MODFLOW – Advanced PEST: SVD, SVD-Assist, Parallel PEST

1 Introduction

The *MODFLOW-Automated Parameter Estimation* tutorial describes the basic functionality of PEST provided in GMS. It illustrates how to parameterize a MODFLOW model and run PEST to obtain optimal parameter values. This tutorial describes new PEST features available in GMS starting with version 7.1. These features include: Truncated Singular Value Decomposition (SVD), SVD-Assist, and Parallel PEST.

Each of these features is intended to decrease the amount of time necessary for PEST to complete the optimization process. The SVD options focus on identifying and removing insensitive parameters from the model inversion process, while Parallel PEST speeds up the inversion process by running more models simultaneously. Since most computers used now have multi-core processing capability, Parallel PEST can run on a single computer and run multiple instances of the MODFLOW model on the various cores of the computer.

1.1 Contents

1 Introduction **1-1**
1.1 Contents 1-1
1.2 Outline 1-2
1.3 Required Modules/Interfaces 1-2
2 Description of Problem **2-2**
3 Getting Started **3-3**
4 Reading in the Project **4-3**
5 Running PEST With SVD Option **5-3**

6 Running Parallel PEST ... 6-4
7 Saving the Project and Running Parallel PEST ... 7-5
8 Running Parallel PEST with SVD-Assist ... 8-5
9 Conclusion ... 9-6

1.2 Outline

This is what you will do:

1. Open a MODFLOW model and solution.
2. Run PEST using SVD.
3. Run Parallel PEST.
4. Run Parallel PEST using SVD-Assist.

1.3 Required Modules/Interfaces

You will need the following components enabled to complete this tutorial:

- Grid
- Geostatistics
- Map
- MODFLOW
- Inverse Modeling
- Parallel PEST

You can see if these components are enabled by selecting the *Help* | *Register* command.

2 Description of Problem

The model we will be calibrating in this tutorial is the same model featured in the *MODFLOW-Model Calibration* tutorial. The model includes observed flow data for the stream and observed heads at a set of scattered observation wells. The conceptual model for the site consists of a set of recharge and hydraulic conductivity zones. These zones will be marked as parameters and an inverse model will be used to find a set of recharge and hydraulic conductivity values that minimize the calibration error. We will utilize two methods for model parameterization: polygonal zones and pilot point interpolation.

3 Getting Started

Let's get started.

1. If necessary, launch GMS. If GMS is already running, select the *File* | *New* command to ensure that the program settings are restored to their default state.

4 Reading in the Project

First, we will read in the modeling project:

1. Select the *Open* button.
2. Locate and open the **tutfiles\MODFLOW\advPEST** directory.
3. Open the file entitled **run1.gpr**.

You should see a MODFLOW model with a solution and a set of map coverages. Three of the coverages are the source/sink, recharge, and hydraulic conductivity coverages used to define the conceptual model. The active coverage contains a set of observed head values from observation wells. If you switch to the source/sink coverage, you will notice that an observed flow value has been assigned to the stream network.

5 Running PEST With SVD Option

We will now run PEST with the current model.

1. Select the *MODFLOW* | *Run MODFLOW* menu command.

This model will run for 10 iterations. Depending on the speed of your computer this may take a few minutes. Notice that after 10 iterations the final model error is 136. We will now run the model with the SVD option turned on.

2. Select the *Close* button to exit the dialog.
3. Select the *MODFLOW* | *Parameter Estimation* menu command.
4. Turn on the *Use SVD* option.
5. Select the *SVD Options* button.

This dialog allows the user to edit the values of the MAXSING, EIGTHRESH, and EIGWRITE variables associated with the SVD process. MAXSING is the maximum number of singular values to include in the inversion process. This value is problem dependent. Often a more appropriate approach is to use EIGTHRESH to limit the number of singular values to include in the inversion process. EIGTHRESH is a ratio of lowest to highest eigenvalues at which truncation is implemented. Only those parameters are estimated whose eigenvalue divided by the maximum eigenvalue exceeds EIGTHRESH.

Finally, EIGWRITE controls items written to the *.svd file. If the value of EIGWRITE is 1 then a more verbose *.svd file is written. The default values work well for the current model.

6. Select *OK* twice to exit both dialogs.
7. Select the *File* | *Save As* command.
8. Enter **run2** for the name of the project and select *OK*.
9. Select the *MODFLOW* | *Run MODFOW* menu command.

Notice that on this model run PEST was able to complete the run in 7 iterations and the final error was less than 1.

10. Select *Close* to exit the dialog.

6 Running Parallel PEST

All of the steps required to run PEST must also be followed to run Parallel PEST. The only additional inputs required by the user are to specify the number of models that can be run simultaneously by PEST as well as a “WAIT” variable. The WAIT variable is described by the PEST documentation as the amount of time that PEST and PSLAVE will pause at certain strategic places in their communication. Normally the default value should work fine. However, if either PEST or PSLAVE reports a sharing violation on your hard drive then you should increase the value of the WAIT variable.

To illustrate Parallel PEST and SVD-Assist we will start with a different project.

1. Select the *File* | *New* command.
2. Select the *Open* button .
3. Locate and open the **tutfiles\MODFLOW\advPEST** directory.
4. Open the file entitled **run1_SvdAssist.gpr**.
5. Select the *MODFLOW* | *Global Options* menu command.
6. Select the *Parameter Estimation* option under *Run options* and select *OK* to exit the dialog.
7. Select the *MODFLOW* | *Parameter Estimation* menu command.
8. Change the *Max # iter. (NOPTMAX)* to **2**.
9. Turn on the *Use Parallel Pest* toggle.
10. Select *OK* to exit the dialog.

7 Saving the Project and Running Parallel PEST

Now we will save our project prior to running parallel PEST.

1. Select the *File* | *Save As* command.
2. Locate the **tutfiles\MODFLOW\ppest** folder.
3. Enter **run2_PPest** for the project name and select the *Save* button.
4. Select the *Run MODFLOW* button.

At this point new command prompt windows are created depending on the number of slaves that can run on your computer. These command prompts that are initially mimimized are running PSLAVE. Also notice that the *MODFLOW-2000/PEST Parameter Estimation* dialog is available from GMS. This dialog reads the output from Parallel PEST and updates the model error and parameter values for each PEST iteration. If the user selects the *Abort* button from this dialog then all of the Parallel PEST processes will be terminated. In addition, selecting the *Stop w/ Statistics* button will allow Parallel PEST to stop at its current iteration.

Parallel PEST should take a few minutes to run two iterations with the current model. On each of the PEST iterations MODFLOW is run 129 times which is once for each pilot point that is being estimated.

5. Select the *Close* button on the *MODFLOW-2000/PEST Parameter Estimation* dialog.

8 Running Parallel PEST with SVD-Assist

Now we will run Parallel PEST again with the SVD-Assist option turned on. This process is particularly advantageous for models that have hundreds or thousands of parameters (such as pilot points).

SVD-Assist involves 3 basic steps. First, PEST runs MODFLOW once for every parameter in order to compute a matrix. This information is used to create super parameters that are combinations of the parameters originally specified. Second, SVDAPREP is run to create a new PEST control file. The options for SVDAPREP are entered by selecting the *SVD-Assist Options* button. For more information on each of these options see the PEST manual in section 8.5.4.2. The most important option entered is the *Specify # super param* and this is set to **No** by default. When this option is set to No, then the information written to the *.svd file will be used to specify the number of super parameters. Third, PEST runs using the new control file written by SVDAPREP. This should result in significantly fewer model runs for each PEST iteration. This often results in an order of magnitude reduction in the number of runs required for each PEST iteration.

1. Select the *MODFLOW* | *Parameter Estimation* menu command.
2. Change the *Max # iter. (NOPTMAX)* to **15**.

3. Turn on the *Use SVD* and the *Use SVD-Assist* toggles.
4. Select *OK* to exit the dialog.
5. Select the *File* | *Save As* menu command.
6. Enter **run2_SvdAssist** for the project name and select the *Save* button.
7. Select the *Run MODFLOW* button.

Notice that each PEST iteration now only requires 11 MODFLOW runs instead of 129. When PEST finishes running the error should be around 0.1.

8. Select the *Close* button on the *MODFLOW-2000/PEST Parameter Estimation* dialog.

When PEST runs with SVD-Assist the parameter values are not written to the *.par file because this file contains the values for the super parameters. The base parameter values are written to a *.bpa file.

9. Select the *MODFLOW* | *Parameters* menu command.
10. Select the *Import Optimal Values* command.
11. Select the **run2_SvdAssist.bpa** file and select Open.

9 Conclusion

This concludes the *MODFLOW – Advanced PEST: SVD, SVD-Assist, Parallel PEST* tutorial. Here are the things that you should have learned in this tutorial:

1. GMS supports the SVD option for PEST.
2. You can use Parallel PEST from GMS.
3. You can run SVD-Assist from GMS by simply turning on the option.