A CAM Dissertation Proposal on

Subgrid Upscaling Preconditioners for Darcy Flow

James M. Rath

Committee:
Todd Arbogast (advisor), Mathematics
Steven L. Bryant, Petroleum and Geosystems Engineering
Clinton N. Dawson, Aerospace Engineering and Engineering Mechanics
Robert A. van de Geijn, Computer Sciences
Jack Xin, Mathematics

Institute for Computational Engineering and Sciences
The University of Texas at Austin
August 2003

Abstract

The proposed research for the Computational and Applied Mathematics Ph. D. degree is to use innovative improvements to and applications of a variational multiscale subgrid upscaling procedure to greatly reduce the time needed to simulate Darcy flow problems. Additionally, these techniques are sufficiently abstract to be applied to other model physical problems with likely success.

Simulation of flow through a porous medium on a large scale can be computationally expensive if the flow is resolved at a fine scale. Numerous techniques have been proposed to "upscale" the problem so that computations can be performed on a coarser scale but still retain information about the fine-scale flow and problem data. We use one kind of upscaling — variational multiscale subgrid upscaling — to construct preconditioners for solving the full fine-scale problem.

In one preconditioning method, multigrid techniques are used to form a two-level iterative scheme with a fine-scale smoother as one level, and the upscaling preconditioner as the second level. Like multigrid, this scheme has a convergence factor dependent on both the permeability field (the ratio of maximum to minimum values) and the relative resolution of the fine and coarse grids. However, the upscaling preconditioner is much more effective at capturing the correct coarse-scale behavior than a coarse-scale smoother (as used in multigrid). Further, the upscaling preconditioner has an operation count similar to that of multigrid, and the extra subgrid computations exhibit perfect parallelism.

In another method, the coarse basis for the variational multiscale subgrid upscaling is modified to allow the possibility of the upscaled solution coinciding with the full fine-scale solution. Starting with the ordinary upscaling solution, iterative nonlinear optimization techniques are applied to update the basis for the upscaling problem. Under suitable conditions, the modified upscaled solution will converge superlinearly to the fine-scale solution. The method only requires solving upscaled problems (an "inexpensive" operation) and some matrix-vector multiplies to find the residual of the fine-scale problem.

The proposed research is multidisciplinary and requires developments and understanding in mathematics, computer science, and engineering. Although there is a degree of overlap between the problems studied and methods used in these disciplines, some goals specific to each area are listed below.

Area A: A-priori error analysis of finite element methods as applied to linear elliptic partial differential equations will be a necessary element to show convergence of the proposed methods. For the two-level method we need to develop error estimates for the smoothing and approximation properties. For the nonlinear method we need to find the relationship between the error in the extra degrees of freedom and the error in the related modified upscaled problem. An appropriate norm for the error in the extra degrees of freedom is needed (perhaps they can be viewed as traces of fluxes on coarse cell boundaries).

Area B: Research of analysis of convergence of iterative schemes for solving large, sparse linear systems is necessary. Both methods use Schur complement techniques (on both subgrid problems and velocities), and (preconditioned) conjugate gradient and Krylov ideas. The two-level scheme requires analysis of smoothers and the performance of a multigrid-like scheme. The nonlinear method requires study of the details of Newton, quasi-Newton, and secant methods. For each of these three, the problem of global convergence needs to be addressed. For the quasi-Newton methods, there is also the choice of approximation to the Jacobian and how it is factored. For the secant method there are the issues of QR updates, the choice of initialization of the approximate Jacobian, and the bounded deterioration of it. Consideration needs to be made of high performance computing and parallelism concerns when designing and implementing algorithms.

Area C: Efficient, accurate Darcy flow simulations are the prime motivation for this research. We need to detail the performance of the proposed algorithms in different geostatistical settings to guide practitioners' use. We need to be able to say how quickly and how well we can approximate large-scale features of the flow. We want to know if long-range correlations in the problem data present any theoretical or practical difficulties for the algorithms, and be able to present some engineering conclusions about the effect of such correlations on flow. We need to provide a well-documented implementation of the proposed algorithms in parallel for three dimensional problems (and perhaps extend it to handle two-phase miscible and/or immiscible displacements). It should be able to model both rate wells and Peaceman bottom-hole-pressure wells.

Contents

List of Figures

1 Introduction

Darcy's law describes fluid flow through a porous medium. It is an empirical law that asserts bulk flow of a fluid through the medium is proportional to the gradient of the pressure across the medium (accounting for hydrostatic differences from gravity) [25, 9, 59, 36]:

$$\mathbf{u} = -\frac{\kappa}{\mu}\left(\nabla p - \rho \mathbf{g}\right)$$

Darcy's law has found wide applicability in modeling subsurface flows, and has been generalized to model multicomponent and multiphase flows. (The above differential form is itself a generalization of the relation Darcy formulated.) Our primary interest is in using Darcy's law to model oil reservoir and groundwater contaminant flow.

Darcy's law alone is insufficient to describe the physics: conservation of mass (the continuity equation) and equations of state (relating density, viscosity, and permeability to phase fraction and temperature) are necessary. In the applications being considered, there is often a need for the velocity to be very accurate and to strictly (locally) observe mass conservation; hence our focus on mixed methods [55, 30, 26, 34]. Although our presentation ignores aspects of multiphase flow, the proposed ideas and software can readily be adapted to model such flows. We do not account for effects of temperature changes in the model.

1.1 Heterogeneity and why it is a problem

Geostatistical modeling is used to generate the necessary data (porosity and permeability) to specify the problem to be approximated [29]. This data is typically given at a very fine resolution [32], but the goal is to predict long-range flow behavior; it is tempting, then, to approximate the problem at a very coarse scale. However, fine-scale features of the problem data can have very large effects on the coarse-scale flow behavior [32]. Therein lies one big difficulty: it is necessary to resolve the flow at very fine scales resulting in computationally poor conditioned problems to solve [37]. Moreover, the resolution cannot be reduced to shrink the size of the system: (1) heterogeneity in the permeability (irregular, short spatial-scale jumps) means p-refinements (high-order approximations) will not help, and (2) spatially-limited resolution and spatially-uniform heterogeneity means h-refinements (coarse scaling) will not help either.

The fine-scale resolution necessary in simulations makes for poor conditioning, yet there is still another difficulty: the jumps in the permeability can sometimes be quite severe (several orders of magnitude changes) between nearby locations (see for example Figure 4). This makes our computation of an approximation even more poorly conditioned. The more heterogeneous and fine-scale the problem, the more computationally expensive it becomes as all known direct/iterative linear solvers have behavior which worsens with greater condition number [39]. We seek to broaden the range of problems that are computationally feasible.

1.2 Proposed solution techniques

One approach used to overcome the poor-conditioning in modeling multiscale phenomena is upscaling [32]. In upscaling techniques, an averaging process is used to determine the influence of fine-scales on the simulation and to adjust the coarse-scale computations accordingly. Typically an upscaled model will more accurately predict the coarse flow, and will also correctly display fine-scale features of the flow. With mixed variational multiscale subgrid upscaling [1, 44], the scales are split into subgrid and coarse parts before approximating, thereby keeping all the fine-scale information in the model. The mixed framework keeps strict conservation of mass.

Since one goal of upscaling is to obtain a better coarse-scale approximation of the flow, it is a natural idea to try substituting the upscaled approximation for an ordinary coarse-scale approximation in a multigrid scheme. The upscaling substitution is also natural in that it is about as computationally expensive as the ordinary coarsening. This substitution forms the basis for one of the proposed algorithms.

In upscaling, the flexibility of (number of degrees of freedom in) our approximation is reduced in order to obtain a smaller or better algebraic problem to solve. The second proposed algorithm attempts to reintroduce the necessary flexibility into the upscaled model to be able to capture the fine-scale solution. A sequence of upscaled problems is solved in which the problem structure (not the problem data) is gradually evolved towards a problem which has a solution which is functionally equivalent to the fine-scale solution. This evolution uses nonlinear optimization techniques which have the potential of superlinear convergence. Each step in the optimization only requires solving an "upscaled" problem and determining its fine-scale residual.

1.3 Summary

The emphasis in the proposed research will be to provide new tools for improving the speed and accuracy of numerical simulation of systems exhibiting significant heterogeneity on a fine spatial scale, rather than developing new theories of transport. This research will emphasize the use of mixed variational multiscale subgrid upscaling to reduce the time necessary to accurately model these flows. Each of the proposed methods attempts to solve a smaller system (with much better conditioning and a special algebraic structure) to obtain an approximation to the full system.

A description of mixed variational multiscale subgrid upscaling is included in Section 2 in order to introduce some terminology and notation, and to remind the reader of how this method works. The next two sections describe the new preconditioners: a two-level iterative method that uses smoothing and subgrid upscaling to solve fine-scale, heterogeneous problems in Section 3; and a nonlinear optimization scheme for solving problems in Section 4. Some of the modeling applications of this work are set out in Section 5, and the software that has been and will be implemented to carry out simulations using the ideas in this proposal is detailed in Section 6.

2 Description of Mixed Variational Multiscale Subgrid Upscaling

Because of the complexity of the proposed algorithms, it is useful to first describe the underlying method and introduce some terminology and notation.

Use Ω to denote the spatial extent of the porous medium and assume that it is a connected, convex, polygonal domain in $\mathbb{R}^n$ (where n is 1, 2, or 3). Let p and $\bar{\mathbf{u}}$ denote the unknown pressure and macroscopic velocity. Our model problem in the interior of the domain consists of Darcy's Law

$$\bar{\mathbf{u}} = -\frac{\kappa}{\mu}\left(\nabla p - \rho \mathbf{g}\right), \tag{1}$$

and the continuity equation

$$\nabla \cdot \bar{\mathbf{u}} = f, \tag{2}$$

where κ, μ, ρ, and $\mathbf{g}$ are the (spatially-varying) permeability, viscosity, density, and gravity, and f is a source term. On one piece of the boundary Γ_N, specify the flow

$$\mathbf{u} \cdot \nu = g_N, \tag{3}$$

and on another, Γ_D, specify the pressure

$$p = g_D. \tag{4}$$

The pieces of the boundary add up to the whole $\partial\Omega = \bar{\Gamma}_N \cup \bar{\Gamma}_D$, the pieces are disjoint $\Gamma_N \cap \Gamma_D = \emptyset$, and ν is the unit outer normal vector to $\partial\Omega$. A cell-centered finite difference method (CCFD) is used to approximate flow equations for the pressure.

The proposed methods, however, apply to the problem (1)–(4) in mixed variational form. Let

$$\mathbf{V} = \left\{ \mathbf{v} \in H(\mathrm{div}, \Omega) \mid \mathbf{v} \cdot \nu = 0 \text{ on } \Gamma_N \right\},$$

$W = L^2(\Omega)$, and $\mathbf{v}_{g_N}$ be an element of $H(\mathrm{div}, \Omega)$ such that $\mathbf{v}_{g_N} \cdot \nu = g_N$ on Γ_N and $\mathbf{v}_{g_N} \cdot \nu = 0$ on Γ_D. (An important special case occurs if $\Gamma_N = \partial\Omega$. In that case, use $W = L^2/\mathbb{R}$ instead; a compatibility condition on f and g_N is also required.) The problem is then to find $\mathbf{u} \in \mathbf{V}$ and $p \in W$ such that

$$(\nabla \cdot \mathbf{u}, w) = (b, w) \qquad \forall w \in W, \tag{5}$$

$$(\mathbf{k}\mathbf{u}, \mathbf{v}) - (p, \nabla \cdot \mathbf{v}) = (\mathbf{c}, \mathbf{v}) - (g_D, \mathbf{v} \cdot \nu)_{\Gamma_D} \qquad \forall \mathbf{v} \in \mathbf{V}. \tag{6}$$

The substitutions $\mathbf{k} = \mu\kappa^{-1}$, $b = f - \nabla \cdot \mathbf{v}_{g_N}$, and $\mathbf{c} = \rho\mathbf{g} - \mathbf{k}\mathbf{v}_{g_N}$ have been made to simplify the notation. Note that the velocity $\bar{\mathbf{u}} = \mathbf{u} + \mathbf{v}_{g_N}$ is the solution to the original problem (1)–(4).

In order for either formulation of the problem to have a solution, the data needs to meet some regularity constraints. In the variational formulation, one of interest in this proposal is that $\mathbf{k}$ must be a symmetric rank-two tensor in $(L^\infty(\Omega))^{n\times n}$ (denote its essential supremum by k^*), and must be uniformly elliptic (denote its essestial infimum by k_*). The ratio of these two values k^*/k_* will have an impact on the performance of the proposed preconditioners.

2.1 A mixed finite element approximation

For a fine-scale problem, assume the permeability data $\mathbf{k}$ is given as piecewise constant on a grid with spacing h, and that measurement uncertainties prevent specifying it at any finer resolution. It then makes sense to find solutions at the same resolution, and no finer. That is, find $\mathbf{u}_h \in \mathbf{V}_h \subset \mathbf{V}$ and $p_h \in W_h \subset W$ such that

$$(\nabla \cdot \mathbf{u}_h, w_h) = (b, w_h) \qquad \forall w_h \in W_h, \tag{7}$$

$$(\mathbf{k}\mathbf{u}_h, \mathbf{v}_h) - (p_h, \nabla \cdot \mathbf{v}_h) = (\mathbf{c}, \mathbf{v}_h) - (g_D, \mathbf{v}_h \cdot \nu)_{\Gamma_D} \qquad \forall \mathbf{v}_h \in \mathbf{V}_h, \tag{8}$$

where $W_h \times \mathbf{V}_h$ are Raviart-Thomas-Nedelec elements of order zero (RT0) [52, 18, 54] on the given grid. See Figure 1 for an illustration of the degrees of freedom of these elements.

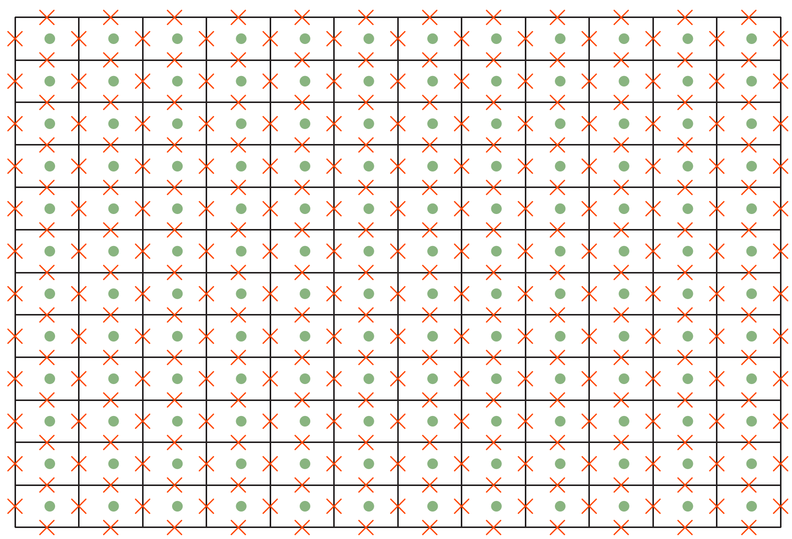

Figure 1: Degrees of freedom for 2-D RT0 elements on a fine rectangular grid: $\times$ normal velocity degrees of freedom, and $\bullet$ pressure degrees of freedom.

Using quadrature — the trapezoidal rule — to compute the term $(\mathbf{k}\mathbf{u}_h, \mathbf{v}_h)$ is equivalent to solving the original problem (5)–(6) using cell-centered finite differences [55, 6].

If a coarse-scale approximation is desired instead, then find $\mathbf{u}_H \in \mathbf{V}_H \subset \mathbf{V}$ and $p_H \in W_H \subset W$ such that

$$(\nabla \cdot \mathbf{u}_H, w_H) = (b, w_H) \qquad \forall w_H \in W_H, \tag{9}$$

$$(\mathbf{k}\mathbf{u}_H, \mathbf{v}_H) - (p_H, \nabla \cdot \mathbf{v}_H) = (\mathbf{c}, \mathbf{v}_H) - (g_D, \mathbf{v}_H \cdot \nu)_{\Gamma_D} \qquad \forall \mathbf{v}_H \in \mathbf{V}_H. \tag{10}$$

where $W_H \times \mathbf{V}_H$ are RT0 elements or, perhaps, Brezzi-Douglas-Duràn-Fortin order one (BDDF1) [16] elements) on a grid with spacing H, an integer multiple of h. See Figure 2 for an illustration.

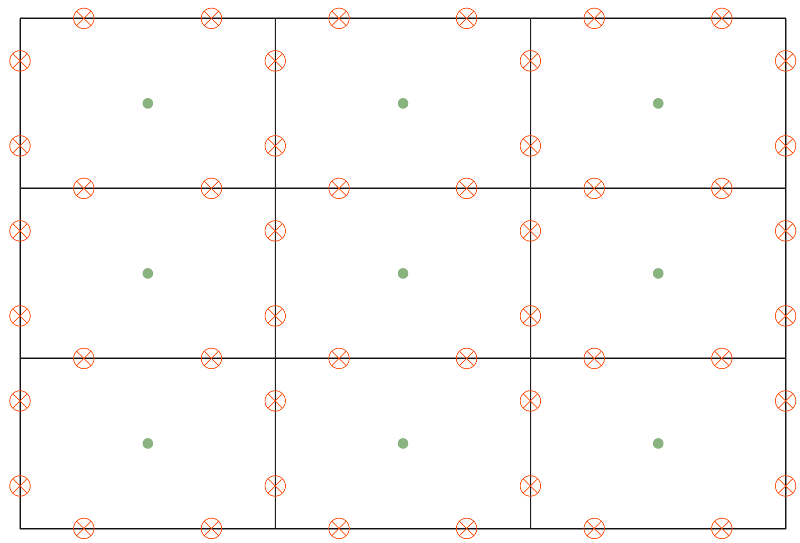

Figure 2: Degrees of freedom for 2-D BDM1 [17] elements on a coarse rectangular grid: $\otimes$ (linear) normal velocity degrees of freedom, and $\bullet$ pressure degrees of freedom.

For the RT0 elements, some error estimates [18, 31, 54] for a grid size h are

$$\|\mathbf{u} - \mathbf{u}_h\|_0 \leq C\|\mathbf{u}\|_1 h \qquad = O(h), \tag{11}$$

$$\|p - p_h\|_0 \leq C\|p\|_2 h \qquad = O(h). \tag{12}$$

For BDM1/BDDF1 elements the velocity is an order of h more accurate

$$\|\mathbf{u} - \mathbf{u}_h\|_0 \leq C\|\mathbf{u}\|_2 h^2 \qquad = O(h^2). \tag{13}$$

2.2 The upscaling technique

The subgrid upscaling technique uses the mixed finite element method (MFEM) with a variational multiscale [44] technique.

Before any approximation, decompose $\mathbf{V}$ and W (using a chosen coarse grid) so that

(i) no information is lost, $\mathbf{V} = \mathbf{V}_c \oplus \delta\mathbf{V}$, $W = W_c \oplus \delta W$;

(ii) mass is conserved, $\nabla \cdot \mathbf{V}_c = W_c$, $\nabla \cdot \delta\mathbf{V} = \delta W$; and

(iii) the fine-scale velocities are locally supported over the coarse grid, $\delta\mathbf{V} \cdot \nu = 0$ on the boundary of each coarse cell.

(Section 3.1 — in particular Theorem 3.3 — of [1] show that this decomposition is always possible and satisfies some additional necessary properties. There is also more than one way to choose the decomposition.)

Note that Condition (iii) above allows us to disconnect the subgrid problems from one another, but not from the coarse grid problem. A Green's or influence function approach is employed to effect a forward elimination of the subgrid problems into the coarse problem, and — once the coarse problem has been solved — a back substitution to recover the subgrid solutions. The two can then be recombined to obtain a fine scale solution.

Since $\mathbf{V}$ and W have been decomposed into direct sums, the problem (5)–(6) can be described as follows. Find $\mathbf{u}_c \in \mathbf{V}_c$ and $p_c \in W_c$, and $\delta\mathbf{u} \in \delta\mathbf{V}$ and $\delta p \in \delta W$ such that on the coarse scale

$$(\nabla \cdot (\mathbf{u}_c + \delta\mathbf{u}), w_c) = (b, w_c) \qquad \forall w_c \in W_c, \tag{14}$$

$$(\mathbf{k}(\mathbf{u}_c + \delta\mathbf{u}), \mathbf{v}_c) - (p_c + \delta p, \nabla \cdot \mathbf{v}_c) = (\mathbf{c}, \mathbf{v}_c) - (g_D, \mathbf{v}_c \cdot \nu)_{\Gamma_D} \quad \forall \mathbf{v}_c \in \mathbf{V}_c, \tag{15}$$

and on the subgrid scale

$$(\nabla \cdot (\mathbf{u}_c + \delta\mathbf{u}), \delta w) = (b, \delta w) \qquad \forall \delta w \in \delta W, \tag{16}$$

$$(\mathbf{k}(\mathbf{u}_c + \delta\mathbf{u}), \delta\mathbf{v}) - (p_c + \delta p, \nabla \cdot \delta\mathbf{v}) = (\mathbf{c}, \delta\mathbf{v}) \qquad \forall \delta\mathbf{v} \in \delta\mathbf{V}. \tag{17}$$

Then $\mathbf{u} = \mathbf{u}_c + \delta\mathbf{u}$ and $p = p_c + \delta p$ solve the original variational problem (5)–(6).

To define the δ-subgrid operator and perform the forward elimination of the coarse components, rewrite the subgrid scale equations with coarse components on the right-hand-side. That is, consider the coarse components as sums of basis elements

$$p_c = \sum_i \alpha_i w_c^i \qquad \text{and} \qquad \mathbf{u}_c = \sum_j \beta_j \mathbf{v}_c^j. \tag{18}$$

The parts of the δ-operator are obtained by substituting these expressions in (16) and (17) and solving for the α_i- and β_j-influence function coefficients.

The constant part of the δ operator is given by solving

$$(\nabla \cdot \delta\bar{\mathbf{u}}, \delta w) = (b, \delta w) \qquad \forall \delta w \in \delta W, \tag{19}$$
$$(\mathbf{k}\delta\bar{\mathbf{u}}, \delta\mathbf{v}) - (\delta\bar{p}, \nabla \cdot \delta\mathbf{v}) = (\mathbf{c}, \delta\mathbf{v}) \qquad \forall \delta\mathbf{v} \in \delta\mathbf{V}. \tag{20}$$

The W_c-linear part of the δ operator is given by solving

$$(\nabla \cdot \delta\tilde{\mathbf{u}}_i, \delta w) = 0 \qquad \forall \delta w \in \delta W, \tag{21}$$
$$(\mathbf{k}\delta\tilde{\mathbf{u}}_i, \delta\mathbf{v}) - (\delta\tilde{p}_i, \nabla \cdot \delta\mathbf{v}) = (w_c^i, \nabla \cdot \delta\mathbf{v}) \qquad \forall \delta\mathbf{v} \in \delta\mathbf{V}. \tag{22}$$

The $\mathbf{V}_c$-linear part of the δ operator is given by solving

$$(\nabla \cdot \delta\hat{\mathbf{u}}_j, \delta w) = -(\nabla \cdot \mathbf{v}_c^j, \delta w) \qquad \forall \delta w \in \delta W, \tag{23}$$
$$(\mathbf{k}\delta\hat{\mathbf{u}}_j, \delta\mathbf{v}) - (\delta\hat{p}_j, \nabla \cdot \delta\mathbf{v}) = -(\mathbf{k}\mathbf{v}_c^j, \delta\mathbf{v}) \qquad \forall \delta\mathbf{v} \in \delta\mathbf{V}. \tag{24}$$

Then

$$\begin{aligned} \delta p &= \sum \alpha_i \delta\tilde{p}_i + \sum \beta_j \delta\hat{p}_j + \delta\bar{p} \\ &\equiv \delta\tilde{p}(p_c) \quad + \delta\hat{p}(\mathbf{u}_c) \quad + \delta\bar{p} \end{aligned} \tag{25}$$

and

$$\begin{aligned} \delta\mathbf{u} &= \sum \alpha_i \delta\tilde{\mathbf{u}}_i + \sum \beta_j \delta\hat{\mathbf{u}}_j + \delta\bar{\mathbf{u}} \\ &\equiv \delta\tilde{\mathbf{u}}(p_c) \quad + \delta\hat{\mathbf{u}}(\mathbf{u}_c) \quad + \delta\bar{\mathbf{u}} \end{aligned} \tag{26}$$

are implicit expressions for δp and $\delta\mathbf{u}$ in terms of p_c and $\mathbf{u}_c$. Note that $\delta\tilde{p}(w_c^i) = \delta\tilde{p}_i$, $\delta\hat{p}(\mathbf{v}_c^j) = \delta\hat{p}_j$, $\delta\tilde{\mathbf{u}}(w_c^i) = \delta\tilde{\mathbf{u}}_i$, and $\delta\hat{\mathbf{u}}(\mathbf{v}_c^j) = \delta\hat{\mathbf{u}}_j$.

Return to the coarse scale equations (14)–(15). Rewrite them substituting in the influence-function expressions for δp and $\delta\mathbf{u}$, and gathering the coarse coefficients α_i and β_j out front. Equations for the coarse components' coefficients result

$$\sum_i \alpha_i(\nabla\cdot\delta\tilde{\mathbf{u}}_i, w_c) + \sum_j \beta_j(\nabla\cdot(\mathbf{v}_c^j + \delta\hat{\mathbf{u}}_j), w_c) = (b - \nabla\cdot\delta\bar{\mathbf{u}}, w_c) \qquad \forall w_c \in W_c, \quad (27)$$

$$\begin{aligned}\sum_i \alpha_i(\mathbf{k}\delta\tilde{\mathbf{u}}_i, \mathbf{v}_c) &+ \sum_j \beta_j(\mathbf{k}(\mathbf{v}_c^j + \delta\hat{\mathbf{u}}_j), \mathbf{v}_c) \\ &- \sum_i \alpha_i(w_c^i + \delta\tilde{p}_i, \nabla\cdot\mathbf{v}_c) - \sum_j \beta_j(\delta\hat{p}_j, \nabla\cdot\mathbf{v}_c) = (\mathbf{c} - \mathbf{k}\delta\bar{\mathbf{u}}, \mathbf{v}_c) + (\delta\bar{p}, \nabla\cdot\mathbf{v}_c) \\ &\qquad \forall \mathbf{v}_c \in \mathbf{V}_c, \quad (28)\end{aligned}$$

or using the operator notation

$$(\nabla\cdot(\mathbf{u}_c + \delta\tilde{\mathbf{u}}(p_c) + \delta\hat{\mathbf{u}}(\mathbf{u}_c)), w_c) = (b - \nabla\cdot\delta\bar{\mathbf{u}}, w_c) \qquad \forall w_c \in W_c, \quad (29)$$

$$\begin{aligned}(\mathbf{k}(\mathbf{u}_c &+ \delta\tilde{\mathbf{u}}(p_c) + \delta\hat{\mathbf{u}}(\mathbf{u}_c)), \mathbf{v}_c) \\ &-(p_c + \delta\tilde{p}(p_c) + \delta\hat{p}(\mathbf{u}_c), \nabla\cdot\mathbf{v}_c) = (\mathbf{c} - \mathbf{k}\delta\bar{\mathbf{u}}, \mathbf{v}_c) + (\delta\bar{p}, \nabla\cdot\mathbf{v}_c) \\ &\qquad - (g_D, \mathbf{v}_c\cdot\nu)_{\Gamma_D} \qquad \forall \mathbf{v}_c \in \mathbf{V}_c, \quad (30)\end{aligned}$$

Solve these equations for α_i and β_j (that is, $\mathbf{u}_c$ and p_c), and write the solution as

$$\begin{aligned}p &= p_c \qquad + \delta p \\ &= \sum_i \alpha_i w_c^i + \sum_i \alpha_i\delta\tilde{p}_i + \sum_j \beta_j\delta\hat{p}_j + \delta\bar{p}\end{aligned} \qquad (31)$$

and

$$\begin{aligned}\mathbf{u} &= \mathbf{u}_c \qquad + \delta\mathbf{u} \\ &= \sum_j \beta_j\mathbf{v}_c^j + \sum_i \alpha_i\delta\tilde{\mathbf{u}}_i + \sum_j \beta_j\delta\hat{\mathbf{u}}_j + \delta\bar{\mathbf{u}}.\end{aligned} \qquad (32)$$

Lastly, (29)–(30) can be written in symmetric form by rearranging terms and using some identities not identified here. See [1] for details, as well as proofs that each of the above problems are well-posed.

Also note that the δ-subgrid operator has a natural definition. No information has been lost in the reformulation of the problem, and no ad-hoc assumptions have been reintroduced to simplify the action of the δ operator.

2.3 Approximating the upscaled problem

Only after deciding on the decomposition is the approximation then made. We are free to pick discretizations for each subgrid independent of each of the others. The discretization of the coarse spaces is only constrained by the already chosen coarse grid. Standard mixed finite element spaces such as Raviart-Thomas or BDDF elements will do.

The subgrid upscaling technique has a number of advantages. Each subgrid problem is independent from the others because of the Neumann boundary conditions along coarse cell edges; the subgrid problems can be solved in parallel with no communication of boundary data needed across coarse cell edges. Each subgrid problem also has many fewer degrees of freedom than the whole problem (by about a factor of the number of coarse grid cells); these problems are far better conditioned, and may be amenable to a direct solver. Each subgrid problem has the same left-hand side and many different right-hand sides creating an opportunity for computational efficiency. The coarse problem has many fewer degrees of freedom than the whole problem (by about a factor of the number of fine cells chosen per coarse cell) and so is also much better conditioned. If the coarse grid spacing is a small multiple of the fine grid spacing, the work in solving the upscaled problem is nearly all done in solving the coarse problem. Lastly, there is only one coarse grid (not many as in multigrid).

Further, the number of degrees of freedom in the upscaling space is nearly as many as that in the fine space (compare Figure 1 with Figure 3 below). However, for small upscaling factors the upscaled solution is only about as computationally expensive as a coarse solution. This would be only mildly interesting if the upscaling solution was only as accurate as a coarse solution, but in fact it captures much more (see example below in Subsection 2.5; also see Figures 9–16 in Section 3 on the multigrid-like preconditioner).

There are some algorithm implementation issues to be considered. First, the elements of δW_h may not be easy to compute with. For instance, suppose RT0 elements are used for W_h and W_H, and $\delta W_h = (W_H)^\perp$ with the complement taken in W_h. Then elements of δW_h do not have local support on fine cells — they must have zero average on coarse cells. A computational trick explained in [4] and [5] avoids this complication.

Second, there is obvious parallelism in the subgrid problems since each one is independent of all the others, and the coarse-scale influence functions allow disconnecting the subgrid from the coarse scale. However, for coarse-scale problem there is not any obvious parallelism. To overcome this lack of parallelism, one could use a domain decomposition method [38] or another technique.

Lastly, all of the problems being considered are saddle point problems because of the mixed formulation. For small upscaling factors, the subgrid problems may be solved directly. In other cases and for the coarse problem, iterative solvers may be necessary. The Uzawa algorithm [18, 11] may be used to solve the indefinite system. Also, the hybrid mixed method [8, 18] or a Schur complement could be used to transform the indefinite problem into a positive definite one.

2.4 Sample approximation of the upscaled problem

As an example, use RT0 elements to approximate δW and $\delta \mathbf{V}$, denoted δW_h and $\delta \mathbf{V}_h$, and BDDF1 elements for W_c and $\mathbf{V}_c$, denoted W_H and $\mathbf{V}_H$. Let $W_{H,h} = W_H \oplus \delta W_h$ and $\mathbf{V}_{H,h} = \mathbf{V}_H \oplus \delta \mathbf{V}_h$. Elements of W_H are functions which are piecewise constant on coarse cells. Elements of δW_h are functions which are piecewise constant on fine cells, but must have a zero average over each coarse cell. Elements of $\mathbf{V}_H$ are vector-valued functions where the components normal to coarse cell faces are continuous across those faces, and vary linearly along those faces. Lastly, elements of $\delta \mathbf{V}_h$ are vector-valued functions where the components normal to fine cell faces are continuous across those faces, and are constant along those faces; the normal components across coarse cell faces are zero. See Figure 3 for an illustration.

No closure assumption is made regarding the permeability. Accuracy is simply a question of approximation theory — how well does $\mathbf{V}_{H,h}$ approximate $\mathbf{V}$? — and not a question of how much the physics is changed. Defining effective coarse permeabilities is not necessary.

The closure assumption can be said to be that all flux across a coarse element face is due to the coarse-scale functions. (The δ-subgrid operator is approximated using well-worn techniques; it is not altered in an ad-hoc fashion to facilitate computation.) In this example, to compensate for the coarseness of the restriction of velocities along coarse edges, higher order accurate basis functions have been employed.

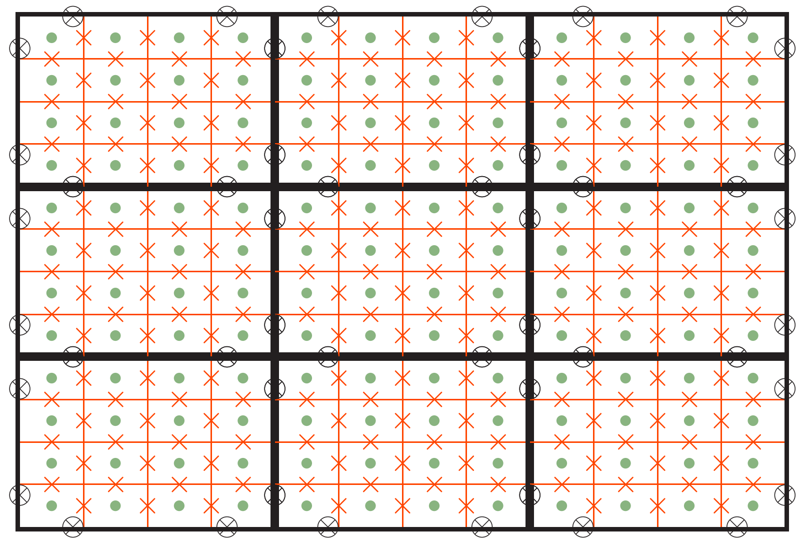

Figure 3: Degrees of freedom for 2-D RT0/BDDF1 elements on an "upscaled" rectangular grid: $\otimes$ coarse velocity (linear) degrees of freedom, $\times$ subgrid velocity degrees of freedom, and $\bullet$ pressure degrees of freedom. [4]

For the BDDF1/RT0 elements, some error estimates are [1]

$$\|\mathbf{u}-\mathbf{u}_{H,h}\|_0 \leq C\left\{\|p\|_1 h^2 + \left(\|\mathbf{u}\|_2 + \|\mathbf{u}\cdot\nu\|_{2,\Gamma_D}\right)H^2\right\} = O(H^2), \tag{33}$$

$$\|p-p_{H,h}\|_0 \leq C\left\{\|p\|_1 h + \|\nabla\cdot\mathbf{u}\|_1 h^2 + \left(\|\mathbf{u}\|_2 + \|\mathbf{u}\cdot\nu\|_{2,\Gamma_D}\right)H^3\right\} = O(h+H^3). \tag{34}$$

However, the simplicity of the above expressions belies some error analysis complications; for instance, the $\pi_{H,h}$ operator for the upscaled elements is obtained not just from gluing together the operators from the coarse elements π_H and each of the subgrid elements $\delta\pi_{h,E_c}$, but also solving an auxiliary problem (except under additional assumptions). The error estimates above also do not capture all the subtleties of the subgrid upscaled solution (see, for instance, the graphs of the eigenstructure of the subgrid upscaling preconditioner in Section 3). One thing to note, though, is that if H/h is small, then $O(h+H^3) = O(h)$ for the pressure and $O(H^2) = O((H/h)^2h^2)$ for the velocity. A comparison between the upscaling error estimates with those for BDM1/BDDF1 elements on fine grid, (12) and (13), shows that they are the same.

2.5 Some practical results

To show the ability of the subgrid upscaling to approximate problems almost as well as a full fine-scale solution (and far better than a coarse-scale one), some results from a simulated quarter five-spot oil reservoir water flood are presented. The above ideas were adapted to two-phase immiscible displacements, and run on the following example [3].

The domain is a 40 m by 40 m square with a uniform, rectangular, 40×40 grid. The base-10 logarithm of the permeability field is shown in Figure 4; the porosity is 25%. The field initially has a 20% saturation of water. There is an injection well in the lower-left corner, and a production well in the top-right corner. Each has a rate of $0.2\,\mathrm{m}^2/\mathrm{day}$, and water is being injected.

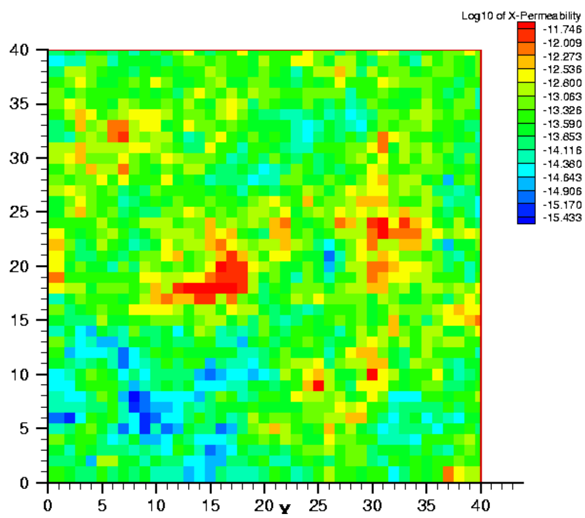

Figure 4: Logarithm of the permeability field (geostatistically generated).

Figures 5, 6, and 7 speak for themselves. The upscaled and fine-scale solutions are certainly close in the eyeball norm; the coarsened solution is not close to either and fails qualitatively to capture some behavior. The upscaled solution requires only about as much time to compute as the coarse solution, but has nearly as many degrees of freedom as the fine solution; hence the "success."

Throughout most of the rest of the paper, only RT0/RT0 elements will be used. Although the ideas are applicable to others, the description, analysis,

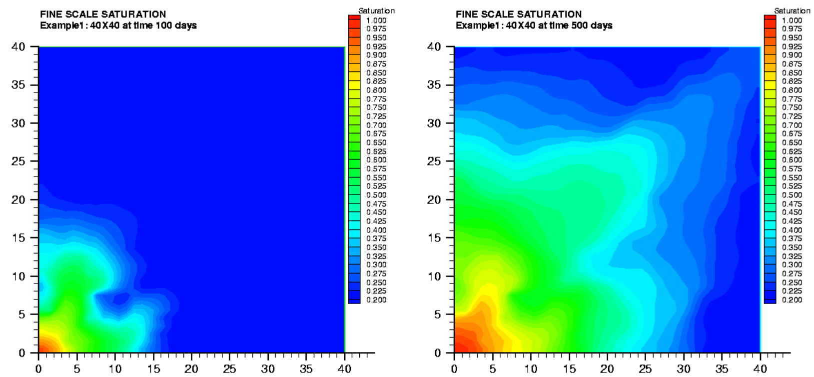

Figure 5: Water saturation contours at 100 and 500 days for the fine 40×40 solution.

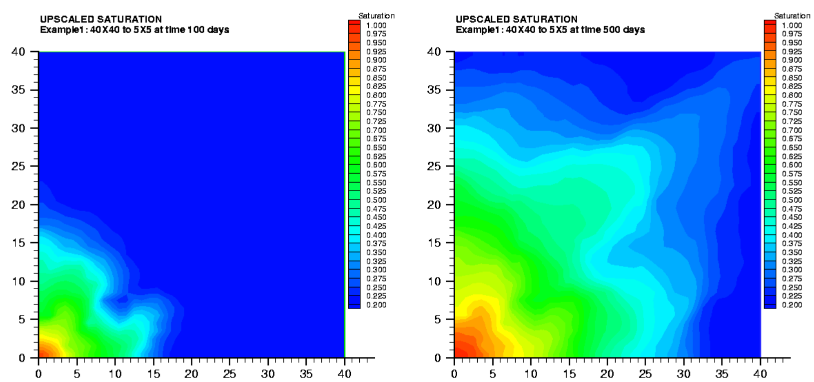

Figure 6: Water saturation contours at 100 and 500 days for the upscaled 5×5 solution.

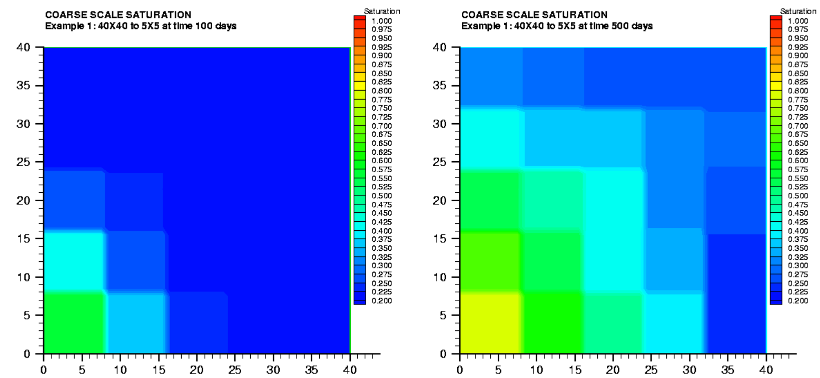

Figure 7: Water saturation contours at 100 and 500 days for the coarse 5×5 solution.

and implementation of higher order combinations gets much more complicated.

2.6 Other upscaling techniques

Upscaling techniques are numerous; surveys of various ones currently under study can be found in [32, 35, 53, 62]. A few descriptions follow. Chen et al. [22] describe their technique as a local/global one: use a global coarse simulation to determine appropriate boundary conditions for local simulations that then determine an effective coarse permeability. Other techniques are local; for instance, periodic boundary conditions can be imposed on the subgrids when computing effective permeabilities as in [10, 58]. Oversampling is another possibility where a local problem (with various boundary conditions) that is slightly larger than a subgrid is used to determine the effective permeability for that subgrid; Chen and Hou [23] and Hou and Wu [43] use a multiscale finite element technique to implement this idea. Wu, Efendiev, and Hou also use this idea [33], and in [65] apply a flow-based gridding dynamic approach to limit the necessary upscaling. Holden and Nielsen [42] have developed a global upscaling method that uses a least-squares problem to determine effective coarse permeabilities.

Some distinguishing features of the upscaling technique used here are the use of mixed methods, the lack of ad-hoc assumptions about the δ-subgrid operator, and the lack of computed effective coarse permeabilities.

3 A Two-level Multigrid-like Scheme

In traditional multigrid, one pairs smoothings on fine and coarse grids together with the intent of each smoother reducing the error in different parts of the spectrum. The coarse-grid smoother reduces the error on large scales, and the fine-grid smoother reduces it on short scales [19].

However, fine-scale details of the problem have an effect on coarse-scale flow behavior. Thus, instead of a simple coarse-grid corrector in multigrid, a corrector that better approximates the true coarse-scale behavior could be used. Since a corrector that is much more computationally expensive than the simple coarse one is not a reasonable choice, it is natural to try an upscaled coarse corrector. This is the basis for the first proposed algorithm. A schematic is shown in Figure 8.

The intuition as to why multigrid works so well is that smoothing the solution iterates on different levels reduces the error in different parts of the spectrum. Towards that end, we have developed code to compute the eigenvectors (and corresponding eigenvalues) for smoothers at each level — the fine-scale level (with ordinary Jacobi smoothing) and the upscale smoothing — and to plot the norm of the gradient of the eigenvectors (considered as functions) versus the corresponding eigenvalues. The matrices analyzed have the form $I - M\,A$; this is from the error update equation $\mathbf{e}_{i+1} = (I - M\,A)\mathbf{e}_i$ that describes the action of the preconditioner M on the error $\mathbf{e}$.

The results of applying the code to several two dimensional examples can be seen in Figures 9–16. Each of the examples has the unit square as the domain with uniform porosity and Dirichlet boundary conditions. The first example has a very simple permeability: it takes one value on the left half of the domain, and another value on the right half (shown in Figure 9).

In Figure 10 a "typical" result for the eigenstructure analysis as applied to the simple two-level permeability field is shown. On the horizontal axis is measured the norm of the gradient of the eigenvector when interpreted as a 2-D pressure field (this measure is a stand-in for frequency). The vertical axis shows on a logarithmic scale the absolute value of the corresponding eigenvalue. The blue dots show the check-mark-like structure for a weighted Jacobi smoother. This picture is similar to ones usually used in multigrid analysis [19]. The cyan dots show the structure one obtains for the corresponding weighted Jacobi smoother on the coarse level. It has the same shape, but the small-eigenvalue dip comes at lower frequencies (coarser modes) as expected. The magenta dots show the structure for the preconditioner that exactly

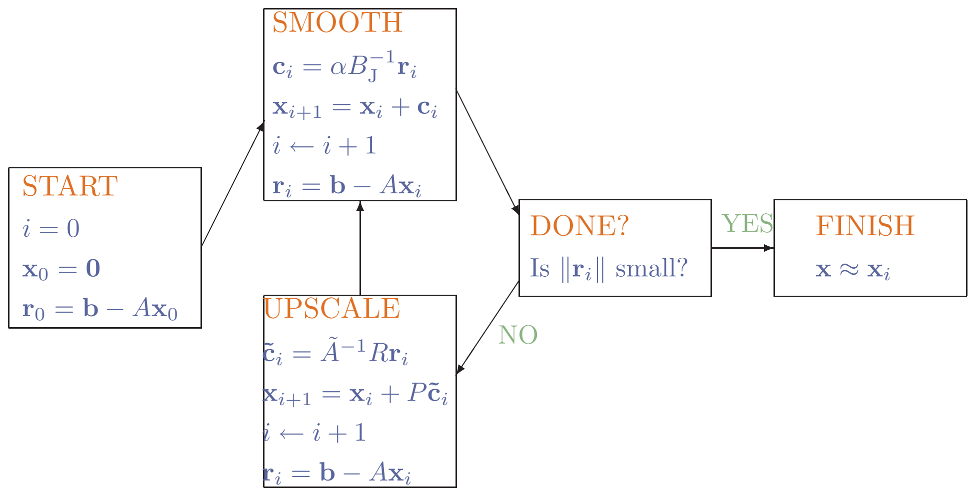

Figure 8: Schematic of a two-level scheme to solve the linear system $A\mathbf{x} = \mathbf{b}$. The matrix A describes the interactions among cells from a CCFD discretization of Darcy's Law and the continuity equation. The pressures $\mathbf{x}$ are the unknowns, and the information from sources, boundary terms, and gravity $\mathbf{b}$ forms the right-hand side. There is also the matrix $\tilde{A}$ for the corresponding upscaling MFEM problem, the ith guess at pressure unknowns $\mathbf{x}_i$, the ith-stage residual $\mathbf{r}_i = \mathbf{b} - A\mathbf{x}_i$, the ith-stage smoothing corrector $\mathbf{c}_i$, and ith-stage upscaling corrector $\tilde{\mathbf{c}}_i$ (pressure and velocity unknowns). Lastly, there is B_{J} from the diagonal elements of A used in Jacobi smoothing step, the restriction operator $R : W_h \to W_{H,h} \times \mathbf{0}$, and the prolongation operator $P : W_{H,h} \times \mathbf{V}_{H,h} \to W_h$. Also note that usually many applications of the smoother may occur in between upscaling steps.

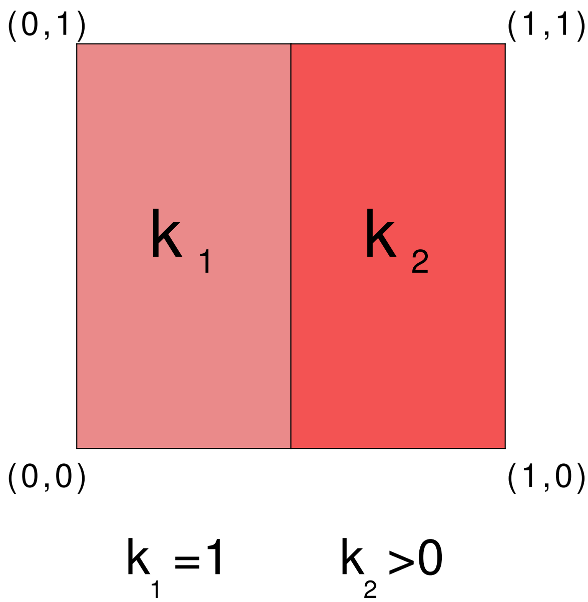

Figure 9: A simple permeability field.

solves the coarse problem. It has eigenvalues greater than one in the coarser modes because the upscaling factor H/h is equal to three, and not two (the more usual result with eigenvalues bounded by one can be seen in the bottom right of Figure 12). Lastly, the red dots show the structure for the upscaling preconditioner. Some notable features are its complicated organization, that coarse modes (over a large range) have quite small eigenvalues, that there is a steady rise in eigenvalues versus frequency, and that the eigenvalues top out at greater than one. However, this maximum occurs near where the weighted Jacobi smoother is performing at its best.

Figure 11 shows the effect of varying h, the grid size, while leaving the upscaling factor H/h and the permeability field alone. As can be seen, the overall structure remains the same independent of h, but the level of detail one sees in the diagrams increases, as does the scale of the frequencies plotted.

Figure 12 shows the effect of varying H/h, the upscaling factor, while leaving the fine grid size h and the permeability field alone. As H/h increases, there is a shift to lower frequencies for the coarse smoother and the upscaling preconditioner. (As noted above, the eigenvalues of coarse modes for the exact coarse preconditioner get larger with increasing H/h.) There is a general

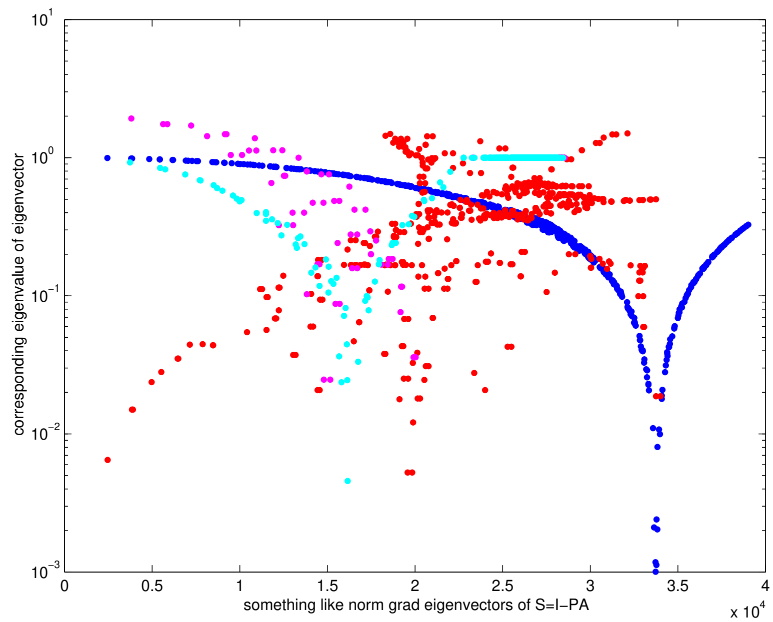

Figure 10: A "typical" eigenstructure result for the smoothers and upscaler (RT0 elements used at all levels). The fine grid used was 24×24, the coarse grid 8×8 (with $H/h = 3$), and $k_2 = 0.1$. The colored dots are eigenpairs for the • coarse smoothing preconditioner, • coarse exact preconditioner, • fine smoothing preconditioner, and • subgrid upscaling preconditioner.

Vary h ; Fix $H/h = 3$; Fix $k_2 = 0.1$

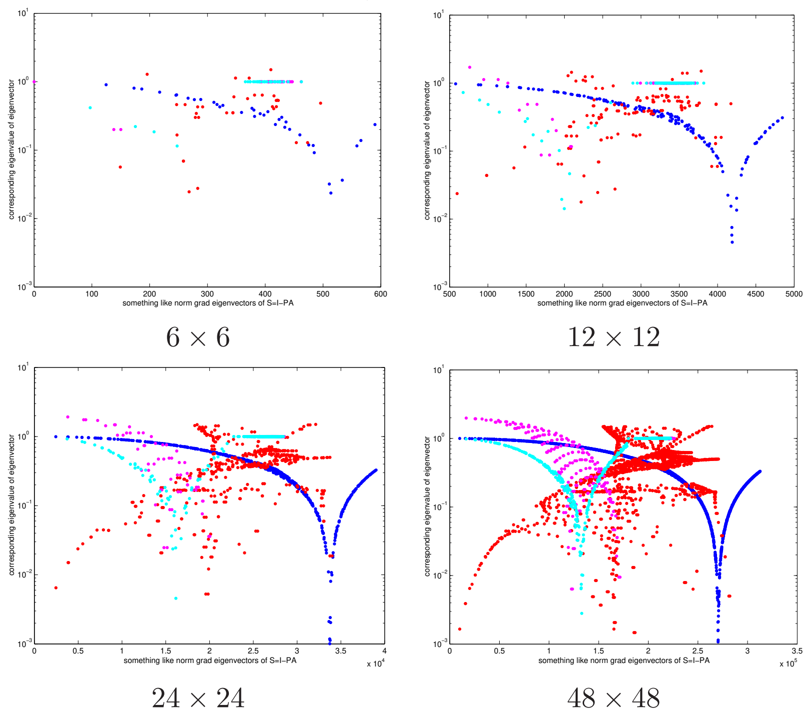

Figure 11: Eigenstructure results for the preconditioners for problems with various fine grid sizes h, but fixed upscaling ratio H/h and fixed permeability. The colored dots are eigenpairs for the • coarse smoothing preconditioner, • coarse exact preconditioner, • fine smoothing preconditioner, and • subgrid upscaling preconditioner.

worsening of performance with increasing H/h as would be expected.

Fix $h = 1/24$; Vary H/h ; Fix $k_2 = 0.1$ w/RT0

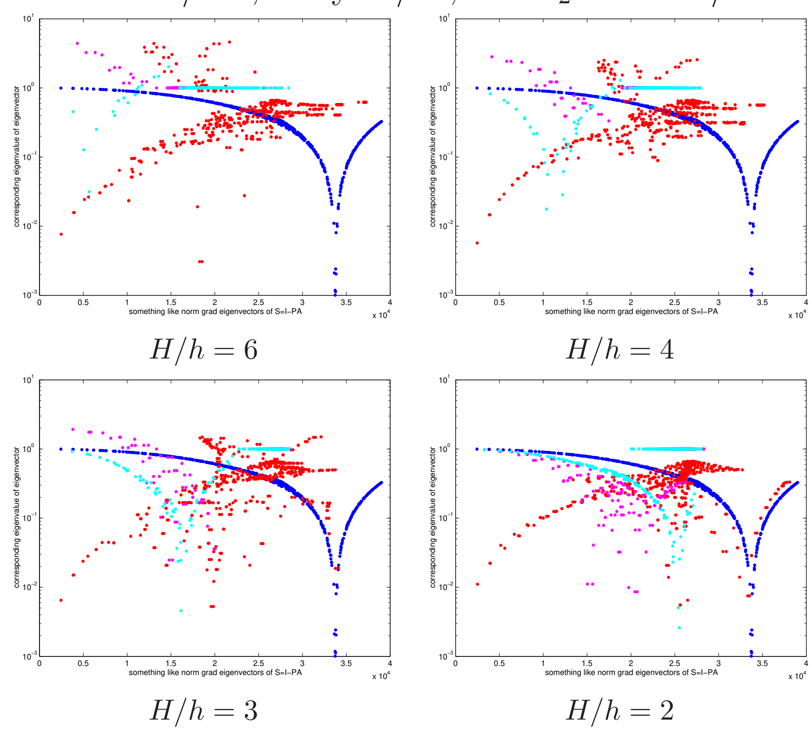

Figure 12: Eigenstructure results for the preconditioners for problems with various upscaling ratios H/h, but fixed fine grid size h and fixed permeability. The colored dots are eigenpairs for the • coarse smoothing preconditioner, • coarse exact preconditioner, • fine smoothing preconditioner, and • subgrid upscaling preconditioner.

Figure 13 shows the Arcoperm permeability field [7], and Figure 15 shows the Brent permeability field [47]. Both show considerably more variation than the simple two-level permability field. They both also have a greater ratio between the greatest and least values of the permeability: the Arcoperm field about a half an order of magnitude, and the Brent field more than four orders of magnitude. For the Arcoperm field, Figure 14 shows the same rich structure as seen for the two-level permeability field, but surprisingly the upscaling preconditioner has a similar performance: small eigenvalues for coarse modes with a general trend towards larger (but not too large) eigenvalues for fine modes. At the greatest upscaling factor H/h of six, the middle modes begin to suffer, too. The results in Figure 16 show the same general trends. However, the middle modes become much more of a problem (note the vertical scale has changed from a high of 10^1 before to 10^2 now).

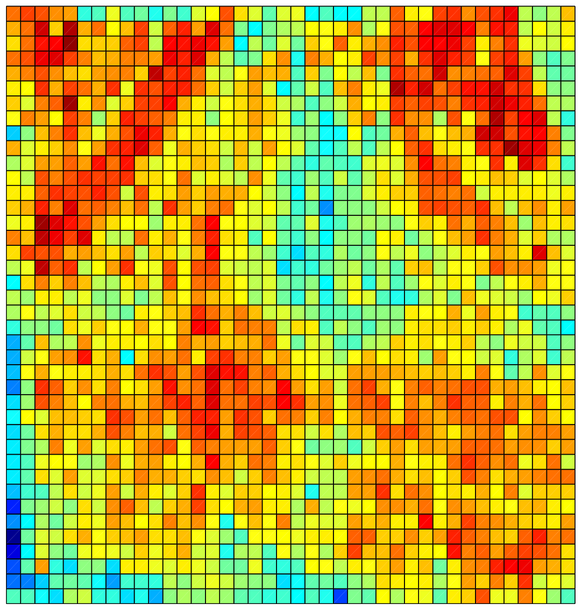

Figure 13: The Arcoperm permeability field. The minimum of the base-10 logarithm of the permeability values is -0.2916 and the maximum 0.1477.

One obvious conclusion to draw from the eigenstructure diagrams is that the performance of the upscaling level suffers in the mid-frequency eigenvectors (see also the convergence histories below). Perhaps Krylov space methods (a CG-like iteration) [20, 39, 27], a third level with an upscaler based on a grid staggered relative to the first upscaler, semi-coarsening in different levels in different directions [24], or perhaps a traditional multigrid coarse smoother might be used to improve the method.

The diagrams also lead to the reasonable inference that the behavior of the proposed two-level scheme depends on H/h and k^*/k_*, but not the absolute level of h. This mirrors the behavior of multigrid, but we have not proven this yet for this scheme.

In this proposal, we have been assuming that a single coarsening of the grid is enough to result in a problem that is computationally inexpensive

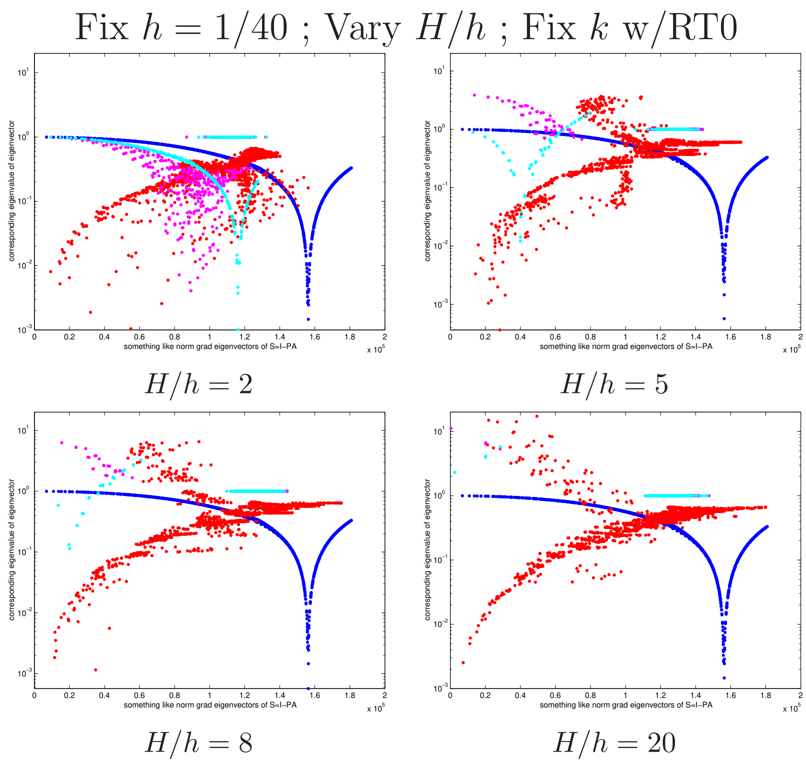

Figure 14: Eigenstructure results for the preconditioners for problems with various upscaling ratios H/h with the Arcoperm permeability field in Figure 13. The colored dots are eigenpairs for the • coarse smoothing preconditioner, • coarse exact preconditioner, • fine smoothing preconditioner, and • subgrid upscaling preconditioner.

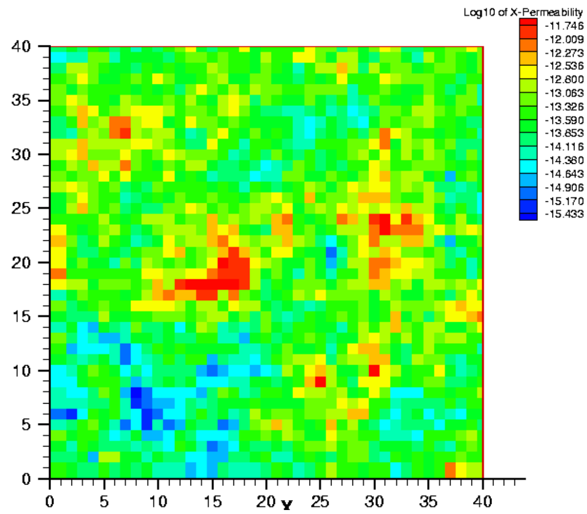

Figure 15: The Brent permeability field. The minimum of the base-10 logarithm of the permeability values is -0.6909 and the maximum 3.5230.

to use. Sometimes this "coarse" grid just is not coarse enough, and the associated problem is *still* too computationally expensive to work with. In this case, a true variational *multi*-scale (and not just two-scale) method for the upscaling level would be appropriate. This might be accomplished say by decomposing $\mathbf{V} = \mathbf{V}_c \oplus \delta\mathbf{V} = \mathbf{V}_{sc} \oplus \Delta\mathbf{V} \oplus \delta\mathbf{V}$ and $W = W_c \oplus \delta W = W_{sc} \oplus \Delta W \oplus \delta W$ into three levels (or more) with a correspoding approximation with multiple levels. The details of this possibility have yet to be worked out. It might also be possible to use traditional algebraic multigrid on the coarsened system that comes from the upscaled problem (29)–(30).

The eigenstructure diagrams also give us a way to estimate how often the Jacobi smoother is applied relative to the number of times the upscaling preconditioner is applied. It is also possible to then estimate the overall error reduction factor for an iteration of our two-level method. This is accomplished by examining frequency by frequency the product of the error reduction factors (eigenvalues) of the smoother and upscaling preconditioner, and by computing the maximum such product. This allows us also to estimate the relative performance of multigrid with the two-level upscaling/smoother method for a given problem. Of course, this is assuming that eigenvectors with like "frequency" are alike; in reality, the eigenbases differ and the comparison for a given frequency is not apples-to-apples.

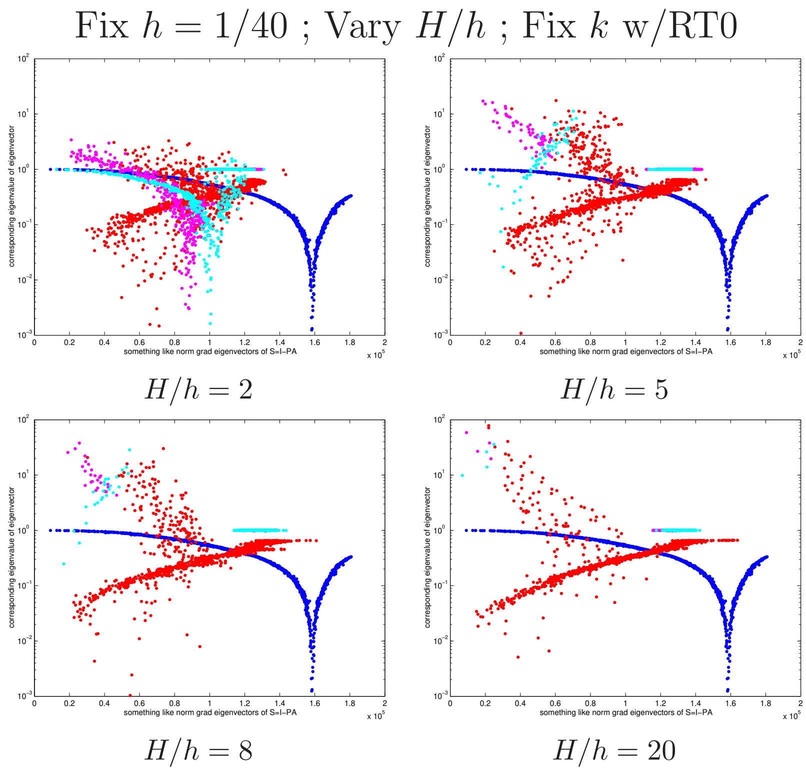

Figure 16: Eigenstructure results for the preconditioners for problems with various upscaling ratios H/h with the Brent permeability field in Figure 15. The colored dots are eigenpairs for the • coarse smoothing preconditioner, • coarse exact preconditioner, • fine smoothing preconditioner, and • subgrid upscaling preconditioner.

Some examples of convergence histories using this analysis are shown in Figure 17. The size of the residuals/errors versus iteration number are plotted for both the two-level upscaling scheme and a two-level multigrid scheme. Significantly faster convergence is obtained with the upscaling than with multigrid (although both display linear convergence and dependence on k^*/k_*). A zero initial guess for the solution is used along with a random source term.

The diagram in the top row shows the convergence history for the simple permeability field (shown in Figure 9) on a 24×24 grid with $H/h = 4$ and $k_2 = 0.1$. An estimation from the eigenstructure diagram in the top-right of Figure 12 indicates that a single Jacobi smoothing step pre- and post-upscaling in a V-cycle will ensure convergence. Indeed this is the case: the upscaling scheme has a convergence factor of about 0.6 per V-cycle step. On the other hand, multigrid has a factor of about 0.94 for the same V-cycle with a smoother on the coarse level.

The diagram on the left in the bottom row shows the convergence history for the Arcoperm permeability field shown in Figure 13 for $H/h = 5$. An estimation from the eigenstructure diagram in the top-right of Figure 14 indicates that a single Jacobi smoothing step pre- and post-upscaling in a V-cycle again will ensure convergence. This is the case again; however, this time the asymptotic convergence factor is only about the same for the upscaling scheme as it is for multigrid. There is, though, a steep initial decline. This is probably attributable to the initial guess for the solution (all zeros) having an error that has large components in the direction of the coarse eigenmodes of the upscaling preconditioner (this may also explain the large jump in the first step (only) for the simple permeability field above). The "stalling" of the convergence is probably attributable to the poor performance of the preconditioner on middle eigenmodes.

Modifying the upscaling preconditioner is one fix. Another is simply increasing the number of Jacobi smoothings on the fine level from one to two. As can be seen in the left diagram of Figure 18, this has a dramatic effect. The upscaling scheme now clearly outperforms multigrid. It would seem that over-estimating the number of necessary fine smoothings has a beneficial effect (and is not very computationally expensive).

This may also partly explain the good performance of the upscaling scheme in the last, bottom-right diagram of Figure 17. This is the convergence history of the Brent permeability field with $H/h = 5$. Estimating the number of necessary smoothings from the top-right eigenstructure di-

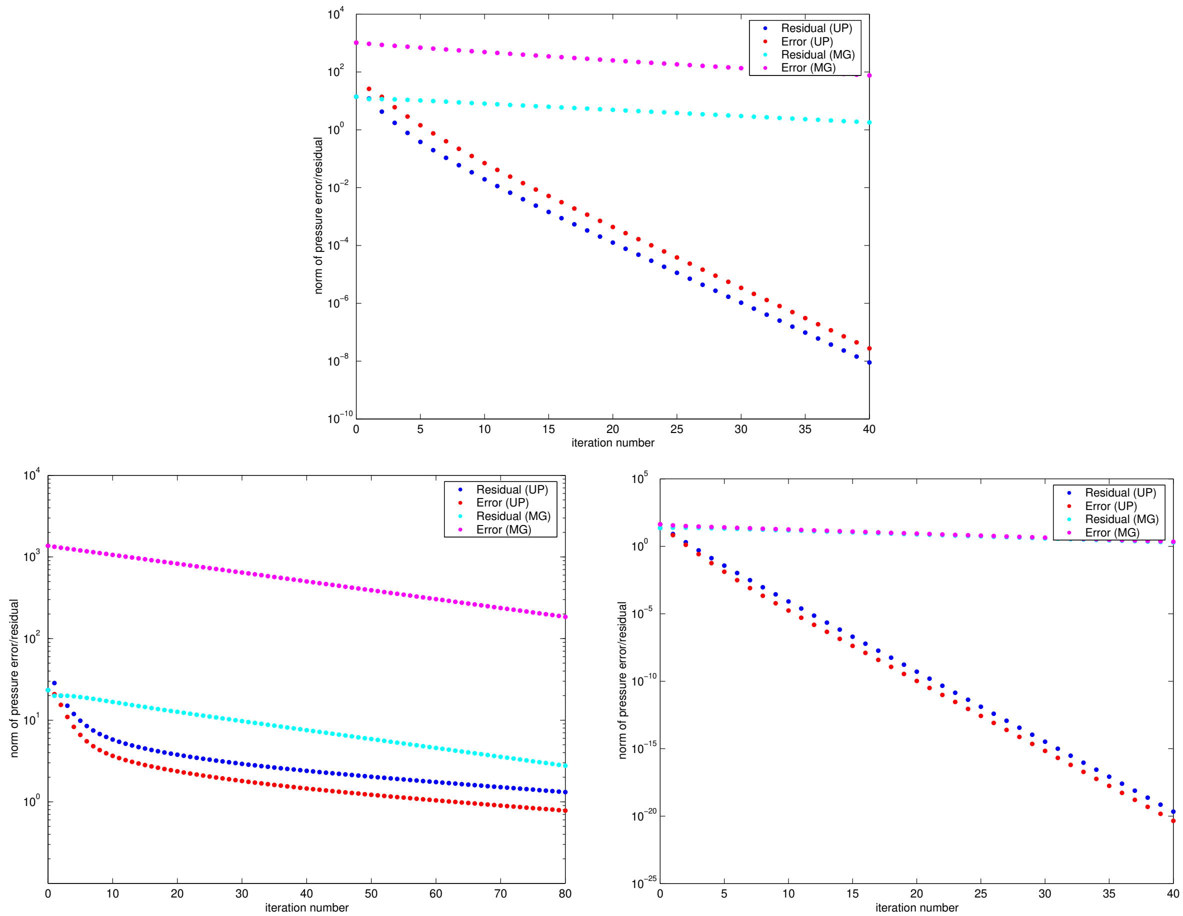

Figure 17: Convergence histories of the two-level upscaling and multigrid schemes for the three sample permeability fields in Figures 9, 13, and 15. The light and dark blue colored dots mark the sizes of the residual for the multigrid and upscaling schemes, respectively. The light and dark red dots mark the sizes of the error.

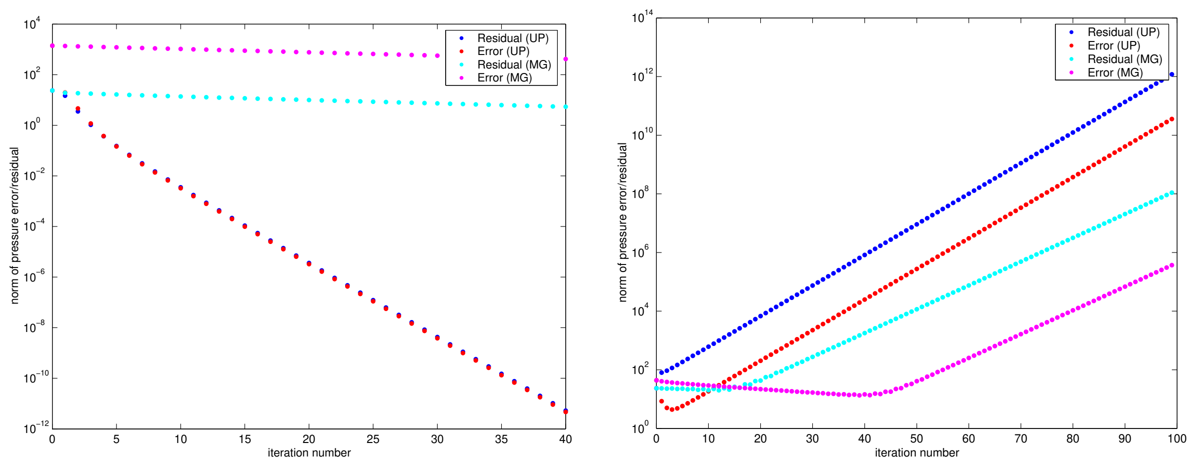

Figure 18: More convergence histories of the two-level upscaling and multigrid schemes that illustrate the effect of over- and under-estimating the number of necessary smoothings. The coloring scheme is the same as used in Figure 17. The diagram on the left is comparable to the one in the bottom-left of Figure 17; two smoothings per step are used instead of one. The diagram on the right is comparable to the one in the bottom-right of Figure 17; five smoothings per step are used instead of forty.

agram of Figure 16 is more difficult this time (note the most smoothings comes from the coarsest eigenmode — the red dot furthest to the left). Forty smoothings were used; this is most likely an over-estimate of the number of smoothings necessary for a convergent scheme. Again, this seems to have a beneficial effect for the upscaling scheme relative to multigrid.

The penalty to be paid for under-estimating the number of necessary smoothings can be seen in the right-hand diagram of Figure 18. A divergent scheme may result. (This is the "convergence" history for the Brent field with $H/h = 5$ as before, but with only one smoothing.) One last feature to note is that for all the convergent schemes, the error is monotonically decreasing (as is expected for a multigrid-like scheme).

In the future we would like to produce plots of the size of *spectrum* of the error versus iteration number. A break-down of inter-step frequency reduction (the spectrum before and after smoothing, and before and after upscaling) would also be useful. The ability to visualize a given eigenmode (that is, make an n-dimensional plot of the pressure that corresponds to an eigenvector) by clicking on the eigenstructure plot would be a handy tool.

In future research, we hope to be able to prove that our two-level scheme converges linearly (independent of h, but dependent on H/h and maybe k^*/k_*). Toward that end, the methods used in standard multigrid will certainly be useful. There are several monographs [40, 48, 63, 12, 41, 20, 56, 60, 19, 11] and review articles [66, 69, 67, 68] that provide useful information (several of the monographs on iterative methods generally). As well, Bramble, Pasciak, and Xu have developed a general framework for subspace correction methods [15, 13, 14, 66, 12, 67, 68]. Their work will be useful in having developed a unified approach to multigrid-like methods, and also specifically because they treat the possibility of non-nested spaces for corrections ($\mathbf{V}_{H,h}$ is not necessarily a subset of $\mathbf{V}_h$), and perturbed linear forms (e.g., using quadrature on the term $(\mathbf{k}\mathbf{u}_h, \mathbf{v}_h)$).

Lastly, we might try to modify the upscaling preconditioner so that it itself is a smoother. Also, the upscaling preconditioner might be modified for use in discontinous Galerkin (DG) methods, or the expanded mixed method [6].

4 Change the Problem, Not the RHS: Using nonlinear optimization techniques to adjust the basis of the mixed variational multiscale method

In constructing the subgrid upscaling, the "closure" assumption in upscaling was that all flow between coarse cells occurred on the coarse scale. Then, in approximating the coarse velocity space, no (or a limited) variation of the flow is allowed along coarse cell edges (RT0 elements have a constant flux through cell boundaries, BDDF1 elements only a linearly-varying one). Since the fine-scale flow almost always varies along coarse cell edges, our upscaled solution cannot coincide with the fine-scale solution (that is, usually $\mathbf{u}_h \notin \mathbf{V}_{H,h}$ even though $p_h \in W_{H,h} = W_h$ always for RT0 elements).

Our goal is to obtain the fine-scale solution (and to do so without entailing the work of finding it more directly). In the two-level scheme, the "defect" of coarse grained inter-coarse-cell flow was overcome by pairing the upscaler with a fine scale smoother. We now consider instead modifying the discretization of the upscaled solution so as to permit the fine-scale and upscaled solutions to coincide (that is, find a $\tilde{\mathbf{V}}_{H,h}$ that is structured like $\mathbf{V}_{H,h}$ and has $\mathbf{u}_h = \tilde{\mathbf{u}}_{H,h} \in \tilde{\mathbf{V}}_{H,h}$).

In approximating the upscaled solution, all the pressure degrees of freedom are left in and some velocity ones are discarded (as compared to the fine-scale approximation). All of the discarded degrees of freedom are along coarse cell boundaries — to recall, see the differences between Figures 1 and 3. Figure 19 shows a sample modified basis element. In it are reintroduced exactly those degrees of freedom (call them β_i's) along coarse cell boundaries that are "missing" from the basis for the upscaled approximation. This gives a parameterized family $\tilde{\mathbf{V}}_{H,h}(\boldsymbol{\beta})$ of approximation spaces of $\mathbf{V}$ that can be used in the upscaling scheme. In the decomposition $\mathbf{V}_{H,h} = \mathbf{V}_H \oplus \delta\mathbf{V}_h$ just change $\mathbf{V}_H$ to $\tilde{\mathbf{V}}_H$, leave $\delta\mathbf{V}_h$ alone, and set $\tilde{\mathbf{V}}_{H,h} = \tilde{\mathbf{V}}_H \oplus \delta\mathbf{V}_h$. With this new approximation space $\tilde{\mathbf{V}}_{H,h}$ we can solve upscaled flow problems just as before (thinking of $\boldsymbol{\beta}$ as fixed for any particular problem).

If the extra degrees of freedom $\boldsymbol{\beta}$ are set uniformly, the same old upscaled basis results; that is, $\tilde{\mathbf{V}}_{H,h}(\mathbf{1}) = \mathbf{V}_{H,h}$. If for a given problem the extra degrees of freedom are set in proportion to the fine-scale solution to that problem, the modified upscaled problem will then have the same solution as the fine-scale problem.

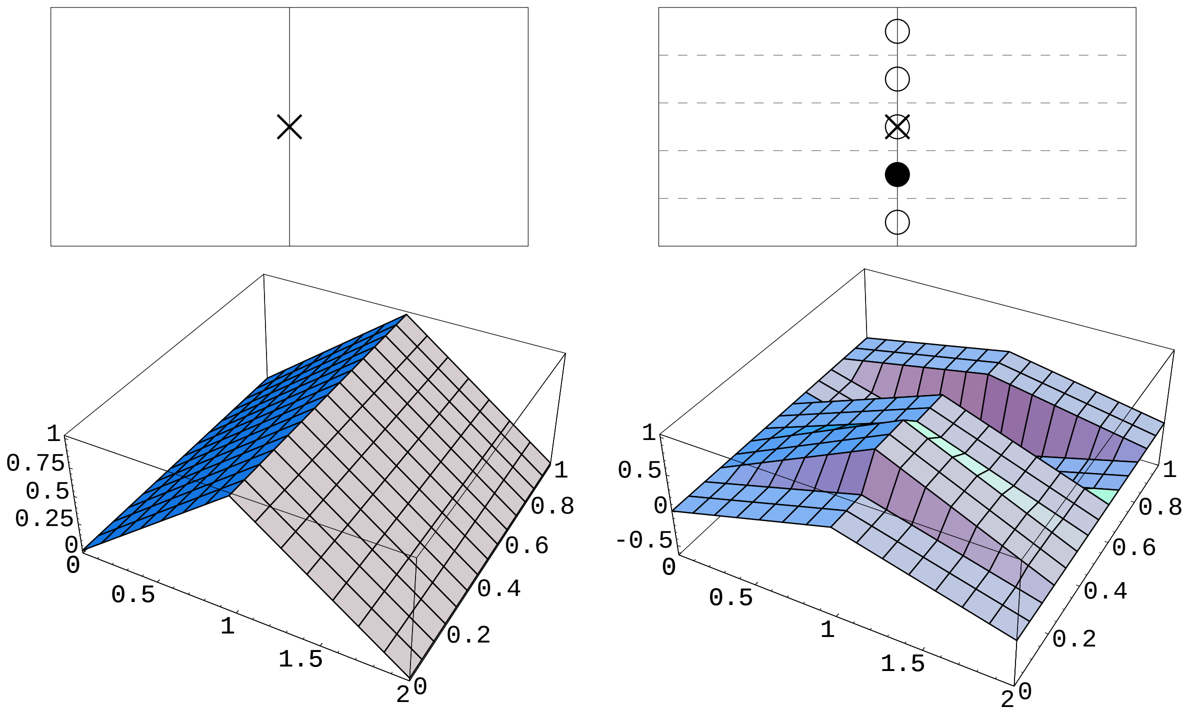

Figure 19: Unmodified and modified velocity basis elements for coarse flow. An "X" represents the coarse degree of freedom. Circles represent degrees of freedom that are adjusted in the nonlinear optimization. The filled circle represents the normalized degree of freedom. The modified basis functions may be discontinuous across the dashed lines as well as across the outer coarse cell boundaries.

This raises a question: how should the coarse basis be adjusted to give a new upscaled problem that has the fine-scale solution as its answer? If given the fine-scale solution, it could be used to apportion the extra coarse-scale velocity degrees of freedom. Of course, if the fine-scale solution were known, there would be no more work to do. However, if we have already solved some modified upscaled problem, its fine-scale residual can easily be computed and used as an error indicator. That is, prolongate the modified upscaled solution to the fine-scale space then compute the residual as usual. (With RT0 elements for the fine-scale basis and RT0/RT0 elements for the upscaled basis, prolongation is the identity operation (on functions). The modifications to the upscaled basis are conforming to the fine-scale elements.) The residual of the velocity part of the equations is used as an error indicator — where the residual is large along a coarse cell edge, the error in the approximate velocity is large — to determine how to reapportion the extra coarse-scale velocity degrees of freedom to better approximate the fine-scale solution. With the new $\boldsymbol{\beta}$ another upscaled problem is set-up, and the process repeated. Stop when the residual is zero everywhere (or "small" to our satisfaction). A diagram illustrating the steps is shown in Figure 20.

0. **Start** with $m = 0$ and $\boldsymbol{\beta}^0 = \mathbf{1}$.
1. **Solve** an upscaled problem with the spaces $W_{H,h} \ni p^m_{H,h}$ and $\tilde{\mathbf{V}}_{H,h}(\boldsymbol{\beta}^m) \ni \tilde{\mathbf{u}}^m_{H,h}$.
2. **Compute** the velocity residual $\mathbf{r}^m_h$ of $P\tilde{\mathbf{u}}^m_{H,h}$.
3. **Scale** the residual $\mathbf{s}^m_h = \omega \mathbf{D}^{-1}\mathbf{r}^m_h$ using the diagonal of the matrix $\mathbf{D}$ from the fine-scale term $(\mathbf{k}\mathbf{u}_h, \mathbf{v}_h)$.
4. **Test** if $\mathbf{s}^m_h \cdot \mathbf{r}^m_h$ is small. If so, stop.
5. **Restrict** the scaled residual $c^m = R\mathbf{s}^m_h$ to coarse cell edges.
6. **Update** parameters $\boldsymbol{\beta}^{m+1} = \boldsymbol{\beta}^m + C(c^m)$ with a function C of the corrector c^m. Increment the counter $m \leftarrow m + 1$.
7. **Repeat** the process starting at step 1 with the new $\boldsymbol{\beta}$.

Figure 20: Corrector scheme for $\boldsymbol{\beta}$. In it, solve a sequence of upscaled problems each with a different $\boldsymbol{\beta}^m$. The goal is to have $\boldsymbol{\beta}^m \to \boldsymbol{\beta}^*$. The prolongation $P : \tilde{\mathbf{V}}_{H,h} \to \mathbf{V}_h$ is simply the identity operation if RT0 elements are used throughout. The restriction $R : \mathbf{V}_h \to \Lambda_{H,h}$ gives the trace of the normal component of the velocity along coarse cell edges. The space $\Lambda_{H,h}$ is like the Lagrange multiplier space of [8] but with h-scale variation on coarse cell edges of size H. The corrector function C may have many forms (linear or nonlinear); see the discussion in the text.

There are some choices in how the coarse basis functions are updated. The simplest one perhaps is a Jacobi-like correction of the heights:

$$\boldsymbol{\beta} \leftarrow \boldsymbol{\beta} + R\omega \mathbf{D}^{-1}\mathbf{r} \tag{35}$$

where $\boldsymbol{\beta}$ is the vector of coarse basis heights, ω is a relaxation parameter, $\mathbf{D}$ is the diagonal of the (fine) Darcy matrix $(\mathbf{k}\mathbf{v}_i, \mathbf{v}_j)$ used to scale the residual properly, and $\mathbf{r}$ is the residual velocity (that is, use $C = I$ in the scheme in

Figure 20). However, there is a whole other class of updates we could use: Newton-like schemes [28]. The residual is a rational polynomial function of $\boldsymbol{\beta}$, and we seek a zero of the function $\mathbf{r}(\boldsymbol{\beta})$. Newton's method could be applied directly; this would require solving a linear system involving the Jacobian of the residual — a sparse linear system with as many unknowns as degrees of freedom in the modified coarse basis functions (roughly $2H/h$ times the number of coarse cells in 2-D, or $3(H/h)^2$ in 3-D). Alternatively, a quasi-Newton method could be used with an approximation to the Jacobian that is more easily factored (such as its diagonal or a block diagonal form defined by coarse cell edges). Also possible is Broyden's (secant) method; to initialize the approximation of the Jacobian, there is the exact Jacobian (at the starting point), the exact Jacobian at the fine-scale solution (if this proves to have a special structure), an approximation to the Jacobian, or simply the identity. Initializing with the Jacobian at the starting point would give the same first step as Newton's method, but different later steps. Initializing with the identity would give the same first step as the Jacobi-like scheme, but different later steps.

Why go through all this trouble? Newton's method has quadratic convergence. Each step in the iteration requires only solving upscaled systems (which is virtually like solving a coarse system) and performing fine-scale matrix-vector multiplications. Even Broyden's method has superlinear convergence whereas with linear problems and linear iterative solvers we expect only linear convergence. Of course, such fast convergence is only guaranteed starting from a small distance away from the solution sought. Here we are (almost) saved: initializing with uniform heights gives the ordinary upscaled solution, and we have an a-priori error analysis which says that this solution is already "close" to the fine-scale solution. It is not known whether this is always close enough to guarantee convergence. Even if the starting point of uniform heights is not good enough to guarantee convergence, some help may be provided by already known trust-region, line-search, or other global-convergence-aiming modifications to the Newton scheme. At least for sure there is only one root to find: we assume that the fine-scale problem has a unique solution; thus there is only one way to apportion the heights in the modified coarse basis (define $\boldsymbol{\beta}^*$ so that $\mathbf{u}_h \in \tilde{\mathbf{V}}_{H,h}(\boldsymbol{\beta}^*)$).

There is only one way to apportion the heights, but how the scale is chosen is left open. For the ordinary, Lagrangian elements, we have height one everywhere along the coarse edge; the solved-for unknown in the algebraic problem is the magnitude and direction. If on the other hand we have varying heights along the edge, there is some redundancy if we are allowed to specify the absolute height of each segment along with the solved-for unknown. In order to obtain a unique set of heights and resolve the redundancy, a normalization of the heights in the basis is necessary. Here again are some choices; the maximal magnitude set to one, sum of squares set to one, or some other strategy are possibilities.

The choice of normalization has some subtle effects on the analysis of the problem. When adjusting the heights using the residual flux along coarse edges, there is a correspondence between the segments along which we compute the residual and segments where we can adjust the basis. Introducing normalization complicates this. If on the one hand we choose the maximum strategy, when we compute the residual function we can just pay attention to the non-maximal degrees of freedom. Also, as we get close to the root of the residual function the normalization does not change. However, far from the root, a step in the algorithm may force a change in normalization and which components of the residual are computed. If on the other hand we choose the sum-of-squares strategy, the normalization always changes (even near the root). Further, the residual function is harder to characterize: on a given coarse edge, inputs are points on the sphere, and outputs are anywhere in space. Updates need to keep to the sphere (which is not difficult computationally, but complicates the analysis; maybe some methods of updating are not even viable). Nonetheless, it does have some advantages (see the example below, and its discussion of semi-definiteness in the Jacobian), and a certain analytic appeal.

One other note: among normalization choices, the simplest would be to pick a particular β_i to fix on each coarse edge. However, this (usually) won't work. The true solution may have zero flux on that segment of the coarse edge giving an analytic problem. Also, the true solution may have near zero flux there giving computational problems since the scale is exaggerated (problems such as poor conditioning, overflow, and cancellation errors). Other choices of normalization that exhibit such behavior would be considered poor choices; also, not normalizing during a computation (to save time) could lead to similarly bad behavior. A good choice of normalization should guarantee to not worsen the condition number (and perhaps improve it).

The use of upscaling (and not more usual finite element methods) is important here. There are otherwise too many degrees of freedom in the nonlinear part of the problem when using simple coarsening. This is true even in two dimensions with $H/h = 2$; the situation gets worse for larger

H/h and in three dimensions. Upscaling of course is not necessary, but the general intuition is that nonlinear problems get harder to solve more quickly than linear ones as the number of unknowns increases.

We digress for a moment to show an example of the ideas above, and to shed some light on normalization and its effects. Let us apply these ideas to Laplace's equation in two dimensions on the unit square. Assume there are homogeneous Neumann boundary conditions, and use a regular grid with only two coarse cells (so that there is only one coarse edge with flow). Suppose that there is a three-by-three subgrid in each coarse cell to make for a six-by-three grid of fine cells, and that RT0 elements are used (for the coarse and subgrid scales in the upscaled problem, and everywhere in the fine-scale problem). We will ignore the gravity term. The source term q was chosen to be randomly either 0 or 1 on fine cells.

The analytic form of the objective function

$$\mathbf{F} = (2/3)\mathbf{D}^{-1}\mathbf{r} \tag{36}$$

is $(14175(\beta_1+\beta_2+\beta_3)^2)^{-1}$ times

$$\begin{pmatrix} -4140\beta_2^2 - 1939\beta_2\beta_3 - 8633\beta_3^2 + \beta_1(6694\beta_2 + 10133\beta_3) \\ 2(-3347\beta_1^2 + (2445\beta_2 - 2972\beta_3)\beta_3 + \beta_1(2070\beta_2 + 3064\beta_3)) \\ -10133\beta_1^2 + 2\beta_2(-2445\beta_2 + 2972\beta_3) + \beta_1(-4189\beta_2 + 8633\beta_3) \end{pmatrix}. \tag{37}$$

(This definition is slightly different than discussed above; it includes the scaling effect of $\mathbf{D}$.) Then the root $\boldsymbol{\beta}^*$ defined by $\mathbf{F}(\boldsymbol{\beta}^*) = \mathbf{0}$ has

$$\beta_1^* = \frac{179797}{223672}\beta_3^* \qquad \text{and} \qquad \beta_2^* = \frac{290873}{447344}\beta_3^*. \tag{38}$$

This $\boldsymbol{\beta}^*$ is the same one that gives us $\mathbf{u}_h \in \tilde{\mathbf{V}}_{H,h}(\boldsymbol{\beta}^*)$. The level curves for $\beta_3 = 1$ and $F_i = 0$ are shown in Figure 21. In order to detect faux roots, some "vertical" scale is necessary. As a reference, the uniform weights $\boldsymbol{\beta} = \mathbf{1}$ are reasonable, and

$$\mathbf{F}(\mathbf{1}) = (\frac{47}{2835}, \frac{8}{405}, -\frac{103}{2835})^{\mathrm{T}}. \tag{39}$$

Level curves with F_i equal to 0 and plus or minus 10% the values in the reference are shown in Figure 22. Using this criterion to detect whether a Newton-like scheme might be fooled by values of the objective function near zero, there seem to be no faux roots.

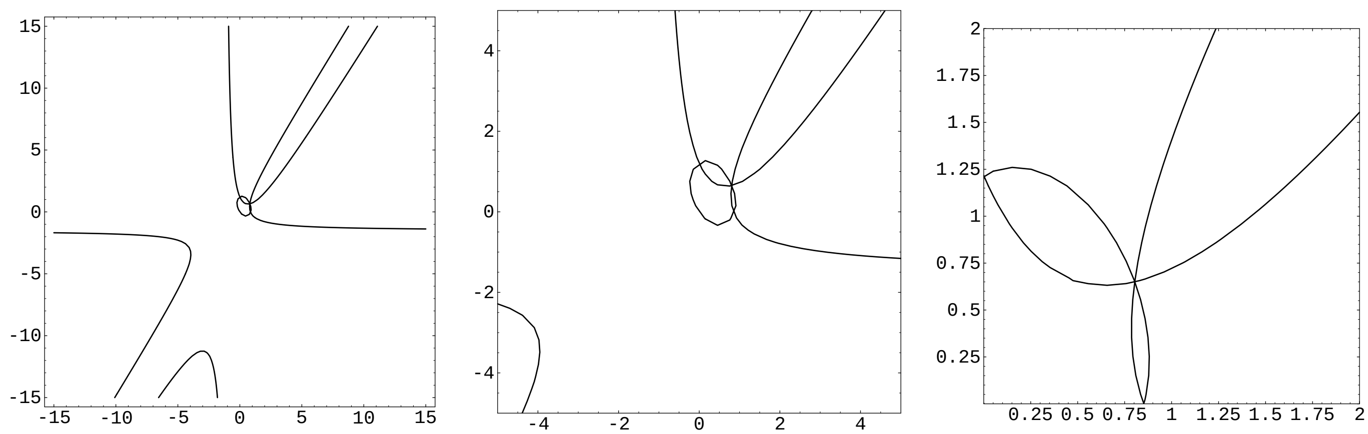

Figure 21: Level curves for each component of the objective function in (37) with $\beta_3 = 1$. Only the zero level is shown. The curves are shown at three different magnifications; at larger magnifications the hyperbolic-like curves simply extend their trends (they do not intersect). There is only one location where the curves from all three components intersect. For reference, the level curve of the third component is the circular curve. The level curve of the first component touches the top-center of the middle figure; the second touches top-center.

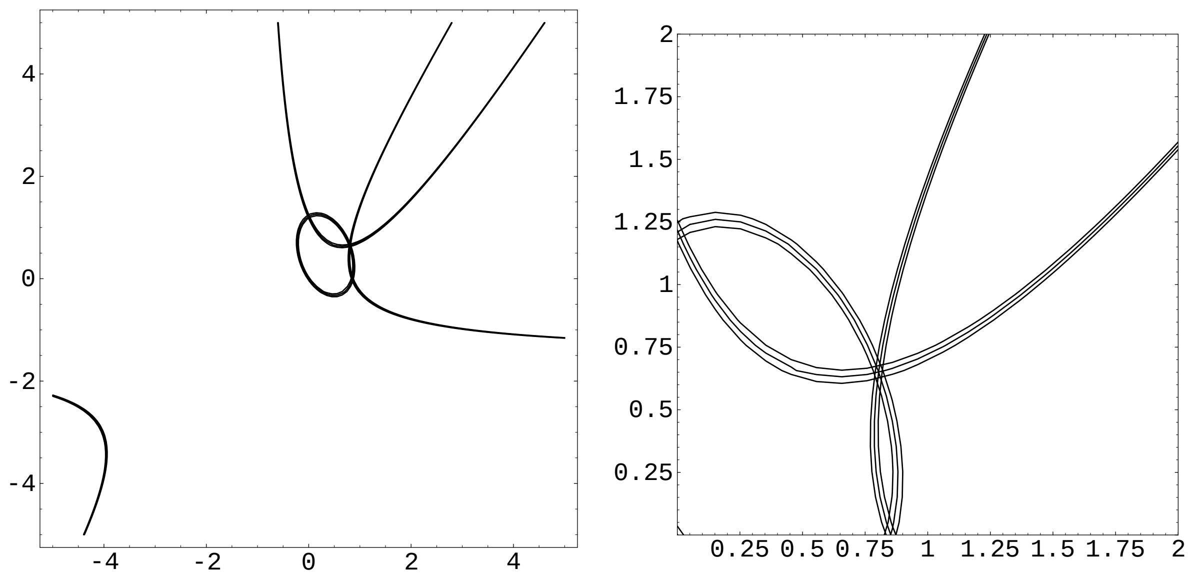

Figure 22: Level curves for each component of the objective function in (37) with $\beta_3 = 1$. Curves at the zero level as well as $\pm 10\%$ the reference level in (39) are shown.

The Jacobian of the objective function $\mathbf{F}$ from (37) is $(14175(\beta_1+\beta_2+\beta_3)^3)^{-1}$ times

$$\left(\begin{array}{l} 14974\beta_2^2 + 20705\beta_2\beta_3 + 27399\beta_3^2 - \beta_1(6694\beta_2 + 10133\beta_3) \\ 6694\beta_1^2 + \beta_3(-6341\beta_2 + 15327\beta_3) - \beta_1(14974\beta_2 + 15511\beta_3) \\ 10133\beta_1^2 + \beta_2(6341\beta_2 - 15327\beta_3) - \beta_1(5194\beta_2 + 27399\beta_3) \end{array}\right.$$
$$\begin{array}{c} 4140\beta_2^2 + 488\beta_2\beta_3 + 18016\beta_3^2 - 28\beta_1(626\beta_2 + 697\beta_3) \\ 2(8764\beta_1^2 - \beta_1(2070\beta_2 + 1613\beta_3) + \beta_3(-2445\beta_2 + 8389\beta_3)) \\ 2(9758\beta_1^2 + \beta_1(1369\beta_2 - 9008\beta_3) + \beta_2(2445\beta_2 - 8389\beta_3)) \end{array}$$
$$\left.\begin{array}{c} 5591\beta_2^2 - 7444\beta_2\beta_3 + 8633\beta_3^2 - 3\beta_1(5359\beta_2 + 9633\beta_3) \\ 16077\beta_1^2 + 4\beta_3(-3931\beta_2 + 1486\beta_3) - \beta_1(5591\beta_2 + 15511\beta_3) \\ 28899\beta_1^2 + \beta_1(22955\beta_2 - 8633\beta_3) + 4\beta_2(3931\beta_2 - 1486\beta_3) \end{array}\right). \quad (40)$$

At the "initial" value of uniform weights $\boldsymbol{\beta} = \mathbf{1}$, the Jacobian has eigenvalues of 0, $\frac{13}{135}$, and $\frac{9383}{42525}$. At the root of the objective function (38), the Jacobian has eigenvalues 0, $\sim$0.120153, and $\sim$0.272359 divided by β_3 (that is, the eigenvalues depend on the scaling/normalization scheme). The Jacobian is indefinite because we have not normalized the $\boldsymbol{\beta}$'s, but otherwise the

Jacobian is positive (semi-)definite. Whether or not the Jacobian is positive semi-definite everywhere remains unknown; for sure, the Jacobian has a zero eigenvalue everywhere.

The convergence history of the Jacobi-like scheme of (35) is shown in Figure 23. The linear rate of convergence of the scheme is apparent. The approach of the iterates towards the root (shown at right) seems to be orthogonal to the level curve of the second component. The significance of this (if any) is not known.

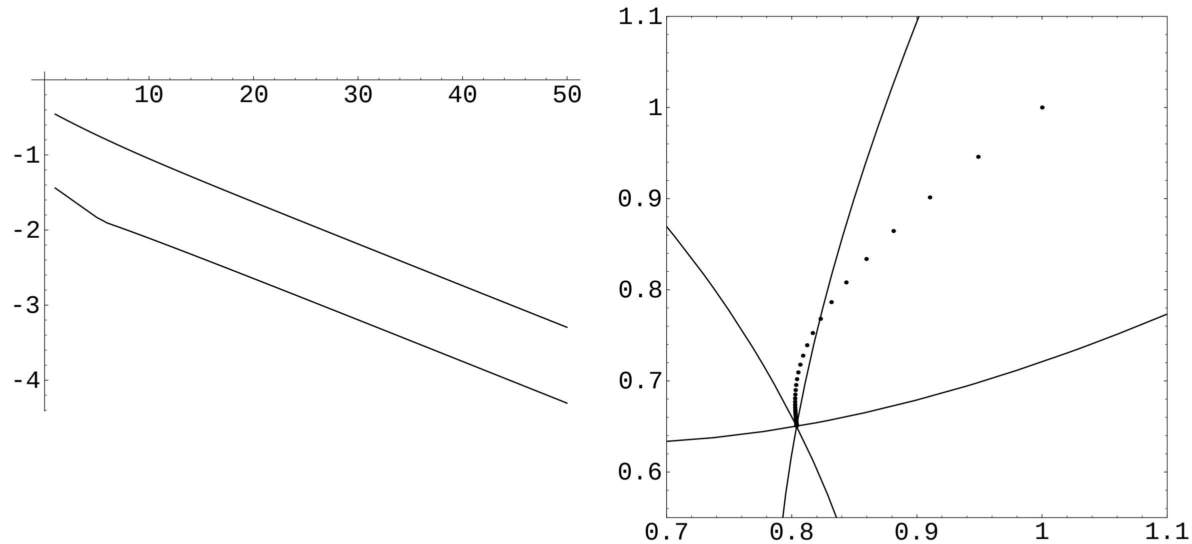

Figure 23: Convergence of the Jacobi-like scheme of (35) as applied to the sample problem. The left diagram shows the base-10 logarithms of the l_1 size of the error in $\boldsymbol{\beta}$ (top line) and values of the corresponding objective function $\mathbf{F}(\boldsymbol{\beta})$ (bottom line) versus iteration number. The right diagram plots the points (β_1, β_2) for each iteration (β_3 has been normalized to 1 — it is the maximal component for every iteration). Level curves for each component of $\mathbf{F}$ are shown for reference.

The convergence history of Newton's method is shown in Figure 24. The quadratic rate of convergence of the scheme is apparent. Note the vertical scale: after just 10 iterations $\boldsymbol{\beta}$ has a mean-square error of just 10^{-640}! In applying Newton's method, there is the problem of semi-definiteness of the Jacobian. To overcome this, the pseudo-inverse was used instead. Also, no graph of the iterates is shown (to mirror the right diagram of Figure 23) because the iterates converge too quickly. At any given magnification, only one iterate at a time is distinguishable from the root (the intersection of the zero-level curves).

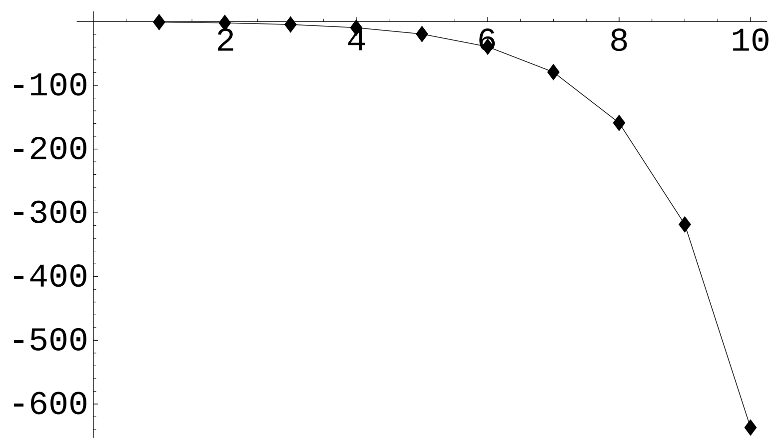

Figure 24: Convergence of Newton's method as applied to the sample problem. The diagram shows on a semilog scale the l_2 size of the error in $\boldsymbol{\beta}$ versus iteration number.

In the near future, diagrams showing the convergence of the scheme for problems with an arbitrary permeability will be possible. Problems with a gravity term will be included, too. In addition to the Jacobi-like scheme and Newton's method, a quasi-Newton using the diagonal of the Jacobian will be implemented. Broyden's method initialized the Jacobian at starting (and ending) point, the diagonal of the Jacobian at the same, and identity matrix will be implemented as well.

To summarize, the proposed research for this topic is to develop tools necessary to prove that this algorithm converges, and can be implemented in a manner that is efficient, accurate, and stable. For convergence, we need to be able to measure progress made in an iterative scheme: what is the relationship between the extra degrees of freedom in the modified coarse basis and the error in the computed upscaled solution. Along the same lines, we need to better measure the closeness of the upscaled and fine solutions (both unmodified and modified); how close is the starting point in the nonlinear optimization? We also need to know that our approximation to the Jacobian possesses bounded deterioration.

The approximation to the Jacobian is also key to the efficiency of the method: iteration counts and iteration cost are affected. Can we get away with a secant method for simplicity? And initialize it with the identity or the diagonal of the Jacobian (at the initial or final points)? If we use a quasi-Newton method, how sparse an approximation of the Jacobian can we use? Diagonal, block diagonal from coarse cell edges, or the full Jacobian? These all affect the storage required, the parallelism we can exploit, and the amount of arithmetic necessary per iteration and per simulation. Features of conjugate gradient, truncated Newton, Krylov subspace, and limited-memory methods [45] are applicable here. Further, can we exploit the analytic form of the residual to speed finding the root? Lastly, how does normalization of the extra degrees of freedom impact the performance of the algorithm as regarding global convergence (if we start close, it does not matter) and implementation? How does normalization affect the semi-definiteness of the Jacobian? In the update step (step six of Figure 20), can we update using any size corrector and normalize the result? That is, the normalized β's are not members of a vector space; so what algebra of the "+" in the update is appropriate?

Lastly, this nonlinear optimization idea might well be applied to DG and the expanded mixed method. It might also be applied to making *hp*-refinements: let newly-added degrees of freedom from a refinement be the $\boldsymbol{\beta}$'s. The solution of the unrefined problem is the initial guess for Newton's method.

5 Applications to Modeling Flow in Porous Media

Efficient, accurate Darcy flow simulations [49, 21, 46] are the prime motivation for this research. However, in order to make this research useful to practitioners, the performance of the proposed algorithms in different geostatistical settings [57, 29] needs to be investigated. That is, a guide for practitioners' use needs to be developed by running our code on a number of geostatistical realizations of permeability fields, and analyzing the performance for strengths and weaknesses as a function of geostatistical (and other physical) parameters.

Performance can be measured by how quickly and how well large-scale features of the flow can be approximated by the algorithms. What effect, if any, the geostatistics of the problem data have on this performance needs to be described. Also, if long-range correlations in the problem data present any theoretical or practical difficulties for the algorithms, this should be noted and detailed.

Performance can also be measured by how accurately the algorithms predict break-through times in two-phase simulations. In some sense, the algorithms will always be successful because they produce the same flow field as a full fine-scale simulation. However, the algorithms are iterative in nature so they introduce a truncation error on top of that of the discretization. Whether or not this has a strong effect on predicted break-through times needs to be investigated.

It would also be worthwhile to draw some engineering conclusions about the effect of long-range correlations on flow, or perhaps the particular flow features of a few specific natural formations.

Lastly, as a matter for testing, evaluation, and practical use, a well-documented implementation of the proposed algorithms in parallel for three dimensional problems is necessary. Perhaps it can be extended to handle two-phase miscible and/or immiscible displacements. It should be able to model both rate wells and Peaceman bottom-hole-pressure wells [50, 51].

6 Software

Demonstration software has been developed to solve systems using the ideas in this proposal. Additional software was developed to produce the eigenstructure diagrams for the two-level scheme and the analytic expressions for the residual in the nonlinear optimization scheme.

Although some of this software is fully parallel and can work on problems in three dimensions, not all of the software can. Also, not all the software exploits the sparsity or the special structure of the upscaling problems.

As part of the dissertation, it is proposed to further the development of the software already produced. This includes a stand-alone package to solve elliptic problems (this package could also be integrated into Parssim [2] or IPARS [61, 64]), and analysis tools for the practitioner (along the lines of the eigenstructure diagrams) to help guide the use of and parameter setting in the solver. This software should be fully parallel and be able to handle three dimensional problems. It should exploit the sparsity and special structure of the methods. Lastly, it should be modular, readable, and (self-)documented.

References

[1] T. Arbogast. Analysis of a two-scale, locally conservative subgrid upscaling for elliptic problems. *Submitted*.

[2] T. Arbogast. User's guide to Parssim1: The parallel subsurface simulator, single phase. Technical Report TICAM Report 98–13, The Center for Subsurface Modeling, Texas Institute for Computational and Applied Mathematics, The University of Texas at Austin, Austin, Texas, May 1998.

[3] T. Arbogast. Numerical subgrid upscaling of two-phase flow in porous media. In Z. Chen, R. E. Ewing, and Z.-C. Shi, editors, *Numerical treatment of multiphase flows in porous media*, volume 552 of *Lecture Notes in Physics*, pages 35–49. Springer, Berlin, 2000.

[4] T. Arbogast. Implementation of a locally conservative numerical subgrid upscaling scheme for two-phase Darcy flow. *Computational Geosciences*, 6:453–481, 2002.

[5] T. Arbogast and S. L. Bryant. A two-scale numerical subgrid technique for waterflood simulations. *SPE J.*, pages 446–457, Dec. 2002.

[6] T. Arbogast, M. F. Wheeler, and I. Yotov. Mixed finite elements for elliptic problems with tensor coefficients as cell-centered finite differences. *SIAM J. Numer. Anal.*, 34:828–852, 1997.

[7] Arcoperm permeability field data. Personal correspondence, 1997.

[8] D. N. Arnold and F. Brezzi. Mixed and nonconforming finite element methods: implementation, postprocessing and error estimates. *RAIRO Modél. Math. Anal. Numér.*, 19:7–32, 1985.

[9] J. Bear. *Dynamics of Fluids in Porous Media*. Dover, New York, 1972.

[10] A. Bensoussan, J. L. Lions, and G. Papanicolaou. *Asymptotic Analysis for Periodic Structure*. North Holland, Amsterdam, 1978.

[11] D. Braess. *Finite elements : theory, fast solvers, and applications in solid mechanics*. Cambridge University Press, New York, second edition, 2001.

[12] J. H. Bramble. *Multigrid Methods*. Number 294 in Pitman Research Notes in Mathematics. Longman Scientific and Technical, New York, 1993.

[13] J. H. Bramble, J. E. Pasciak, J. Wang, and J. Xu. Convergence estimates for multigrid algorithms without regularity assumptions. *Mathematics of Computation*, 1991.

[14] J. H. Bramble, J. E. Pasciak, J. Wang, and J. Xu. Convergence estimates for product iterative methods with applications to domain decomposition. *Mathematics of Computation*, 57:1–21, 1991.

[15] J. H. Bramble, J. E. Pasciak, and J. Xu. The analysis of multigrid algorithms with nonnested spaces or noninherited quadratic forms. *Mathematics of Computation*, 56:1–34, 1991.

[16] F. Brezzi, J. Douglas, Jr., R. Duràn, and M. Fortin. Mixed finite elements for second order elliptic problems in three variables. *Numer. Math.*, 51:237–250, 1987.

[17] F. Brezzi, J. Douglas, Jr., and L. D. Marini. Two families of mixed elements for second order elliptic problems. *Numer. Math.*, 47:217–235, 1985.

[18] F. Brezzi and M. Fortin. *Mixed and hybrid finite element methods*. Springer-Verlag, New York, 1991.

[19] W. L. Briggs, V. E. Henson, and S. F. McCormick. *A Multigrid Tutorial*. SIAM, 2000.

[20] A. M. Bruaset. *A Survey of Preconditioned Iterative Methods*. Number 328 in Pitman Research Notes in Mathematics. Longman Scientific and Technical (John Wiley), New York, 1995.

[21] G. Chavent and J. Jaffré. *Mathematical models and finite elements for reservoir simulation*. Elsevier Science Publishers, New York, 1986.

[22] Y. Chen, L. J. Durlofsky, M. Gerritsen, and X.-H. Wen. A coupled local-global upscaling approach for simulating flow in highly heterogeneous formations. *Advances in Water Resources*, 2003. Submitted.

[23] Z. Chen and T. Y. Hou. A mixed multiscale finite element method for elliptic problems with oscillating coefficients. *Math. Comp.*, 2002.

[24] L. C. Cowsar, A. Weiser, and M. F. Wheeler. Parallel multigrid and domain decomposition algorithms for elliptic equations. In D. Keyes et al., editors, *Fifth International Symposium on Domain Decomposition Methods for Partial Differential Equations*, pages 376–385. SIAM, Philadelphia, 1992.

[25] H. Darcy. *The Public Fountains of the City of Dijon*, chapter Appendix D. Victor Dalmont, Paris, 1856.

[26] B. L. Darlow, R. E. Ewing, and M. F. Wheeler. Mixed finite element methods for miscible displacement problems in porous media, SPE 10501. *Soc. Petrol. Eng. J.*, 24:391–398, 1984.

[27] J. W. Demmel. *Applied numerical linear algebra*. SIAM, Philadelphia, 1997.

[28] J. E. Dennis and R. B. Schnabel. *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*. Number 16 in Classics in Applied Mathematics. SIAM, Philadelphia, 1996.

[29] C. V. Deutsch and A. G. Journel. *GSLIB geostatistical software library and user's guide*. Oxford University Press, New York, second edition, 1997.

[30] J. Douglas, Jr., R. E. Ewing, and M. F. Wheeler. Approximation of the pressure by a mixed method in the simulation of miscible displacement. *R.A.I.R.O. Modèl. Math. Anal. Numèr.*, 17:17–33, 1983.

[31] J. Douglas, Jr. and J. E. Roberts. Global estimates for mixed methods for second order elliptic equations. *Math. Comp.*, 44:39–52, 1985.

[32] L. J. Durlofsky. Upscaling of geological models for reservoir simulation: Issues and approaches. *Computational Geosciences*, 6:1–4, 2002.

[33] Y. R. Efendiev, T. Y. Hou, and X.-H. Wu. Convergence of a nonconforming multiscale finite element method. *SIAM J. Numer. Anal.*, 37:888–910, 2000.

[34] R. E. Ewing, T. F. Russell, and M. F. Wheeler. Convergence analysis of an approximation of miscible displacement in porous media by mixed finite elements and a modified method of characteristics. *Comp. Meth. in Appl. Mech. and Engng.*, 47:73–92, 1984.

[35] C. L. Farmer. Upscaling: a review. *International Journal for Numerical Methods in Fluids*, 40(1–2):63–78, 2002.

[36] R. A. Freeze and J. A. Cherry. *Groundwater*. Prentice-Hall, Englewood Cliffs, New Jersey, 1979.

[37] J. Glimm et al. Sharp and diffuse fronts in oil reservoirs: Front tracking and capillarity. In W. E. Fitzgibbon, editor, *Mathematical and Computational Methods in Seismic Exploration and Reservoir Modeling*, pages 54–76, Philadelphia, 1985. SIAM.

[38] R. Glowinski and M. F. Wheeler. Domain decomposition and mixed finite element methods for elliptic problems. In R. Glowinski et al., editors, *First International Symposium on Domain Decomposition Methods for Partial Differential Equations*, pages 144–172. SIAM, Philadelphia, 1988.

[39] G. H. Golub and C. F. Van Loan. *Matrix Computations*. The Johns Hopkins University Press, Baltimore, third edition, 1996.

[40] W. Hackbusch. *Multi-grid methods and applications*. Number 4 in Computational mathematics. Springer-Verlag, New York, 1985.

[41] W. Hackbusch. *Iterative Solution of Large Sparse Systems of Equations*. Number 95 in Applied Mathematical Sciences. Springer-Verlag, New York, 1993.

[42] L. Holden and B. F. Nielsen. Global upscaling of permeability in heterogeneous reservoirs; the output least squares (OLS) method. *Transport Porous Media*, 40:115–143, 2000.

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

[44] T. J. R. Hughes, G. R. Feijóo, L. Mazzei, and J.-B. Quincy. The variational multiscale method—a paradigm for computational mechanics. *Comp. Meth. in Appl. Mech. and Engng.*, 166:3–24, 1998.

[45] C. T. Kelley. *Iterative methods for linear and nonlinear equations.* SIAM, Philadelphia, 1995.

[46] L. W. Lake. *Enhanced Oil Recovery.* Prentice Hall, Englewood Cliffs, New Jersey, 1989.

[47] B. Lindquist. Geostatistically generated permeability field data. Personal correspondence, October 1992.

[48] S. F. McCormick, editor. *Multigrid methods.* SIAM, 1987.

[49] D. W. Peaceman. *Fundamentals of numerical reservoir simulation.* Elsevier, Amsterdam, 1977.

[50] D. W. Peaceman. Interpretation of well-block pressures in numerical reservoir simulation. *Society of Petroleum Engineers Journal*, pages 183–194, June 1978.

[51] D. W. Peaceman. Interpretation of well-block pressures in numerical reservoir simulation with nonsquare grid blocks and anisotropic permeability. *Society of Petroleum Engineers Journal*, pages 531–543, June 1983.

[52] R. A. Raviart and J. M. Thomas. A mixed finite element method for 2nd order elliptic problems. In *Mathematical Aspects of Finite Element Methods*, number 606 in Lecture Notes in Mathematics, pages 292–315. Springer-Verlag, New York, 1977.

[53] P. Renard and G. de Marsily. Calculating equivalent permeability: a review. *Advances in Water Resources*, 20:253–278, 1997.

[54] J. E. Roberts and J.-M. Thomas. Mixed and hybrid methods. In P. G. Ciarlet and J. L. Lions, editors, *Handbook of Numerical Analysis*, volume 2, pages 523–639. Elsevier Science Publishers B.V. (North-Holland), Amsterdam, 1991.

[55] T. F. Russell and M. F. Wheeler. Finite element and finite difference methods for continuous flows in porous media. In R. E. Ewing, editor, *The Mathematics of Reservoir Simulation*, number 1 in Frontiers in Applied Mathematics, pages 35–106, Chapter II. Society for Industrial and Applied Mathematics, Philadelphia, 1983.

[56] Y. Saad. *Iterative Methods for Sparse Linear Systems*. Brooks/Cole, 1995.

[57] M. Sahimi. *Flow and Transport in Porous Media and Fractured Rock: From classical methods to modern approaches*. VCH, New York, 1995.

[58] E. Sanchez-Palencia. *Non-homogeneous Media and Vibration Theory*. Number 127 in Lecture Notes in Physics. Springer-Verlag, New York, 1980.

[59] A. E. Scheidegger. *The Physics of Flow Through Porous Media, 3*rd *ed.* University of Toronto Press, Toronto, 1974.

[60] V. V. Shaidurov. *Multigrid methods for finite elements*, volume 318 of *Mathematics and its applications*. Kluwer Academic Publishers, Boston, 1995.

[61] P. Wang, I. Yotov, M. Wheeler, T. Arbogast, C. Dawson, M. Parashar, and K. Sepehrnoori. A new generation EOS compositional reservoir simulator: Part I—Formulation and discretization. In *Proceedings of the 14th SPE Symposium on Reservoir Simulation held in Dallas, Texas*, June 8–11, 1997. SPE 37979.

[62] X.-H. Wen and J. J. Gómez-Hernández. Upscaling hydraulic conductivities in heterogeneous media: An overview. *Journal of Hydrology*, 1996.

[63] P. Wesseling. *An Introduction to Multigrid Methods*. John Wiley, New York, 1992.

[64] M. Wheeler, T. Arbogast, S. Bryant, J. Eaton, Q. Lu, M. Peszynska, and I. Yotov. A parallel multiblock/multidomain approach for reservoir simulation. In *Proceedings of the 15th SPE Symposium on Reservoir Simulation held in Houston, Texas*, February 14–17, 1999. SPE 51884.

[65] X. H. Wu, Y. Efendiev, and T. Y. Hou. Analysis of upscaling absolute permeability. *Discrete Contin. Dyn. Syst. Ser. B*, 2:185–204, 2002.

[66] J. Xu. Iterative methods by space decomposition and subspace correction: A unifying approach. *SIAM Review*, 34:581–613, 1992.

[67] J. Xu. An introduction to multigrid convergence theory. In R. H. Chan, T. F. Chan, and G. H. Golub, editors, *Iterative methods in scientific computing*, Singapore, 1997. Springer.

[68] J. Xu. An introduction to multilevel methods. In M. Ainsworth, J. Levesley, M. Marletta, and W. A. Light, editors, *Wavelets, Multilevel Methods and Elliptic PDEs*, Numerical Mathematics and Scientific Computation (Oxford science publications), pages 213–302, Oxford, 1997. Clarendon Press.

[69] H. Yserentant. Old and new convergence proofs for multigrid methods. *Acta Numerica*, 3:285–326, 1993.