the U.S. Geological Survey Water Availability and Use Science Program

ZONEBUDGET for MODFLOW 6

Version mf6.4.1—December 09, 2022

Contents

Blank page

Abstract

The computer program ZONEBUDGET 6, which is written in FORTRAN, calculates subregional water budgets using results from MODFLOW 6. ZONEBUDGET uses cell-by-cell flow data saved by the model in order to calculate the budgets. Subregions of the modeled region are designated by zone numbers. The user assigns a zone number for each cell in the model.

Introduction

ZONEBUDGET is a FORTRAN program that calculates subregional water budgets using results from MODFLOW 6. Hydrologists commonly calculate water budgets in order to help gain an understanding of a groundwater flow system. Computer programs that simulate groundwater flow typically produce water budgets for the modeled region, but often it is useful to have a water budget for a particular subregion of the modeled region. ZONEBUDGET calculates budgets by tabulating the budget data that MODFLOW produces using the cell-by-cell flow option. The user simply specifies the subregions for which budgets will be calculated. The purpose of this report is to document the design of ZONEBUDGET and how to use it.

Functional Design of the Program

A subregion of a model for which ZONEBUDGET calculates a water budget is termed a zone, and a zone is indicated by a zone number. Zone numbers can be any negative or positive integer. A zone number must be assigned to each cell in the model grid. Although 0 is a valid zone number, a budget is not calculated for Zone 0. Thus, the user can exclude cells from budget calculations by assigning them to Zone 0. Zones are defined by reading a file, called the zone file, which contains one zone value for each cell.

ZONEBUDGET makes use of the cell-by-cell flow data that MODFLOW saves on disk. This cell-by-cell data defines the flow into each model cell from each possible source or sink. Outflows are indicated as negative inflows. Possible sources and sinks include stress package inflows and outflows, storage, and flow between adjacent cells.

Stresses and storage represent flow into or out of the simulated flow system, and are straightforward components of the calculated water budget. Flow between adjacent cells is an internal flow that is not directly a part of the water budget for a subregion; however, flow between adjacent cells along a subregion's boundary is used to calculate flows in and out of the subregion. Groundwater flow between subregions must be included in subregional budgets in order to account for all flow. Flow across a subregion's boundary is separately accumulated for each adjacent subregion. That is, if there are 25 zones, Zone 1 may be adjacent to Zones 2-24; and if so, ZONEBUDGET will produce a separate accounting of boundary flow from Zone 1 to each of the other 24 zones.

Each source of water that is listed in the budget is referred to as a budget term. Thus, the budget for each zone will generally contain several stress budget terms, a storage term, a constant-head term, and a number of terms for flow to other zones.

MODFLOW calculates both a volumetric flow rate budget and a cumulative volume budget over time. ZONEBUDGET also calculates a volumetric and cumulative volume budget. If flow terms from the MODFLOW simulation are not saved for every time step and stress period, then the cumulative budget written by ZONEBUDGET will not be comparable to the one written by MODFLOW.

If ZONEBUDGET is used to calculate a budget for the entire modeled area, values for each budget term should match the MODFLOW budget provided that all flow terms for the MODFLOW simulation are written to the budget file.

When calculating the constant-head budget term, ZONEBUDGET does not directly use cell-by-cell values for constant-head flow as saved by MODFLOW. This is because the values for constant-head flow that are saved by MODFLOW are combinations of flow into the model through the sides of each constant-head cell. Some of the flows may be inflows and some outflows, but only the single net value is saved for each constant-head cell. In order to separately accumulate the inflows and outflows to each constant-head cell, ZONEBUDGET uses the flow between adjacent cells budget term, called "FLOW JA FACE" to calculate the constant-head flow. The constant-head budget term saved by MODFLOW is still needed by ZONEBUDGET in order to identify constant-head cells. Consistent with the MODFLOW 6 approach, ZONEBUDGET does not include in the constant-head budget the flow between two adjacent constant-head cells.

ZONEBUDGET implements Zone 0, for which no budget is calculated, as a way to prevent a budget from being calculated in areas for which no budget is desired. Although one might think that it is necessary to assign Zone 0 to no-flow cells, this is not necessary because MODFLOW specifies 0 stresses and flows for any no-flow cells, whether initially no-flow or set to no-flow as a result of dewatering. That is, if no-flow cells are included in a non-zero zone, the no-flow cells will have no impact on the budget.

Instructions for Using the Program

Cell-by-cell flow terms from a model run must be saved prior to executing ZONEBUDGET. Each stress package and the NPF and STO Packages determine whether their respective cell-by-cell terms are saved. The NPF Package generates budget terms for constant-head and flow between adjacent cells. The STO Package generates budget terms for storage changes. The stress packages, NPF Package, and the STO Package allow terms to be saved or not depending on whether the "SAVE_FLOWS" option is specified in the OPTIONS block for the package. All cell-by-cell terms should be saved, and all terms should be saved in the same file. If some terms are not saved, the budget will be incomplete, and a large percent discrepancy will appear in the program output.

The Output Control (OC) Package of MODFLOW controls the frequency of cell-by-cell output. Flow terms will be saved whenever "SAVE BUDGET" is specified for a PERIOD block. The following example OC file shows settings that will cause the budgets to be saved to "mymodel.bud" for every time step and stress period in the simulation:

```
# This is an output control file
BEGIN OPTIONS
  BUDGET FILEOUT mymodel.bud
END OPTIONS

BEGIN PERIOD 1
  SAVE BUDGET ALL
END PERIOD
```

ZONEBUDGET will print budgets for all zones at all the times for which MODFLOW has saved flow terms. Remember that the input to individual stress packages and the NPF and STO Packages as well as the input to the Output Control option must be properly specified when using MODFLOW in order for cell-by-cell data to be properly saved for use by ZONEBUDGET.

After cell-by-cell zones have been saved, ZONEBUDGET can be executed; however, the user must also prepare a zone file in order to specify a zone value for each model cell. The instructions for preparing a zone file are contained in a following section (see "Input Instructions for the Zone File").

When ZONEBUDGET is executed, the program will immediately look for a ZONEBUDGET name file with the name "zbud.nam". If no such file exists, then the program will terminate with an error. Users can

also specify the name of the ZONEBUDGET name file as an argument to the program, such as "zbud6.exe myzbud.nam". The format for the ZONEBUDGET name file is as follows:

```
BEGIN ZONEBUDGET
  BUD <budgetfile>
  ZON <zonefile>
  [GRB <binarygridfile>]
END ZONEBUDGET
```

where:

- `BUD <budgetfile>`—is the keyword and name of the MODFLOW 6 budget file. This file contains double precision numeric values of simulated flow.

- `ZON <zonefile>`—is the keyword and name of the zone input file.

- `GRB <binarygridfile>`—is the keyword and name of the binary grid file created by MODFLOW 6. Note that if the NOGRB keyword is specified in the OPTIONS block of the Discretization Package input file, then MODFLOW 6 will not create the binary grid file needed to run ZONEBUDGET. The binary grid file must be provided if the budget file is for a Groundwater Flow (GWF) Model. The binary grid file is not needed when ZONEBUDGET is used to process the budget file from one of the advanced packages (described later).

The following is an example of a ZONEBUDGET name file:

```
BEGIN ZONEBUDGET
  BUD mymodel.bud
  ZON zbud.zon
  GRB mymodel.dis.grb
END ZONEBUDGET
```

ZONEBUDGET will create two output files, with the ".lst" and ".csv" file extensions. The root file name for these files will be set to the root file name of the ZONEBUDGET name file. Thus, if the user does not change the default name file, then these files will be named "zbud.lst" and "zbud.csv". ZONEBUDGET will overwrite these files if they exist.

An example of the budget table written to the listing file by ZONEBUDGET is shown below.

```
VOLUME BUDGET FOR ZONE 1 AT END OF TIME STEP    1, STRESS PERIOD    1
-----------------------------------------------------------------------------------------

    CUMULATIVE VOLUME      L**3        RATES FOR THIS TIME STEP      L**3/T        PACKAGE/MODEL
    -----------------                  ------------------------                   ----------------

         IN:                                IN:
         ---                                ---
       STORAGE-SS =          0.0000        STORAGE-SS =          0.0000
       STORAGE-SY =          0.0000        STORAGE-SY =          0.0000
            WELLS =          0.0000             WELLS =          0.0000     WEL
    RIVER LEAKAGE =      49404.5424     RIVER LEAKAGE =      49404.5424     RIV
   HEAD DEP BOUNDS =         0.0000    HEAD DEP BOUNDS =         0.0000     GHB-TIDAL
         RECHARGE =          0.1230          RECHARGE =          0.1230     RCH-ZONE_1
         RECHARGE =      2.6350E-02          RECHARGE =      2.6350E-02     RCH-ZONE_2
         RECHARGE =      8.2325E-02          RECHARGE =      8.2325E-02     RCH-ZONE_3
              EVT =          0.0000               EVT =          0.0000     EVT
```

```
          TOTAL IN =       49404.7740              TOTAL IN =       49404.7740

          OUT:                                     OUT:
          ----                                     ----
       STORAGE-SS =           0.0000            STORAGE-SS =           0.0000
       STORAGE-SY =           0.0000            STORAGE-SY =           0.0000
            WELLS =           0.0000                 WELLS =           0.0000      WEL
    RIVER LEAKAGE =           0.0000         RIVER LEAKAGE =           0.0000      RIV
   HEAD DEP BOUNDS =       49404.7739        HEAD DEP BOUNDS =       49404.7739    GHB-TIDAL
         RECHARGE =           0.0000              RECHARGE =           0.0000      RCH-ZONE_1
         RECHARGE =           0.0000              RECHARGE =           0.0000      RCH-ZONE_2
         RECHARGE =           0.0000              RECHARGE =           0.0000      RCH-ZONE_3
              EVT =           0.0000                   EVT =           0.0000      EVT


        TOTAL OUT =       49404.7739             TOTAL OUT =       49404.7739

        IN - OUT =          1.0223E-04            IN - OUT =          1.0223E-04

PERCENT DISCREPANCY =           0.00      PERCENT DISCREPANCY =           0.00
```

Note that the of the table reads "VOLUME BUDGET FOR ZONE 1". A separate table is written for each zone. Also note that the far-right side of the table contains the PACKAGE/MODEL name. This identifier is required for MODFLOW 6 applications because users may specify more than one package of the same type. In this table, for example, three separate recharge packages were specified, with the names RCH-ZONE_1, RCH-ZONE_2, and RCH-ZONE_3.

The CSV file is similar in structure to the list file. The CSV file displays the complete budget for one zone and one time in a single line. Each column has a separate inflow or outflow budget term. The rows can be sorted by time within the spreadsheet program, which makes it possible to easily see how each term changes with time.

Input Instructions for the Zone File

The zone file consists of a DIMENSIONS block and a GRIDDATA block. The format for the zone file is as follows:

```
BEGIN DIMENSIONS
  NCELLS <ncells>
END DIMENSIONS

BEGIN GRIDDATA
  IZONE [LAYERED]
    <izone(ncells) --- READARRAY>
END GRIDDATA
```

where:

- NCELLS <ncells>—is the keyword and number of cells (or reaches) contained in the budget file.

- <izone>—is an integer array containing the zone number for each cell in the grid. IZONE values can be positive or negative integers. An IZONE value of zero indicates cells for which no budget is desired

or calculated. The LAYERED option can only be used if a binary grid file is provided, and the model is layered. For GWF Models, the DIS or DISV Package must be used in order for the model to be layered. This izone array is read by the READARRAY utility. READARRAY allows an array to be read in several different forms:

- CONSTANT iconst
  The word "CONSTANT" is a keyword that signifies that the entire array should have the same value, which is specified in the ICONST field.

- INTERNAL [FACTOR ifact] [IPRN iprn]
  "INTERNAL" indicates that the zone values for the grid are read from the zone file immediately following the control record. FACTOR ifact is a keyword and multiplier for the array; they are not required. If specified ifact should normally be set to 1. The values are read one at a time, starting with the first cell, using Fortran free format. The optional IPRN iprn keyword and integer value specifies whether or not the zone values that are read from the zone input file are also printed to the output file. If IPRN is greater than or equal to 0, the zone values will be printed. If IPRN is less than 0, zone values will not be printed.

- OPEN/CLOSE fname [FACTOR ifact] [IPRN iprn]
  "OPEN/CLOSE" indicates that the zone values for the grid are read from a separate file with the name fname. FACTOR ifact is a keyword and multiplier for the array; they are not required. If specified ifact should normally be set to 1. The values are read one at a time, starting with the first cell, using Fortran free format. The optional IPRN iprn keyword and integer value specifies whether or not the zone values that are read from the zone input file are also printed to the output file. If IPRN is greater than or equal to 0, the zone values will be printed. If IPRN is less than 0, zone values will not be printed.

The following is an example of a zone file:

```
BEGIN DIMENSIONS
  NCELLS 12
END DIMENSIONS

BEGIN GRIDDATA
  IZONE
    INTERNAL FACTOR 1 IPRN 0
    1  1  1  1  2  2  2  2  3  3  3  3
END GRIDDATA
```

Using ZONEBUDGET to Process an Advanced Package Budget File

ZONEBUDGET can also be used to process the binary budget file for the advanced stress packages (SFR, LAK, MAW, and UZF). The budget file for these packages has a similar form to the budget file written by the GWF Model. When applying ZONEBUDGET to the output of one of these packages, the GRB entry in the ZONEBUDGET name file should be excluded, and the number of cells specified in the zone input file should be set to the number of reaches (SFR Package), number of lakes (LAK Package), number of multi-aquifer wells (MAW Package), or the number of unsaturated zone flow cells (UZF Package).

History

This section describes changes introduced into MODFLOW 6 with each official release. These changes may substantially affect users.

- Version 6.4.0
    - Fixed error that prevented the program from working properly when the path to the ZONEBUD-GET executable contained one or more spaces in it.

- Version 6.1.0—December 12, 2019
    - Added functionality to allow for non-sequential zone numbers. User-specified zone numbers are internally converted to sequential zone numbers but results are still reported using user-specified zone numbers. This functionality eliminates memory allocation issues when large zone numbers (for example, Hydrologic Unit Codes (HUC)) are used. This functionality also permits use of negative zone numbers.
    - Fixed issue related to deferred length character array dfntxt used in grb.f90 to read ascii header data in grb file when compiled with gfortran version 4.9. Revised code to use deferred length characters variables instead of deferred length character arrays.

- Version 6.0.2—February 22, 2018
    - None.

- Version 6.0.1—September 28, 2017
    - Added the ability to specify LAYERED input for the IZONE array for GWF Models that use the DIS or DISV Packages. Layered input requires that a binary grid file is provided to ZONEBUD-GET.
    - Added additional information to the listing file to help with error detection.

- Version mf6beta0.0.02—May 19, 2017
    - Corrected this ZONEBUDGET user guide with the zone array instructions.

- Version mf6beta0.9.00: Beta release of ZONEBUDGET 6

Known Issues

This section describes known issues with this release of ZONEBUDGET 6.

1. None.

Installation

There is no installation of ZONEBUDGET 6. To make the executable versions of ZONEBUDGET 6 accessible from any directory, the directory containing the executables should be included in the PATH environment variable. Also, if a prior release of ZONEBUDGET 6 is installed on your system, the directory containing the executables for the prior release should be removed from the PATH environment variable.

As an alternative, the executable file (named zbud6.exe on Microsoft Windows) in the mf6.4.1/bin directory can be copied into a directory already included in the PATH environment variable.

System Requirements

ZONEBUDGET 6 is written in Fortran. It uses features from the 95, 2003, and 2008 language. The code has been used on UNIX-based computers and personal computers running various forms of the Microsoft Windows operating system.

Disclaimer and Notices

This software is preliminary or provisional and is subject to revision. It is being provided to meet the need for timely best science. The software has not received final approval by the U.S. Geological Survey (USGS). No warranty, expressed or implied, is made by the USGS or the U.S. Government as to the functionality of the software and related material nor shall the fact of release constitute any such warranty. The software is provided on the condition that neither the USGS nor the U.S. Government shall be held liable for any damages resulting from the authorized or unauthorized use of the software.