This project weights 20% of your final grade!

Consider the following one dimensional fractional flow formulation for incompressible water-saturation transport through incompressible rock, in the absence of gravitational effects:

$$\phi \frac{\partial S_w}{\partial t} + u_t \frac{\partial f_w}{\partial x} = 0, \qquad (0.1)$$

where

$$f_w = \frac{\lambda_w}{\lambda_w + \lambda_n} - \frac{\lambda_w \lambda_n}{\lambda_w + \lambda_n} \frac{1}{u_t} \frac{\partial p_c}{\partial x} \quad and \quad \lambda_\alpha = \frac{K k_r^\alpha}{\mu^\alpha} \ \left(\alpha = \{o, w\}\right)$$

For the remainder consider the spatial domain $x \in [0, \infty)$ and consider (0.1) subject to the following boundary conditions:

$$S^w(0,t) = 1, \qquad S^w(\infty,t) = \begin{cases} S_c^w & if\ p^c = 0 \\ 1.01 S_c^w & if\ p^c > 0 \end{cases},$$

and initial condition of $S^w(x, t=0) = S_c^w$.

Ignore the (non-linear) capillary term for Tasks 1 and 2 below. Use the velocity field from your pressure solver, and assume they do not change during multiphase simulation. If your flow simulator was not correct, assume U = 1 m/day for 1D domains.

Task list:

1. Develop a 1D saturation transport simulator and analyse its results **[5]**:
   a. Explicit in time, first-order upwind method in space **[3]**:
      1.1. For validation, also compare your results with the analytical solution.
      1.2. Show that the upwind approximation is 1^{st} order in space. Also, Discuss the stability of your explicit solutions.
      1.3. Study the effect of fluid properties in the saturation plots.
   b. Implicit in time, first-order upwind method in space **[2]**.

2. Develop a 2D saturation transport simulator. Study the solutions with different fluid properties and space-time grid resolutions. **[5]**

3. **Extra: consider Pc in your equations [2]**
   a. Obtain the above given fractional flow function with Pc.
   b. Report the results.

Two-phase fluid properties:

$$k_{r,\alpha} = k_{r,\alpha,e}(S_{\alpha,e})^{n_\alpha} \qquad S_{\alpha,e} = \frac{S_\alpha - S_{\alpha,r}}{1 - \Sigma_{\beta=1}^{\#phases} S_{\beta,r}} \qquad P_c = \sigma \sqrt{\frac{\phi}{k}} J(S_w) \approx P_d J(S_w)$$

Water end-point relative permeability 0.7, Oil endpoint relative permeability 0.6
Connate or residual water saturation 0.2, Residual oil saturation 0.1
Corey-coefficient n_w=3.5, n_o=2, Capillary entry pressure P_d = 5000 [Pa]

Analytical solution to 1D Buckley-Leverett Equation is given in the next page.

This project weights 20% of your final grade!

Appendix – Matlab code for analytical BL profile $(x_{anl}(S_{anl}), S_{anl})$

```
% ------------------------------------------------------
% Hadi Hajibeygi - All Rights Reserved - 29 April 2015
%   ANALITYCAL SOLUTION OF BUCKLEY LEVERETT EQUATION
% ------------------------------------------------------
clear all; close all;
% parameters
visco_w = 1.0e-3;
visco_o = 10.0e-3;
nw      = 3.5;
no      = 2;
ut      = 1;
swc     = 0.2;
sor     = 0.1;
krwe    = 0.7;
kroe    = 0.8;
porosity= 0.3;

% saturation
nsw     = 101;                        % number of gridpoints
dsw     = (1-sor-swc)/(nsw-1);
sw      = swc:dsw:(1-sor);            % saturation range
% relperms + mobility + fracflow
krw     = krwe*((sw-swc)/(1-swc-sor)).^nw;
kro     = kroe*((1-sw-sor)/(1-swc-sor)).^no;
mob_w   = krw/visco_w;
mob_o   = kro/visco_o;
fw      = mob_w./(mob_w+mob_o);
% fracflow derivative (numerical)
rr      = 2:(nsw-1);
dfw1    = [0 (fw(rr+1)-fw(rr-1))./(sw(rr+1)-sw(rr-1)) 0];
vw      = ut/porosity*dfw1;
% jump velocity
dfw2    = (fw - fw(1))./(sw-sw(1));
[shock_vel, shock_index] = max(dfw2);
shock_sat = sw(shock_index);
rr2     = shock_index:nsw;
% valid saturation range within BL solution
sw2     = [sw(1)   sw(1)   sw(rr2)];
vw2     = vw(rr2);
vw2     = [1.5*vw2(1) vw2(1) vw2];
t       = 10;
x       = vw2*t;

% plot of saturation as function of position
figure(1);
subplot(2,1,2); plot(x,sw2,'.-','linewidth',2);
xlabel('x [m]'); ylabel('S_w [-]');
% plot of fractional flow derivative + jump velocity
subplot(2,1,1); plot(sw,dfw1,sw,dfw2,[shock_sat shock_sat],[0
max(dfw1)],'r:',[swc (1-sor)],[shock_vel shock_vel],'r:');
xlabel('Sw'); ylabel('derivative');
legend('d{f_w}/d{S_w}','\Deltaf_w/\DeltaS_w');
```