



---

## PREDICT: PeRmEability Distributions of Clay-smeared faultS

MATLAB® r2020b License GPLv3 last commit repo not found

[Overview](#) • [Citation](#) • [Requirements](#) • [Download](#) • [Installation](#) • [Examples](#) • [License](#) • [Acknowledgements](#)

---

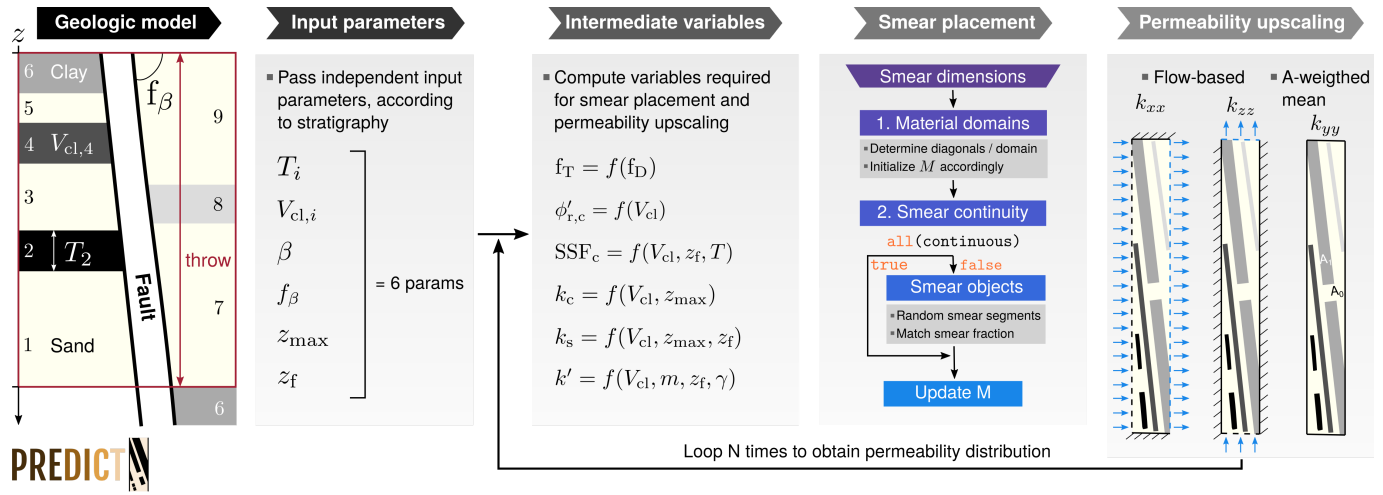
### Short Description

Computes upscaled fault permeability distributions using a physics-based, probabilistic description of clay and sand smears. MATLAB and MRST are required to run this code.

### Overview

**PREDICT** is a novel algorithm designed to *predict*, i.e., compute, the diagonal components of the fault permeability tensor (across/perpendicular to the fault,  $k_{xx}$ ; along strike,  $k_{yy}$ ; and up/down-dip,  $k_{zz}$ ) in relatively shallow siliciclastic sequences ( $< \sim 3$  km of depth). The computation is done for a given throw interval, and the material distributions and output permeability values are representative of the main shear zone within a faulted sediment volume. Hence, **PREDICT** represents an architectural domain typically referred to as the fault core (see figure below).

First, the algorithm takes a set of numerical quantities, the input parameters, that describe the faulted stratigraphy. Second, **PREDICT** generates samples for another set of numerical quantities based on the input parameters; these are intermediate random variables required to compute the dimensions, distribution and permeability of the clay- and sand-based fault zone materials. Third, object-based simulation is used to place the clay-smears within the fault. The sand-based fault zone material is placed next, according to the amount and location of the space available. Finally, the equivalent or upscaled fault permeability computation for the modeled throw interval is based on the material distribution within the fault, and uses flow-based upscaling for  $k_{xx}$  and  $k_{zz}$ , while area-weighted arithmetic average is used for  $k_{yy}$ . Steps two to four are repeated multiple times, each one representing one realization or simulation, until the full permeability distribution for each component is obtained (see figure below).



## Citation

TBD. Paper will be added here when available.

bibtex reference here.

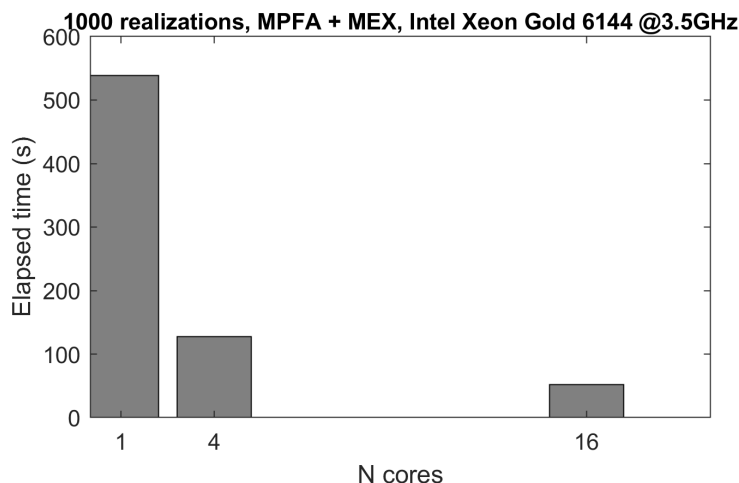
## Requirements

**Hardware:** When running thousands of simulations, a machine with multiple cores (4+) and few GB of RAM (6-8+) is recommended. Running times will be greatly reduced (see below) and some of the output structures may be somewhat heavy.

**Software:** PREDICT can be run on Windows, macOS and Linux (i.e. any OS where [MATLAB](#) can be installed). The code incorporates [MRST](#) functionality for flow-based permeability upscaling, so it requires an installation of both MATLAB and MRST (very straightforward, see steps below). Development took place using [MATLAB r2020b](#) and the development version of MRST. The code has also been tested with MATLAB r2020a and the current MRST public release ([v2021a](#)). Backward compatibility with previous MATLAB versions also supported by MRST 2021a is likely, but it has not been tested.

In addition, PREDICT uses the following MATLAB add-on toolboxes:

- [Statistics and Machine Learning Toolbox](#) (**required**): For generating intermediate variable distributions and samples.
- [Parallel Computing Toolbox](#) (*recommended*): Not required, but *highly* recommended for anyone using the code beyond exploration purposes. Time gains when running parallel simulations are illustrated below for 1000 realizations/simulations of a given stratigraphic case.
- [Curve Fitting Toolbox](#) (*recommended*): Not required, but may be useful for output analysis.



## Download

- **MATLAB:** Can be installed following the instructions [on the website](#), and your academic institution likely provides campus-wide access free of charge.
- **MRST:** The latest public release of MRST can be downloaded [here](#):
- **PREDICT:** The repository can be cloned or downloaded from [here](#) (green button "clone or download"). [This repository is private.]

## Installation

We show installation steps for both MRST and PREDICT.

1. Download the latest MRST release (see above).
2. Download PREDICT (see above).
3. From within MATLAB, run the `startup.m` file in the main MRST folder. You will be prompted whether you want to enable MEX acceleration, select enable. This can be changed later by running the command `mrstSettings()`.
4. From within MATLAB, right click on the `predict` folder and select "Add to Path > Selected Folders and Subfolders".

You can now run PREDICT (see Examples section below). Note that, for flow-based permeability upscaling, both a TPFA and a MPFA can be used. If you are running PREDICT without MEX, TPFA is recommended (otherwise, it will be slow).

## Examples

Examples are provided in the folder [examples](#). For a comprehensive introductory example, run `example0_singleStrati.m` or, in the live editor, `example0_singleStrati_live.mlx`:

## Example 0: Single stratigraphic case + analysis

This is a complete introductory example. It shows how to load the appropriate MRST modules, define the inputs according to a given faulted stratigraphy, and generate the output permeability distributions. A comprehensive analysis of the results is also shown.

We first make sure that the workspace is clean:

```
clear
close all force
```

## 1. Load Required MRST Modules

First, navigate to the mrst folder and run `startup.m`. We can then load the appropriate modules for generating MRST grids and upscale the permeability:

```
mrstModule add mrst-gui coarsegrid upscaling incomp mpfa
```

## 2. Define Model and Upscale Permeability

### 2.1 Mandatory input parameters

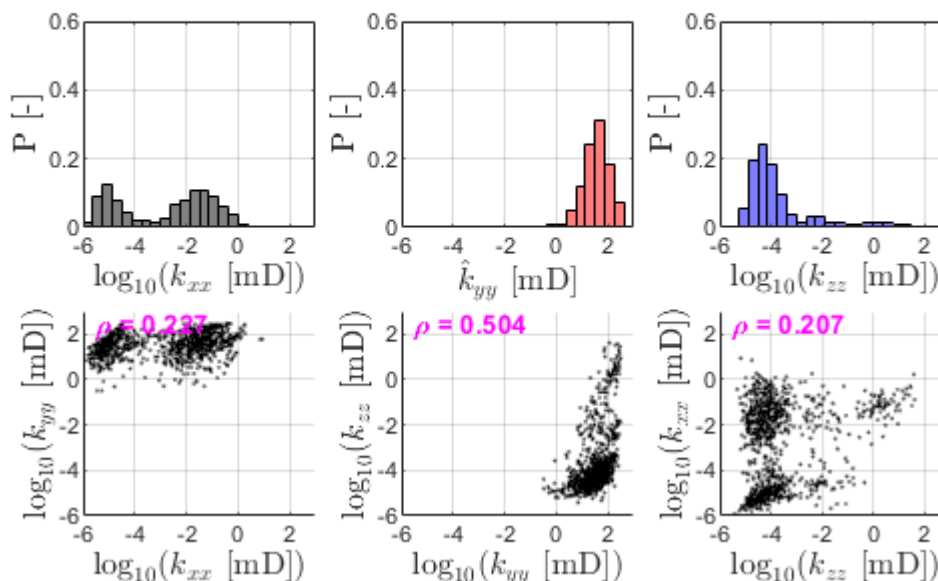
Footwall first and hangingwall next, e.g. `{[footwall, FW], [hangingwall, HW]}`. We need to define the layer thickness, clay content, layer dip angle, fault dip angle, faulting depth, and burial depth. Further details about input parameter formatting, etc can always be checked from the documentation in the classes and functions.

```
thickness = {repelem(25, 1, 4), [5 10 15 10 20 10 10 5 15]}; % [m]
vcl       = {[0.1 0.4 0.2 0.5], ... % fraction [-]
              [0.3 0.6 0.1 0.7 0.2 0.8 0.3 0.9 0.1]};
dip       = [0, -5]; % [deg.]
faultDip  = 70; % [deg.]
zf        = [500, 500]; % [FW, HW], [m]
zmax      = {repelem(2000, numel(vcl{1})), repelem(2000, numel(vcl{2}))}; % {FW, HW}
```

### 3.4. Visualize upscaled permeability

Plot upscaled permeability distributions (all simulations)

```
plotUpscaledPerm(faults)
```



PREDICT incorporates MRST functionality, so it legally becomes an extension of MRST. This means that, if publicly released, it should be done under the terms of the GPL license.

## Acknowledgements

This work was funded by **ExxonMobil** through the project *"Modeling and Mitigation of Induced Seismicity and Fault Leakage during CO2 storage"*. L.S. and R.J. greatly appreciate the numerous, productive discussions with the ExxonMobil team participating in the project.

L.S. would like to acknowledge **The MathWorks, Inc.** and the **School of Engineering at MIT** for funding through a 2020-2021 *MathWorks Engineering Fellowship*.

The authors would also like to thank Youssef Marzouk for his helpful comments on multivariate statistical modeling, as well as Olav Møyner and the MRST development team for outstanding support.

Readme design based on examples from [Awesome README](#).