Adaptive Variational Multiscale Methods

Axel Målqvist and Mats G. Larson

axel.malqvist@it.uu.se

Division of Scientific Computing,

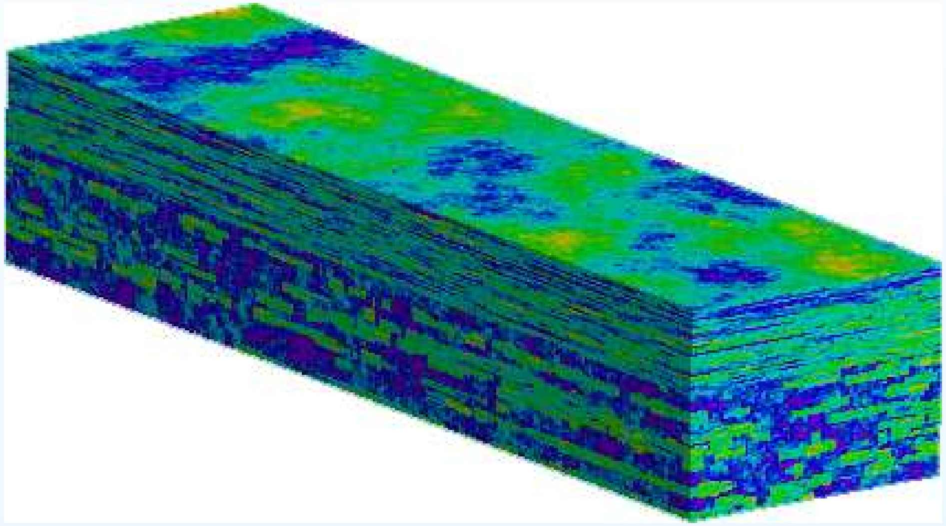
Department of Information Technology, Uppsala University

Outline

- Multiscale application
- Model problem: Poisson equation
- Multiscale method
- Implementation
- A posteriori error estimation
- Extension to mixed formulation
- Numerical examples
- Extension to convection-dominated problems
- Conclusions

An application

The figure illustrates data taken from a model oil reservoir.



The size of the reservoir is about $368m \times 671m \times 52m$. The problem features many different scales. We see the x-component of the permeability a.

An application

We seek the water concentration c that solves the system of a flow and a transport equation,

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

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

This is a simple model, e.g. there is only one way coupling, in general a will depend on c.

The model problem

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,

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

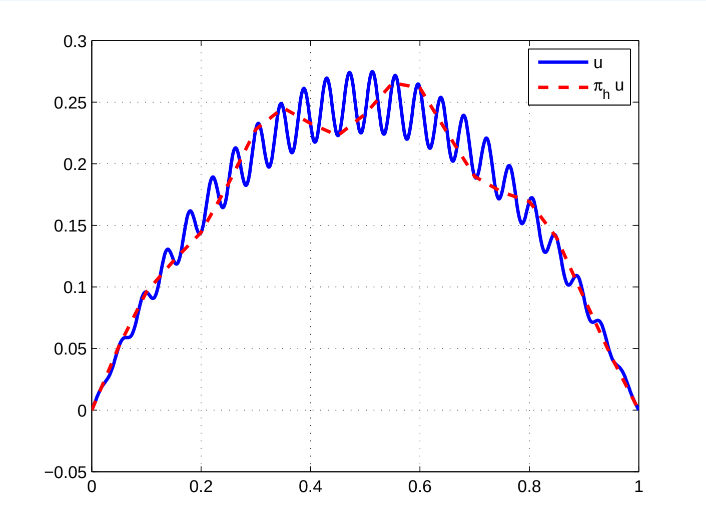
where $a(v, w) = \int_\Omega a\nabla v \cdot \nabla w \, dx$, $l(v) = \int_\Omega f v \, dx$, $f \in L^2(\Omega)$ and Ω is a domain in $\mathbf{R}^d$, $d = 1, 2, 3$. We will also use the notation (v, w) for dual pairing between v and w, in most cases $(v, w) = \int_\Omega vw \, dx$.

Why multiscale method?

- If we assume a to oscillate at a characteristic scale ϵ we have (Hou),

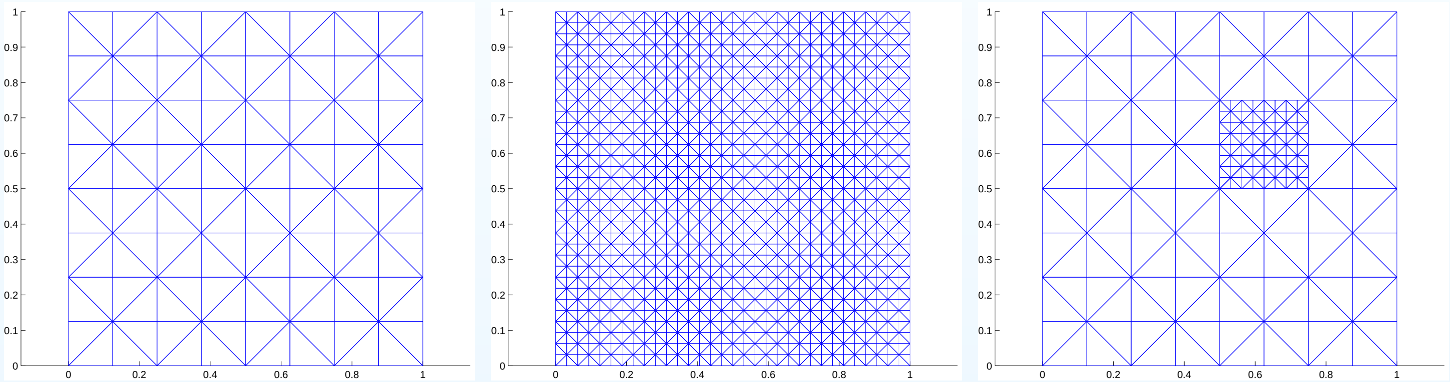
$$\|\nabla u - \nabla U\| \leq C \frac{H}{\epsilon} \|f\|.$$

- $H > \epsilon$ will give unreliable results even with exact quadrature.
- $H < \epsilon$ will be to computationally expensive to solve on a single mesh.



Conclusion of the simple estimate

We need to solve PDE:s on a scale that captures the oscillations but we can not afford to do it on the entire domain.



Coarse $H > \epsilon$ and fine $h < \epsilon$ mesh. This will not be done by meshrefinement but by solving local problems decoupled from each other and from the coarse mesh.

Various multiscale methods

- Upscaling techniques: Durlofsky et al, Nielsen et al. Here an effective permeability $\bar{a}$ is computed by local solves and then a coarse scale equation is solved.
- Multiscale finite element method: Hou et al., Efendiev-Ginting, Aarnes-Lie. Here local solves are used to modify coarse basis functions.
- Multiscale finite volume method: Jenny et al. As above.
- Variational multiscale method: Hughes et al. , Arbogast, Larson-Målqvist. Here the weak form is modified with new stabilizing terms. It can sometimes also be viewed as modification of basis functions.

The variational multiscale method

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

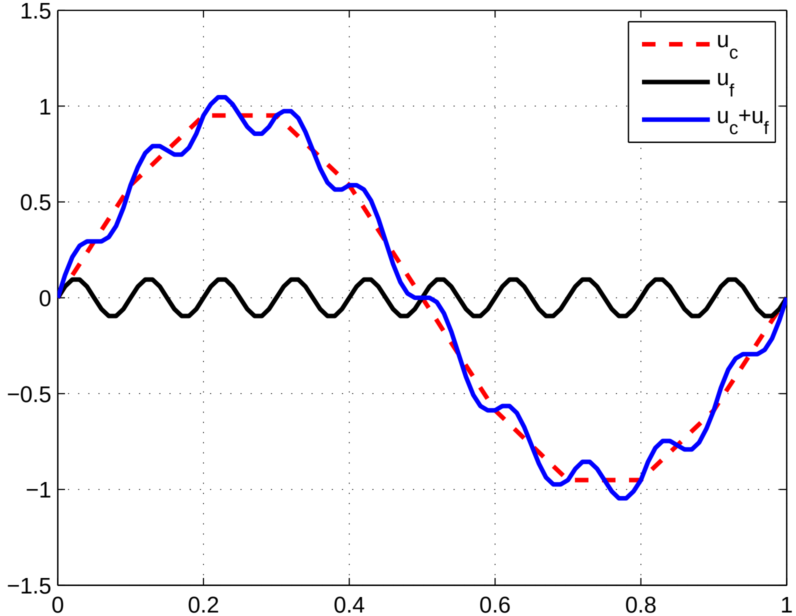
$$a(u_c + u_f, v_c + v_f) = l(v_c + v_f),$$

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

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

where we introduce the residual distribution $R : V \to V'$, $(R(v), w) = l(w) - a(v, w)$, for all $v, w \in V$. Here V_c is a coarse finite element space of piecewise linear basis functions and V_f is therefore zero in all coarse nodes.

The variational multiscale method



Example on what u_c, u_f, and $u_c + u_f$ may look like.

Approximation (our version)

We derive the method in two steps. Remember the fine scale equations, $a(u_f, v_f) = (R(u_c), v_f)$.

- We decouple the fine scale equations by introducing a partition of unity $\sum_{i \in \mathcal{N}} \varphi_i = 1$ (typically consists of coarse basis functions),

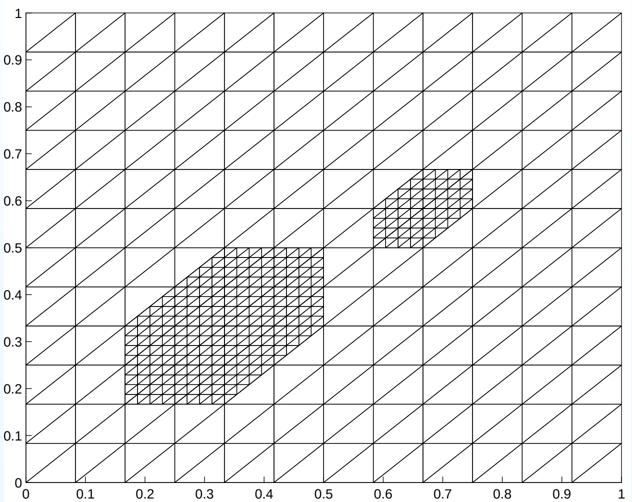
$$a(u_{f,i}, v_f) = (\varphi_i R(u_c), v_f) \quad \text{for all } v_f \in V_f.$$

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

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

We use homogeneous Dirichlet bc.

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.

This leads to an overlapping method.

Our method

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

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

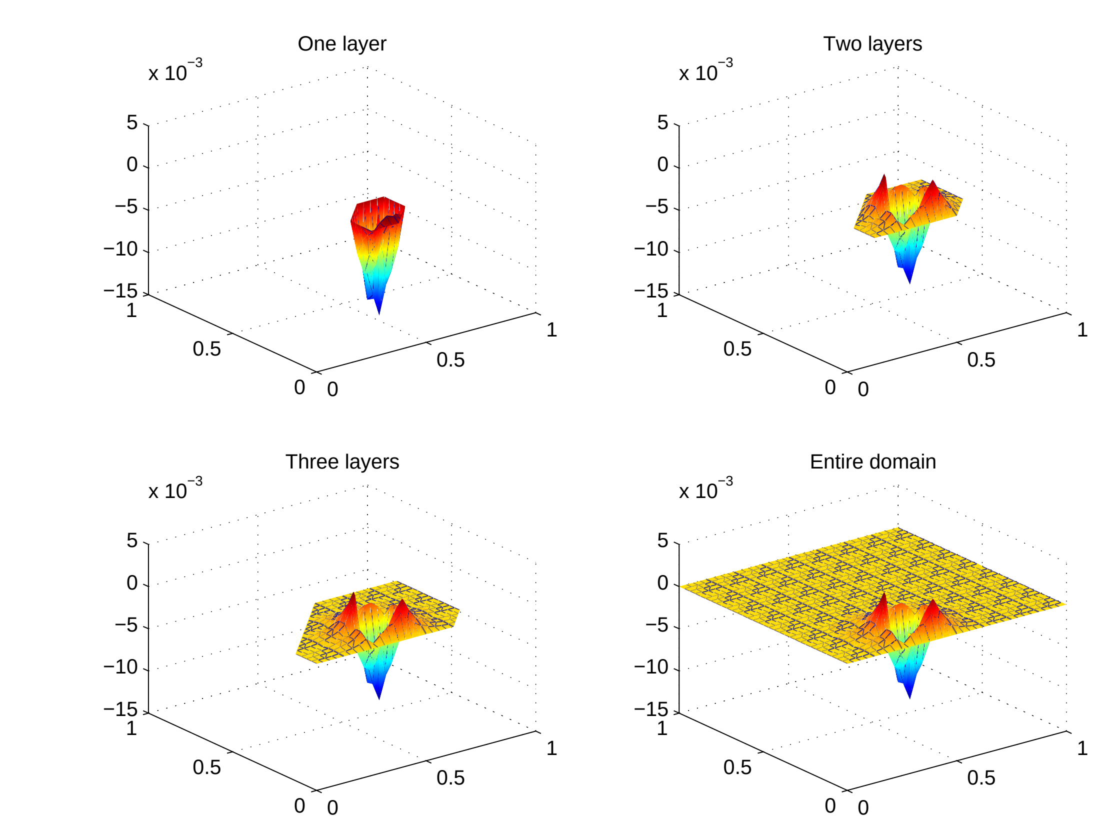
for all $v_c \in V_c$, $v_f \in V_f^h(\omega_i)$, and $i \in \mathcal{N}$.

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

An iterative approach where we start with a given U_c is possible

but we will instead consider a direct solution approach.

The local solution $U_{f,i}$



The solution improves as the patch size increases.

Motivation for 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, $\varphi_i R(U_c)$.
- The fine scale solution $U_{f,i} \in V_f^h(\omega_i)$ which is a slice space. This means that $U_{f,i} = 0$ in all coarse nodes.

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

Implementation

We have: find $U_{f,i} \in V_f^h(\omega_i)$ such that

$$a(U_{f,i}, v_f) = (f, v_f \varphi_i) - a(U_c, v_f \varphi_i)$$

for all $v_f \in V_f^h(\omega_i)$. Instead we solve: find $\chi_i^k, \eta_i \in V_f^h(\omega_i)$ such that

$$\begin{cases} a(\chi_i^k, v_f) = -a(\varphi_k, v_f \varphi_i) \\ a(\eta_i, v_f) = (f, v_f \varphi_i). \end{cases}$$

for all $v_f \in V_f^h(\omega_i)$ and $supp(\varphi_k) \cap supp(\varphi_i) \neq \emptyset$. In the first equation we replace f by 0 and $U_c = \sum U_c^k \varphi_k$ by φ_k (coarse basis function) and in the second equation we keep f and replace U_c by 0.

Implementation

This means that: $\sum_{k\in\mathcal{N}} U_c^k \chi_i^k + \eta_i$ solves:

$$a(\sum_{k\in\mathcal{N}} U_c^k \chi_i^k + \eta_i, v_f) = (f, v_f \varphi_i) - a(U_c, v_f \varphi_i),$$

so $U_{f,i} = \sum_{k\in\mathcal{N}} U_c^k \chi_i^k + \eta_i$ and

$$U_f = \sum_{i\in\mathcal{N}} \sum_{k\in\mathcal{N}} U_c^k \chi_i^k + \eta_i = \sum_{k\in\mathcal{N}} U_c^k \chi^k + \eta,$$

where $\chi^k = \sum_{i\in\mathcal{N}} \chi_i^k$ and $\eta = \sum_{i\in\mathcal{N}} \eta_i$. All this works because the equation is linear.

Implementation

We include this in the coarse scale equations: Find $U_c = \sum_{i\in\mathcal{N}} U_c^i \varphi_i$ such that,

$$\begin{aligned}(f,\varphi_j) =& a(U_c,\varphi_j) + a(U_f,\varphi_j)\\ &= a(\sum_{i\in\mathcal{N}} U_c^i\varphi_i,\varphi_j) + a(\sum_{i\in\mathcal{N}} U_c^i\chi^i + \eta,\varphi_j),\end{aligned}$$

for all $j \in \mathcal{N}$ or

$$\sum_{i\in\mathcal{N}} U_c^i a(\varphi_i + \chi^i,\varphi_j) = (f,\varphi_j) - a(\eta,\varphi_j),$$

which gives a modified system. The degrees of freedom is the same but more non-zero elements compared to Galerkin on coarse mesh.

Implementation

This can now be written on matrix form as,

$$(A+T)U = b - d$$

where,

$$\begin{cases} A_{mn} = a(\varphi_m, \varphi_n), \\ T_{mn} = a(\chi^m, \varphi_n), \\ b_n = (f, \varphi_n), \\ d_n = a(\eta, \varphi_n). \end{cases}$$

or alternatively a symmetric formulation,

$$\begin{cases} A_{mn} + T_{mn} = a(\varphi_m + \chi_m, \varphi_n + \chi_n), \\ b_n = (f, \varphi_n + \chi_n), \\ d_n = a(\eta, \varphi_n + \chi_n). \end{cases}$$

Implementation

Implementing the method comes down to calculating T and d on patches ω_i, $T = \sum_{i \in \mathcal{N}} T^i$ and $d = \sum_{i \in \mathcal{N}} d^i$, where

$$T^i_{mn} = a(\chi^m_i, \varphi_n)$$

and

$$d^i_n = a(\eta_i, \varphi_n).$$

1. Compute A and b.
2. Compute T^i and d^i on the patches ω_i.
3. Solve $(A + T)U = b - d$.
4. Estimate error in U and improve resolution if necessary.

Given the vector U, $\{\chi^m\}_{m=1}^N$, and η we can reconstruct the fine scale solution locally.

How do we choose 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, such as coarse mesh size, fine mesh size, and patch sizes, automatically.

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

We introduce the coarse and fine scale error $e_c = u_c - U_c$, $e_{f,i} = u_{f,i} - U_{f,i}$, and $e = e_c + \sum_{i\in\mathcal{N}} e_{f,i}$. We have the following orthogonality properties,

$$a(e_c, v_c) + a(e_f, v_c) = 0, \quad \text{for all } v_c \in V_c$$

and

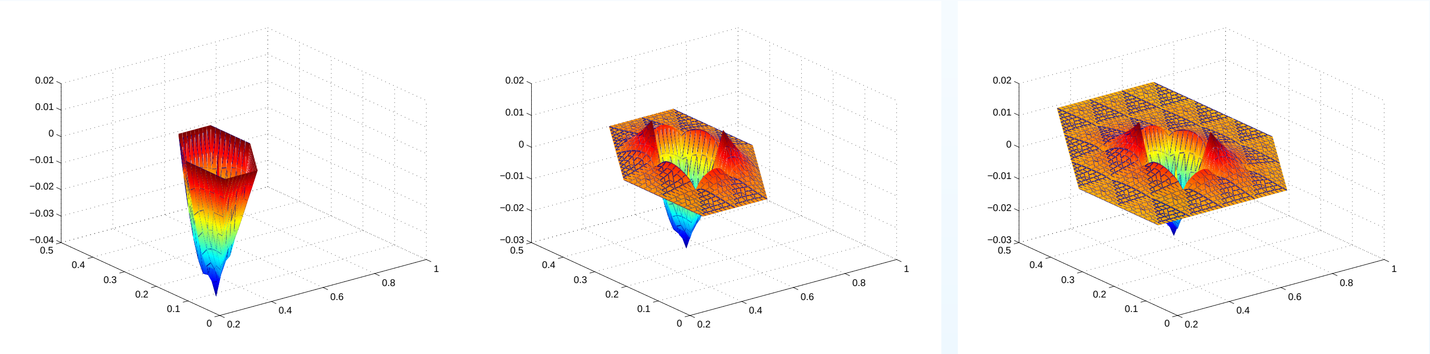
$$a(e_{f,i}, v_f) + a(e_c, \varphi_i v_f) = 0, \quad v_f \in V_f^h(\omega_i).$$

and estimate,

$$\|e\|_a^2 \leq \sum_{i\in\mathcal{C}} C_a \|H\mathcal{R}(U_c)\|_{\omega_i}^2 + \sum_{i\in\mathcal{F}} C_a \left(\|h\mathcal{R}_i(U_{f,i})\|_{\omega_i}^2 + \|\sqrt{H}\Sigma(U_{f,i})\|_{\partial\omega_i}^2 \right)$$

Energy norm estimate

The boundary part $\|\sqrt{H}\Sigma(U_{f,i})\|^2_{\partial\omega_i}$, where $\Sigma(U_{f,i})$ is an approximation of $a\partial_n U_{f,i}$, decays rapidly as the patch increases.



Here the patches are one, two, and three layer stars. The term $\mathcal{R}(U_c)$ is a bound of the coarse scale residual ("$f - \nabla \cdot a\nabla U_c$") and $\mathcal{R}_i(U_{f,i})$ is a bound of the fine scale residual.

Adaptive algorithm based on error estimate

$$\|e\|_a^2 \leq \sum_{i\in\mathcal{C}} C_a \|H\mathcal{R}(U_c)\|_{\omega_i}^2 + \sum_{i\in\mathcal{F}} C_a \left(\|h\mathcal{R}_i(U_{f,i})\|_{\omega_i}^2 + \|\sqrt{H}\Sigma(U_{f,i})\|_{\partial\omega_i}^2 \right)$$

1. Start with given H, r, and L where $h = H/2^r$.
2. Calculate U using AVMS.
3. Calculate $E_H^i = \|H\mathcal{R}(U_c)\|_{\omega_i}^2$, $E_r^i = \|h\mathcal{R}_i(U_{f,i})\|_{\omega_i}^2$, and $E_L^i = \|\sqrt{H}\Sigma(U_{f,i})\|_{\partial\omega_i}^2$.
4. Stop if E_H^i, E_h^i, and E_L^i are small enough else if E_H^i is big start solving local problems there with $r = L = 1$, if $E_h^i > E_L^i$ let $r_{\text{new}} := r + 1$ and if $E_L^i > E_h^i$ let $L_{\text{new}} = L + 1$ end return to 2.

Summary of the main idea

- We want to avoid to refine a given coarse mesh
- Still the problem makes it necessary to use better resolution
- We have a method for solving decoupled fine scale problems that are used to modify the coarse scale equation
- We have an error estimate that tells us where and how accurate these fine scale problems need to be solved
- We have an adaptive algorithm that makes these decisions automatically.

We now study other equations and some applications.

Extension to a mixed setting with application

We have also extended this theory to the mixed formulation of the Poisson equation (which is used in oil reservoir simulation),

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

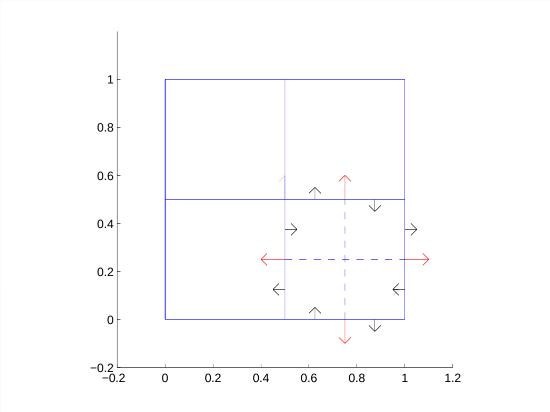
Find $\boldsymbol{\sigma}_c \in \boldsymbol{V}_c$, $\boldsymbol{\sigma}_f \in \boldsymbol{V}_f$, $u_c \in W_c$, and $u_f \in W_f$ such that,

$$\begin{cases} (\frac{1}{a}\boldsymbol{\sigma}_c, \boldsymbol{v}_c) + (\frac{1}{a}\boldsymbol{\sigma}_f, \boldsymbol{v}_c) + (u_c, \nabla \cdot \boldsymbol{v}_c) + (u_f, \nabla \cdot \boldsymbol{v}_c) = 0 \\ -(\nabla \cdot \boldsymbol{\sigma}_c, w_c) - (\nabla \cdot \boldsymbol{\sigma}_f, w_c) = (f, w_c) \\ (\frac{1}{a}\boldsymbol{\sigma}_f, \boldsymbol{v}_f) + (u_f, \nabla \cdot \boldsymbol{v}_f) = -(\frac{1}{a}\boldsymbol{\sigma}_c, \boldsymbol{v}_f) - (u_c, \nabla \cdot \boldsymbol{v}_f) \\ -(\nabla \cdot \boldsymbol{\sigma}_f, w_f) = (f, w_f) + (\nabla \cdot \boldsymbol{\sigma}_c, w_f) \end{cases}$$

for all $\boldsymbol{v}_c \in \boldsymbol{V}_c$, $\boldsymbol{v}_f \in \boldsymbol{V}_f$, $w_c \in W_c$, and $w_f \in W_f$.

Basis for V_c, V_f, W_c, and W_f.

- For V_c and W_c: Lowest order Raviart-Thomas elements on rectangles together with piecewise constants.
- For V_f and W_f: Hierarchical extension.



We note, $(w_c, \nabla \cdot \boldsymbol{v}_f) = \sum_K w_c^K \int_{\partial K} \boldsymbol{n} \cdot \boldsymbol{v}_f \, dx = 0$ where w_c^K is the constant at coarse element K, $(w_f, \nabla \cdot \boldsymbol{v}_c) = \sum_K \nabla \cdot \boldsymbol{v}_c^K \int_K w_f \, dx = 0$.

Variational multiscale method

Find $\boldsymbol{\sigma}_c \in \boldsymbol{V}_c$, $\boldsymbol{\sigma}_f \in \boldsymbol{V}_f$, $u_c \in W_c$, and $u_f \in W_f$ such that,

$$\begin{cases} (\frac{1}{a}\boldsymbol{\sigma}_c, \boldsymbol{v}_c) + (\frac{1}{a}\boldsymbol{\sigma}_f, \boldsymbol{v}_c) + (u_c, \nabla\cdot\boldsymbol{v}_c) + (u_f, \nabla\cdot\boldsymbol{v}_c) = 0 \\ -(\nabla\cdot\boldsymbol{\sigma}_c, w_c) - (\nabla\cdot\boldsymbol{\sigma}_f, w_c) = (f, w_c) \\ (\frac{1}{a}\boldsymbol{\sigma}_f, \boldsymbol{v}_f) + (u_f, \nabla\cdot\boldsymbol{v}_f) = -(\frac{1}{a}\boldsymbol{\sigma}_c, \boldsymbol{v}_f) - (u_c, \nabla\cdot\boldsymbol{v}_f) \\ -(\nabla\cdot\boldsymbol{\sigma}_f, w_f) = (f, w_f) + (\nabla\cdot\boldsymbol{\sigma}_c, w_f) \end{cases}$$

for all $\boldsymbol{v}_c \in \boldsymbol{V}_c$, $\boldsymbol{v}_f \in \boldsymbol{V}_f$, $w_c \in W_c$, and $w_f \in W_f$.

If we let $\boldsymbol{\varphi}_i$ be a coarse Raviart-Thomas base function on rectangles (bricks), $\boldsymbol{\varphi}_i = \begin{bmatrix} \varphi_i^x \\ \varphi_i^y \end{bmatrix}, \varphi_i = \begin{bmatrix} \varphi_i^x & 0 \\ 0 & \varphi_i^y \end{bmatrix}$, will be a partition of unity, $\sum_{i\in\mathcal{N}} \varphi_i = 1$. We let $\psi_i = 1/(2d)$ one the support of φ_i.

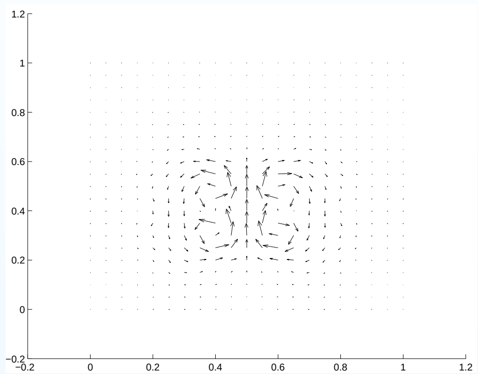
Partition of unity

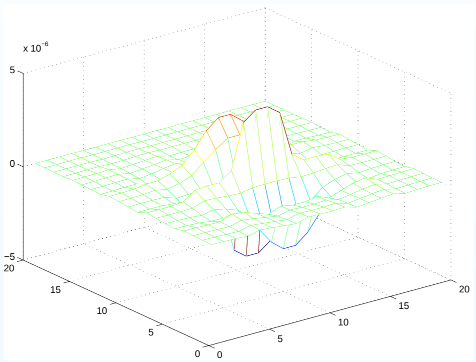
Find $\boldsymbol{\sigma}_c$, u_c, $\boldsymbol{\sigma}_f = \sum_{i\in\mathcal{N}} \boldsymbol{\sigma}_{f,i}$, and $u_f = \sum_{i\in\mathcal{N}} u_{f,i}$ such that,

$$\left\{\begin{aligned} (\tfrac{1}{a}\boldsymbol{\sigma}_c, \boldsymbol{v}_c) + (\tfrac{1}{a}\boldsymbol{\sigma}_f, \boldsymbol{v}_c) + (u_c, \nabla\cdot\boldsymbol{v}_c) &= 0, \\ -(\nabla\cdot\boldsymbol{\sigma}_c, w_c) &= (f, w_c), \\ (\tfrac{1}{a}\boldsymbol{\sigma}_{f,i}, \boldsymbol{v}_f) + (u_{f,i}, \nabla\cdot\boldsymbol{v}_f) &= -(\tfrac{1}{a}\boldsymbol{\sigma}_c, \varphi_i\boldsymbol{v}_f), \\ -(\nabla\cdot\boldsymbol{\sigma}_{f,i}, w_f) &= (f, \psi_i w_f), \end{aligned}\right.$$

for all $\boldsymbol{v}_c \in \boldsymbol{V}_c$, $\boldsymbol{v}_f \in \boldsymbol{V}_f$, $w_c \in W_c$, and $w_f \in W_f$.

Local solutions





The local solutions $\boldsymbol{\sigma}_{f,i}$ and $u_{f,i}$ for $a = 1$. We introduce patches since,

- The right hand side has support on $supp(\boldsymbol{\varphi}_i) = supp(\psi_i)$.
- The equations are solved in a slice space where solutions decay rapidly. $\int_E \boldsymbol{n} \cdot \boldsymbol{\sigma}_{f,i}\, dx = 0$ and $\int_K u_{f,i}\, dx = 0$.

The smallest patch now consist of two coarse elements since the

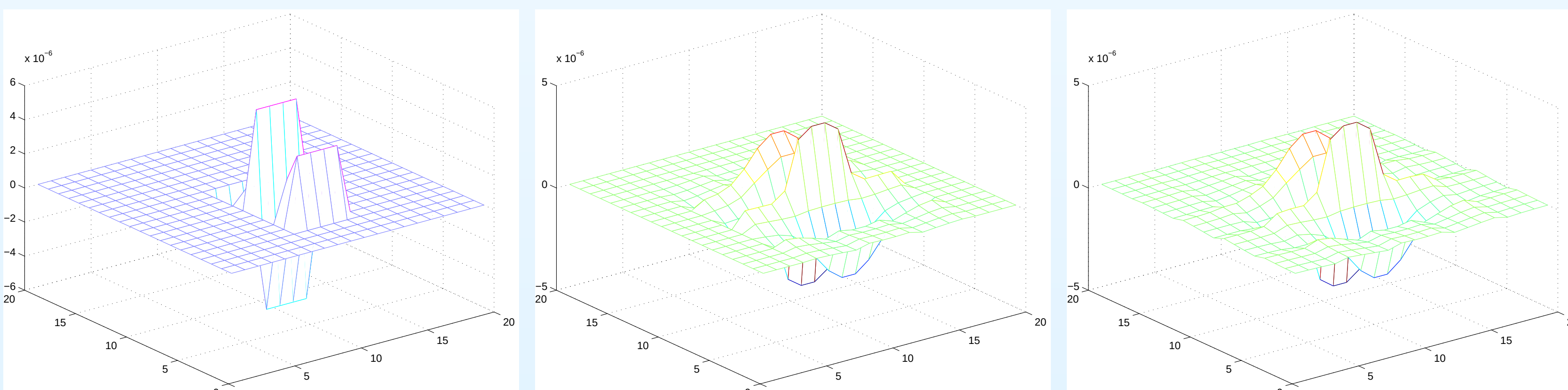
RT basis functions have support on two elements.

Solving local Neumann problems

Find $\boldsymbol{\Sigma}_c \in \boldsymbol{V}_H$, $\boldsymbol{\Sigma}_{f,i} \in \boldsymbol{V}_h(\omega_i)$, $U_c \in W_H$, and $U_{f,i} \in W_h(\omega_i)$

$$\left\{\begin{aligned} (\tfrac{1}{a}\boldsymbol{\Sigma}_c, \boldsymbol{v}_c) + (\tfrac{1}{a}\boldsymbol{\Sigma}_f, \boldsymbol{v}_c) + (U_c, \nabla\cdot\boldsymbol{v}_c) &= 0, \\ -(\nabla\cdot\boldsymbol{\Sigma}_c, w_c) &= (f, w_c), \\ (\tfrac{1}{a}\boldsymbol{\Sigma}_{f,i}, \boldsymbol{v}_f) + (U_{f,i}, \nabla\cdot\boldsymbol{v}_f) &= -(\tfrac{1}{a}\boldsymbol{\Sigma}_c\varphi_i, \boldsymbol{v}_f), \\ -(\nabla\cdot\boldsymbol{\Sigma}_{f,i}, w_f) &= (f, w_f\psi_i), \end{aligned}\right.$$

for all $\boldsymbol{v}_c \in \boldsymbol{V}_H$, $\boldsymbol{v}_f \in \boldsymbol{V}_h(\omega_i)$, $w_c \in W_H$, and $w_f \in W_h(\omega_i)$.



Energy norm estimate $\|\boldsymbol{v}\|_a^2 = (\frac{1}{a}\boldsymbol{v}, \boldsymbol{v})$

Next we present an estimate of the error.

$$\|\boldsymbol{\sigma} - \boldsymbol{\Sigma}\|_a^2 \leq \sum_i C_a \|\frac{1}{a}(\boldsymbol{\Sigma}_c \varphi_i + \boldsymbol{\Sigma}_{f,i}) - \nabla U_{f,i}^*\|_{\omega_i}^2$$

$$+ \sum_i C_a \|h(f\psi_i + \nabla \cdot (\boldsymbol{\Sigma}_c \varphi_i + \boldsymbol{\Sigma}_{f,i}))\|_{\omega_i}^2.$$

$$+ \sum_i C_a \|\frac{1}{2\sqrt{h}} U_{f,i}^*\|_{\partial\omega_i \setminus \Gamma}^2$$

U^* is a post processed version of U modified by information from the flux according to paper by Stenberg et. al.

Adaptive Strategy

- Calculate $\boldsymbol{\Sigma}$.
- Calculate the error indicators on each patch,

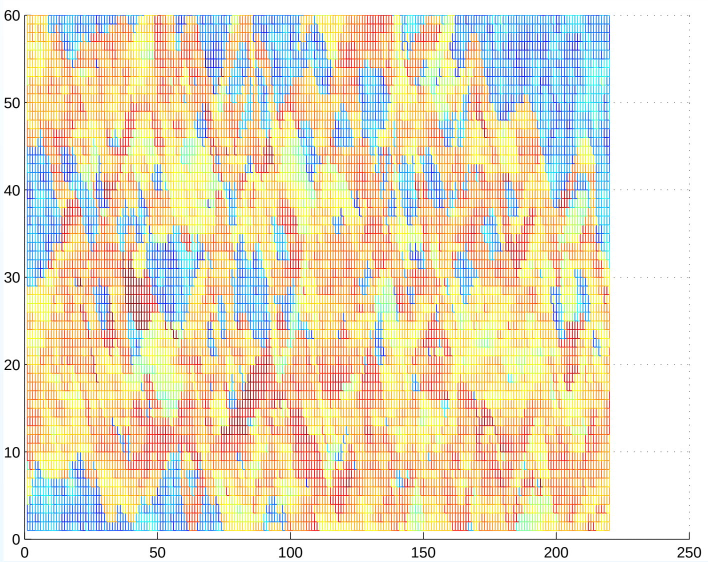
$$X_i(h) = \|\frac{1}{a}(\boldsymbol{\Sigma}_c\varphi_i + \boldsymbol{\Sigma}_{f,i}) - \nabla U^*_{f,i}\|^2_{\omega_i}$$

$$Y_i(h) = \|h(f\psi_i + \nabla\cdot(\boldsymbol{\Sigma}_c\varphi_i + \boldsymbol{\Sigma}_{f,i}))\|^2_{\omega_i}$$

$$Z_i(L) = \|\frac{1}{2\sqrt{h}}U^*_{f,i}\|^2_{\partial\omega_i\backslash\Gamma}$$

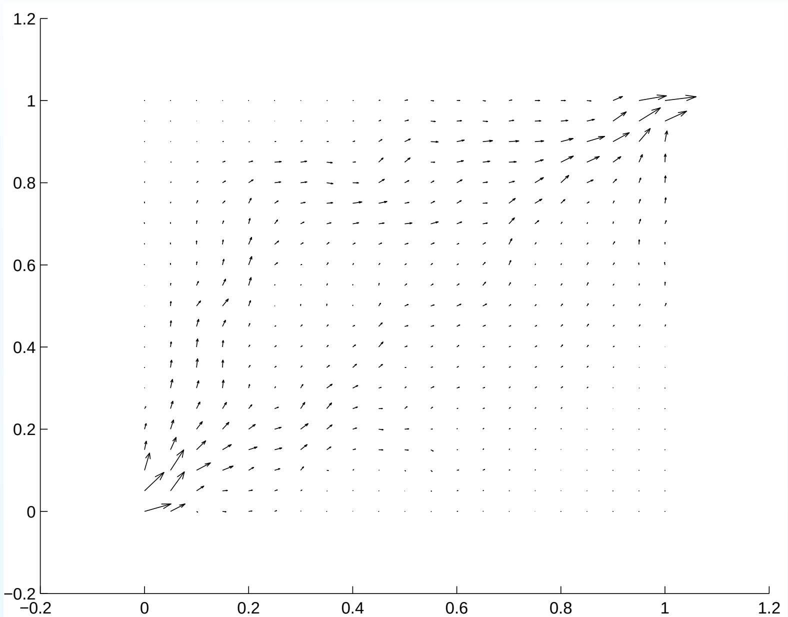
- If indicators $X_i(h)$ or $Y_i(h)$ are big on a patch we decrease h.
- If indicator $Z_i(L)$ is big we increase the size of the patch.
- Go back to the first step or stop if the solution is good enough.

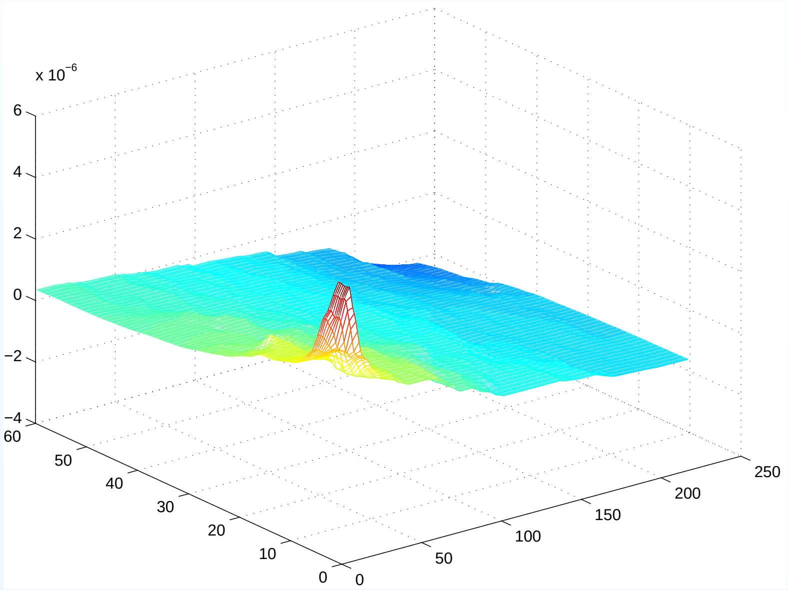
Numerical examples



2D slice of the x-component of the permeability a (in log scale) taken from the tenth SPE comparative solution project.

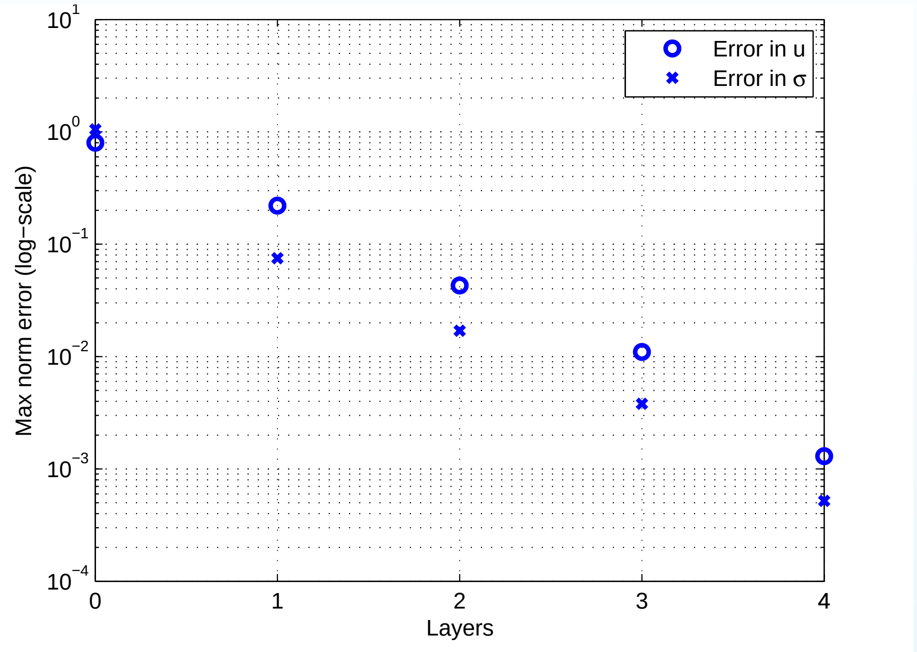
Reference solutions





Above we see the reference solution, (left) flux $-\boldsymbol{\Sigma}$ and (right) pressure u.

Convergence

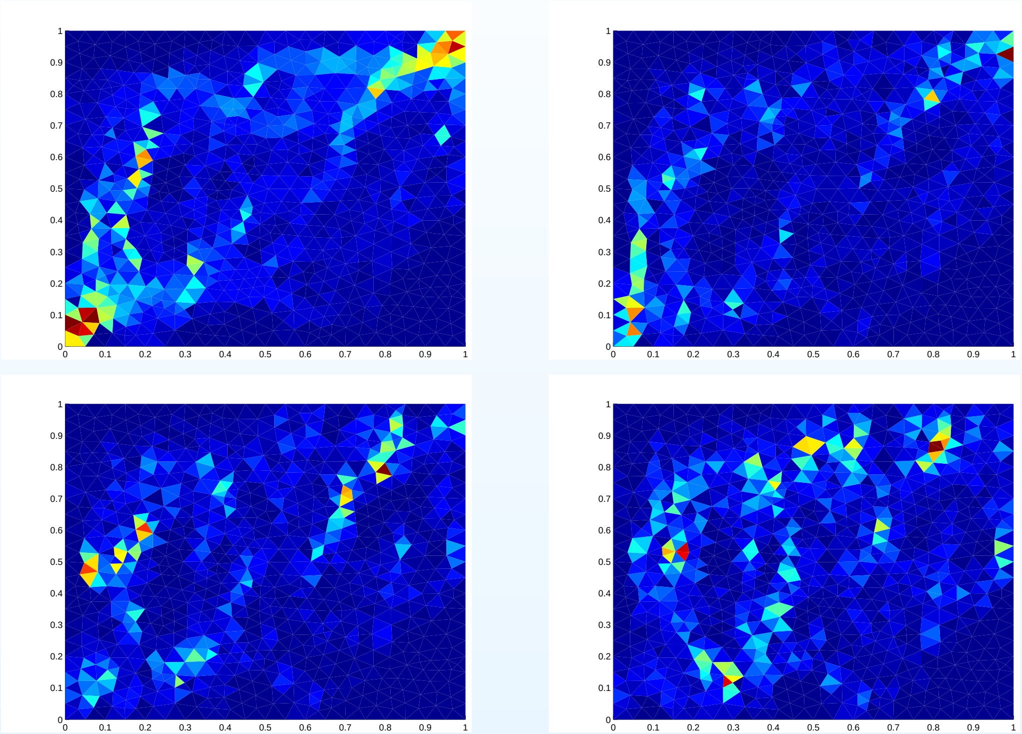


Max norm error (compared to reference solution) in log scale versus number of layers. The coarse mesh has 55×15 elements and the reference mesh has 220×60 elements.

Example using the adaptive algorithm

35% of the patches increased in each iteration and 35% of the fine scale meshes refined in each iteration.

Relative error in energy norm



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

Extension to convection dominated problem

Model problem: Convection-Diffusion problem with multiscale features in b, $\epsilon > 0$,

$$-\epsilon \triangle u + \nabla \cdot (bu) = f \quad \text{in } \Omega,$$
$$u = 0 \quad \text{on } \Gamma.$$

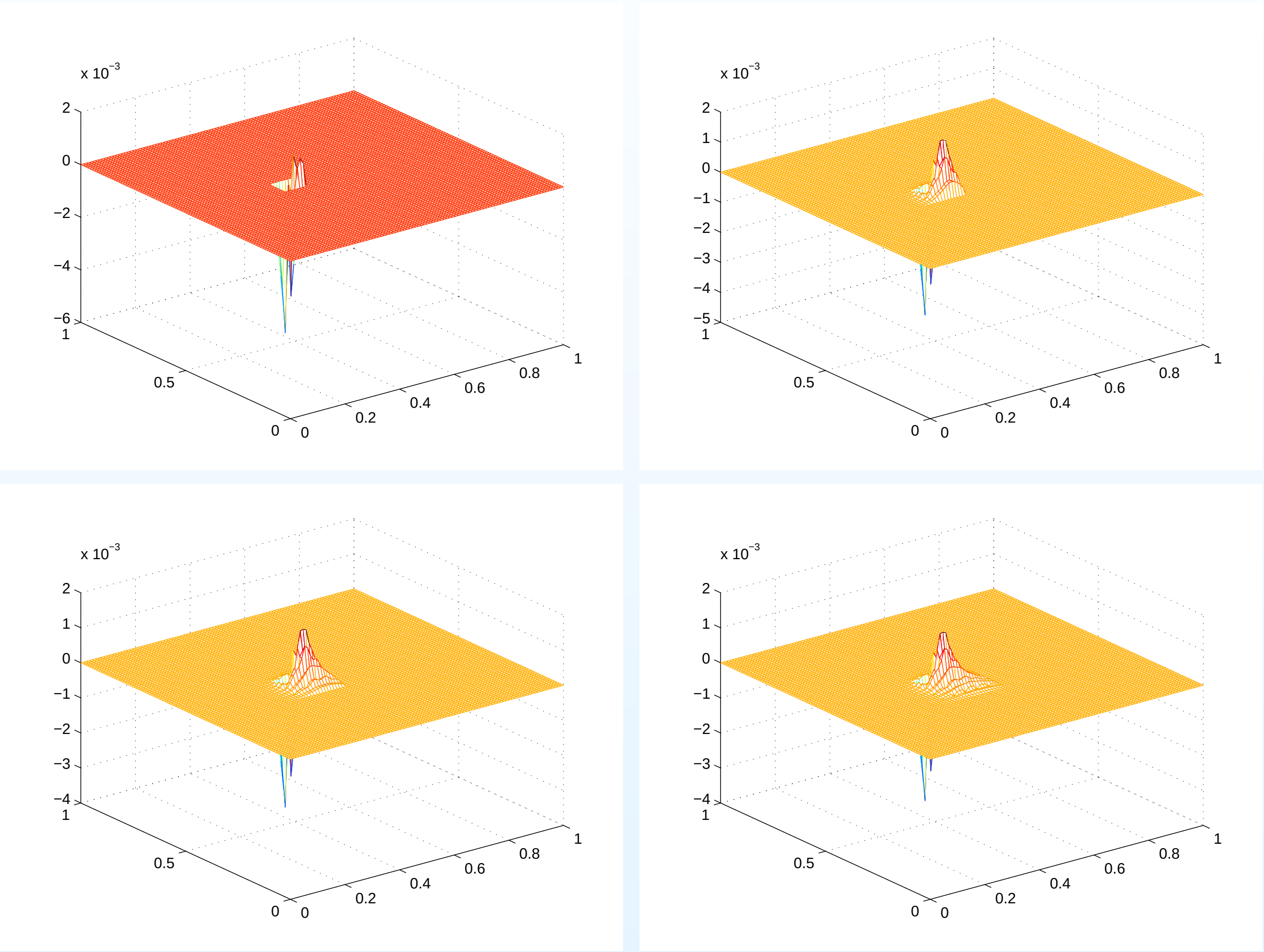
Weak form: Find $u \in V = H_0^1(\Omega)$ such that,

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

where $a(v, w) = \int_\Omega \epsilon \nabla v \cdot \nabla w \, dx + \int_\Omega \nabla \cdot (bv) w \, dx$ and $l(v) = \int_\Omega f v \, dx$.

We use the same method to approximate the solution.

The local solution $U_{f,i}$



The solution improves as the patch size increases.

Duality based error analysis

Find $\phi \in V$ such that

$$(\epsilon\nabla\phi, \nabla w) - (b \cdot \nabla\phi, w) = (\psi, w), \quad \text{for all } w \in V,$$

note that we use the adjoint operator i.e.

$$a^*(\phi, w) = (\psi, w).$$

We end up with the following error representation formula,

$$\begin{aligned}(e, \psi) = a^*(\phi, e) = a(e, \phi) &= l(\phi) - a(U, \phi) \\ &= \sum_{i \in \mathcal{N}} l(\varphi_i\phi) - a(U_c, \varphi_i\phi) - a(U_{f,i}, \phi).\end{aligned}$$

Error representation formula

We continue the calculation using coarse and fine scale Galerkin Orthogonality,

$$
\begin{aligned}
(e,\psi) &= l(\phi - \pi_c\phi) - a(U, \phi - \pi_c\phi) \\
&= \sum_{i\in\mathcal{C}} l(\varphi_i(\phi - \pi_c\phi)) - a(U_c, \varphi_i(\phi - \pi_c\phi)) \\
&\quad + \sum_{i\in\mathcal{F}} l(\varphi_i(\phi_f - \pi^0_{f,i}\phi_f)) - a(U_c, \varphi_i(\phi_f - \pi^0_{f,i}\phi_f)) \\
&\quad - a(U_{f,i}, \phi_f - \pi^0_{f,i}\phi_f),
\end{aligned}
$$

Where $\pi^0_{f,i}$ is the interpolant onto $V^h_f(\omega_i)$ i.e. zero on $\partial\omega_i$. Remember, any function in $V^h_f(\omega_i)$ can be subtracted.

Error representation formula

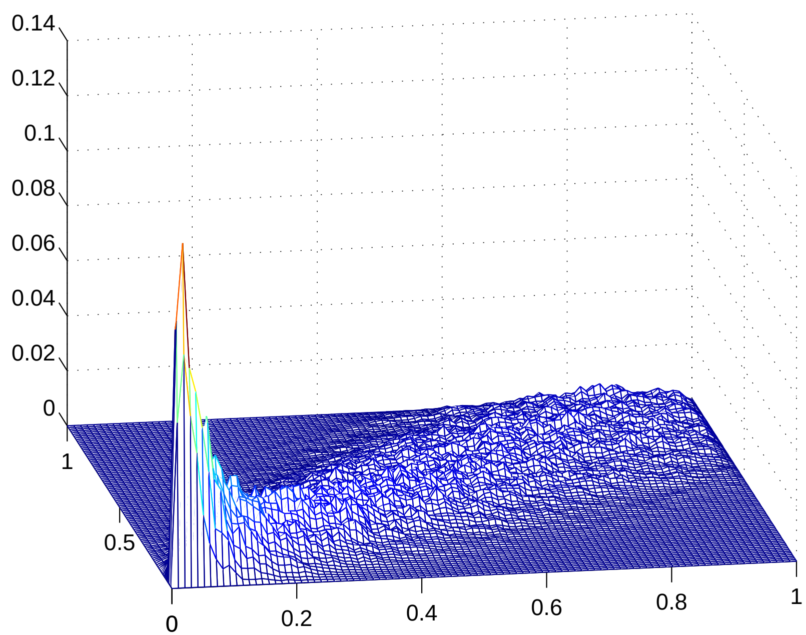
We can also introduce $\pi_{f,i}$ as the nodal interpolant on the mesh associated with $V_f^h(\omega_i)$ and express the error representation formula in terms of $\pi_{f,i}$ and $\pi^0_{f,i} - \pi_{f,i}$.

$$
\begin{aligned}
(e,\psi) = \sum_{i\in\mathcal{C}} l(\varphi_i(\phi - \pi_c\phi)) - a(U_c, \varphi_i(\phi - \pi_c\phi)) \\
+ \sum_{i\in\mathcal{F}} l(\varphi_i(\phi_f - \pi_{f,i}\phi_f)) - a(U_c, \varphi_i(\phi_f - \pi_{f,i}\phi_f)) \\
- a(U_{f,i}, \phi_f - \pi_{f,i}\phi_f) \\
+ \sum_{i\in\mathcal{F}} l(\varphi_i(\pi_{f,i}\phi_f - \pi^0_{f,i}\phi_f)) - a(U_c, \varphi_i(\pi_{f,i}\phi_f - \pi^0_{f,i}\phi_f)) \\
- a(U_{f,i}, \pi_{f,i}\phi_f - \pi^0_{f,i}\phi_f).
\end{aligned}
$$

First depend on H second on h third on patch size.

Numerical examples

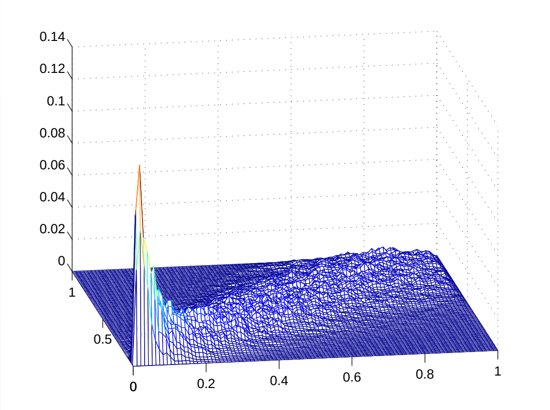
We let $\epsilon = 0.01$, $f = I_{\{x+y<0.05\}}$, and $B = \text{rand}(96)$, $b = [B(i,j), B(i,j)]$ for $i/n < x < (i+1)/n$ and $j/n < y < (j+1)/n$.

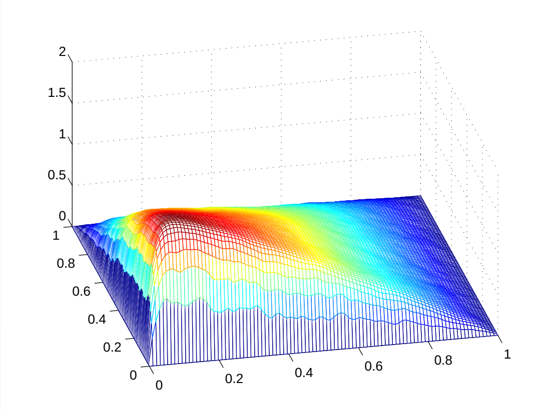


Solving the dual problem for adaptivity

Remember the dual problem: find $\phi \in V$ such that,

$$(\epsilon \nabla \phi, \nabla w) - (b \cdot \nabla \phi, w) = (1, w), \quad \text{for all } w \in V.$$

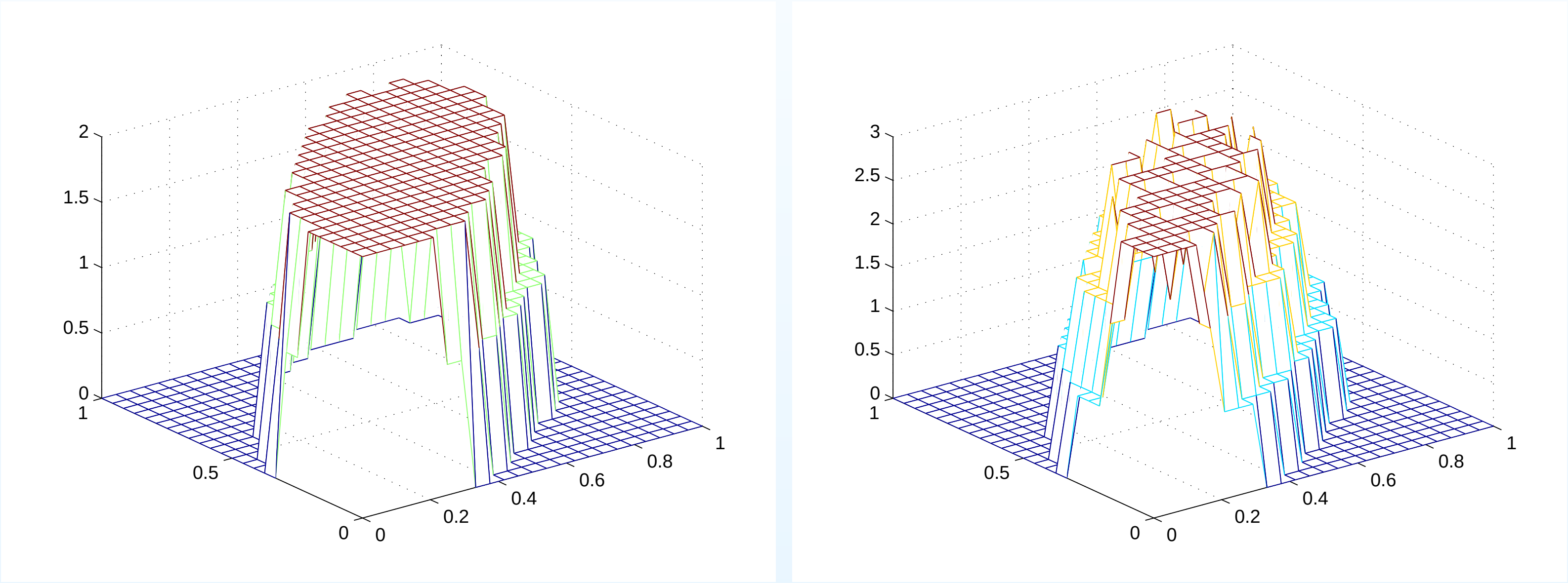




- Computing approximation Φ on the reference mesh or use AVMS with more refinement $\rightarrow$ good approx. of the error.
- Computing Φ using the same method as the primal or $h = H/2$ for all local problems $\rightarrow$ good indicator for adaptivity.

Numerical Examples

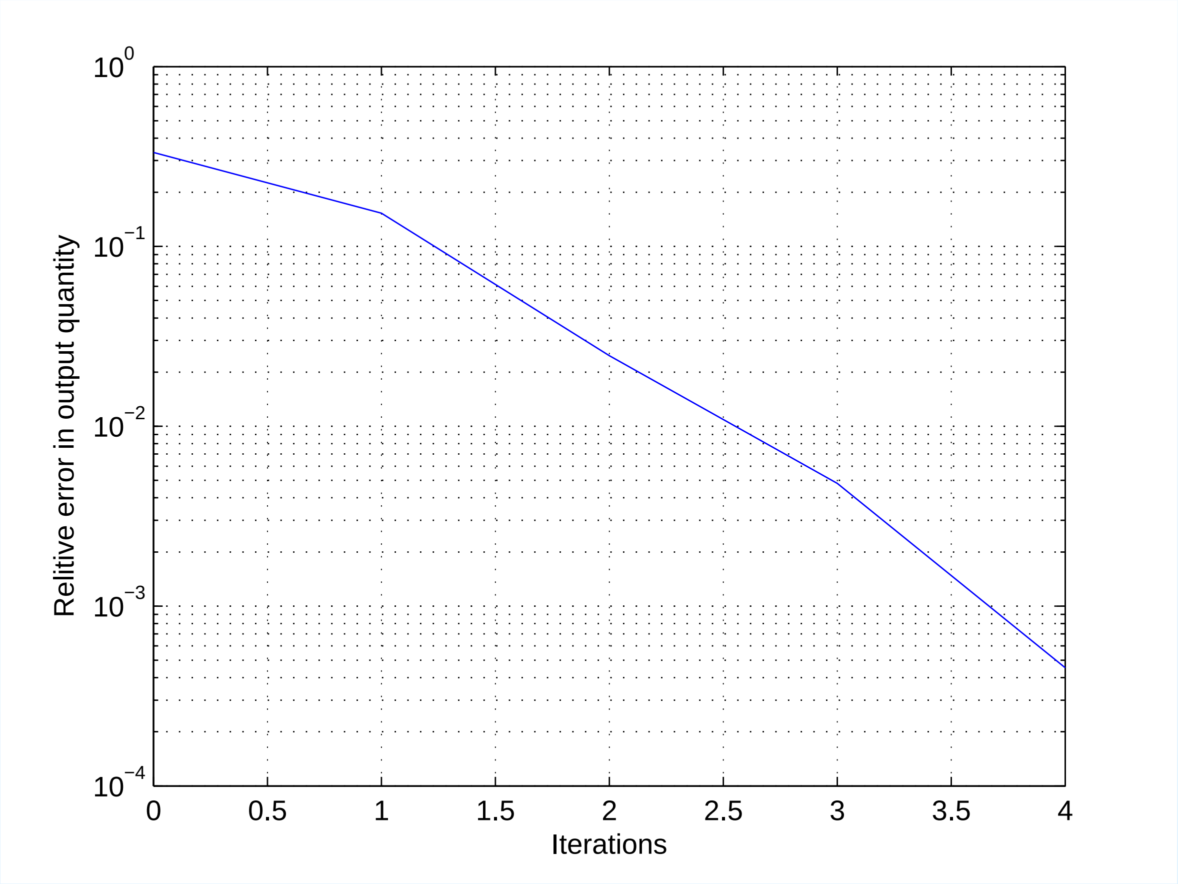
We let $\psi = 1$ and use a refinement level of 40%, $h = 1/96$, $H = 1/24$.



Refinements and Patchsizes chosen by the adaptive algorithm.

Numerical Examples

We plot the relative error compared to a reference solution in the quantity of interest (e, ψ). We solve the dual and the primal using the same method.



Future work

- Use more then two scales and consider more extreme scale separation.
- Make an evaluation of how the method performs compared to other methods.
- Prove a priori error estimates for the multiscale method.
- Numerics in 3D.
- Adaptive patch shapes for local problems in the convection dominated problem.
- Consider more challenging equations, e.g. time dependent and non-linear.
- Other interesting applications.