Classical pore network two-phase flow simulator – pnflow

Prepare by Ali Q Raeini a.q.raeini@imperial.ac.uk

August 22, 2020

This document contains a list of instructions for compiling and running the pnflow two-phase network flow model and a list of the main keywords used in the input file to this network flow model.

The pnflow code is very similar to the poreflow, developed by Valvatne and Blunt (2004). The differences mostly are related to the I/O, visualization, and structure of the code. Specifically, this code supports a new network model format more sutable for data mining from micro-CT images of flow experiments on porous media. A brief summary of recent changes made in the code are given in the ChangeLog file in the folder containing pnflow source code.

Contents

1 Input file

Syntax: The pnflow input file is contains a set of "keyword: data;". The data can span multiple lines. Note that unlike in the poreflow code, we use semi-colon(;) to mark the end of keyword data. Anything following `%` or `//` are treated as comments and ignored.

Most of the keywords used in the original poreflow code are active here as well. This documentation primarily describes the main keywords read by pnflow. See the documentation for the original *two phase network modelling code* by Valvatne and Blunt (2004) for other keywords not discussed here: http://www.imperial.ac.uk/earth-science/research/research-groups/perm/research/pore-scale-modelling/software.

```
// -*- C -*- syntax style to highlight/comment...
TITLE: Berea; // base name for the output files

networkFile: BereaNet.xmf; // name of the network file

//!cycle# Final Sw Final Pc  Sw steps Compute: Kr  RI
cycle1:   0.00     2.00E+06   0.05           T   T;
Cycle2:   1.00    -2.00E+06   0.05           T   T;
cycle3:   0.00     2.00E+06   0.05           T   T;

//!cycle# Inject-from  Produce-from  Boundary-condition
//!      .-----^-----. .-----^-----. .--------^--------.
//!       Left  Right   Left   Right   Type  Water   Oil
cycle1_BC:  T    F      T      T     DP    1.00   10.0;
cycle2_BC:  T    F      T      T     DP    10.0   1.00;
cycle3_BC:  T    F      T      T     DP    1.00   10.0;


CALC_BOX: 0.15 1.0; // X bounds of network for computing Sw, Kr & RI

// Note provide Morrow's model number as well, similar to EQUIL_CON_ANG
INIT_CONT_ANG: 1  0  0  -0.2 -3.0   rand;

//             Mdl min max del   eta   RCrl   Mdl2Sep
EQUIL_CON_ANG: 3   45 45   0.2   3.0   rand   25.0;

//FRAC_CON_ANG: 0.7 T 120 150 0.2  3.0  corr 7; //Mixed wettability

//!  viscosity(Pa.s) resistivity(Ohm.m) density(kg/m3)
Water: 0.001             1.2              1000.0
Oil:   0.001            1000.0            1000.0
ClayResistivity:         2.0      ;

WaterOilInterface:       0.03; //! interfacial tension (N/m)

RES_FORMAT:  upscaling; // Excel, Matlab or upscaling (~json) formats
```

Figure 1: Sample input file for the pnflow two-phase flow simulations explaining the important keywords.

1.1 Contact angles

The keyword `INIT_CONT_ANG` is used to assign the contact angles for the primary drainage cycle and `EQUIL_CON_ANG` is used to assign the contact angles for the second (water-injection) cycle, which can be different from the primary drainage due to wettability alteration. The first argument of these keywords indicates the contact angle hysteresis model, which can be 1, 2 or 3, indicating Morrow's contact angle hysteresis models 1 to 3. There is also an option for model 4, which is essentially same as model 3, but expects the advancing contact angles, which is particularly helpful as the advancing contact angles are what will be used in the calculations in the second (water-injection) cycle.

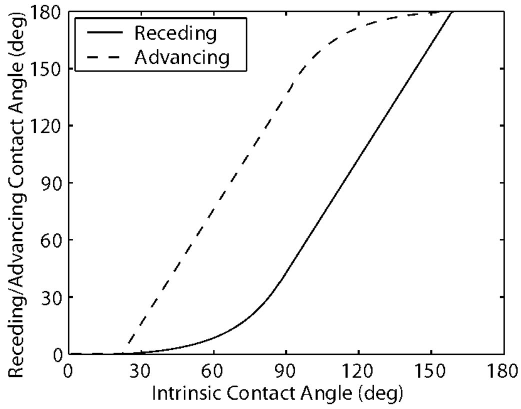

Figure 2: Relation between advancing, receding and intrinsic contact angles in Morrow's (1975) model 3.

The keywords `INIT_CONT_ANG` and `EQUIL_CON_ANG` assign the contact angles using a random or Weibull distribution. The Weibull distribution parameters are the second to sixth arguments of these keywords. The *second* argument is the smallest contact angle, the *third* is the largest contact angle, the *fourth* and *fifth* are the Weibull coefficients, δ and η, and the *sixth* is the contact angle correlation with pore size. The correlation with pore/throat size can be rMin, rMax or rand. rMin implies smaller contact angles are assigned to smaller pores, while the rMax option leads to the opposite, and rand means no correlation to pore size. For Contact angle model 2, the seventh argument can be provided to overwrite the default separation angle (25 degrees)

If the given Weibull distribution coefficient are negative, a uniform distribution between the min and max contact angles will be used instead. Otherwise, the two parameters describe the δ and η of the truncated Weibull distribution equation:

$$\theta = \theta_{min} + (\theta_{max} - \theta_{min}) \left(-\delta \ln \left[x(1 - e^{-1/\delta}) + e^{-1/\delta} \right] \right)^{1/\eta}$$

where x is a random number between 0 and 1.

1.2 Additional input keywords

```
//visualization,
//      Full(T) or Radius Resolution -- Visualize: ------------------ All
//      light (F)  factor  (6-18)    Init  Drainage Imbibition Corners steps
visualize: F        .1      8       T      F        F         F    F;


// Compatibility with Statoil/Oren (1998) network format.
// For simulating flow on networks generated by old network extraction codes,
// e.g. Berea_link1.dat, Berea_link2.dat, Berea_node1.dat and Berea_node2.dat,
// use NETWORK keyword Instead of networkFile above:
NETWORK: Berea; // Base name of the network files in Statoil/Oren format


// PORE_FILL_ALG  blunt2;
// PORE_FILL_WGT   0.0  0.5  1.0  2.0  5.0  10.0;



//           min     Memory Scaling  Solver   Verbose Conductance
//         tolerance     Factor       output   Solver   cut-off
SOLVER_TUNE: 1.0E-30      8            0        F       0.0;

SAT_COVERGENCE:
//  minNumFillings  initStepSize  cutBack  maxIncr  stable disp
      10             0.1        0.8      2.0        F;

DRAIN_SINGLETS:   T;  // T for yes, F for no, singlets are dead-end pores

RAND_SEED:     1002; // seed to C++ (pseudo) random number generator


//  Network modification keywords:

//clayFraction 0.2; // Adding clay, TODO: check
//or
//CLAY_EDIT 0.2; // note these can have different impact, test and see
//or
//AddClay  0.0 0.0 -0.2  -3.0  rand; //For ‘adding‘ clay distribution
```

Keyword `AddClay` expects the Weibull distribution parameters, similar to contact angles. The first argument is the minimum clay fraction (in a throat), the second is the maximum clay fraction, the third and fourth are the Weibull coefficients, and the fifth is the correlation with pore size.

2 Running pnflow

To run the pnflow executable, you should first generate the networks, see the documentation of pnextract executable. Then you can copy the sample input file from the src/doc folder and edit it by setting the networkFile and other keywords, described above, and run the following command in terminal or in Windows command-prompt (cmd).

```
pnflow input_pnflow.dat
```

The above command works if you put pnflow in system PATH. Otherwise, instead of pnflow, you should type the full path othe pnflow executable. In Windows, that is something like:

```
c:\PATH_TO_PNFLOW_EXE\pnflow.exe input_pnflow.dat
```

3 Compiling the code

Open a Linux terminal in the upper-most directory in the source code. and type 'make'. This should compile the Hypre linear equation solver as well as the pnflow code. The command 'make mgw' cross-compiles the code into Windows executable.

For code developers:

A pnflow*.pro file is located in the src/pnm/pnflow directory, which can be imported to qtcreator IDE for project-based compilation of the code. Alternatively you can use geany IDE and work with the provided makefiles directly.

We did not try to compile the code in Windows, although in theory this should be possible using a combination of cmake for compiling Hypre, and nmake or Microsoft Visual-Studio for compiling the pnflow codes.

4 Contact and References

For further information and contact information, please visit:
http://www.imperial.ac.uk/earth-science/research/research-groups/perm/research/pore-scale-modelling

[1] Morrow, N. R. (1975), Effects of surface roughness on contact angle with special reference to petroleum recovery, Journal of Canadian Petroleum Technology, 14, 42-53.

[2] Øren, P. E., S. Bakke, and O. J. Arntzen (1998), Extending predictive capabilities to network models, SPE Journal, 3, 324-336.

[3] Valvatne, P. H. and M J Blunt (2004), Predictive pore-scale modeling of two-phase flow in mixed wet media," Water Resources Research, 40, W07406

[4] Bultreys, T., Q. Lin, Y. Gao, A. Q. Raeini, A. AlRatrout, B. Bijeljic, and M. J. Blunt (2018), Validation of model predictions of pore-scale fluid distributions during two-phase flow. Physical Review E, 97: 053104

For a complete list of publications related to pore network modelling and other pore scale modelling and imaging techniques see: `http://www.imperial.ac.uk/earth-science/research/research-groups/perm/research/pore-scale-modelling/publications/` .

Further explanation on the pnflow keywords not covered here, see the documentation for the original *two phase network modelling code* by Valvatne and Blunt (2004): http://www.imperial.ac.uk/earth-science/research/research-groups/perm/research/pore-scale-modelling/software