# USGS

*science for a changing world*

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

Chapter A1

# A MODULAR THREE-DIMENSIONAL FINITE-DIFFERENCE GROUND-WATER FLOW MODEL

By Michael G. McDonald and
Arlen W. Harbaugh

This chapter supersedes U.S. Geological
Survey Open-File Report 83-875

Book 6

MODELING TECHNIQUES

$$d_n = E_n + \omega(A'_n + T'_n + C'_n + G'_n + U'_n + R'_n)$$

$$- a_n g_{nll} - b_n f_{nrl} - c_n e_{ncl} \tag{108-j}$$

$$e_n = (F_n - \omega(A'_n + C'_n))/d_n \tag{108-k}$$

$$f_n = (H_n - \omega(T'_n + G'_n))/d_n \tag{108-l}$$

$$g_n = (S_n - \omega(R'_n + U'_n))/d_n \tag{108-m}$$

$$v_n = (RES_n - a_n v_{nll} - b_n v_{nrl} - c_n v_{ncl})/d_n \tag{108-n}$$

Since the backward substitution requires all values of $e_n$, $f_n$, $g_n$, and $v_n$, space is allocated in the SIP Package for four arrays to store those values. Each of these arrays has as many elements as there are cells in the grid.

## Iteration Parameters

While Weinstein, Stone and Kwan (1969) define three iteration parameters in their theoretical development, they utilize a single value in practice. Thus the terms $\alpha$, $\beta$ and $\gamma$ of equation (93) are replaced by a single parameter, $\omega$, which multiplies each term on the right side of the equation; however, $\omega$ must be cycled through a series of values in successive iterations to achieve satisfactory rates of convergence. In the model described herein, values of $\omega$ are calculated from the expression

$$\omega(\lambda) = 1-(WSEED)^{(\lambda-1)/(NPARM-1)} \qquad \lambda=1,2....NPARM \tag{109}$$

where NPARM is the total number of $\omega$ values to be used; $\lambda$ is an index taking on integral values from 1 to NPARM; $w(\lambda)$ is the corresponding iteration parameter value; and WSEED is the iteration parameter "seed", calculated according to rules outlined below, and used as a basis for determining the sequence of $\omega$ values.

The value of WSEED is in turn developed as follows. The terms $\rho_1$, $\rho_2$, and $\rho_3$ are calculated for each cell in the mesh using the conductances between that cell and its neighbors, as follows

$$\rho_1 = \frac{CC_{max} + CV_{max}}{CR_{min}} \quad (110)$$

$$\rho_2 = \frac{CR_{max} + CV_{max}}{CC_{min}} \quad (111)$$

$$\rho_3 = \frac{CR_{max} + CC_{max}}{CV_{min}} \quad (112)$$

where $CC_{max}$ for a given cell, i,j,k, is the larger of $CC_{i-1/2,j,k}$ and $CC_{i+1/2,j,k}$, while $CC_{min}$ is the smaller of these values; and similarly $CR_{max}$ is the larger of $CR_{i,j-1/2,k}$ and $CR_{i,j+1/2,k}$, while $CR_{min}$ is the smaller, and $CV_{max}$ is the larger of $CV_{i,j,k-1/2}$ and $CV_{i,j,k+1/2}$, while $CV_{min}$ is the smaller. Using these values, the terms

$$\frac{\pi^2}{2(NCOL)^2(1+\rho_1)} \quad , \quad \frac{\pi^2}{2(NROW)^2(1+\rho_2)} \quad \text{and} \quad \frac{\pi^2}{2(NLAY)^2(1+\rho_3)}$$

are computed for each cell in the grid, where again NCOL is the number of columns in the model, NROW is the number of rows and NLAY is the number of layers; and the minimun value for each of these three terms is taken to be the cell seed. The seed term, WSEED, is then taken as average of all the cell seeds. The iteration parameters, $\omega(\lambda)$, generated from the WSEED value are then used sequentially in successive iterations, recycling each time the entire set has been used (i.e., each NPARM iterations).

The process described above for calculating the sequence of $\omega$ values differs slightly from that used by Weinstein, Stone and Kwan (1969), but produces values that are in the same range and that appear to function well in many problems. However, several points should be made regarding iteration parameter selection. First, the process is essentially empirical,

and there is little understanding of why one sequence of parameters performs better than another. Second, the parameters chosen affect the rate of convergence but (assuming that convergence is achieved) should not influence the final solution. Third, the influence of the parameters on the rate of convergence is extremely significant.

The model described herein provides for an additional iteration parameter, referred to here as the acceleration parameter to distinguish it from $\omega$. This parameter, which is designated ACCL in the program, functions as a multiplier of $\{RES^\ell\}$; thus where a parameter of this type is not to be used, ACCL is simply assigned a value of one. Parameters similar to ACCL have been used in various versions of SIP (Peaceman, 1977, page 130) although Weinstein, Stone and Kwan (1969) do not employ a parameter of this type. ACCL is not cycled, but rather is assigned a single value by the user. As a general rule, it should initially be given a value of one, and improvement in the rate of convergence should be pursued through adjustment of the seed term, as explained below. If problems with convergence persist, values of ACCL other than one can be tried.

Experience has shown that setting the acceleration/relaxation parameter (ACCL) to 1 and using the seed value calculated by the program does not always produce optimum convergence--that is, the number of iterations required to achieve convergence is not minimum. Convergence rates will deviate from the optimum if the absolute value of head change in each iteration is consistently either too small or too large. When the head change is too large, the computed head overshoots the correct value, and oscillations occur as the head change repeatedly reverses to compensate

for the overshoot. Severe overshoot causes divergence, while moderate overshoot simply slows down convergence. When head change is too small, the opposite problem occurs; head tends to approach the correct value monotonically, but very slowly. In severe situations, the head changes at each iteration may be so small that the criterion for convergence is satisfied at all points, even though the computed heads are still far from the correct values. In such situations, a significant volumetric budget imbalance will occur.

Weinstein, Stone, and Kwan (1969) suggest that a trial and error method can be used to improve the choice of seed. This can be done by making an initial run using the seed calculated by the program or chosen from experience, and using ACCL=1. The trend of head change per iteration, with increasing iteration number, is observed for the iterations of a single time step. There is normally some variation in head change from one iteration to the next due to the cycling of iteration parameters, but this variation is often superimposed on an overall trend in which head change tends either to increase or decrease as iterations continue; it is this overall trend (which is often most evident in the later iterations of the test) that is of interest here. Some oscillations (reversals in sign) of the computed head change are normal during convergence; however, repeated oscillation is a sign of overshoot, indicating that computed head changes are too great for optimal convergence. Head changes which are too small, on the other hand, are indicated by a very flat overall trend. For proper evaluation of the trend, the trial should generally be run for a number of iterations equal to 4 or 5 times the number of iteration parameters, unless convergence occurs before this.

Following the initial trial, the seed is multiplied by a number between two and ten if head changes in the initial trial appear to be too great, and divided by a number between two and ten if those head changes appear to be too small. If the trend in the initial trial is unclear either multiplication or division of the seed may be tried. In any case, a second run is made using the new seed value, and the trend of head change vs. iteration level is again examined. The results are compared with those of the initial trial to see if the rate of convergence has improved. If both runs have converged, the comparison is based on the number of iterations required for convergence; if they have failed to converge, the comparison is based on the head changes observed in the final iterations.

The trial runs can be continued to further refine the choice of seed; in general the seed value will be multiplied or divided by progressively smaller numbers at each step of the procedure. However, it is usually not worthwhile to carry the process too far; multiplication or division of the seed by factors less than 2 is seldom warranted.

In most cases, a satisfactory seed value developed by this procedure will remain satisfactory even though changes in the model are introduced-- for example, additional stresses, modifications in boundary conditions or changes in the model mesh. However, if convergence problems arise after such changes, the trial and error procedure can be repeated. It should be noted that the more strongly diagonal the coefficient matrix, the less important the choice of seed will be. Thus, source terms such as evapotranspiration or stream seepage, which affect only coefficients on the main diagonal, normally tend to make the choice of seed less critical; and the addition of such terms to a model seldom necessitates modification of the seed.

For each iteration, the program stores the value of $|\Delta h_\ell|_{max}$, where $|\Delta h_\ell|$ refers to the absolute value of the head change computed during iteration $\ell$ at a given node, and $|\Delta h_\ell|_{max}$ is the maximum such absolute value for the entire mesh in that iteration. For the last time step of each stress period, a table of $|\Delta h_\ell|_{max}$ values for each iteration of the time step is printed in the program output; at the user's option, a table of these values may be printed for every time step, or for time steps which fall at specified intervals. (See Narrative for Module SIP1AP and Sample Problem Output in Appendix D.) In addition to the $|\Delta h_\ell|_{max}$ value, each entry in the table shows the indices of the node at which the maximum change was recorded, and a sign indicating whether the change was positive or negative. This table can be used in the head change trend evaluations described above, under the assumption that the behavior of the $|\Delta h_\ell|_{max}$ values is representative of the behavior of head change throughout the mesh.

Improvements in the rate of convergence can also be obtained by adjusting the acceleration parameter, ACCL. Increases in ACCL will cause increases in the head change at each iteration, while decreases in ACCL will cause decreases in head change. The trial procedure described above can be used for this case as well; however changes in the seed and in ACCL should never be attempted in the same set of trial runs.

It is sometimes necessary to slow the process of convergence in order to prevent cells from converting to the no-flow condition as a result of head overshoot during an iteration. In these situations, optimal convergence cannot be considered convergence in the minimum number of iterations, but rather convergence in the smallest number of iterations that does not involve head overshoot. The procedure of examining head

change per iteration and adjusting iteration parameters can again be used

to determine when this condition is being met, and to develop the required

seed or ACCL terms.

# Strongly Implicit Procedure Package Input

Input to the Strongly Implicit Procedure (SIP) Package is read from the unit specified in IUNIT(9).

FOR EACH SIMULATION

### SIP1AL

1. Data:  MXITER    NPARM
   Format: I10       I10

### SIP1RP

2. Data:  ACCL      HCLOSE    IPCALC    WSEED     IPRSIP
   Format: F10.0     F10.0     I10       F10.0     I10

### Explanation of Fields Used in Input Instructions

MXITER--is the maximum number of times through the iteration loop in one time step in an attempt to solve the system of finite-difference equations. Fifty iterations are generally sufficient.

NPARM--is the number of iteration parameters to be used. Five parameters are generally sufficient.

ACCL--is the acceleration parameter. It must be greater than zero and is generally equal to one. If a zero is entered, it is changed to one.

HCLOSE--is the head change criterion for convergence. When the maximum absolute value of head change from all nodes during an iteration is less than or equal to HCLOSE, iteration stops.

IPCALC--is a flag indicating where the iteration parameter seed will come from.

   0 - the seed will be entered by the user.

   1 - the seed will be calculated at the start of the simulation from problem parameters.

WSEED--is the seed for calculating iteration parameters. It is only specified if IPCALC is equal to zero.

IPRSIP--is the printout interval for SIP. If IPRSIP is equal to zero, it is changed to 999. The maximum head change (positive or negative) is printed for each iteration of a time step whenever the time step is an even multiple of IPRSIP. This printout also occurs at the end of each stress period regardless of the value of IPRSIP.

SAMPLE INPUT TO THE SIP PACKAGE(USER SPECIFIES THE SEED)

| DATA ITEM | EXPLANATION | INPUT RECORDS | | | | |
|---|---|---|---|---|---|---|
| 1 | {MXITER, NPARM} | 50 | 5 | | | |
| 2 | {ACCL, HCLOSE, IPCALC, WSEED, IPRSIP} | 1. | .01 | 0 | .98 | 10 |

SAMPLE INPUT TO THE SIP PACKAGE(PROGRAM CALCULATES THE SEED)

| DATA ITEM | EXPLANATION | INPUT RECORDS | | |
|---|---|---|---|---|
| 1 | {MXITER, NPARM} | 100 | 6 | |
| 2 | {ACCL, HCLOSE, IPCALC, WSEED, IPRSIP} | 1. | .01 | 1 |

## Module Documentation for the Strongly Implicit Procedure Package

The Stongly Implicit Procedure Package (SIP1) consists of three primary modules and two submodules. They are:

### Primary Modules

SIP1AL      Allocates space for SIP work arrays.

SIP1RP      Reads control information needed by the SIP
Package and calculates iteration parameters if
the seed is specified by the user.

SIP1AP      Performs one iteration of the strongly implicit
procedure.

### Submodules

SSIP1P      Prints the largest head change for each iteration.

SSIP1I      Calculates iteration parameters when the seed
is calculated by the program.

## Narrative for Module SIP1AL

Module SIP1AL allocates space in the X array for SIP arrays. The four arrays, EL, FL, GL, and V hold intermediate results during the solution process. Each of these contains one element for each model cell. Additionally, three arrays, HDCG, LRCH, and W are required. HDCG holds the maximum head change each iteration and LRCH holds the cell location at which the maximum occurred. The number of elements in HDCG is MXITER, and that in LRCH is three times MXITER, where MXITER is the maximum number of iterations allowed in a time step and is specified by the user. Array W holds iteration parameters. One element in W is used for each of the NPARM iteration parameters. (NPARM is specified by the user.)

Module SIP1AL performs its functions in the following order:

1. Print a message identifying the SIP Package.

2. Read and print MXITER and NPARM.

3. Allocate the required space in the X array. The X-array location pointer (ISUM) is saved in variable ISOLD prior to allocation so that the space required for SIP can be calculated in step 4.

4. Calculate and print the space used in the X array. The space used by SIP is ISUM - ISOLD. The total space allocated by all packages so far is ISUM - 1.

5. RETURN.

## Flow Chart for Module SIP1AL

X array is the pool of memory space
from which space is allocated for
arrays used by various packages.

```
        ╭─────────────╮
        │    ENTER    │
        │    SIP1AL   │
        ╰─────────────╯
               │
               ▼          1
        ┌─────────────┐
        │PRINT A MESSAGE│
        │ IDENTIFYING │
        │ SIP PACKAGE │
        └─────────────┘
               │
               ▼              2
     ┌──────────────────────┐
     │    READ AND PRINT    │
     │ MAXIMUM # OF ITERATIONS│
     │  AND # OF ITERATION  │
     │      PARAMETERS      │
     └──────────────────────┘
               │
               ▼          3
        ┌─────────────┐
        │ALLOCATE SPACE│
        │ FOR SIP ARRAYS│
        │ IN THE X ARRAY│
        └─────────────┘
               │
               ▼          4
        ┌─────────────┐
        │CALCULATE AND │
        │PRINT SPACE USED│
        │ IN THE X ARRAY│
        └─────────────┘
               │
               ▼          5
        ╭─────────────╮
        │   RETURN    │
        ╰─────────────╯
```
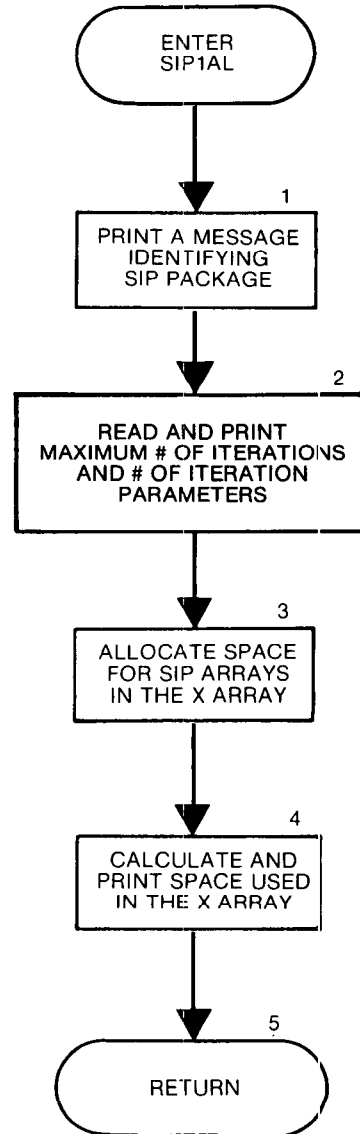
```
      SUBROUTINE SIP1AL(ISUM,LENX,LCEL,LCFL,LCGL,LCV,LCHDCG,LCLRCH,
     1          LCW,MXITER,NPARM,NCOL,NROW,NLAY,IN,IOUT)
C
C-----VERSION 1110 31DEC1986 SIP1AL
C
C     ******************************************************************
C     ALLOCATE STORAGE IN THE X ARRAY FOR SIP ARRAYS
C     ******************************************************************
C
C        SPECIFICATIONS:
C     ------------------------------------------------------------------
C     ------------------------------------------------------------------
C
C1------PRINT A MESSAGE IDENTIFYING SIP PACKAGE
      WRITE(IOUT,1)IN
    1 FORMAT(1H0,'SIP1 -- STRONGLY IMPLICIT PROCEDURE SOLUTION PACKAGE'
     1,', VERSION 1, 9/1/87',' INPUT READ FROM UNIT',I3)
C
C2------READ AND PRINT MXITER AND NPARM
      READ(IN,2) MXITER,NPARM
    2 FORMAT(2I10)
      WRITE(IOUT,3) MXITER,NPARM
    3 FORMAT(1X,'MAXIMUM OF',I4,' ITERATIONS ALLOWED FOR CLOSURE'/
     1          1X,I2,' ITERATION PARAMETERS')
C
C3------ALLOCATE SPACE FOR THE SIP ARRAYS
      ISOLD=ISUM
      NRC=NROW*NCOL
      ISIZ=NRC*NLAY
      LCEL=ISUM
      ISUM=ISUM+ISIZ
      LCFL=ISUM
      ISUM=ISUM+ISIZ
      LCGL=ISUM
      ISUM=ISUM+ISIZ
      LCV=ISUM
      ISUM=ISUM+ISIZ
      LCHDCG=ISUM
      ISUM=ISUM+MXITER
      LCLRCH=ISUM
      ISUM=ISUM+3*MXITER
      LCW=ISUM
      ISUM=ISUM+NPARM
C
C4------CALCULATE AND PRINT THE SPACE USED IN THE X ARRAY
      ISP=ISUM-ISOLD
      WRITE(IOUT,4) ISP
    4 FORMAT(1X,I8,' ELEMENTS IN X ARRAY ARE USED BY SIP')
      ISUM1=ISUM-1
      WRITE(IOUT,5) ISUM1,LENX
    5 FORMAT(1X,I8,' ELEMENTS OF X ARRAY USED OUT OF',I8)
      IF(ISUM1.GT.LENX) WRITE(IOUT,6)
    6 FORMAT(1X,'    ***X ARRAY MUST BE DIMENSIONED LARGER***')
C
C5------RETURN
      RETURN
      END
```

## List of Variables for Module SIP1AL

| Variable | Range | Definition |
|---|---|---|
| IN | Package | Primary unit number from which input for this package will be read. |
| IOUT | Global | Primary unit number for all printed output. IOUT = 6. |
| ISIZ | Module | Number of cells in the grid. |
| ISOLD | Package | Before this module allocates space, ISOLD is set equal to ISUM. After allocation, ISOLD is subtracted from ISUM to get ISP, the amount of space in the X array allocated by this module. |
| ISP | Module | Number of words in the X array allocated by this module. |
| ISUM | Global | Index number of the lowest element in the X array which has not yet been allocated. When space is allocated for an array, the size of the array is added to ISUM. |
| ISUM1 | Module | ISUM-1. |
| LCEL | Package | Location in the X array of the first element of array EL. |
| LCFL | Package | Location in the X array of the first element of array FL. |
| LCGL | Package | Location in the X array of the first element of array GL. |
| LCHDCG | Package | Location in the X array of the first element of array HDCG. |
| LCLRCH | Package | Location in the X array of the first element of array LRCH. |
| LCV | Package | Location in the X array of the first element of array V. |
| LCW | Package | Location in the X array of the first element of array W. |
| LENX | Global | Length of the X array in words. This should always be equal to the dimension of X specified in the MAIN program. |
| MXITER | Package | Maximum number of iterations. |
| NCOL | Global | Number of columns in the grid. |
| NLAY | Global | Number of layers in the grid. |
| NPARM | Package | Number of iteration parameters. |
| NRC | Module | Number of cells in a layer. |
| NROW | Global | Number of rows in the grid. |

## Narrative for Module SIP1RP

Module SIP1RP reads data for the SIP package: the acceleration parameter (ACCL), the closure criterion (HCLOSE), the iteration-parameter seed (WSEED), a flag indicating whether WSEED is to be calculated or specified by the user (IPCALC), and the interval for printing head change (IPRSIP). If IPCALC is zero, iteration parameters are calculated using WSEED as the seed. Module SIP1RP performs its functions in the following order:

1. Read the data. If ACCL is zero, substitute a default of 1.0. If IPRS1P is less than or equal to zero, substitute an interval of 999 time steps. The defaults are provided as a convenience to the user.

2. Print the data read in step 1.

3. Check IPCALC which is a flag that indicates the source of the iteration-parameter seed (WSEED).

   (a)  If IPCALC is not zero, submodule SSIP1I will calculate a seed and the resulting iteration parameters at the start of the first iteration. Print a message telling of this option.

   (b)  If IPCALC is zero, use WSEED to calculate iteration parameters. The i-th iteration parameter ($I_i$) is given by the expression

$$I_i = 1 - (WSEED)^{\frac{i-1}{NPARM-1}}.$$
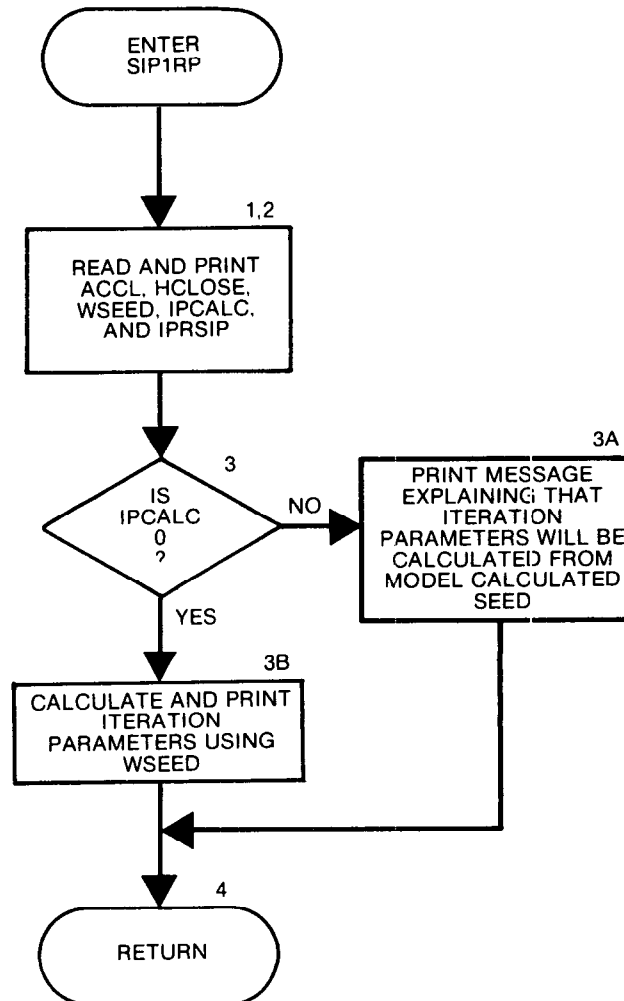
Print the parameters.

4. RETURN.

ACCL is a multiplier of
calculated head change
which is used to control
the convergence rate.

HCLOSE is the head change
closure criterion. When
head change in all model
cells is less than or
equal to HCLOSE, iteration
stops.

WSEED is the seed, specified
by the user, on which the
calculation of iteration
parameters is based if
IPCALC is zero.

IPRSIP is the time step
interval for printing the
maximum head change for
each iteration of a time
step. Head change is
printed every IPRSIP time
step. Head change is
printed at the end of a
stress period regardless
of the interval.

IPCALC is a flag. If it is
set equal to one, iteration
parameters will be calculated
from a seed calculated
within the program. If it
is clear (equal to zero),
iteration parameters will
be calculated from a seed
provided by the user.

```
        ┌──────────────┐
        │    ENTER     │
        │   SIP1RP     │
        └──────┬───────┘
               │
               ▼                    1,2
        ┌──────────────┐
        │ READ AND PRINT│
        │ ACCL, HCLOSE, │
        │ WSEED, IPCALC,│
        │  AND IPRSIP   │
        └──────┬───────┘
               │
               ▼            3
            ╱╲                              3A
           ╱  ╲         NO      ┌─────────────────┐
          ╱ IS ╲────────────────│  PRINT MESSAGE  │
         ╱IPCALC ╲               │ EXPLAINING THAT │
         ╲  0   ╱                │    ITERATION    │
          ╲  ? ╱                 │ PARAMETERS WILL BE│
           ╲  ╱                  │ CALCULATED FROM │
            ╲╱                   │ MODEL CALCULATED│
            │ YES                │      SEED       │
            │                    └────────┬────────┘
            ▼         3B                  │
     ┌──────────────┐                     │
     │CALCULATE AND PRINT│                │
     │   ITERATION  │                     │
     │ PARAMETERS USING│                  │
     │     WSEED    │                     │
     └──────┬───────┘                     │
            │◄───────────────────────────┘
            │
            ▼      4
     ┌──────────────┐
     │              │
     │    RETURN    │
     │              │
     └──────────────┘
```

```
      SUBROUTINE SIP1RP(NPARM,MXITER,ACCL,HCLOSE,W,IN,IPCALC,IPRSIP,
     1                  IOUT)
C
C-----VERSION 0925 16DEC1982 SIP1RP
C
C     ******************************************************************
C     READ DATA FOR SIP
C     ******************************************************************
C
C        SPECIFICATIONS:
C     ------------------------------------------------------------------
      DIMENSION W(NPARM)
C     ------------------------------------------------------------------
C
C1------READ ACCL,HCLOSE,WSEED,IPCALC,IPRSIP
      READ(IN,1) ACCL,HCLOSE,IPCALC,WSEED,IPRSIP
    1 FORMAT(2F10.0,I10,F10.0,I10)
      IF(ACCL.EQ.0.) ACCL=1.
C
C2------PRINT DATA VALUES JUST READ
      WRITE(IOUT,100)
  100 FORMAT(1H0,///57X,'SOLUTION BY THE STRONGLY IMPLICIT PROCEDURE'
     1/57X,43('-'))
      WRITE(IOUT,115) MXITER
  115 FORMAT(1H0,47X,'MAXIMUM ITERATIONS ALLOWED FOR CLOSURE =',I9)
      WRITE(IOUT,120) ACCL
  120 FORMAT(1H ,63X,'ACCELERATION PARAMETER =',G15.5)
      WRITE(IOUT,125) HCLOSE
  125 FORMAT(1H ,52X,'HEAD CHANGE CRITERION FOR CLOSURE =',E15.5)
      IF(IPRSIP.LE.0)IPRSIP=999
      WRITE(IOUT,130) IPRSIP
  130 FORMAT(1H ,52X,'SIP HEAD CHANGE PRINTOUT INTERVAL =',I9)
C
C3------CHECK IF SPECIFIED VALUE OF WSEED SHOULD BE USED OR IF
C3------SEED SHOULD BE CALCULATED
      IF(IPCALC.EQ.0) GO TO 150
C
C3A-----CALCULATE SEED & ITERATION PARAMETERS PRIOR TO 1ST ITERATION
      WRITE(IOUT,140)
  140 FORMAT(1H0,52X,'CALCULATE ITERATION PARAMETERS FROM MODEL',
     1' CALCULATED WSEED')
      GO TO 1000
C
C3B-----USE SPECIFIED VALUE OF WSEED
C3B-----CALCULATE AND PRINT ITERATION PARAMETERS
  150 P1=-1.
      P2=NPARM-1
      DO 160 I=1,NPARM
      P1=P1+1.
  160 W(I)=1.-WSEED**(P1/P2)
      WRITE(IOUT,161) NPARM,WSEED,(W(J),J=1,NPARM)
  161 FORMAT(1H0,/,I5,' ITERATION PARAMETERS CALCULATED FROM',
     1       ' SPECIFIED WSEED =',F11.8,' :'//(10X,6E15.7))
C
C4------RETURN
 1000 RETURN
      END
```

## List of Variables for Module SIP1RP

| Variable | Range | Definition |
|---|---|---|
| ACCL | Package | Acceleration parameter. |
| HCLOSE | Package | Closure criterion for the iterative procedure. |
| I | Module | Index for iteration parameters. |
| IN | Package | Primary unit number from which input for this package will be read. |
| IOUT | Global | Primary unit number for all printed output. IOUT = 6. |
| IPCALC | Package | Flag.<br><br>$\neq$ 0, seed for iteration will be calculated in the program.<br>= 0, seed will be specified by the user. |
| IPRSIP | Package | Frequency (in time steps) with which the maximum head changes for each iteration will be printed. |
| J | Module | Index for iteration parameters. |
| MXITER | Package | Maximum number of iterations. |
| NPARM | Package | Number of iteration parameters. |
| P1 | Module | I - 1. |
| P2 | Module | NPARM - 1. |
| W | Package | DIMENSION (NPARM), Iteration parameters. |
| WSEED | Module | Seed for calculating iteration parameters. |