Additive Schwarz Preconditioned Exact Newton Method as a Nonlinear Preconditioner for Multiphase Porous Media Flow

Øystein S. Klemetsdal[1] Arthur Moncorgé[2] Halvor Møll Nilsen[1]
Olav Møyner[1] Knut-Andreas Lie[1]

[1]Department of Mathematics and Cybernetics, SINTEF Digital, Norway
[2]Total E&P, Pau, France

ECMOR XVII – 17th European Conference on the Mathematics of Oil Recovery
14–17 September 2020, Edinburgh, UK (Online)

Introduction: Domain decomposition methods in reservoir simulation

	Variable	Space
Linear		
Nonlinear		

Introduction: Domain decomposition methods in reservoir simulation

	Variable	Space
Linear	✓	✓
Nonlinear		

- Linear domain decomposition
  - Well-established means to construct scalable algorithms
    Wallis et al. [1985], Killough and Wheeler [1987], Stueben [2001], Lie et al. [2017]

Introduction: Domain decomposition methods in reservoir simulation

	Variable	Space
Linear	✓	✓
Nonlinear	✓*	

- Linear domain decomposition
  - Well-established means to construct scalable algorithms
    Wallis et al. [1985], Killough and Wheeler [1987], Stueben [2001], Lie et al. [2017]
- Nonlinear domain decomposition
  - Popular in variable domain $\rightarrow$ sequential splitting
    Watts [1986], Trangenstein and Bell [1989], Jenny et al. [2006], Møyner and Lie [2016]

Introduction: Domain decomposition methods in reservoir simulation

	Variable	Space
Linear	✓	✓
Nonlinear	✓*	✗

- Linear domain decomposition
  - Well-established means to construct scalable algorithms
    Wallis et al. [1985], Killough and Wheeler [1987], Stueben [2001], Lie et al. [2017]
- Nonlinear domain decomposition
  - Popular in variable domain $\rightarrow$ sequential splitting
    Watts [1986], Trangenstein and Bell [1989], Jenny et al. [2006], Møyner and Lie [2016]
  - Not so much in spatial domain
  - However: problem equations are strongly coupled, highly nonlinear, unbalanced
    $\rightarrow$ Spatial domain decomposition is an excellent nonlinear a preconditioner

Introduction: Domain decomposition methods in reservoir simulation

Additive Schwarz Preconditioned Inexact Newton (ASPIN) (Cai and Keyes [2002])

- Spatial domain decomposition (Liu et al. [2013], Skogestad et al. [2013] (two-phase flow))
- Variable domain decomposition (Li et al. [2019] (MS wells), Wong [2018] (geothermal))
- Related: Dolean et al. [2016] (Forchheimer equation), Liu et al. [2013] (Multiplicative SPIN)

Introduction: Domain decomposition methods in reservoir simulation

Additive Schwarz Preconditioned Inexact Newton (ASPIN) (Cai and Keyes [2002])

- Spatial domain decomposition (Liu et al. [2013], Skogestad et al. [2013] (two-phase flow))
- Variable domain decomposition (Li et al. [2019] (MS wells), Wong [2018] (geothermal))
- Related: Dolean et al. [2016] (Forchheimer equation), Liu et al. [2013] (Multiplicative SPIN)

In this work: apply nonlinear domain decomposition preconditioning (NLDDP) (exact/inexact) to complex problems with realistic geology and fluid physics

Nonlinear domain decomposition preconditioning

- Conservation of mass of a component i on discrete, implicit form

$$\mathbf{R}_i^{n+1} = \frac{1}{\Delta t^n}(\underbrace{\mathbf{M}_i^{n+1}}_{\text{Mass}} - \mathbf{M}_i^n) + \mathtt{div}(\underbrace{\mathbf{V}_i^{n+1}}_{\text{Flux}}) - \underbrace{\mathbf{Q}_i^{n+1}}_{\text{Sources/sinks}} = 0$$

- `div`: discrete divergence operator (two-point, multipoint, mimetic, etc.)
  - In this work: linear two-point flux approximation

Nonlinear domain decomposition preconditioning

- Conservation of mass of a component i on discrete, implicit form

$$\mathbf{R}_i^{n+1} = \frac{1}{\Delta t^n}(\underbrace{\mathbf{M}_i^{n+1}}_{\text{Mass}} - \mathbf{M}_i^{n}) + \mathtt{div}(\underbrace{\mathbf{V}_i^{n+1}}_{\text{Flux}}) - \underbrace{\mathbf{Q}_i^{n+1}}_{\text{Sources/sinks}} = 0$$

- `div`: discrete divergence operator (two-point, multipoint, mimetic, etc.)
  - In this work: linear two-point flux approximation
- Gather in nonlinear system $\mathbf{R}(\mathbf{u}) = \mathbf{0}$, linearize, and neglect higher-order terms $\rightarrow$ iterative Newton's method

$$\mathbf{u}^{k+1} = \mathbf{u}^k + \Delta\mathbf{u}, \quad -\frac{\partial \mathbf{R}}{\partial \mathbf{u}}\Delta\mathbf{u} = \mathbf{R}(\mathbf{u}^k)$$

Nonlinear domain decomposition preconditioning

Nonlinear preconditioning with two subdomains

- Partition unknowns $\mathbf{u}$ into two non-overlapping subdomains

$$\mathbf{R}(\mathbf{u}) = (\mathbf{R}_1(\mathbf{u}_1, \mathbf{u}_2), \mathbf{R}_2(\mathbf{u}_1, \mathbf{u}_2)) = \mathbf{0},$$

- Define solution operator $\mathcal{L}(\mathbf{u}) = (\mathcal{L}_1(\mathbf{u}), \mathcal{L}_2(\mathbf{u}))$, where

$$\mathbf{R}_1\big(\mathcal{L}_1(\mathbf{u}), \mathbf{u}_2\big) = \mathbf{0}, \quad \text{and} \quad \mathbf{R}_2\big(\mathbf{u}_1, \mathcal{L}_2(\mathbf{u})\big) = \mathbf{0}.$$

Nonlinear preconditioning with two subdomains

- Partition unknowns $\mathbf{u}$ into two non-overlapping subdomains

$$\mathbf{R}(\mathbf{u}) = (\mathbf{R}_1(\mathbf{u}_1, \mathbf{u}_2), \mathbf{R}_2(\mathbf{u}_1, \mathbf{u}_2)) = \mathbf{0},$$

- Define solution operator $\mathcal{L}(\mathbf{u}) = (\mathcal{L}_1(\mathbf{u}), \mathcal{L}_2(\mathbf{u}))$, where

$$\mathbf{R}_1\big(\mathcal{L}_1(\mathbf{u}), \mathbf{u}_2\big) = \mathbf{0}, \quad \text{and} \quad \mathbf{R}_2\big(\mathbf{u}_1, \mathcal{L}_2(\mathbf{u})\big) = \mathbf{0}.$$

Equivalent, fixed-point formulation of $\mathbf{R}(\mathbf{u}) = \mathbf{0}$

Find $\mathbf{u}$ so that $\mathbf{u} = \mathcal{L}(\mathbf{u})$, or $\mathbf{F}(\mathbf{u}) \equiv \mathbf{u} - \mathcal{L}(\mathbf{u}) = \mathbf{0}$

Nonlinear domain decomposition preconditioning

Equivalent, fixed-point formulation of $\mathbf{R}(\mathbf{u}) = \mathbf{0}$

Find $\mathbf{u}$ so that $\mathbf{u} = \mathcal{L}(\mathbf{u})$, or $\mathbf{F}(\mathbf{u}) \equiv \mathbf{u} - \mathcal{L}(\mathbf{u}) = \mathbf{0}$

- Fixed-point schemes tend to have poor convergence properties
  - Acceleration: Aitken, Anderson, quasi-Newton (Jiang and Tchelepi [2019])

Nonlinear domain decomposition preconditioning

Equivalent, fixed-point formulation of $\mathbf{R}(\mathbf{u}) = \mathbf{0}$

Find $\mathbf{u}$ so that $\mathbf{u} = \mathcal{L}(\mathbf{u})$, or $\mathbf{F}(\mathbf{u}) \equiv \mathbf{u} - \mathcal{L}(\mathbf{u}) = \mathbf{0}$

- Fixed-point schemes tend to have poor convergence properties
  - Acceleration: Aitken, Anderson, quasi-Newton (Jiang and Tchelepi [2019])
- Here: apply Newton's method directly to $\mathbf{F}(\mathbf{u})$:

$$\mathbf{u}^{k+1} = \mathbf{u}^k + \Delta\mathbf{u}, \quad -\frac{\partial \mathbf{F}}{\partial \mathbf{u}}\Delta\mathbf{u} = \mathbf{F}(\mathbf{u}^k), \quad \text{where} \quad \frac{\partial \mathbf{F}}{\partial \mathbf{u}} = \mathbf{I} - \begin{bmatrix} \frac{\partial \mathcal{L}_1}{\partial \mathbf{u}} \\ \frac{\partial \mathcal{L}_2}{\partial \mathbf{u}} \end{bmatrix}$$

  - Challenge: $\mathbf{F}$ implicitly defined through operator $\mathcal{L}$ – how to compute $\partial\mathbf{F}/\partial\mathbf{u}$?

Nonlinear domain decomposition preconditioning

Equivalent, fixed-point formulation of $\mathbf{R}(\mathbf{u}) = \mathbf{0}$

Find $\mathbf{u}$ so that $\mathbf{u} = \mathcal{L}(\mathbf{u})$, or $\mathbf{F}(\mathbf{u}) \equiv \mathbf{u} - \mathcal{L}(\mathbf{u}) = \mathbf{0}$

- Use that $\mathbf{R}_1(\mathcal{L}_1(\mathbf{u}), \mathbf{u}_2) = \mathbf{0}$ to find

$$\frac{\partial \mathbf{R}_1}{\partial \mathbf{u}} = \frac{\partial \mathbf{R}_1}{\partial \mathbf{u}_1} \frac{\partial \mathcal{L}_1}{\partial \mathbf{u}} + \frac{\partial \mathbf{R}_1}{\partial \mathbf{u}_2} \frac{\partial \mathbf{u}_2}{\partial \mathbf{u}} = \mathbf{0}$$

- Rearrange to obtain

$$\frac{\partial \mathcal{L}_1}{\partial \mathbf{u}} = -\left(\frac{\partial \mathbf{R}_1}{\partial \mathbf{u}_1}\right)^{-1} \frac{\partial \mathbf{R}_1}{\partial \mathbf{u}_2} \frac{\partial \mathbf{u}_2}{\partial \mathbf{u}} \quad \text{and} \quad \frac{\partial \mathcal{L}_2}{\partial \mathbf{u}} = -\left(\frac{\partial \mathbf{R}_2}{\partial \mathbf{u}_2}\right)^{-1} \frac{\partial \mathbf{R}_2}{\partial \mathbf{u}_1} \frac{\partial \mathbf{u}_1}{\partial \mathbf{u}}$$

- Natural extension to m subdomains

Nonlinear domain decomposition preconditioning

Equivalent, fixed-point formulation of $\mathbf{R}(\mathbf{u}) = \mathbf{0}$

Find $\mathbf{u}$ so that $\mathbf{u} = \mathcal{L}(\mathbf{u})$, or $\mathbf{F}(\mathbf{u}) \equiv \mathbf{u} - \mathcal{L}(\mathbf{u}) = \mathbf{0}$

- Jacobian $\partial\mathbf{F}/\partial\mathbf{u}$ generally dense $\rightarrow$ expensive to build, challenging to precondition
- Breakdown of Jacobian blocks reveals that

$$\frac{\partial \mathbf{F}}{\partial \mathbf{u}} = \begin{bmatrix} \frac{\partial \mathbf{R}_1}{\partial \mathbf{u}_1} & \mathbf{0} \\ \mathbf{0} & \frac{\partial \mathbf{R}_2}{\partial \mathbf{u}_2} \end{bmatrix}^{-1} \frac{\partial \mathbf{R}}{\partial \mathbf{u}} \equiv \mathbf{D}^{-1} \frac{\partial \mathbf{R}}{\partial \mathbf{u}}$$

Nonlinear domain decomposition preconditioning

Equivalent, fixed-point formulation of $\mathbf{R}(\mathbf{u}) = \mathbf{0}$

Find $\mathbf{u}$ so that $\mathbf{u} = \mathcal{L}(\mathbf{u})$, or $\mathbf{F}(\mathbf{u}) \equiv \mathbf{u} - \mathcal{L}(\mathbf{u}) = \mathbf{0}$

- Jacobian $\partial \mathbf{F}/\partial \mathbf{u}$ generally dense $\rightarrow$ expensive to build, challenging to precondition
- Breakdown of Jacobian blocks reveals that

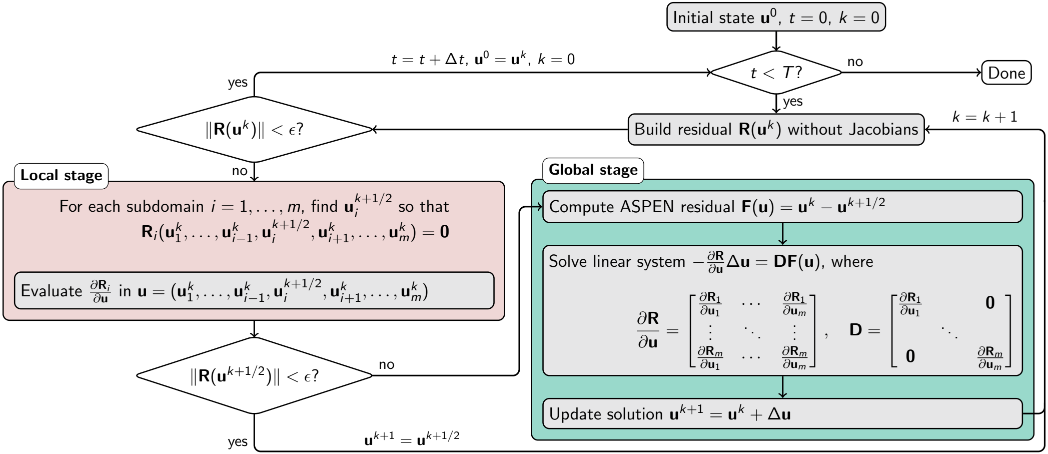
$$\frac{\partial \mathbf{F}}{\partial \mathbf{u}} = \begin{bmatrix} \frac{\partial \mathbf{R}_1}{\partial \mathbf{u}_1} & \mathbf{0} \\ \mathbf{0} & \frac{\partial \mathbf{R}_2}{\partial \mathbf{u}_2} \end{bmatrix}^{-1} \frac{\partial \mathbf{R}}{\partial \mathbf{u}} \equiv \mathbf{D}^{-1} \frac{\partial \mathbf{R}}{\partial \mathbf{u}}$$

Original problem Jacobian (almost)

$\rightarrow$ Can interpret linearized system as

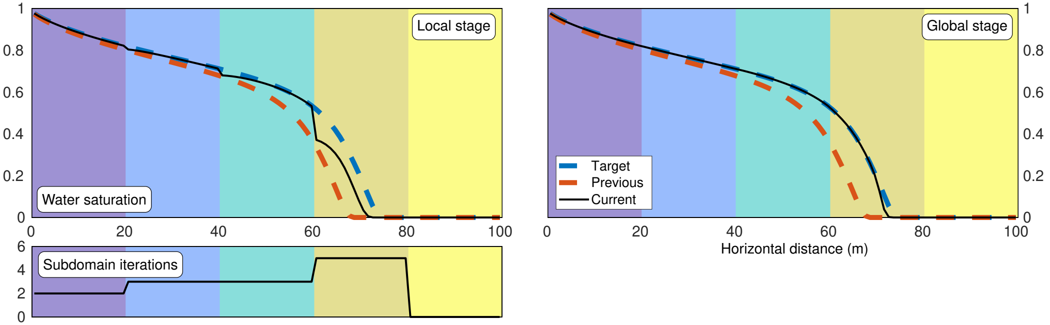
$$-\frac{\partial \mathbf{F}}{\partial \mathbf{u}} \Delta \mathbf{u} = \mathbf{F}(\mathbf{u}) \quad \Longleftrightarrow \quad -\frac{\partial \mathbf{R}}{\partial \mathbf{u}} \Delta \mathbf{u} = \mathbf{D}\mathbf{F}(\mathbf{u}).$$

- We're back on home ground – we know what preconditioners to use!

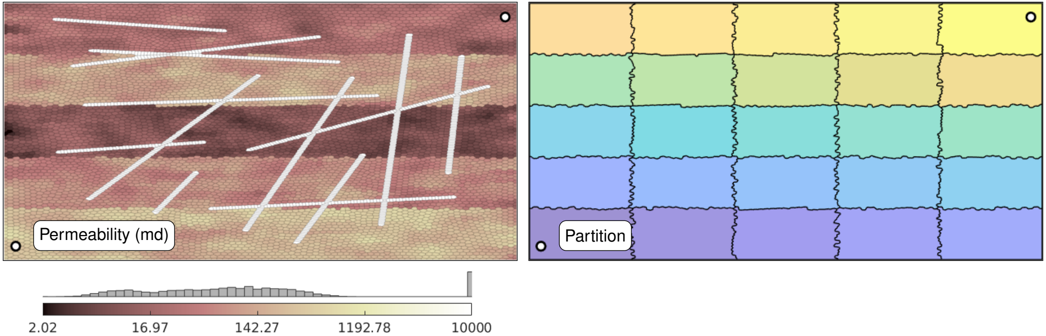


Initial state $\mathbf{u}^0$, $t = 0$, $k = 0$
$t < T$?
no
Done
yes
$t = t + \Delta t$, $\mathbf{u}^0 = \mathbf{u}^k$, $k = 0$
yes
Build residual $\mathbf{R}(\mathbf{u}^k)$ without Jacobians
$k = k + 1$
$\|\mathbf{R}(\mathbf{u}^k)\| < \epsilon$?
no
Local stage
For each subdomain $i = 1, \ldots, m$, find $\mathbf{u}_i^{k+1/2}$ so that $\mathbf{R}_i(\mathbf{u}_1^k, \ldots, \mathbf{u}_{i-1}^k, \mathbf{u}_i^{k+1/2}, \mathbf{u}_{i+1}^k, \ldots, \mathbf{u}_m^k) = \mathbf{0}$
Evaluate $\frac{\partial \mathbf{R}_i}{\partial \mathbf{u}}$ in $\mathbf{u} = (\mathbf{u}_1^k, \ldots, \mathbf{u}_{i-1}^k, \mathbf{u}_i^{k+1/2}, \mathbf{u}_{i+1}^k, \ldots, \mathbf{u}_m^k)$
$\|\mathbf{R}(\mathbf{u}^{k+1/2})\| < \epsilon$?
no
yes
$\mathbf{u}^{k+1} = \mathbf{u}^{k+1/2}$
Global stage
Compute ASPEN residual $\mathbf{F}(\mathbf{u}) = \mathbf{u}^k - \mathbf{u}^{k+1/2}$
Solve linear system $-\frac{\partial \mathbf{R}}{\partial \mathbf{u}} \Delta \mathbf{u} = \mathbf{D}\mathbf{F}(\mathbf{u})$, where
$\frac{\partial \mathbf{R}}{\partial \mathbf{u}} = \begin{bmatrix} \frac{\partial \mathbf{R}_1}{\partial \mathbf{u}_1} & \cdots & \frac{\partial \mathbf{R}_1}{\partial \mathbf{u}_m} \\ \vdots & \ddots & \vdots \\ \frac{\partial \mathbf{R}_m}{\partial \mathbf{u}_1} & \cdots & \frac{\partial \mathbf{R}_m}{\partial \mathbf{u}_m} \end{bmatrix}, \quad \mathbf{D} = \begin{bmatrix} \frac{\partial \mathbf{R}_1}{\partial \mathbf{u}_1} & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \frac{\partial \mathbf{R}_m}{\partial \mathbf{u}_m} \end{bmatrix}$
Update solution $\mathbf{u}^{k+1} = \mathbf{u}^k + \Delta \mathbf{u}$

Example 1: Buckley-Leverett displacement



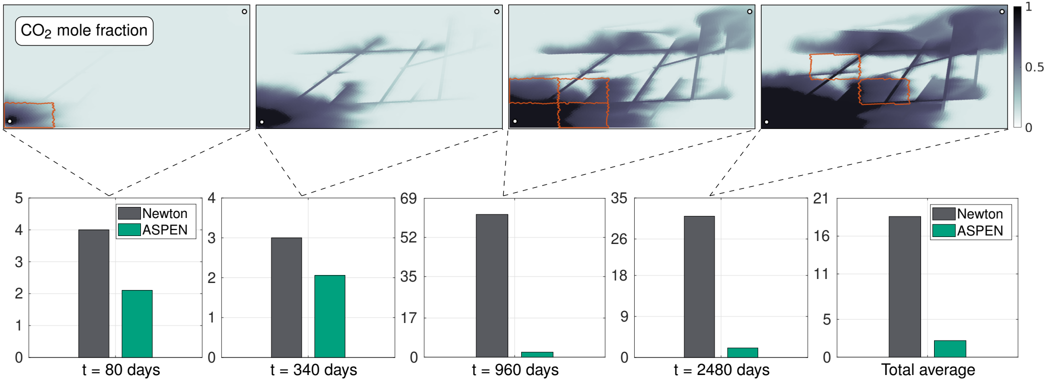
Example 2: Fractured reservoir



- 1000×500 m^2 reservoir with thirteen fractures on layered, low-perm background[1]
- Two-phase liquid-gas model with n-decane, carbon dioxide, and methane
- Simulate injection of n-decane and carbon dioxide mixture over 2555 days

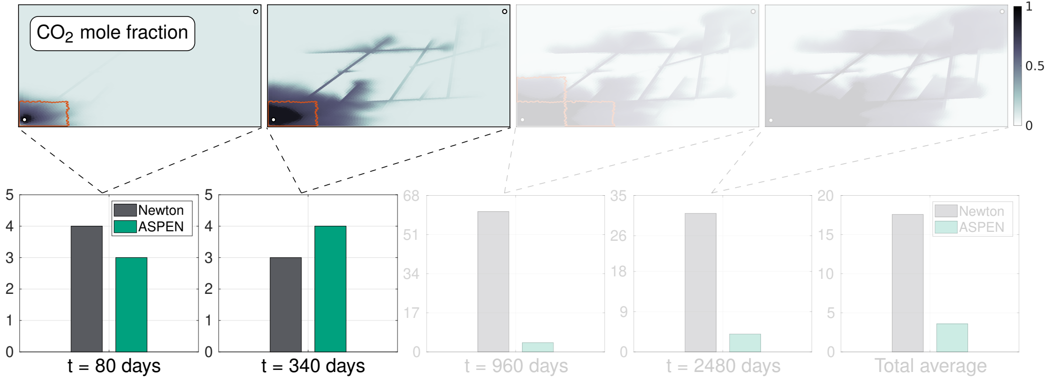
[1] Slightly modified from Møyner and Tchelepi [2018]

Example 2: Fractured reservoir



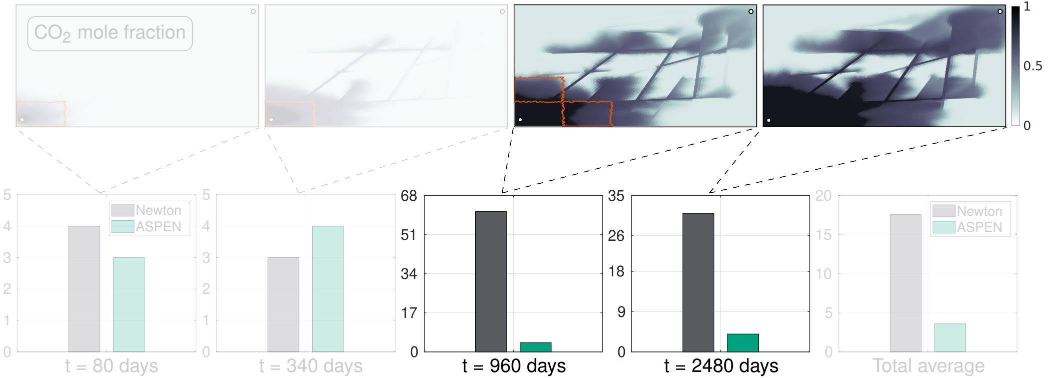
- Subdomains with more than seven iterations in total outlined in red

Example 2: Fractured reservoir



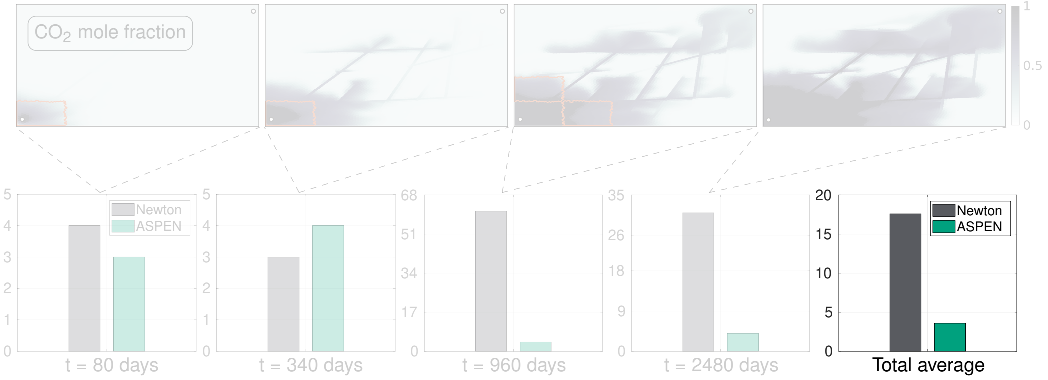
- Subdomains with more than seven iterations in total outlined in red
- Steady convergence for both solvers during the first 900 days of injection

Example 2: Fractured reservoir



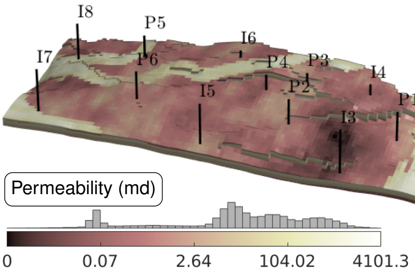
- Subdomains with more than seven iterations in total outlined in red
- Steady convergence for both solvers during the first 900 days of injection
- Global solver struggles significantly when injected fluids reach producer

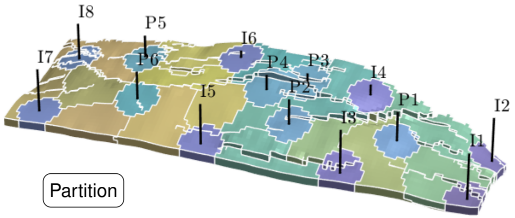
Example 2: Fractured reservoir



- Subdomains with more than seven iterations in total outlined in red
- Steady convergence for both solvers during the first 900 days of injection
- Global solver struggles significantly when injected fluids reach producer

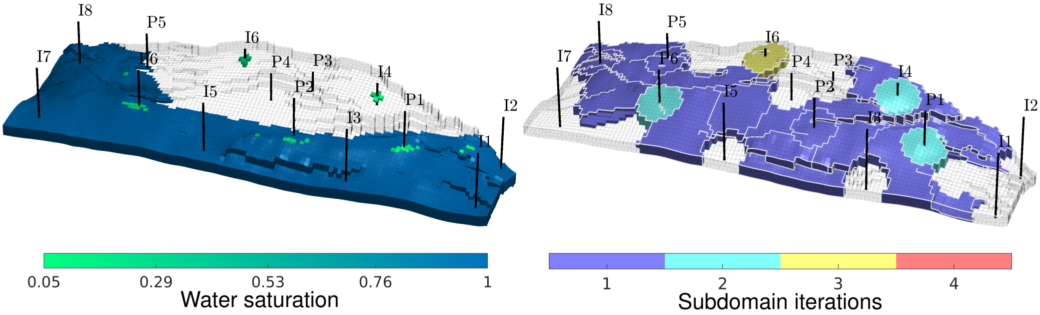
Example 4: Field-scale model (SAIGUP)



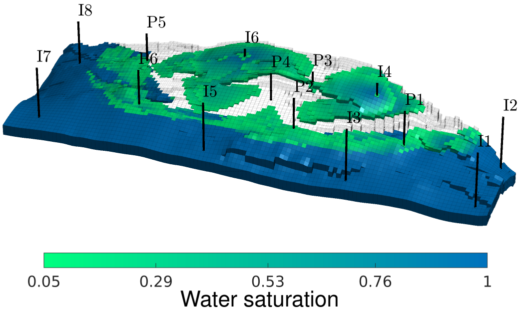


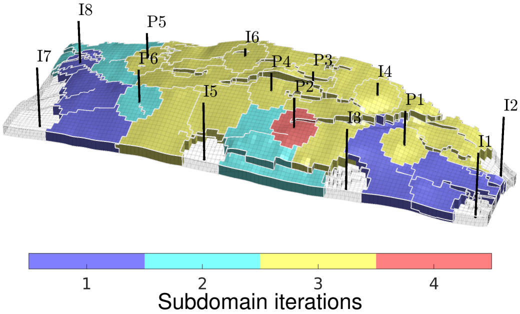
- Shallow-marine oil reservoir, modeled in the SAIGUP study (Manzocchi et al. [2008])
- Spans lateral area of $\sim 9 \times 3$ km^2, $40 \times 120 \times 20$ corner-point grid, several major faults
- Simulate 30 years of water injection with slightly compressible two-phase oil-water model
- Subdomains constructed by combining METIS partition with tubes around each well

Example 4: Field-scale model (SAIGUP)

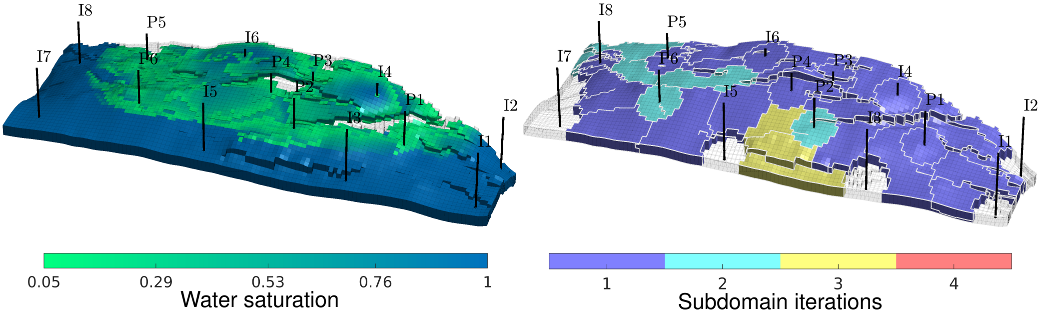


Example 4: Field-scale model (SAIGUP)

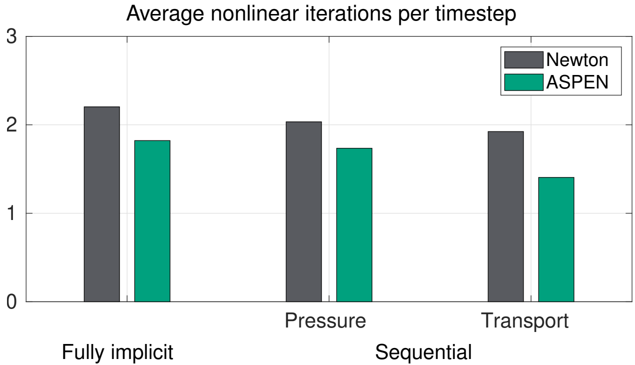




Example 4: Field-scale model (SAIGUP)

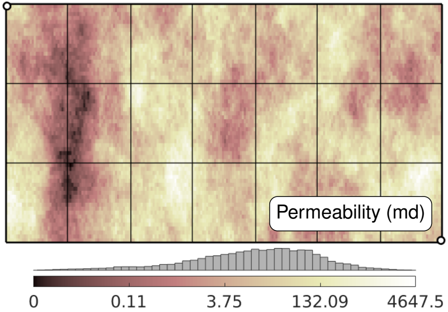


Example 4: Field-scale model (SAIGUP)



- Moderate CFL, simple fluid physics $\rightarrow$ near-optimal global nonlinear solver performance
- ASPEN nevertheless uses less iterations than global solver
  - Arguably equally important as excellent performance on challenging corner cases

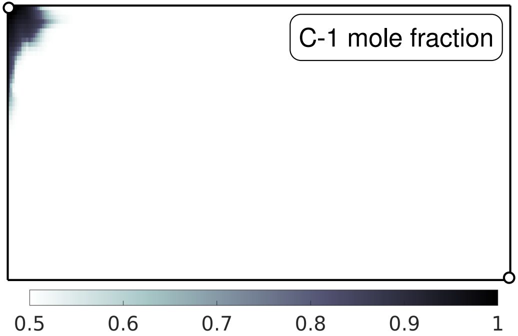
Example 5: Water-alternating gas injection (WAG)

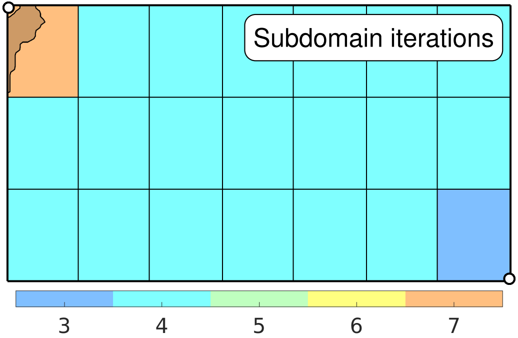


- First layer of SPE10 Model 2, three-phase, six-component fluid (C1, 3, 6, 10, 15, and 20)
- WAG injection: C1 gas for 5000 days + water for 5000 days + C1 gas for 5000 days[1]
- Simulate four setups with target timesteps of 25, 50, 100 and 200 days

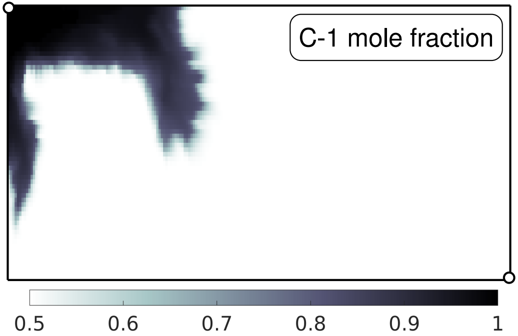
[1] Example from Moncorgé et al. [2018]

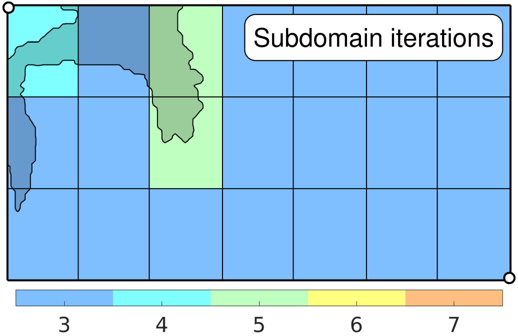
Example 5: Water-alternating gas injection (WAG)



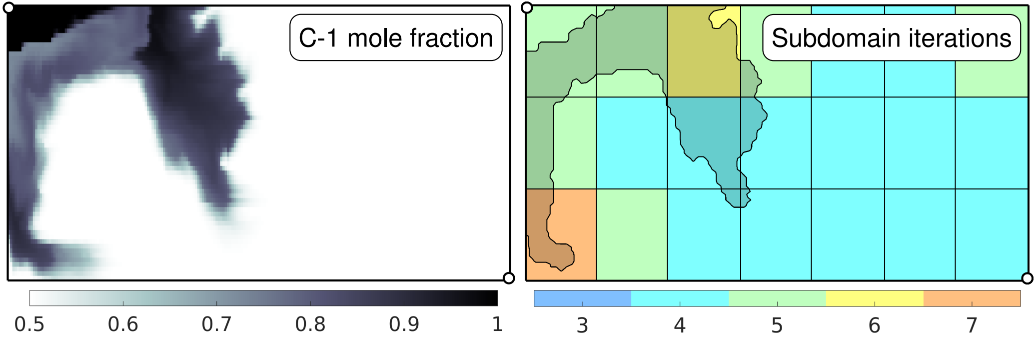


Example 5: Water-alternating gas injection (WAG)

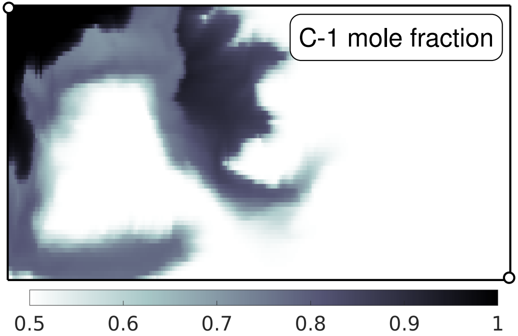


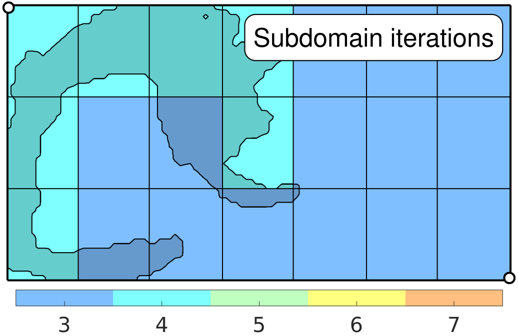


Example 5: Water-alternating gas injection (WAG)

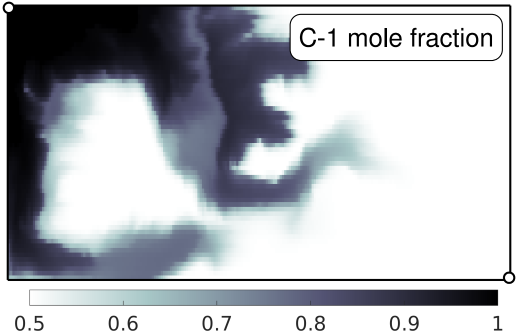


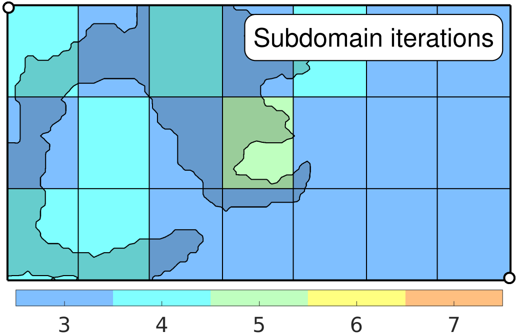
Example 5: Water-alternating gas injection (WAG)



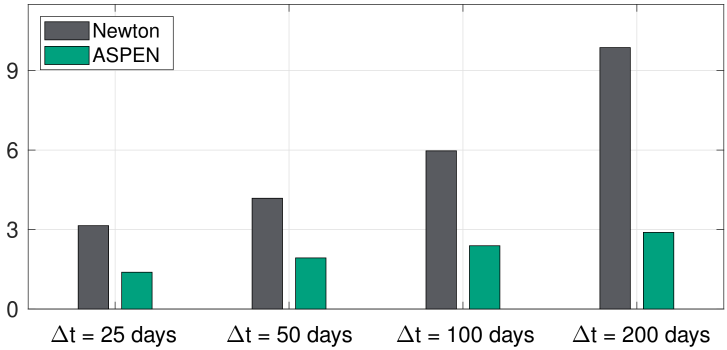


Example 5: Water-alternating gas injection (WAG)





Example 5: Water-alternating gas injection (WAG)



- Significant increase in global solver iterations with Δt – only slight increase for ASPEN
- Robustness with respect to Δt due to local control of nonlinear solution process

Concluding remarks

Conclusions

- Nonlinear domain decomposition preconditioner applicable to realistic problems
- Significant reduction in nonlinear iterations for wide range of examples
- Robust with respect to timestep size
- Reformulation enables using established iterative linear solvers

Concluding remarks

Conclusions

- Nonlinear domain decomposition preconditioner applicable to realistic problems
- Significant reduction in nonlinear iterations for wide range of examples
- Robust with respect to timestep size
- Reformulation enables using established iterative linear solvers

Future work

- Multiplicative NLDDP for transport – reordering (Natvig and Lie [2008])
- Further utilize localization
  - Local timestepping techniques (Linga et al. [2020])
  - Adaptive sequential fully-implicit methods (Møyner and Moncorgé [2019])
  - Dynamic coarsening (Klemetsdal and Lie [2020])

References

X.-C. Cai and D. E. Keyes. Nonlinearly preconditioned inexact Newton algorithms. *SIAM J. Sci. Comput.*, 24(1):183–200, 2002. URL https://doi.org/10.1137/S106482750037620X.

V. Dolean, M. J. Gander, W. Kheriji, F. Kwok, and R. Masson. Nonlinear preconditioning: How to use a nonlinear Schwarz method to precondition Newton's method. *SIAM J. Sci. Comput.*, 38(6):A3357–A3380, 2016. URL https://doi.org/10.1137/15M102887X.

P. Jenny, S. H. Lee, and H. A. Tchelepi. Adaptive fully implicit multi-scale finite-volume method for multi-phase flow and transport in heterogeneous porous media. *J. Comput. Phys.*, 217(2):627–641, 2006. ISSN 0021-9991. doi: 10.1016/j.jcp.2006.01.028. URL http://dx.doi.org/10.1016/j.jcp.2006.01.028.

J. Jiang and H. A. Tchelepi. Nonlinear acceleration of sequential fully implicit (SFI) method for coupled flow and transport in porous media. *Comput. Methods Appl. Mech. Eng.*, 352:246–275, 2019. ISSN 00457825. doi: 10.1016/j.cma.2019.04.030. URL https://doi.org/10.1016/j.cma.2019.04.030.

J. E. Killough and M. F. Wheeler. Parallel iterative linear equation solvers: An investigation of domain decomposition algorithms for reservoir simulation. In *SPE Symposium on Reservoir Simulation*, page 20, 1987. doi: 10.2118/16021-MS.

Ø. S. Klemetsdal and K.-A. Lie. Dynamic coarsening and local reordered nonlinear solvers for simulating transport in porous media. *SPE J.*, (January):1–20, 2020. doi: 10.2118/201089-PA.

J. Li, Z. Y. Wong, P. Tomin, and H. Tchelepi. Sequential implicit Newton method for coupled multi-segment wells. In *SPE Reservoir Simululation Conference*, 2019. doi: 10.2118/193833-MS.

K.-A. Lie, O. Møyner, J. R. Natvig, A. Kozlova, K. Bratvedt, S. Watanabe, and Z. Li. Successful application of multiscale methods in a real reservoir simulator environment. *Comput. Geosci.*, 21(5-6):981–998, 2017. ISSN 15731499. doi: 10.1007/s10596-017-9627-2.

G. Linga, O. Møyner, H. Møll, A. Moncorgé, and K.-A. Lie. An implicit local time-stepping method based on cell reordering for multiphase flow in porous media. *J. Comput. Phys.: X*, 6:100051, 2020. doi: 10.1016/j.jcpx.2020.100051.

L. Liu, D. E. Keyes, and S. Sun. Fully implicit two-phase reservoir simulation with the additive Schwarz preconditioned inexact Newton method. In *SPE Reservoir Characterization and Simulation Conference and Exhibition, 16-18 September, Abu Dhabi, UAE*. Society of Petroleum Engineers (SPE), 2013. URL https://doi.org/10.2118/166062-MS.

T. Manzocchi, J. N. Carter, A. Skorstad, B. Fjellvoll, K. D. Stephen, J. A. Howell, J. D. Matthews, J. J. Walsh, M. Nepveu, C. Bos, J. Cole, P. Egberts, S. Flint, C. Hern, L. Holden, H. Hovland, H. Jackson, O. Kolbjørnsen, A. MacDonald, P. A. R. Nell, K. Onyeagoro, J. Strand, A. R. Syversveen, A. Tchistiakov, C. Yang, G. Yielding, and R. W. Zimmerman. Sensitivity of the impact of geological uncertainty on production from faulted and unfaulted shallow-marine oil reservoirs: objectives and methods. *Pet. Geosci.*, 14(1):3–15, 2008. doi: 10.1144/1354-079307-790.

A. Moncorgé, H. A. Tchelepi, and P. Jenny. Sequential fully implicit formulation for compositional simulation using natural variables. *Journal of Computational Physics*, 371:690–711, 2018. ISSN 0021-9991. doi: 10.1016/j.jcp.2018.05.048. URL https://doi.org/10.1016/j.jcp.2018.05.048.

O. Møyner and K.-A. Lie. A multiscale restriction-smoothed basis method for compressible black-oil models. *SPE J.*, 21(06):2079–2096, dec 2016. doi: 10.2118/173265-PA. URL http://dx.doi.org/10.2118/173265-PA.

Extra: Nonlinear domain decomposition preconditioning

Equivalent, fixed-point formulation of $\mathbf{R}(\mathbf{u}) = \mathbf{0}$

Find $\mathbf{u}$ so that $\mathbf{u} = \mathcal{L}(\mathbf{u})$, or $\mathbf{F}(\mathbf{u}) \equiv \mathbf{u} - \mathcal{L}(\mathbf{u}) = \mathbf{0}$

Natural extension to m subdomains $\mathbf{u} = (\mathbf{u}_1, \ldots, \mathbf{u}_m)$, $\big(\mathbf{R}_1(\mathbf{u}), \ldots, \mathbf{R}_m(\mathbf{u})\big)$

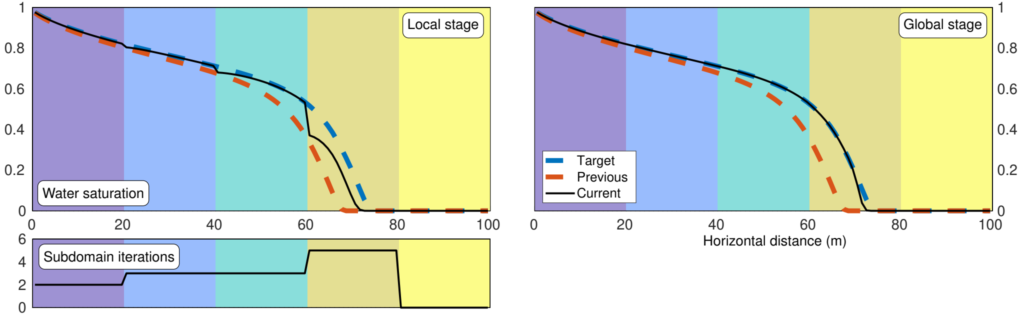
- Corresponding solution operators $\mathcal{L}_1, \ldots, \mathcal{L}_m$

$$\mathbf{R}_i\big(\mathbf{u}_1, \ldots, \mathbf{u}_{i-1}, \mathcal{L}_i(\mathbf{u}), \mathbf{u}_{i+1}, \ldots, \mathbf{u}_m\big) = \mathbf{0}$$

- As for the two-subdomain-case, we get

$$\frac{\partial \mathcal{L}_i}{\partial \mathbf{u}} = -\left(\frac{\partial \mathbf{R}_i}{\partial \mathbf{u}_i}\right)^{-1} \left(\sum_{j=1, j\neq i}^{m} \frac{\partial \mathbf{R}_i}{\partial \mathbf{u}_j} \frac{\partial \mathbf{u}_j}{\partial \mathbf{u}}\right).$$

Extra: Buckley-Leverett displacement (Example 1)

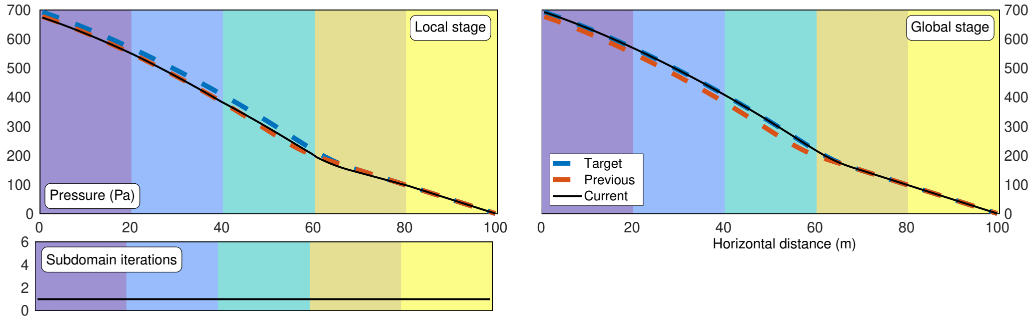


Horizontal 1D channel, CFL $= 1$, quadratic relative permeabilities, equal viscosities

Transport subproblem

- Visual kinks at subdomain boundaries after local solve
- Almost converged after additional global solve

Extra: Buckley-Leverett displacement (Example 1)



Horizontal 1D channel, CFL $= 1$, quadratic relative permeabilities, equal viscosities

Pressure subproblem

- Local solve far from target solution
- Global solve effectively resolves long-range interactions

Extra: Buckley-Leverett displacement (Example 1)

Nonlinear iterations per timestep