Non-linear Newton Solver for a Polymer Two-phase System Using Interface-localized Trust Regions

Ø. S. Klemetsdal[1,2], O. Møyner[2] K.- A. Lie[2]

[1]Norwegian University of Science and Technology, [2]SINTEF Digital

https://goo.gl/yZu45s

(1) – Motivation

- Models of polymer flooding account for several processes such as concentration dependent viscosity, adsorption, incomplete mixing, inaccessible pore space, and reduced permeability effects.
- The resulting nonlinear systems are strongly coupled, and challenging to solve numerically.
- In this work, we present a method that offers unconditional convergence for any time step, and demonstrate its applicability to industry grade complexity.

(2) – Model for polymer flooding

- Diluted polymer is modeled through an additional conservation equation:

$$\partial_t\big(\phi b_w c S_w + \rho_r(1-\phi_0)c_a\big) + \nabla\cdot(c b_w \vec{v}_p) = b_w q_p.$$

- Diluted polymer and water is modelled as an immiscible system using the Todd-Longstaff model:

$$\vec{v}_w = -\frac{k_{rw}(S_w)K}{\mu_{w,\mathrm{eff}}(c)R_k(c)}(\nabla p - \rho_w g\nabla z), \quad \vec{v}_p = m(c)\vec{v}_w.$$

Viscosity enhancement — Permeability reduction

- Industry grade polymer model in the MATLAB reservoir simulation toolbox (MRST) (Bao et al., 2016).

Todd-Longstaff mixing

c Polymer concentration, $0 \le c \le c^*$
c_a Adsorption concentration
ω Mixing parameter, $0 \le \omega \le 1$
μ_m Viscosity of fully mixed system
R_k Actual resistance factor

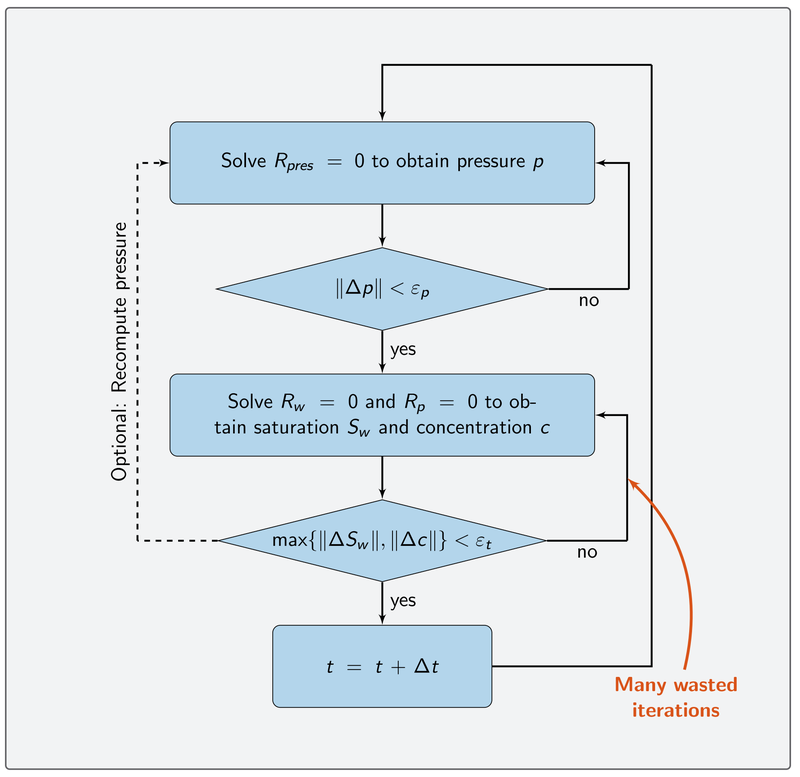
$$\frac{1}{\mu_{w,\mathrm{eff}}} = \frac{1-c/c^*}{\mu_m(c)^\omega \mu_w^{1-\omega}} + \frac{c/c^*}{\mu_m(c)^\omega \mu_m(c^*)^{1-\omega}}$$

$$m(c) = \left[\left(1-\frac{c}{c^*}\right)\left(\frac{\mu_m(c^*)}{\mu_w}\right)^{1-\omega} + \frac{c}{c^*}\right]^{-1}$$

(3) – Sequential solution strategy

We introduce a grid consisting of cells C_i and integrate over each cell in space to obtain finite-volume residual equations for the water and polymer transport, and the pressure:

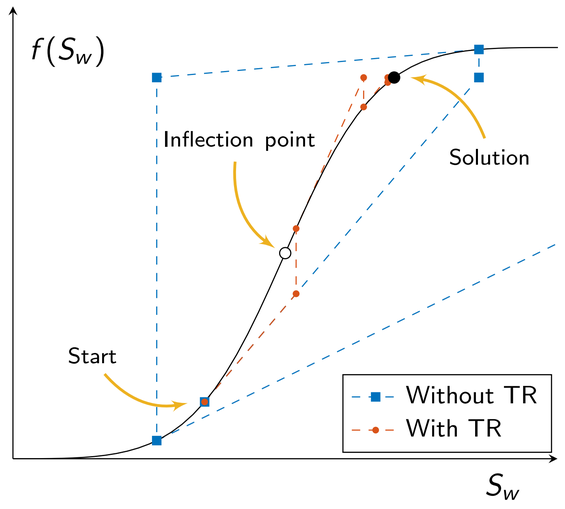
$$R_{w,i}(S_w, c) = 0, \quad R_{p,i}(S_w, c) = 0, \quad R_{pres,i}(p) = 0.$$



The industry standard is to heuristically chop the time step and/or dampen the saturation updates until the Newton solver converges, which may result in a huge number of wasted iterations.

(4) – Trust-region algorithm

Inflection points and kinks in the residual functions causes convergence problems in the Newton solver. The idea of the trust-region algorithm (Jenny et al., 2009) is to identify inflection points/kinks, and use them to determine safe updates.



- Transport problem: Find $\boldsymbol{\xi}$ such that

$$R(\boldsymbol{\xi}) = \mathbf{0}, \text{ where } \begin{cases} \boldsymbol{\xi} &= (S_1, \ldots, S_N, c_1, \ldots, c_N), \\ R &= (R_{w,1}, \ldots, R_{w,N}, R_{p,1}, \ldots, R_{p,N}). \end{cases}$$

- Full Newton update:

$$\boldsymbol{\xi}_i^{\ell+1} = \boldsymbol{\xi}_i^{\ell} + \Delta\boldsymbol{\xi}_i^{\ell}, \quad \Delta\boldsymbol{\xi} = -J(\boldsymbol{\xi})^{-1}R(\boldsymbol{\xi}).$$

- Find *damping factors* $\theta_i \in [0,1]$ so that updates do not pass far beyond problematic points:

$$\boldsymbol{\xi}_i^{\ell+1} = \boldsymbol{\xi}_i^{\ell} + \theta_i\Delta\boldsymbol{\xi}_i^{\ell}.$$

- Can be reduced to a 1D-problem (Møyner, 2016) by considering possible updates in the direction $\boldsymbol{d}$ of the Newton update at each interface:

$$\boldsymbol{d} = \frac{(\Delta\boldsymbol{\xi}_i, \Delta\boldsymbol{\xi}_j)}{\|(\Delta\boldsymbol{\xi}_i, \Delta\boldsymbol{\xi}_j)\|}, \quad \Delta\boldsymbol{\xi}_\ell = (\Delta S_\ell, \Delta c_\ell).$$

- Gives damping factors $\theta_{\gamma,ij}$ for $\gamma = w, p$ and all interfaces $ij \rightarrow$ *interface-localized*.

Figure 1: Subset of SPE 10 Model 2

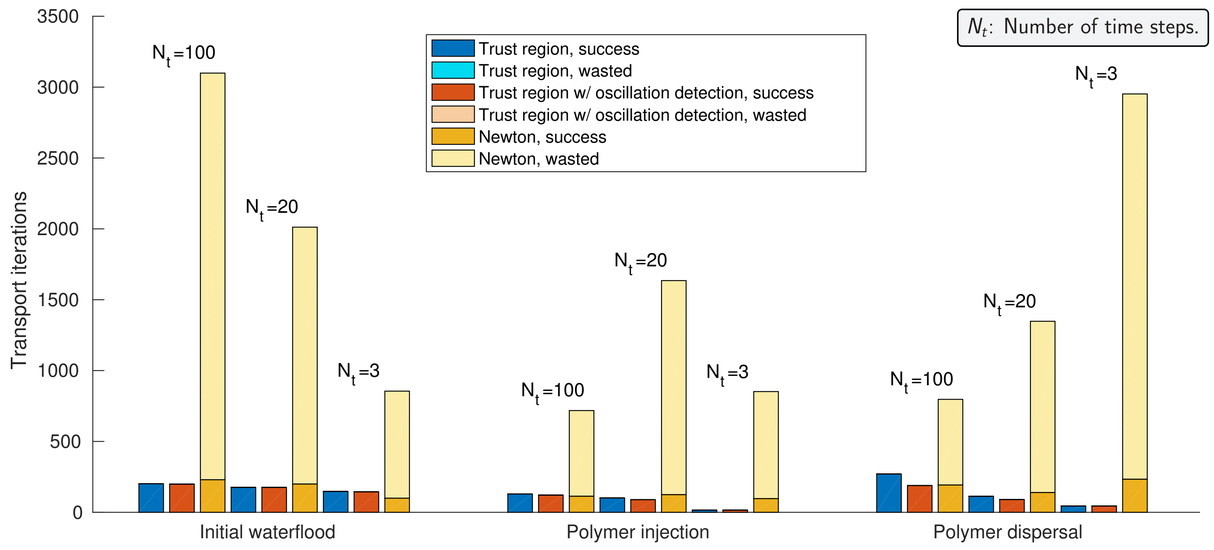
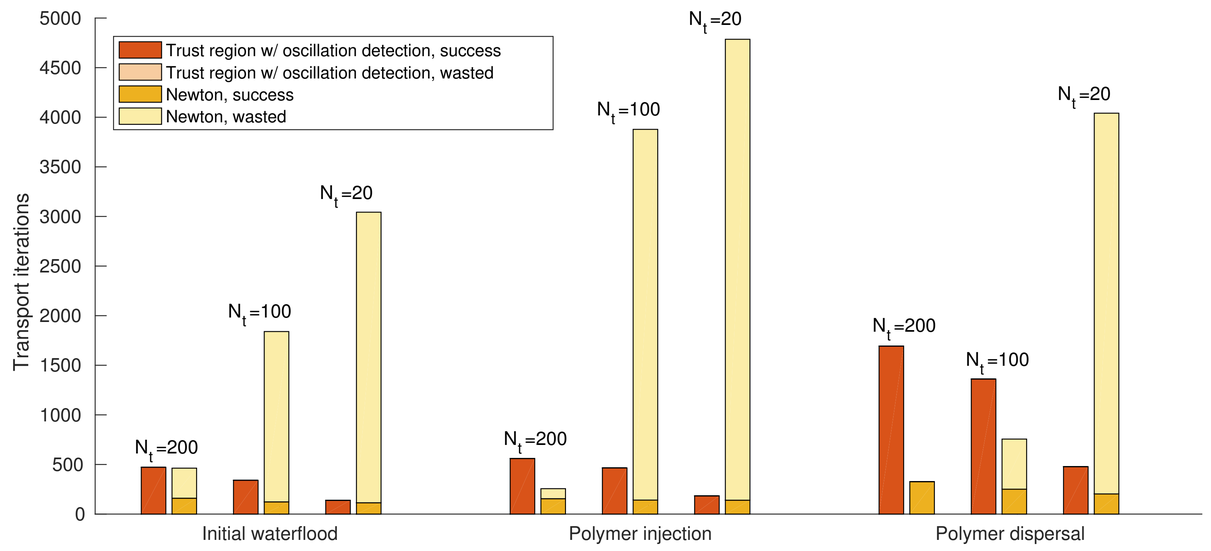


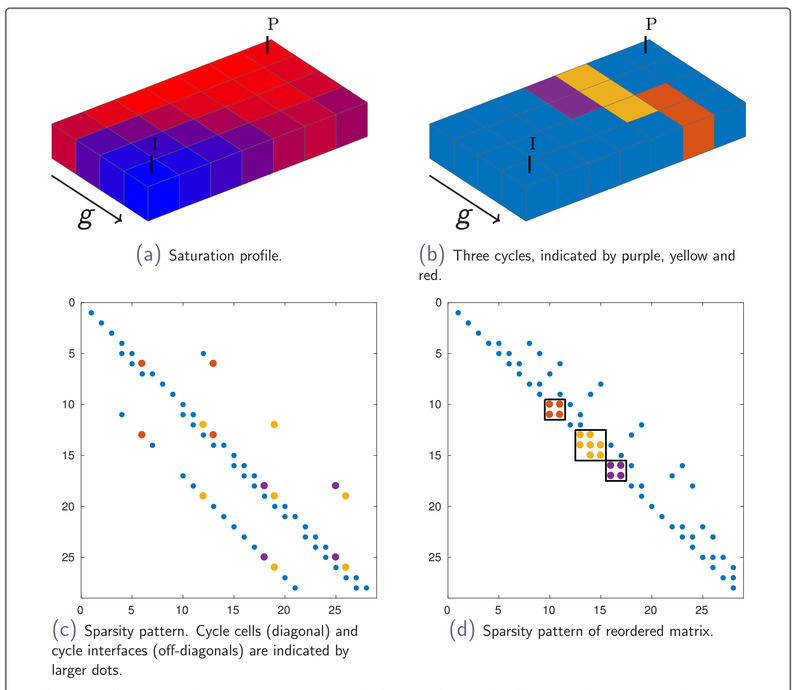
Figure 2: Field model



(5) – Algorithm improvements

Local vs. global chopping:

- Global approach: Set all damping factors equal to the smallest: $\theta = \min_{\gamma,i,j}\{\theta_{\gamma,ij}\}$. May be overly conservative.
- Local approach (Møyner, 2016): Introduce directed graph $C_{\Delta\xi}$, where $(C_{\Delta\xi})_{i,j} = 1$ if either the saturation or concentration update in cell j has an impact on the saturation or concentration error in cell i larger than a given threshold. This is used to assign damping factors to each cell.



(a) Saturation profile.

(b) Three cycles, indicated by purple, yellow and red.

(c) Sparsity pattern. Cycle cells (diagonal) and cycle interfaces (off-diagonals) are indicated by larger dots.

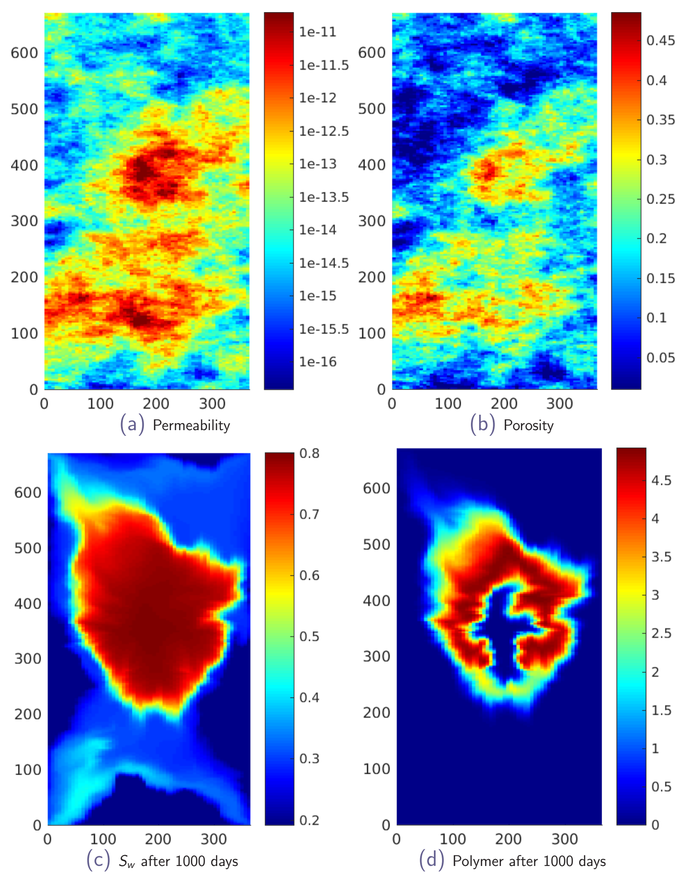
(d) Sparsity pattern of reordered matrix.

Figure: Cycles and sparsity pattern of directed graph $C_{\Delta\xi}$ used in local chopping.

Oscillation detection

- If the saturation or concentration update changes sign from one iteration to the next, we have either passed the solution, an inflection point/kink, or a local min/max.
- To reduce unnecessary use of the trust-region algorithm, we only apply it if we detect oscillations in the updates.

(6) – Example: Subset of SPE 10 Model 2



(a) Permeability

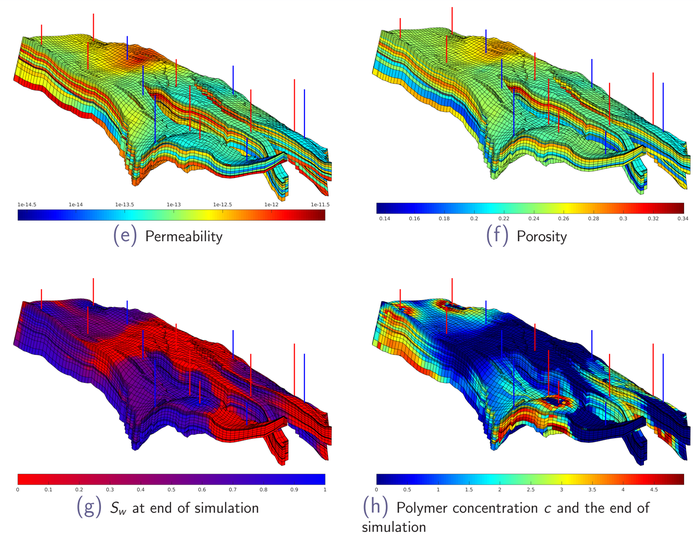
(b) Porosity

(c) S_w after 1000 days

(d) Polymer after 1000 days

- Horizontal layer from the SPE10 model 2.
- Polymer slug injected from 20 % to 40 % of the injection period.
- Three different simulation schedules with $N_t = 100$, 20 and 3 time steps, simulated using MRST (Krogstad et al., 2015).
- The Newton method performs a huge number of wasted iterations. The number of successful iterations are close to those of the trust-region method, which has no wasted iterations (Figure 1).

(7) – Example: Field model



(e) Permeability

(f) Porosity

(g) S_w at end of simulation

(h) Polymer concentration c and the end of simulation

- Slightly modified grid model of the Norne oil and gas field, with artificial well pattern.
- Polymer slug injected from 20 % to 40 % of the injection period in all injectors, assumed to be fully mixed with water.
- Schedules: $N_t = 200$, 100 and 20 uniform time steps.
- Trust-region solver is too conservative using 200 time steps, but uses significantly fewer iterations for 100 and 20 time steps. Moreover, it has no wasted iterations (Figure 2).

(8) – References

Bao, K., Lie, K.-A., Møyner, O., and Liu, M. (2016). Fully implicit simulation of polymer flooding with mrst. In *ECMOR XV-15th European Conference on the Mathematics of Oil Recovery*.

Jenny, P., Tchelepi, H. A., and Lee, S. H. (2009). Unconditionally convergent nonlinear solver for hyperbolic conservation laws with s-shaped flux functions. *J. Comput. Phys.*, 228(20):7497–7512.

Krogstad, S., Lie, K.-A., Møyner, O., Nilsen, H. M., Raynaud, X., and Skaflestad, B. (2015). MRST-AD – an open-source framework for rapid prototyping and evaluation of reservoir simulation problems. In *SPE Reservoir Simulation Symposium, 23–25 February, Houston, Texas*.

Møyner, O. (2016). Nonlinear solver for three-phase transport problems based on approximate trust regions. In *ECMOR XV – 15th European Conference on the Mathematics of Oil Recovery, Amsterdam, Netherlands, 29 Aug–1 Sept*.

Funded by the Research Council of Norway, grant no. 244361

oystein.klemetsdal@ntnu.no