3-9-2020

Multilevel Spectral Coarsening for Graph Laplacian Problems with Application to Reservoir Simulation

Andrew T. Barker
Lawrence Livermore National Laboratory

Stephen V. Gelever
Portland State University

Chak Shing Lee
Lawrence Livermore National Laboratory

Sarah Osborn
Lawrence Livermore National Laboratory

Panayot S. Vassilevski
Portland State University, panayot@pdx.edu

MULTILEVEL SPECTRAL COARSENING FOR GRAPH LAPLACIAN PROBLEMS WITH APPLICATION TO RESERVOIR SIMULATION[*]

ANDREW T. BARKER[†], STEPHAN V. GELEVER[‡], CHAK S. LEE[†], SARAH V. OSBORN[†], AND
PANAYOT S. VASSILEVSKI[†,‡]

Abstract. We extend previously developed two-level coarsening procedures for graph Laplacian problems written in a mixed saddle point form to the fully recursive multilevel case. The resulting hierarchy of discretizations gives rise to a hierarchy of upscaled models, in the sense that they provide approximation in the natural norms (in the mixed setting). This property enables us to utilize them in three applications: (i) as an accurate reduced model, (ii) as a tool in multilevel Monte Carlo simulations (in application to finite volume discretizations), and (iii) for providing a sequence of nonlinear operators in FAS (full approximation scheme) for solving nonlinear pressure equations discretized by the conservative two-point flux approximation. We illustrate the potential of the proposed multilevel technique in all three applications on a number of popular benchmark problems used in reservoir simulation.

Key words. Graph Laplacian, algebraic multigrid, finite volume methods, numerical upscaling, multilevel Monte Carlo, full approximation scheme

AMS subject classifications. 65N55, 65N08, 65F15

1. Introduction. The massive amount of data available today makes it typically impractical to take into account all of the data for decision making, so that it is becoming increasingly important to extract useful information from raw data effectively and efficiently. For example, in reservoir simulations, geological models are often very detailed so that direct simulations on these fine scale models are prohibitively expensive. Traditionally, rock properties (e.g. porosity, permeability) in fine scale geological models are upscaled or homogenized to a coarser resolution that is suitable for simulations [36, 35, 18, 9]. Other than upscaling and homogenization, several sophisticated methods for constructing coarse models were proposed in the past two decades, including (generalized) multiscale finite element methods [19, 15, 14, 4], multiscale finite volume methods [21, 22, 33], heterogeneous multiscale methods [13, 3], and (algebraic) multigrid [28, 24, 34, 23, 7]. Among these methodologies, generalized multiscale finite element methods [14] and spectral AMGe [8, 23, 7] share the idea of constructing coarse spaces via local spectral problems, which allows the user to adjust approximation quality of the coarse space by altering the number of local basis functions (without changing the coarse grid). For exceedingly large problems, multilevel coarsening is desirable to provide the user a range of accuracy levels. In addition, for multilevel Monte Carlo simulations and multilevel solvers for nonlinear problems, nested hierarchies of various resolutions are indispensable. To produce nested hierarchical models, multigrid methods are the obvious choice due to their intrinsic multilevel nature.

In [7], a finite volume discretization using two-point flux approximation (TPFA) for Darcy's flow problems is interpreted as a graph Laplacian, and coarse reservoir models are constructed by algebraically coarsening the Laplacian. The input required by this algebraic method is essentially the linear system only, which makes the method very easy to apply. Moreover, it enables flexible local enrichment of the coarse approximation space, and hence produces coarse models with a range of accuracies. In this paper, our goal is to extend the two-level method developed in [7] to a multilevel method, and demonstrate the applications of the method in numerical upscaling, multilevel Monte Carlo simulations (of finite volume problems), and nonlinear multigrid solvers (full approximation scheme). It is shown in [7] that the coarse spaces have approximation properties in the natural norms (in mixed setting), which is the key for the targeted applications to work effectively.

The coarse system produced by the two-level spectral coarsening method in [7] is not a graph Laplacian, so it seems that the two-level coarsening method cannot be applied directly to coarsen the coarse system further. But if we associate degrees of freedom to vertices/edges of the graph, and think of the coarsening with respect to degrees of freedom rather than with respect to the graph topology, then the coarsening method can actually be applied recursively in a natural way. The main idea is to aggregate degrees of

---

[†]Center for Applied Scientific Computing, Lawrence Livermore National Laboratory, Livermore, CA 94550 (atb@llnl.gov, cslee@llnl.gov, osborn9@llnl.gov, panayot@llnl.gov)

[‡]The Fariborz Maseeh Department of Mathematics and Statistics, Portland State University, Portland, OR 97201 (stephan.gelever@gmail.com, panayot@pdx.edu)

freedom based on their associated vertices/edges of the graph so that the local problems are well-defined. In the finest level, there is a one-to-one correspondence between the vertex/edge degrees of freedom and vertices/edges of the original graph; in coarse levels, the relation between coarse degrees of freedom and vertices/edges of the coarse graph is available from the coarsening procedure. Hence, the required input of the overall multilevel coarsening procedure is still the original fine scale system only.

Since our spectral coarsening method involves solving local eigen and extension problems, the setup of the hierarchy is quite expensive (compared with the classic algebraic multigrid [31]). Therefore, in multilevel Monte Carlo simulations and FAS, it is crucial that the coarse spaces can be reused even when the problem parameters (e.g. permeability coefficient, source term) are changed. This is achieved by first storing the local coarse matrices, and when the coefficient is updated, the local matrices are scaled and then assembled into the final global system. Another issue is that the coarse systems generated by our coarsening method are of saddle point form, even though the original problem (graph Laplacian) is symmetric positive-definite. Consequently, in order to have faster simulation on coarse levels, we need an efficient solver for the coarse saddle point problems. To this end, we again make use of the availability of local matrices (during the coarsening steps) and transform the saddle point problems into symmetric positive-definite problems via hybridization, which is a classic technique for solving discrete problems arising from mixed finite element discretization of partial differential equations [5]. Recently in [26], and [12], an algebraic multigrid with some proper diagonal rescaling is demonstrated be a competitive solver for the hybridized system. Because the solver is constructed algebraically, it can be applied to solve saddle point problems that are not coming from finite element discretizations, which is the case in the current paper.

The rest of the paper is organized as follows. In Section 2, we state the problem of interest and notation. Then the multilevel spectral coarsening algorithm will be presented in detail in Section 3. Lastly, we illustrate some applications of the coarsening algorithm in Section 4 and conclude the paper in Section 5.

2. Some setup and notation. Throughout the paper, we use the convention that level $\ell = 0$ corresponds to the finest resolution while $\ell = L$ corresponds to the coarsest resolution. The number of elements in a set S will be denoted by $\#S$. For any matrix A, its null space and range space are denoted by $\text{Null}\,(A)$ and $\text{Range}\,(A)$ respectively.

We have a connected and undirected (fine) graph $G^0 = (V^0, E^0)$, where V^0 and E^0 are the set of vertices and edges respectively. Let $\mathcal{L}^0$ be a weighted graph Laplacian associated with G^0. It is well-known that $\mathcal{L}^0$ can be decomposed as $\mathcal{L}^0 = D^0(M^0)^{-1}(D^0)^T$, where M^0 is a diagonal matrix containing the inverse of edge weights and $(D^0)^T$ resembles some discrete gradient. More precisely, each row of $(D^0)^T$, which corresponds to some edge connecting vertex v_i^0 to v_j^0, has exactly two nonzeros, 1 at the corresponding column of v_i^0 and -1 at the other. By considering this decomposition, the graph Laplacian problem

$$\mathcal{L}^0 \mathbf{u}^0 = \mathbf{f}^0$$

is equivalent to the mixed graph Laplacian problem

$$\begin{bmatrix} M^0 & (D^0)^T \\ D^0 & \end{bmatrix} \begin{bmatrix} \boldsymbol{\sigma}^0 \\ \mathbf{u}^0 \end{bmatrix} = \begin{bmatrix} 0 \\ -\mathbf{f}^0 \end{bmatrix}$$

where $\mathbf{u}^0$ is in the vertex space $\mathbf{U}^0 := \mathbb{R}^{\#V^0}$ and $\boldsymbol{\sigma}^0$ is in the edge space $\boldsymbol{\Sigma}^0 := \mathbb{R}^{\#E^0}$. The coarsening scheme in this paper is based on the mixed setting.

Our main objective is to construct two sequences of matrices $\{M^\ell\}_{\ell=0}^L$ and $\{D^\ell\}_{\ell=0}^L$ for defining coarse mixed graph Laplacians. This is accomplished by constructing interpolation matrices $\{P_\sigma^\ell\}_{\ell=0}^{L-1}$ and $\{P_u^\ell\}_{\ell=0}^{L-1}$ and applying the recursive definition:

$$\begin{bmatrix} M^{\ell+1} & (D^{\ell+1})^T \\ D^{\ell+1} & \end{bmatrix} = \begin{bmatrix} P_\sigma^\ell & \\ & P_u^\ell \end{bmatrix}^T \begin{bmatrix} M^\ell & (D^\ell)^T \\ D^\ell & \end{bmatrix} \begin{bmatrix} P_\sigma^\ell & \\ & P_u^\ell \end{bmatrix}, \quad \text{for } \ell = 0, 1, \ldots, L-1. \qquad (2.1)$$

For multilevel Monte Carlo simulations and nonlinear multigrid solvers, we will also need projection operators that take quantities from a given level to a coarser level. To this end, we will construct $\{Q_\sigma^\ell\}_{\ell=0}^{L-1}$ and $\{Q_u^\ell\}_{\ell=0}^{L-1}$ such that Q_u^ℓ is a left inverse of P_u^ℓ and Q_σ^ℓ is a left inverse of P_σ^ℓ.

2

3. Multilevel spectral coarsening of graph Laplacians. In [7], a two-level spectral coarsening algorithm for the graph Laplacian $D^0(M^0)^{-1}(D^0)^T$ is introduced. The algorithm constructs interpolation matrices P_u^0 and P_σ^0 for vertex and edge spaces respectively. Then the coarse level operators M^1 and D^1 are defined through the "RAP" procedure (2.1) when $\ell = 0$. Note that the coarse operator $D^1(M^1)^{-1}(D^1)^T$ is not a graph Laplacian in the usual sense (for example, its row sums are not necessarily zero). Therefore, the coarsening procedure in [7] cannot be applied directly to $D^1(M^1)^{-1}(D^1)^T$. To extend the two-level algorithm to a recursive multilevel one, the key idea is to make a distinction between graph entities (vertices and edges) and their associated vector spaces. When defining local problems, we consider aggregation of degrees of freedom (dofs) instead of entities. Before we dive into the algorithm in detail, we shall give a high-level view of the coarsening procedure:

1. Coarsen graph entities: aggregate vertices and edges.
2. Aggregate vertex dofs and edge dofs based on aggregation of vertices and edges.
3. Solve spectral problems on aggregates of fine dofs to define coarse vertex dofs (for good approximation).
4. Define coarse edge dofs associated with edges connecting aggregates (for coupling of coarse dofs).
5. Define coarse edge dofs associated with aggregates of vertices (for stability of the coarse problem).

REMARK 3.1. *We stress here that, when $L = 1$, the multilevel algorithm in the current paper will result in exactly the two-level method proposed in [7].*
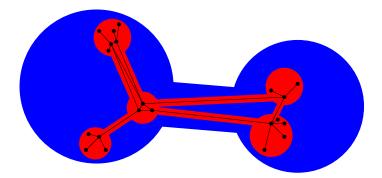

FIG. 3.1. *An example of a 3-level graph hierarchy. Black: level 0, red: level 1, blue: level 2.*

3.1. Graph hierarchy. Starting with the original graph G^0, we will generate a hierarchy of graphs $\{G^\ell\}_{\ell=0}^L$ with $L + 1$ levels of resolution. Given a graph $G^\ell = (V^\ell, E^\ell)$ on level ℓ, we construct a coarser graph $G^{\ell+1}$ as follows. First, we partition the set of vertices V^ℓ into non-overlapping aggregates of vertices $\{\mathcal{A}_j^\ell\}$ such that the subgraph within each aggregate is connected. We identify the set of aggregates on level ℓ as the set of vertices on level $\ell + 1$, that is,

$$V^{\ell+1} = \{v_j^{\ell+1}\} := \{\mathcal{A}_j^\ell\}.$$

Aggregates inherit their connections from the finer level. In particular we say that $\mathcal{A}_i^\ell$ is *connected* to $\mathcal{A}_j^\ell$ if there is some edge $e^\ell = (v_i^\ell, v_j^\ell)$ of the finer graph G^ℓ such that $v_i^\ell \in \mathcal{A}_i^\ell$ and $v_j^\ell \in \mathcal{A}_j^\ell$, and if such a connection exists we define a *face* as the collection of all such edges

$$\mathcal{F}^\ell := \left\{ e^\ell = (v_i^\ell, v_j^\ell) \in E^\ell : v_i^\ell \in \mathcal{A}_i^\ell \text{ and } v_j^\ell \in \mathcal{A}_j^\ell \right\}.$$

Note that the faces on level ℓ can naturally be identified as the set of edges on level $\ell + 1$

$$E^{\ell+1} := \left\{ e^{\ell+1} := \mathcal{F}^\ell = (\mathcal{A}_i^\ell, \mathcal{A}_j^\ell) = (v_i^{\ell+1}, v_j^{\ell+1}) \right\}.$$

This completes the construction of a coarser graph $G^{\ell+1} = (V^{\ell+1}, E^{\ell+1})$ from G^ℓ. This process can easily be applied recursively, resulting in a hierarchy of graphs, as illustrated in Figure 3.1.

3.2. Graph spaces. While the construction in Section 3.1 preserves a graph structure on each level, the spectral coarsening method in [7] does *not* preserve a graph Laplacian structure. The main problem is that the spectral method allows more than one dof per entity in the coarse vertex and edge spaces.

3

Consequently, the matrix graph of the coarse Laplacian may have some pair of vertices that are connected through 3 or more edges. In other words, we cannot extract a normal undirected graph using the coarse Laplacian operator alone. To remedy this situation, we need to independently keep track of the aggregation of graph entities (as in Section 3.1), and their associated dofs. To this end we introduce local graph spaces, whose definitions are motivated by the coarse spaces of the two-level algorithm:

- $\mathbf{U}^\ell(v_i^\ell)$: span of level-$\ell$ vertex-based dofs associated with vertex v_i^ℓ.
- $\boldsymbol{\Sigma}^\ell(e_i^\ell)$: span of level-$\ell$ edge-based dofs associated with edge e_i^ℓ.
- $\boldsymbol{\Sigma}^\ell(v_i^\ell)$: span of level-$\ell$ edge-based dofs associated with vertex v_i^ℓ.

In our local spectral coarsening algorithm, we will need to extract submatrices corresponding to dofs associated with certain aggregates of vertices/edges, so it will be useful to define local graph spaces on some arbitrary aggregate of vertices $\mathcal{B}$ and face $\mathcal{F}$ (aggregate of edges):

$$\mathbf{U}^\ell(\mathcal{B}) := \bigoplus_{v_i^\ell \in \mathcal{B}} \mathbf{U}^\ell(v_i^\ell), \qquad \boldsymbol{\Sigma}^\ell(\mathcal{B}) := \left(\bigoplus_{e_i^\ell \in E^\ell(\mathcal{B})} \boldsymbol{\Sigma}^\ell(e_i^\ell) \right) \bigoplus \left(\bigoplus_{v_i^\ell \in \mathcal{B}} \boldsymbol{\Sigma}^\ell(v_i^\ell) \right),$$

$$\boldsymbol{\Sigma}^\ell(\mathcal{F}) := \bigoplus_{e_i^\ell \in \mathcal{F}} \boldsymbol{\Sigma}^\ell(e_i^\ell), \qquad \partial\boldsymbol{\Sigma}^\ell(\mathcal{B}) := \bigoplus_{e_i^\ell \in \partial E^\ell(\mathcal{B})} \boldsymbol{\Sigma}^\ell(e_i^\ell),$$

where

$$E^\ell(\mathcal{B}) := \left\{ e = (v_i, v_j) \in E^\ell : v_i, v_j \in \mathcal{B} \right\} \quad \text{and} \quad \partial E^\ell(\mathcal{B}) := \left\{ e = (v_i, v_j) \in E^\ell : v_i \in \mathcal{B}, v_j \notin \mathcal{B} \right\}.$$

The global spaces are simply when $\mathcal{B}$ is V^ℓ, the set of all vertices in level ℓ:

$$\mathbf{U}^\ell := \mathbf{U}^\ell(V^\ell), \qquad \boldsymbol{\Sigma}^\ell := \boldsymbol{\Sigma}^\ell(V^\ell).$$

On the finest level ($\ell = 0$), each of $\mathbf{U}^0(v_i^0)$ and $\boldsymbol{\Sigma}^0(e_i^0)$ is just taken to be the space of real numbers $\mathbb{R}$, and $\boldsymbol{\Sigma}^0(v_i^0)$ is empty. Hence, the global spaces match with our earlier definition $\mathbf{U}^0 = \bigoplus_{v_i^0 \in V^0} \mathbf{U}^0(v_i^0) = \mathbb{R}^{\#V^0}$, and $\boldsymbol{\Sigma}^0 = \bigoplus_{e_i^0 \in E^0} \boldsymbol{\Sigma}^0(e_i^0) = \mathbb{R}^{\#E^0}$. On coarser levels ($\ell > 0$), $\mathbf{U}^\ell(v_i^\ell)$ contains eigenvectors of some local spectral problem associated with $v_i^\ell = \mathcal{A}_i^{\ell-1}$. $\boldsymbol{\Sigma}^\ell(e_i^\ell)$ is spanned by the trace extensions associated with $e_i^\ell = \mathcal{F}_i^{\ell-1}$, and $\boldsymbol{\Sigma}^\ell(v_i^\ell)$ is spanned by the bubbles associated with $v_i^\ell = \mathcal{A}_i^{\ell-1}$. The precise definition and detailed construction of all these components in the coarse levels will be given in Section 3.3– 3.4.

Now, for an aggregate of vertices $\mathcal{B}$ and a face $\mathcal{F}$, we define its associated submatrices:

$$\begin{aligned} M_{\mathcal{B}}^\ell &:= M_{\boldsymbol{\Sigma}^\ell(\mathcal{B}), \boldsymbol{\Sigma}^\ell(\mathcal{B})}^\ell, \quad & M_{\partial\mathcal{B}}^\ell &:= M_{\boldsymbol{\Sigma}^\ell(\mathcal{B}), \partial\boldsymbol{\Sigma}^\ell(\mathcal{B})}^\ell, \quad & M_{\mathcal{B},\mathcal{F}}^\ell &:= M_{\boldsymbol{\Sigma}^\ell(\mathcal{B}), \boldsymbol{\Sigma}^\ell(\mathcal{F})}^\ell, \\ D_{\mathcal{B}}^\ell &:= D_{\mathbf{U}^\ell(\mathcal{B}), \boldsymbol{\Sigma}^\ell(\mathcal{B})}^\ell, \quad & D_{\partial\mathcal{B}}^\ell &:= D_{\mathbf{U}^\ell(\mathcal{B}), \partial\boldsymbol{\Sigma}^\ell(\mathcal{B})}^\ell, \quad & D_{\mathcal{B},\mathcal{F}}^\ell &:= D_{\mathbf{U}^\ell(\mathcal{B}), \boldsymbol{\Sigma}^\ell(\mathcal{F})}^\ell. \end{aligned} \tag{3.1}$$

Here, $M_{\boldsymbol{\Sigma}^\ell(\mathcal{B}), \boldsymbol{\Sigma}^\ell(\mathcal{F})}^\ell$ is the submatrix of M^ℓ whose rows and column are respectively restricted to $\boldsymbol{\Sigma}^\ell(\mathcal{B})$ and $\boldsymbol{\Sigma}^\ell(\mathcal{F})$; other submatrices are defined similarly. An analogous notion for subvectors associated with aggregates and faces will also be used.

REMARK 3.2. *The concept of graph spaces is an analogue to finite element spaces in the context of discretizations of partial differential equations, in which each dof is associated with some geometrical entity (e.g. element, face) of a mesh. The present graph-based construction can be seen as a version of the de Rham sequence corresponding only to $H(div)$ and L_2 degrees of freedom on an agglomerated mesh. See [24] for detailed descriptions of such mesh-based constructions.*

3.3. Vertex degrees of freedom. Following [7], when generating coarse vertex degrees of freedom, we consider the local matrices associated with the neighborhood of each aggregate $N(\mathcal{A}_i^\ell)$. That is, in this case $\mathcal{B}$ in (3.1) is taken to be $\mathcal{B} = N(\mathcal{A}_i^\ell)$, where $N(\mathcal{A}_i^\ell)$ is defined to be the set of all vertices connected by an edge to a vertex in $\mathcal{A}_i^\ell$. Define

$$\mathcal{L}_{N(\mathcal{A}_i^\ell)} := D_{N(\mathcal{A}_i^\ell)}^\ell \left(M_{N(\mathcal{A}_i^\ell)}^\ell \right)^{-1} \left(D_{N(\mathcal{A}_i^\ell)}^\ell \right)^T$$

and solve the eigenvalue problem

$$\mathcal{L}_{N(\mathcal{A}_i^\ell)} \mathbf{q} = \lambda \mathbf{q}. \tag{3.2}$$

We select a few eigenvectors corresponding to small eigenvalues, restrict them to $\mathcal{A}_i^\ell$, use SVD to remove possible linear dependence, and choose the remaining linearly independent vectors $\tilde{\mathbf{q}}$ as columns of the matrix $P_{u,\mathcal{A}_i^\ell}^\ell$. By restricting local basis functions to aggregates $\mathcal{A}_i^\ell$, the support of basis functions associated with different aggregates will have no overlap. The final interpolation matrix P_u^ℓ for the vertex space is obtained by diagonally concatenating $P_{u,\mathcal{A}_i^\ell}^\ell$ for all aggregates $\mathcal{A}_i^\ell \in V^{\ell+1}$:

$$
P_u^\ell := \begin{bmatrix} \ddots & & \\ & P_{u,\mathcal{A}_i^\ell}^\ell & \\ & & \ddots \end{bmatrix}.
$$

The local coarse vertex space of each aggregate $\mathcal{A}_i^\ell$ (vertex in level $\ell+1$) is the column space of $P_{u,\mathcal{A}_i^\ell}^\ell$:

$$
\mathbf{U}^{\ell+1}(v_i^{\ell+1}) = \mathbf{U}^{\ell+1}(\mathcal{A}_i^\ell) := \operatorname{Range}\left(P_{u,\mathcal{A}_i^\ell}^\ell \right).
$$

Since P_u^ℓ is block diagonal and the columns of each block are orthonormal, we simply define the projection operator for the coarse vertex space as $\pi_u^\ell := P_u^\ell Q_u^\ell$, where

$$
Q_u^\ell := \left(P_u^\ell \right)^T.
$$

Clearly, Q_u^ℓ is defined locally in each aggregate and

$$
Q_u^\ell P_u^\ell = I_u^{\ell+1}. \tag{3.3}
$$

REMARK 3.3. *Let $\mathbf{1}^0$ be the constant vector of 1's in $\mathbf{U}^0 = \mathbb{R}^{\#V^0}$ (i.e., $\mathbf{1}^0 := [1,1,\cdots,1]^T$). Then, its projection in coarse levels are defined recursively as*

$$
\mathbf{1}^\ell := Q_u^{\ell-1}\,\mathbf{1}^{\ell-1}, \quad for\ \ell = 1,\ldots,L. \tag{3.4}
$$

From Lemma A.5 and Lemma A.4 in the appendix, we know our construction guarantees that $\mathbf{1}_{\mathcal{A}_i^\ell}^\ell$ (the restriction of $\mathbf{1}^\ell$ to $\mathcal{A}_i^\ell$) is always in $\operatorname{Range}\left(P_{u,\mathcal{A}_i^\ell}^\ell \right)$. Therefore, $P_{u,\mathcal{A}_i^\ell}^\ell$ is of the form

$$
P_{u,\mathcal{A}_i^\ell}^\ell = \begin{bmatrix} \mathbf{q}_{\mathcal{A}_i^\ell}^{PV} & P_{u,\mathcal{A}_i^\ell}^{\ell,NPV} \end{bmatrix}, \tag{3.5}
$$

where columns of $P_{u,\mathcal{A}_i^\ell}^{\ell,NPV}$ are orthogonal to $\mathbf{1}_{\mathcal{A}_i^\ell}^\ell$ and $\mathbf{q}_{\mathcal{A}_i^\ell}^{PV}$ is the unit vector parallel to $\mathbf{1}_{\mathcal{A}_i^\ell}^\ell$:

$$
\mathbf{q}_{\mathcal{A}_i^\ell}^{PV} = \left(\|\mathbf{1}_{\mathcal{A}_i^\ell}^\ell\|^{-1} \right) \mathbf{1}_{\mathcal{A}_i^\ell}^\ell. \tag{3.6}
$$

Note that we use the superscript "PV" because $\mathbf{q}_{\mathcal{A}^\ell}^{PV}$ are the coarse basis in an earlier work of Pasciak and Vassilevski [30]. Together with "PV" traces that will be defined in the next section, they are the building blocks of a pair of minimal coarse spaces such that the coarse problem is inf-sup stable, cf. condition (3.7).

Figures 3.2 and 3.3 illustrate some example coarse vertex degrees of freedom in a 3-level hierarchy of some random graph. Notice that the first basis of any aggregate on any level is always constant.

3.4. Edge degrees of freedom. There are two types of coarse edge degrees of freedom, one is called trace extensions while the other one is called bubbles. Each trace extension is associated with a face, and is supported in the aggregates sharing the face. They are the only components of the coarse space that connect neighboring aggregates and hence couple the global problem. On the other hand, each bubble is associated with an aggregate. The support of a bubble is in one and only one aggregate, hence the name bubble.
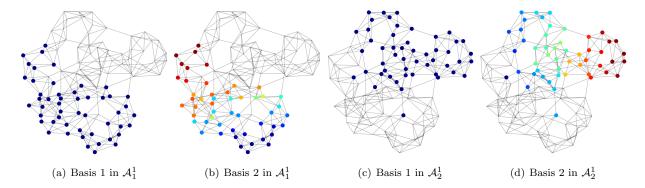
(a) Basis 1 in $\mathcal{A}_1^1$ (b) Basis 2 in $\mathcal{A}_1^1$ (c) Basis 1 in $\mathcal{A}_2^1$ (d) Basis 2 in $\mathcal{A}_2^1$

FIG. 3.2. *Examples of level 1 coarse vertex degrees of freedom of some random graph*

(a) Basis 1 in $\mathcal{A}_1^2$ (b) Basis 2 in $\mathcal{A}_1^2$ (c) Basis 3 in $\mathcal{A}_1^2$ (d) Basis 4 in $\mathcal{A}_1^2$
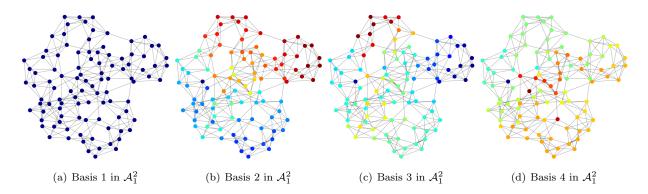
FIG. 3.3. *Examples of level 2 coarse vertex degrees of freedom of some random graph*

3.4.1. Trace extensions. The idea here is to define some representative vectors (which are what we call traces) that live on faces, and then extend them harmonically into the neighboring aggregates. It suffices to focus on one face because trace extensions are associated with faces. Recall that by definition, each face is associated with two aggregates: $\mathcal{F}^\ell = (\mathcal{A}_i^\ell, \mathcal{A}_j^\ell)$. For each associated aggregate and each eigenvector $\mathbf{q}$ selected when solving (3.2), we define the trace

$$\left(\left(M_{N(\mathcal{A}^\ell)}^\ell \right)^{-1} \left(D_{N(\mathcal{A}^\ell)}^\ell \right)^T \mathbf{q} \right) \Big|_{\boldsymbol{\Sigma}^\ell(\mathcal{F}^\ell)} .$$

Also, for inf-sup stability, we need a trace $\boldsymbol{\sigma}_{\mathcal{F}^\ell}^{PV}$ (which we call it the PV trace, cf. Remark 3.3) that satisfies

$$(\mathbf{q}_{\mathcal{A}_i^\ell}^{PV})^T D_{\mathcal{A}_i^\ell, \mathcal{F}^\ell}^\ell \, \boldsymbol{\sigma}_{\mathcal{F}^\ell}^{PV} \neq 0. \tag{3.7}$$

Because there is no guarantee that $\left(M_{N(\mathcal{A}^\ell)}^\ell \right)^{-1} \left(D_{N(\mathcal{A}^\ell)}^\ell \right)^T \mathbf{q}$ will satisfy (3.7), $\boldsymbol{\sigma}_{\mathcal{F}^\ell}^{PV}$ is constructed separately. One computationally inexpensive choice is to take

$$\boldsymbol{\sigma}_{\mathcal{F}^\ell}^{PV} := (D_{\mathcal{A}_i^\ell, \mathcal{F}^\ell}^\ell)^T \mathbf{q}_{\mathcal{A}_i^\ell}^{PV} .$$

Another choice of PV trace takes edge weights into account and requires solving a local problem in $\mathcal{A}_i^\ell \cup \mathcal{A}_j^\ell$. The detailed construction of the second choice will be given in Appendix B. We observe empirically that the second option typically produces higher quality coarse spaces when the edge weights have high contrast (which is usually the case for graphs coming from reservoir simulations), so the numerical results in this paper are generated using the second option.

After collecting all the traces from both sides of the face as well as the PV trace, we perform SVD to remove possible linear dependence due to restriction and union. Let the set of traces associated with $\mathcal{F}^\ell$

after SVD be $Tr(\mathcal{F}^\ell)$. Then for each trace $\boldsymbol{\sigma}_{\mathcal{F}^\ell} \in Tr(\mathcal{F}^\ell)$, we solve

$$\begin{bmatrix} M_\mathcal{A}^\ell & (D_\mathcal{A}^\ell)^T \\ D_\mathcal{A}^\ell & \end{bmatrix} \begin{bmatrix} \boldsymbol{\sigma}_\mathcal{A} \\ \mathbf{u} \end{bmatrix} = \begin{bmatrix} -M_{\mathcal{A},\mathcal{F}^\ell}^\ell \boldsymbol{\sigma}_{\mathcal{F}^\ell} \\ c_\mathcal{A} \mathbf{q}_\mathcal{A}^{PV} - D_{\mathcal{A},\mathcal{F}^\ell}^\ell \boldsymbol{\sigma}_{\mathcal{F}^\ell} \end{bmatrix}. \tag{3.8}$$

Here, $c_\mathcal{A}$ is the constant such that $(\mathbf{q}_\mathcal{A}^{PV})^T (c_\mathcal{A} \mathbf{q}_\mathcal{A}^{PV} - D_{\mathcal{A},\mathcal{F}^\ell}^\ell \boldsymbol{\sigma}_{\mathcal{F}^\ell}) = 0$. By (3.6), $\|\mathbf{q}_\mathcal{A}^{PV}\| = 1$, so

$$c_\mathcal{A} = (\mathbf{q}_\mathcal{A}^{PV})^T D_{\mathcal{A},\mathcal{F}^\ell}^\ell \boldsymbol{\sigma}_{\mathcal{F}^\ell}. \tag{3.9}$$

Each trace extension is then formed by glueing together $\boldsymbol{\sigma}_{\mathcal{F}^\ell}$, $\boldsymbol{\sigma}_{\mathcal{A}_i^\ell}$, and $\boldsymbol{\sigma}_{\mathcal{A}_j^\ell}$. These extended traces are what ultimately end up as columns of P_σ^ℓ.

REMARK 3.4. *Note that adding $c_\mathcal{A} \mathbf{q}_\mathcal{A}^{PV}$ to the right hand side of (3.8) guarantees the solvability of (3.8) because the nullspace of $(D_\mathcal{A}^\ell)^T$ is spanned by $\mathbf{q}_\mathcal{A}^{PV}$ (see Lemma A.5).*

3.4.2. Bubbles. The definition of bubbles are exactly analogous to the two-level case — solve (3.8) with the right-hand-side vector replaced by $\begin{bmatrix} 0 \\ \mathbf{q} \end{bmatrix}$ where $\mathbf{q}$ are just the columns of $P_{u,\mathcal{A}^\ell}^{\ell,NPV}$. Let $P_{\sigma,\mathcal{A}^\ell}^{\ell,B}$ be the matrix whose columns are all bubbles in $\mathcal{A}^\ell$. Thus, by construction
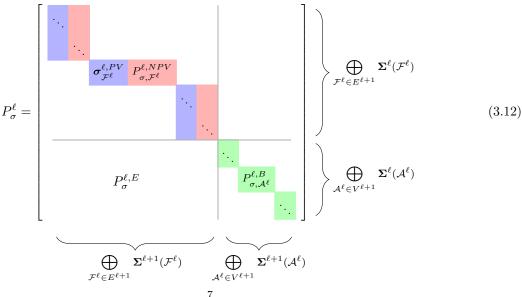
$$D_\mathcal{A}^\ell P_{\sigma,\mathcal{A}^\ell}^{\ell,B} = P_{u,\mathcal{A}^\ell}^{\ell,NPV}. \tag{3.10}$$

We need these bubbles in the coarse edge space to make sure the coarse saddle point problem is inf-sup stable.

3.4.3. Structure of P_σ^ℓ. To summarize Section 3.4.1 and 3.4.2, we give a high level view on the structure of the edge space interpolation matrix P_σ^ℓ here. Also, for the purpose of a clearer matchup when defining the projection operator later, different components are highlighted with different colors.

First, on each face $\mathcal{F}^\ell$, let $P_{\sigma,\mathcal{F}^\ell}^{\ell,T}$ be the matrix whose columns are all traces in $Tr(\mathcal{F}^\ell)$:

$$P_{\sigma,\mathcal{F}^\ell}^{\ell,T} = \begin{bmatrix} \boldsymbol{\sigma}_{\mathcal{F}^\ell}^{PV} & P_{\sigma,\mathcal{F}^\ell}^{\ell,NPV} \end{bmatrix} \tag{3.11}$$

where columns of $P_{\sigma,\mathcal{F}^\ell}^{\ell,NPV}$ are "non-PV" traces on $\mathcal{F}^\ell$. Then the extension part of all traces are collected in $P_\sigma^{\ell,E}$. More precisely, for each column of $P_{\sigma,\mathcal{F}^\ell}^{\ell,T}$, which is a trace $\boldsymbol{\sigma}_{\mathcal{F}^\ell}$, the corresponding column in $P_\sigma^{\ell,E}$ is the extensions of $\boldsymbol{\sigma}_{\mathcal{F}^\ell}$ into the neighboring aggregates sharing $\mathcal{F}^\ell$ (cf. (3.8)). The final interpolation matrix P_σ^ℓ has the following structure:

$$\tag{3.12}$$

3.5. Coarse projections and their properties. We have defined the interpolation matrices P_u^ℓ and P_σ^ℓ, and the coarse vertex and edge spaces in previous sections. To use the hierarchy in multilevel Monte Carlo simulations and nonlinear multigrid solvers, we will also need projection operators that take quantities from a given level to a coarser level. We have already discussed the projection operator $Q_u^\ell = (P_u^\ell)^T$ for the vertex space in Section 3.3. Due to our aggregation-based coarsening, it turns out that projection operators for edge spaces can also be defined locally in each aggregate and face.

PROPOSITION 3.1. *There exists a locally constructed projection $\pi_\sigma^\ell := P_\sigma^\ell Q_\sigma^\ell$ to the coarse edge space , where $Q_\sigma^\ell : \mathbf{\Sigma}^\ell \to \mathbf{\Sigma}^{\ell+1}$ satisfies*

$$Q_\sigma^\ell P_\sigma^\ell = I_\sigma^{\ell+1}.$$

PROPOSITION 3.2. *On each level ℓ, we have*

$$Q_u^\ell D^\ell = D^{\ell+1} Q_\sigma^\ell. \tag{3.13}$$

In other words, the following commutative diagram holds:

$$
\begin{array}{ccc}
\mathbf{\Sigma}^\ell & \xrightarrow{\ D^\ell\ } & \mathbf{U}^\ell \\
Q_\sigma^\ell \downarrow & & \downarrow Q_u^\ell \\
\mathbf{\Sigma}^{\ell+1} & \xrightarrow{\ D^{\ell+1}\ } & \mathbf{U}^{\ell+1}
\end{array}
$$

Proof. The proposition is a direct consequence of Lemma A.3 and A.5. $\qquad\square$

We defer the detailed construction of Q_σ^ℓ, as well as the proof of Proposition 3.1 to Appendix A. With Proposition 3.2, the well-posedness of the coarse saddle point problems can be shown as in Section 7 of [7].

4. Applications of the multilevel hierarchy. In this section, we demonstrate some applications of our multilevel spectral coarsening algorithm, including multilevel upscaling, multilevel Monte Carlo simulations, and nonlinear multigrid solvers. The numerical results here are obtained using the open source software *smoothG* [6], which contains a C++ implementation with MPI parallelization of the multilevel spectral coarsening algorithm. It is available on the github repository: `github.com/LLNL/smoothG`. The graph visualizations in Section 4.1 are generated by NetworkX [16], while the visualizations for PDE related applications in Section 4.2 and 4.3 are generated using MFEM [2] and GLVis [1].

4.1. Multilevel upscaling versus two-level. We consider an internet graph from `http://opte.org`. The graph has 35638 vertices and 42827 edges, and its Fielder vector is depicted in Figure 4.1. In this example, the right hand side $\mathbf{f}$ is set to be the Fiedler vector (scaled by the smallest positive eigenvalue of the associated graph Laplacian). We apply our multilevel upscaling algorithm to obtain 4-level hierarchies of approximation spaces by varying the number of local eigenvectors ($m_\mathcal{A}$) selected as basis functions. The coarsening factor is 16 for all hierarchies. The upscaling errors are summarized in Table 4.1 and Figure 4.2. In particular, Figure 4.2 clearly shows that the upscaling errors decay rapidly in all levels when more local eigenvectors are added to the coarse spaces.

In Figure 4.3, the approximated Fiedler vector is shown on each coarse level. When $m_\mathcal{A} = 1$, we see that the coarse approximations start to deviate from the fine level solution (Figure 4.1); while the coarse solutions corresponding to $m_\mathcal{A} = 4$ approximate the fine level solution very well across all levels.

4.2. Multilevel Monte Carlo for finite volume schemes. In [29], a multilevel Monte Carlo method for forward uncertainty quantification was shown to have efficiency gains over standard Monte Carlo in a mixed finite element setting. Here we present similar results for a finite volume method, which can be understood as a graph problem by considering each finite volume as a graph vertex and each face between volumes as a graph edge, see [7] for details.
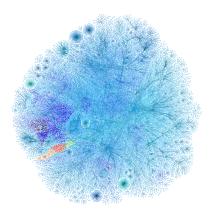
8

FIG. 4.1. *Fiedler vector of an internet graph (dim(*$\mathbf{U}^0$*) = 35638).*

TABLE 4.1
Relative upscaling errors in all levels (%)

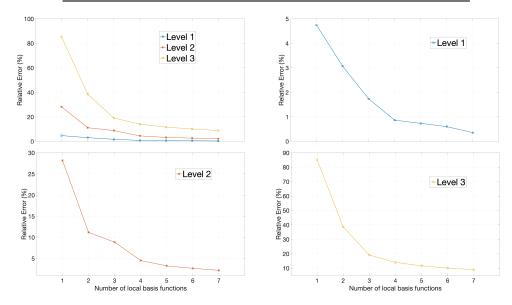|	$m_{\mathcal{A}}$						
	1	2	3	4	5	6	7
Level 1	4.744	3.075	1.739	0.8672	0.7362	0.6041	0.3576
Level 2	28.20	11.19	8.859	4.529	3.260	2.679	2.213
Level 3	85.27	38.55	19.12	14.08	11.64	10.16	8.897

FIG. 4.2. *Rapid decay of upscaling error in all levels as number of local basis functions increases.*

We are interested in the (linear) Darcy equations for a pressure p

$$- \nabla \cdot (\kappa(\omega)\nabla p) = f \text{ in } \Omega \tag{4.1}$$

with appropriate boundary conditions and forcing function f to be described in the test cases below. Here ω is an uncertain parameter describing uncertainty about the coefficient κ. The solution p also depends implicitly on ω, and that uncertainty is what we seek to quantify here; in particular, we want to propagate the uncertainty in κ via the forward problem (4.1) and compute statistics (e.g. expected value) of a functional of the solution p.
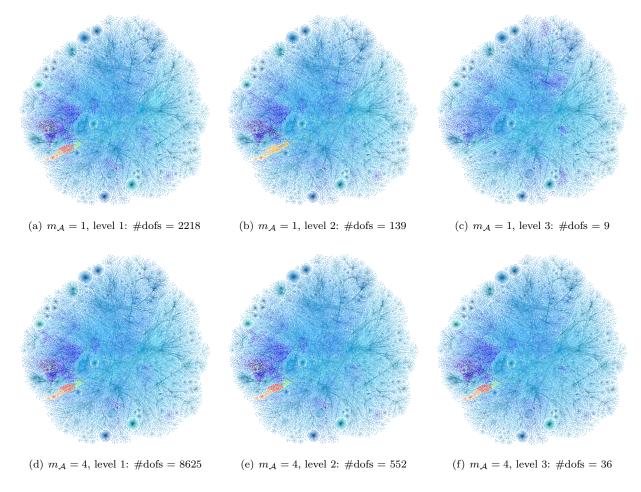
9

(a) $m_{\mathcal{A}} = 1$, level 1: #dofs = 2218   (b) $m_{\mathcal{A}} = 1$, level 2: #dofs = 139   (c) $m_{\mathcal{A}} = 1$, level 3: #dofs = 9

(d) $m_{\mathcal{A}} = 4$, level 1: #dofs = 8625   (e) $m_{\mathcal{A}} = 4$, level 2: #dofs = 552   (f) $m_{\mathcal{A}} = 4$, level 3: #dofs = 36

FIG. 4.3. *Multilevel coarse approximations of Fiedler vector; top: $m_{\mathcal{A}} = 1$, bottom: $m_{\mathcal{A}} = 4$.*

We assume $\kappa(\omega)$ is a lognormal random field, in particular $\kappa = \exp(\theta)$ where θ is a stationary isotropic Gaussian field with exponential covariance given by

$$\mathrm{cov}(x, y) = \sigma^2 \exp(-\gamma \|y - x\|)$$

where σ^2 is the marginal variance and γ is the correlation length.

To be very brief, the basic idea of multilevel Monte Carlo is to estimate the expected value $\mathbb{E}[Q_0]$ of Q_0, a quantity of interest defined on the finest level, by computing standard Monte Carlo estimators (using a sample and average technique) of the telescoping sum

$$\mathbb{E}[Q_0] = \mathbb{E}[Q_L] + \sum_{\ell=0}^{L-1} \mathbb{E}[Q_\ell - Q_{\ell+1}]. \tag{4.2}$$

The correction terms $Y_\ell = Q_\ell - Q_{\ell+1}$ have smaller variance than the original quantities of interest, and so a Monte Carlo estimator for Y_ℓ converges relatively quickly. At the same time, though the coarsest estimator for $\mathbb{E}[Q_L]$ may need a large number of samples, those samples are cheap because they are performed at a coarse level. See, for example, [11] for more details on the efficiency and accuracy of multilevel Monte Carlo.

Because the emphasis here is on the finite volume discretization, the setting for the experiments is somewhat simpler than in [29]. In particular, although we use the stochastic PDE sampler from [27, 29] to generate realizations of the lognormal random field, we do not embed the domain of interest in a larger domain, and so our sampler is subject to some modest boundary effects. There is no real theoretical or computational barrier to using such mesh embedding, however.
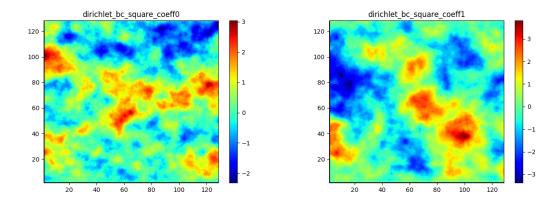
10

FIG. 4.4. *Two typical (log) permeability field realizations for the unit square example.*
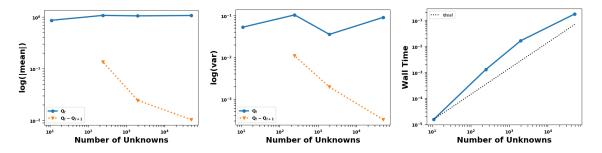

FIG. 4.5. *The mean (left) and variance (middle) of the quantity of interest (or difference in quantity of interest) at several levels of the hierarchy. The scaling (right) of the algorithm with respect to problem size is seen to be close to ideal.*

4.2.1. Unit square results. For demonstration, we employ multilevel Monte Carlo to estimate the expectation of a quantity of interest on the unit square $\Omega = [0,1]^2$ with boundary conditions $\nabla p \cdot n = 0$ on $x = 0, 1$, $p = -1$ on $y = 1$, and $p = 0$ on $y = 0$. The quantity of interest is the average flux on the top boundary, that is

$$Q = \int_{x=0}^{x=1} \kappa \nabla p(x, 1) \cdot n \, dx.$$

Some results are shown in Figures 4.4, 4.5, where we see that the correction terms have smaller variance than the fine-level estimator, as expected, and that the MLMC algorithm scales well with respect to problem size. Computational cost in comparison to a standard single-level Monte Carlo algorithm is shown in Table 4.2, where significant speedup is demonstrated.

4.2.2. Unit cube results. For another demonstration, we apply the multilevel Monte Carlo procedure on the unit cube $\Omega = [0,1]^3$ with boundary conditions $\nabla p \cdot n = 0$ on $x, y = 0, 1$, $p = -1$ on $z = 1$, and $p = 0$ on $z = 0$. Like the unit square problem, the quantity of interest is the average flux across the top boundary ($z = 1$).

Our results are shown in Figures 4.6, 4.7. Computational cost in comparison to a one-level Monte Carlo algorithm is reported in Table 4.3, where significant speedup is demonstrated in reaching a target mean square error using an adaptive algorithm where the number of samples computed on each level is chosen to minimize the variance for a fixed computational cost (see, e.g., [11] for details). In particular, the variance results in 4.7 show that the most expensive levels have the smallest variance, which is the key to the speedup in multilevel Monte Carlo as fewer samples are necessary on the most expensive levels.

4.2.3. Results inspired by the SPE10 dataset. The SPE10 dataset [10] is very often used as a challenging example of a complicated discontinuous coefficient. As given, this dataset is deterministic, and so we adapt one of its prominent features for our sampling and uncertainty quantification tests, namely

TABLE 4.2
Sample counts and computational cost for the unit square problem, including one-level and multilevel Monte Carlo for various tolerances on the mean-square error (MSE). For multilevel Monte Carlo, the sample counts are listed from finest to coarsest level.

	one-level		multilevel		
MSE	samples	time(s)	samples	time(s)	speedup
5e-03	19	3.1	[15, 15, 15, 19]	2.9	1.1
1e-03	72	11.5	[15, 15, 15, 110]	2.9	4.0
5e-04	129	20.7	[15, 15, 15, 1143]	2.9	7.1
1e-04	614	98.1	[15, 32, 282, 7739]	3.6	27.1
1e-05	6773	1081.8	[108, 785, 4525, 120036]	39.4	27.4

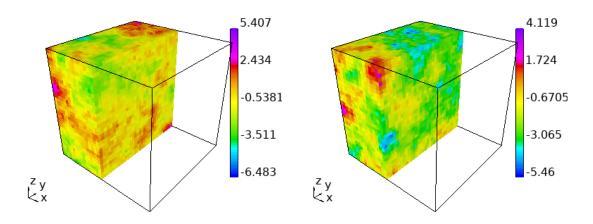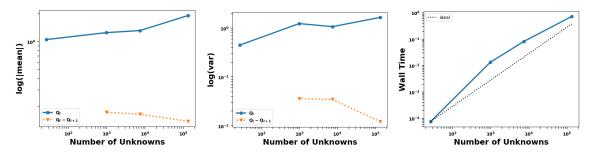FIG. 4.6. *Two typical (log) permeability field realizations for the unit cube example.*

FIG. 4.7. *The mean (left) and variance (middle) of the quantity of interest (or difference in quantity of interest) at several levels of the hierarchy. The scaling (right) of the algorithm with respect to problem size is seen to be close to ideal.*

TABLE 4.3
Sample counts and computational cost for the unit cube problem, including one-level and multilevel Monte Carlo for various tolerances on the mean-square error (MSE). For multilevel Monte Carlo, the sample counts are listed from finest to coarsest level.

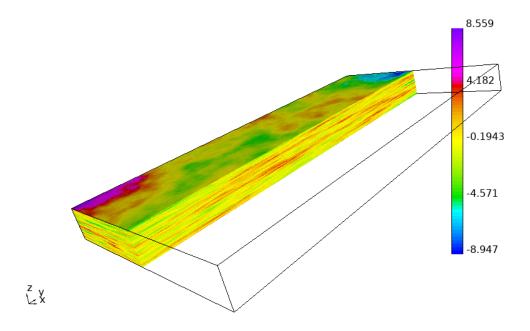|	one-level		multilevel		
MSE	samples	time(s)	samples	time(s)	speedup
5e-03	390	18.5	[15, 66, 146, 382]	2.4	7.7
1e-03	1514	70.9	[28, 347, 547, 1858]	9.0	7.9
5e-04	2771	128.1	[76, 575, 1109, 3495]	17.5	7.3
1e-04	13422	624.9	[434, 2395, 5298, 17890]	85.5	7.3

FIG. 4.8. *A typical (log) permeability field realization for the SPE10 example. The anisotropic covariance leads to a layering structure for the coefficient, where the correlation length is longer in the horizontal directions than in the vertical direction.*
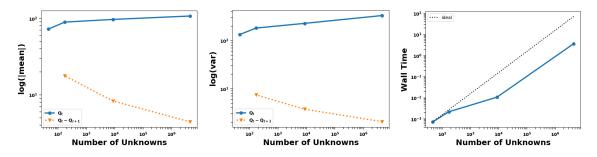

FIG. 4.9. *The mean (left) and variance (middle) of the quantity of interest (or difference in quantity of interest) at several levels of the hierarchy for the SPE10 example. The scaling (right) of the algorithm with respect to problem size is seen to be close to ideal.*

its layered structure. We produce samples with a similar structure on the SPE10 domain by imposing an anisotropic covariance structure, where the correlation length in horizontal directions is much longer than the correlation length in the vertical direction [32].

The results for this example are performed on a parallel computer with eight processors. We set $\kappa = 0.1$ in the vertical direction and $\kappa = 0.001$ in the horizontal direction. The boundary conditions are similar to those for the unit cube case, with $p = 0$ imposed on one side of the domain and $p = -1$ on the opposite side, while the flux across this latter boundary is the quantity of interest. We demonstrate the multilevel Monte Carlo method with some results shown in Figures 4.8, 4.9 which show that the multilevel Monte Carlo procedure is effective in reducing variance also for this problem with an anisotropic covariance.

In Table 4.4, we show the computational efficiency gains of the multilevel Monte Carlo algorithm for this problem. In this problem setting, the computational time is completely dominated by the number of samples taken on the finest level. As a result, MLMC results in very large performance gains, offering speedups of

13

TABLE 4.4

Sample counts and computational cost for the SPE10 problem, including one-level and multilevel Monte Carlo for various tolerances on the mean-square error (MSE). For multilevel Monte Carlo, the sample counts are listed from finest to coarsest level.

MSE	one-level		multilevel		
	samples	time(s)	samples	time(s)	speedup
1e+00	284	1021.1	[20, 20, 49, 274]	73.2	13.9
5e-01	537	1911.1	[20, 31, 128, 663]	72.7	26.3
2e-01	1323	4745.0	[20, 107, 454, 2657]	74.9	63.4
1e-01	2548	9175.3	[24, 650, 2448, 13303]	107.3	85.5
1e-02	—	—	[136, 5827, 20561, 108840]	671.1	—

greater than 50 compared to single-level Monte Carlo. The MLMC algorithm is able to reach an empirical mean-square error level of 0.01 in less time than the single-level algorithm takes to get to an (absolute) error of 1.0 in the quantity of interest.

4.3. Nonlinear multigrid (full approximation scheme). Consider the following nonlinear elliptic problem:

$$-\nabla \cdot \big(\kappa(p)\nabla p\big) = f \text{ in } \Omega,$$
$$p = 0 \text{ on } \partial\Omega. \tag{4.3}$$

The permeability function $\kappa(p)$ takes the form:

$$\kappa(p) = \kappa_0 e^{\alpha p}$$

where κ_0 is some reference permeability field when $p = 0$. As an example, we take the domain Ω to be the Egg model from [20], with $\alpha = 5$, $f = 1$, and κ_0 is realization 27 of the Egg model. The problem is discretized by the finite volume method with two-point flux approximation (TPFA). The discrete problem has the form:

$$D\big(M(\mathbf{p})\big)^{-1}D^T\mathbf{p} = \mathbf{f}. \tag{4.4}$$

Since our coarsening scheme is in the mixed setting, we reformulate (4.4) as:

$$A(\mathbf{x}) := \begin{bmatrix} M(\mathbf{p}) & D^T \\ D & \end{bmatrix} \begin{bmatrix} \mathbf{v} \\ \mathbf{p} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ -\mathbf{f} \end{bmatrix} =: \mathbf{b} \tag{4.5}$$

where $\mathbf{x} := (\mathbf{v}, \mathbf{p})$.

To solve (4.5), we use full approximation scheme (FAS) [17], which applies the idea of multigrid directly to the nonlinear problem. In order for FAS to have rapid convergence, it is important to have coarse spaces with good approximation properties. To this end, we utilize our multilevel coarsening algorithm presented in Section 3 to construct the multigrid hierarchy. In the setup phase, the hierarchy is constructed by considering the reference permeability $\kappa(0) = \kappa_0$. In the solve phase, one cycle of FAS is summarized in Algorithm 1. We compare numerically the performance of FAS to Picard iteration for solving (4.4). For both methods, nonlinear iteration stops when (relative/absolute) ℓ_2 norm of the residual is less than 1e-8/1e-10.

Multigrid and coarsening setting:

Coarsening factor: 28
Nonlinear relaxation: 20 linearized iterations

Number of vertex basis per aggregate: $m_{\mathcal{A}}$
Number of edge basis per face: 1

From Table 4.5 and Figure 4.12, it is evident that using more local spectral basis functions in the hierarchy does reduce the number of nonlinear iteration (and solving time) of FAS, which clearly shows the benefit of using higher order coarse approximation. Both Picard iteration and FAS exhibit optimal scaling

Algorithm 1 Full approximation scheme (V-cycle)

1: **function** NonlinearMG($\mathbf{x}^\ell, \mathbf{b}^\ell$)
2:    **if** $\ell ==$ coarsest level **then**
3:        Solve $A^\ell(\mathbf{x}^\ell) = \mathbf{b}^\ell$ for $\mathbf{x}^\ell$
4:    **else**
5:        $\mathbf{x}^\ell \leftarrow$ NonlinearRelaxation($\mathbf{x}^\ell, \mathbf{b}^\ell$)
6:        $\mathbf{x}^{\ell+1} \leftarrow Q^\ell \mathbf{x}^\ell$
7:        $\mathbf{b}^{\ell+1} \leftarrow (P^\ell)^T(\mathbf{b}^\ell - A^\ell(\mathbf{x}^\ell)) + A^{\ell+1}(\mathbf{x}^{\ell+1})$
8:        $\mathbf{y}^{\ell+1} \leftarrow$ NonlinearMG($\mathbf{x}^{\ell+1}, \mathbf{b}^{\ell+1}$)
9:        $\mathbf{x}^\ell \leftarrow \mathbf{x}^\ell + P^\ell(\mathbf{y}^{\ell+1} - \mathbf{x}^{\ell+1})$
10:       $\mathbf{x}^\ell \leftarrow$ NonlinearRelaxation($\mathbf{x}^\ell, \mathbf{b}^\ell$)
11:    **end if**
12:    **return** $\mathbf{x}^\ell$
13: **end function**

TABLE 4.5
Comparison of solve time (#iter) of FAS and Picard method for solving (4.3)

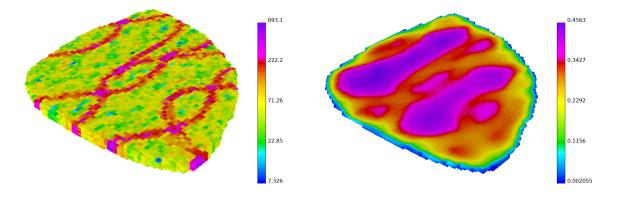|	FAS info		FAS (multilevel Picard)				one-level
#dofs	#levels	OC	$m_\mathcal{A}=1$	$m_\mathcal{A}=5$	$m_\mathcal{A}=10$	$m_\mathcal{A}=15$	Picard
84,580	3	1.22	1.24s (7)	0.95s (5)	0.71s (4)	0.52s (3)	0.96s (11)
636,248	4	1.26	14.79s (9)	5.07s (4)	4.24s (3)	4.52s (3)	9.58s (13)
4,919,776	4	1.3	114.8s (9)	53.4s (4)	41.0s (3)	57.5s (4)	97.3s (13)

FIG. 4.10. *Reference permeability k_0 (left) and pressure solution (right)*

($\mathcal{O}(n)$), while FAS is about 2 times faster than Picard iteration if more local spectral basis functions are used ($m_\mathcal{A} \geq 9$). Notice that we solve the coarse problems by hybridization [5, 26, 12], which reduces the original saddle point problem to a symmetric positive definite problem on edges, so the sparsity patterns of the global systems in coarse levels are determined by the edge-based dofs. In our coarsening setting, we fix the number of edge-based dofs to be 1 per face, so when we increase $m_\mathcal{A}$, the numbers of non-zeros of the global systems and therefore operator complexity (OC) are not affected. Consequently, in the solving stage, the extra cost of using a hierarchy with higher order spectral coarsening is only in the transformation of right hand sides and solutions between the original saddle point problem and the hybridized system, which is local and highly scalable. A more comprehensive study on building nonlinear multigrid solvers using local spectral coarsening is carried out in [25].

5. Conclusions. This paper has extended the two-level graph coarsening algorithm of [7] to a recursive multilevel framework, using the key step of coarsening degrees of freedom rather than graph edges and
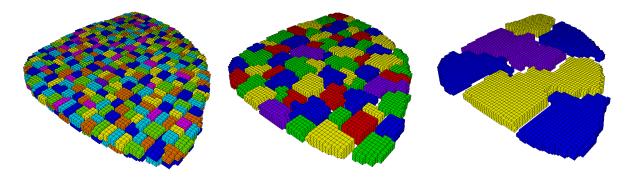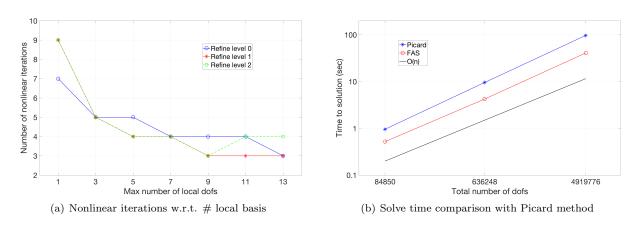
Fig. 4.11. *Hierarchical partitioning*

(a) Nonlinear iterations w.r.t. # local basis

(b) Solve time comparison with Picard method

Fig. 4.12. *Performance of FAS*

vertices. We demonstrated that the coarsening procedure yields coarse representatives of a problem that are much cheaper to solve than the fine problem, but still retains some useful approximation properties that can be used to speed up nonlinear solves and forward uncertainty quantification problems in a finite volume context. Possible future work could include developing faster solvers for the resulting saddle point problems or ways to speed up the (fairly expensive) setup phase by approximating local eigenvalue problems.

REFERENCES

[1] GLVis: OpenGL finite element visualization tool. `glvis.org`.
[2] MFEM: Modular finite element methods library. `mfem.org`.
[3] A. Abdulle, W. E, B. Engquist, and E. Vanden-Eijnden. The heterogeneous multiscale method. *Acta Numerica*, 21:1?87, 2012.
[4] T. Arbogast, G. Pencheva, M. Wheeler, and I. Yotov. A multiscale mortar mixed finite element method. *Multiscale Model. Simul.*, 6(1):319–346 (electronic), 2007.
[5] D. N. Arnold and F. Brezzi. Mixed and nonconforming finite element methods : implementation, postprocessing and error estimates. *ESAIM: M2AN*, 19(1):7–32, 1985.
[6] A. T. Barker, S. A. Gelever, C. S. Lee, C. V. Ponce, and P. S. Vassilevski. smoothG. `github.com/LLNL/smoothG`, 2017.
[7] A. T. Barker, C. S. Lee, and P. S. Vassilevski. Spectral upscaling for graph Laplacian problems with application to reservoir simulation. *SIAM J. Sci. Comput.*, 39:S323–S346, 2017.
[8] T. Chartier, R. Falgout, V. Henson, J. Jones, T. Manteuffel, S. McCormick, J. Ruge, and P. Vassilevski. Spectral AMGe (ρAMGe). *SIAM J. Scientific Computing*, 25:1–26, 2003.
[9] G. A. Chechkin, A. L. Piatnitski, and A. S. Shamaev. *Homogenization: Methods and applications*, volume 234 of *Translations of Mathematical Monographs*. American Mathematical Society, 2007.
[10] M. A. Christie and M. J. Blunt. Tenth SPE comparative solution project: A comparison of upscaling techniques. *Society of Petroleum Engineers*, 2001.
[11] K. A. Cliffe, M. B. Giles, R. Scheichl, and A. L. Teckentrup. Multilevel Monte Carlo methods and applications to elliptic PDEs with random coefficients. *Computing and Visualization in Science*, 14(1):3, 2011.

16

[12] V. Dobrev, T. Kolev, C. S. Lee, V. Tomov, and P. S. Vassilevski. Algebraic hybridization and static condensation with application to scalable $H(\text{div})$ preconditioning. *SIAM Journal on Scientific Computing*, 41(3):B425–B447, 2019.

[13] W. E and B. Engquist. The heterognous multiscale methods. *Commun. Math. Sci.*, 1(1):87–132, 03 2003.

[14] Y. Efendiev, J. Galvis, and T. Hou. Generalized multiscale finite element methods. *Journal of Computational Physics*, 251:116–135, 2013.

[15] Y. Efendiev and T. Y. Hou. *Multiscale Finite Element Methods. Theory and Applications*. Springer Science+Business Media, LLC, 1st edition, 2009.

[16] A. A. Hagberg, D. A. Schult, and P. J. Swart. Exploring network structure, dynamics, and function using NetworkX. In G. Varoquaux, T. Vaught, and J. Millman, editors, *Proceedings of the 7th Python in Science Conference*, pages 11 – 15, Pasadena, CA USA, 2008.

[17] V. E. Henson. Multigrid methods for nonlinear problems: an overview. In *Proc. SPIE*, volume 5016, pages 36–48, 2003.

[18] U. Hornung. *Homogenization and porous media*, volume 6 of *Interdisciplinary Applied Mathematics*. Springer, New York, 1997.

[19] T. Hou and X. Wu. A multiscale finite element method for elliptic problems in composite materials and porous media. *J. Comput. Phys.*, 134:169–189, 1997.

[20] J. D. Jansen, R. M. Fonseca, S. Kahrobaei, M. M. Siraj, G. M. Van Essen, and P. M. J. Van den Hof. The egg model - a geological ensemble for reservoir simulation. *Geoscience Data Journal*, 1(2):192–195, 2014.

[21] P. Jenny, S. Lee, and H. Tchelepi. Multi-scale finite volume method for elliptic problems in subsurface flow simulation. *J. Comput. Phys.*, 187:47–67, 2003.

[22] P. Jenny, S. Lee, and H. Tchelepi. Adaptive fully implicit multi-scale finite-volume method for multi-phase flow and transport in heterogeneous porous media. *Journal of Computational Physics*, 217(2):627 – 641, 2006.

[23] D. Z. Kalchev, C. S. Lee, U. Villa, Y. Efendiev, and P. S. Vassilevski. Upscaling of mixed finited element discretization problems by the spectral AMGe method. *SIAM J. Sci. Comput.*, 38(5):A2912–2933, 2016.

[24] I. V. Lashuk and P. S. Vassilevski. On some versions of the element agglomeration AMGe method. *Numer. Linear Algebra Appl.*, 15:595–620, 2008.

[25] C. S. Lee, N. Castelletto, F. Hamon, P. S. Vassilevski, and J. White. Nonlinear multigrid based on local spectral coarsening. *In preparation.*

[26] C. S. Lee and P. S. Vassilevski. Parallel solver for H(div) problems using hybridization and AMG. *LLNL Report*, LLNL-TR-681025, 2016.

[27] F. Lindgren, H. Rue, and J. Lindström. An explicit link between Gaussian fields and Gaussian Markov random fields: the stochastic differential equation approach. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 73:423–498, 2011.

[28] S. MacLachlan and J. Moulton. Multilevel upscaling through variational coarsening. *Water Resour. Res.*, 42:W02418, doi:10.1029/2005WR003940, 2006.

[29] S. Osborn, P. S. Vassilevski, and U. Villa. A multilevel, hierarchical sampling technique for spatially correlated random fields. *SIAM J. Sci. Comput.*, 39:S543–S562, 2017.

[30] J. E. Pasciak and P. S. Vassilevski. Exact de Rham sequences of spaces defined on macro-elements in two and three spatial dimensions. *SIAM J. Sci. Comput.*, 30(5):2427–2446, 2008.

[31] J. W. Ruge and K. Stüben. Algebraic multigrid (AMG). In S. F. McCormick, editor, *Multigrid Methods*, volume 3 of *Frontiers in Applied Mathematics*, pages 73–130. SIAM, Philadelphia, PA, 1987.

[32] D. Simpson, F. Lindgren, and H. Rue. Think continuous: Markovian Gaussian models in spatial statistics. *Spatial Statistics*, 1:16–29, 2012.

[33] H. A. Tchelepi, P. Jenny, S. H. Lee, and C. Wolfsteiner. Adaptive multiscale finite-volume framework for reservoir simulation. *SPE Journal*, 12(2):188–195, 2007.

[34] P. S. Vassilevski. Coarse spaces by algebraic multigrid: Multigrid convergence and upscaling error estimates. *Advances in Adaptive Data Analysis*, 3(1-2):229–249, 2011.

[35] X.-H. Wen, L. J. Durlofsky, S. H. Lee, and M. G. Edwards. Full tensor upscaling of geologically complex reservoir descriptions. In *SPE Annual Technical Conference and Exhibition*, page 11. Society of Petroleum Engineers, Dallas, Texas, 2000.

[36] X.-H. Wen and J. Gómez-Hernández. Upscaling hydraulic conductivities in heterogeneous media: An overview. *Journal of Hydrology*, 183(1):ix – xxxii, 1996.

A. A locally constructed projection for the edge space. We will construct $Q_\sigma^\ell : \boldsymbol{\Sigma}^\ell \to \boldsymbol{\Sigma}^{\ell+1}$ that is comprised of local operators on each face $\mathcal{F}^\ell$ and each aggregate $\mathcal{A}^\ell$. Some of the notations here have been defined in Section 3.4.

We first restrict ourselves to a face $\mathcal{F}^\ell = (\mathcal{A}_i^\ell, \mathcal{A}_j^\ell)$ on level ℓ. Define a projection onto the PV trace:

$$\pi_{\sigma,\mathcal{F}^\ell}^{\ell,PV} := \boldsymbol{\sigma}_{\mathcal{F}^\ell}^{PV} Q_{\sigma,\mathcal{F}^\ell}^{\ell,PV}, \qquad \text{where} \quad Q_{\sigma,\mathcal{F}^\ell}^{\ell,PV} := \left((\mathbf{q}_{\mathcal{A}_i^\ell}^{PV})^T D_{\mathcal{A}_i^\ell,\mathcal{F}^\ell}^\ell \boldsymbol{\sigma}_{\mathcal{F}^\ell}^{PV} \right)^{-1} (\mathbf{q}_{\mathcal{A}_i^\ell}^{PV})^T D_{\mathcal{A}_i^\ell,\mathcal{F}^\ell}^\ell. \tag{A.1}$$

Note that condition (3.7) guarantees that $Q_{\sigma,\mathcal{F}^\ell}^{\ell,PV}$ is well-defined.

LEMMA A.1. *If $(D_{\mathcal{B}^\ell}^\ell)^T \mathbf{1}_{\mathcal{B}^\ell}^\ell = \mathbf{0}$ for any aggregate $\mathcal{B}^\ell$ of vertices on level ℓ, then*

$$\left((\mathbf{q}_{\mathcal{A}_i^\ell}^{PV})^T D_{\mathcal{A}_i^\ell,\mathcal{F}^\ell}^\ell \boldsymbol{\sigma}_{\mathcal{F}^\ell}^{PV} \right)^{-1} (\mathbf{q}_{\mathcal{A}_i^\ell}^{PV})^T D_{\mathcal{A}_i^\ell,\mathcal{F}^\ell}^\ell = \left((\mathbf{q}_{\mathcal{A}_j^\ell}^{PV})^T D_{\mathcal{A}_j^\ell,\mathcal{F}^\ell}^\ell \boldsymbol{\sigma}_{\mathcal{F}^\ell}^{PV} \right)^{-1} (\mathbf{q}_{\mathcal{A}_j^\ell}^{PV})^T D_{\mathcal{A}_j^\ell,\mathcal{F}^\ell}^\ell.$$

Proof. Let $\mathcal{A}_{\mathcal{F}^\ell} := \mathcal{A}_i^\ell \cup \mathcal{A}_j^\ell$. Then

$$(\mathbf{0}_{\mathcal{A}_{\mathcal{F}^\ell}})^T = (\mathbf{1}_{\mathcal{A}_{\mathcal{F}^\ell}}^\ell)^T D_{\mathcal{A}_{\mathcal{F}^\ell}}^\ell = \begin{bmatrix} (\mathbf{1}_{\mathcal{A}_i^\ell}^\ell)^T D_{\mathcal{A}_i^\ell}^\ell & (\mathbf{1}_{\mathcal{A}_i^\ell}^\ell)^T D_{\mathcal{A}_i^\ell,\mathcal{F}^\ell}^\ell + (\mathbf{1}_{\mathcal{A}_j^\ell}^\ell)^T D_{\mathcal{A}_j^\ell,\mathcal{F}^\ell}^\ell & (\mathbf{1}_{\mathcal{A}_j^\ell}^\ell)^T D_{\mathcal{A}_j^\ell}^\ell \end{bmatrix}$$

$$= \begin{bmatrix} (\mathbf{0}_{\mathcal{A}_i^\ell})^T & (\mathbf{1}_{\mathcal{A}_i^\ell}^\ell)^T D_{\mathcal{A}_i^\ell,\mathcal{F}^\ell}^\ell + (\mathbf{1}_{\mathcal{A}_j^\ell}^\ell)^T D_{\mathcal{A}_j^\ell,\mathcal{F}^\ell}^\ell & (\mathbf{0}_{\mathcal{A}_j^\ell})^T \end{bmatrix}.$$

Thus, we have

$$(\mathbf{1}_{\mathcal{A}_i^\ell}^\ell)^T D_{\mathcal{A}_i^\ell,\mathcal{F}^\ell}^\ell = -(\mathbf{1}_{\mathcal{A}_j^\ell}^\ell)^T D_{\mathcal{A}_j^\ell,\mathcal{F}^\ell}^\ell. \tag{A.2}$$

Recall from (3.6) that $\mathbf{q}_{\mathcal{A}_i^\ell}^{PV} = \|\mathbf{1}_{\mathcal{A}_i^\ell}^\ell\|^{-1} \mathbf{1}_{\mathcal{A}_i^\ell}^\ell$, the lemma now follows easily from this and (A.2). $\qquad\square$

REMARK A.1. *We will show later in Lemma A.5 that the hypothesis in Lemma A.1 is actually valid on all levels. Therefore, although we formally use one of the aggregates $\mathcal{A}_i^\ell$ to define $Q_{\sigma,\mathcal{F}^\ell}^{\ell,PV}$ in (A.1), Lemma A.1 tells us that the definition of $Q_{\sigma,\mathcal{F}^\ell}^{\ell,PV}$ is in fact independent of the choice of $\mathcal{A}_i^\ell$ or $\mathcal{A}_j^\ell$.*

Next, we will construct a projection for "non-PV" traces. Without loss of generality, it is assumed that,

$$(\mathbf{q}_{\mathcal{A}_i^\ell}^{PV})^T D_{\mathcal{A}_i^\ell,\mathcal{F}^\ell}^\ell \boldsymbol{\sigma}_{\mathcal{F}^\ell} = 0, \qquad \forall \boldsymbol{\sigma}_{\mathcal{F}^\ell} \neq \boldsymbol{\sigma}_{\mathcal{F}^\ell}^{PV}. \tag{A.3}$$

Indeed, (A.3) can always be achieved by replacing $\boldsymbol{\sigma}_{\mathcal{F}^\ell}$ with $\boldsymbol{\sigma}_{\mathcal{F}^\ell} - \left(Q_{\sigma,\mathcal{F}^\ell}^{\ell,PV} \boldsymbol{\sigma}_{\mathcal{F}^\ell} \right) \boldsymbol{\sigma}_{\mathcal{F}^\ell}^{PV}$ for $\boldsymbol{\sigma}_{\mathcal{F}^\ell} \neq \boldsymbol{\sigma}_{\mathcal{F}^\ell}^{PV}$. Recall that $P_{\sigma,\mathcal{F}^\ell}^{\ell,NPV}$ defined in (3.11) is the matrix whose columns are the non-PV traces. Now define a projection $\pi_{\sigma,\mathcal{F}^\ell}^{\ell,NPV} := P_{\sigma,\mathcal{F}^\ell}^{\ell,NPV} Q_{\sigma,\mathcal{F}^\ell}^{\ell,NPV}$ onto the span of non-PV traces, where

$$Q_{\sigma,\mathcal{F}^\ell}^{\ell,NPV} := \left((P_{\sigma,\mathcal{F}^\ell}^{\ell,NPV})^T P_{\sigma,\mathcal{F}^\ell}^{\ell,NPV} \right)^{-1} (P_{\sigma,\mathcal{F}^\ell}^{\ell,NPV})^T (I - \boldsymbol{\sigma}_{\mathcal{F}^\ell}^{PV} Q_{\sigma,\mathcal{F}^\ell}^{PV}). \tag{A.4}$$

Combining $\pi_{\sigma,\mathcal{F}^\ell}^{\ell,PV}$ and $\pi_{\sigma,\mathcal{F}^\ell}^{\ell,NPV}$, we obtain a projection for all trace extensions associated with $\mathcal{F}^\ell$:

$$\pi_{\sigma,\mathcal{F}^\ell}^{\ell,T} := \pi_{\sigma,\mathcal{F}^\ell}^{\ell,PV} + \pi_{\sigma,\mathcal{F}^\ell}^{\ell,NPV} = P_{\sigma,\mathcal{F}^\ell}^{\ell,T} Q_{\sigma,\mathcal{F}^\ell}^{\ell,T},$$

where $P_{\sigma,\mathcal{F}^\ell}^{\ell,T}$ is defined in (3.11) and $Q_{\sigma,\mathcal{F}^\ell}^{\ell,T} := \begin{bmatrix} Q_{\sigma,\mathcal{F}^\ell}^{\ell,PV} \\ Q_{\sigma,\mathcal{F}^\ell}^{\ell,NPV} \end{bmatrix}.$

LEMMA A.2. *$Q_{\sigma,\mathcal{F}^\ell}^{\ell,T}$ is a left inverse of $P_{\sigma,\mathcal{F}^\ell}^{\ell,T}$. That is, $Q_{\sigma,\mathcal{F}^\ell}^{\ell,T} P_{\sigma,\mathcal{F}^\ell}^{\ell,T} = I_{\sigma,\mathcal{F}^\ell}^{\ell+1}$, where $I_{\sigma,\mathcal{F}^\ell}^{\ell+1}$ is the identity matrix of size number of traces in $Tr(\mathcal{F}^\ell)$.*

Proof. Recall from (3.11) that columns of $P_{\sigma,\mathcal{F}^\ell}^{\ell,T}$ can be split into PV and non-PV traces

$$P_{\sigma,\mathcal{F}^\ell}^{\ell,T} = \begin{bmatrix} \boldsymbol{\sigma}_{\mathcal{F}^\ell}^{PV} & P_{\sigma,\mathcal{F}^\ell}^{\ell,NPV} \end{bmatrix}.$$

It follows clearly from definition (A.1) and condition (A.3) that

$$Q_{\sigma,\mathcal{F}^\ell}^{\ell,PV} P_{\sigma,\mathcal{F}^\ell}^{\ell,T} = \begin{bmatrix} Q_{\sigma,\mathcal{F}^\ell}^{\ell,PV} \boldsymbol{\sigma}_{\mathcal{F}^\ell}^{PV} & Q_{\sigma,\mathcal{F}^\ell}^{\ell,PV} P_{\sigma,\mathcal{F}^\ell}^{\ell,NPV} \end{bmatrix} = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix}, \tag{A.5}$$

and therefore

$$(I - \boldsymbol{\sigma}_{\mathcal{F}^\ell}^{PV} Q_{\sigma,\mathcal{F}^\ell}^{PV}) P_{\sigma,\mathcal{F}^\ell}^{\ell,T} = P_{\sigma,\mathcal{F}^\ell}^{\ell,T} - \boldsymbol{\sigma}_{\mathcal{F}^\ell}^{PV} \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix} = \begin{bmatrix} \mathbf{0}_{\sigma,\mathcal{F}^\ell}^\ell & P_{\sigma,\mathcal{F}^\ell}^{\ell,NPV} \end{bmatrix}. \tag{A.6}$$

By (A.4) and (A.6),

$$
Q_{\sigma,\mathcal{F}^\ell}^{\ell,NPV}\, P_{\sigma,\mathcal{F}^\ell}^{\ell,T} = \left[\begin{array}{c|c} \mathbf{0}_{\sigma,\mathcal{F}^\ell}^{\ell+1,NPV} & \left((P_{\sigma,\mathcal{F}^\ell}^{\ell,NPV})^T P_{\sigma,\mathcal{F}^\ell}^{\ell,NPV}\right)^{-1}(P_{\sigma,\mathcal{F}^\ell}^{\ell,NPV})^T P_{\sigma,\mathcal{F}^\ell}^{\ell,NPV} \end{array}\right]
$$
$$
= \left[\begin{array}{c|c} \mathbf{0}_{\sigma,\mathcal{F}^\ell}^{\ell+1,NPV} & I_{\sigma,\mathcal{F}^\ell}^{\ell+1,NPV} \end{array}\right],
$$

(A.7)

where $I_{\sigma,\mathcal{F}^\ell}^{\ell+1,NPV}$ is the identity matrix. Combining (A.5) and (A.7),

$$
Q_{\sigma,\mathcal{F}^\ell}^{\ell,T}\, P_{\sigma,\mathcal{F}^\ell}^{\ell,T} = \left[\begin{array}{c} Q_{\sigma,\mathcal{F}^\ell}^{\ell,PV} P_{\sigma,\mathcal{F}^\ell}^{\ell,T} \\ \hline Q_{\sigma,\mathcal{F}^\ell}^{\ell,NPV} P_{\sigma,\mathcal{F}^\ell}^{\ell,T} \end{array}\right] = \left[\begin{array}{c} 1 \\ \hline I_{\sigma,\mathcal{F}^\ell}^{\ell+1,NPV} \end{array}\right].
$$

$\square$

Next, for each aggregate $\mathcal{A}^\ell$, define a projection $\pi_{\sigma,\mathcal{A}^\ell}^{\ell,B} := P_{\sigma,\mathcal{A}^\ell}^{\ell,B}\, Q_{\sigma,\mathcal{A}^\ell}^{\ell,B}$ to the bubble space in $\mathcal{A}^\ell$, where

$$
Q_{\sigma,\mathcal{A}^\ell}^{\ell,B} := (P_{u,\mathcal{A}^\ell}^{\ell,NPV})^T D_{\mathcal{A}^\ell}^\ell. \tag{A.8}
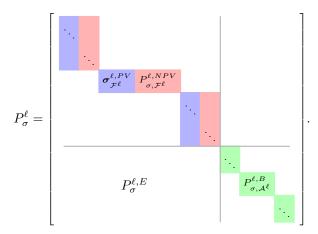$$

We recall that $P_{u,\mathcal{A}^\ell}^{\ell,NPV}$ defined in (3.5) contains the coarse vertex space basis vectors in $\mathcal{A}^\ell$ that are orthogonal to $\mathbf{1}_{\ell^\ell}^\ell$. By (3.10) and the fact that columns of $P_{u,\mathcal{A}^\ell}^{\ell,NPV}$ are orthonormal, we have

$$
Q_{\sigma,\mathcal{A}^\ell}^{\ell,B}\, P_{\sigma,\mathcal{A}^\ell}^{\ell,B} = (P_{u,\mathcal{A}^\ell}^{\ell,NPV})^T\,(D_{\mathcal{A}^\ell}^\ell P_{\sigma,\mathcal{A}^\ell}^{\ell,B}) = (P_{u,\mathcal{A}^\ell}^{\ell,NPV})^T\, P_{u,\mathcal{A}^\ell}^{\ell,NPV} = I_{\sigma,\mathcal{A}^\ell}^{\ell+1,B}, \tag{A.9}
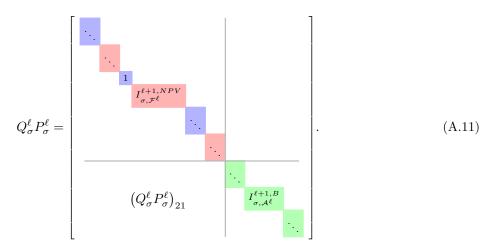$$

where $I_{\sigma,\mathcal{A}^\ell}^{\ell+1,B}$ is the identity. Finally, with $D_{E^{\ell+1}}^\ell$ being the column restriction of D^ℓ to $\bigoplus_{\mathcal{F}^\ell \in E^{\ell+1}} \mathbf{\Sigma}^\ell(\mathcal{F}^\ell)$, we define the projection to the coarse edge space $\pi_\sigma^\ell := P_\sigma^\ell\, Q_\sigma^\ell$, where

(A.10)

Now we are ready to prove Proposition 3.1.

Proof of Proposition 3.1: Recall from (3.12) that P_σ^ℓ has the structure

$$P_\sigma^\ell = \begin{bmatrix} \ddots & & & & & & \\ & \boldsymbol{\sigma}_{\mathcal{F}^\ell}^{\ell,PV} & P_{\sigma,\mathcal{F}^\ell}^{\ell,NPV} & & & & \\ & & & \ddots & & & \\ & & & & \ddots & & \\ & & & & & \ddots & \\ & & P_\sigma^{\ell,E} & & & P_{\sigma,\mathcal{A}^\ell}^{\ell,B} & \\ & & & & & & \ddots \end{bmatrix}.$$

By (A.10), Lemma A.2 and (A.9), we have

$$Q_\sigma^\ell P_\sigma^\ell = \begin{bmatrix} \ddots & & & & & & \\ & \ddots & & & & & \\ & & 1 & & & & \\ & & & I_{\sigma,\mathcal{F}^\ell}^{\ell+1,NPV} & & & \\ & & & & \ddots & & \\ & & & & \ddots & & \\ & & (Q_\sigma^\ell P_\sigma^\ell)_{21} & & & I_{\sigma,\mathcal{A}^\ell}^{\ell+1,B} & \\ & & & & & & \ddots \end{bmatrix}. \tag{A.11}$$

Hence, it suffices to show that $\left(Q_\sigma^\ell P_\sigma^\ell\right)_{21}$, the (2,1)-block of the product $Q_\sigma^\ell P_\sigma^\ell$, is a zero matrix:

$$\left(Q_\sigma^\ell P_\sigma^\ell\right)_{21} = \begin{bmatrix} \ddots & & \\ & (P_{u,\mathcal{A}^\ell}^{\ell,NPV})^T & \\ & & \ddots \end{bmatrix} D_{E^{\ell+1}}^\ell \begin{bmatrix} \ddots & & \\ & P_{\sigma,\mathcal{F}^\ell}^{\ell,T} & \\ & & \ddots \end{bmatrix} + \begin{bmatrix} \ddots & & \\ & Q_{\sigma,\mathcal{A}^\ell}^\ell & \\ & & \ddots \end{bmatrix} P_\sigma^{\ell,E} = 0. \tag{A.12}$$

We will focus on the column of $\left(Q_\sigma^\ell P_\sigma^\ell\right)_{21}$ that is associated with some trace $\boldsymbol{\sigma}_{\mathcal{F}^\ell}$, where $\mathcal{F}^\ell = (\mathcal{A}_i^\ell, \mathcal{A}_j^\ell)$. The only possible nonzeros in this columns are the rows associated with the bubbles in $\mathcal{A}_i^\ell$ and $\mathcal{A}_j^\ell$. For $\mathcal{A}^\ell = \mathcal{A}_i^\ell$ or $\mathcal{A}_j^\ell$, let $\boldsymbol{\sigma}_{\mathcal{A}^\ell}^E$ be the extension of $\boldsymbol{\sigma}_{\mathcal{F}^\ell}$ into $\boldsymbol{\Sigma}^\ell(\mathcal{A}^\ell)$ that is defined by (3.8). Then the submatrix of $\left(Q_\sigma^\ell P_\sigma^\ell\right)_{21}$ restricted to the column associated with $\boldsymbol{\sigma}_{\mathcal{F}^\ell}$ and the rows associated with bubbles in $\mathcal{A}^\ell$ is

$$(P_{u,\mathcal{A}^\ell}^{\ell,NPV})^T D_{\mathcal{A}^\ell,\mathcal{F}^\ell}^\ell \boldsymbol{\sigma}_{\mathcal{F}^\ell} + Q_{\sigma,\mathcal{A}^\ell}^\ell \boldsymbol{\sigma}_{\mathcal{A}^\ell}^E.$$

By the second block of equations in (3.8), we know that

$$(P_{u,\mathcal{A}^\ell}^{\ell,NPV})^T D_{\mathcal{A}^\ell,\mathcal{F}^\ell}^\ell \boldsymbol{\sigma}_{\mathcal{F}^\ell} + Q_{\sigma,\mathcal{A}^\ell}^\ell \boldsymbol{\sigma}_{\mathcal{A}^\ell}^E = (P_{u,\mathcal{A}^\ell}^{\ell,NPV})^T \left(D_{\mathcal{A}^\ell,\mathcal{F}^\ell}^\ell \boldsymbol{\sigma}_{\mathcal{F}^\ell} + D_{\mathcal{A}^\ell}^\ell \boldsymbol{\sigma}_{\mathcal{A}^\ell}^E\right)$$
$$= (P_{u,\mathcal{A}^\ell}^{\ell,NPV})^T \left(c_{\mathcal{A}}^\ell \mathbf{q}_{\mathcal{A}^\ell}^{PV}\right) = 0.$$

Thus, (A.12) and therefore the proposition follow. $\qquad\square$

The rest of this section are some lemmas that lead to the key commutative property (3.13).

LEMMA A.3. *On level ℓ, if $(D_{\mathcal{B}^\ell}^\ell)^T \mathbf{1}_{\mathcal{B}^\ell}^\ell = \mathbf{0}$ for any aggregate $\mathcal{B}^\ell$, then*

$$D^{\ell+1} Q_\sigma^\ell = Q_u^\ell D^\ell. \tag{A.13}$$

Proof. Since $D^{\ell+1} = (P_u^\ell)^T D^\ell P_\sigma^\ell = Q_u^\ell D^\ell P_\sigma^\ell$, proving (A.13) is equivalent to prove that,

$$Q_u^\ell D^\ell \left(P_\sigma^\ell Q_\sigma^\ell \boldsymbol{\sigma} \right) = Q_u^\ell D^\ell \boldsymbol{\sigma}, \qquad \forall \boldsymbol{\sigma} \in \boldsymbol{\Sigma}^\ell. \tag{A.14}$$

Our goal is to show (A.14) by considering rows corresponding to one aggregate at a time. In the aggregate $\mathcal{A}^\ell$, Q_u^ℓ is simply (cf. (3.5))

$$(P_{u,\mathcal{A}^\ell}^\ell)^T = \begin{bmatrix} (\mathbf{q}_{\mathcal{A}^\ell}^{PV})^T \\ (P_{u,\mathcal{A}^\ell}^{NPV})^T \end{bmatrix}.$$

Consider partitions of P_σ^ℓ and Q_σ^ℓ based on trace extensions and bubbles (cf. (3.12) and (A.10))

$$P_\sigma^\ell = \begin{bmatrix} P_\sigma^{\ell,TE} & P_\sigma^{\ell,B} \end{bmatrix}, \qquad Q_\sigma^\ell = \begin{bmatrix} Q_\sigma^{\ell,TE} \\ Q_\sigma^{\ell,B} \end{bmatrix}.$$

By construction, the restriction of $D^\ell \boldsymbol{\sigma}_i$ in $\mathcal{A}^\ell$ is in $\mathrm{Span}\left\{ \mathbf{q}_{\mathcal{A}^\ell}^{PV} \right\}$ if $\boldsymbol{\sigma}_i$ is a trace extension; or it is in $\mathrm{Range}\left(P_{u,\mathcal{A}}^{NPV} \right)$ if $\boldsymbol{\sigma}_i$ is a bubble. This observation and the orthogonality of the columns of $P_{u,\mathcal{A}^\ell}^\ell$ imply that

$$\begin{aligned} (P_{u,\mathcal{A}^\ell}^\ell)^T \left((D^\ell P_\sigma^\ell Q_\sigma^\ell \boldsymbol{\sigma})|_{\mathcal{A}^\ell} \right) &= \begin{bmatrix} (\mathbf{q}_{\mathcal{A}^\ell}^{PV})^T \\ (P_{u,\mathcal{A}^\ell}^{NPV})^T \end{bmatrix} \left(D^\ell P_\sigma^{\ell,TE} Q_\sigma^{\ell,TE} \boldsymbol{\sigma} + D^\ell P_\sigma^{\ell,B} Q_\sigma^{\ell,B} \boldsymbol{\sigma} \right)|_{\mathcal{A}^\ell} \\ &= \begin{bmatrix} (\mathbf{q}_{\mathcal{A}^\ell}^{PV})^T \left(D^\ell P_\sigma^{\ell,TE} Q_\sigma^{\ell,TE} \boldsymbol{\sigma} \right)|_{\mathcal{A}^\ell} \\ (P_{u,\mathcal{A}^\ell}^{NPV})^T \left(D^\ell P_\sigma^{\ell,B} Q_\sigma^{\ell,B} \boldsymbol{\sigma} \right)|_{\mathcal{A}^\ell} \end{bmatrix}. \end{aligned} \tag{A.15}$$

For the bubble part, we know from the structure of Q_σ^ℓ (A.10) that $Q_\sigma^{\ell,B}$ has the form

$$Q_\sigma^{\ell,B} = \begin{bmatrix} \begin{bmatrix} \ddots & & \\ & (P_{u,\mathcal{A}^\ell}^{\ell,NPV})^T & \\ & & \ddots \end{bmatrix} D_{E^{\ell+1}}^\ell \begin{bmatrix} \ddots & & \\ & Q_{\sigma,\mathcal{A}^\ell}^{\ell,B} & \\ & & \ddots \end{bmatrix} \end{bmatrix} \tag{A.16}$$

By (A.8),

$$\begin{bmatrix} \ddots & & \\ & Q_{\sigma,\mathcal{A}^\ell}^{\ell,B} & \\ & & \ddots \end{bmatrix} = \begin{bmatrix} \ddots & & \\ & (P_{u,\mathcal{A}^\ell}^{\ell,NPV})^T D_{\mathcal{A}^\ell}^\ell & \\ & & \ddots \end{bmatrix} = \begin{bmatrix} \ddots & & \\ & (P_{u,\mathcal{A}^\ell}^{\ell,NPV})^T & \\ & & \ddots \end{bmatrix} \begin{bmatrix} \ddots & & \\ & D_{\mathcal{A}^\ell}^\ell & \\ & & \ddots \end{bmatrix} \tag{A.17}$$

Substituting (A.17) into (A.16), we can see that

$$Q_\sigma^{\ell,B} = \begin{bmatrix} \ddots & & \\ & (P_{u,\mathcal{A}^\ell}^{\ell,NPV})^T & \\ & & \ddots \end{bmatrix} D_{E^{\ell+1}}^\ell \begin{bmatrix} \ddots & & \\ & D_{\mathcal{A}^\ell}^\ell & \\ & & \ddots \end{bmatrix} = \begin{bmatrix} \ddots & & \\ & (P_{u,\mathcal{A}^\ell}^{\ell,NPV})^T & \\ & & \ddots \end{bmatrix} D^\ell. \tag{A.18}$$

Hence, by (3.10) and (A.18),

$$(P_{u,\mathcal{A}^\ell}^{NPV})^T \left(D^\ell P_\sigma^{\ell,B} Q_\sigma^{\ell,B} \boldsymbol{\sigma}\right)|_{\mathcal{A}^\ell} = (P_{u,\mathcal{A}^\ell}^{NPV})^T \left((D^\ell P_\sigma^{\ell,B})(Q_\sigma^{\ell,B}\boldsymbol{\sigma})\right)|_{\mathcal{A}^\ell}$$

$$= (P_{u,\mathcal{A}^\ell}^{NPV})^T \left(\begin{bmatrix} \ddots & & \\ & P_{u,\mathcal{A}^\ell}^{\ell,NPV} & \\ & & \ddots \end{bmatrix}\begin{bmatrix} \ddots & & \\ & (P_{u,\mathcal{A}^\ell}^{\ell,NPV})^T & \\ & & \ddots \end{bmatrix} D^\ell\boldsymbol{\sigma}\right)\Bigg|_{\mathcal{A}^\ell} \tag{A.19}$$

$$= (P_{u,\mathcal{A}^\ell}^{NPV})^T P_{u,\mathcal{A}^\ell}^{NPV} (P_{u,\mathcal{A}^\ell}^{NPV})^T \left(D^\ell\boldsymbol{\sigma}\right)|_{\mathcal{A}^\ell}$$

$$= (P_{u,\mathcal{A}^\ell}^{NPV})^T \left(D^\ell\boldsymbol{\sigma}\right)|_{\mathcal{A}^\ell}.$$

For the trace extension part, first note that by Lemma A.1 and the hypothesis of the current lemma, we can take $Q_{\sigma,\mathcal{F}^\ell}^{\ell,PV}$ in the definition of PV-trace projections for faces $\mathcal{F}^\ell \subseteq \partial E^{\ell+1}(\mathcal{A}^\ell)$ to be (cf. Remark A.1)

$$Q_{\sigma,\mathcal{F}^\ell}^{\ell,PV} = \left((\mathbf{q}_{\mathcal{A}^\ell}^{PV})^T D_{\mathcal{A}^\ell,\mathcal{F}^\ell}^\ell \boldsymbol{\sigma}_{\mathcal{F}^\ell}^{PV}\right)^{-1} (\mathbf{q}_{\mathcal{A}^\ell}^{PV})^T D_{\mathcal{A}^\ell,\mathcal{F}^\ell}^\ell. \tag{A.20}$$

Next, note also that the only relevant coarse edge dofs for the aggregate $\mathcal{A}^\ell$ are the trace extensions $\boldsymbol{\sigma}_{\mathcal{F}^\ell}^{TE}$ associated with boundary faces of $\mathcal{A}^\ell$, $\mathcal{F}^\ell \subseteq \partial E^{\ell+1}(\mathcal{A}^\ell)$. Moreover, by definition (3.8), each $\boldsymbol{\sigma}_{\mathcal{F}^\ell}^{TE}$ satisfies

$$\left(D^\ell \boldsymbol{\sigma}_{\mathcal{F}^\ell}^{TE}\right)|_{\mathcal{A}^\ell} = c_{\mathcal{A}^\ell} \mathbf{q}_{\mathcal{A}^\ell}^{PV}. \tag{A.21}$$

Because of (3.9) and (A.3), the constant $c_{\mathcal{A}^\ell}$ is exactly

$$c_{\mathcal{A}^\ell} = \begin{cases} c_{\mathcal{A}^\ell,\mathcal{F}^\ell}^{PV} := (\mathbf{q}_{\mathcal{A}^\ell}^{PV})^T D_{\mathcal{A}^\ell,\mathcal{F}^\ell}^\ell \boldsymbol{\sigma}_{\mathcal{F}^\ell}^{PV}. & \text{if } \boldsymbol{\sigma}_{\mathcal{F}^\ell}^{TE} \text{ is the extension of } \boldsymbol{\sigma}_{\mathcal{F}^\ell}^{PV}, \\ 0 & \text{otherwise.} \end{cases} \tag{A.22}$$

So, only the PV-trace projections associated with $\mathcal{F}^\ell \subseteq \partial E^{\ell+1}(\mathcal{A}^\ell)$ contribute to $\left(D^\ell P_\sigma^{\ell,TE} Q_\sigma^{\ell,TE}\boldsymbol{\sigma}\right)|_{\mathcal{A}^\ell}$. This observation together with (A.21), (A.22), (3.6), (A.20), and the hypothesis of the lemma imply that

$$(\mathbf{q}_{\mathcal{A}^\ell}^{PV})^T \left(D^\ell P_\sigma^{\ell,TE} Q_\sigma^{\ell,TE}\boldsymbol{\sigma}\right)|_{\mathcal{A}^\ell} = (\mathbf{q}_{\mathcal{A}^\ell}^{PV})^T \left((D^\ell P_\sigma^{\ell,TE})(Q_\sigma^{\ell,TE}\boldsymbol{\sigma})\right)|_{\mathcal{A}^\ell}$$

$$= (\mathbf{q}_{\mathcal{A}^\ell}^{PV})^T \sum_{\mathcal{F}^\ell \subseteq \partial E^{\ell+1}(\mathcal{A}^\ell)} \left(c_{\mathcal{A}^\ell,\mathcal{F}^\ell}^{PV} \mathbf{q}_{\mathcal{A}^\ell}^{PV}\right) \left(Q_{\sigma,\mathcal{F}^\ell}^{\ell,PV}\boldsymbol{\sigma}|_{\mathcal{F}^\ell}\right)$$

$$= \sum_{\mathcal{F}^\ell \subseteq \partial E^{\ell+1}(\mathcal{A}^\ell)} \left(c_{\mathcal{A}^\ell,\mathcal{F}^\ell}^{PV} \|\mathbf{q}_{\mathcal{A}^\ell}^{PV}\|^2\right) \left(\left((\mathbf{q}_{\mathcal{A}^\ell}^{PV})^T D_{\mathcal{A}^\ell,\mathcal{F}^\ell}^\ell \boldsymbol{\sigma}_{\mathcal{F}^\ell}^{PV}\right)^{-1} (\mathbf{q}_{\mathcal{A}^\ell}^{PV})^T D_{\mathcal{A}^\ell,\mathcal{F}^\ell}^\ell \boldsymbol{\sigma}|_{\mathcal{F}^\ell}\right) \tag{A.23}$$

$$= \sum_{\mathcal{F}^\ell \subseteq \partial E^{\ell+1}(\mathcal{A}^\ell)} (\mathbf{q}_{\mathcal{A}^\ell}^{PV})^T D_{\mathcal{A}^\ell,\mathcal{F}^\ell}^\ell \boldsymbol{\sigma}|_{\mathcal{F}^\ell}$$

$$= (\mathbf{q}_{\mathcal{A}^\ell}^{PV})^T D_{\mathcal{A}^\ell}^\ell \boldsymbol{\sigma}|_{\mathcal{A}^\ell} + \sum_{\mathcal{F}^\ell \subseteq \partial E^{\ell+1}(\mathcal{A}^\ell)} (\mathbf{q}_{\mathcal{A}^\ell}^{PV})^T D_{\mathcal{A}^\ell,\mathcal{F}^\ell}^\ell \boldsymbol{\sigma}|_{\mathcal{F}^\ell}$$

$$= (\mathbf{q}_{\mathcal{A}^\ell}^{PV})^T \left(D^\ell\boldsymbol{\sigma}\right)|_{\mathcal{A}^\ell}.$$

Finally, (3.13) follows from (A.14), (A.19), and (A.23). $\qquad\square$

LEMMA A.4. *On level $\ell \geq 0$, if $(D_{\mathcal{B}^\ell}^\ell)^T \mathbf{1}_{\mathcal{B}^\ell}^\ell = \mathbf{0}$ for any aggregate of vertices $\mathcal{B}^\ell$, then*

$$Span\left\{\mathbf{1}_{\mathcal{A}_i^\ell}^\ell\right\} \subseteq Range\left(P_{u,\mathcal{A}_i^\ell}^\ell\right) \qquad \forall \mathcal{A}_i^\ell \in V^{\ell+1}. \tag{A.24}$$

Proof. By the assumption and noticing that $\mathcal{L}_{N(\mathcal{A}_i^\ell)}$ is symmetric positive semi-definite, we know $\mathbf{1}_{N(\mathcal{A}_i^\ell)}^\ell$ is an eigenvector of (3.2) corresponding to the smallest eigenvalue $\lambda = 0$. Recall that our algorithm picks the eigenvectors associated with the small eigenvalues, so $\mathbf{1}_{N(\mathcal{A}_i^\ell)}^\ell$ is always selected. After restricting $\mathbf{1}_{N(\mathcal{A}_i^\ell)}^\ell$ to

22

$\mathcal{A}_i^{\ell}$, we get $\mathbf{1}_{\mathcal{A}_i^{\ell}}^{\ell}$. As SVD only removes linear dependence, $\mathbf{1}_{\mathcal{A}_i^{\ell}}^{\ell}$ remains in the range of $P_{u,\mathcal{A}_i^{\ell}}^{\ell}$. $\qquad\square$

LEMMA A.5. *On level $\ell \geq 0$, for any aggregate of vertices $\mathcal{B}^{\ell}$, we have $(D_{\mathcal{B}^{\ell}}^{\ell})^T \mathbf{1}_{\mathcal{B}^{\ell}}^{\ell} = \mathbf{0}$. Furthermore, if $\mathcal{B}^{\ell}$ is connected, then $\mathrm{Null}\left((D_{\mathcal{B}^{\ell}}^{\ell})^T\right) = \mathrm{Span}\left\{\, \mathbf{1}_{\mathcal{B}^{\ell}}^{\ell} \,\right\}$.*

Proof. We will show the lemma by induction on ℓ. When $\ell = 0$, for any aggregate $\mathcal{B}^0$, $(D_{\mathcal{B}^0}^0)^T$ is an incidence matrix of which each row has exactly 2 non-zero entries; one is 1 and the other is -1. Clearly, the constant vector $\mathbf{1}_{\mathcal{B}^0}^0$ is in $\mathrm{Null}\left((D_{\mathcal{B}^0}^0)^T\right)$. Moreover, if $\mathcal{B}^0$ is connected, $\mathbf{1}_{\mathcal{B}^0}^0$ is the only vector spanning $\mathrm{Null}\left((D_{\mathcal{B}^0}^0)^T\right)$. Thus, the lemma holds when $\ell = 0$.

Now suppose the lemma is true on level ℓ. Our goal is to show that the lemma is also true on level $\ell + 1$. Consider an aggregate $\mathcal{B}^{\ell+1}$ on level $\ell + 1$, which by definition is a set of level-$(\ell+1)$ vertices (or equivalently level-ℓ aggregates): $\mathcal{B}^{\ell+1} = \{\mathcal{A}_i^{\ell}\}$. Let $P_{u,\mathcal{B}^{\ell+1}}^{\ell}$ be the block diagonal matrix

$$P_{u,\mathcal{B}^{\ell+1}}^{\ell} = \begin{bmatrix} \ddots & & \\ & P_{u,\mathcal{A}_i^{\ell}}^{\ell} & \\ & & \ddots \end{bmatrix}_{\mathcal{A}_i^{\ell} \in \mathcal{B}^{\ell}}.$$

By the induction hypothesis and Lemma A.4, $\mathbf{1}_{\mathcal{A}_i^{\ell}}^{\ell} \in \mathrm{Range}\left(P_{u,\mathcal{A}_i^{\ell}}^{\ell}\right)$ for all $\mathcal{A}_i^{\ell} \in \mathcal{B}^{\ell+1}$. Hence, $\mathbf{1}_{\mathcal{B}^{\ell+1}}^{\ell}$ is in $\mathrm{Range}\left(P_{u,\mathcal{B}^{\ell+1}}^{\ell}\right)$ and so $P_{u,\mathcal{B}^{\ell+1}}^{\ell} Q_{u,\mathcal{B}^{\ell+1}}^{\ell} \mathbf{1}_{\mathcal{B}^{\ell+1}}^{\ell} = \mathbf{1}_{\mathcal{B}^{\ell+1}}^{\ell}$. Consequently,

$$(D_{\mathcal{B}^{\ell+1}}^{\ell+1})^T \mathbf{1}_{\mathcal{B}^{\ell+1}}^{\ell+1} = \left((P_{\sigma,\mathcal{B}^{\ell+1}}^{\ell})^T (D_{\mathcal{B}^{\ell+1}}^{\ell})^T P_{u,\mathcal{B}^{\ell+1}}^{\ell}\right)\left(Q_{u,\mathcal{B}^{\ell+1}}^{\ell} \mathbf{1}_{\mathcal{B}^{\ell+1}}^{\ell}\right) = (P_{\sigma,\mathcal{B}^{\ell+1}}^{\ell})^T (D_{\mathcal{B}^{\ell+1}}^{\ell})^T \mathbf{1}_{\mathcal{B}^{\ell+1}}^{\ell} = \mathbf{0}. \quad \text{(A.25)}$$

If $\mathcal{B}^{\ell+1}$ is connected, since each $\mathcal{A}_i^{\ell}$ is a connected aggregate of level-ℓ vertices, $\mathcal{B}^{\ell+1}$ can also be viewed as a connected aggregate of level-ℓ vertices. Thus, by the induction hypothesis and Lemma A.3,

$$\mathrm{Null}\left((D_{\mathcal{B}^{\ell+1}}^{\ell+1})^T\right) \subseteq \mathrm{Null}\left((Q_{\sigma,\mathcal{B}^{\ell+1}}^{\ell})^T (D_{\mathcal{B}^{\ell+1}}^{\ell+1})^T\right) = \mathrm{Null}\left((D_{\mathcal{B}^{\ell+1}}^{\ell})^T P_{u,\mathcal{B}^{\ell+1}}^{\ell}\right) = \mathrm{Span}\left\{\, \mathbf{1}_{\mathcal{B}^{\ell+1}}^{\ell+1} \,\right\}.$$

The above set inclusion relation and (A.25) imply $\mathrm{Null}\left((D_{\mathcal{B}^{\ell+1}}^{\ell+1})^T\right) = \mathrm{Span}\left\{\, \mathbf{1}_{\mathcal{B}^{\ell+1}}^{\ell+1} \,\right\}$. $\qquad\square$

B. Another choice of PV trace. In this section, we provide an alternative way to construct a PV trace $\boldsymbol{\sigma}_{\mathcal{F}^{\ell}}^{PV}$ on each face $\mathcal{F}^{\ell} = (\mathcal{A}_i^{\ell}, \mathcal{A}_j^{\ell})$ that satisfies (3.7). Let $\mathcal{A}_{\mathcal{F}^{\ell}} := \mathcal{A}_i^{\ell} \cup \mathcal{A}_j^{\ell}$. Define

$$g_{\mathcal{F}^{\ell}} = \begin{bmatrix} -\mathbf{1}_{\mathcal{A}_i^{\ell}}^{\ell}/\|\mathbf{1}_{\mathcal{A}_i^{\ell}}^{\ell}\|^2 \\ \mathbf{1}_{\mathcal{A}_j^{\ell}}^{\ell}/\|\mathbf{1}_{\mathcal{A}_j^{\ell}}^{\ell}\|^2 \end{bmatrix}.$$

Then consider the following local problem defined on $\mathcal{A}_{\mathcal{F}^{\ell}}$ (i.e., take $\mathcal{B} = \mathcal{A}_{\mathcal{F}^{\ell}}$ in (3.1)):

$$\begin{bmatrix} M_{\mathcal{A}_{\mathcal{F}^{\ell}}}^{\ell} & \left(D_{\mathcal{A}_{\mathcal{F}^{\ell}}}^{\ell}\right)^T \\ D_{\mathcal{A}_{\mathcal{F}^{\ell}}}^{\ell} & \end{bmatrix} \begin{bmatrix} \boldsymbol{\sigma}_{\mathcal{A}_{\mathcal{F}^{\ell}}}^{PV} \\ \mathbf{u} \end{bmatrix} = \begin{bmatrix} 0 \\ g_{\mathcal{F}^{\ell}} \end{bmatrix}. \quad \text{(B.1)}$$

Note that by Lemma A.5, $\mathrm{Null}\left((D_{\mathcal{A}_{\mathcal{F}^{\ell}}}^{\ell})^T\right) = \mathrm{Span}\left\{\, \mathbf{1}_{\mathcal{A}_{\mathcal{F}^{\ell}}}^{\ell} \,\right\}$ and $\left(\mathbf{1}_{\mathcal{A}_{\mathcal{F}^{\ell}}}\right)^T g_{\mathcal{F}^{\ell}} = 0$, so (B.1) is solvable. The actual PV trace is taken as the restriction of the solution $\boldsymbol{\sigma}_{\mathcal{A}_{\mathcal{F}^{\ell}}}^{PV}$ to $\boldsymbol{\Sigma}^{\ell}(\mathcal{F}^{\ell})$. That is, $\boldsymbol{\sigma}_{\mathcal{F}^{\ell}}^{PV} := \boldsymbol{\sigma}_{\mathcal{A}_{\mathcal{F}^{\ell}}}^{PV}|_{\boldsymbol{\Sigma}^{\ell}(\mathcal{F}^{\ell})}$.