Repository information and instructions

This repository contains the dataset used in the research "Propagation of hydropoeaking waves in heterogeneous aquifers: effects on flow topology and uncertainty quantification," written by P. Merchán-Rivera, M. Basilio Hazas, G. Marcolini, and G. Chiogna.

CONTENT

The dataset contains two main folders:

1. Scripts: contains the Python codes to reproduce the research results.
2. Videos: contains a series of videos of the research results as an extension to those presented in the manuscript.

REPRODUCTION OF RESULTS AND EXPERIMENT

Requirements and installation:
Download the dataset and keep the folders in the original location.

We recommend using Spyder as integrated development environment (IDE) to visualize the results straightforwardly. Also, MODFLOW-2005 must be previously installed in order to run these experiments.

The following libraries are required to run the code: numpy (version: 1.19.2), chaospy (version: 4.3.4), matplotlib (version: 3.3.2),

Optionally, the following libraries can be installed to create the videos (.avi files): opencv-python (version: 4.5.5.62) to create the videos. All packages can be installed using the pip package manager. Please, refer to the corresponding package documentation for installation instructions.

Deterministic scenarios:
- Open the folder ./scripts/1_deterministic
- Run the python scripts in the following order: 1) run_sim.py, 2) run_postproc_heads.py, 3) run_postproc_flowfield, 4) run_variance_results.py. The videos can be created with the simulated results with the code ./scripts/1_deterministic/images/create_movie.py
- The results are automatically save in the folder outputs and the figures in the folder images.
- The videos can be created with the simulated results by running the code ./scripts/1_deterministic/images/create_movie.py

Stochastic scenarios:
To reproduce the results from the stochastic scenarios, the folders are required: ./scripts/2_stochastic_simple, ./scripts/3_stochastic_complex, and ./scripts/4_stochastic_results.
Triangle, sine and trapezoid wave scenarios:

- Open the folder .scripts/2_stochastic_simple
- Run the python scripts in the following order: 1) run_sim.py, 2) run_postproc_heads.py, and 3) run_postproc_flowfield
- The results are automatically saved in the folder outputs. Now, these outputs can be moved to the folder .scripts/4_stochastic_results for postprocessing and create the resulting figures.
- Copy all the output files in the folders .scripts/2_stochastic_simple/outputs/sine, .scripts/2_stochastic_simple/outputs/triangle, and .scripts/2_stochastic_simple/outputs/trapezoid and paste the files within the folders .scripts/4_stochastic_results/outputs/sine, .scripts/4_stochastic_results/outputs/triangle, and .scripts/4_stochastic_results/outputs/trapezoid, respectively.
- Copy all the output files in the folder .scripts/2_stochastic_simple/uq_heads and paste in the folder .scripts/4_stochastic_results/outputs/uq_heads
- Copy all the output files in the folder .scripts/2_stochastic_simple/uq_topo and paste in the folder .scripts/4_stochastic_results/outputs/uq_topo

Complex wave scenarios:

- Open the folder .scripts/3_stochastic_complex
- Run the python scripts in the following order: 1) run_sim.py, 2) run_postproc_heads.py, and 3) run_postproc_flowfield
- The results are automatically saved in the folder outputs. Now, these outputs can be moved to the folder .scripts/4_stochastic_results for postprocessing and create the resulting figures.
- Copy all the output files in the folder .scripts/3_stochastic_complex/composed and paste in the folder .scripts/4_stochastic_results/outputs/composed
- Copy all the output files in the folder .scripts/3_stochastic_complex/uq_heads and paste in the folder .scripts/4_stochastic_results/outputs/uq_heads
- Copy all the output files in the folder .scripts/3_stochastic_complex/uq_topo and paste in the folder .scripts/4_stochastic_results/outputs/uq_topo

To postprocess and create figures:

- Open the folder .scripts/4_stochastic_results
- Run the python scripts in the following order: 1) fig_2d_uq, 2) fig_pdf_heads, 3) fig_uq_intime
- The videos can be created with the simulated results with the code .scripts/4_stochastic_results/images/create_movie.py

Final considerations:
Before running all the experiments, be aware that the average unit simulation is 120 seconds and requires a large RAM (16 or 32 GB) and hard disk space (~60 GB). Particularly, the complex wave stochastic scenario in which the quadrature nodes is K=9 and the polynomial
order is n=2 requires 10000 unit simulations. A lite version is included in this dataset (K=4, n=2), which yields the same experimental outcomes.

To reproduce the same experiment of the research, open the files
./scripts/3_stochastic_complex/set_sim.py and
./scripts/4_stochastic_results/set_sim_comp.py, and edit the variables called quadOrder to 9 to reproduce the same experiment of the research. Lighter versions of the experiment can be run by reducing the simulation period. However, no formal analysis of the computational demand has been done as part of this research. Please, contact pablo.merchan@tum.de for questions and issues regarding the simulation codes.

DATA AND CODE DISCLAIMER
The authors bear no responsibility for the uses of the data and the codes, or for interpretations or inferences based on these uses. The authors accept no liability for indirect, consequential, or incidental damages or losses arising from use of the data collection, or from the unavailability of, or break in access to the service for whatever reason.

License
This dataset is licensed under CC-BY 4.0. You can share, copy and modify this dataset so long as you give appropriate credit, provide a link to the CC BY license, and indicate if changes were made, but you may not do so in a way that suggests the rights holder has endorsed you or your use of the dataset. Note that further permission may be required for any content within the dataset that is identified as belonging to a third party.