     1         LUGEOM,LUINIT,LUCVOL
      COMMON /LUNUMS/ READER,PRINTR,PUNCH,DSREF,TDDATA,LUPTRK,LUIFLO,
     1 LUIVOL,LUGEOM,LUINIT,LUCVOL
C
C  END  COMMON_LUNUMS ============================================================
C BEGIN COMMON_DADCOM ============================================================
C
      INTEGER*2 LISTB,LISTA,STRIP,RTCODE
      COMMON /DADCOM/ LISTB,LISTA,STRIP,RTCODE
C
C  END  COMMON_DADCOM ============================================================
C BEGIN COMMON_DAYPMO ============================================================
C
      INTEGER*2 DPERM(12)
```

Program listing—Continued

```
      COMMON /DAYPMO/ DPERM
C
C  END  COMMON_DAYPMO ================================================
C BEGIN COMMON_LOGICS ================================================
C
      LOGICAL*4 PRTMSG, NOCONV, ERROR, OPLOTS, FOUND, NOEXTP,
     1 NOPRIT, DAYSUM, MOREBD, DTPRT, PTPLT, DAOPEN, STAGES, MODETA
      COMMON /LOGICS/ PRTMSG, NOCONV, ERROR, OPLOTS, FOUND, NOEXTP,
     1 NOPRIT, DAYSUM, MOREBD, DTPRT, PTPLT, DAOPEN, STAGES, MODETA
C
C  END  COMMON_LOGICS ================================================
C BEGIN COMMON_BCTIME ================================================
C
      INTEGER*4 IETIME,NETIME
      INTEGER *2 IRDPDY,IYR,IMO,IDA,IHR,IMN,NYR,NMO,NDA,NHR,NMN
      COMMON /BCTIME/ IETIME,NETIME,IRDPDY,IYR,IMO,IDA,IHR,IMN,
     1                                     NYR,NMO,NDA,NHR,NMN
C
C  END  COMMON_BCTIME ================================================
C BEGIN COMMON_DATIME ================================================
C
      INTEGER*4 KYR,KMO,KDA,KHR,KMN,M,KYRS,KMOS,KDAS,KHRS,KMNS
      COMMON /DATIME/ KYR,KMO,KDA,KHR,KMN,M,KYRS,KMOS,KDAS,KHRS,KMNS
C
C  END  COMMON_DATIME ================================================
C BEGIN COMMON_NETWRK ================================================
C
      CHARACTER*80 NETNAM
      COMMON /NETWRK/ NETNAM
C
C  END  COMMON_NETWRK ================================================
C BEGIN COMMON_MODBRCH ===============================================
C
      COMMON /MODBRCH/ TWOCSQ,IDTPDY,TWOG,CW,II,ONECHI,DCHI,DTHETA,
     1                 IBCH,IJZPBC,IJQPBC,DCFM1,KKITER
C
C  END  COMMON_MODBRCH ================================================
      INTEGER*4 MXBH,MXJN,MAXS,MXPT,MXTDBC,MXMD,MAXZBD,
     1 MAXQBD,MAXCZQ,MAXMZQ,MXWIND,MAXBD,MXOTDT
      INTEGER*4 ITQMAX(MAXS), ITQMIN(MAXS),ISTRM(3,MAXS)
      REAL QMAX(MAXS), QMIN(MAXS), QSUM(MAXS), ZQMIN(MAXS), ZQMAX(MAXS),
     1     AQMAX(MAXS), AQMIN(MAXS)
      REAL A(MAXS),Z(MAXS),Q(MAXS),ZP(MAXS),QP(MAXS),AP(MAXS),
     1 BP(MAXS),RP(MAXS),B(MAXS),R(MAXS),BT(MAXS),BTP(MAXS)
      INTEGER*4 XSTATN(MAXS)
      REAL DX(MAXS), T(MAXS), RN(4,MAXS), WANGLE(MAXS), GDATUM(MAXS),
     1 ORIENT(MAXS), BETVEL(MAXS), SUMETA(MAXS), SUMCZQ(MAXS),
     2 SCZQSQ(MAXS),SZQETA(MAXS)
      CHARACTER*4 ITYPEO(4,MAXS)
      INTEGER*4 IPT(MAXS)
      REAL ZA(MXPT,MAXS), AA(MXPT,MAXS), BB(MXPT,MAXS), BS(MXPT,MAXS)
      REAL QA(MXPT,MAXS), TA(MXPT,MAXS), ETA(MXPT,MAXS),
     1 FUNETA(MXPT,MAXS)
      CHARACTER*40 BRNAME(MXBH)
      INTEGER*4 IJF(MXBH),IJT(MXBH),NSEC(MXBH),XSKT(MXBH),
     1 PLTBCH(MXBH),PRTXSG(MXBH),PRTBCH(MXBH),PRTSUM(MXBH),PPLTBH(MXBH)
      CHARACTER*4 ITYPE(MXJN)
      INTEGER*4 IBJNC(MXJN),NDATA(MXJN),IZQBVE(MXJN)
      INTEGER*4 ISTATN(MXJN),KTTDBC
      REAL ZQ(MAXZBD,MXTDBC),DTT(MXJN),DATUM(MXJN),ZQBVCO(4,MXJN),
```

Program listing—Continued

```
     1 ZQPMIN(MXJN)
       LOGICAL*4 LARBPR(MAXS),ZHIGH(MAXS),ZLOW(MAXS),LINPRT(MAXS),ARBERR
       INTEGER*4 IZDATA(MAXZBD)
       INTEGER*4 IQDATA(MAXQBD),INDATA(MAXQBD)
C      EQUIVALENCE (INDATA(1),IQDATA(1),IZDATA(1))
       REAL CLK(MAXS),ZBOT(MAXS)
       REAL WINDSP( MXWIND),WINDDR( MXWIND)
       INTEGER*4 ICT(MXJN)
       REAL W(MXJN)
       REAL*8 C1, C2, C3, C4, UUIJP1, UUIJP2, UUIJP3, UUIJP4
       REAL LAMBDA,MU,SETA,WDTT,TWOCSQ,TWOG,CW,ONECHI,DCHI,DTHETA,TH,WIND
       INTEGER*4 IAR,I,J,K,L,II,IJ,NS,KT,IS,N,NWREAD,NWDATA,INTDBC,IDTPDY
       REAL QTOL,ZTMIN,ZTMAX,ZPMIN,QPMIN,DXMIN,DXMAX
       COMMON /LIMITS/ QTOL,ZTMIN,ZTMAX,ZPMIN,QPMIN,DXMIN,DXMAX
       INTEGER*2 JYR,JMO,JDA,JHR,JMN,MYR,MMO,MDA,MHR,MMN
       INTEGER*4 ND,NDFIRT,NDPART,JETIME,NTSAQ
       COMMON /PARTIM/ ND,NDFIRT,NDPART,JETIME,JYR,JMO,JDA,JHR,JMN,
      1MYR,MMO,MDA,MHR,MMN
       CHARACTER*80 COMENT(9)
       COMMON /CMMNT/ COMENT
       CHARACTER*2 IDETA(7)
       COMMON /ETASYM/ IDETA
       REAL WRATIO,DTZERO,ZTEMP,QTEMP,ZIJ,QIJ,DXIJ,QIJP1,ZIJP1,APZPIJ,
      1      BPZPIJ,BTZPIJ,RPZPIJ,BAVG,BTAVG,AAVG,RAVG,QAVG,ZAVG,BETCOR,
      2      RNIJ,AAVGSQ,AAVGCU,SIGMA,EPSLON,ZETA,OMEGA,GAMMA,DELTA,DET,
      3 DZDT,DQDXC,DQDT,DQDXM,DADX,DZDX,FRIC,ZQPIJ,BIGQ,BIGZ,ZTOL,SOLPDT
      4 ,ALPHA
       INTEGER*4 IBCH,IJZPBC,IJQPBC,KTMATS,LASTN,IJP1,NSM1,JP1,IJ2,IJ4,
      1  IJ4P1,IJ4P2,IJ4P3,IJ4P4,IJ2P1,IJ2P2,I2,I4,I4P1,I4P2,I4P3,I4P4,
      2  I2P1,I2P2,NN,MM,NNN,NBPJ,M0,IBIGZ,JBIGZ,IBIGQ,JBIGQ,IJPNS,ICHK
C
C1----READ PROGRAM CONTROL PARAMETERS AND ASSIGN DEFAULTS
C
       CALL RDCOMP (MXBH,MXJN,MAXS,MXPT,MXTDBC,MXMD,MAXZBD,
     1 MAXQBD,MAXCZQ,MAXMZQ,MXWIND,MAXBD,MXOTDT,
     & ITQMAX,ITQMIN,QMAX,QMIN,QSUM,ZQMIN,ZQMAX,AQMAX,AQMIN,IN)
C
C2----READ BRANCH IDENTIFICATION, INITIAL VALUES, AND GEOMETRY DATA
C
       CALL BRICXS (CLK,ZBOT,MXBH,MXJN,MAXS,MXPT,MXTDBC,MXMD,
     1 MAXZBD,MAXQBD,MAXCZQ,MAXMZQ,MXWIND,MAXBD,MXOTDT,
     3 A,Z,Q,ZP,QP,AP,BP,RP,B,R,BT,BTP,
     & XSTATN,DX,T,RN,WANGLE,GDATUM,ORIENT,BETVEL,SUMETA,SUMCZQ,
     & SCZQSQ,SZQETA,ITYPEO,ZA,AA,BB,BS,IPT,
     & QA,TA,ETA,FUNETA,
     & BRNAME,IJF,IJT,NSEC,XSKT,
     & PLTBCH,PRTXSG,PRTBCH,PRTSUM,PPLTBH,
     & LARBPR,ZHIGH,ZLOW,LINPRT,ARBERR,IN,ISTRM)
C
C3----READ EXTERNAL INFLOW/OUTFLOW AT INTERNAL JUNCTIONS
C
  130 CONTINUE
       WRITE(*,'(A/)') ' Read constant nodal flow at internal junctions!'
       READ (IN,1750) (W(J),J=1,NJNC)
C
C4----READ BOUNDARY-VALUE DATA FOR EXTERNAL JUNCTIONS
C
       READ (IN,1760) LISTB,LISTA
       CALL BVDINP (MXBH,MXJN,MAXS,MXPT,MXTDBC,MXMD,MAXZBD,
      1 MAXQBD,MAXCZQ,MAXMZQ,MXWIND,MAXBD,MXOTDT,
```

Program listing—Continued

```
     & BRNAME,IJF,IJT,NSEC,XSKT,
     & PLTBCH,PRTXSG,PRTBCH,PRTSUM,PPLTBH,
     & ITYPE,IBJNC,NDATA,IZQBVE,ISTATN,
     & ZQ,DTT,DATUM,ZQBVCO,ZQPMIN,KTTDBC,
     & INDATA,IZDATA,IQDATA,IN)
C
C5----READ TIME-DEPENDENT WIND SPEED AND DIRECTION
C
      NWREAD=0
      IF (INWIND.EQ.0) GO TO 440
      WRITE(*,'(A/)') ' Read time-varying wind conditions!'
      READ (IN,1780,END=640) NWDATA,WDTT,NWREAD
      IF (WDTT.EQ.0.0.AND.NWREAD.EQ.0) GO TO 1450
      IF (NWREAD.EQ.0) NWREAD=1440./WDTT
      IF (WDTT.EQ.0.0) WDTT=1440./NWREAD
      IF (NWREAD.NE.1440./WDTT) GO TO 1450
      IF (NWDATA.LT.1.OR.NWDATA.GT.MXWIND) THEN
         WRITE (IOUT,2020) NWDATA
         STOP
      ENDIF
      WDTT=WDTT*60.
      READ (IN,1790) (WINDSP(K),WINDDR(K),K=1,NWDATA)
  440 CONTINUE
 1450 CONTINUE
  640 CONTINUE
C6------RETURN
      RETURN
C
C     INPUT/OUTPUT FORMAT STATEMENTS
C
 1740 FORMAT (2(A2,3X,2(I2,1X)))
 1750 FORMAT (10F8.2)
 1760 FORMAT (37X,I2,6X,I2)
 1780 FORMAT (4X,I3,F2.0,52X,I4)
 1790 FORMAT (8F10.3)
 1920 FORMAT (' ***** ERROR,     INITIAL STAGE VALUE UNSPECIFIED IN BRAN
     1CH ',I2,' SECTION ',I2,' *****')
 1960 FORMAT (' ***** ERROR,     MATRIX NOT SQUARE : REVIEW SCHEMATIZATI
     1ON AND EXTERNAL BOUNDARY-CONDITION SPECIFICATIONS *****')
 1980 FORMAT (' ***** ERROR,     WIND DATA FREQUENCY UNSPECIFIED OR INCO
     1NSISTENT *****')
 1990 FORMAT (' ***** ERROR,     MATRIX IS SINGULAR *****')
 2020 FORMAT (' ***** ERROR,     IMPROPER NUMBER OF WIND DATA SPECIFIED
     1(1<=NWDATA<=',I5,') *****')
 2050 FORMAT (' ***** ERROR,     INITIAL VALUE(S) OUT OF DEFINED RANGE O
     1F CHANNEL GEOMETRY FOR BRANCH ',I2,' SECTION ',I2,' *****')
 2075 FORMAT ('+',T69,'(STEEP SLOPE CHANNEL NEEDS ADDITIONAL A(Z) FOR DA
     1/DX TERM)')
 2080 FORMAT (/' ',11X,A2,'(J,I),J=1,NSEC(I)')
 2090 FORMAT (' ',1X,A2,'( J,',I2,')',12F10.4/11X,12F10.4)
 2110 FORMAT (' WARNING,   MAXIMUM ITERATIONS EXCEEDED AT ',I2,':',I2,'
     1ON ',I2,'/',I2,'/',I2,'     Z-ZP(',I2,',',I2,')=',F7.4,' Q-QP(',I2
     2,',',I2,')=',F7.1)
 2140 FORMAT (/' NUMBER OF SOLUTIONS = ',I5/)
 2145 FORMAT (' SOLUTIONS/TIME STEP = ',F5.1)
 2150 FORMAT (' Z=',E13.6,' Q=',E13.6,' QT=',E13.6,' QX=',E13.6,' AX=',E
     113.6,' ZX=',E13.6,' F=',E13.6,' W=',E13.6)
 2160 FORMAT (1X,8I16)
 2170 FORMAT (1X,8E16.8)
      END
```

LIST OF VARIABLES

(Variables specific to original BRANCH code not included)

Variable	Range	Definition
CLK (MAXS)	Package	Value of *K'/b'* (leakage coefficient).
ISTRM (3, MAXS)	Package	Vector describing row, column, and layer of aquifer model cell that corresponds to channel segment.
MAXCZQ	Package	Maximum number of daily computed results held in storage for plotting purposes.
MAXMZQ	Package	Maximum number of measured data held in storage for plotting purposes.
MAXQBD	Package	Maximum number of discharge boundary value data held in storage for computation.
MAXS	Package	Maximum number of cross sections in the entire channel network.
MAXZBD	Package	Maximum number of stage boundary value data held in storage for computation.
MXBH	Package	Maximum number of branches in the network.
MXJN	Package	Maximum number of junctions in the network.
MXMD	Package	Maximum number of measured data locations accommodated in the network.
MXPT	Package	Maximum number of points used to define a cross section.
MXTDBC	Package	Maximum number of boundaries in the network.
MXWIND	Package	Maximum number of wind data points input.
NCOL	Global	Maximum number of MODFLOW aquifer columns.
NLAY	Global	Maximum number of MODFLOW aquifer layers.
NROW	Global	Maximum number of MODFLOW aquifer rows.
NSEC	Package	Maximum cross sections per branch.
NSEG	Package	Maximum segments per branch.
ZBOT (MAXS)	Package	Elevation of channel bottom.

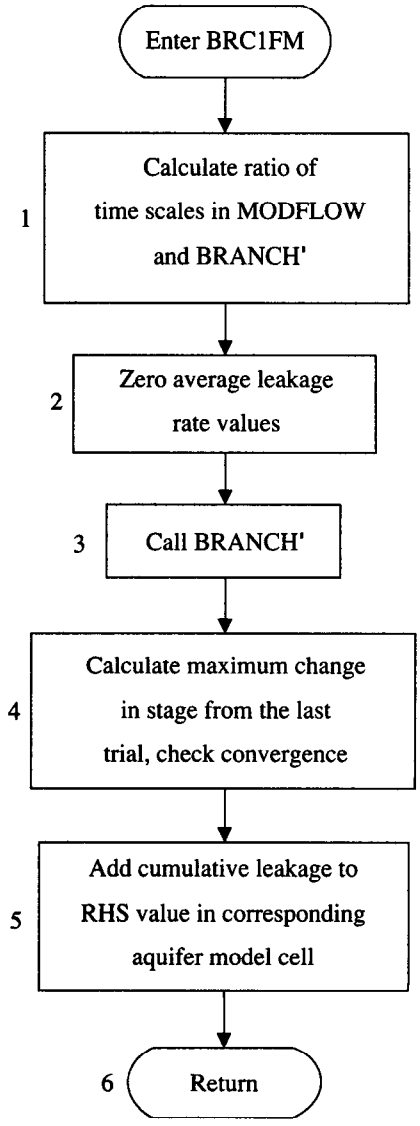

Figure 8. Flowchart of formulation module (BRC1FM).

MODULE BRC1FM

The module BRC1FM is executed when MODFLOW iterates on a time-step solution. This module determines the number of time intervals that BRANCH' calculates to equal the time-step length in MODFLOW, executes BRANCH', and adds the leakage flows calculated in BRANCH' to the MODFLOW calculations. In addition, this module checks the maximum change in river stage from the last MODFLOW iteration to ascertain whether the models have converged. The numbered sections in the narrative correspond to the numbered steps in the flowchart for BRC1FM in figure 8.

NARRATIVE

1. Calculate ratio of time scales in MODFLOW and BRANCH'. Using the ratio of time units (TFCTR), the MODFLOW time-step length (DELT), and the BRANCH' time-interval length in minutes (IDTM), the number of BRANCH' time intervals passing in one MODFLOW time step is calculated (NTSAQ). This computation is made for each MODFLOW time step because if a new stress period is entered, the MODFLOW time step can change and NTSAQ can change. If the MODFLOW time step is not an integral multiple of

the BRANCH' time interval, an error message will print. For steady state, NTSAQ is set to one.

2. Zero average leakage rate values. Before BRANCH' is executed, the average leakage rates for each segment (QLSUM [segment number]) are set to zero so only the leakages for the following BRANCH' run will be saved.
3. Call BRANCH'. BRANCH' is executed for NTSAQ time intervals.
4. Calculate maximum change in stage from last MODFLOW iteration. Using the stage values saved from the final BRANCH' time interval in the last MODFLOW iteration (ZPL [node number]), the difference between these values and the stages in the new iteration (ZN [node number]) is calculated to see if the convergence criteria have been met. If so, a convergence flag is set.
5. Add cumulative leakages to RHS value in aquifer model cell. The corresponding aquifer model cell is located for each river segment. The average leakage rates from the MODFLOW time step (QLSUM [segment number]) are then multiplied by TFCTR to convert from BRANCH' time units to MODFLOW time units. The QLSUM values are added (subtract flow out) to the RHS value in the MODFLOW equations for the aquifer model cell. If NTSAQ is 1, the leakage values are calculated and added to the HCOEF and RHS values instead of adding QLSUM to RHS.
6. Return to MODFLOW.

PROGRAM LISTING FOR MODULE BRC1FM

```
      SUBROUTINE BRC1FM(INDATA,IZDATA,IQDATA,ITQMAX,ITQMIN,
     1 QMAX,QMIN,QSUM,ZQMIN,ZQMAX,AQMAX,AQMIN,A,Z,Q,ZP,QP,
     2 AP,BP,RP,B,R,BT,BTP,XSTATN,DX,T,RN,WANGLE,GDATUM,ORIENT
     3 ,BETVEL,SUMETA,SUMCZQ,SCZQSQ,SZQETA,ITYPEO,ZA,AA,BB,BS,IPT,
     4 QA,TA,ETA,FUNETA,ROW,AM,BMX,BRNAME,IJF,IJT,NSEC,XSKT,
     5 PLTBCH,PRTXSG,PRTBCH,PRTSUM,PPLTBH,ITYPE,IBJNC,NDATA,IZQBVE,
     6 ISTATN,KTTDBC,ZQ,DTT,DATUM,ZQBVCO,ZQPMIN,LARBPR,ZHIGH,
     7 ZLOW,LINPRT,ARBERR,CLK,ZBOT,
     8 MXBH,MXJN,MAXS,MXPT,MXTDBC,MXMD,MAXZBD,
     9 MAXQBD,MAXCZQ,MAXMZQ,MXWIND,MAXBD,MXOTDT,WINDSP,WINDDR,
     & IDX,ICT,W,U,UU,BU,BUU,ZSAV,QSAV,ZPSAV,QPSAV,NELAP,IOUT,ZPL,SLKG,
     & HNEW,HOLD,HCOF,RHS,IBOUND,NCOL,NROW,NLAY,TFCTR,ISS,NTSAQ,
     2 DELT,TOTIM,IBCONV,HCLOSE,ISTRM,IBEGIN,ZN,QLSUM,ITRIAL,DCFM,KKSTP)
C                                                                       C
C-----VERSION   1 1APR1991  BRC1FM                                      C
C     ***************************************************************C
C     CALL BRANCH MODEL AND ADD LEAKAGE TERMS TO RHS QUANTITY           C
C     ***************************************************************C
C                                                                       C
C     SPECIFICATIONS:                                                   C
C     ------------------------------------------------------------------C
C                                                                       C
C BEGIN COMMON_COMCON ================================================
C
      CHARACTER*2 IUNIT,OUNIT
      INTEGER*4 NBCH,NJNC,NBND,NSTEPS,IRDGEO,NIT,IPROPT,IPLOPT,IPLDEV,
     1 IPRMSG,IPLMSG,IEXOPT,INHR,INMN,IDTM,IWRTIC,IRDIC,NUMCOM,INWIND,
     2 TYPETA,OTTDDB,ISMOPT,NTDIOF,IRDNXT,IARDEM
      REAL THETA,QQTOL,ZZTOL,WSPEED,WDIREC,WSDRAG,H2ODEN,CHI,QZCONV,
     1 ZDATUM,DT,G,AIRDEN,GLETA,GLBETA,ETAMIN,ETAMAX,TOLERR
      COMMON /COMCON/ NBCH,NJNC,NBND,NSTEPS,IRDGEO,NIT,IPROPT,
     1 IPLOPT,IPLDEV,IPRMSG,IPLMSG,IEXOPT,INHR,INMN,IDTM,IWRTIC,
     2 IRDIC,NUMCOM,INWIND,THETA,QQTOL,ZZTOL,WSPEED,WDIREC,WSDRAG,
     3 H2ODEN,CHI,ZDATUM,IUNIT,OUNIT,TYPETA,OTTDDB,ISMOPT,G,QZCONV,DT,
     4 AIRDEN,IARDEM,NTDIOF,IRDNXT,GLETA,GLBETA,ETAMIN,ETAMAX,TOLERR
C
C  END  COMMON_COMCON ================================================
C BEGIN  COMMON_DTYPES ================================================
C
      CHARACTER*2 DTYPE,ZTYPE,QTYPE,ATYPE,BTYPE,ZPTYPE,QPTYPE,DPTYPE
      COMMON /DTYPES/ DTYPE,ZTYPE,QTYPE,ATYPE,BTYPE,ZPTYPE,QPTYPE,DPTYPE
C
C  END   COMMON_DTYPES ================================================
C BEGIN COMMON_UNITS ==================================================
C
      CHARACTER*2 IBLK,UNIT,EN,ME,MT,FT,TUNIT,DC
      COMMON /UNITS/ IBLK,UNIT,EN,ME,MT,FT,TUNIT,DC
C
C  END  COMMON_UNITS ==================================================
C BEGIN COMMON_LUNUMS =================================================
C
      INTEGER*4 READER,PRINTR,PUNCH,DSREF,TDDATA,LUPTRK,LUIFLO,LUIVOL,
     1        LUGEOM,LUINIT,LUCVOL
      COMMON /LUNUMS/ READER,PRINTR,PUNCH,DSREF,TDDATA,LUPTRK,LUIFLO,
     1 LUIVOL,LUGEOM,LUINIT,LUCVOL
C
C  END  COMMON_LUNUMS =================================================
C BEGIN COMMON_DADCOM =================================================
C
      INTEGER*2 LISTB,LISTA,STRIP,RTCODE
```

Program listing—Continued

```
      COMMON /DADCOM/ LISTB,LISTA,STRIP,RTCODE
C
C  END  COMMON_DADCOM ================================================
C BEGIN COMMON_DAYPMO ================================================
C
      INTEGER*2 DPERM(12)
      COMMON /DAYPMO/ DPERM
C
C  END  COMMON_DAYPMO ================================================
C BEGIN COMMON_LOGICS ================================================
C
      LOGICAL*4 PRTMSG, NOCONV, ERROR, OPLOTS, FOUND, NOEXTP,
     1 NOPRIT, DAYSUM, MOREBD, DTPRT, PTPLT, DAOPEN, STAGES, MODETA
      COMMON /LOGICS/ PRTMSG, NOCONV, ERROR, OPLOTS, FOUND, NOEXTP,
     1 NOPRIT, DAYSUM, MOREBD, DTPRT, PTPLT, DAOPEN, STAGES, MODETA
C
C  END  COMMON_LOGICS ================================================
C BEGIN COMMON_BCTIME ================================================
C
      INTEGER*4 IETIME,NETIME
      INTEGER *2 IRDPDY,IYR,IMO,IDA,IHR,IMN,NYR,NMO,NDA,NHR,NMN
      COMMON /BCTIME/ IETIME,NETIME,IRDPDY,IYR,IMO,IDA,IHR,IMN,
     1                                     NYR,NMO,NDA,NHR,NMN
C
C  END  COMMON_BCTIME ================================================
C BEGIN COMMON_DATIME ================================================
C
      INTEGER*4 KYR,KMO,KDA,KHR,KMN,M,KYRS,KMOS,KDAS,KHRS,KMNS
      COMMON /DATIME/ KYR,KMO,KDA,KHR,KMN,M,KYRS,KMOS,KDAS,KHRS,KMNS
C
C  END  COMMON_DATIME ================================================
C BEGIN COMMON_NETWRK ================================================
C
      CHARACTER*80 NETNAM
      COMMON /NETWRK/ NETNAM
C
C  END  COMMON_NETWRK ================================================
C BEGIN COMMON_MODBRCH ===============================================
C
      COMMON /MODBRCH/ TWOCSQ,IDTPDY,TWOG,CW,II,ONECHI,DCHI,DTHETA,
     1                          IBCH,IJZPBC,IJQPBC,DCFM1,KKITER
C
C  END  COMMON_MODBRCH ================================================
      INTEGER*4 MXBHMMXJN,MAXS,MXPT,MXTDBC,MXMD,MAXZBD,
     1 MAXQBD,MAXCZQ,MAXMZQ,MXWIND,MAXBD,MXOTDT
      INTEGER*4 ITQMAX(MAXS), ITQMIN(MAXS),ISTRM(3,MAXS),ITRIAL(MAXS,
     1 MAXCZQ)
      REAL QMAX(MAXS), QMIN(MAXS), QSUM(MAXS), ZQMIN(MAXS), ZQMAX(MAXS),
     1     AQMAX(MAXS), AQMIN(MAXS)
      REAL A(MAXS),Z(MAXS),Q(MAXS),ZP(MAXS),QP(MAXS),AP(MAXS),
     1 BP(MAXS),RP(MAXS),B(MAXS),R(MAXS),BT(MAXS),BTP(MAXS),
     2 ZSAV(MAXS),QSAV(MAXS),ZPSAV(MAXS),QPSAV(MAXS),ZPL(MAXS),
     3 ZN(MAXS)
      INTEGER*4 XSTATN(MAXS)
      REAL DX(MAXS), T(MAXS), RN(4,MAXS), WANGLE(MAXS), GDATUM(MAXS),
     1 ORIENT(MAXS), BETVEL(MAXS), SUMETA(MAXS), SUMCZQ(MAXS),
     2 SCZQSQ(MAXS),SZQETA(MAXS)
      CHARACTER*4 ITYPEO(4,MAXS)
      INTEGER*4 IPT(MAXS)
      REAL ZA(MXPT,MAXS), AA(MXPT,MAXS), BB(MXPT,MAXS), BS(MXPT,MAXS)
```

Program listing—Continued

```
      REAL QA(MXPT,MAXS), TA(MXPT,MAXS), ETA(MXPT,MAXS),
     1 FUNETA(MXPT,MAXS)
      INTEGER*4 ROW(4*MXBH)
      REAL*4 AM((4*MXBH)**2),BMX(4*MXBH)
      CHARACTER*40 BRNAME(MXBH)
      INTEGER*4 IJF(MXBH),IJT(MXBH),NSEC(MXBH),XSKT(MXBH),
     1 PLTBCH(MXBH),PRTXSG(MXBH),PRTBCH(MXBH),PRTSUM(MXBH),PPLTBH(MXBH)
      CHARACTER*4 ITYPE(MXJN)
      INTEGER*4 IBJNC(MXJN),NDATA(MXJN),IZQBVE(MXJN)
      INTEGER*4 ISTATN(MXJN),KTTDBC
      REAL ZQ(MAXZBD,MXTDBC),DTT(MXJN),DATUM(MXJN),ZQBVCO(4,MXJN),
     1 ZQPMIN(MXJN)
      LOGICAL*4 LARBPR(MAXS),ZHIGH(MAXS),ZLOW(MAXS),LINPRT(MAXS),ARBERR
      INTEGER*4 IZDATA(MAXZBD)
      INTEGER*4 IQDATA(MAXQBD),INDATA(MAXQBD)
C     EQUIVALENCE (INDATA(1),IQDATA(1),IZDATA(1))
      REAL CLK(MAXS),ZBOT(MAXS)
      REAL WINDSP( MXWIND),WINDDR( MXWIND)
      INTEGER*4 IDX(MXJN,MXBH), ICT(MXJN)
      REAL W(MXJN)
      REAL*8 U(2*MAXS), UU(4*MAXS), BU(2*MXBH),BUU(4*MXBH)
      REAL*8 C1, C2, C3, C4, UUIJP1, UUIJP2, UUIJP3, UUIJP4
      REAL LAMBDA,MU,SETA,WDTT,TWOCSQ,TWOG,CW,ONECHI,DCHI,DTHETA,TH,WIND
      INTEGER*4 IAR,I,J,K,L,II,IJ,NS,KT,IS,N,NWREAD,NWDATA,INTDBC,IDTPDY
      REAL QTOL,ZTMIN,ZTMAX,ZPMIN,QPMIN,DXMIN,DXMAX
      COMMON /LIMITS/ QTOL,ZTMIN,ZTMAX,ZPMIN,QPMIN,DXMIN,DXMAX
      INTEGER*2 JYR,JMO,JDA,JHR,JMN,MYR,MMO,MDA,MHR,MMN
      INTEGER*4 ND,NDFIRT,NDPART,JETIME,NTSAQ
      COMMON /PARTIM/ ND,NDFIRT,NDPART,JETIME,JYR,JMO,JDA,JHR,JMN,
     1MYR,MMO,MDA,MHR,MMN
      CHARACTER*80 COMENT(9)
      COMMON /CMMNT/ COMENT
      CHARACTER*2 IDETA(7)
      COMMON /ETASYM/ IDETA
      REAL WRATIO,DTZERO,ZTEMP,QTEMP,ZIJ,QIJ,DXIJ,QIJP1,ZIJP1,APZPIJ,
     1      BPZPIJ,BTZPIJ,RPZPIJ,BAVG,BTAVG,AAVG,RAVG,QAVG,ZAVG,BETCOR,
     2      RNIJ,AAVGSQ,AAVGCU,SIGMA,EPSLON,ZETA,OMEGA,GAMMA,DELTA,DET,
     3 DZDT,DQDXC,DQDT,DQDXM,DADX,DZDX,FRIC,ZQPIJ,BIGQ,BIGZ,ZTOL,SOLPDT
     4 ,ALPHA
      INTEGER*4 IBCH,IJZPBC,IJQPBC,KTMATS,LASTN,IJP1,NSM1,JP1,IJ2,IJ4,
     1  IJ4P1,IJ4P2,IJ4P3,IJ4P4,IJ2P1,IJ2P2,I2,I4,I4P1,I4P2,I4P3,I4P4,
     2  I2P1,I2P2,NN,MM,NNN,NBPJ,M0,IBIGZ,JBIGZ,IBIGQ,JBIGQ,IJPNS,ICHK,
     3  IBOUND(NCOL,NROW,NLAY)
      DOUBLE PRECISION HNEW
      DIMENSION HNEW(NCOL,NROW,NLAY),HOLD(NCOL,NROW,NLAY),HCOF(NCOL,NROW
     2,NLAY),QLSUM(MAXS+MXJN),RHS(NCOL,NROW,NLAY)
C     ------------------------------------------------------------------C
        IDBG=0
C                                                                       C
C       WRITE(*,*) 'DOWN TO BRC1FM'
C                                                                       C
C1 ---------CALCULATE RATIO OF TIMESTEPS IN MODFLOW AND BRANCH
       IF(ISS.EQ.0) THEN
      IF(MOD(INT(TFCTR*DELT+0.5),60*IDTM).NE.0) THEN
      WRITE(*,600) IDTM,INT(TFCTR*DELT+0.5)/60
      STOP
      ENDIF
      NTSAQ1=INT(TFCTR*DELT+0.5)/(60*IDTM)
      IF(KKSTP.EQ.1.AND.NELAP.NE.1) NELAP=(NELAP-1)*NTSAQ/NTSAQ1+1
      NTSAQ=NTSAQ1
```

Program listing—Continued

```
      ENDIF
C
C1A-----FOR STEADY-STATE SIMULATIONS ...
      IF(ISS.NE.0) THEN
      NTSAQ=1
      IDTM=1
      ENDIF
C
C2 -----ZERO AVERAGE LEAKAGE RATE VALUES
      DO 80 I=1,NBCH
      NSM1=NSEC(I)-1
      IJ=MAXS-XSKT(I)
      IJFI=MAXS+IJF(I)
      IJTI=MAXS+IJT(I)
      DO 81 J=1,NSM1
      IJ=IJ+1
      IJP1=IJ+1
      QLSUM(IJ)=0.0
      QLSUM(IJP1)=0.0
81    CONTINUE
      QLSUM(IJFI)=0.0
      QLSUM(IJTI)=0.0
80    CONTINUE
      IBRPRN=0
C
C3 ----CALL BRANCH MODEL
C
      CALL BRCH(INDATA,IZDATA,IQDATA,ITQMAX,ITQMIN,
     1 QMAX,QMIN,QSUM,ZQMIN,ZQMAX,AQMAX,AQMIN,A,Z,Q,ZP,QP,
     2 AP,BP,RP,B,R,BT,BTP,XSTATN,DX,T,RN,WANGLE,GDATUM,ORIENT
     3 ,BETVEL,SUMETA,SUMCZQ,SCZQSQ,SZQETA,ITYPEO,ZA,AA,BB,BS,IPT,
     4 QA,TA,ETA,FUNETA,ROW,AM,BMX,BRNAME,IJF,IJT,NSEC,XSKT,
     5 PLTBCH,PRTXSG,PRTBCH,PRTSUM,PPLTBH,ITYPE,IBJNC,NDATA,IZQBVE,
     6 ISTATN,KTTDBC,ZQ,DTT,DATUM,ZQBVCO,ZQPMIN,LARBPR,ZHIGH,
     7 ZLOW,LINPRT,ARBERR,CLK,ZBOT,
     8 MXBH,MXJN,MAXS,MXPT,MXTDBC,MXMD,MAXZBD,
     9 MAXQBD,MAXCZQ,MAXMZQ,MXWIND,MAXBD,MXOTDT,WINDSP,WINDDR,
     & IDX,ICT,W,U,UU,BU,BUU,ZSAV,QSAV,ZPSAV,QPSAV,NELAP,IOUT,
     & NTSAQ,HOLD,HNEW,IBOUND,NCOL,NROW,NLAY,ISTRM,ZPL,IBEGIN,
     & IBRPRN,ZN,ISS,QLSUM,ITRIAL,DCFM)
C
C4 -----CALCULATE MAXIMUM CHANGE OVER MODFLOW ITERATION
      IBCONV=0
      DHMAX=0.0
      IF(IDBG.GT.2) WRITE(*,*) 'UNIT DISCHARGE    DEPTH OF FLOW'
      DO 70 I=1,NBCH
      NS=NSEC(I)
      IJ=MAXS-XSKT(I)
      DO 71 J=1,NS
            DH=ZN(IJ)-ZPL(IJ)
            IF(ABS(DH).GT.DHMAX) DHSIGN=DH
            IF(ABS(DH).GT.DHMAX) DHMAX=ABS(DHSIGN)
            ZPL(IJ)=ZN(IJ)
      IF(IDBG.GT.2) WRITE(*,*) ZN(IJ)
71    CONTINUE
70    CONTINUE
      IF(IDBG.GT.2) WRITE(*,*)'MAXIMUM CHANGE IN STAGE = ',DHMAX
C      IF (DHMAX.GT.0) WRITE(*,*) HCLOSE*100/DHMAX,DHSIGN
C     WRITE(IOUT,1000) DHMAX
      IF(DHMAX.LE.HCLOSE) IBCONV=1
```

Program listing—Continued

```
C                                                                       C
C4A-----DETERMINE LAYER, ROW, COLUMN OF EACH REACH.                     C
      DO 500 I=1,NBCH
      NSM1=NSEC(I)-1
      L=MAXS-XSKT(I)
      DO 501 J=1,NSM1
      L=L+1
      IF(ISTRM(3,L).LT.0) GO TO 501
      IC=ISTRM(1,L)
      IR=ISTRM(2,L)
      IL=ISTRM(3,L)
C5------ADD LEAKAGE TO RHS                                             C
C
C5A ---- IF NTSAQ=1, THE LEAKAGE IS CALCULATED IMPLICITLY
      IF(NTSAQ.EQ.1) THEN
      DXL=DX(L)
      HSTR=ZN(L)+ZDATUM
      PERIM=BP(L)
      CSTR=CLK(L)*PERIM*DXL*TFCTR
      HAQ=HNEW(IC,IR,IL)
      BOT=ZBOT(L)
      IF(HAQ.GT.BOT) THEN
        FLOBOT=0.5*CSTR*(HSTR-HAQ)
        RHS(IC,IR,IL)=RHS(IC,IR,IL)-0.5*CSTR*HSTR
        HCOF(IC,IR,IL)=HCOF(IC,IR,IL)-0.5*CSTR
      ELSE
        IF(HSTR.LT.BOT) HSTR=BOT
        FLOBOT=0.5*CSTR*(HSTR-BOT)
        RHS(IC,IR,IL)=RHS(IC,IR,IL)-FLOBOT
      END IF
      HSTR=ZN(L+1)+ZDATUM
      PERIM=BP(L+1)
      CSTR=CLK(L)*PEIM*DXL*TFCTR
      BOT=ZBOT(L+1)
      IF(HAQ.GT.BOT) THEN
        FLOBOT=FLOBOT+0.5*CSTR*(HSTR-HAQ)
        RHS(IC,IR,IL)=RHS(IC,IR,IL)-0.5*CSTR*HSTR
        HCOF(IC,IR,IL)=HCOF(IC,IR,IL)-0.5*CSTR
      ELSE
        IF(HSTR.LT.BOT) HSTR=BOT
        FLOBOT=FLOBOT+0.5*CSTR*(HSTR-BOT)
        RHS(IC,IR,IL)=RHS(IC,IR,IL)-0.5*CSTR*(HSTR-BOT)
      END IF
      ELSE
C
C5B ----IF NTSAQ>1, LEAKAGE IS TAKEN FROM BRANCH
        RHS(IC,IR,IL)=RHS(IC,IR,IL)-QLSUM(L)*TFCTR
      ENDIF
  501 CONTINUE
  500 CONTINUE
 600   FORMAT(1H0,' WARNING, THE RATIO OF BRANCH TO MODFLOW TIMESTEPS IS
     2 NOT AN INTEGER.'/'BRANCH TIMESTEP IS',1X,1I4,1X,'MINUTES,
     3 MODFLOW TIMESTEP IS',1X,1I7,1X,'MINUTES.'/)
 1000 FORMAT (/' MAXIMUM STAGE CHANGE IN BRANCH PER ITERATION',
     1 1PG12.5/)
C                                                                       C
C6 -----RETURN.                                                        C
      RETURN
      END
```

LIST OF VARIABLES

(Variables specific to original BRANCH code not included)

Variable	Range	Definition
BOT	Module	Local variable for channel bottom elevation.
CLK (MAXS)	Package	Value of K'/b' (leakage coefficient).
CSTR	Module	Local variable for $K'/b' \cdot B \cdot L \cdot$ TFCTR.
DCFM	Package	Multiplier for the friction term in the momentum equation when the channel runs dry.
DELT	Global	Time-step length in MODFLOW.
DH	Module	Difference in stage between the last two MODBRANCH iterations.
DHMAX	Module	Absolute value of DHSIGN.
DHSIGN	Module	Maximum difference in stage between the last two MODBRANCH iterations.
DXL	Module	Local variable for segment length.
FLOBOT	Module	Leakage rate between BRANCH and MODFLOW.
HAQ	Module	Aquifer head at beginning of BRANCH time interval.
HCLOSE	Module	Convergence criteria for head.
HNEW (NCOL, NROW, NLAY)	Global	Aquifer head at end of MODFLOW time step.
HOLD (NCOL, NROW, NLAY)	Global	Aquifer head at beginning of MODFLOW time step.
HSTR	Module	Local variable for stage in the channel.
IBCONV	Package	Flag indicating BRANCH convergence (0, no converge; 1 converge).
IBEGIN	Package	Flag indicating if BRANCH' is being called for the first time (0, first time; 1, not first time).
IBOUND (NCOL, NROW, NLAY)	Global	Status of each cell (<0 is constant head cell, =0 is inactive cell, >0 is variable head cell).
IBRPRN	Package	Flag indicating if BRANCH' is being called from the formulation or budget package (0, formulation; 1, budget).
IOUT	Package	FORTRAN unit number for printed output.
ISS	Global	Flag indicating steady-state simulation (0, not steady state; 1, steady state).
ISTRM (3, MAXS)	Package	Vector describing row, column, and layer of aquifer model cell that corresponds to channel segment.

LIST OF VARIABLES —Continued

(Variables specific to original BRANCH code not included)

Variable	Range	Definition
ITRIAL (MAXS, MAXCZQ)	Package	Flag indicating the wet-dry condition of a channel reach and the relative position of the surrounding aquifer to the riverbed by a two-digit number. For first digit, 0 is reach wet, 1 is upstream node dry, 2 is downstream node dry, and 3 to 8 are both nodes dry with successively high frictional resistance. For second digit, 0 is aquifer above riverbed at both nodes, 1 is aquifer below riverbed at upstream node, 2 is aquifer below riverbed at downstream node, and 3 is aquifer below riverbed at both nodes.
KDAS	Package	Saved value of day.
KHRS	Package	Saved value of hour.
KKITER	Global	MODFLOW iteration number.
KKSTP	Global	MODFLOW time-step number.
KMNS	Package	Saved value of minute.
KMOS	Package	Saved value of month.
KYRS	Package	Saved value of year.
MAXCZQ	Package	Maximum number of daily computed results held in storage for plotting purposes.
MAXMZQ	Package	Maximum number of measured data held in storage for plotting purposes.
MAXQBD	Package	Maximum number of discharge boundary value data held in storage for computation.
MAXS	Package	Maximum number of cross sections in the entire channel network.
MAXZBD	Package	Maximum number of stage boundary value data held in storage for computation.
MXBH	Package	Maximum number of branches in the network.
MXJN	Package	Maximum number of junctions in the network.
MXMD	Package	Maximum number of measured data locations accommodated in the network.
MXPT	Package	Maximum number of points used to define a cross section.
MXTDBC	Package	Maximum number of boundaries in the network.
MXWIND	Package	Maximum number of wind data points input.

LIST OF VARIABLES —Continued

(Variables specific to original BRANCH code not included)

Variable	Range	Definition
NCOL	Global	Maximum number of MODFLOW aquifer columns.
NELAP	Package	Number of elapsed MODFLOW time steps since beginning of simulation.
NLAY	Global	Maximum number of MODFLOW aquifer layers.
NROW	Global	Maximum number of MODFLOW aquifer rows.
NSEC	Package	Maximum cross sections per branch.
NTSAQ	Package	Number of BRANCH' time intervals in one MODFLOW time step.
NTSAQ1	Package	Adjustable NTSAQ for varying MODFLOW time-step length.
PERIM	Module	Local variable for channel wetted perimeter.
QLSUM (MAXS)	Package	Average leakage rate out of a river segment over one MODFLOW time step.
QPSAV (MAXS)	Package	Value of discharge at end of first BRANCH' time interval after beginning of a MODFLOW time step.
QSAV (MAXS)	Package	Value of discharge at beginning of first BRANCH' time interval in a MODFLOW time step.
SLKG (MAXS)	Package	Vector of leakage rates.
TFCTR	Module	Number of seconds in one time unit used in MODFLOW.
ZBOT (MAXS)	Module	Elevation of channel bottom.
ZN (MAXS)	Module	Value of stage at end of final BRANCH' time interval in a MODFLOW time step.
ZPL (MAXS)	Module	Value of stage at end of final BRANCH' time interval in a MODFLOW time step for previous trial.
ZPSAV (MAXS)	Module	Value of stage at end of first BRANCH' time interval after beginning of a MODFLOW time step.
ZSAV (MAXS)	Package	Value of stage at beginning of first BRANCH' time interval after beginning of a MODFLOW time step.

MODULE BRC1BD

This module calculates the volumetric budget for leakage for a single MODFLOW time step. It is executed each time the two models have converged and calls BRANCH' with a flag set to allow printout in BRANCH'. The numbered sections in the narrative correspond to the numbered steps in the flowchart for BRC1BD shown in figure 9.

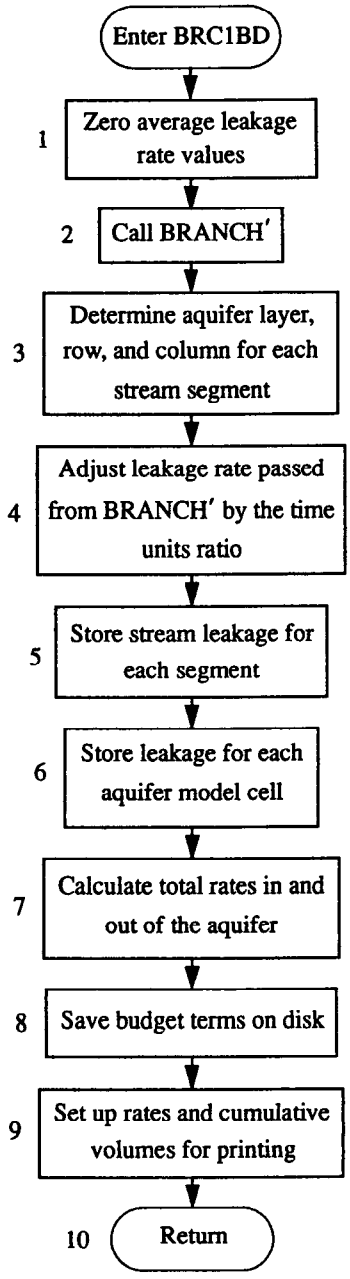

Figure 9. Flowchart of budget module (BRC1BD).

NARRATIVE

1. Zero average leakage rate values. Before BRANCH' is executed, the average leakage rates for each segment (QLSUM [segment number]) are set to zero.
2. Call BRANCH'. BRANCH' is executed with a flag set (IBRPRN=1) so that normal printouts of results are