Programs for calibration-based Monte Carlo simulation of recharge areas

Supplemental material

Instructions for installing Python and associated modules

There are two ways to execute these scripts. The binary executable files can be run in the Windows® operating systems by opening them in Windows Explorer or by invoking them in a batch file. The binary files include the relevant Python modules, which are loaded into memory when the script is executed, making the executable files rather large and the initial loading of the script somewhat slow. The speed of the computations performed by the scripts is not affected by the large file size or slow loading of the modules.

If Python and additional required modules are installed on the user's computer, the scripts can be run directly. In addition to Python, the modules NumPy (for Numerical Python) and SciPy (for Scientific Python) are required. The scripts were developed and tested using Python 2.7.2, NumPy 1.6.1, and SciPy 0.9, although other versions may work. The following files should be installed in this order: (1) Python 2.7.2 (from http://www.python.org/download/), (2) NumPy 1.6.1 (from http://sourceforge.net/projects/numpy/files/NumPy/1.6.1 Note: use numpy-1.6.1-win32-superpack-python2.7.exe), and (3) SciPy 0.9 (from http://sourceforge.net/projects/scipy/files/scipy/ Note: use scipy-0.9.0-win32-superpack-python2.7.exe). The superpack installation files will detect the user's Python installation and install NumPy and SciPy correctly. Some computer systems may require the user to have Administrator privileges to install Python.

Prior to running the scripts

Before these scripts can be used, the user needs to have successfully estimated parameters of a groundwater flow model using MF2K (Harbaugh et al. 2000 and Hill et al. 2000), UCODE (Poeter et al. 2005), or PEST (Doherty 2008). This paper is not intended to document the use of modeling or parameter-estimation computer programs; familiarity with these and the techniques used therein is assumed. More information on parameter estimation can be found in Hill and Tiedeman (2007) as well as in the documentation for MF2K, UCODE, and PEST.

Files containing optimal parameter values, standard deviations, and correlation matrices are needed to generate realizations. These files are produced MF2K, UCODE, or PEST, although the file formats are slightly different. Optimal parameter values are read from one file. Another file containing the parameter variance-covariance matrix is used to calculate parameter standard deviations by taking the square root of the diagonals of the matrix. A third file contains the parameter correlation matrix, which is used to impose correlation on the randomly selected parameter values. As an alternative to using the nonlinear regression correlation matrix, the user could create the correlation matrix manually using the file format of MF2K or UCODE. Correlation matrices that are created by the user may not meet the requirement of being positive definite, but an iterative routine within GEN_LHS will attempt to modify the matrix to make it positive definite, as suggested by Iman and Davenport (1982).

The user also must create an error-free MODPATH simulation according to the following guidelines. Contributing recharge areas will be delineated using a forward particle tracking run of

MODPATH. Particles should be started over an area larger than the expected probabilistic contributing recharge area; placing particles at all recharge boundaries is one good strategy. The boundary array in the main MODPATH file is used to identify the model cell(s) containing the sink(s). This identifier is recorded in the particle endpoint file and identifies which particles terminate at the target cell. For example, the MODPATH boundary array (not the MODFLOW boundary array) in each of the test cases uses a zone code of 70 to identify the sink location.

Description of test cases

Test cases from the MODFLOW-2000 distribution are provided to verify that the scripts run without errors and to illustrate their input structure. Selected test cases were converted to MODFLOW-2005/UCODE and MODFLOW-2005/PEST to demonstrate the different input/output requirements among these nonlinear regression programs. More details are given at the end of this document. The Python scripts GEN_LHS and MCDRIVER_LHS, documentation, and test cases are distributed in a compressed file called MCDRIVER.zip. The user simply needs to extract the files. The directory created from MCDRIVER.zip is structured to simplify running the test cases, but the scripts do not depend on any particular directory structure. Batch files that use relative path names are included to run the test cases on the Windows® operating system within the MCDRIVER directory structure. After extracting the compressed distribution file MCDRIVER.zip, the following test cases will be in the directory MCDRIVER\test_cases.

1. MODFLOW-2000
    a. ETSDRT—Described by Banta (2000)
    b. ETSDRT_noisy_obs—ETSDRT with random normal error having a standard deviation of 0.5 feet added to observed heads
    c. TC2—Described by Hill et al (2000)
    d. TC2_noisy—TC2 with random normal error having a standard deviation of 10.0 feet added to observed heads
    e. TC3—Described by Cooley and Naff (1990)
2. MODFLOW-2005_UCODE—same as above but modified to work with MODFLOW-2005 and UCODE. Test case ETSDRT is not included because it uses drain observations and MODFLOW-2005 does not support drain observations. Drain observations could be included by the user by customizing an instruction file for UCODE to read drain observations from model output.
    a. TC2
    b. TC2_noisy—TC2 with random normal error having a standard deviation of 5.0 feet added to observed heads
    c. TC3
3. PEST
    a. TC3

Each test case directory has 3 subdirectories:

1. Mc_in—contains files for running the parameter-estimation model and the Monte Carlo simulation, including the batch file to run MCDRIVER_LHS. Also contains output from Monte Carlo simulation.
2. Output—contains one pre-calculated RANVAR file produced by running GEN_LHS and three output files produced running MCDRIVER_LHS (see detailed input instructions—step 3 for a description of the files). The RANVAR file is included in case the user wishes to run MCDRIVER_LHS without running parameter estimation first. The three

output files are provided for comparison of the test_case results with those produced using the pre-calculated RANVAR file.

3. Test-win—contains batch files to run parameter estimation and GEN_LHS, along with a response file that provides responses to the screen prompts from GEN_LHS so that the user does not have to type them in.

In each of the test cases, GEN_LHS was used to generate 100 repetitions of 20 quantiles of the parameter distributions, but MCDRIVER_LHS was run with a lower maximum number of iterations (20 in most test cases) to reduce execution time. In general, more iterations should be run. The modeling software needed to run the test cases is included in MCDRIVER/test_cases/bin. MODFLOW-2000, MODFLOW-2005, MODPATH, and UCODE also are available at *http://water.usgs.gov/nrp/gwsoftware/* (accessed July 28, 2010). PEST is available from *http://www.pesthomepage.org/* (accessed July 28, 2010).

Simplified instructions for running test cases

Instructions are given for the ETSDRT test case. This test case is in MCDRIVER\test_cases\MODFLOW-2000\ETSDRT. Instructions for other test cases are identical except for substituting the name of the test case.

1. Extract MCDRIVER.zip, leaving the file structure intact.
2. Run parameter estimation by opening MCDRIVER\test_cases\MODFLOW-2000\ETSDRT\test-win\ETSDRT.bat.
3. Run GEN_LHS by opening MCDRIVER\test_cases\MODFLOW-2000\ETSDRT\test-win\lhs.bat.
4. Run MCDRIVER_LHS by opening MCDRIVER\test_cases\MODFLOW-2000\ETSDRT\mc_in\mc.bat.

Instruction 2 will create the optimal parameter estimates, standard deviations, and correlation files (ETSDRT._PC, ETSDRT._MV, and ETSDRT._MC respectively). Instruction 3 will read four pieces of information from the provided response file. The first response is the name of the parameter estimation program used: MF2K, UCODE, or PEST (in this example, MF2K). The response is not case sensitive. The next response is for the rootname of the output files, in this case “ETSDRT.” The next two responses are for the number of quantiles and the number of repetitions. In the test case, 20 quantiles and 100 repetitions are used. The file estdrt.RANVAR will be created in MCDRIVER\test_cases\MODFLOW-2000\ETSDRT\test-win. This file is the link between parameter estimation and Monte Carlo simulation. Instruction 4 will run MCDRIVER_LHS. Three output files described above will be created and written to MCDRIVER\test_cases\MODFLOW-2000\ETSDRT\mc_in. The results of the Monte Carlo simulation are provided in MCDRIVER\test_cases \MODFLOW-2000\ETSDRT\output for comparison to the results of running the ETSDRT test case, but the results from any two Monte Carlo simulations will only be similar and not identical because of the random component.

Detailed input instructions

Step 1—Generate realizations using GEN_LHS

Input Instructions

Four lines of screen input are needed. Alternatively, responses to the following questions can be placed in a file and passed to GEN_LHS as an argument in a batch file. Please see the test cases for examples of executing GEN_LHS using a batch file. The responses are not case sensitive. The first prompt is to choose whether the required matrices were generated using MF2K, UCODE, or PEST. A utility script called REC2UNDERSCORE is provided to convert the ".REC" format generated by PEST to the formats produced by UCODE. The second prompt is to supply the root file name, for example "*rootname*" if the correlation matrix file is "*rootname*._MC." The third prompt is for the number of quantiles. The last prompt is for the number of repetitions, where a repetition is defined as a set of realizations taken from all the quantiles of the complete parameter distribution. The total number of realizations equals the product of the number of quantiles and the number of repetitions. After successful completion, GEN_LHS displays a message containing the maximum element-wise difference between input and output correlation matrices. This is an indication of how well the original correlation matrix is represented by the realizations. Smaller differences indicate better representation. Increasing the number of realizations, either by increasing the number of repetitions or the number of quantiles, will decrease this value. The number of quantiles and repetitions to use is problem dependent, and the user is encouraged to explore the effects of using different values. As a starting point, 20 quantiles and 500 repetitions can be used.

Output files

Parameter realizations are written to a file named *rootname*.RANVAR. In the ".RANVAR" file, each row represents one realization and each column contains all generated values for the same parameter.

Step 2—Create a template file

A template file needs to be prepared by the user so that MCDRIVER_LHS knows how to write model input for each new realization. This file is a modified copy of an existing input file, which could be the SEN file in MF2K, the PVAL file in MF05, the file containing the PARAMETER_DATA block in UCODE, or the PST file in PEST. The file on which the template file is based must contain values for all the parameters represented in the RANVAR file. The input file should be modified as follows: any values to be replaced must be enclosed by @ symbols. MCDRIVER_LHS will scan the template file and create a new input file by replacing any combination of text or spaces between pairs of @ symbols with the next parameter value from the file of parameter realizations (RANVAR). MCDRIVER_LHS does not check that the parameters are in any particular order; therefore, the number of replacements to make and the order in which they are made must match the order and number of parameters in the RANVAR file. An example of a template file is in MCDRIVER\test_cases\MODFLOW-2000\ETSDRT_noisy_obs\mc_in\ ETSDRT_mc.tplt.

Step 3: Run probabilistic simulation using MCDRIVER_LHS

Input instructions

The user must generate a text file called MCDRIVER.IN that is located in the same directory in which MCDRIVER_LHS is executed. Input variables are specified with one value per line; only the first whitespace-delimited item is read from each line; any other values on the line are ignored. The order of input variables is given below. Please see MCDRIVER\test_cases\MODFLOW-2000\ETSDRT_noisy_obs\mc_in\ MCDRIVER.IN as an example.

Line Description

1. The maximum number of model runs. The Monte Carlo simulation terminates when this number of runs is reached, but it may terminate earlier if the convergence criterion is met.
2. The total number of forward-tracked particles used in MODPATH.

3. The name of the MODFLOW list file. The first two acceptance criteria are read from this text file according to whether MF2K or MF05 was used.
4. The name of the UCODE ".#uout" or PEST ".rec" file. The third acceptance criterion (standard error) is calculated from values in this file. If MF2K is used, put a dummy name on this line.
5. The name of the MODPATH endpoint file.
6. The name of the RANVAR file containing parameter realizations that was generated using GEN_LHS.
7. The zone code used in the MODPATH boundary array to identify sinks.
8. The highest acceptable volumetric balance error, in percent. Typical values are 0.01 to 0.05.
9. The highest acceptable standard error of the regression. Acceptable values depend on the simulation. The user can start with a value on the order of twice the standard error of the regression, which is produced by parameter estimation and contained in the global file for MODFLOW-2000, the .#uout file for UCODE, or the rec file in PEST.
10. The name of model software used for the forward model. Either "MF2K", "UCODE", or "PEST." The latter two are included because they can be used to include residence times or solute concentrations in the model output.
11. The stopping criterion, based on the maximum absolute difference in particle probability from the previous model run. Typical values are 0.01 or lower.
12. The name of the batch file or executable file (including pathname) that runs the forward model. The forward model can be run by UCODE or PEST; this allows residuals for observation types that are not supported by MODFLOW, such as residence times, to be included in the standard error calculation. The name of the batch file or executable file (including pathname) that runs MODPATH. Make
13. The name of the template file (including pathname) used for substituting parameter realizations into model input files. The model is executed in forward mode in MCDRIVER_LHS, so parameter estimation should be turned off in the template file.
14. The name of the model parameter input file(including pathname) that is generated by the template file. For MODFLOW-2000, this is the sensitivity input file, usually with an extension of ".SEN". For UCODE, this is the file containing the PARAMETER_DATA block, which can be in the UCODE input file, or which can be a separate file, usually with an extension ".PAR". For PEST, this is name of the PEST Control File.
15. The number of the column in the file or data block that contains the flag indicating log-transformation of the parameter. The column numbering starts with 0.
16. Name of file containing weighted residuals.
17. Number by which to divide MODPATH travel times. For example, this should be 365.25 if model output is in days and you want to report results in years.
18. The number of quantiles selected in GEN_LHS.

Output files

These three files are generated by MCDRIVER_LHS.

1. Endpoint.txt contains one line for each particle and one column for each model run. The values in the file are residence times, which can be used to generate probabilistic descriptions of residence time or to examine the characteristics of a particular model run in terms of residence time.
2. Parameters.txt contains one line for each model run. The columns correspond to the parameter set number, the iteration number, four columns indicating whether the four

criteria were met (0 is not met and 1 is met), the total of the previous four columns (a "4" indicates that all four criteria were met), the mass balance error, the standard error, the number of particles reaching the target, the maximum change in probability, and the cumulative number of iterations accepted.

3. Probpoints.txt contains one line for each particle starting location. The first two columns are the x and y location of the particle starting location, in model grid coordinates, taken from the MODPATH endpoint file. The third column is a dummy variable and should be ignored (this variable may be used in future revisions of the script). The fourth column is the probability that the particle reaches the target. The format of this file is such that it can be imported into plotting software or a Geographic Information System.

Post processing in a Geographic Information System

There are many ways to present the output of MCDRIVER_LHS graphically. The most straightforward is to display the particle starting x and y locations, which are contained in the ProbPoints.txt file, using any plotting software or Geographic Information System. In most such software, the color of the point can be assigned a color based on a third value, in this case the probability value. If this is done, the spatial pattern of color represents the effects of parameter uncertainty on the contributing area. Alternatively, the point data can be mapped onto polygons so that continuous areas can be shown. A Python script (MAKE_SHAPEFILE) is provided that uses particle starting locations to generate Thiessen polygons in the ArcGIS Geographic Information System [product names are for identification only and do not constitute endorsement of the software]. In order to do this, the user's computer must have either ArcGIS 9.3 or ArcGIS 10.0 installed. The resulting polygons are in the shapefile format and are useful for checking results of MCDRIVER_LHS or for visualizing the results, but they are not intended for publication-quality graphics. An example of how to run MAKE_SHAPEFILE from a batch file is given in MCDRIVER\test-output\MODFLOW-2000\ETSDRT_noisy_obs\mc_out\makeshape.bat. Operation of MAKE_SHAPEFILE may vary by computer set-up, and the user may have to modify the script. Detailed instruction on scripting in ArcGIS is beyond the scope of this paper.

Detailed test descriptions

Test case TC2 (Hill et al. 2000)

Grid dimensions: 3 Layers, 18 Rows, 18 Columns
Stress periods: 1 steady
Optional processes: OBS, SEN, PES
Mode: Parameter Estimation
Flow package: LPF
Stress packages: WEL, DRN, EVT, GHB, RCH
Solver: PCG2
Observation types:
  HOB (42 observations)
  DROB (5 observations)
  GBOB (5 observations)
Types of parameters defined / package:
  HK / LPF
  VANI / LPF

DRN / DRN
EVT / EVT
GHB / GHB
RCH / RCH

Test case TC3 (Cooley and Naff 1990)

Grid dimensions: 1 Layer, 17 Rows, 16 Columns
Stress periods: 1 steady
Optional processes: OBS, SEN, PES
Mode: Parameter Estimation
Flow package: LPF
Stress packages: WEL, RIV, CHD, RCH
Solver: PCG2
Observation types:
HOB (32 observations)
Types of parameters defined / package:
HK / LPF
Q / WEL
RIV / RIV
RCH /RCH
CHD /CHD

Test case ETSDRT (Banta 2000)

Grid dimensions: 1 Layer, 11 Rows, 11 Columns
Stress periods: 1 steady
Optional processes: OBS, SEN PES
Mode: Parameter Estimation
Flow package: LPF1
Stress packages: DRT, ETS, RCH
Solver: PCG
Types of parameters defined / package:
ETS / ETS
DRT / DRT
RCH / RCH
HK / LPF

References Cited

Banta, E.R. 2000 MODFLOW-2000, The U.S. geological survey modular ground-water model—documentation of packages for simulating evapotranspiration with a segmented function (ETS1) and drains with return flow (DRT1): U.S. Geological Survey Open-File Report 00–466. Reston, Virginia: USGS.

Cooley, R.L. and R.L. Naff. 1990. Regression modeling of ground-water flow: U.S. Geological Survey Techniques of Water-Resources Investigations, Book 3, Chapter B4. Reston, Virginia: USGS.

Doherty, J., 2008. Manual and Addendum for PEST: Model Independent Parameter Estimation. Brisbane, Australia: Watermark Numerical Computing.

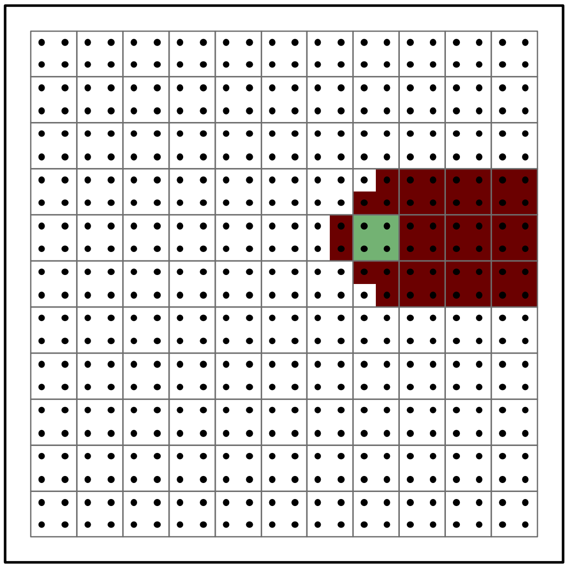
Harbaugh, A.W., E.R. Banta, M.C. Hill, and M.G. McDonald. 2000. MODFLOW-2000, the U.S. Geological Survey modular ground-water model—User guide to modularization concepts and the ground-water flow process: U.S. Geological Survey Open-File Report 00–92. Reston, Virginia: USGS.

Hill, M.C., E.R Banta, A.W. Harbaugh, and E.R. Anderman. 2000. MODFLOW-2000, The U.S. Geological Survey modular ground-water model—User guide to the observation, sensitivity, and parameter-estimation processes and three post-processing programs: U.S. Geological Survey Open-File Report 00-184. Reston, Virginia: USGS.

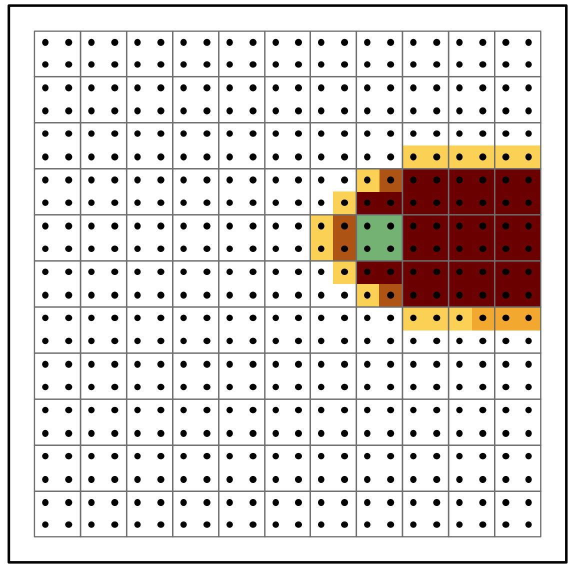
Hill, M.C. and C.R. Tiedeman. 2007. Effective groundwater model calibration with analysis of data, sensitivities, and uncertainty. New Jersey: John Wiley and Sons.

Iman, R.L. and J.M. Davenport. 1982. An iterative algorithm to produce a positive definite correlation matrix from an “approximate correlation matrix” (with program user’s guide). SANDIA Report SAND81-1376. Albuquerque, New Mexico: Sandia National Laboratories.

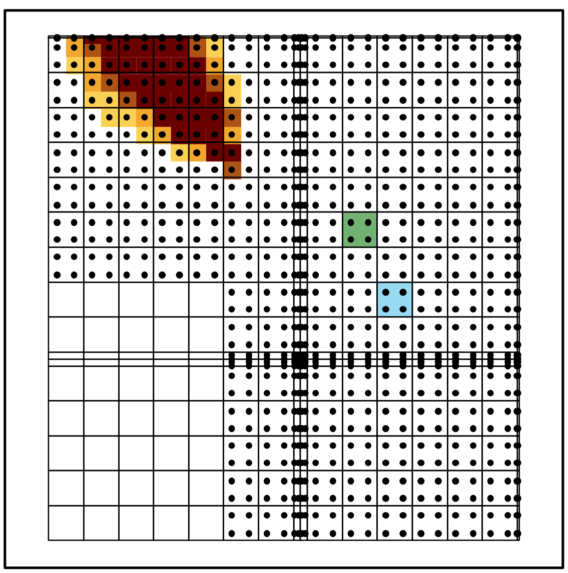
Poeter, E.P., M.C. Hill, E.R. Banta, S. Mehl, and S. Christensen. 2005. UCODE_2005 and six other computer codes for universal sensitivity analysis, calibration, and uncertainty evaluation. U.S. Geological Survey Techniques and Methods 6-A11. Reston, Virginia: USGS.



A. ETSDRT with perfect calibration data.
Drain sink cell in green.



B. ETSDRT with noisy calibration data.
Drain sink cell in green.



C. TC3 with 100 repetitions of parameter distributions. Recharge area shown is for a well in green cell. A well in the blue cell is simulated in MODFLOW but not in the Monte Carlo simulation. Finer grid cells correspond to simulated rivers.

EXPLANATION

Probability that particle is in the source area to receptor (a well in TC3 and a drain in ETSDRT)

- 0.0 to 0.20
- > 0.20 to 0.40
- > 0.40 to 0.60
- > 0.60 to 0.80
- > 0.80 to 1.00

Grid cell

Starting location of a particle in an active cell