Nonlinear systems arising from PDEs are typically solved by a variant of Newton iteration, which may not converge or converge very slowly for the problems that are “nonlinearly stiff”. Additive Schwarz Preconditioned Inexact Newton (ASPIN) [1], a form of nonlinear subspace correction method, is intended to enhance robustness by changing coordinates. ASPIN works well for some fluids systems with unbalanced nonlinearity [2], but few applications to reservoir simulation have been reported.

Consider the incompressible two-phase flow for the pressure $p$, and saturation $S$ (where $p$ is capillary pressure):

$$
\nabla \cdot \left( \lambda \nabla p \right) = q_t + \nabla \cdot \left( \lambda w \nabla p_w \right) - \nabla \cdot \left( (\lambda_s p_s + \lambda_w p_w) \kappa_{g} \right),
$$

(1)

$$
\frac{\partial S}{\partial t} - \nabla \cdot \left( \lambda_w \kappa_{g} (\nabla p_w - \rho_w g) \right) = q_w.
$$

(2)

Using the standard two-point flux-approximation (TPFA) finite volume scheme, we obtain a nonlinear system at the time $t_k$

$$
F(u^k) = \left[ F_1(u^k), F_2(u^k) \right] = 0
$$

(3)

where $u^k = [p_1^k, p_2^k, \ldots, p_n^k, S_1, S_2, \ldots, S_N]^T$.

**Key idea:** Finding the solution $u^*$ by solving an equivalent nonlinear system

$$
F(u^*) = 0 \iff F(u^*) = 0
$$

using the Inexact Newton method with Backtracking (INB).

How to construct the equivalent nonlinear system? Sum the independent Newton corrections for each subdomain of the original problem!

At the root, these go to zero:

$$
F_{i_{\Omega_i}}(u - T_{i_{\Omega_i}}(u)) = 0, \quad i = 1, \ldots, N.
$$

(4)

$$
F(u) = \sum_{i=1}^{N} F_{i_{\Omega_i}}(u), \quad \bigcup_{i=1}^{N} \Omega_i = \Omega.
$$

(5)

The Jacobian of $F(u)$ can be approximated as

$$
\sum_{i=1}^{N} T_{i_{\Omega_i}}(u) = J \approx \tilde{J} \equiv \sum_{i=1}^{N} J_{i_{\Omega_i}}^{-1} J.
$$

(6)

**Advantage over Newton**

Nonlinear iteration temporal convergence history for Layer 26 from the SPE10 benchmark, comparing convergence of ASPIN and global inexact Newton with backtracking for 300 days. Time step is adaptively selected up to a maximum of 10 days.

**Scaling behavior**

<table>
<thead>
<tr>
<th>Subdomain partition</th>
<th>case A</th>
<th>case B</th>
<th>case C</th>
<th>case D</th>
</tr>
</thead>
<tbody>
<tr>
<td>$2 \times 2 = 4$</td>
<td>2.3</td>
<td>2.5</td>
<td>2.7</td>
<td>2.7</td>
</tr>
<tr>
<td>$4 \times 4 = 16$</td>
<td>2.9</td>
<td>3.0</td>
<td>2.8</td>
<td>2.8</td>
</tr>
<tr>
<td>$8 \times 8 = 64$</td>
<td>3.6</td>
<td>3.6</td>
<td>2.9</td>
<td>2.9</td>
</tr>
</tbody>
</table>

Avg. execution time per timestep, strong scaling

<table>
<thead>
<tr>
<th>Subdomain partition</th>
<th>case A</th>
<th>case B</th>
<th>case C</th>
<th>case D</th>
</tr>
</thead>
<tbody>
<tr>
<td>$2 \times 2 = 4$</td>
<td>3.724e+03</td>
<td>4.424e+03</td>
<td>1.192e+04</td>
<td>7.654e+03</td>
</tr>
<tr>
<td>$4 \times 4 = 16$</td>
<td>6.924e+03</td>
<td>8.584e+02</td>
<td>5.338e+03</td>
<td>3.965e+03</td>
</tr>
<tr>
<td>$8 \times 8 = 64$</td>
<td>1.983e+02</td>
<td>2.525e+02</td>
<td>3.850e+03</td>
<td>2.644e+03</td>
</tr>
</tbody>
</table>

**Follow-on work**

Nonlinear scaling is excellent! To improve scalability of linear solver, two-level ASPIN methods are needed.

Current solver is implemented in a synchronous way for the outer loop.

1. Potentially leads to load imbalance among the processors handling the local problems.
2. Some local problems terminate earlier than others.
3. In production, these processors could “steal” work.
4. Alternatively, the decomposition strategy could adapt to approximately equidistribute subdomain problem difficulty.

Other decomposition strategies that respect inhomogeneity and anisotropy in preserving local strong coupling may be more efficient than our simple geometric strategy.

**References**
