A multiscale restriction-smoothed basis method for high contrast porous media represented on unstructured grids

Olav Møyner and Knut–Andreas Lie

SINTEF Digital, Department of Mathematics & Cybernetics, Oslo, Norway

IPAM OILWS1 - Multiphysics, Multiscale, and Coupled Problems in Subsurface Physics, April 4 2017
Multiscale idea formulated as discrete operators

\[ Ax = q \]

Initial fine-scale system, incorporating all details of geological model

\[ x = Px_c \]
\[ P = \text{basis}(A) \]
\[ A_c = \text{RAP} \]
\[ q_c = \text{R} q \]

Multiscale expansion:
generate basis functions, restrict fine-scale system and right-hand side

\[ x_c = A_c^{-1} q_c \]
\[ x \approx Px_c \]

Solve reduced system, prolongate to obtain approximate pressure

Lee, Lunati, Nordbotten, Tchelepi, Zhou, ... (2008)
From Poisson’s equation to reservoir simulation

Flow physics

MsFV, MsMFE, 2003

\[-\nabla (K \nabla p) = q\]

Geology

0 = \partial_t (\phi b_w S_w) + \nabla \cdot (b_w \vec{v}_w) - b_w q_w
0 = \partial_t (\phi b_o S_o) + \nabla \cdot (b_o \vec{v}_o) - b_o q_o
0 = \partial_t [\phi (b_g S_b + b_o r_s S_o)] + \nabla \cdot (b_g \vec{v}_g)
+ \nabla \cdot (b_o r_s \vec{v}_o) - b_g q_g - b_o r_s q_o
Why is this challenging?

- Geological models: complex unstructured grids having many obscure challenges
- Flow models: system of highly nonlinear parabolic PDEs with elliptic and hyperbolic sub-character
- Well models: analytic sub-models, strong impact on flow

Challenges:
- Industry standard: corner-point / stratigraphic grids
- Grid topology is unstructured
- Geometry: deviates from box shape, high aspect ratios, many faces/neighbors, small faces, . . .
- As a general rule: coarse blocks will be unstructured
- Coarse blocks will have strange shapes, many special cases to be handled
- Coarse partition should adapt to features relevant to flow: petrophysical properties, faults, flow direction, wells, . . .
Qualitatively correct solution $\rightarrow$ small fine-scale residual

upscaling

accuracy

computational cost

fine scale solution
Qualitatively correct solution $\rightarrow$ small fine-scale residual

- Qualitatively correct solution
- small fine-scale residual
- computational cost
- upscaling
- multiscale
- fine scale solution
- accuracy
Qualitatively correct solution $\rightarrow$ small fine-scale residual

Residual iteration:

$$p^* = p^\nu + S(q - Ap^\nu)$$
$$p^{\nu+1} = p^* + A_{ms}^{-1}(q - Ap^*)$$

$S$ is some inexpensive smoother, e.g., ILU(0)

Hajibeygi, Jenny, Tchelepi, Wang, ... (2008–2015)
The multiscale finite-volume (MsFV) method

Developed by the INTERSECT research alliance (Chevron, Schlumberger, ++)

In 2012: extensive research over the past decade, more than 40 papers by Jenny, Lee, Tchelepi, Lunati, Hajibeygi, etc:

- correction functions to handle non-elliptic features
- extension to compressible flow
- adaptive updating of basis functions (and transport equations)
- iterative formulation with smoothers (Jacobi, GMRES, …)
- algebraic formulation

However…

- Monotonicity issues requires many iterations for strong heterogeneity
- Method only applied to Cartesian models with conceptual faults

Our focus: Extension to unstructured grids with realistic geology
MsFE/MsFV: prolongation operator in more detail

\[ \nabla_t \cdot (K \nabla_t p) = 0 \]

\[ \Phi = 1 \]

\[ \Phi = 0 \]

\[ \nabla \cdot (K \nabla p) = 0 \]

MsFE/MsFV: prolongation operator in more detail

\[ \nabla_t \cdot (K \nabla_t p) = 0 \]

\[ \Phi = 1 \]

\[ \Phi = 0 \]

\[ \nabla \cdot (K \nabla p) = 0 \]

MsFE/MsFV: prolongation operator in more detail

\[ \nabla_t \cdot (K \nabla_t p) = 0 \]

\[ \Phi = 1 \]
\[ \Phi = 0 \]

MsFE/MsFV: prolongation operator in more detail

\[ \nabla_t \cdot (K \nabla_t p) = 0 \]
\[ \nabla \cdot (K \nabla p) = 0 \]

\[ \Phi = 1 \]
\[ \Phi = 0 \]

MsFE/MsFV: prolongation operator in more detail
MsFV for unstructured grids (Møyner & Lie, 2013)

Algorithm for generating admissible primal–dual partitions on general grids

- automated on rectilinear, curvilinear, triangular, and Voronoi grids
- semi-automated on corner-point grids and grids with non-matching faces
Problems encountered: permeability contrasts

Automated algorithms will generally give:

- Dual block centers in low-permeable regions
- Dual edges crossing strong permeability contrasts (twice)
- Large number of cells categorized as edges

→ nonmonotone multipoint stencil for coarse-scale equations
→ poor decoupling, does not reproduce linear flow
Rethinking the basis functions: Requirements

What do we want from numerical basis functions $P$?

- Partition of unity to represent constant fields
  
  $\sum_j P_{ij} = 1 \rightarrow$ Exact interpolation of constant modes

- Algebraically smooth:
  
  Minimize $\|AP\|_1 \rightarrow AP_{pc} \approx Ap$ locally.

- Localization:
  
  Coarse system $A_c = RAP$ becomes dense as support of basis functions grow
Basis functions require a coarse grid and a support region

- Support region: logical indices, topological search, distance measures,..
- Region constructed using triangulation of nodal coarse neighbors, resulting in a multipoint stencil on the coarse scale
- Avoid solving reduced flow problem along perimeter
- Main point: simple to implement in 3D for fully unstructured meshes
MsRSB: restricted, smoothed basis functions

Ideally, operators are both smooth and local

1. Start with constant functions on primal grid
2. Apply Jacobi-like iterations as in algebraic multigrid methods (Vanek et al),

\[ P^{n+1} = P^n - \omega D^{-1}(AP) \]

3. Restrict each function to its support region
4. Repeat steps 2 and 3 until convergence
MsRSB: restricted, smoothed basis functions

Ideally, operators are both *smooth* and *local*

1. Start with constant functions on primal grid
2. Apply Jacobi-like iterations as in algebraic multigrid methods (Vanek et al),

\[ P^{n+1} = P^n - \omega D^{-1}(AP) \]

3. Restrict each function to its support region
4. Repeat steps 2 and 3 until convergence

---

Permeability and grid

Initial constant basis
After one pass
After 10 passes
Converged \((n \approx 100)\)
MsRSB: computing basis functions

Define preliminary update by Jacobi relaxation,

\[ \hat{d}_j = -\omega D^{-1} A P_j^n. \]

Modify the update according to cell category,

\[ d_{ij} = \begin{cases} 
\hat{d}_{ij} - P_{ij}^n \sum_{k \in H_i} \hat{d}_{ik} & \text{if } i \in I_j, i \in G, \\
\hat{d}_{ij} & \text{if } i \in I_j, i \notin G, \\
0 & \text{if } i \notin I_j. 
\end{cases} \]

Finally, apply the update and proceed to next iteration

\[ P_{ij}^{n+1} = P_{ij}^n + d_{ij}. \]

- Jacobi iteration ensures algebraic smoothness
- Limited support by construction
- Modified update for partition of unity
Examples: Single-phase flow
SPE10 full model

<table>
<thead>
<tr>
<th>Error</th>
<th>Grid</th>
<th>$p \ (L_2)$</th>
<th>$p \ (L_\infty)$</th>
<th>$v \ (L_2)$</th>
<th>$v \ (L_\infty)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>MsFV</td>
<td>$6 \times 11 \times 17$</td>
<td>3.580</td>
<td>128.461</td>
<td>2.288</td>
<td>11.957</td>
</tr>
<tr>
<td>MsRSB</td>
<td>$6 \times 11 \times 17$</td>
<td>0.039</td>
<td>0.309</td>
<td>0.397</td>
<td>0.487</td>
</tr>
</tbody>
</table>
SPE10 full model

Horizontal permeability

MsRSB

Reference solution

MsFV, $p \notin [0, 1]$

<table>
<thead>
<tr>
<th>Error</th>
<th>Grid</th>
<th>$p (L_2)$</th>
<th>$p (L_\infty)$</th>
<th>$v (L_2)$</th>
<th>$v (L_\infty)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>MsFV</td>
<td>$6 \times 11 \times 17$</td>
<td>3.580</td>
<td>128.461</td>
<td>2.288</td>
<td>11.957</td>
</tr>
<tr>
<td>MsRSB</td>
<td>$6 \times 11 \times 17$</td>
<td>0.039</td>
<td>0.309</td>
<td>0.397</td>
<td>0.487</td>
</tr>
</tbody>
</table>
Example: unstructured PEBI grid

Porosity and grid

Permability from SPE 10, Layer 35

Detailed view of refinement

- Unstructured grid designed to minimize grid orientation effects
- Two embedded radial grids near wells
- Fine grid adapts to faults
- The faults are sealed, i.e. allow no fluid flow through
Example: unstructured PEBI grid

Two-step preconditioner, ILU(0) as 2nd stage, Richardson iterations
Example: unstructured PEBI grid

Adapted grid gives better prolongation operator → faster initial convergence

Metis grid has more block connections → faster two-level convergence

Slow convergence: ILU(0)

Two-step preconditioner, ILU(0) as 2nd stage, Richardson iterations
Example: Gullfaks field

- Early field model of a giant reservoir from the Norwegian North Sea
- 216 000 cells with a large number of faults and eroded layers
- Very challenging anisotropic permeability and grid
- Model includes cells with nearly 40 faces
- Synthetic well configuration with four vertical wells
MsRSB: Gullfaks field

- First coarsening strategy:
  Uniform blocks, split over faults

- Second coarsening strategy:
  Use Metis with same number of DoF

<table>
<thead>
<tr>
<th>Grid type</th>
<th>DoF</th>
<th>$p(L_2)$</th>
<th>$p(L_\infty)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$15 \times 15 \times 20$</td>
<td>416</td>
<td>0.032</td>
<td>0.102</td>
</tr>
<tr>
<td>Metis</td>
<td>416</td>
<td>0.032</td>
<td>0.100</td>
</tr>
<tr>
<td>$10 \times 10 \times 10$</td>
<td>1028</td>
<td>0.028</td>
<td>0.597</td>
</tr>
<tr>
<td>Metis</td>
<td>1028</td>
<td>0.015</td>
<td>0.112</td>
</tr>
</tbody>
</table>

Fine scale (216 000 DoF)

MsRSB (416 DoF)
MsRSB: Gullfaks field

![Graph showing the comparison between Control volume and Finite element methods with different degrees of freedom (DoF): 10 DoF, 100 DoF, 250 DoF, 500 DoF, and 1000 DoF. The y-axis represents the magnitude of a variable, ranging from $10^{-6}$ to $10^{-2}$, and the x-axis represents a range from 20 to 100.]
Examples: Compressible, multi-phase flow
The black-oil equations on residual form:

\[
\mathcal{R}_w = \frac{1}{\Delta t} \left[ (\phi b_w S_w)^{n+1} - (\phi b_w S_w)^n \right] + \nabla \cdot (b_w \vec{v}_w)^{n+1} - (b_w q_w)^{n+1} = 0, \\
\mathcal{R}_o = \frac{1}{\Delta t} \left[ (\phi b_o S_o)^{n+1} - (\phi b_o S_o)^n \right] + \nabla \cdot (b_o \vec{v}_o)^{n+1} - (b_o q_o)^{n+1} = 0, \\
\mathcal{R}_g = \frac{1}{\Delta t} \left[ (\phi b_g S_g + \phi r_{so} b_o S_o)^{n+1} - (\phi b_g S_g + \phi r_{so} b_o S_o)^n \right] \\
+ \nabla \cdot (b_g \vec{v}_g + b_o r_{so} \vec{v}_o)^{n+1} - (b_g q_g + b_o r_{so} q_o)^{n+1} = 0.
\]

Pressure equation found by eliminating saturation values at the next time step,

\[
\mathcal{R}_p = \frac{\mathcal{R}_w}{b_w^{n+1}} + \left[ \frac{1}{b_o^{n+1}} - \frac{r_{so}^{n+1}}{b_g^{n+1}} \right] \mathcal{R}_o + \frac{\mathcal{R}_g}{b_g^{n+1}} = 0,
\]

Transport step: fractional flow formulation with standard two-point, upstream-mobility weighting.
Sequentially-implicit solution strategy

Set \( p_{0}^{n+1} = p^{n}, \ i = 0 \)

Assemble \( J_{p} \) and \( r_{p} \)

Solve \(- J_{p} \Delta p = r_{p} \) to rough tolerance \( \epsilon_{p} \) using MS

Update pressure, \( p_{i+1}^{n+1} = p_{i}^{n+1} + \Delta p \)

Converged? \( i = i + 1 \)

Set \( s_{0}^{n+1} = s^{n}, \ j = 0 \)

Assemble \( J_{s} \) and \( r_{s} \)

Solve \(- J_{s} \Delta s = r_{s} \)

Update saturation, \( s_{i+1}^{n+1} = s_{i}^{n+1} + \Delta s \)

Converged? \( i = i + 1 \)

Compute \( v_{T} = \sum_{\alpha} v_{\alpha} \)

Reconstruct velocity field

Converged? \( i = i + 1 \)

Next timestep
Iterated sequential solver:
- 0.001 pressure increment tolerance
- $10^{-6}$ tolerance for algebraic multigrid

Iterated multiscale solver:
- 0.005 pressure increment tolerance
- $10^{-2}$ tolerance for MsRSB solver

Approximate MsRSB solver is ten times faster than baseline sequential
Example: realistic waterflooding

Watt Field: water flooding
415 711 active cells, three rock types
7 injectors, 15 horizontal producers

Injector BHP:
Producer BHP:

Møyner & Lie, SPE J. (2016)
Example: realistic waterflooding

Thin solid: fine-scale solution
Thick dashed: multiscale solution
Multiscale: 800 blocks, tolerance 0.05
Solver speedup: 9×
Example: 3-phase flow

- Synthetic model with fluid model based on SPE1 benchmark
- Gas is injected at constant rate into a undersaturated reservoir
- Producer at fixed bottom hole pressure
- Highly sensitive to pressure approximation

Gas saturation at breakthrough

Møyner & Lie, SPE J. (2016)
Example: 3-phase flow
Example: INTERSECT Prototype on Gullfaks

- Giant North Sea field, started production in 1986
- Mainly water injection, but also gas and water-alternating-gas in some areas
- Coarse $80 \times 100 \times 19$ simulation model with real history (3-phase black oil)
- MsRSB basis functions in Intersect R&P Multiscale simulator

Recent developments: Compositional flow
Compositional flow: Governing equations

Aqueous phase,

\[ \mathcal{R}_w = \partial_t (\phi \rho_w S_w) + \nabla \cdot (\rho_w \vec{v}_w) - \rho_w q_w = 0. \]

Component \( i \),

\[ \mathcal{R}_i = \partial_t (\phi [\rho_l S_l X_i + \rho_v S_v Y_i]) + \nabla \cdot (\rho_l X_i \vec{v}_l + \rho_v Y_i \vec{v}_v) - \rho_l X_i q_l - \rho_v Y_i q_v = 0. \]

- Hydrocarbons assumed to exist in vapor/liquid – not in aqueous
- Generalized cubic equation-of-state (Peng-Robinson in examples)
- Lohrenz-Bray-Clark viscosity correlation

Møyner & Tchelepi, SPE RSC (2017)
Flash equations,

\[ f_{il}(p, T, x_1, ..., x_n, Z_l) - f_{iv}(p, T, y_1, ..., y_n, Z_v) = 0, \text{ for } i \in \{1, ..., N\} \]
\[ z_i - Lx_i - (1 - L)y_i = 0, \text{ for } i \in \{1, ..., N\} \]
\[ \sum_{i=1}^{N} x_i - y_i = 0. \]

- Applies in cells with two hydrocarbon phases.
- Overall composition: Flash to be solved at every iteration
Sequential total mass scheme: Pressure

Scheme suggested by Hajibeygi & Tchelepi (SPE J, 2014), where pressure is found by total mass balance,

\[ R_p = \frac{\phi}{\Delta t} \left[ R_{t}^{n+1} - R_{t}^{n} \right] + \nabla \cdot \vec{V}_t - Q_t = 0. \]

from unweighted sum over component equations.

Define total density, total mass fluxes

\[ R_t = \sum_{\beta=w,l,v} \rho_{\beta} S_{\beta}, \quad \vec{V}_t = \sum_{\beta=w,l,v} \rho_{\beta} \vec{u}_{\beta}, \quad Q_t = \sum_{\beta=w,l,v} \rho_{\beta} q_{\beta}. \]

Total mass does not change during transport – reasonable?
Sequential-implicit total mass scheme: Transport

Transport equation for hydrocarbon component $i$,

$$R_{ti} = \frac{\phi}{\Delta t} \left[ (X_i R_l)^{n+1} + (Y_i R_v)^{n+1} - (X_i R_l)^n - (Y_i R_v)^n \right]$$

$$+ \nabla \cdot \left( X_i \vec{V}_l + Y_i \vec{V}_v \right) - X_i Q_l - Y_i Q_v = 0.$$ 

Where we have used fixed masses to obtain,

$$R_\alpha = \frac{\rho_\alpha S_\alpha}{\sum_{\beta=w,l,v} \rho_\beta S_\beta} R_t,$$

$$Q_\alpha = \frac{\rho_\alpha \lambda_\alpha}{\sum_{\beta=w,l,v} \rho_\beta \lambda_\beta} Q_t,$$

$$\vec{V}_\alpha = \frac{\lambda_\alpha \rho_\alpha}{\sum_{\beta=w,l,v} \lambda_\beta \rho_\beta} (\vec{V}_t + K \sum_{\beta=w,l,v} \rho_\beta \lambda_\beta (\rho_\beta - \rho_\alpha) \vec{g} \nabla z).$$
Sequential total volume scheme: Pressure

Pressure equation as total volume balance

Defined weighted sum (see Watts, 1986 or review by Coats, 2000)

\[ R_p = \sum_{i=1}^{N} w_i R_i \]

Weights (partial component volumes) chosen such that accumulation

\[ A_P = \frac{\partial}{\partial t} \left[ \sum_{i=1}^{N} w_i (\rho_l S_l X_i + \rho_v S_v Y_i) \right] \]

has zero derivatives w.r.t. all primary variables except pressure.
Sequential-implicit total volume scheme: Transport

Transport equation for hydrocarbon component $i$,

$$
R_{ti} = \partial_t (\phi [\rho_l S_l X_i + \rho_v S_v Y_i]) + \nabla \cdot (\rho_l X_i \vec{v}_l + \rho_v Y_i \vec{v}_v) - \rho_l X_i q_l - \rho_v Y_i q_v = 0.
$$

Phase velocity by fractional flow, keeping total velocity fixed

$$
\vec{v}_\alpha = \frac{\lambda_\alpha}{\sum_{\beta=w,l,v} \lambda_\beta} (\vec{v}_t + K \sum_{\beta=w,l,v} \lambda_\beta (\rho_\beta - \rho_\alpha) \vec{g} \nabla z)
$$
CO₂ injection: Layer of SPE 10 Model 2

- Six component fluid from Mallison et al. (2005)
- Initial concentration
  - \(\text{N}_2 + \text{CH}_4\): 0.463
  - \(\text{CO}_2\): 0.01640
  - \(\text{C}_{2-5}\): 0.20520
  - \(\text{C}_{6-13}\): 0.19108
  - \(\text{C}_{14-24}\): 0.08113
  - \(\text{C}_{25-80}\): 0.04319
- \(\text{CO}_2\) and water injected at opposite corners
CO$_2$ injection: Layer of SPE 10 Model 2

Sequential saturation at 2/3 PVI

MsRSB saturation at 2/3 PVI
CO$_2$ injection: Layer of SPE 10 Model 2

Sequential saturation at 2/3 PVI

$|S_{ms} - S|$ at 2/3 PVI
CO$_2$ injection: Layer of SPE 10 Model 2

CO$_2$ concentration at 2/3 PVI
$\text{CO}_2$ concentration at $2/3$ PVI

$|z_{ms} - z|$ at $2/3$ PVI
**CO₂ injection: Layer of SPE 10 Model 2**

![Graph of CO₂ injection: Layer of SPE 10 Model 2](image)

- **Component rates:**
  - Sequential
  - MsRSB
  - Fully-implicit

- **Component mass [kg/day]:**
  - N₂/CH₄
  - CO₂
  - C₂⁻⁵
  - C₈⁻¹₃
  - C₁₄⁻²₄
  - C₂₅⁻₈₀

- **Time [days]:**
  0 200 400 600 800 1000 1200 1400 1600 1800 2000
CO$_2$ injection: Layer of SPE 10 Model 2

Gas and oil production

Surface rates [m$^3$/day] vs. Time [days]
Example: Norne field

- Subset of Norne field model
- Synthetic wells injecting \( N_2 \)
- Reservoir contains Methane, n-Pentane and n-Decane
- Low field pressure makes phase behavior sensitive
- Model contains faults, anisotropy, pinched cells, ...
- 40,000 fine cells, 200 coarse blocks

![Permeability (millidarcy)]
Example: Norne field

Field pressure

Gas saturation
Example: Norne field
Example: Norne field

Cumulative oil production

Oil production
Example: Norne field

Cumulative gas production

Gas production
Example: Norne field

Field pressure

Gas production
Example: Nitrogen test

- SPE 10 model with very thin layer
- Use same fluid as for Norne field
- Constructed to produce oscillations in mass-scheme
- Compare total mass and total volume schemes
Example: Nitrogen test

Fully implicit

Cumulative gas production

Gas production
Example: Nitrogen test

**Total mass splitting**

**Cumulative gas production**

**Gas production**
Example: Nitrogen test

Total volume splitting

Cumulative gas production

Gas production
Example: Nitrogen test
Recent developments: Feature-enrichment
Room for improvements

- Slow convergence in certain cases with strong contrasts and long correlation lengths
- Desire to adapt coarse grid to geological features
- Want improved resolution near wells
- Flux reconstruction for transport can be expensive

Previous work:
- generalized multiscale element methods (Efendiev et al)
- hybrid finite-volume/Galerkin method (Cortinovis and Jenny)
New idea: multiple multiscale operators

Assume $N$ prolongation operators $P^1, \ldots, P^N$ that may come from different coarse grids and support regions, or different multiscale methods (MsRSB, MsFV, ...)

Likewise, there are $N$ restriction operators $R^1, \ldots, R^N$
New idea: multiple multiscale operators

Assume $N$ prolongation operators $P^1, \ldots, P^N$ that may come from different coarse grids and support regions, or different multiscale methods (MsRSB, MsFV, \ldots)

Likewise, there are $N$ restriction operators $R^1, \ldots, R^N$

Multiplicative multistep method:

$$p^* = p^{k+(\ell-1)/N} + S(q - A p^{k+(\ell-1)/N})$$

$$p^{k+\ell/N} = p^* + P^\ell \left( R^\ell A P^\ell \right)^{-1} R^\ell (q - A p^*)$$

Example setup: $P^1$ is general and covers domain evenly, whereas $P^2, \ldots, P^N$ are feature specific

Lie, Møyner, Natvig, SPE RSC 2017
Minimal assumptions on operators

1. $P^\ell$ and $R^\ell$ are constructed from a non-overlapping partition of the fine grid. Each column $j$ in $P^\ell$ is called a *basis function* and is associated with a coarse grid block $\overline{\Omega}^\ell_j$.

2. The support $S^\ell_j$ of each basis function is compact and contains $\overline{\Omega}^\ell_j$.

3. The columns of $P^\ell$ form a partition of unity, i.e., each row in $P^\ell$ has unit row sum.
Examples of partition types

- Rectilinear or structured subdivisions
- Adapting to facies, rock types, saturation regions, etc
- Partitions adapting to faults, fractures, . . .
- From block-structured grids, LGR, . . .
- Unstructured graph-based partitions
- Amalgamations based on indicators
- Adapting dynamically to flow
- Separating near-well and far-field
- . . .
Examples of partition types

- Rectilinear or structured subdivisions
- Adapting to facies, rock types, saturation regions, etc
- Partitions adapting to faults, fractures, ...
- From block-structured grids, LGR, ...
- Unstructured graph-based partitions
- Amalgamations based on indicators
- Adapting dynamically to flow
- Separating near-well and far-field
- ...

![Partition Types Diagram]
Examples of partition types

- Rectilinear or structured subdivisions
- Adapting to facies, rock types, saturation regions, etc
- Partitions adapting to faults, fractures, . . .
- From block-structured grids, LGR, . . .
- Unstructured graph-based partitions
- Amalgamations based on indicators
- Adapting dynamically to flow
- Separating near-well and far-field
- . . .
Examples of partition types

- Rectilinear or structured subdivisions
- Adapting to facies, rock types, saturation regions, etc
- Partitions adapting to faults, fractures, ...
- From block-structured grids, LGR, ...
- Unstructured graph-based partitions
- Amalgamations based on indicators
- Adapting dynamically to flow
- Separating near-well and far-field
- ...
Examples of partition types

- Rectilinear or structured subdivisions
- Adapting to facies, rock types, saturation regions, etc
- Partitions adapting to faults, fractures, ...
- From block-structured grids, LGR, ...
- Unstructured graph-based partitions
- Amalgamations based on indicators
- Adapting dynamically to flow
- Separating near-well and far-field
- ...
Examples of partition types

- Rectilinear or structured subdivisions
- Adapting to facies, rock types, saturation regions, etc
- Partitions adapting to faults, fractures, ...
- From block-structured grids, LGR, ...
- Unstructured graph-based partitions
- Amalgamations based on indicators
- Adapting dynamically to flow
- Separating near-well and far-field
- ...
Examples of partition types

- Rectilinear or structured subdivisions
- Adapting to facies, rock types, saturation regions, etc
- Partitions adapting to faults, fractures,…
- From block-structured grids, LGR,…
- Unstructured graph-based partitions
- Amalgamations based on indicators
- Adapting dynamically to flow
- Separating near-well and far-field
- …
Examples of partition types

- Rectilinear or structured subdivisions
- Adapting to facies, rock types, saturation regions, etc
- Partitions adapting to faults, fractures, . . .
- From block-structured grids, LGR, . . .
- Unstructured graph-based partitions
- Amalgamations based on indicators
- Adapting dynamically to flow
- Separating near-well and far-field
- . . .
Numerical example: SPE10

Layer 85: $220 \times 60$ subsample, pressure drop from north to south, linear relperms, equal viscosities

<table>
<thead>
<tr>
<th>Partition</th>
<th>$L^2$</th>
<th>$L^\infty$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rectangular</td>
<td>0.0307</td>
<td>0.1782</td>
</tr>
<tr>
<td>Metis</td>
<td>0.0791</td>
<td>0.5506</td>
</tr>
<tr>
<td>Combined</td>
<td>0.0293</td>
<td>0.2929</td>
</tr>
</tbody>
</table>
Layer 85: 220 × 60 subsample, pressure drop from north to south, linear relperms, equal viscosities

<table>
<thead>
<tr>
<th>Partition</th>
<th>$L^2$</th>
<th>$L^\infty$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rectangular</td>
<td>0.0307</td>
<td>0.1782</td>
</tr>
<tr>
<td>Metis</td>
<td>0.0791</td>
<td>0.5506</td>
</tr>
<tr>
<td>Combined</td>
<td>0.0293</td>
<td>0.2929</td>
</tr>
</tbody>
</table>
Numerical example: SPE10
Numerical example: SPE10

![Graph showing residual values for Rectangular, Metis, and Combined methods over iterations.](image)

- **Rectangular**
- **Metis**
- **Combined**

Key:
- $A_{ms}^{-1}$
- Four ms iterations

Graph indicates comparison of residual values across iterations for different methods, with notable peaks and slopes.
Numerical example: Unstructured grid

PEBI grid adapting to five faults and thirteen volumetric fractures.

Faults: 0.01 trans. multiplier
Fractures: 5 darcy
Background: average 100 md
Wells:
  – injector, bhp: 500 bar
  – producer, bhp: 200 bar
Numerical example: Unstructured grid

Faults: 0.01 trans. multiplier
Fractures: 5 darcy
Background: average 100 md

Wells:
– injector, bhp: 500 bar
– producer, bhp: 200 bar

Global basis
Global + well basis
Cart
Metis
Features
Cart + Metis
All three
Numerical example: Unstructured grid

PEBI grid adapting to five faults and thirteen volumetric fractures.

Faults: 0.01 trans. multiplier
Fractures: 5 darcy
Background: average 100 md
Wells:
– injector, bhp: 500 bar
– producer, bhp: 200 bar

Global basis
Global + well basis
Cart
Metis
Features
Cart + Metis
All three

with 12 iterations
Numerical example: Gullfaks

Higher resolution: $80 \times 100 \times 52$ cells, 416 000 active
Partition: rectangular (upper) and by Metis (lower)
Numerical example: Gullfaks
Conclusion

- Multiscale basis functions for pressure, fully unstructured
- Applicable to wide range of flow problems through finite-volume framework
- Emphasis on robust local method for fine-scale transport
- Unstructured coarsening allows for adaption to features
- Very simple to implement regardless of grid complexity
- Prototype in commercial simulator and MRST
Backup slides
Basis functions: MsFV vs MsRSB

The matrices report the net fluxes into or out of the neighboring coarse blocks induced by a unit pressure differential.
Example: water-based EOR

- Full Eclipse 100 polymer model with adsorption, Todd-Langstaff mixing, inaccessible pore volume, and permeability reduction
- Polymer concentration changes water viscosity to achieve better sweep
- Model includes shear thinning, i.e., water-polymer viscosity depends on the velocity.
- Non-Newtonian fluid rheology makes the pressure equation highly nonlinear
Example: validation on SPE10 layers

<table>
<thead>
<tr>
<th>Error</th>
<th>Grid</th>
<th>$p$ (L$^2$)</th>
<th>$p$ (L$^\infty$)</th>
<th>$v$ (L$^2$)</th>
<th>$v$ (L$^\infty$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MsFV</td>
<td>6 × 11</td>
<td>0.0313</td>
<td>0.0910</td>
<td>0.1138</td>
<td>0.4151</td>
</tr>
<tr>
<td>MsRSB</td>
<td>6 × 11</td>
<td>0.0204</td>
<td>0.0766</td>
<td>0.0880</td>
<td>0.4071</td>
</tr>
</tbody>
</table>
Example: validation on SPE10 layers

<table>
<thead>
<tr>
<th>Error</th>
<th>Grid</th>
<th>$p$ ($L^2$)</th>
<th>$p$ ($L^\infty$)</th>
<th>$v$ ($L^2$)</th>
<th>$v$ ($L^\infty$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MsFV</td>
<td>$6 \times 11$</td>
<td>0.2299</td>
<td>2.0725</td>
<td>0.4913</td>
<td>0.7124</td>
</tr>
<tr>
<td>MsRSB</td>
<td>$6 \times 11$</td>
<td>0.0232</td>
<td>0.0801</td>
<td>0.1658</td>
<td>0.3240</td>
</tr>
</tbody>
</table>
Numerical example: well basis

<table>
<thead>
<tr>
<th>Basis</th>
<th>Gaussian $L^2$</th>
<th>Gaussian $L^\infty$</th>
<th>Layered $L^2$</th>
<th>Layered $L^\infty$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$6 \times 6$</td>
<td>0.0641</td>
<td>0.1679</td>
<td>0.0619</td>
<td>0.1750</td>
</tr>
<tr>
<td>Well basis</td>
<td>0.0760</td>
<td>0.1131</td>
<td>0.1015</td>
<td>0.1215</td>
</tr>
<tr>
<td>Well basis + $6 \times 6$</td>
<td>0.0303</td>
<td>0.1136</td>
<td>0.0280</td>
<td>0.0634</td>
</tr>
</tbody>
</table>
Example: Validation of compositional simulator

- Simple one dimensional example
- Pressure drop of 50 bar over domain
- Compare different schemes to validate

Reservoir quantities, $p_{min} = 75$ bar
Example: Validation of compositional simulator

Compare MRST implementation to AD-GPRS research simulator

- Simple one dimensional example
- Pressure drop of 50 bar over domain
- Compare different schemes to validate

Mole fractions, $p_{min} = 75$ bar
Example: Validation of compositional simulator

Compare MRST implementation to AD-GPRS research simulator

- Simple one dimensional example
- Pressure drop of 50 bar over domain
- Compare different schemes to validate

Reservoir quantities, $p_{min} = 275$ bar
Example: Validation of compositional simulator

Compare MRST implementation to AD-GPRS research simulator

- Simple one dimensional example
- Pressure drop of 50 bar over domain
- Compare different schemes to validate

Mole fractions, $p_{min} = 275$ bar
Example: unstructured PEBI grid

- Unstructured grid designed to minimize grid orientation effects
- Two embedded radial grids near wells
- Fine grid adapts to faults
- The faults are sealed, i.e. allow no fluid flow through
Example: unstructured PEBI grid

Two-step preconditioner, ILU(0) as 2nd stage, Richardson iterations
Example: unstructured PEBI grid

Adapted grid gives better prolongation operator → faster initial convergence

Metis grid has more block connections → faster two-level convergence

Slow convergence: ILU(0)

Two-step preconditioner, ILU(0) as 2nd stage, Richardson iterations
Example: unstructured PEBI grid

- **Injector:** 1 PVI at constant rate. **Producer:** fixed bottom-hole pressure.
- **Relative mobility:** $\lambda_{rw} = s_w^2$, $\lambda_{ro} = (1 - s_w)^2 / 5$
- **Basis functions:** adapted grid, updated by reapplying smoother

Water front, fine-scale solution

Water front, multiscale solution
Example: unstructured PEBI grid

- Injector: 1 PVI at constant rate. Producer: fixed bottom-hole pressure.
- Relative mobility: $\lambda_{rw} = s_w^2$, $\lambda_{ro} = (1 - s_w)^2 / 5$
- Basis functions: adapted grid, updated by reapplying smoother

Water front, fine-scale solution

Water front, multiscale solution
Example: unstructured PEBI grid

- **Injector**: 1 PVI at constant rate. **Producer**: fixed bottom-hole pressure.
- **Relative mobility**: $\lambda_{rw} = s_w^2$, $\lambda_{ro} = (1 - s_w)^2 / 5$
- **Basis functions**: adapted grid, updated by reapplying smoother
MsRSB: computing basis functions

Divide set of fine cells $F$ into $m$ coarse blocks,

$$C_j \subseteq F, \quad C_j \cap C_i = \emptyset \quad \forall \quad i \neq j, \quad i, j \in [1, m], \quad |F| = n.$$  

Define support $I_j$ and its boundary $B_j$ for each block,

$$P_j(x) > 0, \quad x \in I_j \quad P_j(x) = 0 \text{ otherwise.}$$

For convenience, define global boundary/dual as union of all boundaries

$$G = B_1 \cup B_2 \cup \ldots \cup B_{m-1} \cup B_m.$$  

For cells in $G$, let $H_i$ be the set of blocks where it is active,

$$H_i = \{j \mid i \in I_j, i \in G\}.$$
Define preliminary update by Jacobi relaxation,

$$\hat{d}_j = -\omega D^{-1} A P_j^n.$$ 

Modify the update according to cell category,

$$d_{ij} = \begin{cases} 
\hat{d}_{ij} - P_{ij}^n \sum_{k \in H_i} \frac{\hat{d}_{ik}}{1 + \sum_{k \in H_i} \hat{d}_{ik}}, & i \in I_j, i \in G, \\
\hat{d}_{ij}, & i \in I_j, i \notin G, \\
0, & i \notin I_j. 
\end{cases}$$

Finally, apply the update and proceed to next iteration

$$P_{ij}^{n+1} = P_{ij}^n + d_{ij}$$
MsRSB: computing basis functions

- Jacobi iterations ensures algebraic smoothness
- Limited support by construction
- Does the proposed basis functions have partition of unity?

Two cases: \( i \in G \) and \( i \notin G \). First, consider \( i \notin G \):

\[
\sum_j P_{ij}^{n+1} = \sum_j P_{ij}^n - \frac{\omega}{A_{ii}} \sum_j \sum_k A_{ik} P_{kj}^n
\]

\[
= 1 - \frac{\omega}{A_{ii}} \sum_k A_{ik} \left( \sum_j P_{kj}^n \right)
\]

\[
= 1 - \frac{\omega}{A_{ii}} \sum_k A_{ik} = 1.
\]

We have used that \( \sum_{j \in H_i} P_{ij}^n = 1 \) by assumption and that \( P_{ij}^n \) is nonzero only in \( H_i \).
MsRSB: computing basis functions

- Jacobi iterations ensures algebraic smoothness
- Limited support by construction
- Does the proposed basis functions have partition of unity?

Two cases: \( i \in G \) and \( i \notin G \). Next, consider \( i \in G \).

\[
\sum_{j \in \{1, \ldots, m\}} P_{ij}^{n+1} = \sum_{j \in H_i} \left( P_{ij}^n + \frac{\hat{d}_{ij} - P_{ij}^n \sum_{k \in H_i} \hat{d}_{ik}}{1 + \sum_{k \in H_i} \hat{d}_{ik}} \right)
\]

\[
= 1 + \sum_{j \in H_i} \frac{\hat{d}_{ij} - P_{ij}^n \sum_{k \in H_i} \hat{d}_{ik}}{1 + \sum_{k \in H_i} \hat{d}_{ik}} - \sum_{k \in H_i} \frac{\hat{d}_{ik}}{1 + \sum_{k \in H_i} \hat{d}_{ik}} \sum_{j \in H_i} P_{ij}^n = 1.
\]

We have used that \( \sum_{j \in H_i} P_{ij}^n = 1 \) by assumption and that \( P_{ij}^n \) is nonzero only in \( H_i \).