Contract No:

This document was prepared in conjunction with work accomplished under Contract No. DE-AC09-08SR22470 with the U.S. Department of Energy (DOE) Office of Environmental Management (EM).

SRNL-STI-2017-00348

June 5, 2017

TO: Dr. Madhukar Rao

FROM: B. T. Butcher

Vadose Zone Flow Convergence Test Suite

Performance Assessment (PA) simulations for engineered disposal systems at the Savannah River Site involve highly contrasting materials and moisture conditions at and near saturation. These conditions cause severe convergence difficulties that typically result in unacceptable convergence or long simulation times or excessive analyst effort. Adequate convergence is usually achieved in a trial-and-error manner by applying under-relaxation to the Saturation or Pressure variable, in a series of ever-decreasing RELAxation values. SRNL would like a more efficient scheme implemented inside PORFLOW to achieve flow convergence in a more reliable and efficient manner. To this end, a suite of test problems that illustrate these convergence problems is provided to facilitate diagnosis and development of an improved convergence strategy. The attached files are being transmitted to you describing the test problem and proposed resolution.

Vadose Zone Flow Convergence Test Suite

Background

Performance Assessment (PA) simulations for engineered disposal systems at the Savannah River Site involve highly contrasting materials and moisture conditions at and near saturation. These conditions cause severe convergence difficulties that typically result in unacceptable convergence or long simulation times or excessive analyst effort. Adequate convergence is usually achieved in a trial-and-error manner by applying under-relaxation to the Saturation or Pressure variable, in a series of ever-decreasing RELAxation values. SRNL would like a more efficient scheme implemented inside PORFLOW to achieve flow convergence in a more reliable and efficient manner. To this end, a suite of test problems that illustrate these convergence problems is provided to facilitate diagnosis and development of an improved convergence strategy.

Materials

Three materials are considered: GRAVEL, SAND and CONCRETE. Moisture characteristic curves are defined by the van Genuchten (1980) functional form

$$S_e = \left[\frac{1}{1 + (\alpha p_c)^{n_{vG}}}\right]^{m_{vG}}$$

$$k_r = S_e^L \left[1 - \left(1 - S_e^{1/m_{vG}}\right)^{m_{vG}}\right]^2$$

where

$$S_e \equiv \frac{S - S_r}{1 - S_r} = \frac{\theta - \theta_r}{\theta_s - \theta_r}$$

and

$$K \equiv k_r K_s$$

Material properties are specified in Table 1. Figures 1 through 3 illustrate the resulting water retention, relative permeability and unsaturated hydraulic conductivity curves.

Table 1 - van Genuchten parameters for GRAVEL, SAND and CONCRETE materials.

Parameter	*GRAVEL*	*SAND*	*CONCRETE*
Saturated hydraulic conductivity K_s	1.50E-01	5.00E-04	3.50E-08
Porosity n / saturated water content θ_s	0.29843	0.38103	0.082
Residual water content θ_r	0.01564	0.1349	0
Residual saturation $S_r = \theta_r/\theta_s$	0.0524076	0.35404	0
van Genuchten α parameter (cm^{-1})	1.43E-01	2.95E-02	2.0856E-06
van Genuchten n_{vG} parameter	1.45746	1.40995	1.9433
van Genuchten m_{vG} parameter	0.313875	0.290755	0.485411414

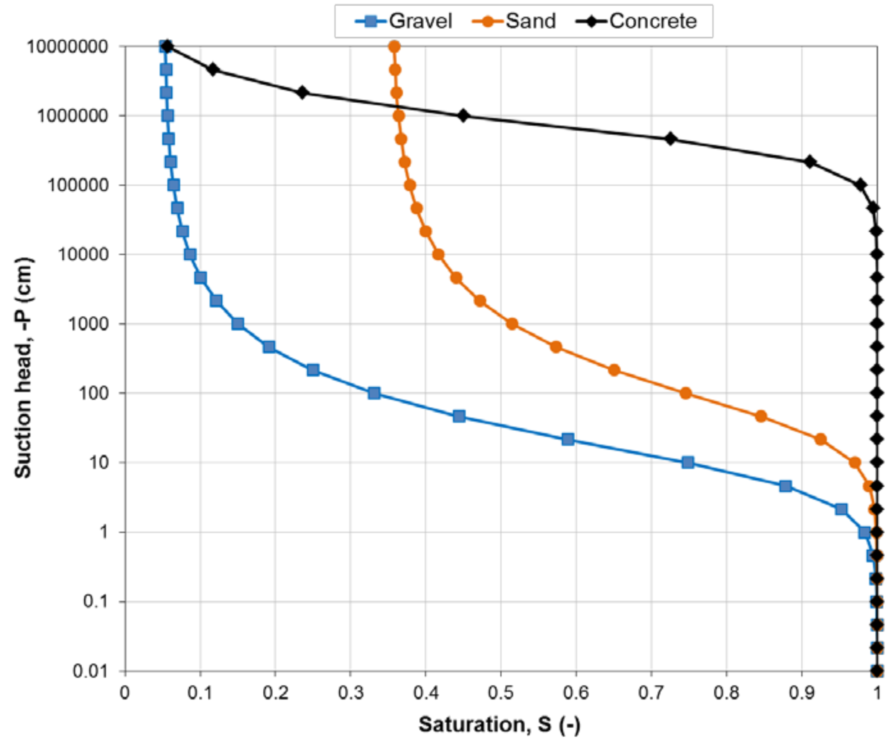


Figure 1 - Water retention curves.

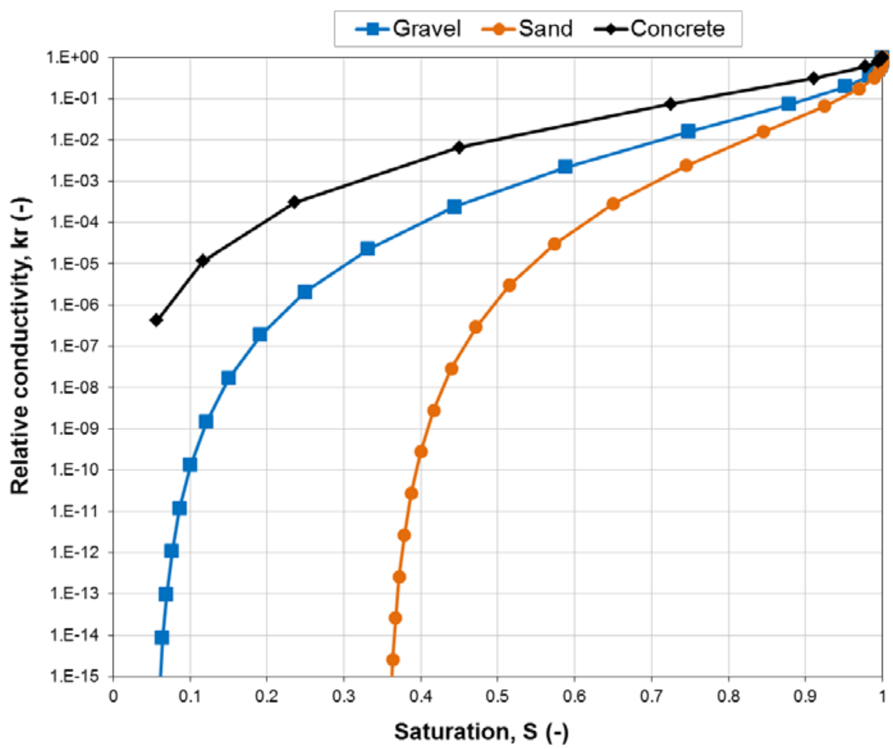


Figure 2 - Relative permeability curves.

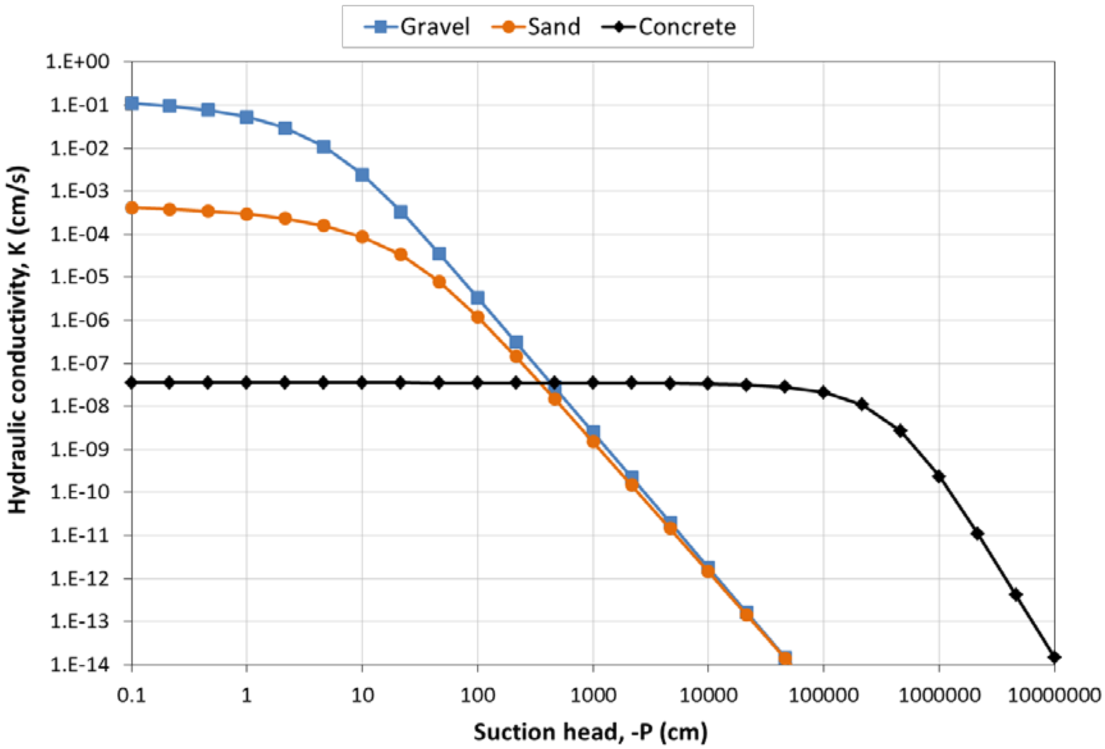


Figure 3 - Unsaturated hydraulic conductivity curves.

Grid zones

Figure 4 illustrates the modeling domain and grid zones. The nominal dimensions of the domain are -30 < X < +30 meters and 0 < Y < +40 meters. The domain is divided into three grid zones: ZONE[1,2,3]. ZONE1 is the default zone representing the area outside of ZONE2 and ZONE3, usually assigned SAND properties. The ZONE2 region, near the center of the domain, generally represents a waste disposal unit / tank and is assigned CONCRETE properties. ZONE3 is a narrow (4 cm) vertical path through the center of ZONE2, generally representing a fracture (fast-flow path, crack). However, each grid zone may be assigned any material from the above palette: GRAVEL, SAND or CONCRETE.

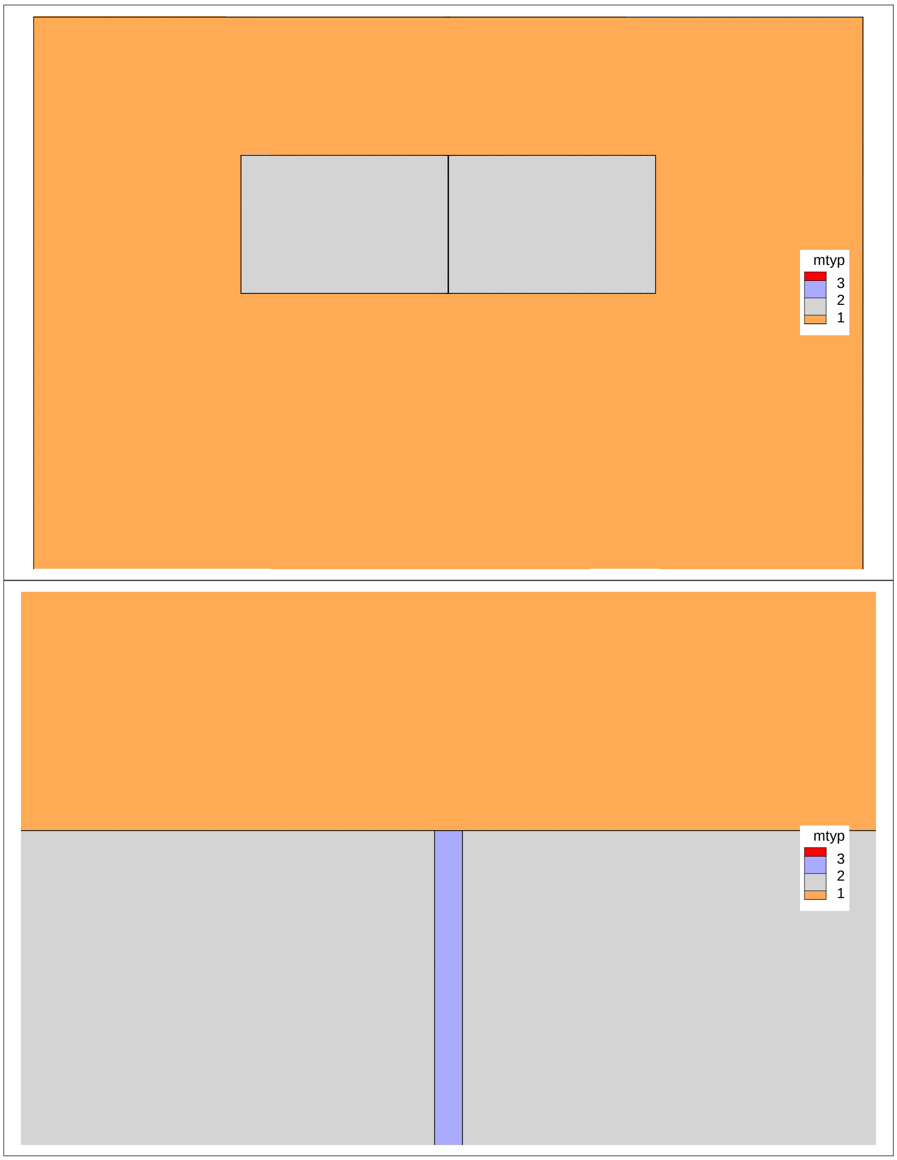


Figure 4 - Grid zones that are assigned material properties (top image = entire domain; bottom image = entrance to ZONE3).

Computational grid

Figure 5 illustrates the two-dimensional computational grid, which is 85 by 62 PORFLOW NODEs.

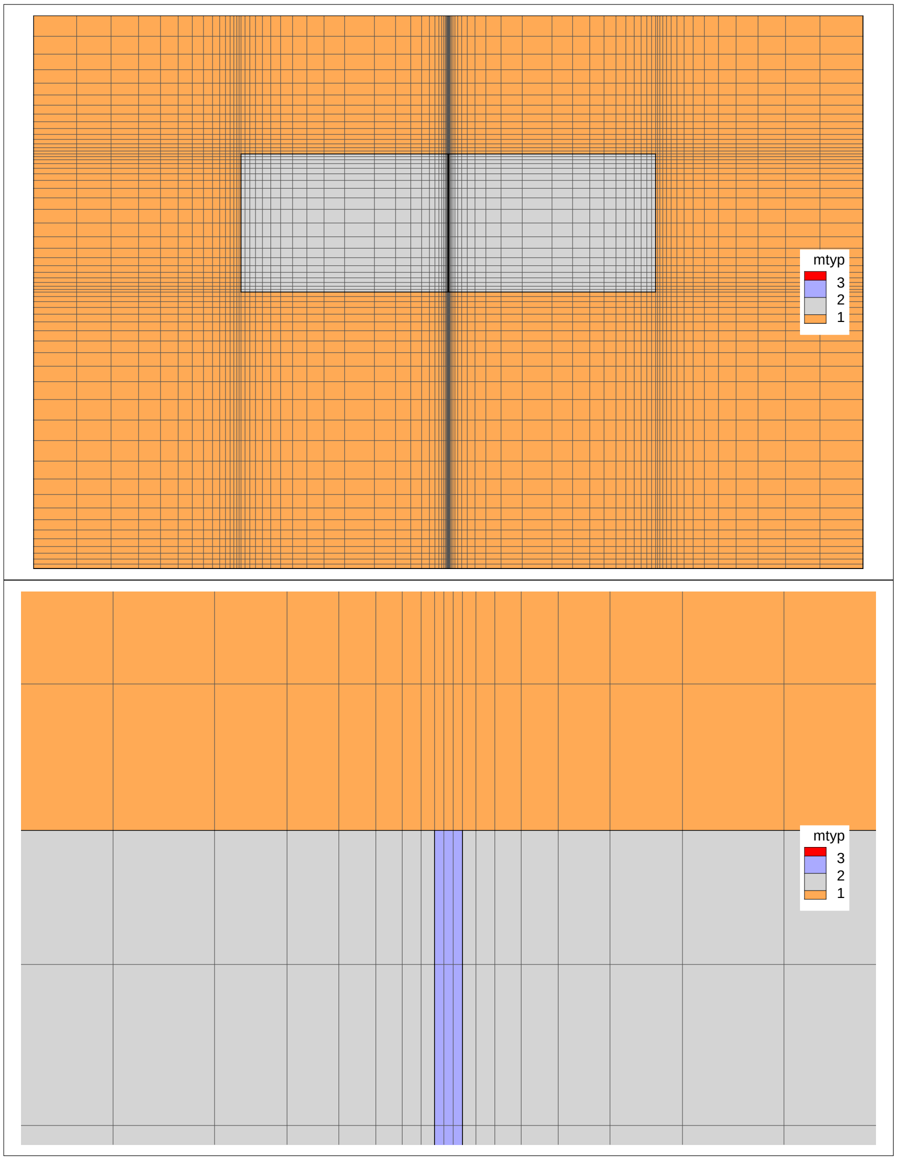


Figure 5 - Computational grid (top image = entire domain; bottom image = entrance to ZONE3).

Numerical convergence schemes

Four schemes for achieving numerical convergence were tested as shown in Table 2. “PAscheme” is a strategy that has proved successful when van Genuchten water retention and relative permeability curves are specified through tabular input (as opposed to defining van Genuchten functions through parameter input). “PAalt” is an alternative to “PAscheme” that allows more iterations before relaxation is applied. “Prelax” and “Srelax” impose Pressure and Saturation relaxation, respectively, in a uniform manner with steadily decreasing relaxation.

Table 2 - Numerical convergence schemes.

PAscheme	*PAalt*	*Prelax*	*Srelax*
// PA scheme !!Migrate to solution neighborhood CONVergence for P 1.e-6, 10 iterations max SOLVe STEAdy 5 RELAx S 0.7 SOLVe STEAdy 5 RELAx S 0.3 SOLVe STEAdy 15 RELAx S 0.1 SOLVe STEAdy 45 !!Mixed purpose CONVergence for P 1.e-6, 30 iterations max RELAx S 0.03 SOLVe STEAdy 50 RELAx S 0.01 SOLVe STEAdy 50 !!Suppress noise / sharpen mass balance CONVergence for P 1.e-6, 100 iterations max RELAx S 0.003 SOLVe STEAdy 20 RELAx S 0.001 SOLVe STEAdy 20 RELAx S 0.0003 SOLVe STEAdy 20 RELAx S 0.0001 SOLVe STEAdy 20	// PA scheme !!Migrate to solution neighborhood CONVergence for P 1.e-6, 10 iterations max SOLVe STEAdy 100 RELAx S 0.7 SOLVe STEAdy 5 RELAx S 0.3 SOLVe STEAdy 15 RELAx S 0.1 SOLVe STEAdy 45 !!Mixed purpose CONVergence for P 1.e-6, 30 iterations max RELAx S 0.03 SOLVe STEAdy 50 RELAx S 0.01 SOLVe STEAdy 50 !!Suppress noise / sharpen mass balance CONVergence for P 1.e-6, 100 iterations max RELAx S 0.003 SOLVe STEAdy 20 RELAx S 0.001 SOLVe STEAdy 20 RELAx S 0.0003 SOLVe STEAdy 20 RELAx S 0.0001 SOLVe STEAdy 20	// Pressure relaxation CONVergence for P 1.e-6, 100 iterations max SOLVe STEAdy 100 RELAx P 0.5 SOLVe STEAdy 100 RELAx P 0.2 SOLVe STEAdy 100 RELAx P 0.1 SOLVe STEAdy 100 RELAx P 0.05 SOLVe STEAdy 100 RELAx P 0.02 SOLVe STEAdy 100 RELAx P 0.01 SOLVe STEAdy 100 RELAx P 0.005 SOLVe STEAdy 100 RELAx P 0.002 SOLVe STEAdy 100 RELAx P 0.001 SOLVe STEAdy 100 RELAx P 0.0005 SOLVe STEAdy 100 RELAx P 0.0002 SOLVe STEAdy 100 RELAx P 0.0001 SOLVe STEAdy 100	// Saturation relaxation CONVergence for P 1.e-6, 100 iterations max SOLVe STEAdy 100 RELAx S 0.5 SOLVe STEAdy 100 RELAx S 0.2 SOLVe STEAdy 100 RELAx S 0.1 SOLVe STEAdy 100 RELAx S 0.05 SOLVe STEAdy 100 RELAx S 0.02 SOLVe STEAdy 100 RELAx S 0.01 SOLVe STEAdy 100 RELAx S 0.005 SOLVe STEAdy 100 RELAx S 0.002 SOLVe STEAdy 100 RELAx S 0.001 SOLVe STEAdy 100 RELAx S 0.0005 SOLVe STEAdy 100 RELAx S 0.0002 SOLVe STEAdy 100 RELAx S 0.0001 SOLVe STEAdy 100

Simulation Cases

Table 3 summarizes the simulation cases considered. Unless otherwise noted in the Comments column, PORFLOW ran to completion.

Table 3 - Simulation cases.

Case name	ZONE1 Material	ZONE2 Material	ZONE3 Material	van Genuchten curve specification	Convergence scheme	Comments
Combinations of SAND and GRAVEL						
PAscheme_GGG_table	GRAVEL	GRAVEL	GRAVEL	table	PAscheme	
PAscheme_SGG_table	SAND	GRAVEL	GRAVEL	table	PAscheme	
PAscheme_SSG_table	SAND	SAND	GRAVEL	table	PAscheme	
PAscheme_SSS_table	SAND	SAND	SAND	table	PAscheme	
Combinations of SAND and GRAVEL + PAalt convergence scheme + van Genuchten table						
PAalt_GGG_table	GRAVEL	GRAVEL	GRAVEL	table	PAalt	
PAalt_SGG_table	SAND	GRAVEL	GRAVEL	table	PAalt	
PAalt_SSG_table	SAND	SAND	GRAVEL	table	PAalt	
PAalt_SSS_table	SAND	SAND	SAND	table	PAalt	
Concrete monolith + various convergence schemes + van Genuchten table						
PAscheme_woFracture_table	SAND	CONCRETE	CONCRETE	table	PAscheme	
PAalt_woFracture_table	SAND	CONCRETE	CONCRETE	table	PAalt	
Prelax_woFracture_table	SAND	CONCRETE	CONCRETE	table	Prelax	
Srelax_woFracture_table	SAND	CONCRETE	CONCRETE	table	Srelax	
Concrete monolith + various convergence schemes + van Genuchten function						
PAscheme_woFracture_vG	SAND	CONCRETE	CONCRETE	function	PAscheme	
PAalt_woFracture_vG	SAND	CONCRETE	CONCRETE	function	PAalt	
Prelax_woFracture_vG	SAND	CONCRETE	CONCRETE	function	Prelax	
Srelax_woFracture_vG	SAND	CONCRETE	CONCRETE	function	Srelax	
Concrete with fracture + various convergence schemes + van Genuchten table						
PAscheme_wFracture_table	SAND	CONCRETE	GRAVEL	table	PAscheme	fatal error†
PAalt_wFracture_table	SAND	CONCRETE	GRAVEL	table	PAalt	fatal error†
Prelax_wFracture_table	SAND	CONCRETE	GRAVEL	table	Prelax	fatal error†
Srelax_wFracture_table	SAND	CONCRETE	GRAVEL	table	Srelax	fatal error†
Concrete with fracture + various convergence schemes + van Genuchten function						
PAscheme_wFracture_vG	SAND	CONCRETE	GRAVEL	function	PAscheme	
PAalt_wFracture_vG	SAND	CONCRETE	GRAVEL	function	PAalt	
Prelax_wFracture_vG	SAND	CONCRETE	GRAVEL	function	Prelax	
Srelax_wFracture_vG	SAND	CONCRETE	GRAVEL	function	Srelax	

† "Fatal Error in the NSPCG linear solver. Error code -12 received from NSPCG. Zero pivot encountered in factorization."

Diagnostic plots

Diagnostic plots are provided for each simulation case using the …`/Tools/PlotFlow2D_merge_vG` program to generate a Tecplot data file. An example is shown in Figure 6.

The upper left image is Pressure head. The upper right image is Saturation. The lower left image is a relative mass-balance plot of the `fcnet` variable defined with `isNormalized=.true.` and `isSignCoded=.true.` as shown in this code snippet:

```
if (isNormalized) then
  fcnorm1 = sqrt(0.25*(fcxm(i,j)**2 + fcym(i,j)**2 + fcxp(i,j)**2 + fcyp(i,j)**2))
  fcnorm2 = rechflow * (area/domain)
  fcnet(i,j) = fcdiff/max(fcnorm1,fcnorm2)    !fcnet(i,j) = fcdiff/fcnorm1

  if (isSignCoded) then
    if (fcnorm1 .gt. fcnorm2) then
      fcnet(i,j) = +abs(fcnet(i,j))
    else
      fcnet(i,j) = -abs(fcnet(i,j))
    end if
  end if
end if
```

The lower right image is `dels`, the difference between saturation computed from the van Genuchten function using PORFLOW-reported Pressure, and the Saturation variable reported by PORFLOW. When relaxation is applied Pressure and Saturation do not generally agree with the specified van Genuchten water retention function.

A simulation is considered to have achieved convergence is the relative mass balance (`fcnet`) and saturation delta (`dels`) are small everywhere in the domain (green color throughout the lower two images).

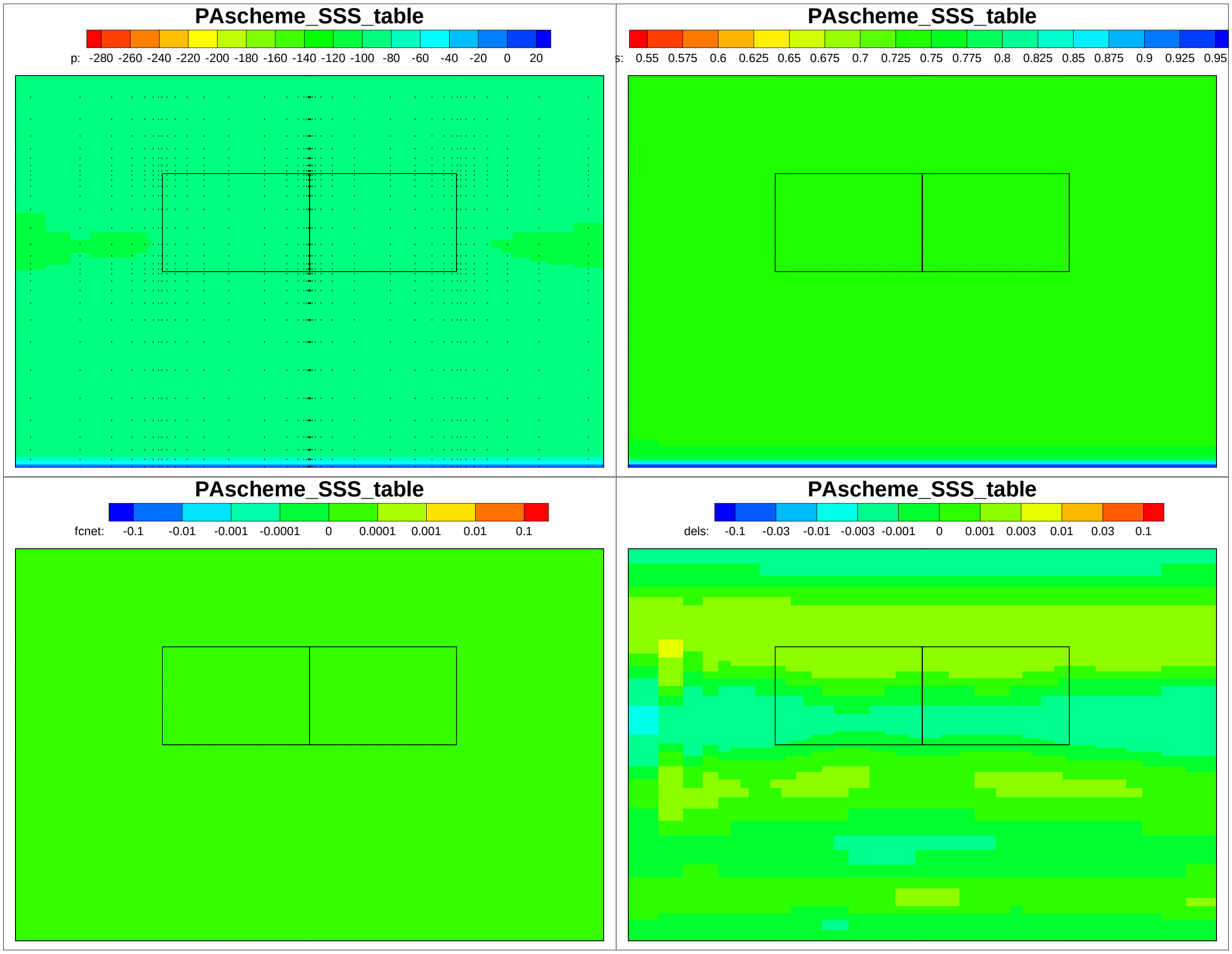


Figure 6 - Example diagnostic plot for "PAscheme_SSS_table" case.

Results and discussion

All simulations were run using PORFLOW version 6.30.2 and UPWInding and the direct solver option (`PROPerty P UPWI; MATRix for P NSPC SYMM CHOL CONJ`). The initial condition was Pressure = 0 (fully saturated).

Figures 7 through 30 summarize the simulation and diagnostic results for the cases defined in Table 3.

Some simulations were relatively successful (e.g. Figures 15, 16, 18 and 29). Other exhibited a fatal error (Table 3, Figures 23- 26). Many exhibit mass balance and/or saturation delta problems at simulation completion.

Adequate convergence could likely be achieved for most if not all cases through a careful manual effort to adjust the relaxation scheme. However, this approach is costly (in labor charges) and not practical if thousands of simulation cases are required, which is typically the case.

We would like a more reliable scheme for achieving adequate numerical convergence.

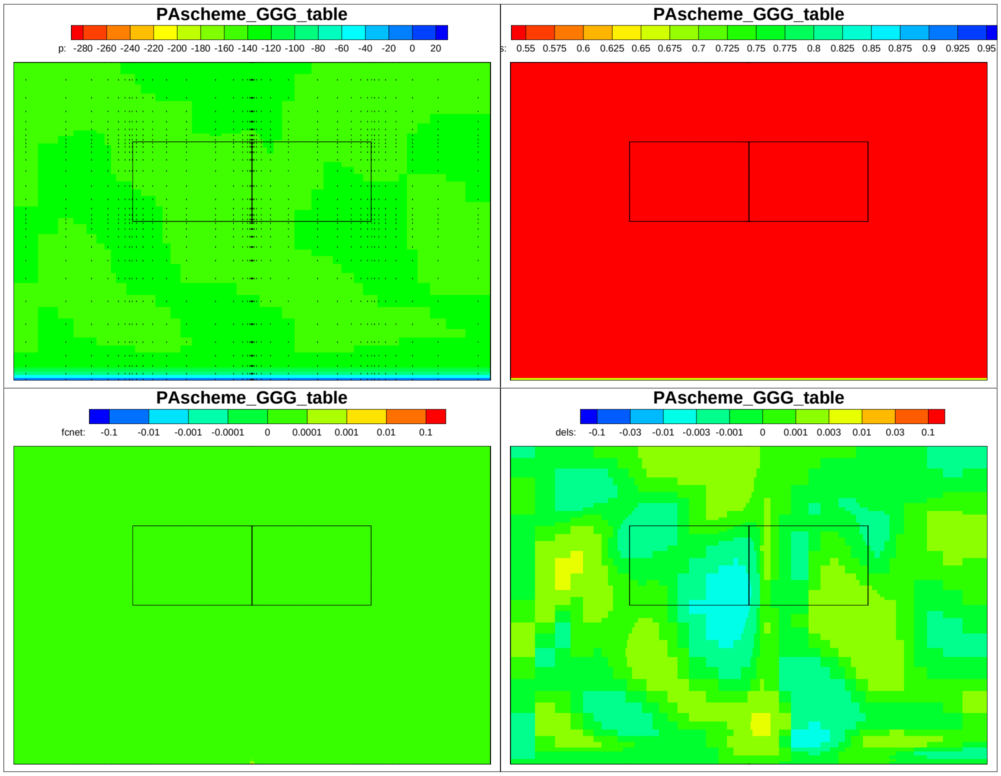


Figure 7 - Diagnostic plot for "PAscheme_GGG_table" case.

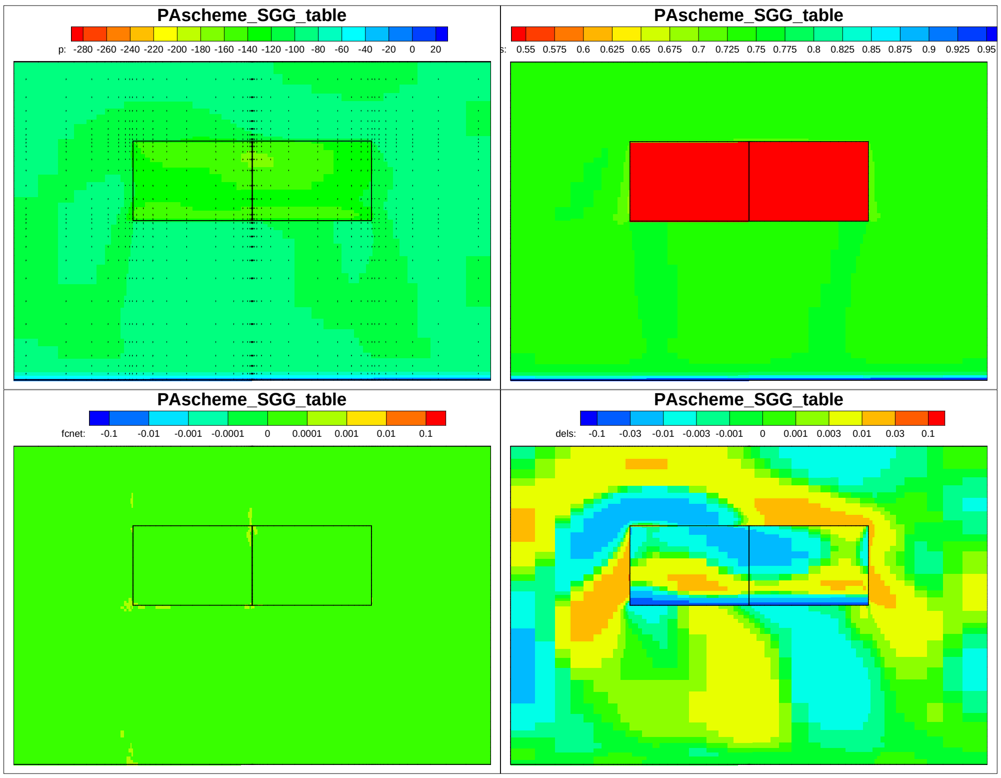


Figure 8 - Diagnostic plot for "PAscheme_SGG_table" case.

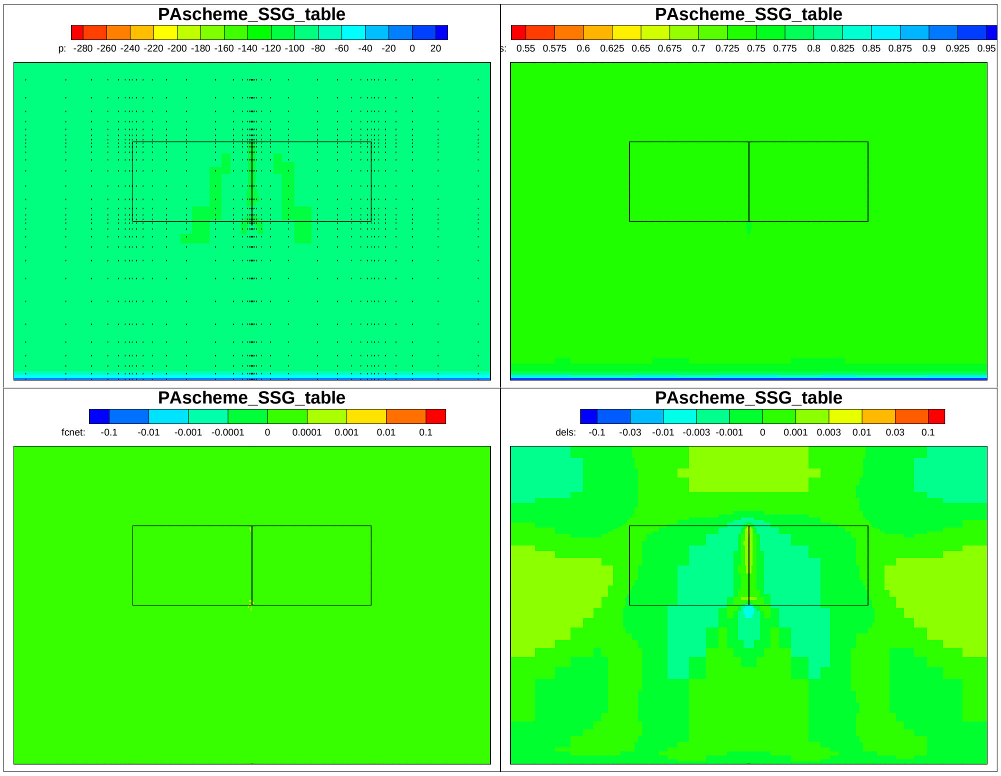


Figure 9 - Diagnostic plot for “PAscheme_SSG_table” case.

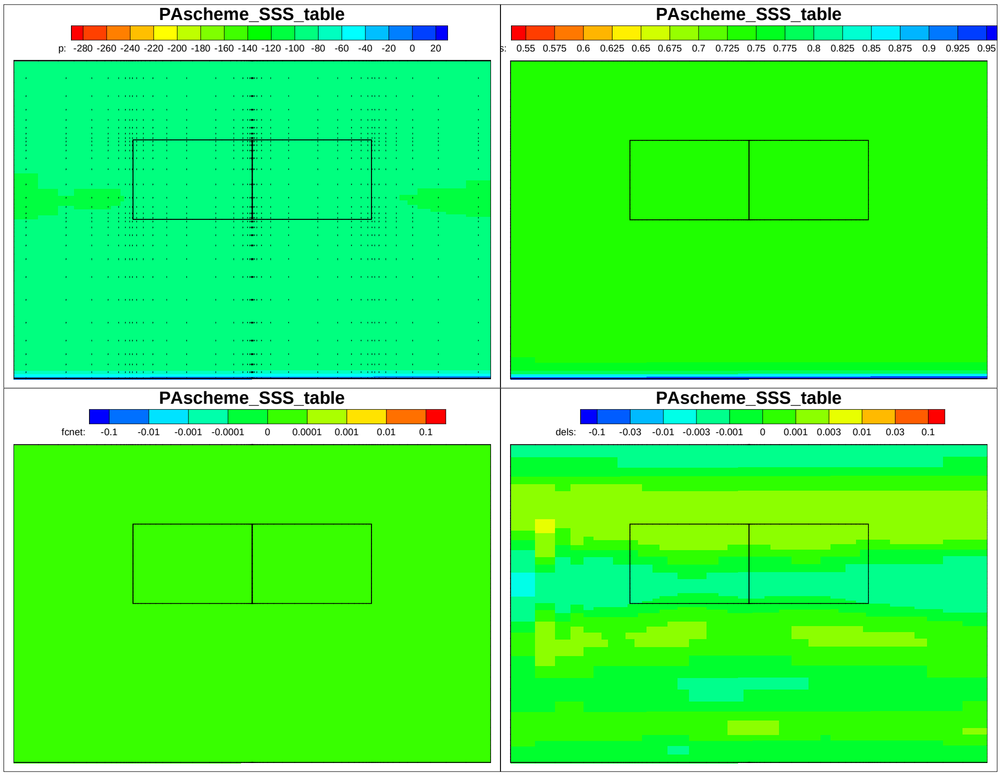


Figure 10 - Diagnostic plot for “PAscheme_SSS_table” case.

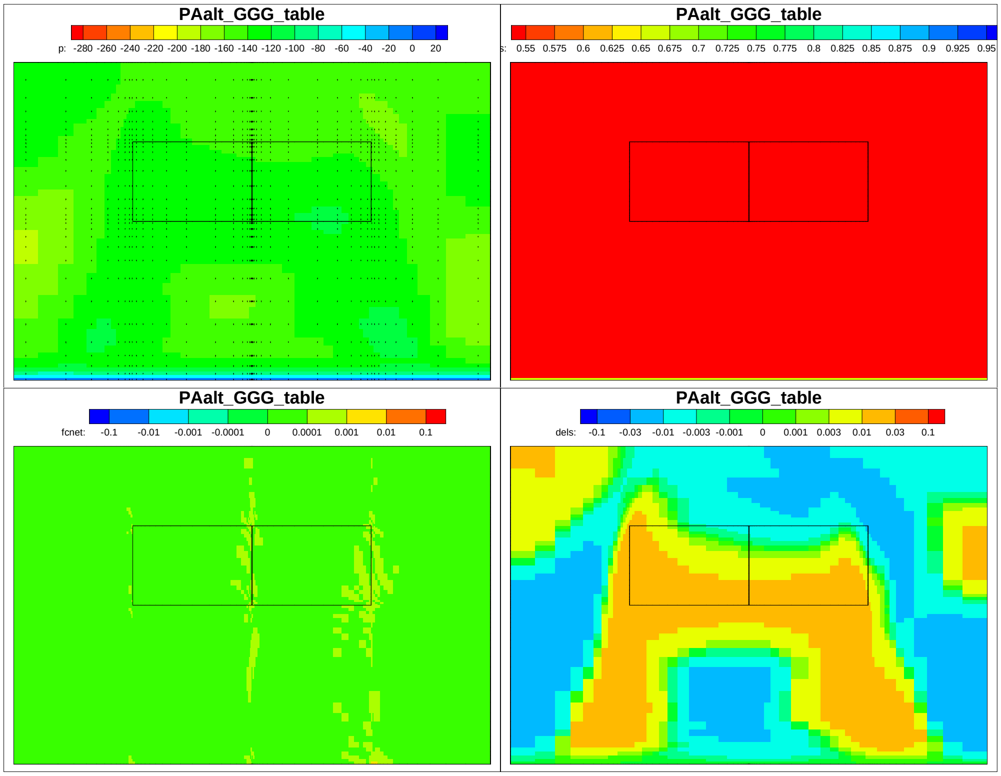


Figure 11 - Diagnostic plot for "PAalt_GGG_table" case.

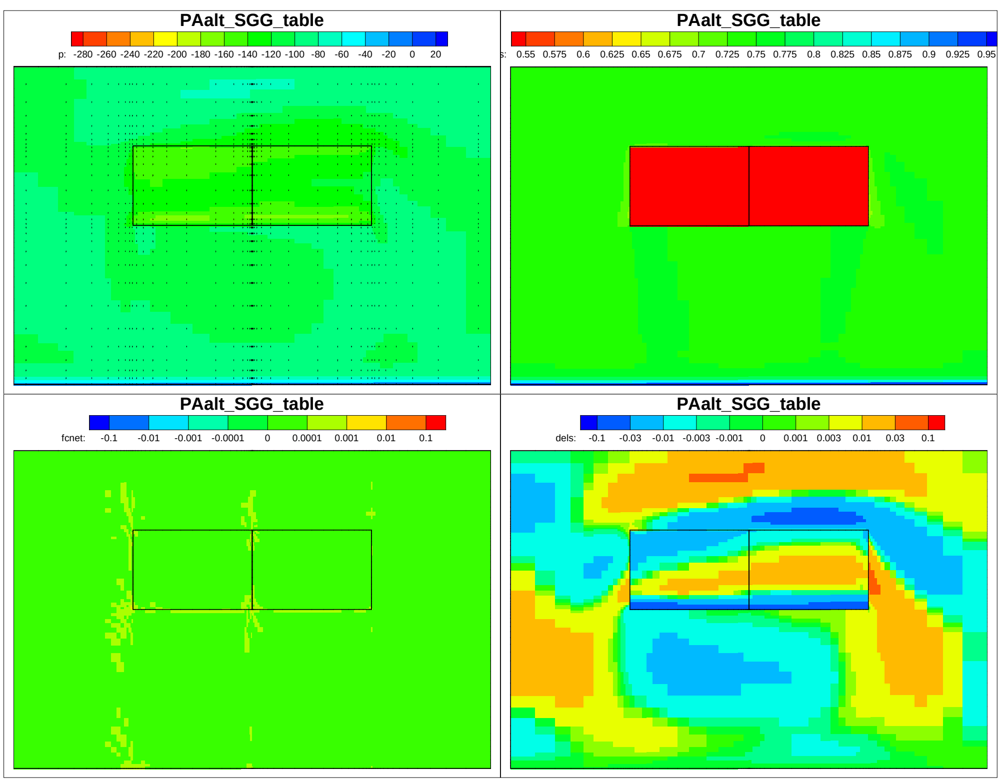


Figure 12 - Diagnostic plot for "PAalt_SGG_table" case.

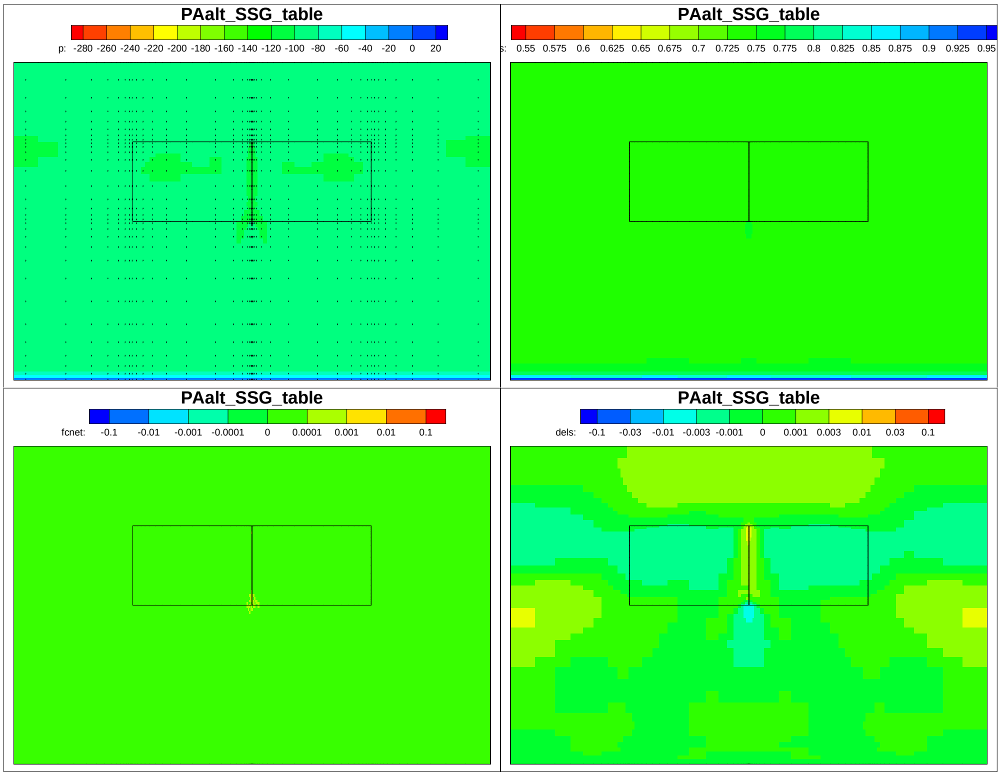


Figure 13 - Diagnostic plot for "PAalt_SSG_table" case.

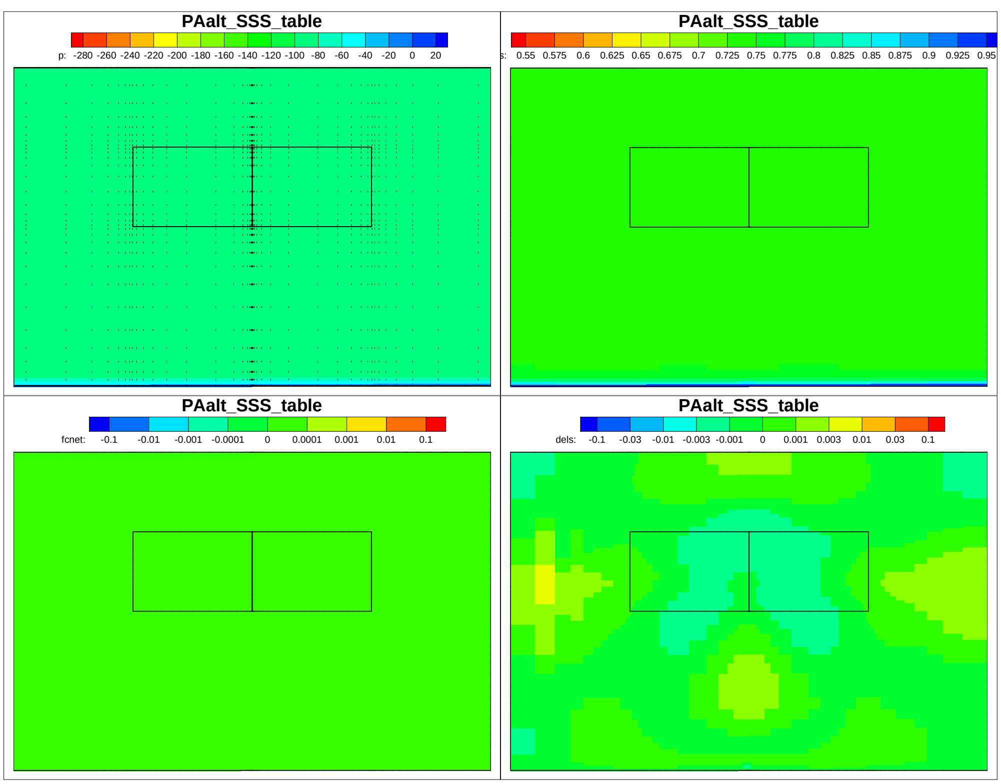


Figure 14 - Diagnostic plot for "PAalt_SSS_table" case.

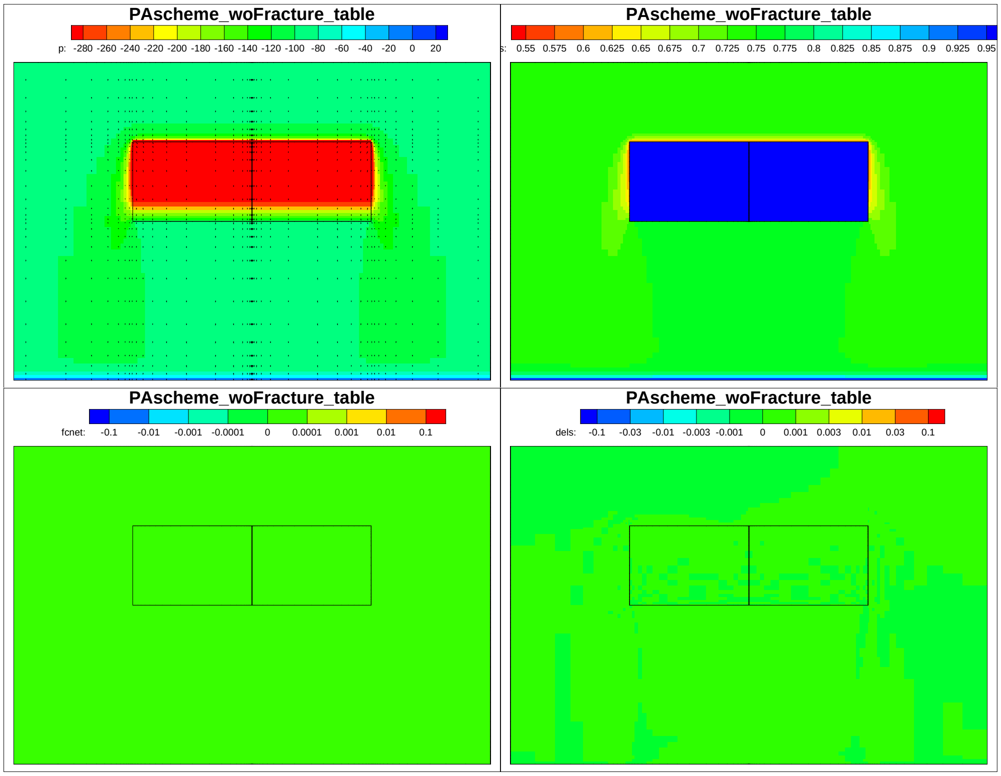


Figure 15 - Diagnostic plot for “PAscheme_woFracture_table” case.

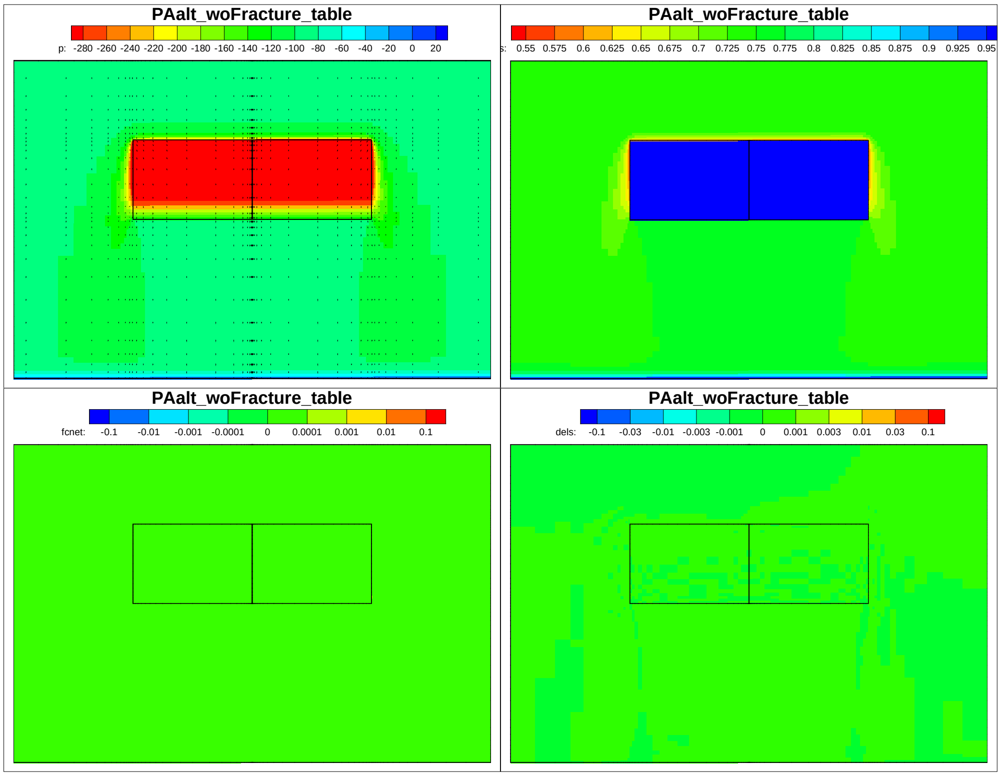


Figure 16 - Diagnostic plot for “PAalt_woFracture_table” case.

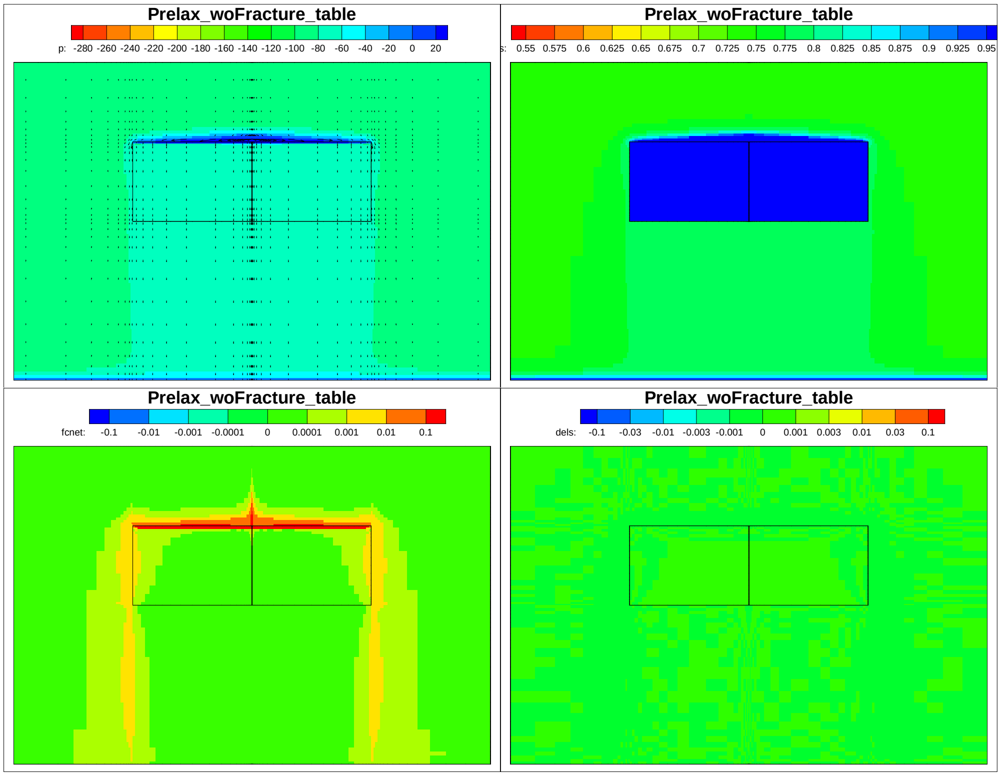


Figure 17 - Diagnostic plot for “Prelax_woFracture_table” case.

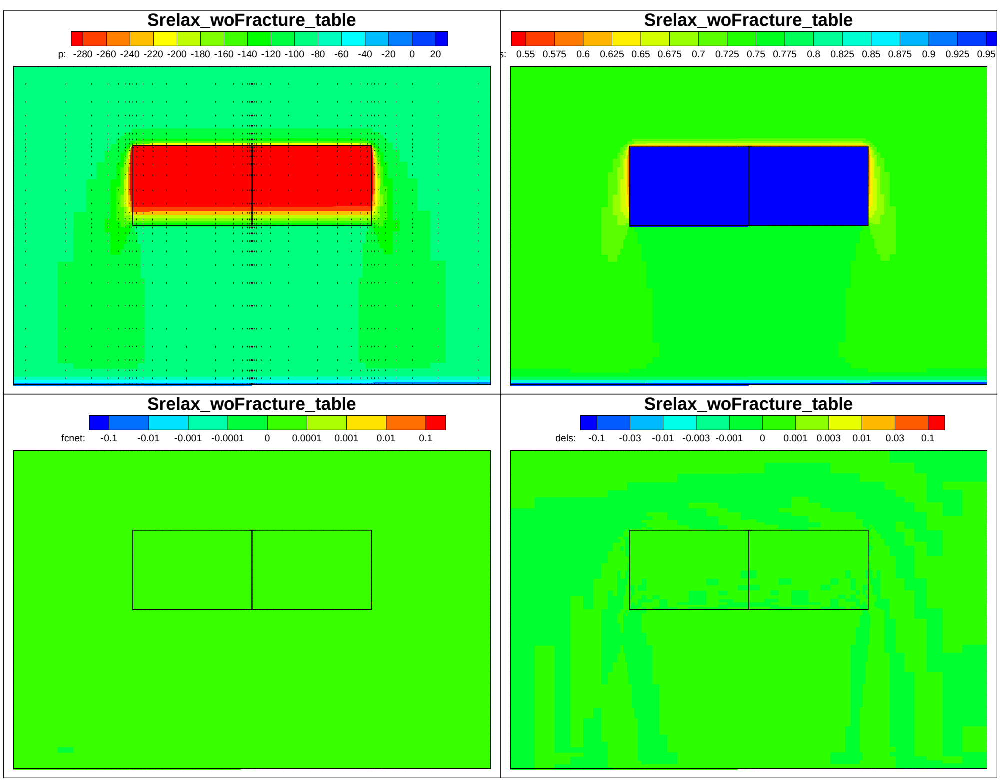


Figure 18 - Diagnostic plot for “Srelax_woFracture_table” case.

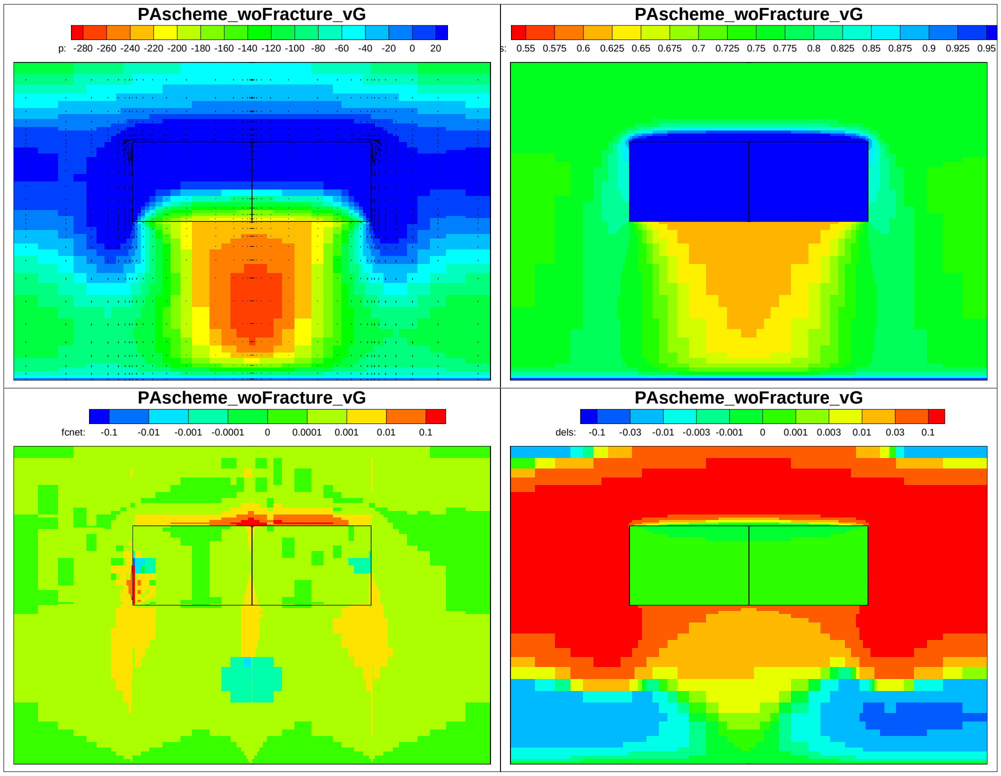


Figure 19 - Diagnostic plot for “PAscheme_woFracture_vG” case.

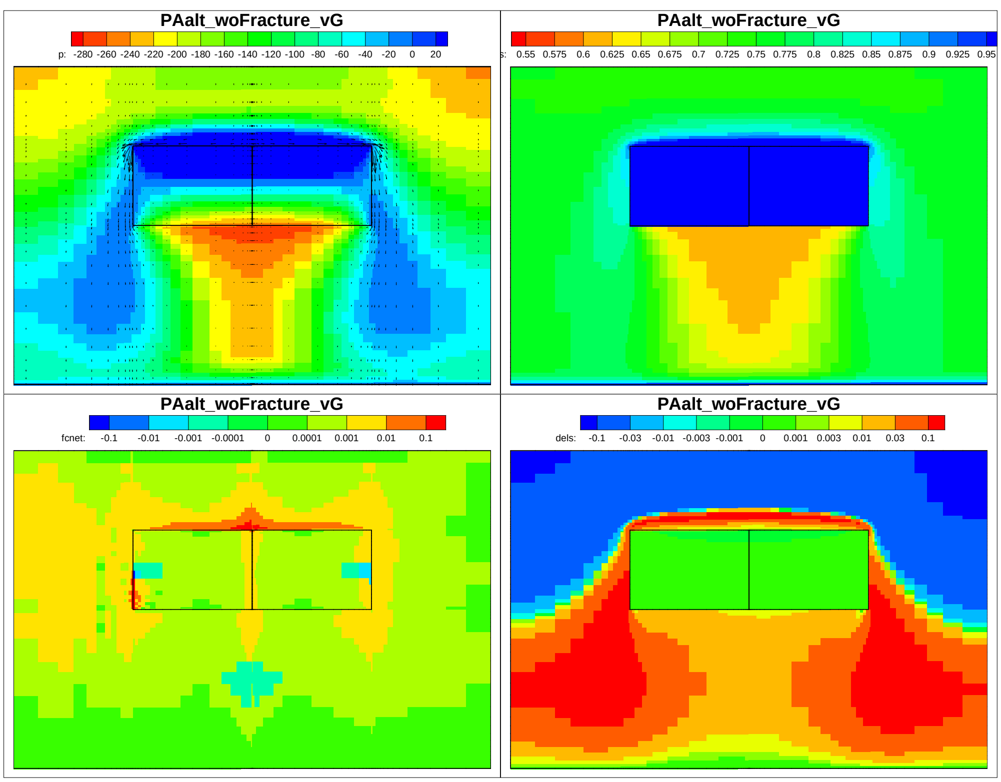


Figure 20 - Diagnostic plot for “PAalt_woFracture_vG” case.

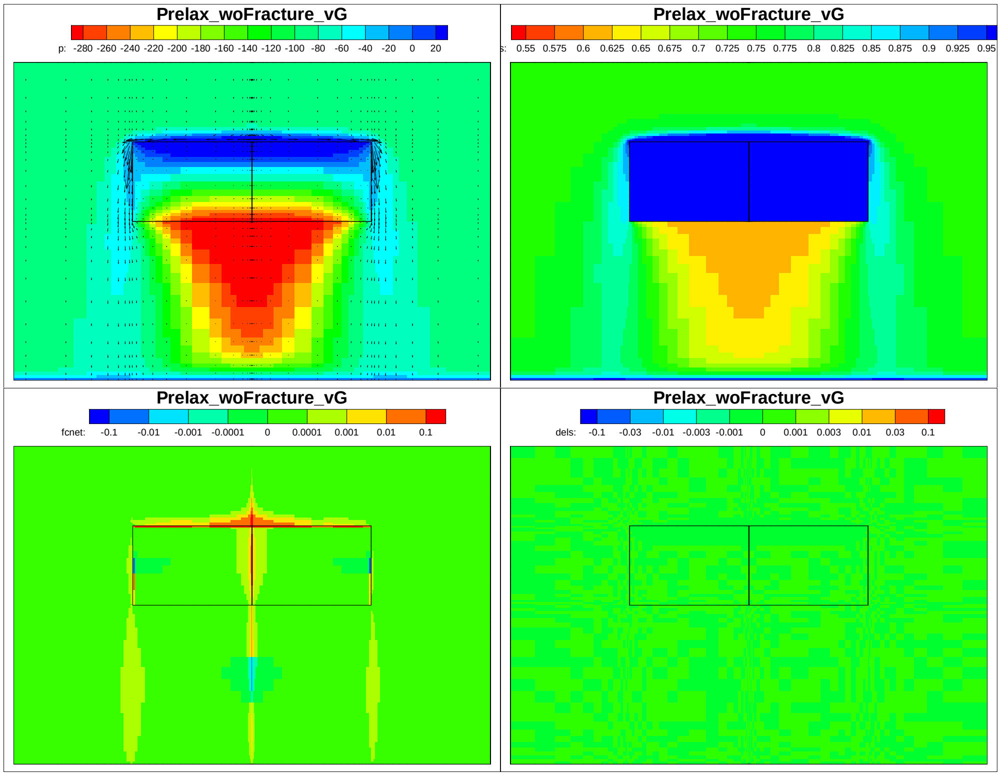


Figure 21 - Diagnostic plot for "Prelax_woFracture_vG" case.

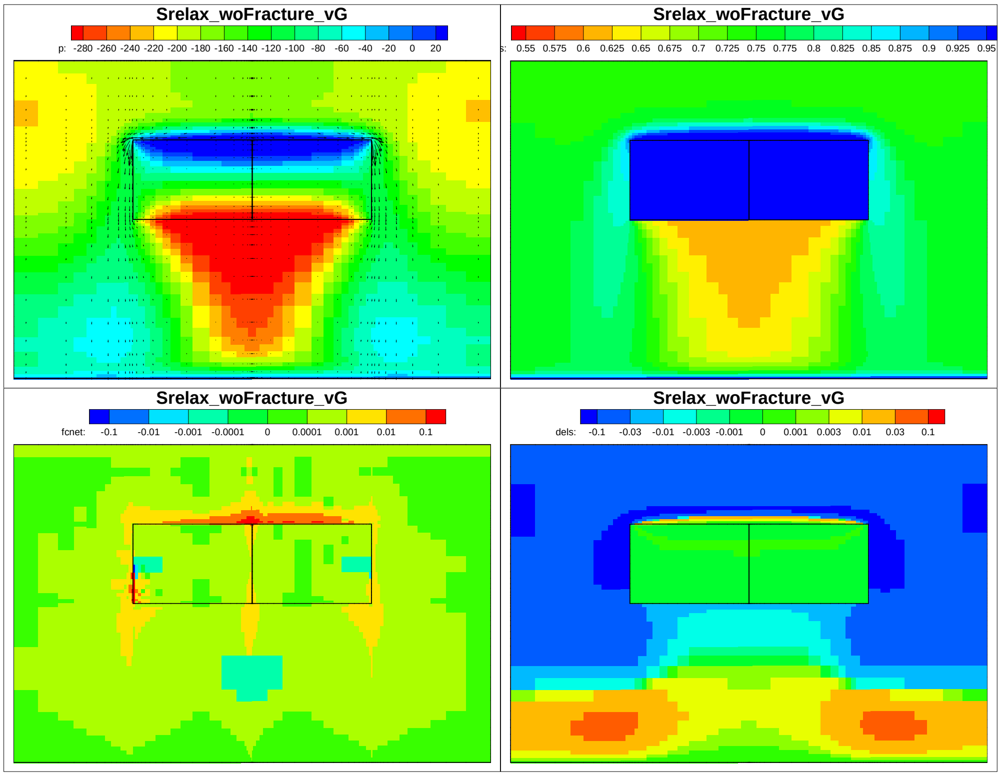


Figure 22 - Diagnostic plot for "Srelax_woFracture_vG" case.

(PORFLOW fatal error)

Figure 23 - Diagnostic plot for “PAscheme_wFracture_table” case.

(PORFLOW fatal error)

Figure 24 - Diagnostic plot for “PAalt_wFracture_table” case.

(PORFLOW fatal error)

Figure 25 - Diagnostic plot for “Prelax_wFracture_table” case.

(PORFLOW fatal error)

Figure 26 - Diagnostic plot for “Srelax_wFracture_table” case.

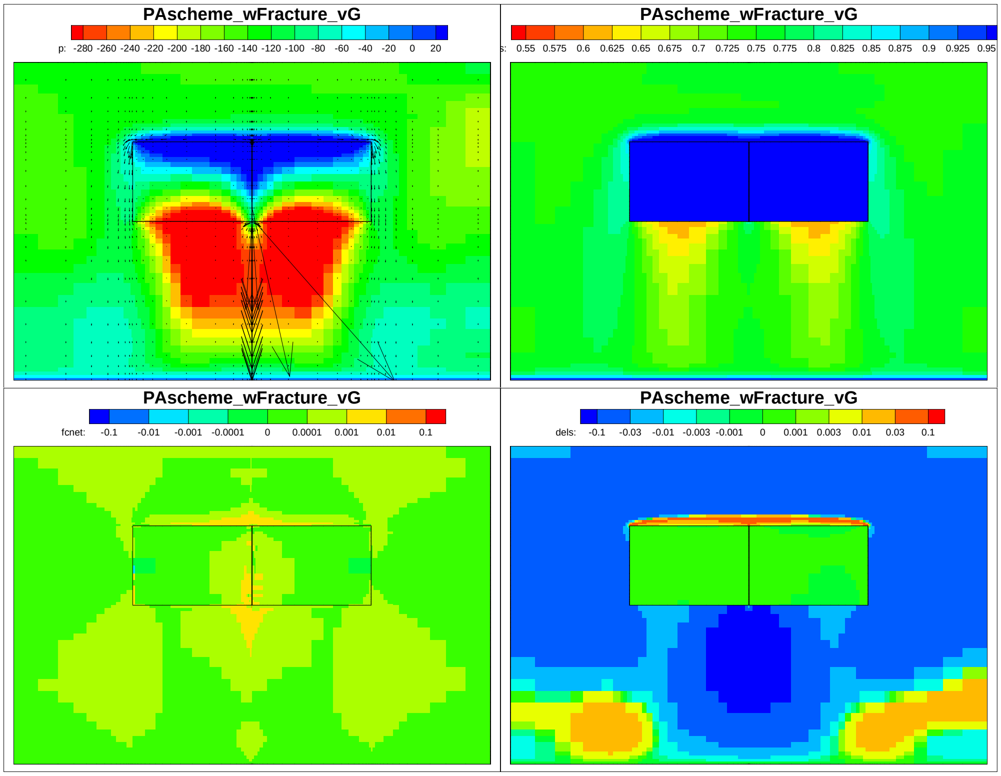


Figure 27 - Diagnostic plot for "PAscheme_wFracture_vG" case.

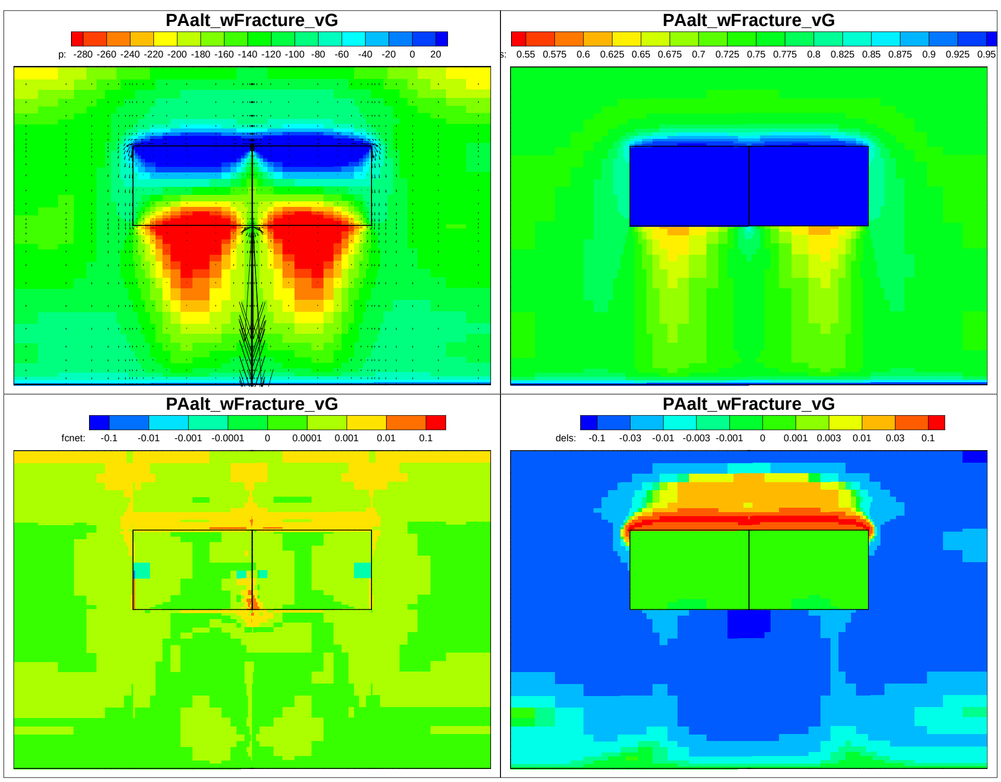


Figure 28 - Diagnostic plot for "PAalt_wFracture_vG" case.

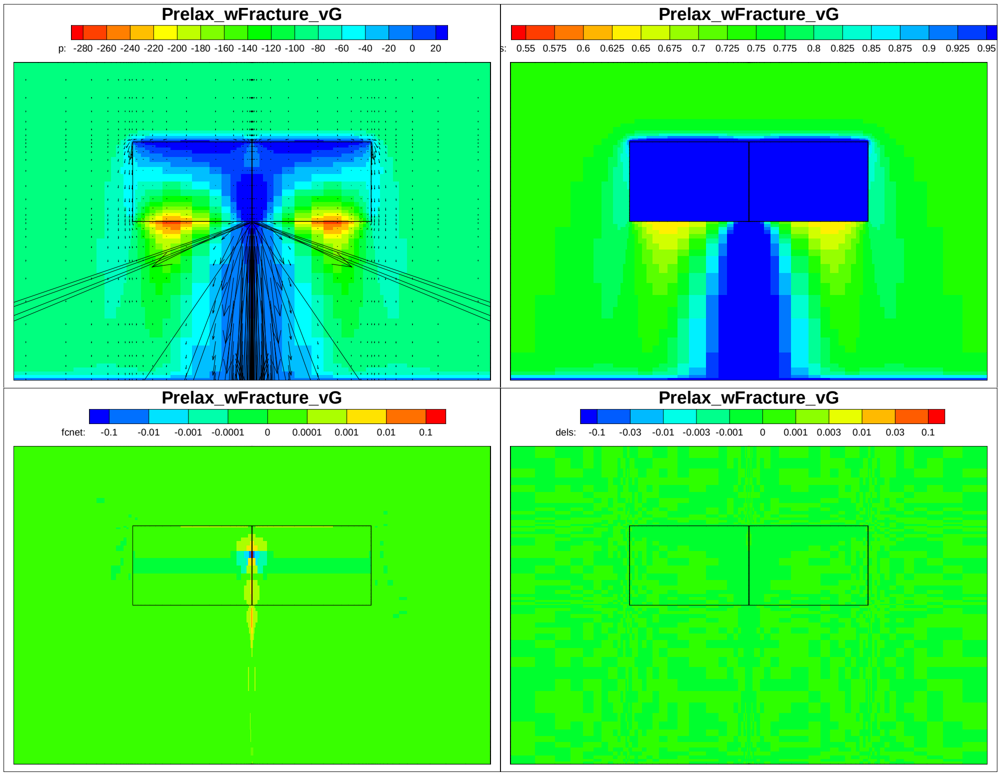


Figure 29 - Diagnostic plot for "Prelax_wFracture_vG" case.

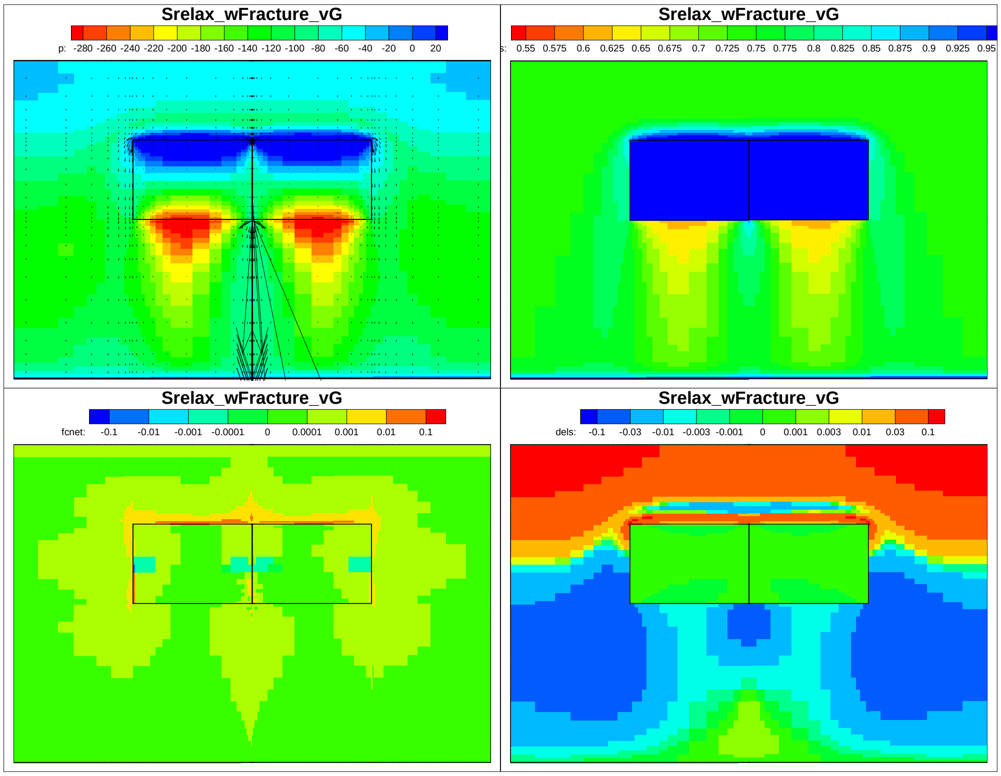


Figure 30 - Diagnostic plot for "Srelax_wFracture_vG" case.

Update to flow convergence test using new convergence approach

<u>"ACRi" scheme to test from Dr. Rao (6/17/2016)</u>

[1] replace NSPCG

```
MATRix P NSPCg SYMMetry CHOLesky CONJugate gradient
```

with my implementation of the same, i.e.,

```
MATRix P CHOLesky CONJugate gradient
```

[2] replace the convergence command:

```
CONVergence P    1.e-6, 100 iterations max
```

with the following commands,

```
MATRix P ITERAtions 200
CONVergence P    1.e-6, 1 iterations max
```

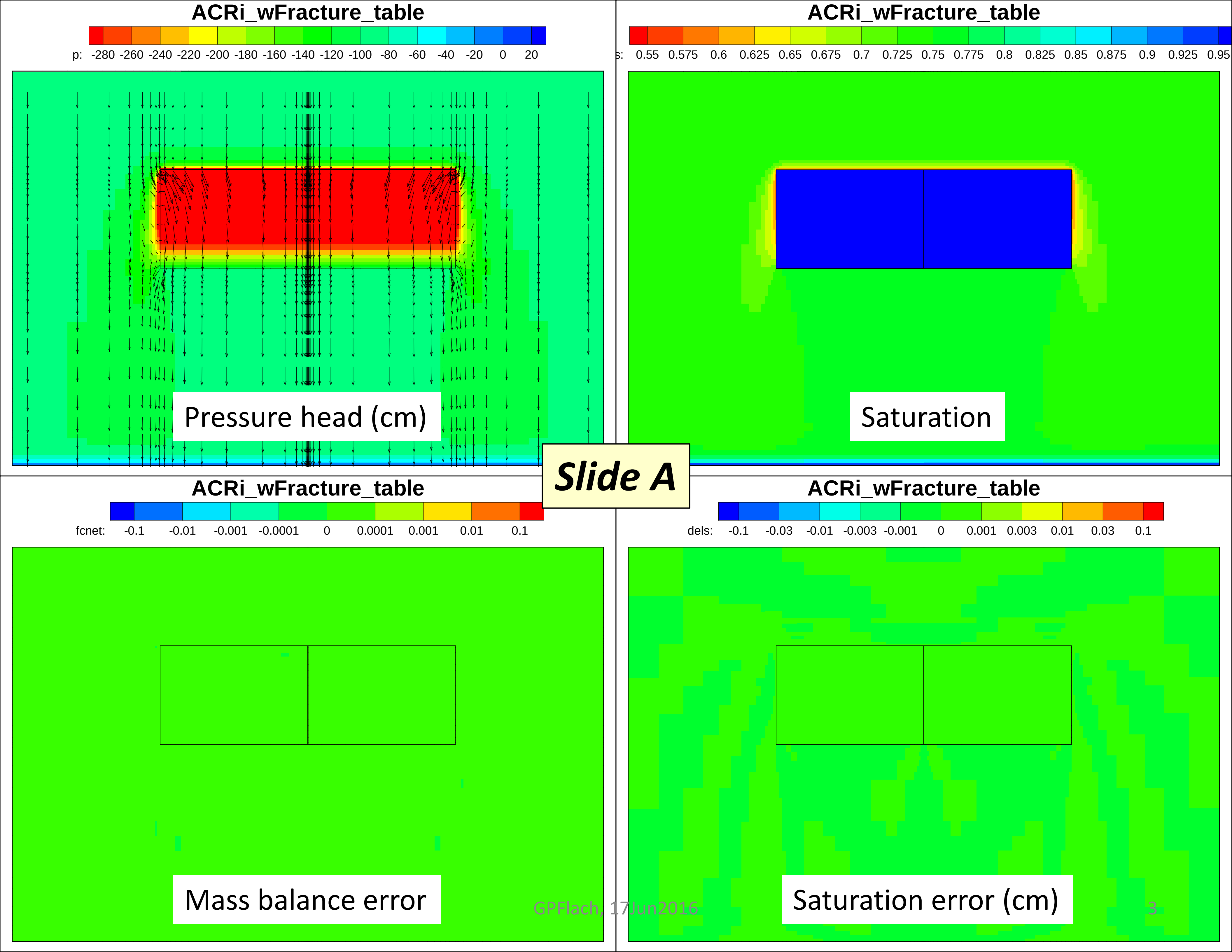
The matrix iterations saves me cpu time by eliminating some incomplete Cholesky factorizations.

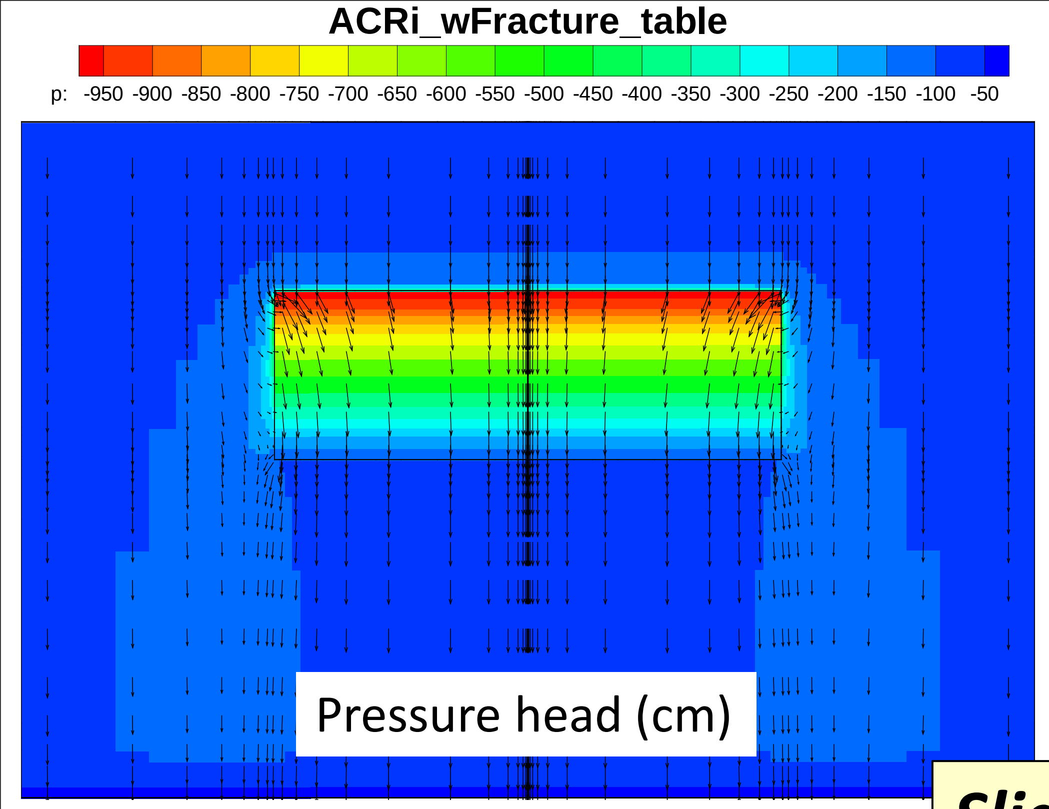
[3] Thereafter, I continued your approach of applying a sequence of reducing relaxation parameters for S,

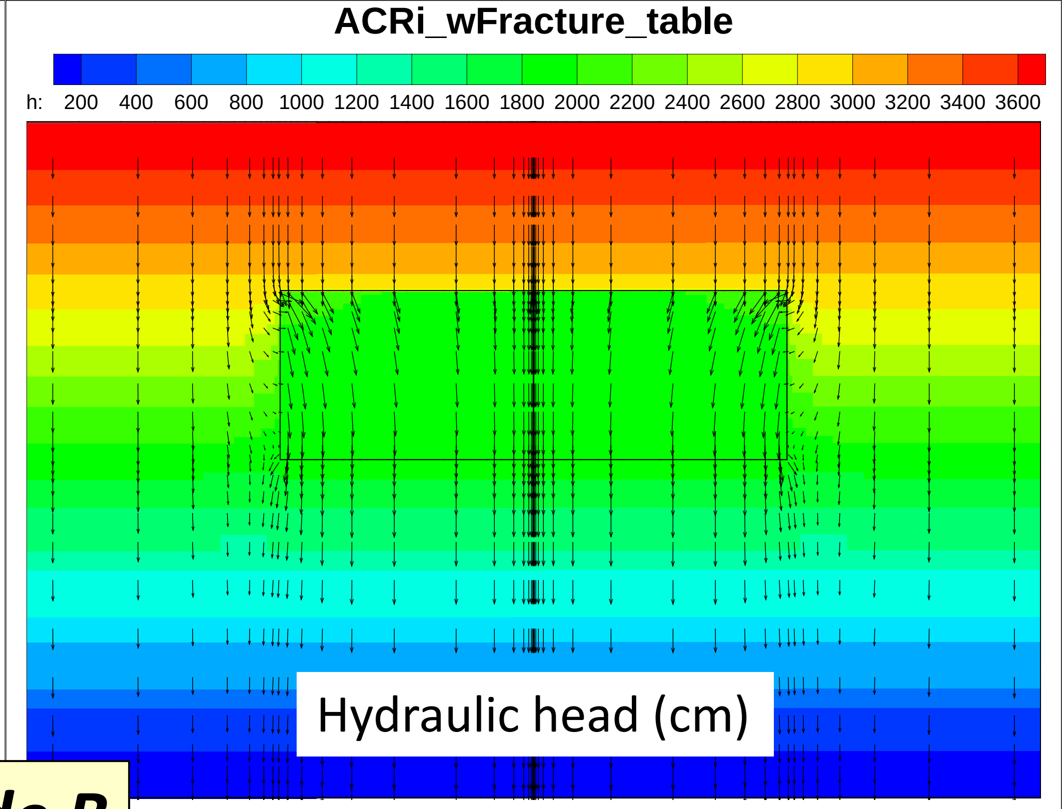
```
// Saturation relaxation
RELAx S 0.8
SOLVe STEAdy 100
```

Update to flow convergence test using new convergence approach

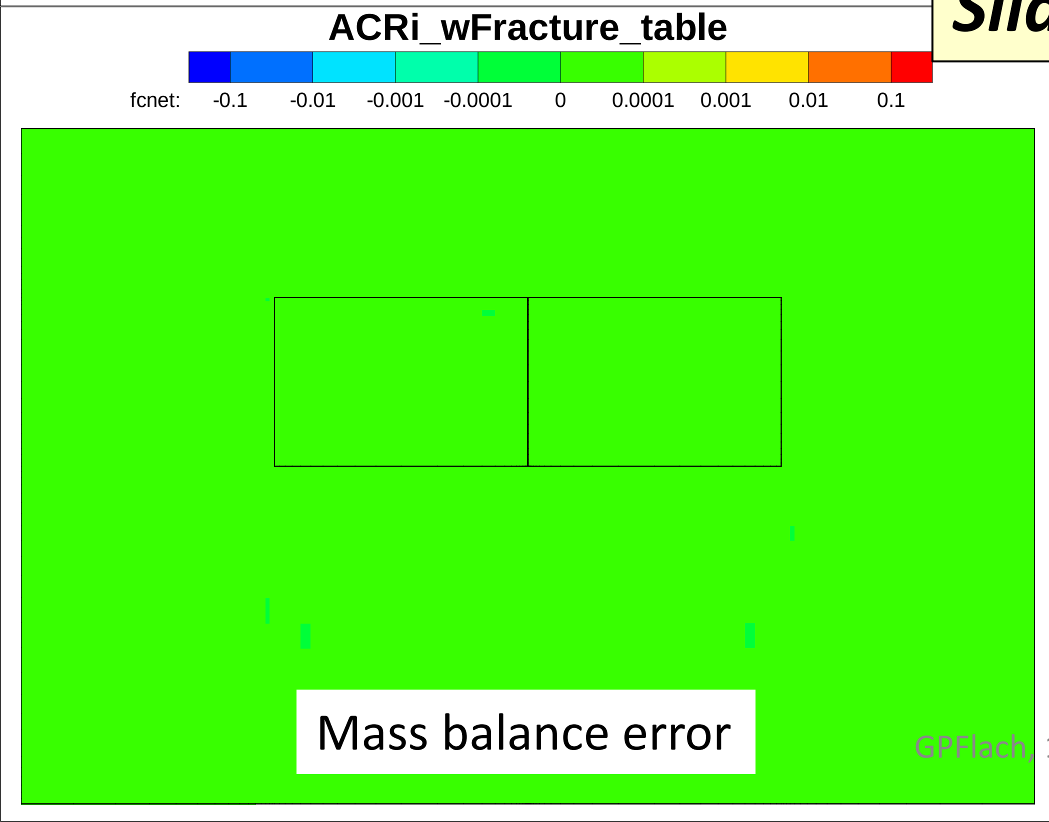
- **"ACRi" convergence scheme + vG curves in tabular form ("tabular")**
  - Slide A
    - Small mass balance (lower-left) and saturation (lower-right) errors
    - However, flow through low-permeability concrete is far too high
    - Flow does not divert around concrete and through fracture
  - Slide B
    - Pressure in upper-left rescaled
    - Saturation in upper-right replaced with hydraulic head
      - Small head gradient in concrete yet flow is ~40x saturated K
      - Darcy's law apparently not satisfied
- **"ACRi" convergence scheme + vG curves in parameter form ("vG")**
  - Slide C
    - Low mass balance errors (lower-left)
    - Large saturation errors (lower-right)
    - Flow diverts around concrete and through fracture as expected
  - Slide B
    - Pressure in upper-left rescaled
    - Saturation in upper-right replaced with hydraulic head
      - Flow is roughly equal to saturated K value (~ 1 cm/yr), about what is expected

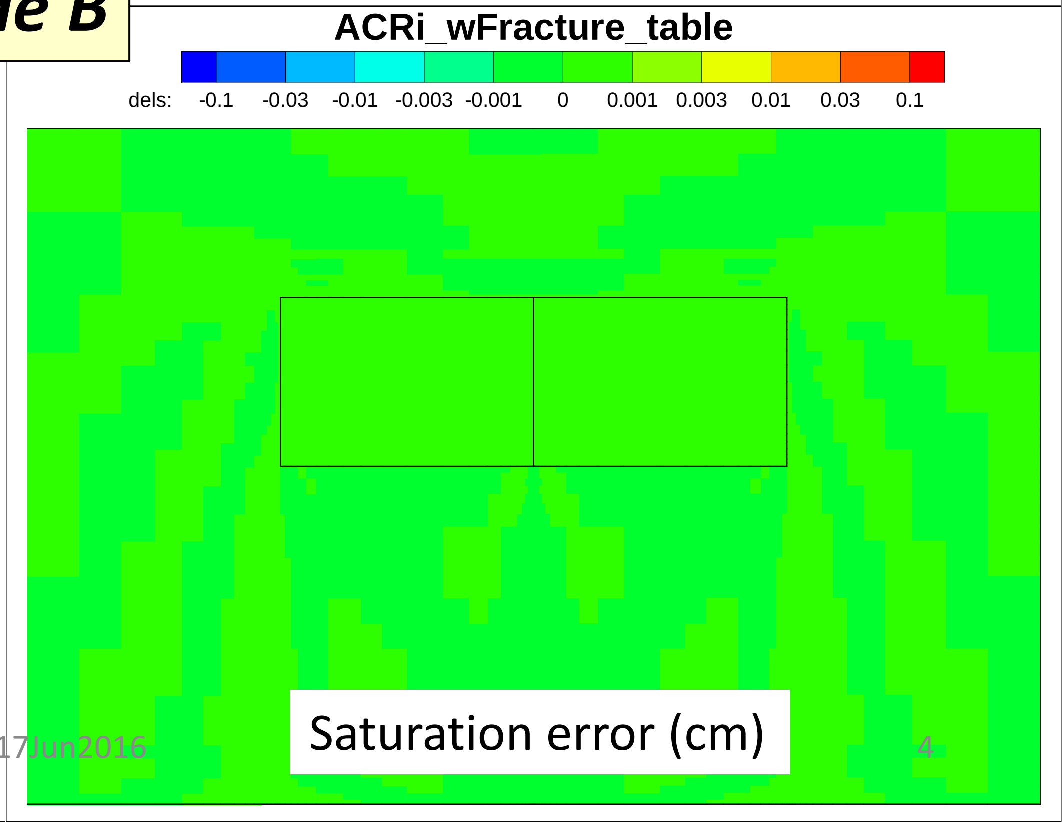

ACRi_wFracture_table
p: -280 -260 -240 -220 -200 -180 -160 -140 -120 -100 -80 -60 -40 -20 0 20
Pressure head (cm)
ACRi_wFracture_table
s: 0.55 0.575 0.6 0.625 0.65 0.675 0.7 0.725 0.75 0.775 0.8 0.825 0.85 0.875 0.9 0.925 0.95
Saturation
Slide A
ACRi_wFracture_table
fcnet: -0.1 -0.01 -0.001 -0.0001 0 0.0001 0.001 0.01 0.1
Mass balance error
ACRi_wFracture_table
dels: -0.1 -0.03 -0.01 -0.003 -0.001 0 0.001 0.003 0.01 0.03 0.1
Saturation error (cm)

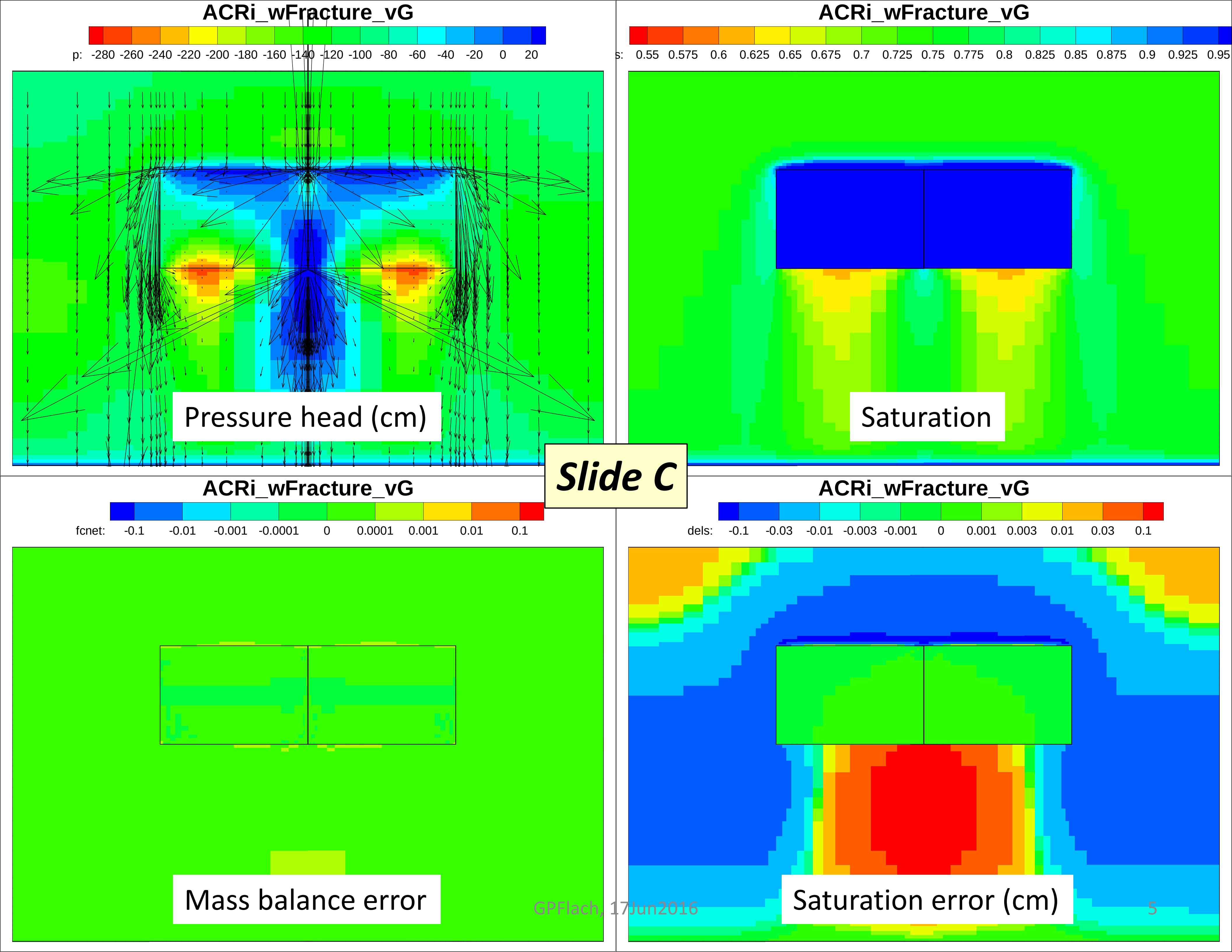

ACRi_wFracture_table
p: -950 -900 -850 -800 -750 -700 -650 -600 -550 -500 -450 -400 -350 -300 -250 -200 -150 -100 -50
Pressure head (cm)

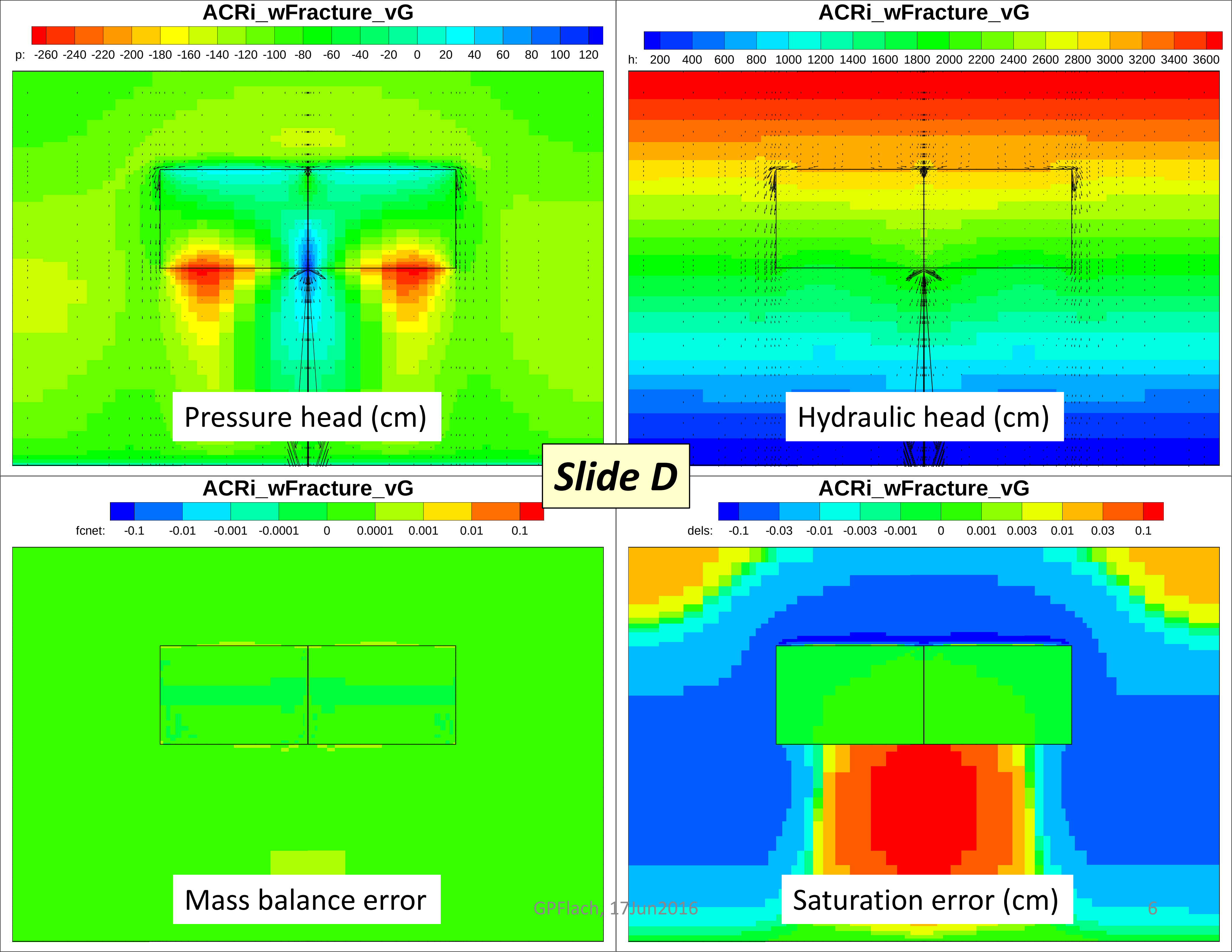

ACRi_wFracture_table
h: 200 400 600 800 1000 1200 1400 1600 1800 2000 2200 2400 2600 2800 3000 3200 3400 3600
Hydraulic head (cm)

Slide B


ACRi_wFracture_table
fcnet: -0.1 -0.01 -0.001 -0.0001 0 0.0001 0.001 0.01 0.1
Mass balance error


ACRi_wFracture_table
dels: -0.1 -0.03 -0.01 -0.003 -0.001 0 0.001 0.003 0.01 0.03 0.1
Saturation error (cm)


ACRi_wFracture_vG
p: -280 -260 -240 -220 -200 -180 -160 -140 -120 -100 -80 -60 -40 -20 0 20
Pressure head (cm)
ACRi_wFracture_vG
s: 0.55 0.575 0.6 0.625 0.65 0.675 0.7 0.725 0.75 0.775 0.8 0.825 0.85 0.875 0.9 0.925 0.95
Saturation
Slide C
ACRi_wFracture_vG
fcnet: -0.1 -0.01 -0.001 -0.0001 0 0.0001 0.001 0.01 0.1
Mass balance error
ACRi_wFracture_vG
dels: -0.1 -0.03 -0.01 -0.003 -0.001 0 0.001 0.003 0.01 0.03 0.1
Saturation error (cm)


ACRi_wFracture_vG
p: -260 -240 -220 -200 -180 -160 -140 -120 -100 -80 -60 -40 -20 0 20 40 60 80 100 120
Pressure head (cm)
ACRi_wFracture_vG
h: 200 400 600 800 1000 1200 1400 1600 1800 2000 2200 2400 2600 2800 3000 3200 3400 3600
Hydraulic head (cm)
Slide D
ACRi_wFracture_vG
fcnet: -0.1 -0.01 -0.001 -0.0001 0 0.0001 0.001 0.01 0.1
Mass balance error
ACRi_wFracture_vG
dels: -0.1 -0.03 -0.01 -0.003 -0.001 0 0.001 0.003 0.01 0.03 0.1
Saturation error (cm)

31 May 2017

Proposed Relaxation Scheme

Conventional relaxation scheme (Varga 1962) implemented in PORFLOW (RELAX manual page, version 6.40.0):

$$\Phi^{k+1} = \Theta\Phi^{computed} + (1-\Theta)\Phi^{k} \qquad \text{(in whole model domain)}$$

Rewrite as:

$$\Phi^{k+1} = \Phi^{k} + \Theta\left(\Phi^{computed} - \Phi^{k}\right) \equiv \Phi^{k} + \Delta\Phi^{k+1} \qquad \text{(in whole model domain)}$$

Proposed enhancement:

$$\Delta\Phi^{k+1} = min\left[\Theta\left(\Phi^{computed} - \Phi^{k}\right), \Delta\Phi_{max}\right] \quad \text{in ID=idsub}$$

Proposed keyword implementation (example using saturation variable):

RELAxation factor in ID=**subrgn** for **P** = **N$_1$** with maximum **ABSO**lute change = **N$_2$**

(applies to whole model domain if no ID=**subrgn** specified)

Motivations:

- Apply relaxation only to regions of the model domain presenting convergence difficulties
- Make the effective relaxation factor, $\Theta_{eff} \equiv \Delta\Phi^{k+1}/\left(\Phi^{computed} - \Phi^{k}\right)$, smaller in regions exhibiting greater numerical instability, and larger in regions exhibiting lower variability
- Allow well-behaved grid variables to migrate unconstrained to their converged solution values

Example schemes of interest:

RELAxation factor for **P** = **0.5**
SOLVe **STEA**dy **10** iterations

RELAxation factor for **P** = **0.5** with maximum **ABSO**lute change = **100.0** centimeters
SOLVe **STEA**dy **10** iterations

RELAxation factor for **P** = **0.5** with maximum **ABSO**lute change = **10.0** centimeters
SOLVe **STEA**dy **10** iterations

RELAxation factor for **P** = **0.5** with maximum **ABSO**lute change = **1.0** centimeters
SOLVe **STEA**dy **10** iterations

or using saturation

RELAxation factor for **S** = **0.5** with maximum **ABSO**lute change = **0.01**
SOLVe **STEA**dy **10** iterations
etc.

RE: Enhanced RELAxation capability?
Gregory Flach to: runchal 06/01/2017 12:00 PM
Cc: luther.hamm, "Madhukar M. Rao \(adjunct professor\)", tom.butcher

Dear Akshai,

1) Re: "The proposed scheme does present a problem if it is to be applied to a sub-region" . . . that aspect is not critical. A global max change limit should suffice. Let's forget about restricting relaxation to a subregion.

2) Re: "it is not very elegant ..." . . . I agree.

3) Re: "... and creates one more arbitrary decision for the user" . . . Relaxation applied as a change cutoff rather than a factor may be more effective than our current scheme and thus require less of the user. However, point well taken -- relieving the user of decisions would be better.

4) Re: "I think we need to tackle it at the solver level" . . . that is preferable from my perspective. We could postpone consideration of a max change limit if you think pursuing something at the solver level would be more productive. Or, if implementing a global change limiter is relatively easy, then we could pursue parallel tracks.

5) Re: "Is it possible that you can make up a simple (the simplest possible) problem that illustrates this feature" . . . yes. I'm revisiting the suite of flow convergence test cases transmitted about a year ago. **Can you provide an upload link?**

6) I'll be on vacation from this afternoon through next week. A phone call (proposed by Larry) would be best scheduled for the week of 6/12.

Greg Flach
Savannah River National Laboratory
773-42A, Savannah River Site, Aiken, SC 29808
803-725-5195
gregory.flach@srnl.doe.gov

"runchal"	The practical way to solve the stability issue is t...	06/01/2017 07:59:24 AM

From: "runchal" <runchal@gmail.com>
To: <gregory.flach@srnl.doe.gov>
Cc: <tom.butcher@srnl.doe.gov>, <luther.hamm@srnl.doe.gov>, "Madhukar M. Rao \(adjunct professor\)" <Madhukar.Rao@ACRiCFD.com>
Date: 06/01/2017 07:59 AM
Subject: RE: Enhanced RELAxation capability?

The practical way to solve the stability issue is to solve for P and S simultaneously coupling the P, S, Hydraulic conductivity at nodes, Hydraulic conductivity at faces, into one set of unknowns and using Picard or Newton-Raphson to iterate.

cheers,
Madhu

Dear Greg & Larry:

The proposed scheme does present a problem if it is to be applied to a sub-region. The reason is that once we enter the solver, we do not carry any subregion identity into the solver. Also it will add to cost of adding extra subregion based DO loop rather than over the matrix. It is doable but both time consuming and will add some cost to solver. In my opinion it is not very elegant also and creates one more arbitrary decision for the user.. I am copying this to Dr. Rao who may have a different take on it.

We have talked about finding a different approach to solve this speed problem. I think we need to tackle it at the solver level and look for a different solver or solver-preconditioner system, I think at one time Larry had some ideas about how to go about.
Is it possible that you can make up a simple (the simplest possible) problem that illustrates this features and we try finding a better solver? Or a better method of solving it. Perhaps Larry can take alook at the Matric and see something that can be improved upon.

Cheers
Akshai
==================================

The CFD Innovators
==================================

From: gregory.flach@srnl.doe.gov [mailto:gregory.flach@srnl.doe.gov]
Sent: Wednesday, May 31, 2017 10:12 AM
To: runchal <runchal@gmail.com>
Cc: tom.butcher@srnl.doe.gov; luther.hamm@srnl.doe.gov
Subject: Enhanced RELAxation capability?

Dear Akshai,

As you know from prior conversations, we have struggled to achieve converged steady-state PORFLOW solutions for certain unsaturated flow simulations. Our current practice is to apply increasing under-relaxation, as in the example snippet shown below. This approach generally produces a converged solution, but is slow and not always reliable. The attached PDF proposes two extensions to the current RELAxation keyword: 1) optional specification of a maximum change, and 2) optional restriction to a sub-region. Compared to relaxation applied uniformly, these extensions are intended to focus relaxation on grid locations exhibiting the greatest instability. We are guessing that both enhancements (or at least the maximum change limiter) could be implemented with a relatively small effort. Could this scope be pursued as evolutive development under our annual software support contract? SRNL would evaluate the efficacy of this scheme and report our findings.

I'll be on vacation starting midday Thursday June1 and return to the office Monday June 5. Please reply-to-all and Tom Butcher and/or Larry Hamm can respond as needed during my absence.

Greg Flach
Savannah River National Laboratory
773-42A, Savannah River Site, Aiken, SC 29808
803-725-5195
gregory.flach@srnl.doe.gov

```
========
!!Migrate to solution neighborhood
CONVergence for P    1.e-6, 10 iterations max
SOLVe STEAdy 5
RELAx S 0.7
SOLVe STEAdy 5
RELAx S 0.3
SOLVe STEAdy 15
RELAx S 0.1
SOLVe STEAdy 45
!!Mixed purpose
CONVergence for P    1.e-6, 30 iterations max
RELAx S 0.03
SOLVe STEAdy 50
RELAx S 0.01
SOLVe STEAdy 50
!!Suppress noise / sharpen mass balance
CONVergence for P    1.e-6, 100 iterations max
RELAx S 0.003
SOLVe STEAdy 20
RELAx S 0.001
SOLVe STEAdy 20
RELAx S 0.0003
SOLVe STEAdy 20
RELAx S 0.0001
SOLVe STEAdy 20
RELAx S 0.00001
SOLVe STEAdy 20
RELAx S 0.000001
SOLVe STEAdy 20
```