Abstract
A novel multiscale method for non M-matrices using Multiscale Restricted Smoothed Basis (MsRSB) functions is presented. The original MsRSB method is enhanced with a filtering strategy enforcing M-matrix properties to enable the robust application of MsRSB as a preconditioner. Through applications to porous media flow and linear elastic geomechanics, the method is proven to be effective for scalar and vector problems with multipoint finite volume (FV) and finite element (FE) discretization schemes, respectively. Realistic complex (un)structured two- and three-dimensional test cases are considered to illustrate the method’s performance.

Keywords: Multiscale methods, MsRSB, Multipoint flux approximation, Finite element method, Preconditioning, Geomechanics

1. Introduction
Large-scale numerical simulations are often required to understand and predict real world dynamics. In many applications, the use of high-resolution grids is required to characterize the heterogeneity of the material properties and the geometric complexity of the domains. Such simulations impose severe computational challenges and motivate the need for efficient solution schemes. Attractive multilevel strategies to achieve this are multiscale methods [1]. In this paper, we propose a generalization of the multiscale restriction-smoothed basis method (MsRSB) recently put forward in [2], and investigate its use as an effective preconditioner for multipoint flux approximation finite volume (FV) and finite element (FE) discretizations of second-order elliptic problems. Specifically we focus on applications to porous media flow and linear elastic geomechanics.

The original idea underlying multiscale discretization methods for heterogeneous second-order elliptic problems can be traced back four decades [3, 4]. In essence, these methods aim at constructing accurate coarse-scale problems that preserve information of fine scale heterogeneity and can be solved at low computational cost. This is accomplished by numerically computing multiscale basis functions, which are local solutions of the original problem, that are used to both: (i) construct the coarse-scale problem, and (ii) interpolate the coarse-scale solution back to the fine-scale. Various methods to obtain these basis functions have been developed, for example generalized finite-element (GFE) methods [5], multiscale finite-element (MsFE) methods [6], numerical-subgrid upscaling [7], multiscale mixed finite-element (MsMFE) methods [8], multiscale finite-volume (MsFV) methods [9], multiscale mortar mixed finite-element (MsMMFE) methods [10], multilevel multiscale mimetic (M²) methods [11], multiscale mixed/mimetic finite-element (MsMFEM) [12] and generalized multiscale finite element (GMsFE) [13] methods, to name a few. In the geoscience community, multiscale methods have been extensively applied both as single-pass [9] and iterative schemes [14, 15]
to resolve some of the limitations of existing upscaling methods. They have established a solid framework for simu-
lating complex subsurface flow processes, e.g. [16–28]. Multiscale methods for linear elastic problems have focused
primarily on the derivation of accurate coarse space basis functions which are robust with respect to material prop-
erty heterogeneities and enable scalable performance [29–33]. Applications to the poroelasticity equations include
[34–40].

The MsRSB method was proposed in the context of FV simulation for fluid flow in highly heterogeneous porous
media [2]. Based on a two-grid approach, the MsRSB method constructs multiscale basis functions through restricted
smoothing on the fine-scale matrix. In more detail, the basis functions, which are consistent with the local differential
operators, are constructed with a cheap relaxation scheme, i.e. a weighted Jacobi iteration, similar to approaches
used in smoothed aggregation multigrid methods [41–43]. An important advantage of MsRSB is that smoothing by
relaxation provides a great deal of flexibility in handling unstructured grids, an essential requirement, for example,
in applications involving complex geological structures. MsRSB has been widely proven and implemented in open
source and commercial simulators using a linear two-point flux approximation (TPFA) [26].

Because of the two-point structure, the linear TPFA scheme is monotone [44], i.e. it preserves the positivity of
the differential solution [45], and leads to an M-matrix with a small stencil. This is the reason why linear TPFA is the
scheme of choice in most engineering software. Unfortunately, the consistency of TPFA is not guaranteed for arbitrary
grids and anisotropic permeability distributions, potentially leading to inaccurate results [46]. Therefore, other FV
methods such as multipoint flux approximation (MPFA) and/or nonlinear schemes [44, 47] must be considered to
achieve consistent fluxes. To date, MsRSB has not been combined with MPFA or other consistent discretizations.
Hence, in this paper, we focus on enhancing MsRSB to enable the solution of second-order elliptic problems using
discretization methods that do not result in an M-matrix. Based on the MPFA-O method [48], we show that the
MsRSB basis construction as presented in [2] can fail due to divergent iterations for an anisotropic diffusion problem.
We propose a variant of the original MsRSB approach that restores the desired behavior by enforcing M-matrix
properties based on a filtering strategy. We develop the new method focusing on FV discretizations for porous media
single-phase flow, and extend its use to vector elliptic problems by targeting FE-based simulation of linear elastic
geomechanics.

The paper is structured as follows. First, the original multiscale restriction-smoothed basis method is briefly
reviewed in 2. Second, MsRSB for an MPFA flow discretization is analyzed and the novel approach is proposed
in Section 3. Next, the proposed method is extended to geomechanics in Section 4. Challenging two- and three-
dimensional experiments are presented to demonstrate properties, robustness and scalability of the method throughout
Section 3 and 4, including comparisons to existing methods and published results. Finally the report is concluded and
future work specified.

2. The Multiscale Restriction-Smoothed Basis method (MsRSB)

We propose a two-level preconditioning framework based on MsRSB for accelerating iterative Krylov methods to
solve linear systems of the form:

\[ Au = f, \]  \hspace{1cm} (1)

where the coefficient matrix \( A \in \mathbb{R}^{n \times n} \) arises from a finite volume (FV) or finite element (FE) discretization of a
scalar or vector second-order elliptic problem. Furthermore, \( u = \{ u_j \}_{j=1}^n \in \mathbb{R}^n \) is the solution vector containing the
unknown degrees of freedom, and \( f = \{ f_j \}_{j=1}^n \in \mathbb{R}^n \) is the discrete forcing term. In this work we develop the method and
illustrate its performance focusing on two simple but representative models routinely employed in practical simulation
of subsurface processes: (i) the incompressible single-phase flow equation, and (ii) the linear elastostatic equations.
For the flow problem we will concentrate on FV fine-scale discretizations while for the elastostatics problem we will
consider the FE method. A review of governing equations and the derivation of the matrix form in (1) are provided
for both models in Appendix A and Appendix B, respectively.

The essence of any multiscale formulation and solution algorithm lies in the construction of a representative
coarse-scale problem capable of capturing fine-scale features of a high-resolution model. The connection among
scales is accounted for by computing basis functions, i.e. localized fine-scale solutions, which are used to construct
a coarse-scale (upscaled) problem and reconstruct a fine-scale (downscaled) solution from the coarse solution.
reconstruction stage can be represented through the prolongation operator \( P \), a sparse linear operator that stores the basis function associated to each coarse degree of freedom in the corresponding column such that

\[
P : \mathbb{R}^{n_c} \rightarrow \mathbb{R}^{n_f}, \quad u_c \mapsto u_f = Pu_c.
\]

(2)

Here and in the following, subscripts \( f \) and \( c \) indicate quantities associated with the fine and the coarse problem. In particular, \( n_c \) and \( n_f \) denote the number of coarse- and fine-scale degrees of freedom, respectively.

Assuming \( P \), to be specified hereafter, is available, the definition of the coarse problem proceeds as follows. First the fine-scale solution in (1) is replaced with the approximation \( u = u_f \approx Pu_c \). Second the resulting residual vector, namely \( r = (f - APu_c) \), is orthogonalized against \( n_c \) vectors in \( \mathbb{R}^{n_c} \) that form the rows of the operator \( R \). Hence, \( u_c \) is the solution to the linear system with \( n_c \) equations

\[
A_c u_c = f_c \quad \text{with} \quad A_c = RAP, \quad f_c = RF.
\]

(3)

We refer to \( R \) as the restriction operator, a linear operator mapping vectors from the fine- to the coarse-scale

\[
R : \mathbb{R}^{n_f} \rightarrow \mathbb{R}^{n_c}, \quad u_f \mapsto u_c = Ru_f.
\]

(4)

Different options may be considered for \( R \). If the fine-scale matrix \( A \) is symmetric positive definite (SPD), a Galerkin orthogonalization, i.e. \( R = P^T \), is typically the strategy of choice since it provides a coarse scale operator \( A_c \) that is still SPD. Alternatively, a Petrov-Galerkin approach is often used. For example, in the MSFV method [9], which is designed for diffusion problems, \( R \) is constructed such that discrete mass conservation also holds for the coarse-scale problem.

2.1. MsRSB for TPFA finite volume schemes

A crucial component of multiscale methods is the efficient and accurate construction of the prolongation operator, that is the computation of the basis functions. Originally proposed for the cell-centered finite volume solution to the diffusion equation for flow through porous media [2], the MsRSB method computes the basis functions iteratively with restricted smoothing. To describe this process, some terminology is first defined. A primary coarse grid is defined as a partitioning of the fine grid. In each coarse cell, a coarse node is chosen as the representative fine cell for that coarse cell. Furthermore, support boundary cells are defined as the cells connecting neighboring coarse nodes. Support edge cells are defined as the cells connecting a coarse node to its neighbors. Note that edge cells for one coarse node will be boundary cells for other coarse nodes. Fig. 1 shows an example of the coarse grid structure for a flow problem.

![Figure 1: Coarse grid features for the internal block of a regular 3×3 partition: (a) primal grid and coarse nodes (●); (b) support boundary (●); and edge (■) cells; and (c) basis function.](image-url)
The initial guess for the basis function associated to the \( j \)th coarse node consists of the characteristic function of the primary \( j \)th coarse cell, hence it is equal to 1 for fine scale cells belonging to the \( j \)th coarse cell and 0 elsewhere. Let \( L \) and \( U \) denote the strictly lower-triangular and upper-triangular part of the matrix \( G \) which we want to compute basis functions for. We assume that \( G \) has the same dimensions as the fine-scale discretization matrix \( A \), but otherwise leave the relationship ambiguous for the time being. Let \( \hat{D} \) be the diagonal matrix such that the entries in each row of matrix \( \hat{G} = (L + \hat{D} + U) \) sum to zero, i.e.

\[
[\hat{D}]_i = - \sum_{j=1, j \neq i}^{n} [G]_{ij}, \quad \forall i \in \{1, 2 \ldots, n\}.
\] (5)

Each MsRSB basis function is then iteratively smoothed by applying the following relaxation scheme

\[
[P]_{kj}^{k+1} = [P]_{kj}^{k} - \omega \hat{D}^{-1} \hat{G}[P]_{kj}^{k},
\] (6)

where \([P]_{kj}\) denotes the \( j \)th column of \( P \), i.e. the basis function corresponding to coarse node \( j \), and \( k \) is the iteration count. Furthermore, \( \omega \) is a relaxation factor which in this work is set to \( 2/3 \), i.e. the value warranting the optimal smoothing factor of the weighted Jacobi iteration for the homogeneous Poisson’s equation [49]. If the smoothing update extends the basis function outside of its support region, the update is adjusted to enforce that the basis support is enclosed by the boundary cells. This is done by adding overflowing basis function values to the neighboring basis functions. We refer the reader to [2] for additional details. The original MsRSB basis function construction process is also incorporated in algorithm 1, for a given coarse mesh and support regions. Importantly the described smoothing process theoretically guarantees the conservation of the initial partition of unity at each iteration. However, the authors note that round-off errors in a numerical implementation can cause a violation of this condition. As such, global basis function rescaling is required to guarantee the partition of unity. These are inexpensive operations which can be done every few iterations \( (n_{it}) \).

3. MsRSB for non-M matrices: multipoint FV schemes

In order to demonstrate the challenges of applying a multiscale method to discretized systems which do not result in M-matrices, we will consider a specific fine-scale discretization of the incompressible single-phase pressure equation

\[
- \text{div}(\Lambda \cdot \text{grad } p) = q,
\] (7)

where \( p \) is the scalar pressure field, \( q \) a source term distributed in the domain and \( \Lambda \) a positive-definite tensor describing the diffusion properties of the medium. Generally, \( \Lambda \) is principally characterized by the permeability tensor \( \kappa \) and fluid viscosity \( \mu \). A detailed description of the governing equations is provided in Appendix A.1. The standard approach for discretizing (7) is a two-point flux approximation (TPFA) scheme which is only consistent when the principal axes of the permeability tensor are aligned with the grid [50]—i.e., for so-called \( \kappa \)-orthogonal grids. One possible choice to solve pressure on rough grids with non-diagonal permeability tensors is the MPFA-O method [48], which may not result in M-matrices for grids of interest. In the MPFA-O method, transmissibilities are computed by enforcing continuity of fluxes over each half-face for local reconstructions of linear flow. These fluxes rely on all the cell pressure unknowns surrounding a grid node and guarantee by construction that the set of equations, related to the half faces connected to a grid node, result in a solvable system. As it is not the focus of this paper, the authors refer to the cited paper for more details on the well-established method. Furthermore, we note that the implementation of MsRSB for MPFA was done using the Matlab Reservoir Simulation Toolbox [46]. In the following, we address two points: i) we will consider scalar systems discretized with MPFA-O as a proxy for the inherent difficulties in applying multiscale methods to non-M-matrices and ii) demonstrate a practical approach for implementing MsRSB or similar methods to MPFA-type discretizations for flow. While the MsRSB method is well studied for TPFA-type problems, it has not, to the authors’ knowledge, been combined with consistent discretizations in the literature.
3.1. Extension to non M-matrices

To avoid issues related to non M-matrices, a method is devised to alter, with minimal intrusion, the original fine-scale matrix $A$ such that it satisfies M-matrix properties. This approximation of the linear system is justified because we aim to find an approximate solution using multiscale methods, where the basis functions should account for the M-matrix like part of the system matrix. This is conceptually similar to e.g. using incompressible or steady-state basis functions for flow when the problem under consideration is nonlinear due to compressibility [51, 52]. Moreover, this work’s primary objective is to employ the proposed method as a preconditioner for GMRES and other iterative Krylov solvers, where an inexpensive local solver will target any local errors in the approximation. In the following, we assume that the discretization matrix has by convention a positive diagonal and primarily negative off-diagonal entries.

To enforce M-matrix properties, it is sufficient to filter out all positive-diagonal entries and construct a modified system matrix $\tilde{A}$

$$[\tilde{A}]_{ij} = \min([A]_{ij}, 0), \quad \forall (i, j) \in \{1, 2, \ldots, n\} \times \{1, 2, \ldots, n\},$$

from which we can easily compute the basis functions by setting $G = \tilde{A}$. As the resulting matrix $\hat{G}$ from eq. (6) now has zero row sum with only non-positive off-diagonal entries, the enhanced MsRSB method guarantees the robust generation of partition-of-unity basis functions with entries in $[0, 1]$. Furthermore, note that these changes do not affect a problem described by an M-matrix and as such the method can be implemented generally. To verify that the method has the desired effect, a simple test case is devised. Starting from an equidistant 2-D Cartesian grid, all grid nodes are perturbed randomly in both $x$- and $y$-direction. Additionally the grid is stretched by a factor 10 in the $y$-direction. The resulting test case primal grid is shown in Fig. 2a. The fine grid has 9x9 cells where we coarsen by a ratio of 3 in each direction. Furthermore all flow properties are homogeneous. Note that MPFA-O for a $\kappa$-orthogonal Cartesian grid is equivalent to the TPFA method. Therefore the test case is designed to vary from an orthogonal grid substantially.

![Figure 2](attachment:figure2.png)

**Figure 2:** Basis function for the internal block of a regular 3x3 partition: (a) primal grid and coarse nodes (○); (b) support boundary (■), edge (□) and internal cells (●); (c) MsRSB basis function using the original fine-scale system; and (d) MsRSB basis function using the filtered fine-scale system. The convergence criteria is $||e_i||_\infty < 10^{-12}$, where $e_i$ is the smoother update on the internal cells (see algorithm 1). However as the case with the original linear system diverges, the basis function obtained after 11 iterations is plotted in (c).

The basis functions corresponding to the central coarse node are plotted in Figure 2. Figure 2c displays the basis function obtained after 11 iterations when using the original linear system. It is evident from the plot that the prolongator is diverging. Figure 2d presents the converged basis function when using the altered linear system. In
In this case, the desired monotone basis functions are obtained. Note that due to the cancellation of certain connections in the matrix, the basis function does not spread over the full domain. This effect is especially present in the corners of the dual-region. As will be shown in numerical examples in Section 4, although this has no impact on the final solution, it is favorable to avoid small coarsening ratios to ensure good performance. The proposed multiscale method is summarized in Algorithm 1. Furthermore, the multiscale preconditioning strategy employed in the numerical results is described in Algorithms 2-4.

**Algorithm 1** Enhanced MsRSB method to construct prolongation operator.

1: function EnhancedMsRSB(A,P)  
2: \[ \tilde{A} = \min(A, 0) \] \hspace{1cm} \text{Eq. 8}  
3: \[ \hat{A} \leftarrow \tilde{A} - \text{diag}(\text{rowsum}(\tilde{A})) \] \hspace{1cm} \text{Eq. 5}  
4: \[ \hat{D} = \text{diag}(\hat{A}) \]  
5: \[ k = 0, e_{it} = \infty \]  
6: while \( e_{it} > \epsilon \) do  
7: \[ \delta P = -\omega \hat{D}^{-1} \hat{A} P \] \hspace{1cm} \text{Eq. 6}  
8: Modify \( \delta P \) to avoid stencil growth outside of support region \hspace{1cm} \text{See section 3.3 step 2 of [2]}  
9: \[ P \leftarrow P + \delta P \]  
10: if \((k \mod n_f) = 0\) then  
11: \[ [P]_{i, i} \leftarrow [P]_{i, i} / \sum_{j}([P]_{i, j}) \] \hspace{1cm} \text{Rescale to guarantee numerical partition of unity}  
12: end if  
13: \[ e_{it} = \max_{i, j}(|[\delta P]_{i, j}|), \quad i \notin \text{support edges} \]  
14: \[ k \leftarrow k + 1 \]  
15: end while  
16: return \( P \) \hspace{1cm} \text{The converged prolongation operator}  

**Algorithm 2** Setup of enhanced MsRSB preconditioner.

1: procedure SetupMsRSBpreconditioner(A)  
2: if Not initialized then  
3: Initialize \( P \)  
4: end if  
5: \[ P = \text{EnhancedMsRSB}(A, P) \]  
6: if Petrov-Galerkin restriction then  
7: \[ R \leftarrow \text{Construct restriction operator} \] \hspace{1cm} \text{Compute restriction operator}  
8: end if  
9: \[ R = P^T \]  
10: end if  
11: \[ A_c = RAP \] \hspace{1cm} \text{Compute coarse-scale system matrix}  
12: \[ M_c^{-1} \approx A_c^{-1} \] \hspace{1cm} \text{Set up coarse system solver}  
13: end procedure

### 3.2. MsRSB for an MPFA-O discretization: 2D test case

The example problem is shown in Figure 3 and consists of a 100 by 100 structured grid discretizing a rectangular domain of 20 by 150 meters. We impose a unit pressure drop from \( x = 0 \) m to \( x = 20 \) m. Interior nodes are perturbed by a factor \( 0.2\Psi \Delta x \) where \( \Psi \in [-\frac{1}{2}, \frac{1}{2}] \) is a uniformly random variable with expected value 0. The tensor \( \Lambda \) has a diagonal value of \( \lambda_{xx} = \lambda_{yy} = 100 \text{ md-cP}^{-1} \) with off-diagonal elements \( \lambda_{xy} = \lambda_{yx} = 25 \text{ md-cP}^{-1} \). The combined effect of the non-orthogonal grid and the strength of the off-diagonal permeability means that directly applying the MsRSB iterative...
Algorithm 3 Application of the enhanced MsRSB

1: function APPLYMsRSBPRECONDITIONER(v)  
2: \[ v_c = Rv \]  > Restrict residual  
3: \[ w_c = M^{-1}_c v_c \]  > Solve coarse problem  
4: \[ w = Pw_c \]  > Interpolate the solution  
5: return w  
6: end function

Algorithm 4 Application of the two-level preconditioner with pre- and post-smoothing.

1: function APPLYMsRSBTOOLEVELPRECONDITIONER(A,v)  
2: \[ z \leftarrow z + M^{-1}_{\text{pre.}}(v - Az) \]  > Pre-smoothing relaxation starting from \( z = 0 \)  
3: \[ w = APPLYMsRSBPRECONDITIONER(v - Az) \]  > Compute correction  
4: \[ z \leftarrow z + w \]  > Apply correction  
5: \[ z \leftarrow z + M^{-1}_{\text{post.}}(v - Az) \]  > Post-smoothing relaxation  
6: return z  
7: end function

Process to the system matrix results in rapid divergence of the basis functions. With the proposed regularization, however, MsRSB can be employed and satisfactory convergence rates are obtained for both Richardson and GMRES-accelerated iterations. We partition the domain into 400 coarse blocks, each comprised of a 5 by 5 segment of fine cells and solve the MPFA system to a tolerance of \( 10^{-8} \). The results are displayed in Figure 4, where Symmetric Gauss-Seidel (SGS) or ILU(0) is used as the second stage of the preconditioner. We observe a clear improvement to convergence rates indicating that the basis functions successfully capture the local features of the system and resolve low-frequency errors. As a non-accelerated stand-alone solver, i.e. Richardson, MsRSB+ILU(0) converges in 30 iterations while ILU(0) fails to converge in 150 iterations. The set-up with the cheaper but less effective smoother SGS fails to converge in both cases although the multiscale stage again leads to a significantly higher convergence rate. The performance difference between the single- and two-level solvers is reduced when GMRES is used to accelerate the solution process, nonetheless the multiscale solvers still only require half as many iterations as smoothers alone.

3.3. MsRSB for an MPFA-O discretization: 3D Field Test Case

Next a somewhat more realistic conceptual problem is considered. The test case consists of a 50 by 50 structured grid, with 30 layers in the vertical direction. The physical domain has a horizontal extent of 1000 by 1000 meters,
with a vertical thickness of 100 meters. Similar to the 2D case, the nodes of the grid are perturbed to create a rough grid. Additionally, the top surface has a varying topography. The model has five different regions of a log-normally distributed diffusion tensor, with mean values of 700, 1000, 300, 800 and 100 md·cP$^{-1}$, respectively as shown in Figure 5a.

We consider a logically structured coarse mesh with block sizes of 5 by 5 by 5 fine cells, resulting in a total of 600 coarse blocks to partition the fine-grid with 75,000 cells. We apply a simple boundary condition resulting in flow from $x = 0$ to $x = 1000$ meters. In engineering applications, flow would typically be driven by wells, but our goal here is to produce flow over the entire domain to verify our implementation. The reference solution is plotted in Figure 5b.

The convergence rates of MsRSB + ILU(0), ILU(0), MsRSB + SGS and SGS are compared in Figure 6 with and without Krylov acceleration. We observe that the benefits of the multiscale stage are more significant than in the 2D example. This is likely due to the denser stencil in 3D which leads to a fundamentally more difficult linear system, even for an equivalent number of degrees of freedom. We point out that the multiscale solver with SGS smoothing exhibits robust convergence with GMRES acceleration while the smoother-only setup stagnates. We also observe that the performance of the multiscale solver with ILU(0) is excellent, using 21 and 11 iterations without and with GMRES, respectively.

4. MsRSB for non-M matrices: FE simulation of linear elastic geomechanics

In this section, we investigate the extension of the enhanced MsRSB preconditioner to linear systems arising from geomechanical problems assuming linear elastic behavior. By means of this application, we extend the method’s applicability to problems with vector unknowns. The linear elastostatics governing equations and their FE discretization using a classical displacement formulation [53] are reviewed in Appendix A.2 and Appendix B.2, respectively. To compute the nodal displacement discrete solution, the FE method requires the solution of linear systems characterized by a fine scale stiffness matrix $A$ that is SPD but not an M-matrix. Therefore, in the construction of the prolongation operator, a naïve implementation of MsRSB would exhibit the same convergence issues as discussed for the pressure equation in Section 3.1.

To apply the enhanced MsRSB preconditioner, we observe that if the displacement degrees of freedom are ordered based on each coordinate direction, the stiffness matrix $A$ possesses a block structure, namely:
Figure 5: The permeability, grid (a) and the reference solution (b) for the 3D MPFA example.

Figure 6: Stand-alone iterative (a) and GMRES-accelerated iterative performance (b) for the 3D MPFA example.
\[ A = \begin{bmatrix} A_{xx} & A_{xy} & A_{xz} \\ A_{yx} & A_{yy} & A_{yz} \\ A_{zx} & A_{zy} & A_{zz} \end{bmatrix}, \]  

which reflects the full coupling between \( x, y, \) and \( z \) components of displacements. For preconditioning purposes, the complete matrix is often replaced with a sparser block diagonal approximation

\[ A^{(\text{sd})} = \begin{bmatrix} A_{xx} & A_{xy} & 0 \\ A_{yx} & A_{yy} & 0 \\ 0 & 0 & A_{zz} \end{bmatrix}, \]  

namely the separate displacement component (SDC) approximation proposed in [54]. The motivation behind the SDC approximation is that, using Korn’s inequality, one can show that \( A^{(\text{sd})} \) is spectrally equivalent to \( A \) [55, 56]. Note that this approximation breaks down in the incompressible elasticity limit, i.e. Poisson ratio \( \nu \to 0.5 \). As each diagonal block in (9) corresponds to the finite element discretization of an anisotropic diffusion operator for the corresponding displacement component—see Remark Appendix B.2—, enhanced MsRSB can be readily applied to (10) to obtain a preconditioner for the fine-scale matrix. In FE-based elasticity simulation, two or three MsRSB basis functions are associated with displacement degrees of freedom located at each coarse node in a two- or three-dimensional space, respectively. Computing such basis functions using \( A^{(\text{sd})} \) implies that the fine-scale displacement field in each coordinate direction is expressed as a linear combination of coarse nodal displacement in the corresponding direction only. This represents a major difference with a classic MSFE approach [32] that accounts instead for additional coupling, at the price of a much denser prolongation operator, by representing the displacement solution in each direction employing basis functions in all coordinated directions. Nevertheless, we emphasize that the coarse operator, \( A_c \), computed using MsRSB basis functions will still capture the full coupling influences between \( x, y, \) and \( z \) displacement components.

This extension to the enhanced MsRSB method is tested on a simple homogeneous problem defined in terms of dimensionless quantities with unit Lamé parameters, i.e. setting \( E=1 \) and \( \nu=0.25 \). The fine grid is chosen to be \( 12 \times 12 \) Cartesian. The coarse grid has 5 coarse nodes in each direction. Finally, the grid is rescaled by a factor of 20 in the \( y \)-dimension. The high aspect ratio is chosen to induce strong non-M matrix properties of the resulting linear system. Figure 7a depicts the coarse and fine grid of the test case on the \( x-y \) plane. To assess the robustness of the enhanced MsRSB method including the extension to vector physics, a basis function obtained using a naïve implementation of MsRSB is compared to the same basis function obtained with the enhanced method. Here, a naïve implementation of MsRSB is a straightforward application of the original method for M matrices to the linear system. Figure 7 depicts the obtained basis functions for a given coarse node. It is obvious from the plots that the naïve implementation leads to diverging basis functions whereas the enhanced method recovers the expected bi-linear interpolators of a Cartesian homogeneous problem. The results reemphasize the findings of section 3.1 and illustrate the performance of the enhanced MsRSB method for vector physics.

4.1. MsRSB for Geomechanics: A 2D Heterogeneous Test Case

To compare the enhanced MsRSB method to an existing multiscale preconditioner for geomechanics [32], we investigate a 2D test case consisting of an elastic isotropic domain with a heterogeneous (layered) distribution of Young’s modulus, shown in Figure 8. Four mesh families (cart, skew, trig and rand) are considered, each with a different type of geometric distortion applied to the coarse elements. In each case a \( 224 \times 224 \) fine-scale mesh and \( 7 \times 7 \) coarse mesh are defined on the domain. The reader is referred to section 4.1 of [32] for a more detailed description of the test case setup. Two Krylov solver setups are tested. The first variant is Preconditioned Conjugate Gradient (PCG), with a symmetric preconditioning operator constructed by pre– and post–smoothing the multiscale (MsRSB) operator with no–fill Incomplete Cholesky factorization (IC(0)). The second version employs Bi-conjugate Gradient Stabilized (BiCGStab) with a two–stage preconditioning scheme which consists of the multiscale operator as the global step and no–fill incomplete LU factorization (ILU(0)) as the local smoother. This setup is identical to the one used in [32], except that MSFE is replaced by enhanced MsRSB to construct the multiscale operator.
Figure 7: Basis function relative to the displacement degree of freedom in the $x$-direction associated to the innermost coarse node of a regular $4 \times 4$ partition: (a) primal grid; (b) coarse ($\bullet$), support boundary ($\circ$), edge ($\bullet$) and internal ($\bullet$) nodes; (c) MsRSB basis function using the original fine-scale system; and (d) enhanced MsRSB basis function using the filtered fine-scale system. The convergence criteria is $||e||_{\infty} < 10^{-3}$, where $e$ is the smoother update on the internal nodes (see algorithm 1). However as the case with the original linear system diverges, the basis function obtained after 20 iterations is plotted in (c).

Figure 8: Structured 2D heterogeneous case [32]: Young’s modulus distribution and computational mesh setup.
Table 1: Structured 2D heterogeneous case: number of PCG iterations to converge to $10^{-8}$ for different mesh families, coarse grid sizes and Young’s modulus variations.

| $\beta$ | # coarse elements | laterally constrained | laterally unconstrained | simple shear |
|---------|-------------------|-----------------------|------------------------|------------|
|         | cart | skew | trig | rand | cart | skew | trig | rand | cart | skew | trig | rand |
| 1       | 56 x 56 | 5 | 7 | 7 | 6 | 5 | 7 | 7 | 6 | 6 | 7 | 7 | 6 |
| 28 x 28 | 9 | 12 | 13 | 10 | 9 | 13 | 13 | 11 | 12 | 15 | 15 | 13 |
| 14 x 14 | 16 | 21 | 22 | 20 | 18 | 24 | 23 | 22 | 24 | 27 | 27 | 23 |
| 7 x 7   | 37 | 40 | 46 | 40 | 41 | 45 | 48 | 45 | 50 | 43 | 47 | 43 |
| 2       | 56 x 56 | 5 | 6 | 7 | 6 | 7 | 7 | 8 | 7 | 7 | 7 | 8 | 7 |
| 28 x 28 | 8 | 11 | 12 | 10 | 9 | 12 | 12 | 10 | 13 | 15 | 17 | 14 |
| 14 x 14 | 17 | 22 | 25 | 21 | 19 | 24 | 27 | 24 | 32 | 32 | 32 | 30 |
| 7 x 7   | 47 | 44 | 52 | 48 | 58 | 53 | 55 | 55 | 67 | 50 | 54 | 52 |
| 3       | 56 x 56 | 5 | 7 | 10 | 8 | 9 | 9 | 11 | 9 | 10 | 10 | 11 | 10 |
| 28 x 28 | 8 | 12 | 14 | 11 | 11 | 13 | 15 | 13 | 16 | 17 | 19 | 16 |
| 14 x 14 | 16 | 21 | 25 | 30 | 23 | 31 | 44 | 33 | 39 | 40 | 52 | 40 |
| 7 x 7   | 50 | 54 | 66 | 60 | 72 | 64 | 85 | 71 | 94 | 63 | 75 | 64 |

Table 2: Structured 2D heterogeneous case: number of BICGSTAB iterations to converge to $10^{-8}$ for different mesh families, coarse grid sizes and Young’s modulus variations.

| $\beta$ | # coarse elements | laterally constrained | laterally unconstrained | simple shear |
|---------|-------------------|-----------------------|------------------------|------------|
|         | cart | skew | trig | rand | cart | skew | trig | rand | cart | skew | trig | rand |
| 1       | 56 x 56 | 5 | 7 | 6 | 5 | 4 | 7 | 6 | 5 | 6 | 7 | 7 | 5 |
| 28 x 28 | 8 | 10 | 11 | 8 | 8 | 12 | 11 | 10 | 13 | 16 | 17 | 14 |
| 14 x 14 | 16 | 20 | 19 | 16 | 18 | 21 | 22 | 19 | 26 | 29 | 32 | 26 |
| 7 x 7   | 37 | 33 | 40 | 40 | 41 | 38 | 43 | 43 | 40 | 47 | 44 | 42 |
| 2       | 56 x 56 | 4 | 6 | 6 | 5 | 5 | 6 | 6 | 6 | 5 | 6 | 6.5 | 5 |
| 28 x 28 | 7 | 10 | 12 | 8 | 9 | 10 | 12 | 9 | 12 | 19 | 20 | 17 |
| 14 x 14 | 17 | 26 | 26 | 18 | 18 | 23 | 25 | 22 | 37 | 39 | 46 | 37 |
| 7 x 7   | 46 | 50 | 54 | 49 | 52 | 58 | 53 | 53 | 51 | 59 | 55 | 51 |
| 3       | 56 x 56 | 4 | 6 | 8 | 6 | 6 | 8 | 10 | 9 | 6 | 7 | 8 | 7 |
| 28 x 28 | 10 | 11 | 13 | 10 | 11 | 12 | 17 | 10 | 17 | 19 | 19 | 18 |
| 14 x 14 | 22 | 30 | 39 | 30 | 21 | 32 | 40 | 34 | 46 | 49 | 47 | 50 |
| 7 x 7   | 49 | 61 | 73 | 63 | 63 | 71 | 86 | 71 | 62 | 67 | 61 | 58 |

Table 1 summarizes the iteration counts using PCG for different mesh families and levels of contrast in material properties. With exception of the case with extremely large coarse elements (7 x 7, which results in each coarse element consisting of 32 x 32 fine elements), the preconditioner offers robust performance over a variety of mesh families and types of boundary conditions. In a direct comparison against an identical setup using MSFE, from [32], Table 2 shows similar results obtained with BiCGStab and a single ILU(0) smoothing step. Here, the proposed MsRSB method achieves similar performance compared to MSFE, with only a modest (about 20–25% on average) increase in the number of iterations for most cases, with a notable exception of Cartesian laterally constrained case, where MSFE basis functions provide an exact interpolation. Note that the BiCGStab solver applies the preconditioner twice at every iteration, which corresponds to two applications of both multiscale operator and smoother, whereas PCG employs a single application of the preconditioner, which involves two smoothing steps (pre- and post-smoothing) and only one multiscale step. Nevertheless, comparable performance is observed in terms of iteration count, suggesting PCG with symmetric preconditioning should lead to a more efficient option due to its lower cost per iteration.
4.2. MsRSB for Geomechanics: A Geological 2D Cross-Section Test Case

A second geomechanics test case is designed to represent a 2D cross-section of an elastic subsurface porous media domain, characterized by distinct geological layers and faults (shown in Figure 9a). A vertical distribution of Young’s modulus is prescribed, based on a correlation for uniaxial compressibility developed in [57] and recently used in [32]. Specifically, vertical compressibility is computed as

\[ c_M = 0.01241 |\sigma'_y|^{1.1342} \]  \hspace{1cm} (11)

where

\[ \sigma'_y = \sigma_y + p = -0.12218 |z|^{1.0766} + 0.1 |z| \]  \hspace{1cm} (12)

is the vertical effective stress, consisting of vertical total stress \( \sigma_y \) [bar] and hydrostatic pressure \( p \) (both in units of [bar\(^{-1}\)]). Young’s modulus is expressed as

\[ E = \frac{(1 - 2\nu)(1 + \nu)}{(1 - \nu)c_M} \]  \hspace{1cm} (13)

with a Poisson ratio \( \nu \) set to 0.3 everywhere. In addition, a constant value is added to Young’s modulus in each layer, specifically the mean Young’s modulus in that layer multiplied by a layer-dependent coefficient. This is done to emulate discontinuities in material properties between layers as often encountered in real subsurface systems. The resulting distribution spans 3 orders of magnitude over the domain and is depicted in Figure 9c.

The domain is gridded with an unstructured triangular mesh that conforms to the layers and faults (see Figure 9b) with the faults themselves considered inactive (no slip between fault surfaces). Seven different resolutions of the mesh are considered, ranging between 11,879 and 745,900 elements — see Table 3 for detailed information on mesh resolution and corresponding problem sizes. The domain is subject to roller boundary conditions on three sides, while the ground surface is traction-free. The deformation process is driven by a constant pressure drawdown of \( \Delta p = 20 \) bar prescribed in a small reservoir zone inside the domain (shown in Figure 9b), that acts as an external distributed force.

To evaluate the algorithmic scalability of the MsRSB method for mechanics, the problem is solved with a Krylov solver (namely PCG) to a relative tolerance of \( 10^{-8} \). Three symmetric two-stage preconditioning operators are constructed using different choices of pre- and post-smoother: 2 sweeps of \( l_1 \)-Jacobi, symmetric Gauss-Seidel, and no-fill incomplete Cholesky factorization. The coarse grid for the multiscale solver is generated by agglomerating fine-scale cells based on face connectivity using METIS [58] graph partitioning software. For each mesh resolution, the ratio of fine–to–coarse elements and nodes is kept approximately the same, which results in the size of the coarse problem growing with mesh resolution. As a comparative baseline, a smoothing-only preconditioner (i.e. not involving the global multiscale step) is also applied to the problem for each choice of smoother. Krylov iteration counts are recorded to evaluate performance.

Table 4 summarizes the findings. The multiscale solver convergence remains well bounded for all mesh resolutions and only exhibits very mild mesh dependence, while the baseline approach does not scale well, in some cases failing to achieve convergence within 1000 CG iterations. This example demonstrates a clear benefit of using a multiscale approach for subsurface mechanical problems compared to relying on incomplete factorizations only. It also emphasizes the method’s excellent algorithmic scalability and mesh independence on two-dimensional unstructured grids.

4.3. MsRSB for Geomechanics: A 3D Test Case

A third test is performed on a 3D poromechanical domain representing a 16 \( \times 16 \times 4 \) km subsurface formation. The vertical distribution of Young’s modulus prescribed by Eq. 11–13 is applied without discontinuities, and Poisson ratio is again set to 0.3. Also similar to the previous case, boundary conditions are imposed to be rollers (zero normal displacement) on all sides except for the traction-free top surface. Forcing is prescribed through pressure drawdown of \( \Delta p_1 = 15 \) bar and \( \Delta p_2 = 22 \) bar, respectively, in two reservoirs located around the center of the domain highlighted in Figure 10a.
Physical domain, layers and faults.

Fine and coarse scale grids in addition to support regions of selected basis functions. Red markers denote coarse nodes; yellow markers are support boundaries; green markers are support edges.

Young’s modulus distribution.

Examples of MsRSB displacement basis functions after 10 iterations and their support regions.

Figure 9: Geological 2D cross-section test case.
Table 3: Geological 2D cross-section test case: refinement levels and corresponding fine- and coarse-scale problem sizes.

| ℓ  | fine grid | coarse grid | coarsening ratio |
|----|-----------|-------------|------------------|
|    | # cell    | # node      | # dof            | # cell | # node | # dof | cell | dof |
| 0  | 11,879    | 6,119       | 12,238           | 12     | 26     | 52    | 989.9 | 235.4 |
| 1  | 23,390    | 11,947      | 23,894           | 25     | 51     | 102   | 935.6 | 234.3 |
| 2  | 46,932    | 23,817      | 47,634           | 50     | 102    | 204   | 938.6 | 233.5 |
| 3  | 93,129    | 47,085      | 94,170           | 100    | 202    | 404   | 931.3 | 233.1 |
| 4  | 186,940   | 94,165      | 188,330          | 200    | 408    | 816   | 934.7 | 230.8 |
| 5  | 372,360   | 187,210     | 374,420          | 400    | 803    | 1,606 | 930.9 | 233.1 |
| 6  | 745,900   | 374,350     | 748,690          | 800    | 1,603  | 3,206 | 932.4 | 233.5 |

Table 4: Geological 2D cross-section test case: PCG iteration counts. "MsRSB" denotes the two-stage multiscale preconditioner with pre- and post-smoothing. "no MS" columns reports results for the baseline approach (only pre- and post-smoother, without the multiscale step). Dashes are shown for cases where the solver failed to converge in 1000 iterations.

| ℓ  | Ms_Jacobi (×2) | Sym. Gauss-Seidel | IC(0) |
|----|-----------------|-------------------|-------|
|    | MsRSB no MS     | MsRSB no MS       | MsRSB no MS |
| 0  | 117 259         | 66 145            | 47 97  |
| 1  | 113 369         | 63 204            | 43 137 |
| 2  | 122 517         | 68 290            | 47 194 |
| 3  | 126 727         | 71 407            | 48 268 |
| 4  | 129 —           | 72 570            | 49 385 |
| 5  | 133 —           | 75 816            | 51 544 |
| 6  | 134 —           | 75 —              | 50 769 |

The domain is initially gridded with a 70 × 70 × 70 structured Cartesian grid, labeled as cart. It is then coarsened with a factor of 10 in each dimension, resulting in a 7 × 7 × 7 coarse-scale grid. In addition, nodes of the grid are shifted, resulting in a skewed grid, shown in Figure 10a, labeled skew. For both grids, the multiscale preconditioner was constructed using the same choices of pre- and post-smoothers as in the previous example, i.e. ℓ1-Jacobi, symmetric Gauss-Seidel and no-fill incomplete Cholesky (the latter not being used with the skewed mesh due to numerical breakdowns in factorization which are not related to the multiscale method). A sequence of progressively finer coarse grids is used, containing between 5 and 14 coarse cells in each dimension. Table 5 reports the observed iteration counts using CG as the chosen Krylov method and Figure 11 compares convergence histories obtained using both multiscale (with a 10 × 10 × 10 coarse element size) and smoother-only preconditioned CG solvers.

The results are in line with the lower dimensional test cases. Acceptable iteration counts and good scalability is observed. Moreover, the test case indicates that the developed enhanced MsRSB method generates a robust multiscale preconditioner for 3D geomechanics or other vectorial physics yielding non-M matrices.

Table 5: 3D domain test case: coarse grid dimensions and CG iteration counts with different smoother options.

| # coarse cells | coarse cell size | ℓ1-Jacobi ×2 | Gauss-Seidel | IC(0) |
|----------------|------------------|--------------|--------------|-------|
|                | cart  skew       | cart  skew   | cart  skew   |       |
| 5 × 5 × 5      | 14 × 14 × 14     | 257 269      | 93 96        | 24 —  |
| 7 × 7 × 7      | 10 × 10 × 10     | 213 229      | 77 82        | 21 —  |
| 10 × 10 × 10   | 7 × 7 × 7        | 178 203      | 67 73        | 17 —  |
| 14 × 14 × 14   | 5 × 5 × 5        | 149 169      | 58 63        | 13 —  |
(a) Mesh and reservoir zones.

(b) Young’s modulus distribution.

Figure 10: 3D skewed mesh test case.

Figure 11: CG-accelerated iterative performance for the Cartesian 3D mesh (a) and for the skewed structured 3D mesh (b).
5. Conclusion

A novel multiscale method for non M-matrices using Multiscale Restricted Smoothed Basis (MsRSB) functions is presented. The essence of the enhanced MsRSB method consists of enforcing M-matrix properties on the fine-scale linear system based on a filtering technique. Using the resulting approximate linear system, basis functions can robustly be constructed using the original iterative MsRSB strategy. The method is demonstrated for the single phase flow problem and the linear elastic geomechanics problem. Enhanced MsRSB is validated to be effective through various test cases including heterogeneous, unstructured 2-dimensional and 3-dimensional problems. Moreover, when it is applied as a preconditioner, the results show similar iteration counts compared to existing multiscale methods while enabling increased flexibility, easier implementation and sparser systems. Studies on CPU times and computational efficiency are subject of further work.

Finally, as shown for the porous media problem, the proposed method allows for the application of multiscale methods to multipoint stencils. Noting that multiscale operators are themselves inherently multipoint, the adapted MsRSB enables multilevel multiscale. Such a development has not yet been achieved in literature and is the topic of current research.

Acknowledgements

SB is supported by a named Stanford Graduate Fellowship in Science and Engineering (SGF). Funding for SK and NC was provided by Total S.A. through the FC-MAELSTROM Project. OM is funded by VISTA, which is a basic research program funded by Equinor and conducted in close collaboration with The Norwegian Academy of Science and Letters. Portions of this work were performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07-NA27344. The authors also thank Dr. Igor Shovkun and Dr. Jacques Franc for their feedback and Prof. Hamdi Tchelepi for many helpful discussions.

Appendix A. Model problems: Governing equations

Let \( \Omega \subset \mathbb{R}^{n_{sd}} \) and \( \Gamma \) denote a domain occupied by a heterogeneous porous medium and its boundary, respectively, with \( n_\Gamma \) the unit outward normal vector to \( \Gamma \), \( x \) the position vector in \( \mathbb{R}^{n_{sd}} \), and \( n_{sd} (= 2 \text{ or } 3) \) the spatial dimension of the problem.

Appendix A.1. Incompressible single-phase flow

Let \( p \) denote the pore pressure. For the application of the boundary conditions, let \( \Gamma \) be decomposed as \( \Gamma = \Gamma^D \cup \Gamma^N \), where \( \Gamma^D \cap \Gamma^N = \emptyset \). The strong form of the incompressible single-phase flow boundary value problem (BVP) may be formally stated as follows: given \( q : \Omega \to \mathbb{R} \), \( g_D : \Gamma^D \to \mathbb{R} \), and \( g_N : \Gamma^N \to \mathbb{R} \), find \( p : \overline{\Omega} \to \mathbb{R} \) such that

\[
\begin{align*}
\text{div } w(p) &= q, & \text{in } \Omega & \quad \text{(pressure equation)}, \\
p &= g_D, & \text{on } \Gamma^D & \quad \text{(prescribed boundary pressure)}, \\
w(p) \cdot n_t &= g_N, & \text{on } \Gamma^N & \quad \text{(prescribed boundary flux)}, \\
w(p) &= -\Lambda \cdot \text{grad } p, & \text{in } \Omega & \quad \text{(Darcy's law)}.
\end{align*}
\]

Here, \( w(p) \) is the Darcy velocity, \( \Lambda = (k/\mu) \) is the the rank-two tensor characterizing the diffusion properties of the medium, with \( k \) the intrinsic symmetric positive definite permeability tensor and \( \mu \) the fluid viscosity, which is assumed constant, and \( q \) denotes a volumetric source term.
Appendix A.2. Linear elastostatics

Let \( \mathbf{d} = [d_\ell]_{\ell \in \mathbb{N}} \) be the displacement-vector, where \( d_\ell = (e_\ell \cdot \mathbf{d}) \) are the displacement components with respect to the Euclidean basis \( \{e\}_{\ell=1}^{n_{sd}} \) in \( \mathbb{R}^{n_{sd}} \). Let us consider \( n_{sd} \) non-overlapping partitions of the domain boundary into two segments associated with Dirichlet, \( \Gamma^D_\ell \), and Neumann boundary conditions, \( \Gamma^N_\ell \), respectively, such that \( \Gamma = \Gamma^D_\ell \cup \Gamma^N_\ell \), with \( \Gamma^D_\ell \cap \Gamma^N_\ell = \emptyset \), \( \ell \in \{1, \ldots, n_{sd}\} \). The strong form of the linear elastostatic BVP reads as: given \( \mathbf{b} : \Omega \rightarrow \mathbb{R}^3 \), \( g_{D,\ell} : \Gamma^D_\ell \rightarrow \mathbb{R} \), and \( g_{N,\ell} : \Gamma^N_\ell \rightarrow \mathbb{R} \), find \( \mathbf{d} : \overline{\Omega} \rightarrow \mathbb{R}^3 \) such that
\[
-\text{div} \ \sigma(\mathbf{d}) = \mathbf{b}, \quad \text{in } \Omega \quad \text{(equilibrium equations),} \\
d_\ell = g_{D,\ell}, \quad \text{on } \Gamma^D_\ell \quad \text{(prescribed boundary displacements),} \\
\sigma(\mathbf{d}) : (e_\ell \otimes n_\ell) = g_{N,\ell}, \quad \text{on } \Gamma^N_\ell \quad \text{(prescribed boundary tractions),} \\
\sigma(\mathbf{d}) = C : \text{sym}(\text{grad } \mathbf{d}), \quad \text{in } \Omega \quad \text{(generalized Hooke’s law),}
\]
with \( \ell \in \{1, \ldots, n_{sd}\} \). Here, \( \sigma(\mathbf{d}) \) is the rank-2 stress-tensor, respectively, \( C_{\ell,\ell} \) is the rank-4 elasticity tensor, and \( \mathbf{b} \) is a body force. In this work we will focus on isotropic linear elastic materials, hence only two independent elastic coefficients are required for the definition of \( C \), namely
\[
C = \lambda (I \otimes I) + 2G I,
\]
where \( I \) and \( I \) are the second-order and fourth-order identity tensor, respectively, and \( \lambda = \frac{E}{3(1-2\nu)} \) and \( G = \frac{E}{2(1+\nu)} \) are the Lamé parameters of the material, with \( E \) the Young modulus and \( \nu \) the Poisson ratio. Note that the subscripts \( x, y, \) and \( z \) are also used to denote a quantity associated with the spatial dimension \( \ell \) equal to 1, 2, and 3, respectively.

Appendix B. Model problems: Discrete formulation

Appendix B.1. Incompressible single-phase flow: Finite volume formulation

Given a partition \( \mathcal{T}^h \) of the domain \( \overline{\Omega} \) consisting of non-overlapping conforming cells, a finite volume discretization of (A.1) consists of writing the pressure equation for each cell (control-volume) in \( \mathcal{T}^h \) in integral form \[59\]. Let \( \mathcal{F}^h \) be the set of interfaces in \( \mathcal{T}^h \), namely edges \( (n_{sd} = 2) \), or faces \( (n_{sd} = 3) \). Let \( \mathcal{V}^h \) be the space of piecewise constant cell-wise functions associated with \( \mathcal{T}^h \). We consider a discrete approximation for the pressure field such that \( p \approx p^h \in \mathcal{V}^h \). Let \( g^h_{D,\ell} \) denote the piecewise constant interface-wise interpolant of \( g_{D,\ell} \) having support in \( \mathcal{F}^{h,\ell} \subset \mathcal{F}^h \), namely the set of interfaces belonging to \( \Gamma^D_\ell \). Similarly, \( \mathcal{F}^{h,\ell} \subset \mathcal{F}^h \) is the set of boundary interfaces lying on \( \Gamma^N_\ell \). Finally, let \( \tilde{w}^\gamma \) denote a conservative numerical flux approximating the volumetric flux through an interface \( \gamma \in \mathcal{F}^h \), namely \( \tilde{w}^\gamma = \int_{\Gamma^h_\gamma} w \cdot n_\gamma \text{d}F \), with \( n_\gamma \) a unit normal vector defining a unique global orientation for \( \gamma \). Based on a suitable functional dependence on \( u^h \) and \( g^h_{D,\ell} \), for linear flux approximation schemes \( \tilde{w}^\gamma(u^h, g^h_{D,\ell}) \) can be split as a sum of two terms;
\[
\tilde{w}^\gamma(u^h, g^h_{D,\ell}) = \tilde{w}^\gamma(u^h) + \tilde{w}^\gamma(g^h_{D,\ell}),
\]
(B.1)
to highlight the contribution to the flux related to \( u^h \) and \( g^h_{D,\ell} \), respectively. Clearly, \( \tilde{w}^\gamma(g^h_{D,\ell}) \) is nonzero only in the presence of non homogeneous pressure boundary conditions. The first term to the right-hand side in (B.1) is expressed as a linear combination of pressure values from selected cells—e.g., the two cells sharing \( \gamma \) in the TPFA method, or the cells sharing at least a vertex with \( \gamma \) in the MPFA-O method \[48, 50\]—using constant transmissibility coefficients. A similar linear expression is utilized for \( \tilde{w}^\gamma(g^h_{D,\ell}) \). Nonlinear flux approximation schemes are not considered in this work. For a review and details on recent developments on finite volume discretizations for anisotropic diffusion problems in heterogeneous media, we refer the reader to \[44, 47\] and references therein.

The finite volume discretization provides an approximation \( u^h \) to the weak pressure solution of (A.1) by solving a set of discrete balance equations that are equivalent to the following mesh-dependent variational problem: find \( u^h \in \mathcal{V}^h \) such that
\[ a^h(v, u) = F^h(v) \quad \forall v \in V^h, \]  
(\text{B.2})

where the discrete bilinear form \( a^h : V^h \times V^h \to \mathbb{R} \) and the discrete linear form \( F : V^h \to \mathbb{R} \) are defined as

\[
d^h(v, u) = \sum_{\gamma \in F_{\Omega}^{\text{in}}} \left[ v \right]_{\gamma} \tilde{w}^\gamma(u),
\]
(\text{B.3})

\[
F^h(v) = \int_{\Omega} v \, \delta \, d\Omega + \sum_{\gamma \in F_{\Omega}^{\text{in}}} \left[ v \right]_{\gamma} \tilde{\sigma}^\gamma + \sum_{\gamma \in F_{\Omega}^{\text{in}}} \left[ v \right]_{\gamma} \tilde{w}^\gamma(s_D^h).
\]
(\text{B.4})

Here, the symbol \( \left[ \right]_{\gamma} \) denotes the jump of a quantity across an interface \( \gamma \in \mathcal{F}^h \). For internal interfaces, \( \left[ v \right]_{\gamma} = (v^h_{\gamma} - v^h_{\gamma}) \), with \( v^h_{\gamma} \) and \( v^h_{\gamma} \) the restriction of \( v \) on cells \( \tau_K \) and \( \tau_L \) sharing \( \gamma \), with \( \mathbf{n}_r \) pointing from \( \tau_K \) to \( \tau_L \).

For domain boundary interfaces, the jump expression simplifies to \( \left[ v \right]_{\gamma} = -v^h_{\gamma} \). To obtain the matrix form of the FV discrete problem, we introduce the basis \( \{ \chi_i \}_{i \in N^h} \) for \( V^h \), with \( \chi_i \) the characteristic function of the \( i \)th cell \( \tau_i \) in \( \mathcal{T}^h \) such that \( \chi_i(x) = 1 \), if \( x \in \tau_i \), \( \chi_i(x) = 0 \), if \( x \notin \tau_i \), and \( N^h = \{ 1, \ldots, n_r \} \) with \( n_r \) the total number of cells. Hence, the approximate pressure field is expressed as \( p(x) \approx u^h(x) = \sum_{i \in N^h} u_i \chi_i(x) \), with \( u_i \) the unknown cell pressure values. Requiring that \( a^h \) satisfy (\text{B.4}) for each function of the basis itself yields the system of discrete balance equations for the unknown coefficients vector \( \mathbf{u} = [u_i] \)

\[
A_p \mathbf{u} = \mathbf{f}_p,
\]
(\text{B.5})

with the system matrix \( A_p \) and the right-hand side \( \mathbf{f}_p \) such that \( [A_p]_{ij} = a(\chi_i, \chi_j) \) with \( i, j \in N^h \times N^h \) and \( [\mathbf{f}_p]_{i} = F(\chi_i) \), with \( i \in N^h \). The properties of matrix \( A_p \) depend on the flux approximation scheme chosen for \( \tilde{w}^\gamma \). For example, a TPFA scheme produces a symmetric positive definite matrix whereas MPFA methods typically lead to a non-symmetric \( A_p \) [44].

Appendix B.2. Linear elastostatics: Galerkin finite element formulation

The weak form of the linear elastostatics BVP is derived based on the classical displacement formulation [53] by eliminating \( \sigma(d) \) in (A.2a) using (A.2d). Under appropriate regularity assumptions, (A.1) admits a unique solution \( d \) that can be obtained by solving an equivalent variational problem [53]. Let \( V = \{ v \in H^1(\Omega)^{n_d} : v_{\gamma} = (e_{\ell} \cdot v) \in V_{L} \} \) denote the space of test functions, where \( V_{L} = \{ v \in H^1(\Omega) : v_{\gamma} = 0 \}, \ell \in \{ 1, \ldots, n_d \} \). Let us consider an extension of the Dirichlet boundary datum \( \bar{g}_D \in [H^1(\Omega)]^{n_d} \) such that \( (e_{\ell} \cdot \bar{g}_D) = g_{D,\ell} \) on \( \Gamma_D \), \( \ell \in \{ 1, \ldots, n_d \} \). By expressing the displacement as \( d = \bar{g}_D + u \), the weak form of (A.2) reads as: find \( u \in V \) such that

\[
a(v, u) = F(v) \quad \forall v \in V,
\]
(\text{B.6})

where the bilinear form \( a : V \times V \to \mathbb{R} \) and the linear form \( F : V \to \mathbb{R} \) are defined as

\[
a(v, u) = \int_{\Omega} \text{sym} (\text{grad } v) : \mathbf{C}_{dr} : \text{sym} (\text{grad } u) \, d\Omega,
\]
(\text{B.7})

\[
F(v) = \int_{\Omega} \mathbf{b} \cdot d\Omega + \sum_{\ell=1}^{n_d} \int_{\Gamma_D} v^1 \bar{g}_{D,\ell} \, d\Gamma - \int_{\Omega} \text{sym} (\text{grad } v) : \mathbf{C}_{dr} : \text{sym} (\text{grad } \bar{g}_D) \, d\Omega
\]
(\text{B.8})

Let \( X^h \) be the finite element space of piecewise polynomial vector functions that are continuous in \( \overline{\Omega} \) associated with a conforming triangulation \( \mathcal{T}^h \) of \( \Omega \). Let \( \{ \eta_i \}_{i \in N^h} \) be the standard (vector) nodal basis for \( X^h \), with \( N^h = \{ 1, \ldots, n_n\} \) and \( n_n \) the number of node points in \( \mathcal{T}^h \). We define the finite dimensional counterpart of \( V \) as \( V^h = X^h \cap V \) and denote
its basis as \(\{\eta_i\}_{i< N^h}\), with \(N^h u \subset N^h\) the set of indexes of basis functions of \(X^h\) vanishing on \(\Gamma^D, \ell \in \{1, \ldots, n_{sd}\}\). The discrete approximation to the displacement field can then be expressed as

\[
d(x) \approx \tilde{g}_D^h(x) + u^h(x) = \sum_{i=1}^{n_{sd}} \sum_{j < N^h} g_{D, \ell}^i(\phi_i)(x) e^{\ell}_i + \sum_{j < N^h} u^i \eta_i(x),
\]

where \(\tilde{g}_D^h \in X^h\) is the trivial discrete extension of the Dirichlet boundary datum such that \(e^{\ell} \cdot \tilde{g}_D^h\) on \(\Gamma^D\) is equal to the finite element interpolant of \(g_{D, \ell}, \ell \in \{1, \ldots, n_{sd}\}\), \(\phi_i = (e^{\ell}_i \cdot \eta_i(x))\), and \(u^h \in V^h\) is an approximate solution to the corresponding homogeneous Dirichlet problem with \(u^i\) the unknown nodal displacement degrees of freedom. Substituting \(g_D^h\) by \(\tilde{g}_D^h\) in (B.8) and requiring that \(u^h\) satisfy (B.6) for each basis function of \(V^h\) yields the matrix form of the variational problem, namely the system of equations for the unknown coefficients vector \(u = \{u_i\}\)

\[
A_d u = f_d,
\]

with \(A_d\) and \(f_d\) the symmetric positive definite (SPD) stiffness matrix and force vector, respectively, such that \([A_d]_{ij} = a(\eta_i, \eta_j)\) with \([i, j] \in N^h u \times N^h u\) and \([f_d]_i = F(\eta_i),\) with \(i \in N^h u\).

Remark Appendix B.1. For efficiency reasons, the linear system (B.11) is typically assembled ignoring the essential (Dirichlet) boundary conditions—i.e., \(\{\eta_i, \eta_j\}\) is the range over the bases for \(X^h\). This is also the strategy adopted in our implementation. The Dirichlet conditions are introduced by using a so-called symmetric diagonalization approach [60]. In this approach, rows and columns of the stiffness matrix associated with displacement degrees of freedom where such conditions apply are modified while preserving symmetry, with the right-hand-side updated accordingly.

Remark Appendix B.2. If displacement degrees of freedom are ordered based on each coordinate direction, \(A_d\) possesses a \(n_{sd} \times n_{sd}\) block structure, namely:

\[
A_d = \begin{bmatrix} A_{11} & \cdots & A_{1 n_{sd}} \\ \vdots & \ddots & \vdots \\ A_{n_{sd} 1} & \cdots & A_{n_{sd} n_{sd}} \end{bmatrix},
\]

which reflects the full coupling between \(\ell\)-components of displacements, \(\ell = \{1, \ldots, n_{sd}\}\). Each diagonal block \(A_{\ell \ell}\) in (9) corresponds to the finite element discretization of the anisotropic diffusion operator \(\text{div}(A \cdot \text{grad} d^\ell)\), with anisotropic diffusion tensor \(A = GI + (G + \lambda) e^{\ell} \otimes e^{\ell}\), \(\ell = \{1, \ldots, n_{sd}\}\).

References

References

[1] Y. Efendiev, T. Y. Hou, Multiscale Finite Element Methods: Theory and Applications, Springer, New York, NY, USA, 2009. doi:10.1007/978-0-387-09496-0.

[2] O. Møyner, K.-A. Lie, A multiscale restriction-smoothed basis method for high contrast porous media represented on unstructured grids, Journal of Computational Physics 304 (2016) 46–71. doi:10.1016/j.jcp.2015.10.010.

[3] L. G. Strakhovskaya, R. P. Fedorenko, A version of the finite element method, USSR Computational Mathematics and Mathematical Physics 19 (4) (1979) 162–173. doi:10.1016/0041-5553(79)90164-2.

[4] I. Babuška, J. E. Osborn, Generalized finite element methods: Their performance and their relation to mixed methods, SIAM Journal on Numerical Analysis 20 (3) (1983) 510–536. doi:10.1137/0720034.

[5] I. Babuška, G. Caloz, J. E. Osborn, Special finite element methods for a class of second order elliptic problems with rough coefficients, SIAM Journal on Numerical Analysis 31 (4) (1994) 945–981. doi:10.1137/0731051.

[6] T. Y. Hou, X.-H. Wu, A multiscale finite element method for elliptic problems in composite materials and porous media, Journal of Computational Physics 134 (1) (1997) 169–189. doi:10.1006/jcph.1997.5682.

[7] T. Arbogast, Implementation of a locally conservative numerical subgrid upscaling scheme for two-phase Darcy flow, Computational Geosciences 6 (3–4) (2002) 453–481. doi:10.1023/A:1021295215383.

[8] Z. Chen, T. Y. Hou, A mixed multiscale finite element method for elliptic problems with oscillating coefficients, Mathematics of Computation 72 (2003) 541–576. doi:10.1090/S0025-5718-02-01441-2.
[41] P. Vanek, J. Mandel, M. Brezina, Algebraic multigrid by smoothed aggregation for second and fourth order elliptic problems, Computing 56 (3) (1996) 179–196. doi:10.1007/BF02238511.

[42] P. Vanek, J. Mandel, M. Brezina, Convergence of algebraic multigrid based on smoothed aggregation, Numerische Mathematik 88 (3) (2001) 559–579. doi:10.1007/s0211-001-8015-y.

[43] M. Brezina, R. Falgout, S. MacLachlan, T. Manteuffel, S. McCormick, J. Ruge, Adaptive smoothed aggregation (oSA) multigrid, SIAM Review 47 (2) (2005) 317–346. doi:10.1137/050626272.

[44] J. Droniou, Finite volume schemes for diffusion equations: Introduction to and review of modern methods, Mathematical Models and Methods in Applied Sciences 24 (08) (2014) 1575–1619. doi:10.1142/S0218202514400041.

[45] A. Berman, R. J. Plemmons, Nonnegative Matrices in the Mathematical Sciences, Society for Industrial and Applied Mathematics, 1994. doi:10.1137/1.9781611971262.

[46] K.-A. Lie, An Introduction to Reservoir Simulation Using MATLAB/GNU Octave, Cambridge University Press, 2019. doi:10.1017/9781108591416.

[47] K. M. Terekhov, B. T. Mallison, H. A. Tchelepi, Cell-centered nonlinear finite-volume methods for the heterogeneous anisotropic diffusion problem, Journal of Computational Physics 330 (2017) 245–267. doi:10.1016/j.jcp.2016.11.010.

[48] I. Aavatsmark, An introduction to multipoint flux approximations for quadrilateral grids, Computational Geosciences 6 (3-4) (2002) 405–432. doi:10.1023/A:1021291114476.

[49] Y. Saad, Iterative Methods for Sparse Linear Systems, 2nd Edition, Society for Industrial and Applied Mathematics, 2003. doi:10.1137/1.9780898718003.

[50] M. G. Edwards, C. F. Rogers, Finite volume discretization with imposed flux continuity for the general tensor pressure equation, Computational Geosciences 4 (2) (1998) 259–290. doi:10.1023/A:1011510505406.

[51] O. Axelsson, I. Gustafsson, Iterative methods for the solution of the Navier equations of elasticity, Computer Methods in Applied Mechanics and Engineering 15 (2) (1978) 241–258. doi:10.1016/0045-7825(78)90026-9.

[52] R. Blaheta, Displacement decomposition-incomplete factorization preconditioning techniques for linear elasticity problems, Numerical Linear Algebra with Applications 1 (2) (1994) 107–128. doi:10.1002/nla.1680010203.

[53] I. Gustafsson, G. Lindskog, On parallel solution of linear elasticity problems: Part i: Theory, Numerical Linear Algebra with Applications 5 (2) (1998) 123–139. doi:10.1002/(SICI)1099-1506(199803/04)5:2<123::AID-NLA130>3.0.CO;2-D.

[54] D. Bui, M. Ferronato, G. Gambolati, P. Teatini, Basin-scale compressibility of the northern Adriatic by the radioactive marker technique, Géotechnique 52 (8) (2002) 605–616. doi:10.1680/geot.52.8.605.38832.

[55] G. Karypis, V. Kumar, A fast and highly quality multilevel scheme for partitioning irregular graphs, SIAM Journal on Scientific Computing 20 (1) (1998) 359–392. doi:10.1137/S1064827596287997.

[56] R. Eymard, T. Gallouët, R. Herbin, Finite volume methods, in: P. G. Ciarlet, J. L. Lions (Eds.), Handbook of Numerical Analysis, Vol. 7, Elsevier, 2000, pp. 713–1018. doi:10.1016/S1570-8650(00)70008-8.

[57] L. Formaggia, F. Saleri, A. Veneziani, Solving Numerical PDEs: Problems, Applications, Exercises, Springer, 2012.