Adaptive Variational Multiscale Methods

Axel Målqvist

`axel@math.colostate.edu`

Department of Mathematics, Colorado State University

Main results of the thesis

- A new adaptive variational multiscale method based on energy norm error estimation.

- An extension of the adaptive variational multiscale method to a mixed setting.

- Framework for adaptivity in multi-physics.

The new multiscale method

Model problem: The Poisson equations with coefficient $a > 0$,

$$-\nabla \cdot a\nabla u = f \quad \text{in } \Omega,$$
$$u = 0 \quad \text{on } \Gamma.$$

Weak form: Find $u \in V = H_0^1(\Omega)$ such that,

$$a(u, v) = l(v) \quad \text{for all } v \in H_0^1(\Omega),$$

where $a(v, w) = \int_\Omega a\nabla v \cdot \nabla w \, dx$, $l(v) = \int_\Omega fv \, dx$, $f \in L^2(\Omega)$ and Ω is a domain in $\mathbf{R}^d$, $d = 1, 2, 3$.
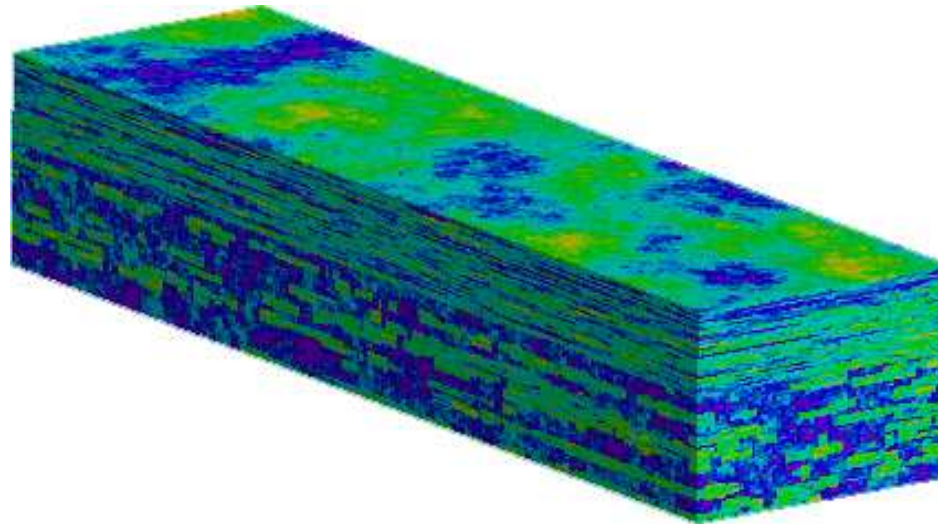
The permeability a

Figure 1: The permeability a (in log scale).

Why multiscale method?

- If we for the moment assume a to be periodic $a = a(x/\epsilon)$ we have (Hou),

$$\|\nabla u - \nabla U\| \leq C\frac{H}{\epsilon}\|f\|.$$

- $H > \epsilon$ will give unreliable results even with exact quadrature.

- $H < \epsilon$ will be too computationally expensive to solve on a single mesh.

- Parallelized local problems must be solved.

The variational multiscale method

Find $u_c \in V_c$ and $u_f \in V_f$, $V_c \oplus V_f = V$ such that,

$$a(u_c + u_f, v_c + v_f) = l(v_c + v_f),$$

for all $v_c \in V_c$ and $v_f \in V_f$.

$$a(u_c, v_c) + a(u_f, v_c) = l(v_c) \quad \text{for all } v_c \in V_c,$$
$$a(u_f, v_f) = (R(u_c), v_f) \quad \text{for all } v_f \in V_f.$$

where we introduce the residual distribution,
$(R(u_c), v_f) = l(v_f) - a(u_c, v_f).$
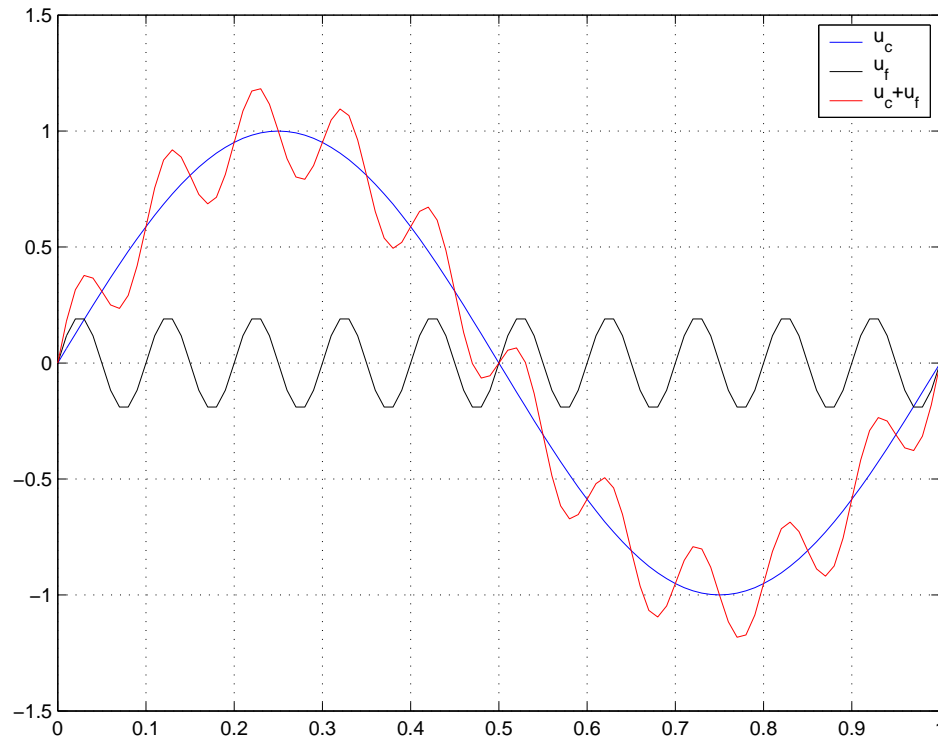
The variational multiscale method

Figure 2: u_c, u_f, and $u_c + u_f$.
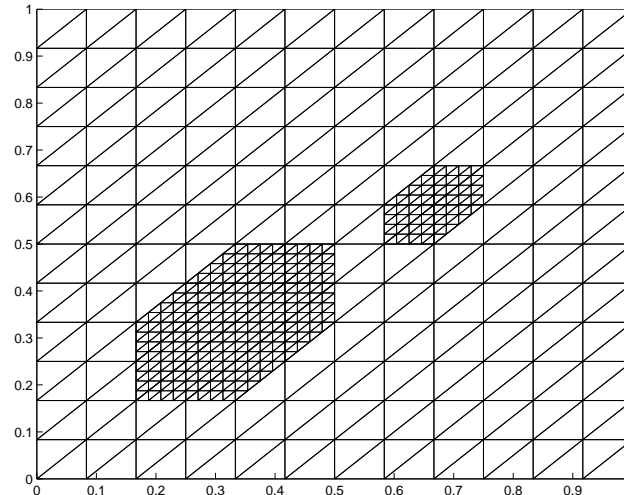
Approximation

We derive the method in two steps.

- We decouple the fine scale equations by introducing a partition of unity $\sum_{i \in \mathcal{N}} \varphi_i = 1$,

$$a(u_{f,i}, v_f) = (R(u_c), \varphi_i v_f) \quad \text{for all } v_f \in V_f.$$

- For each $i \in \mathcal{N}$ we discretize V_f and solve the resulting problem on a patch ω_i rather then Ω,

$$a(U_{f,i}, v_f) = (R(U_c), \varphi_i v_f) \quad \text{for all } v_f \in V_f^h(\omega_i).$$

The patch ω_i

To the right we see a mesh star to the left what we call a two layer mesh star. The coarse mesh size is denoted H and the fine mesh size is denoted h.
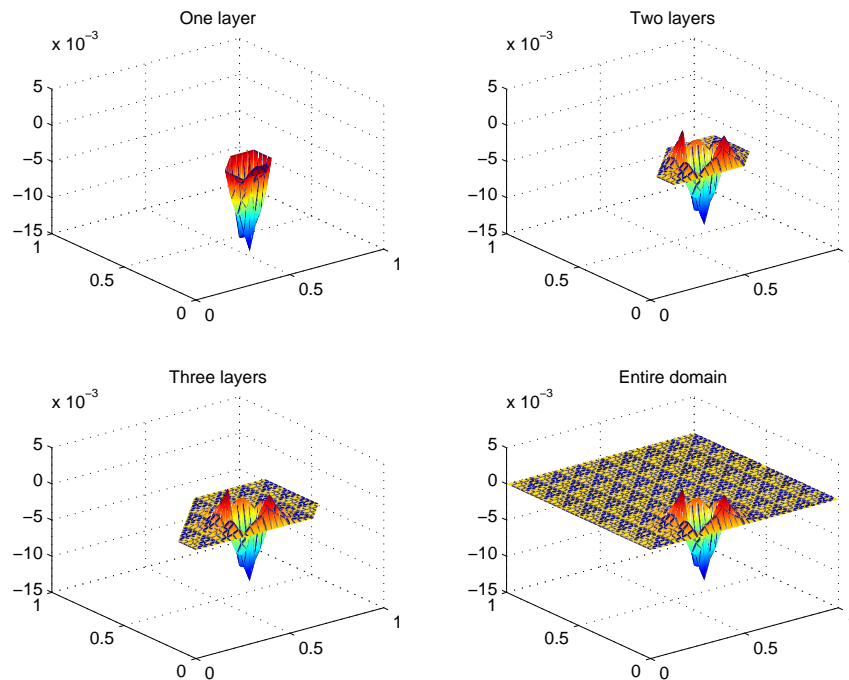
The new vms

The resulting method reads: find $U_c \in V_c$ and $U_f = \sum_{i \in \mathcal{N}} U_{f,i}$ where $U_{f,i} \in V_f^h(\omega_i)$ such that

$$a(U_c, v_c) + a(U_f, v_c) = l(v_c),$$
$$a(U_{f,i}, v_f) = (R(U_c), \varphi_i v_f),$$

for all $v_c \in V_c$, $v_f \in V_f^h(\omega_i)$, and $i \in \mathcal{N}$.

The patch is chosen such that $\text{supp}(\varphi_i) \subset \omega_i \subset \Omega$.

The local solution $U_{f,i}$

The solution improves as the patch size increases.

Motivation of the method

Why do we expect the method to work?

- The right hand side of the fine scale equations has support on a coarse mesh star, $(R(U_c), \varphi_i v_f)$.

- The fine scale solution $U_{f,i} \in V_f^h(\omega_i)$ which is a slice space.

This makes $U_{f,i}$ decay rapidly, which makes it possible to get a good approximation using small patches.

How do we choose the patchsize and h?

Our aim is to create a method that tunes critical parameters by itself.

- A posteriori error estimation bounds the error from above in terms of known quantities.

- Based on this we formulate an adaptive algorithm.

- The algorithm tunes the critical parameters automatically.

Energy norm estimate, $\|e\|_a^2 = a(e, e)$

$$\|e\|_a \leq \sum_{i \in \mathcal{C}} C_i \|H\mathcal{R}(U_c)\|_{\omega_i}$$

$$+ \sum_{i \in \mathcal{F}} C_i \left(\|\sqrt{H}\Sigma(U_{f,i})\|_{\partial \omega_i} + \|h\mathcal{R}_i(U_{f,i})\|_{\omega_i} \right)$$

- The first term is the coarse scale mesh error.
- The second term is the error committed by restriction to patches $\Sigma(U_{f,i}) \approx \boldsymbol{n} \cdot a\nabla U_{f,i}$.
- The third term is the fine scale mesh error.

The local solution $U_{f,i}$
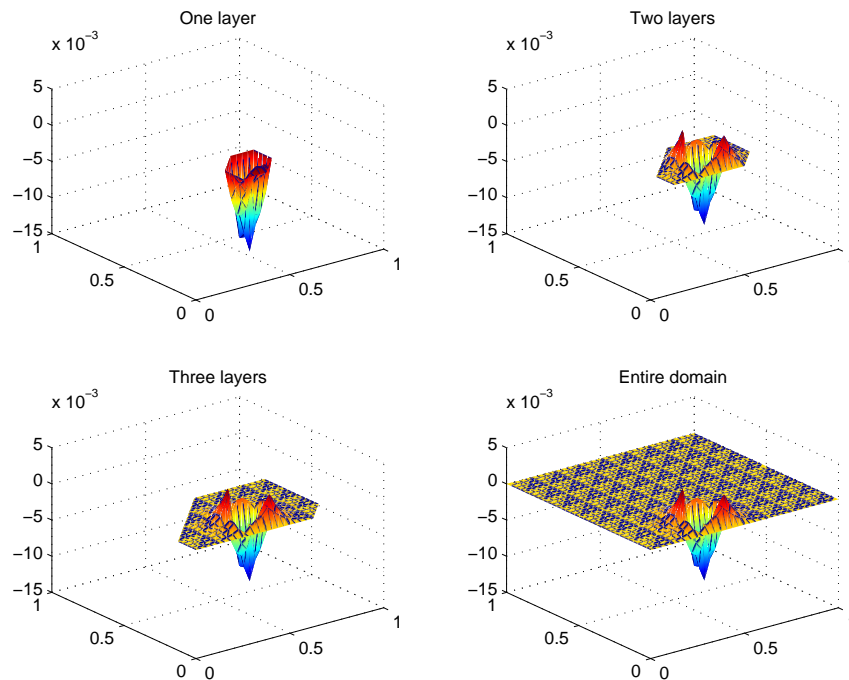
The term $\boldsymbol{n} \cdot a \nabla U_{f,i}$ decreases on the boundary $\partial \omega_i$ as the patch size increases.

Adaptive strategy

$$\|e\|_a \leq \sum_{i \in \mathcal{C}} C_i \|H\mathcal{R}(U_c)\|_{\omega_i}$$

$$+ \sum_{i \in \mathcal{F}} C_i \left(\|\sqrt{H}\Sigma(U_{f,i})\|_{\partial\omega_i} + \|h\mathcal{R}_i(U_{f,i})\|_{\omega_i} \right)$$

- We calculate these for each $i \in \{\text{coarse fine}\}$.
- Large values $i \in$ coarse $\rightarrow$ more local problems.
- Large values $i \in$ fine $\rightarrow$ more layers or smaller h.

Extension to a mixed setting

In Paper IV we have extended this theory to the mixed formulation of the Poisson equation,

$$\begin{cases} \frac{1}{a}\boldsymbol{\sigma} - \nabla u = 0 & \text{in } \Omega, \\ -\nabla \cdot \boldsymbol{\sigma} = f & \text{in } \Omega, \\ n \cdot \boldsymbol{\sigma} = 0 & \text{on } \Gamma. \end{cases}$$

We give error estimates in energy norm $\|\frac{1}{\sqrt{a}}(\boldsymbol{\sigma} - \boldsymbol{\Sigma})\|$ and for a linear functional $(\boldsymbol{\sigma} - \boldsymbol{\Sigma}, \omega)$.

Numerical examples
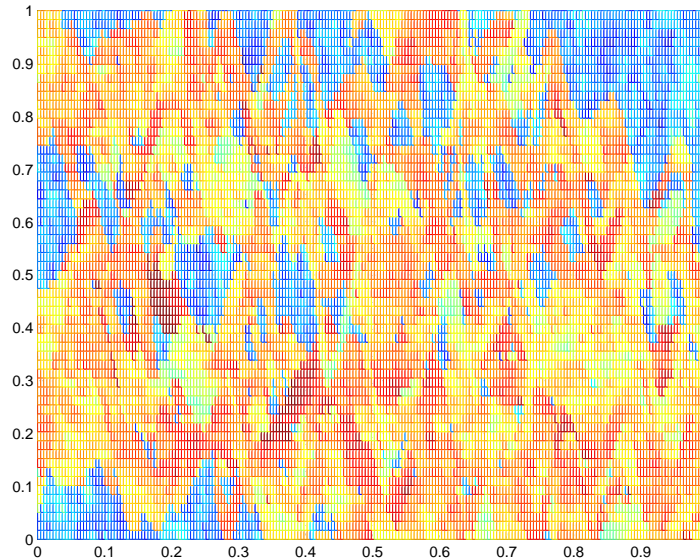
Figure 3: 2D slice of the permeability a (in log scale) taken from the tenth SPE comparative solution project.

Reference solutions
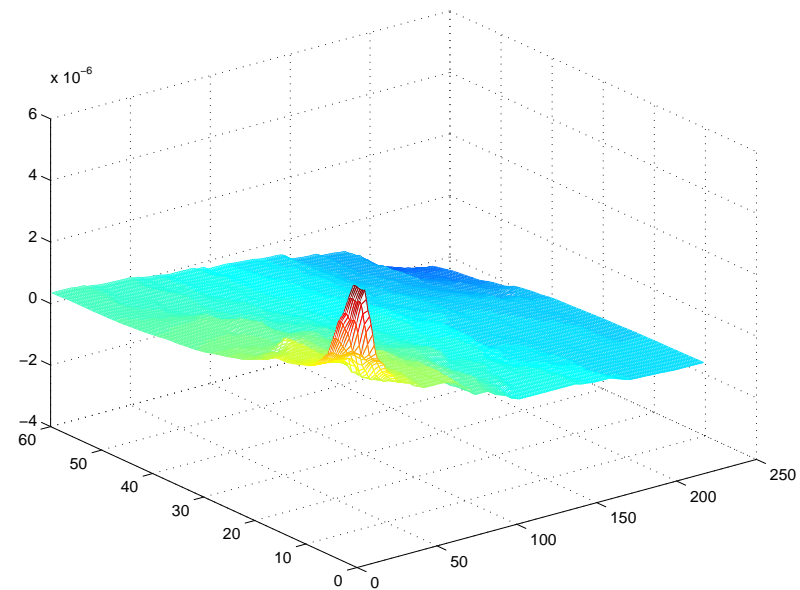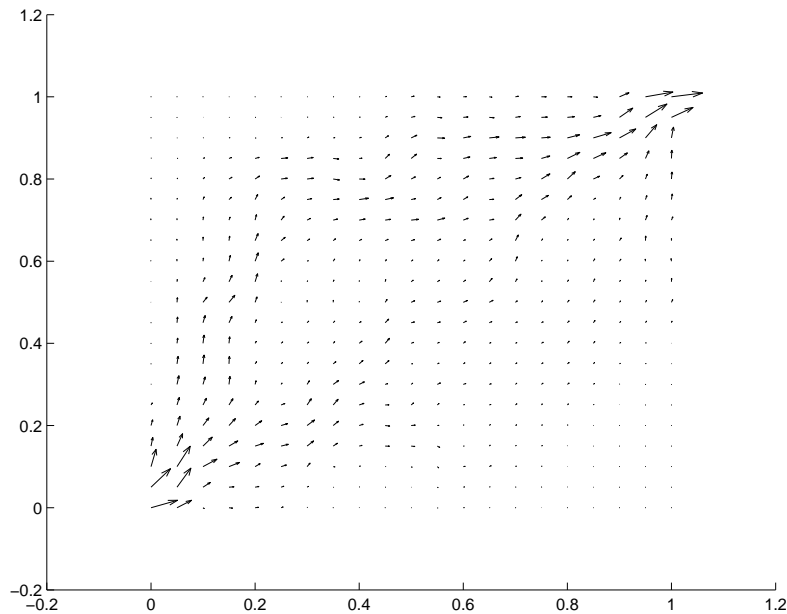
Figure 4: Above we see the reference solution, (left) flux $-\Sigma$ and (right) pressure u.
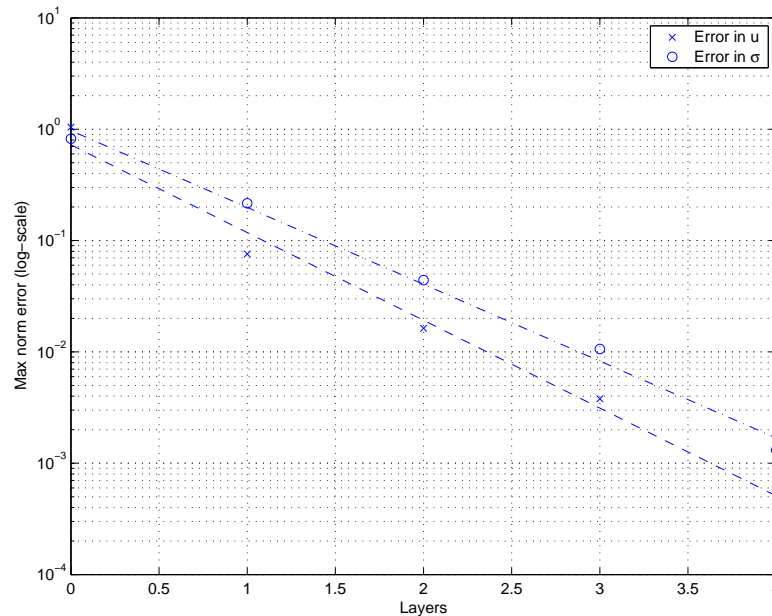
Convergence

Figure 5: Max norm error (compared to reference solution) in log scale versus number of layers.
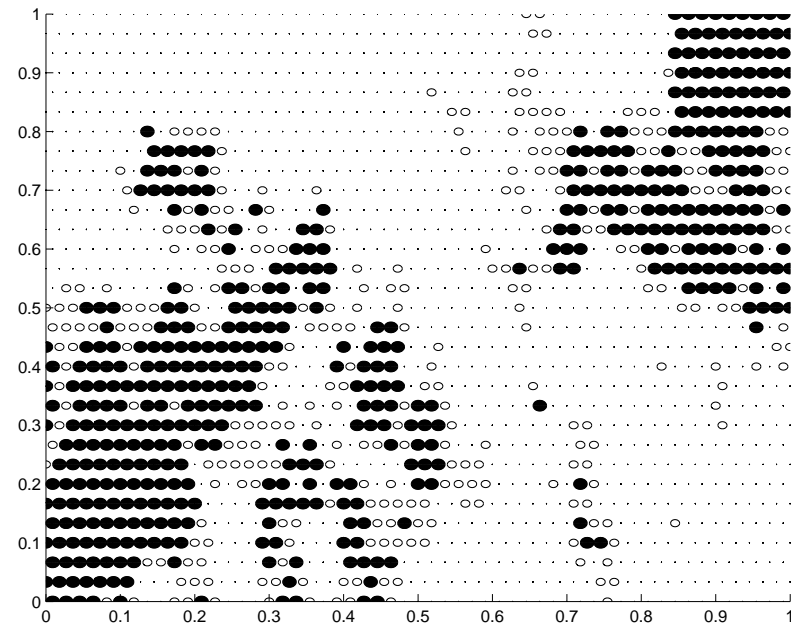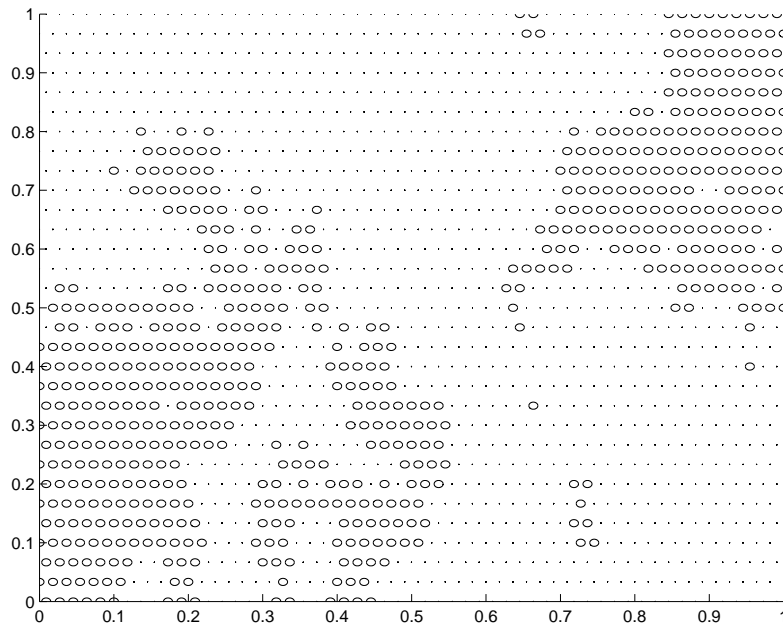
Example using the adaptive algorithm

Figure 6: 35% of the patches increased in each iteration.
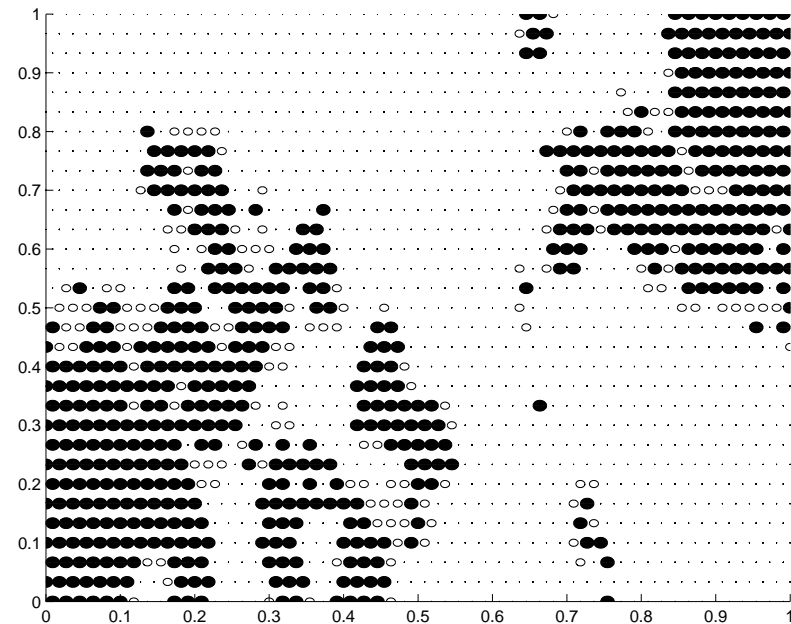
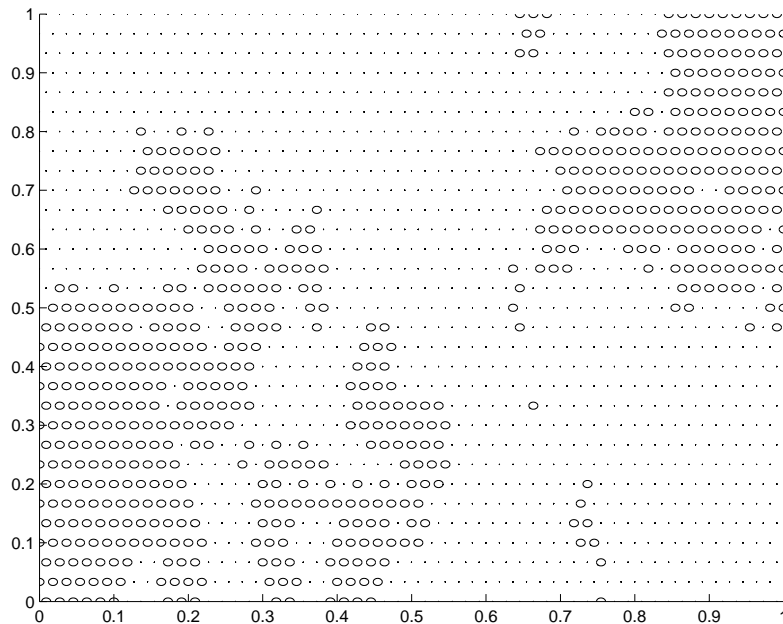# Example using the adaptive algorithm

Figure 7: 35% of the fine scale meshes refined in each iteration.
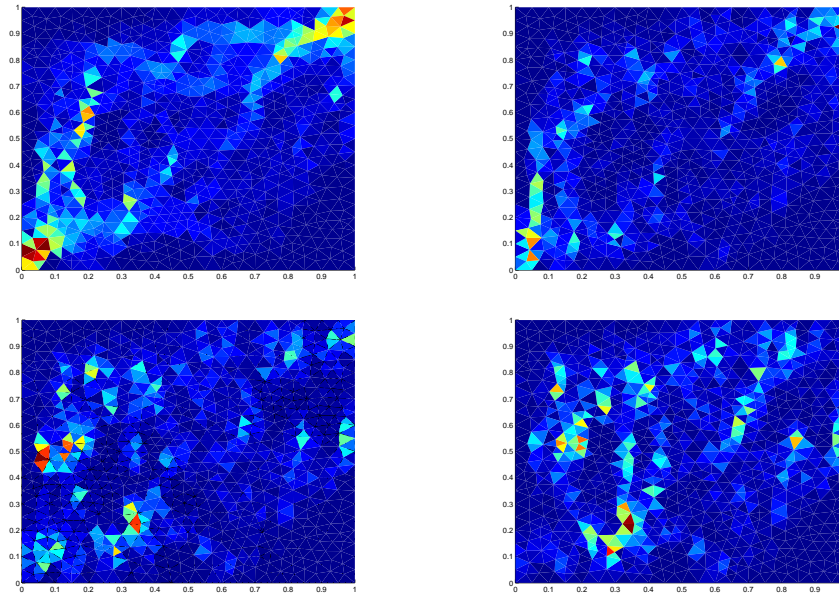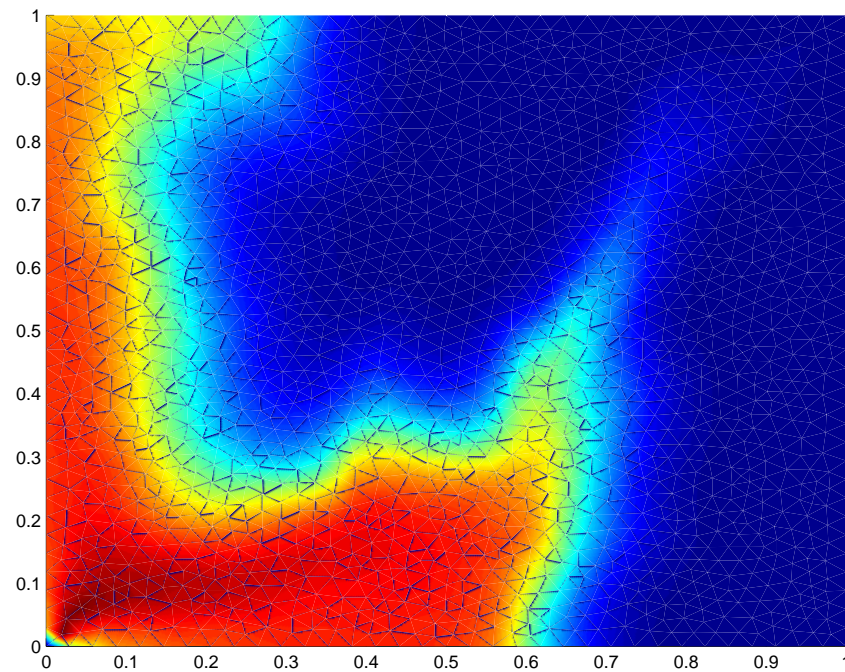
Relative error in energy norm

Figure 8: Relative error in energy norm: 106%, 16%, 10%, and 8%.

Adaptivity in multi-physics

Adaptivity in multi-physics

We seek the water concentration c that solves the system,

$$\begin{cases} \frac{1}{a}\boldsymbol{\sigma} - \nabla u = 0 & \text{in } \Omega, \\ -\nabla \cdot \boldsymbol{\sigma} = f & \text{in } \Omega, \\ \boldsymbol{n} \cdot \boldsymbol{\sigma} = 0 & \text{on } \Gamma, \end{cases}$$

$$\begin{cases} \dot{c} + \nabla \cdot (\boldsymbol{\sigma} c) - \epsilon \triangle c = g & \text{in } \Omega \times (0, T], \\ \boldsymbol{n} \cdot \nabla c = 0 & \text{on } \Gamma, \\ c = c_0 & \text{for } t = 0. \end{cases}$$
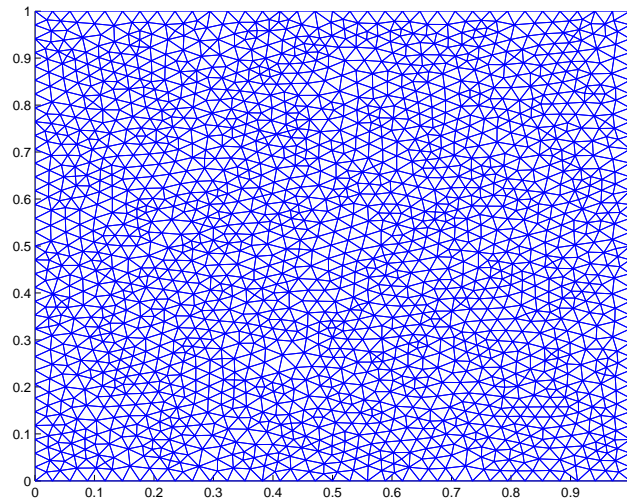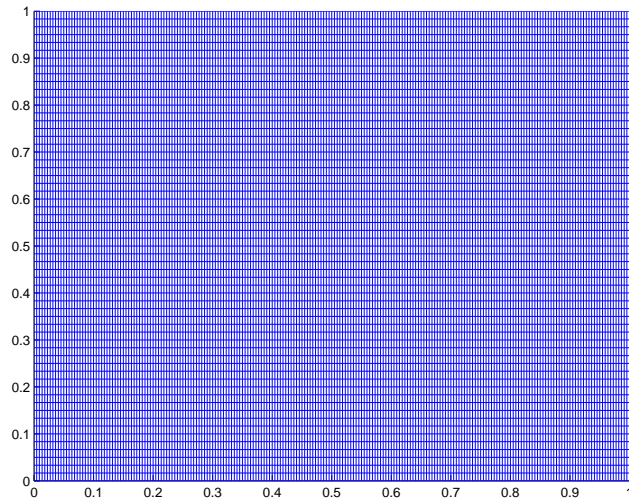
Meshes

Figure 9: The mesh for the flow problem is denoted $\mathcal{Q}$ and the transport problem $\mathcal{K}$.
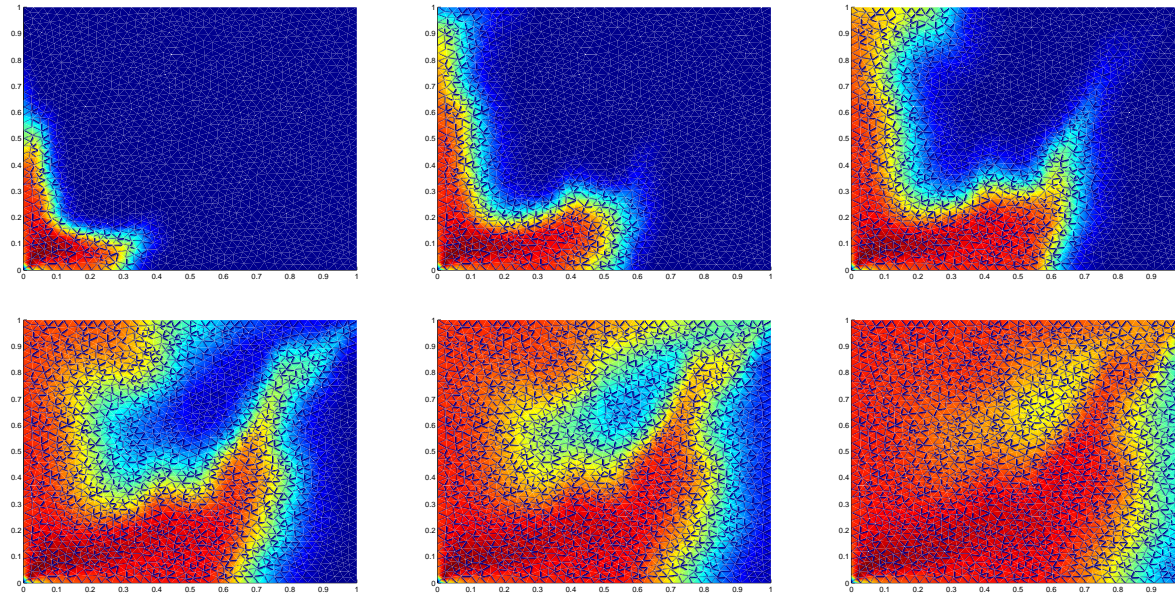
Adaptivity in multi-physics

Figure 10: The solution to the transport problem.

Error estimates based on duality

We introduce the dual problems:

$$\begin{cases} -\dot{\phi} - \boldsymbol{\Sigma} \cdot \nabla \phi - \epsilon \triangle \phi = \psi & \text{in } \Omega \times (0, T], \\ \boldsymbol{n} \cdot \nabla \phi = 0 & \text{on } \Gamma, \\ \phi = 0 & \text{for } t = T, \end{cases}$$

$$\begin{cases} \frac{1}{a}\boldsymbol{\chi} - \nabla \eta = \int_0^T c \nabla \phi \, dx & \text{in } \Omega, \\ -\nabla \cdot \boldsymbol{\chi} = 0 & \text{in } \Omega, \\ \boldsymbol{n} \cdot \boldsymbol{\chi} = 0 & \text{on } \Gamma, \end{cases}$$

Error estimates based on duality

$$\int_0^T (e, \psi)\, dt \leq \sum_{K \in \mathcal{K}} \int_0^T \rho_K(C)(\triangle t \|\dot{\phi}\|_K + h\|\nabla\phi\|_K)\, dt$$

$$+ \sum_{K \in \mathcal{Q}} (\|\nabla U^* - \frac{1}{a}\boldsymbol{\Sigma}\|_K + h^{-1/2}\|[U^*]\|_{\partial K \backslash \Gamma})\|\boldsymbol{\chi}\|_K$$

$$+ \sum_{K \in \mathcal{Q}} h\|\nabla \cdot \boldsymbol{\Sigma} + f\|_K \|\nabla\eta\|_K,$$

$$\rho_K(C) = \|\dot{C} + \nabla \cdot (\boldsymbol{\Sigma}C) - \epsilon\triangle C - g\|_K + h^{-1/2}\|\epsilon[\boldsymbol{n}\cdot\nabla C]\|_{\partial K}.$$

Adaptive algorithm

- Calculate the solutions Σ and U to the flow problem on $\mathcal{Q}$.

- Calculate the solution to the transport problem C on $\mathcal{K}$.

- Calculate an approximate solution to the dual transport problem Φ, given ψ, on $\mathcal{K}$.

- Calculate an approximation to $\int_0^T c\nabla\phi\,dx$ from C and Φ.

- Calculate the approximate solutions to the dual flow problem χ and η on the mesh $\mathcal{Q}$.

Adaptive algorithm
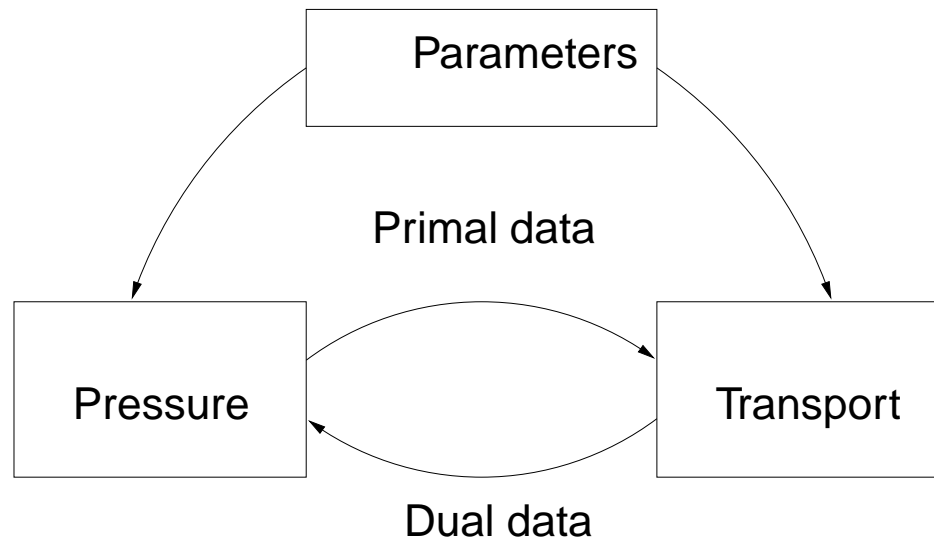

Figure 11: Information flow between solvers and data base.

Adaptive algorithm

- Calculate error indicators,

$$
\begin{cases}
I_1^K = \int_0^T \rho_K(C)(\triangle t \|\dot{\phi}\|_K + h\|\nabla\phi\|_K)\, dt, \\
I_2^K = (\|\nabla U^* - \frac{1}{a}\boldsymbol{\Sigma}\|_K + h^{-1/2}\|[U^*]\|_{\partial K \backslash \Gamma})\|\boldsymbol{\chi}\|_K \\
\qquad + h\|\nabla \cdot \boldsymbol{\Sigma} + f\|_K \|\nabla\eta\|_K.
\end{cases}
$$

- If $I_1 = \sum_{K \in \mathcal{K}} I_1^K$ and $I_2 = \sum_{K \in \mathcal{Q}} I_2^K$ ok stop.

- If $I_1 \gg I_2$ refine $\mathcal{K}$, $I_2 \gg I_1$ refine $\mathcal{Q}$, return.

- If non of these hold we refine both $\mathcal{K}$ and $\mathcal{Q}$ and return.
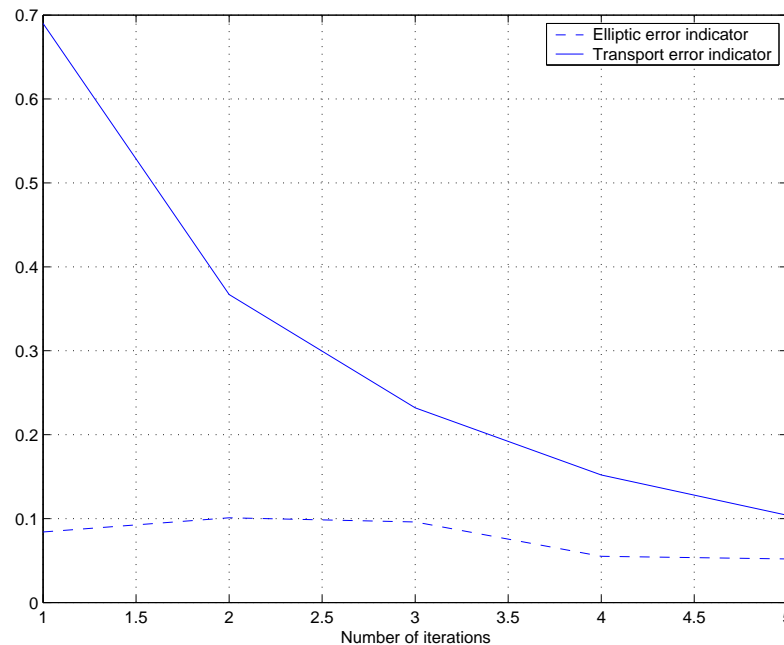
Error indicators

Figure 12: The error indicators I_1 and I_2. We use 15 and 100% refinement level.
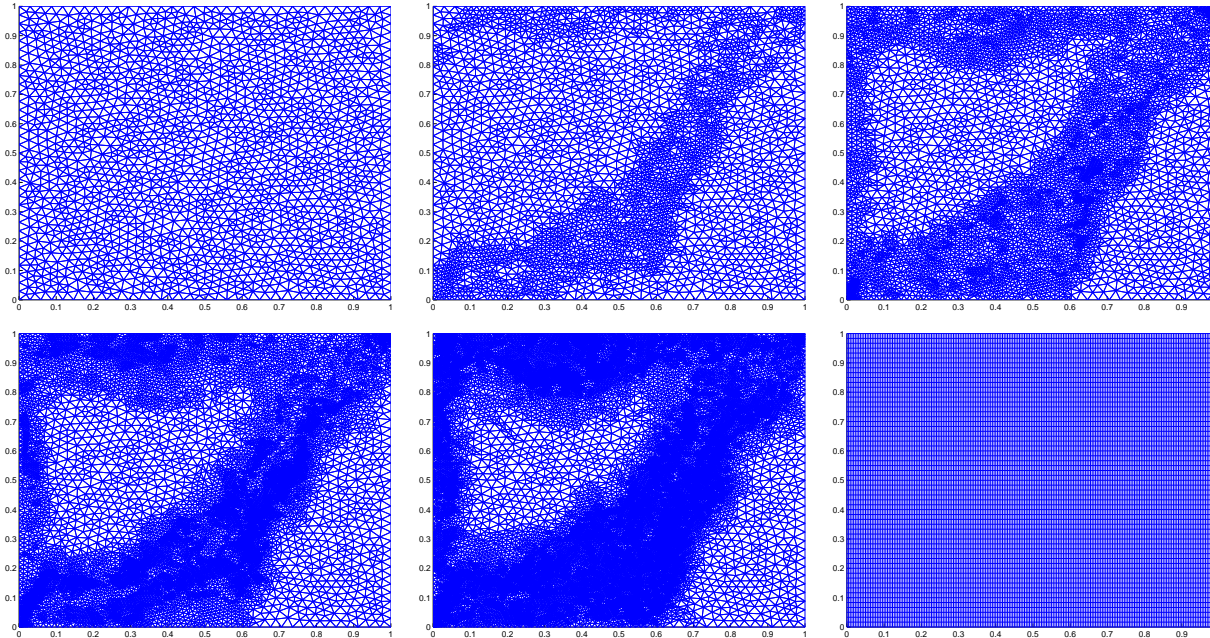

Adaptive meshes

Figure 13: Mesh after each of the five iterations. Rectangular mesh for the flow problem.
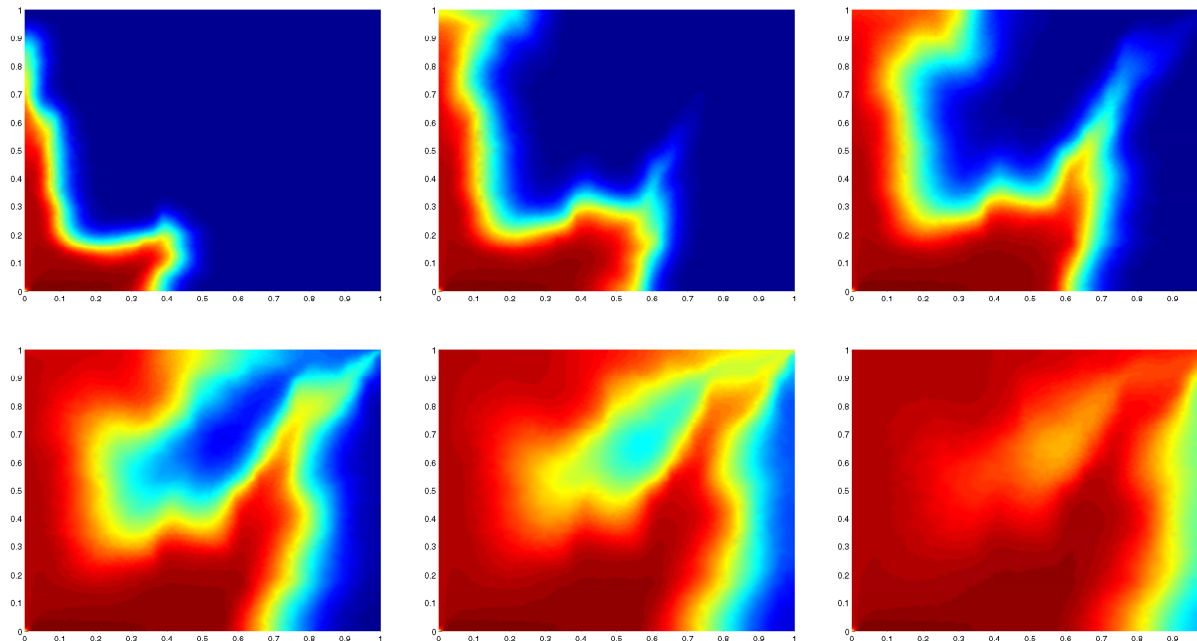
Solution after five iterations

Figure 14: The final solution to the transport problem after five iterations.
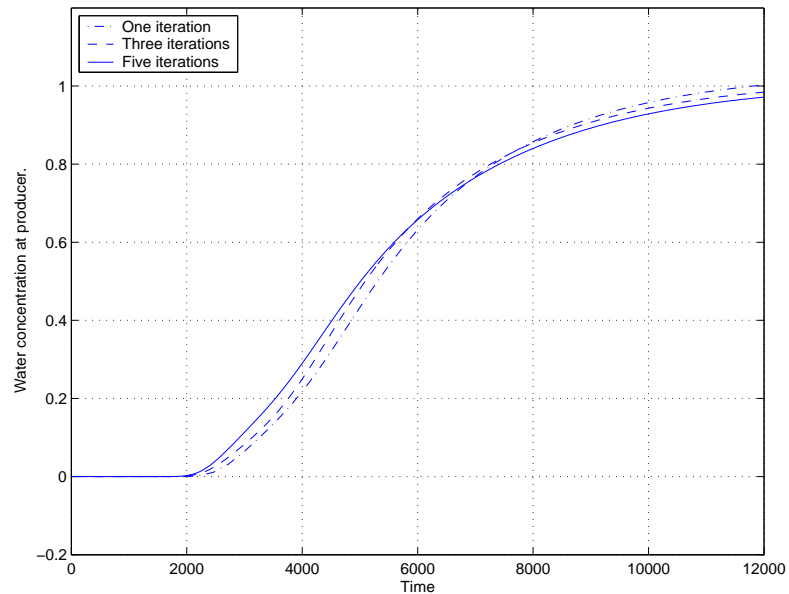
Specific output quality

Figure 15: The water concentration at the producer at different times for approximations after one, three, and five iterations.
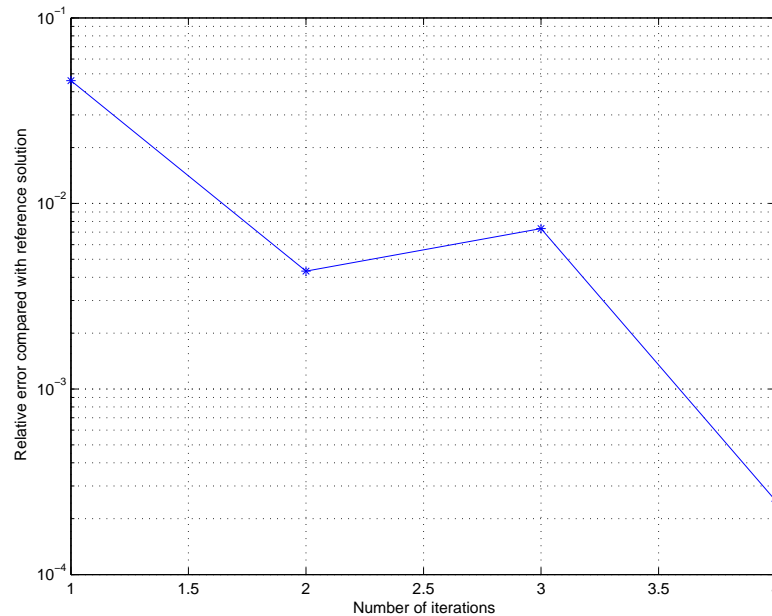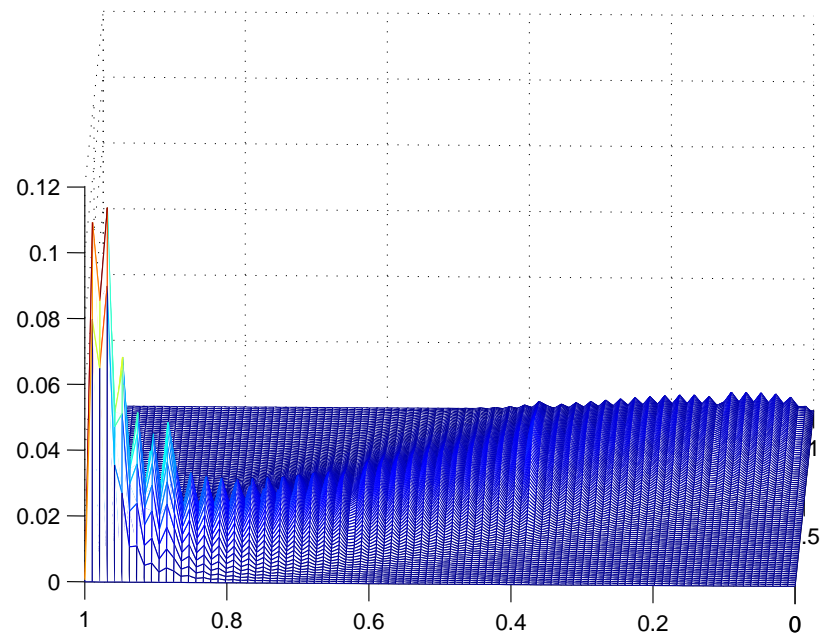
Convergence to reference solution

Figure 16: Convergence to reference solution (last iterate).

Extension to convection-diffusion

Ongoing research

Model problem:

$$-\nabla \cdot \epsilon \nabla u + b \cdot \nabla u = f \quad \text{in } \Omega,$$

$$u = 0 \quad \text{on } \Gamma.$$

Weak form: Find $u \in V = H_0^1(\Omega)$ such that,

$$a(u, v) + b(u, v) = l(v) \quad \text{for all } v \in H_0^1(\Omega),$$

where $a(v, w) = \int_\Omega \epsilon \nabla v \cdot \nabla w \, dx$,
$b(u, v) = \int_\Omega b \cdot \nabla u \, v \, dx$ and $l(v) = \int_\Omega f v \, dx$.
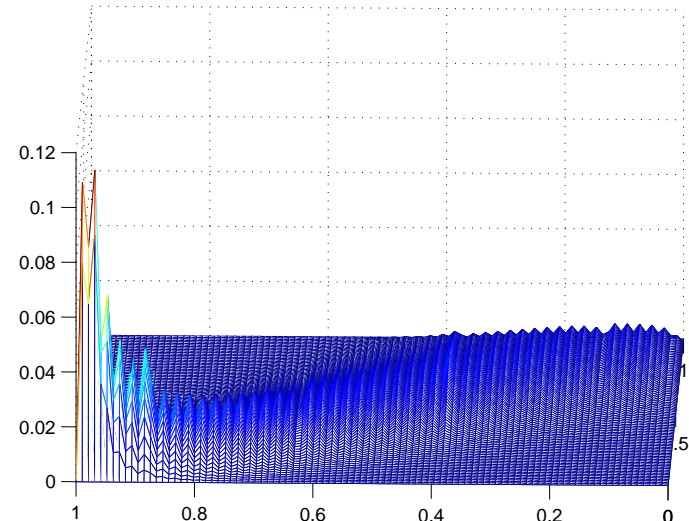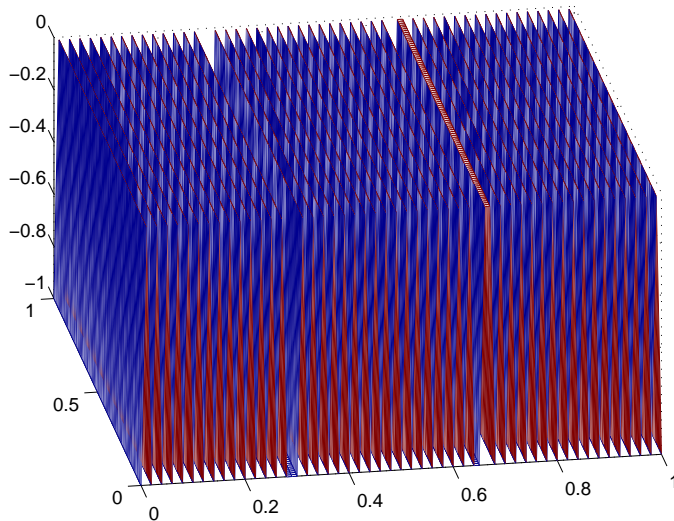
Multiscale method for this problem

The multiscale method reads: find $U_c \in V_c$ and $U_f = \sum_{i \in \mathcal{N}} U_{f,i}$ where $U_{f,i} \in V_f^h(\omega_i)$ such that,

$$a(U_c, v_c) + a(U_f, v_c) + b(U_c, v_c) + b(U_f, v_c) = l(v_c),$$
$$a(U_{f,i}, v_f) + b(U_{f,i}, v_f) = (R(U_c), \varphi_i v_f),$$

for all $v_c \in V_c$, $v_f \in V_f^h(\omega_i)$, and $i \in \mathcal{N}$.

Numerical example

Let $\epsilon = 0.01$, f is a big close to $[0,0]$, and $b = -[b_x, b_x]$ as in figure:
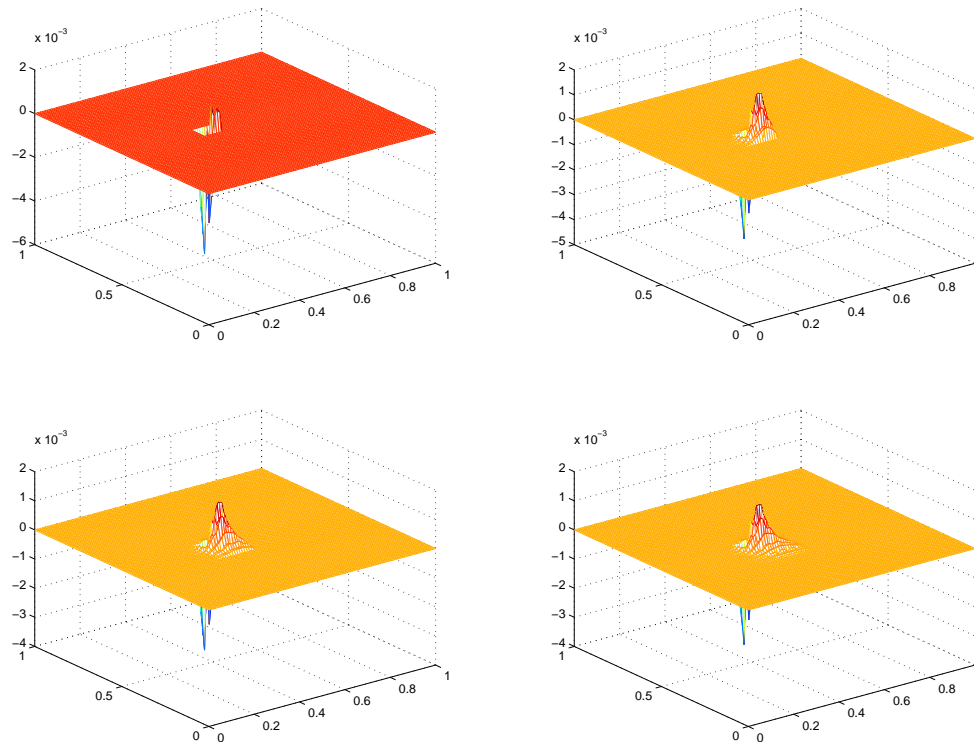
Numerical example

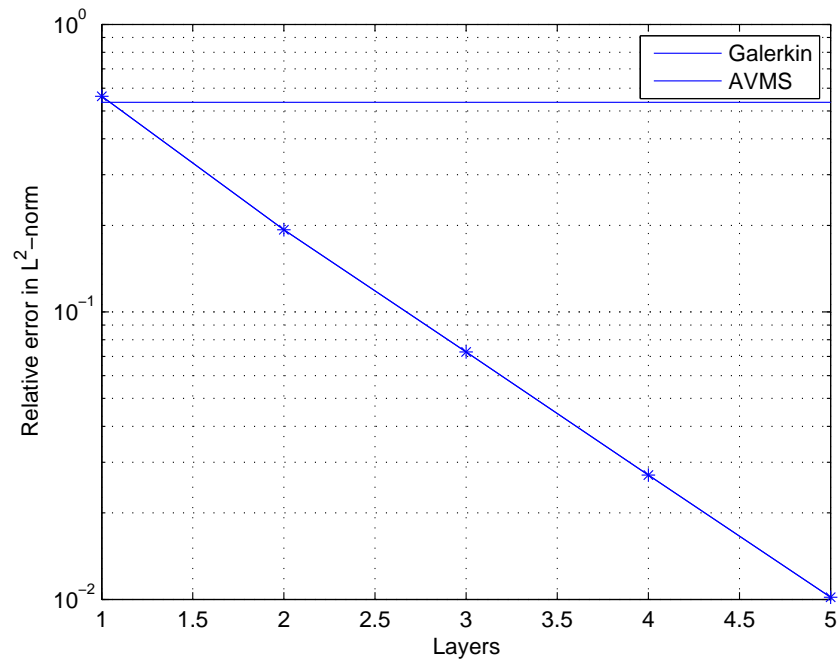Figure 17: $U_{f,i}$ for different patches.

Convergence

Figure 18: In this example $h = 1/100$ and $H = 4h$.

Comments and questions

- Study quadrature error since b will not be resolved on coarse meshes. This will help to determine h.

- Adaptivity should now create patches that are very influenced by b. The boundary error should be seen as a function on the boundary.

- Is it possible to do a better split $V = V_c \oplus V_f$ in this case?

- What happens when $\epsilon \to 0$?