Adaptive Variational Multiscale Methods

Axel Målqvist and Mats G. Larson

axel@math.colostate.edu

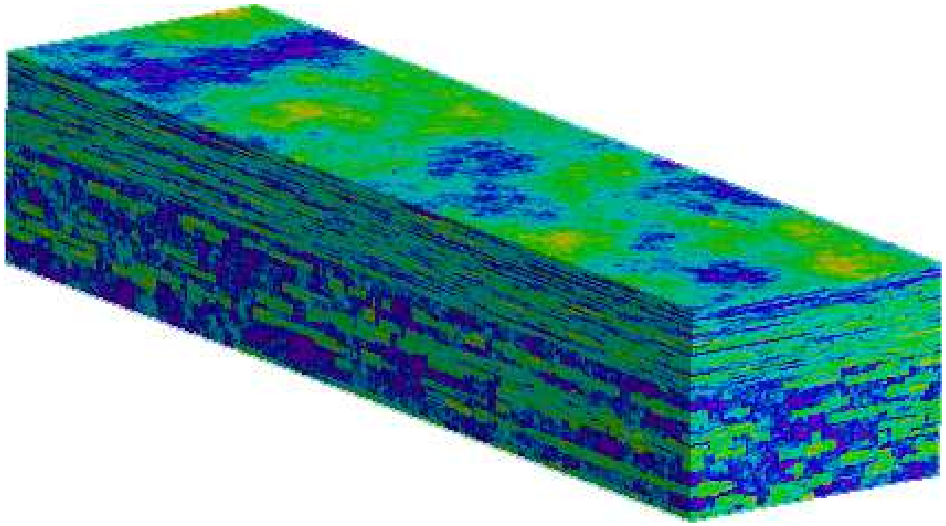
Department of Mathematics, Colorado State University

Outline

- A multiscale application
- The model problem
- A new multiscale method
- Implementation
- A posteriori error estimation
- Extension to mixed formulation
- Numerical examples

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.

An application

We seek the water concentration c that solves the system,

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

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

Main results of my PhD thesis

- *Develop a new multiscale method for solving elliptic problems (with fine scale features) where error estimation and adaptivity is an integrated part of the method.*
- Develop a framework for error estimation and adaptivity for multi-physics problems, and apply it to a coupled elliptic and transport problem.

The model problem

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

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

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 $\mathbf{R}^d$, $d = 1, 2, 3$.

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 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.

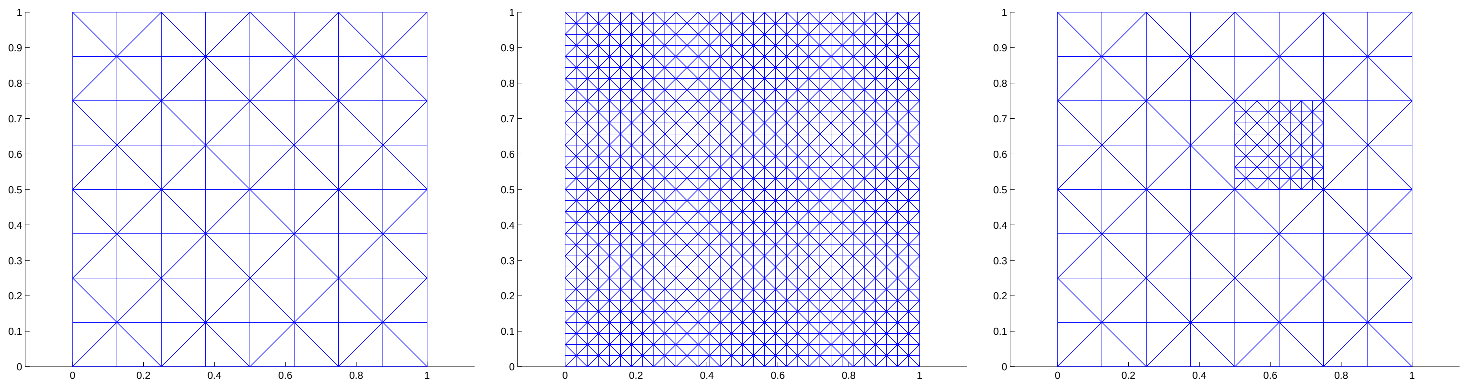


Figure 1: Coarse $H > \epsilon$ and fine $h < \epsilon$ mesh.

Since $\|\nabla e\| \sim \frac{h}{\epsilon}$, $\|\nabla e\| \sim H$ is reasonable.

Various multiscale methods

- Upscaling techniques: Durlofsky et al, Nielsen et al.
- Multiscale finite element method: Hou et al., Efendiev-Ginting, Aarnes-Lie
- Multiscale finite volume method: Jenny et al.
- Variational multiscale method: Hughes et al. , Arbogast, Larson-Målqvist

The variational multiscale method

Find $u_c \in V_c$ and $u_f \in V_f$, $V_c \bigoplus V_f = V$ 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 $R : V \rightarrow V'$, $(R(v), w) = l(w) - a(v, w)$, for all $v, w \in V$.

The variational multiscale method

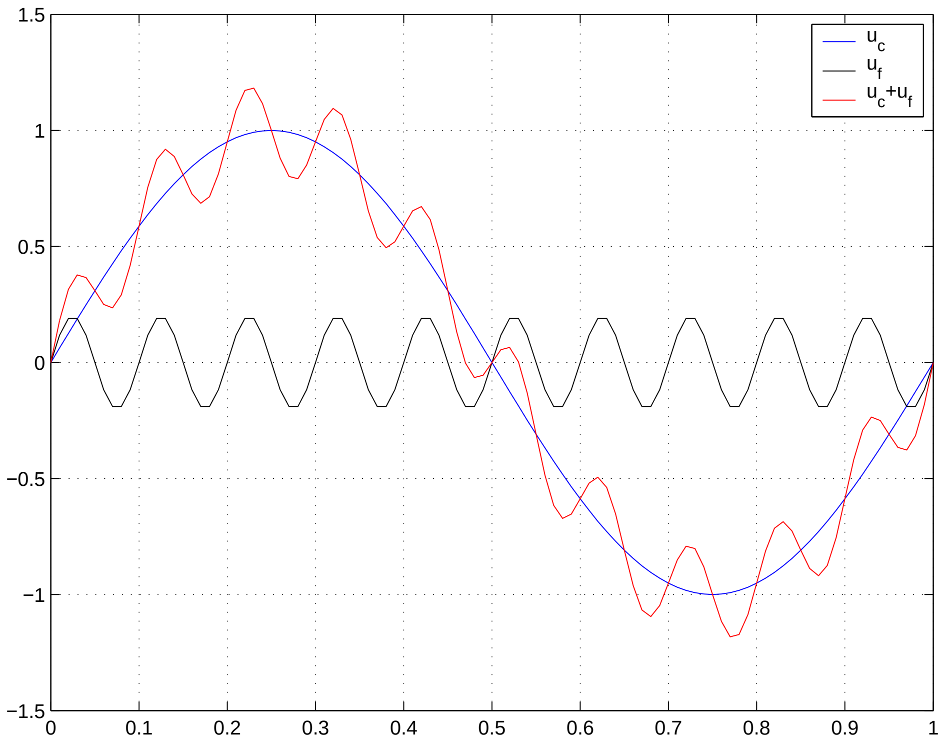


Figure 2: u_c, u_f, and $u_c + u_f$.

Approximation (Our version)

We derive the method in two steps.

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

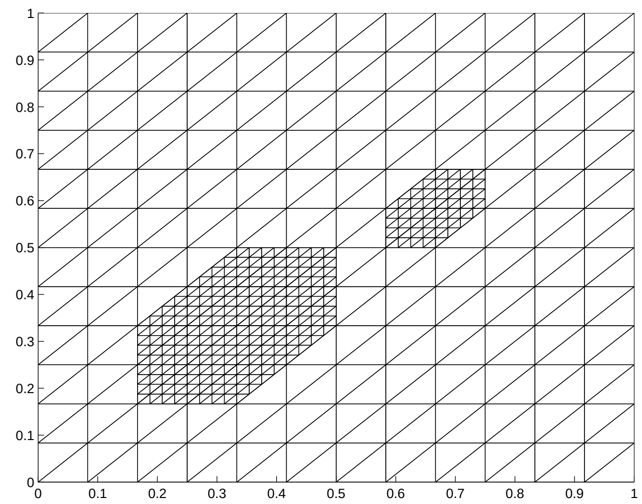
$$a(u_{f,i}, v_f) = (\varphi_i R(u_c), v_f) \quad \text{for all } v_f \in V_f.$$

- 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.

Our method

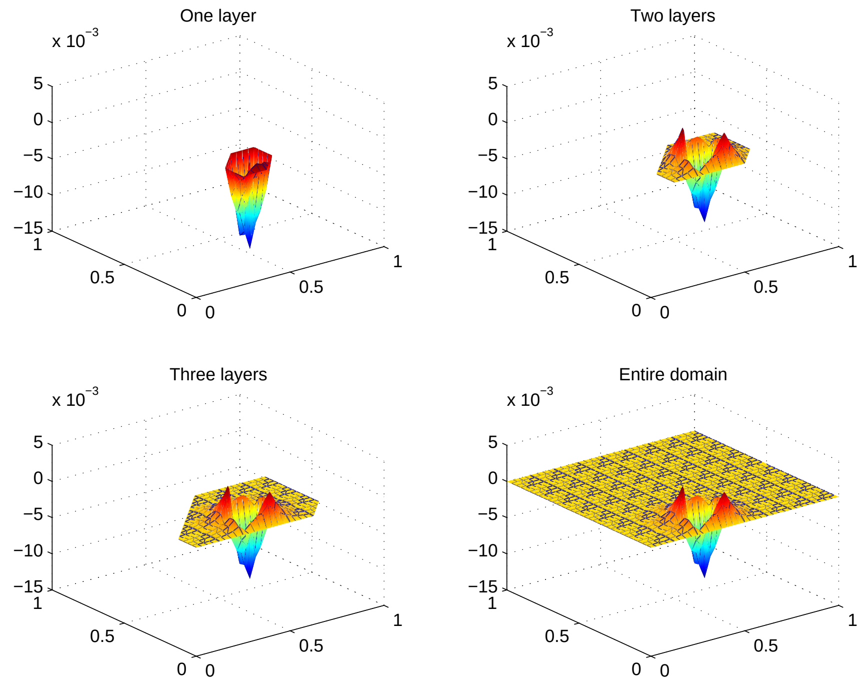
The resulting method reads: find $U_c \in V_c$ and $U_f = \sum_{i\in\mathcal{N}} U_{f,i}$ where $U_{f,i} \in V_f^h(\omega_i)$ such that

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

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 $\mathrm{supp}(\varphi_i) \subset \omega_i \subset \Omega$.

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, $\varphi_i R(U_c)$.
- 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.

Implementation: Iterative

Iterative $U_{f,i}^{0} = 0$,

$$(a\nabla U_c^k, \nabla v_c) = (f, v_c) - (a\nabla U_f^{k-1}, \nabla v_c),$$
$$(a\nabla U_{f,i}^k, \nabla v_f) = (R(U_c^k), \varphi_i v_f),$$

or in matrix form,

$$A_c U_c^k = b_c(U_f^{k-1})$$
$$\hat{A}_f U_{f,i}^k = b_f(U_c^k)$$

Implementation: Direct

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

$$a(U_{f,k}, v_f) = (f, v_f \phi_k) - a(U_c, v_f \phi_k)$$

for all $v_f \in V_f^h(\omega_k)$. Instead we solve: find $\chi_k^i, \eta_k \in V_f^h(\omega_k)$ such that

$$\begin{cases} a(\chi_k^i, v_f) = -a(\phi_i, v_f \phi_k) \\ a(\eta_k, v_f) = (f, v_f \phi_k). \end{cases}$$

for all $v_f \in V_f^h(\omega_k)$ and $supp(\phi_i) \cap supp(\phi_k) \neq \emptyset$.

Implementation: Direct

This means that: $\sum_{i\in\mathcal{N}} U_c^i \chi_k^i + \eta_k$ solves:

$$a(\sum_{i\in\mathcal{N}} U_c^i \chi_k^i + \eta_k, v_f) = (f, v_f\phi_k) - a(U_c, v_f\phi_k),$$

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

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

where $\chi^i = \sum_{k\in\mathcal{N}} \chi_k^i$ and $\eta = \sum_{k\in\mathcal{N}} \eta_k$.

Implementation: Direct

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

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

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

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

Implementation: Direct

This can now be written on matrix form as,

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

where,

$$\begin{cases} A^{ij} = a(\phi_i, \phi_j), \\ T_{ij} = a(\chi^i, \phi_j), \\ b_j = (f, \phi_j), \\ d_i = a(\eta, \phi_j). \end{cases}$$

Implementation: Direct

Implementing the method comes down to calculating T and d locally, $T = \sum_{k \in \mathcal{N}} T^k$ and $d = \sum_{k \in \mathcal{N}} d^k$.

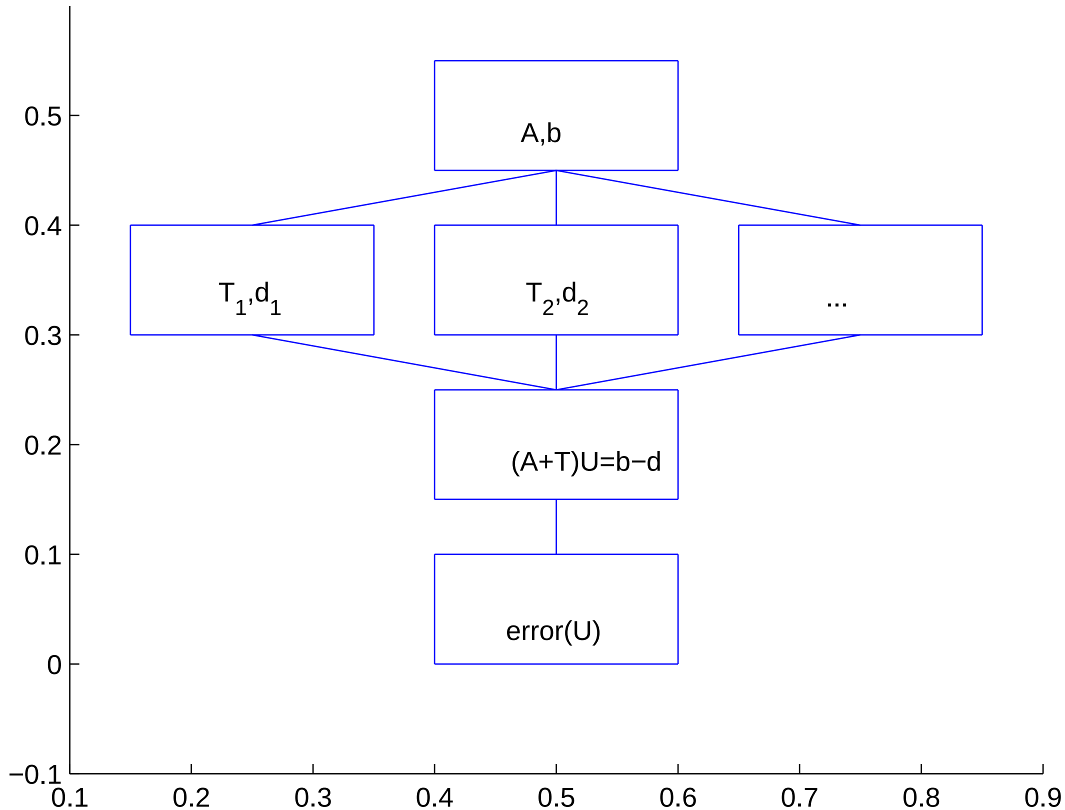
$$T_{ij}^k = a(\chi_k^i, \phi_j),$$

and

$$d_j^k = a(\eta_k, \phi_j).$$

These can be computed on the patches without knowing U_c.

Algorithm



A,b
T1,d1
T2,d2
...
(A+T)U=b−d
error(U)
0.5
0.4
0.3
0.2
0.1
0
−0.1
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9

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 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

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

Energy Norm Estimate

$$
\begin{aligned}
\|e\|_a^2 &= a(e,e) = a(e, e-\pi_c e) \\
&= (f, e-\pi_c e) - a(U_c, e-\pi_c e) - a(U_f, e-\pi_c e) \\
&= \sum_{i\in\mathcal{N}} (f, \phi_i(e-\pi_c e)) - a(U_c, \phi_i(e-\pi_c e)) \\
&\qquad - a(U_{f,i}, e-\pi_c e)
\end{aligned}
$$

Remember

$$a(U_{f,i}, v_f) + a(U_c, \phi_i v_f) - (f, \phi_i v_f) = 0, \quad v_f \in V_f^h(\omega_i).$$

Energy Norm Estimate

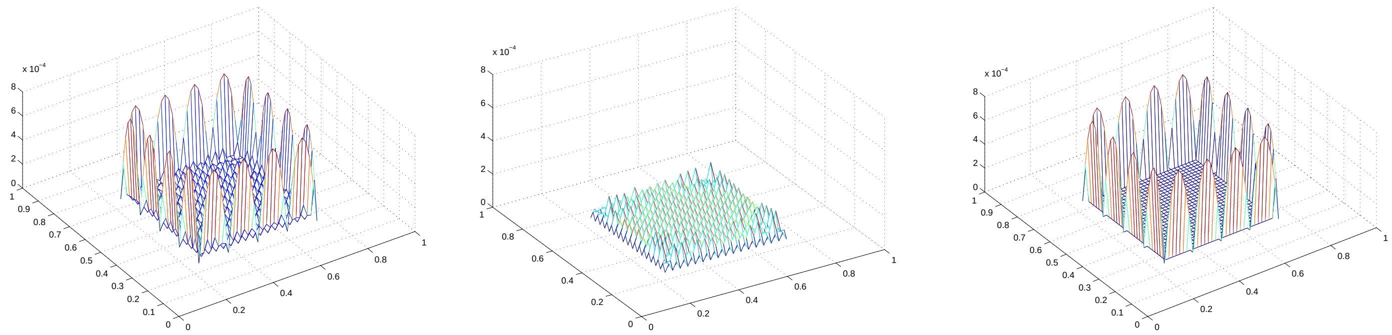


Figure 3: We study $z = e - \pi_c e$. We have $z - \pi_f^0 z$ to the left, $z - \pi_f z$ in the middle and $\pi_f z - \pi_f^0 z$ to the right.

Energy Norm Estimate

We work with these two parts separately. We start with $z - \pi_f z$ where $z = e - \pi_c e$,

$$\sum_{i\in\mathcal{N}} (f, \phi_i(z - \pi_f z)) - a(U_c, \phi_i(z - \pi_f z))$$

$$- a(U_{f,i}, z - \pi_f z)$$

$$\leq \left(\sum_{i\in\mathcal{N}} C_a \| h\mathcal{R}_i(U_{f,i}) \|_{\omega_i}^2 \right)^{1/2} \|e\|_a.$$

Energy Norm Estimate

Next we study the boundary part $\pi_f z - \pi_f^0 z$,

$$\sum_{i\in\mathcal{N}} (f, \phi_i(\pi_f z - \pi_f^0 z)) - a(U_c, \phi_i(\pi_f z - \pi_f^0 z))$$

$$- a(U_{f,i}, \pi_f z - \pi_f^0 z)$$

$$\leq \left(\sum_{i\in\mathcal{N}} C_a \| \sqrt{H} \Sigma(U_{f,i}) \|^2_{\partial\omega_i} \right)^{1/2} \|e\|_a.$$

$\Sigma(U_{f,i})$ is a variational approximation of $a\partial_n U_{f,i}$.

Explaining Σ.

We have,

$$(f, v_f \phi_i) - a(U_c, v_f \phi_i) - a(U_{f,i}, v_f) = 0,$$

for all $v_f \in V_f^h(\omega_i)$. Let "$V_{\partial\omega_i} = V_f^h(\bar{\omega}_i)|_{\partial\omega_i}$" and let, $\Sigma \in V_{\partial\omega_i}$ solve,

$$(\Sigma, v)_{\partial\omega_i} = (f, v\phi_i) - a(U_c, v\phi_i) - a(U_{f,i}, v),$$

for all $v \in V_f^h(\bar{\omega}_i)$ equal to zero on interior nodes, This can be written as,

$$M_{\partial\omega_i}\Sigma = b,$$

Why is $\Sigma \approx a\partial_n U_{f,i}$?

Compare

$$-\nabla \cdot a\nabla u = f,$$

$$a(u,v)-(f,v) = a(u,v)+(\nabla \cdot a\nabla u, v) = (a\partial_n u, v)_\Gamma.$$

and,

$$(\Sigma, v)_{\partial\omega_i} = (f, v\phi_i) - a(U_c, v\phi_i) - a(U_{f,i}, v).$$

Note that $a(U_c, v_f\phi_i) = 0$ for these functions v if $L > 1$,

Back to the estimate

$$\sum_{i\in\mathcal{N}}(f,\varphi_i(\pi_f z-\pi_f^0 z))-a(U_c,\varphi_i(\pi_f z-\pi_f^0 z))$$

$$-a(U_{f,i},\pi_f z-\pi_f^0 z)=\sum_{i\in\mathcal{N}}(\Sigma(U_{f,i}),\pi_f z)_{\partial\omega_i}$$

$$(\Sigma(U_{f,i}),\pi_f z)_{\partial\omega_i}\leq C\|\sqrt{H}\Sigma(U_{f,i})\|_{\partial\omega_i}\|\frac{1}{\sqrt{H}}\pi_f z\|_{\partial\omega_i},$$

remember $z=e-\pi_c e$,

$$\|\pi_f z\|^2_{\partial\omega_i}\leq H^{-1}\|z\|^2_{\omega_i}+H\|\nabla z\|^2_{\omega_i}\leq H\|\nabla e\|^2_{\omega_i}.$$

Energy Norm Estimate

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

If we just solve local problems on part of the domain $\mathcal{F}$ we get,

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

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 on $\partial\omega_i$.

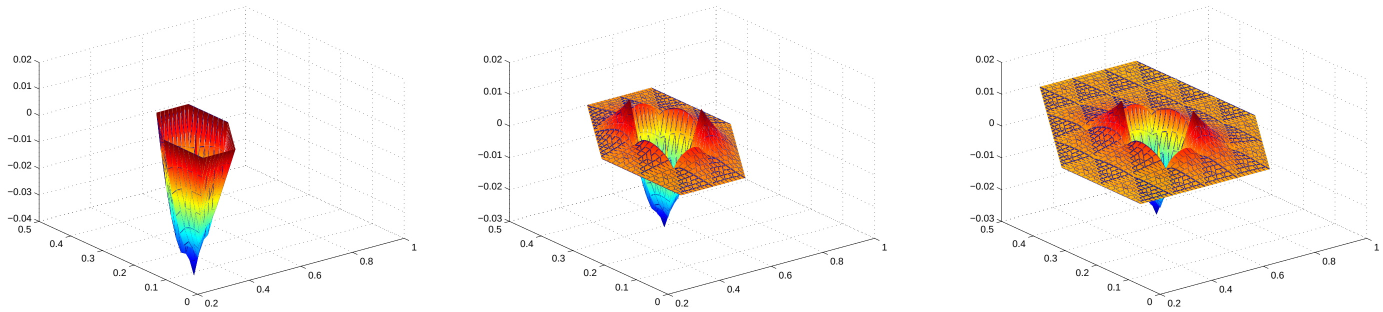


Figure 4: One, two, and three layer stars.

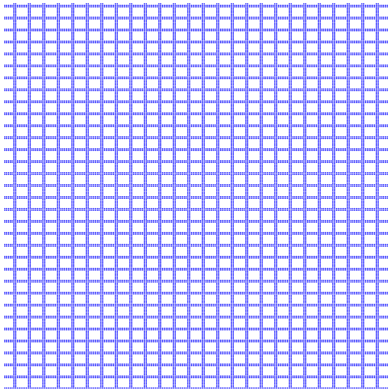
Adaptive Algorithm

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

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

Adaptive Algorithm

Example using the Adaptive Algorithm,



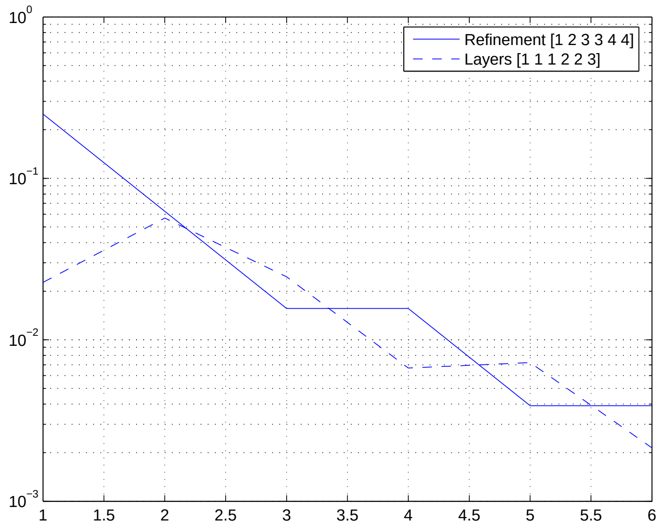


Figure 5: Periodic a $(0.05$ blue 1 white) with $\epsilon = H$ and 129×129 coarse nodes.

Linear Functional Error (e, ψ)

We introduce the dual problem: find ϕ such that

$$a(v, \phi_c) + a(v, \phi_f) = (v, \psi), \quad \text{for all } v \in V.$$

From the orthogonality showed earlier on we have $a(e, \phi_c) = 0$. We get

$$\begin{aligned}(e, \psi) = a(e, \phi) = a(e, \phi_f) = (f, \phi_f) - a(U, \phi_f) \\ = \sum_{i \in \mathcal{C}} (f, \varphi_i \phi_f) - a(U_c, \varphi_i \phi_f) \\ + \sum_{i \in \mathcal{F}} (f, \varphi_i \phi_f) - a(U_c, \varphi_i \phi_f) - a(U_{f,i}, \phi_f).\end{aligned}$$

Linear Functional Error (e, ψ)

We need to calculate an approximation of ϕ_f numerically.

- If possible ϕ_f can be calculated by a global calculation on a mesh where $h < h_\phi < H$.
- Or ϕ_f can be calculated by AVMS. We need to keep track on neighboring patches to form ϕ_f locally.

Linear Functional Error (e, ψ)

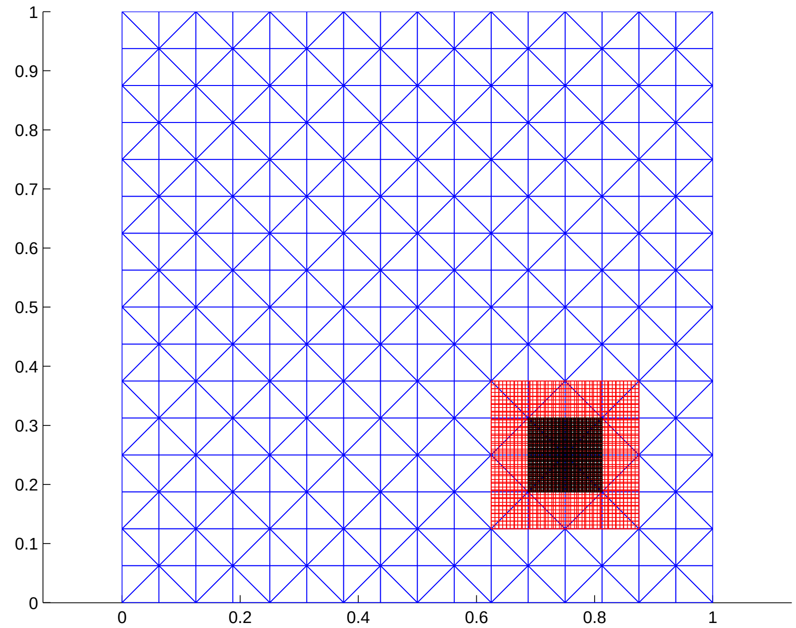


Figure 6: The dual solution on the red patches affects the error calculated on the black patch.

Linear Functional Error (e, ψ)

We consider an example where we seek a very accurate solution in part of the domain.

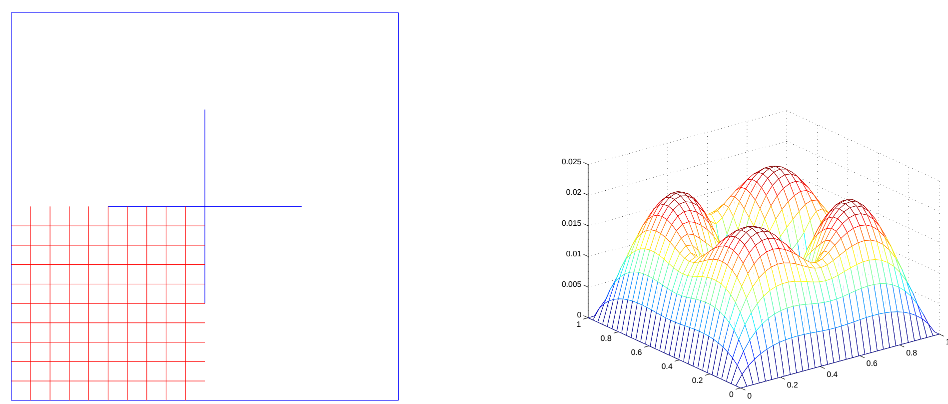


Figure 7: Geometry to the left with region of interest marked $\psi = I_{\{0 \le x, y \le 0.5\}}$. Solution to the right, $a = f = 1$.

Linear Functional Error (e, ψ)

We solve the dual problem with $\psi = I_{\{0 \le x, y \le 0.5\}}$.

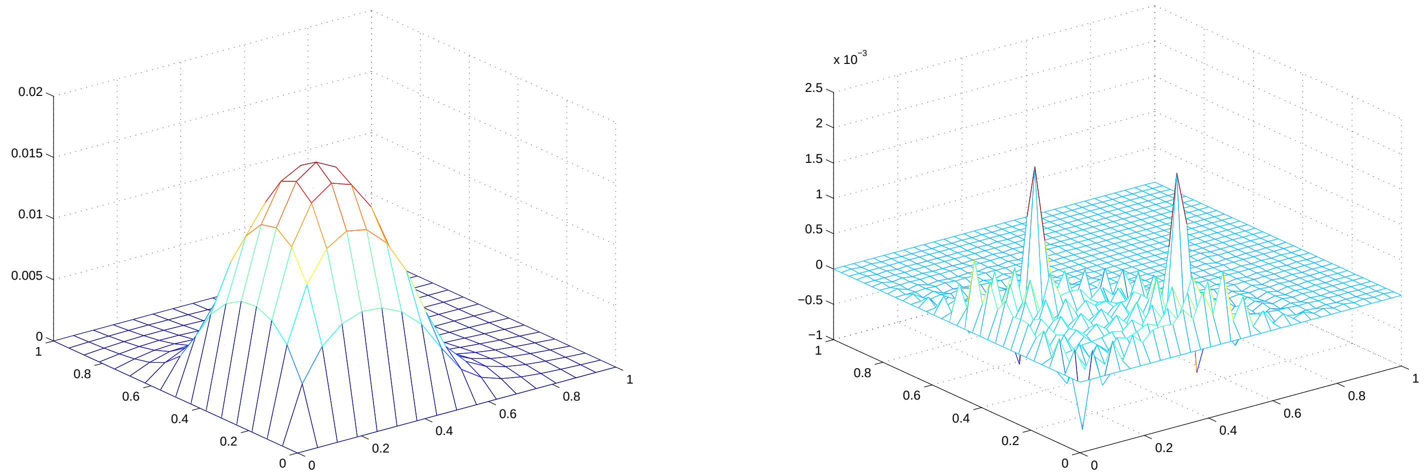


Figure 8: Dual solution ϕ to the left and ϕ_f to the right.

Adaptive algorithm

We remember the error estimate,

$$(e,\psi) = \sum_{i\in\mathcal{C}} (f,\varphi_i\phi_f) - a(U_c,\varphi_i\phi_f)$$
$$+ \sum_{i\in\mathcal{F}} (f,\varphi_i\phi_f) - a(U_c,\varphi_i\phi_f) - a(U_{f,i},\phi_f).$$

1. Start with all nodes in $\mathcal{C}$. Calculate U and ϕ_f.
2. Calculate error estimators, solve local problems or increase number of layers.
3. Stop if the error is small enough else go to 2.

Numerical example

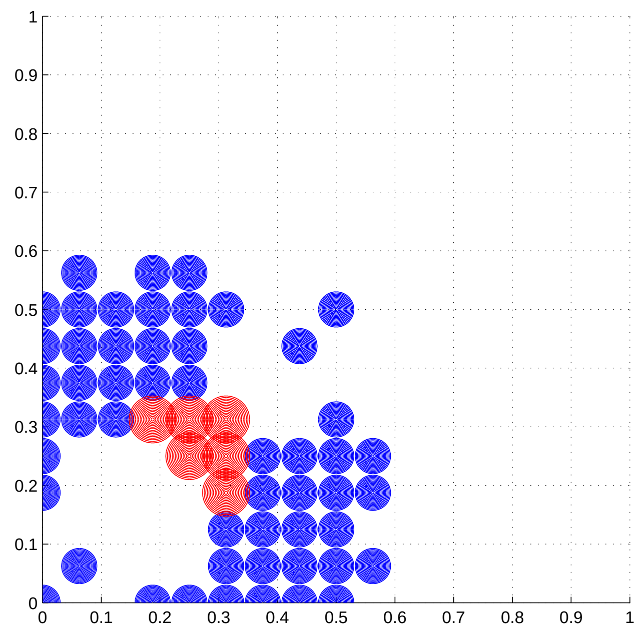


Figure 9: Local problems are solved using two and three layer stars in marked nodes.

Numerical example

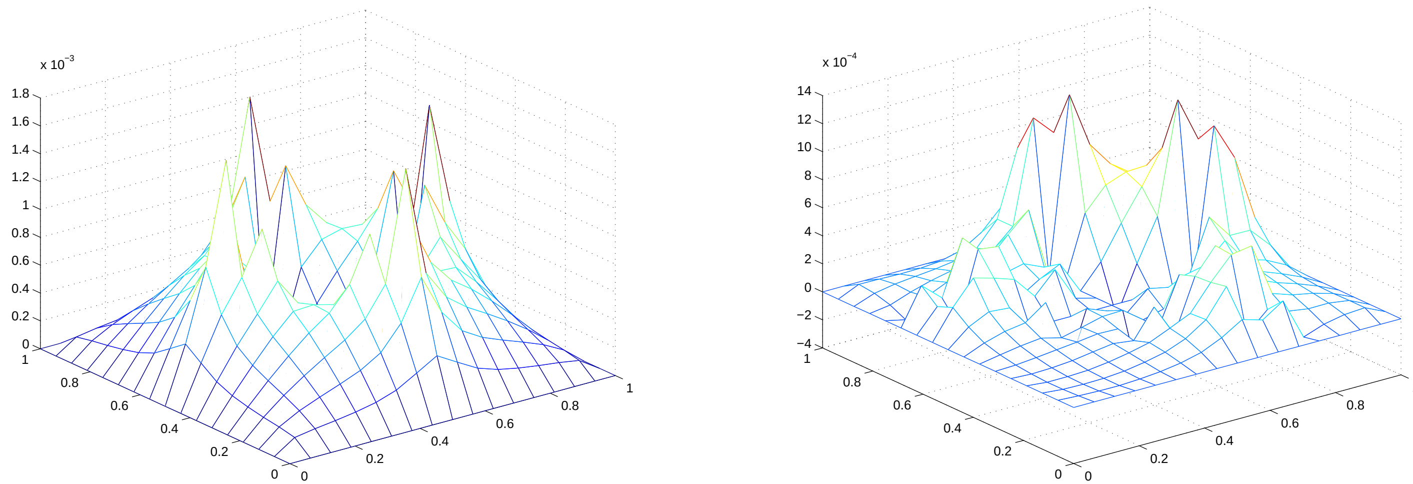


Figure 10: Error compared to reference mesh for standard Galerkin (left) and two iterations of AVMS as described above (right).

Standard A Posteriori Error Estimates

For duality based algorithms we know that:

- The dual solution ϕ need to be approximated but not in V_c.
- Regular refinement or higher order methods allocate lots of memory.

Instead we solve the dual problem by AVMS in each coarse node,

$$(e, \psi) = \sum_{i=1}^{n} (R(U), \Phi_{f,i}) + a(e, \phi_f - \Phi_f).$$

Standard A Posteriori Error Estimates

The second term can be estimated in the following way,

$$\begin{aligned} a(e, \phi_f - \Phi_f) &\leq \|e\|_a \, \|\phi_f - \Phi_f\|_a \\ &\leq \|e\|_a \, \|\phi - (\Phi_c + \Phi_f)\|_a . \end{aligned}$$

Both these terms can be estimated.

We can use standard Galerkin on the primal and AVMS on the dual and have control on the error committed by not solving the dual exactly.

Extension to a mixed setting

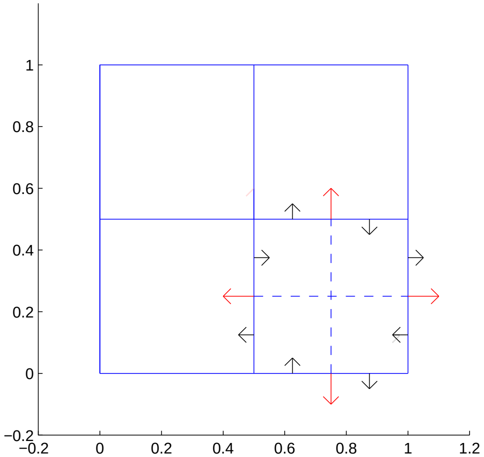
We have also extended this theory to the mixed formulation of the Poisson equation,

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

We use homogeneous Neumann boundary conditions.

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.



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 $w_f \in W_f$.

Coarse and Fine Scale solutions

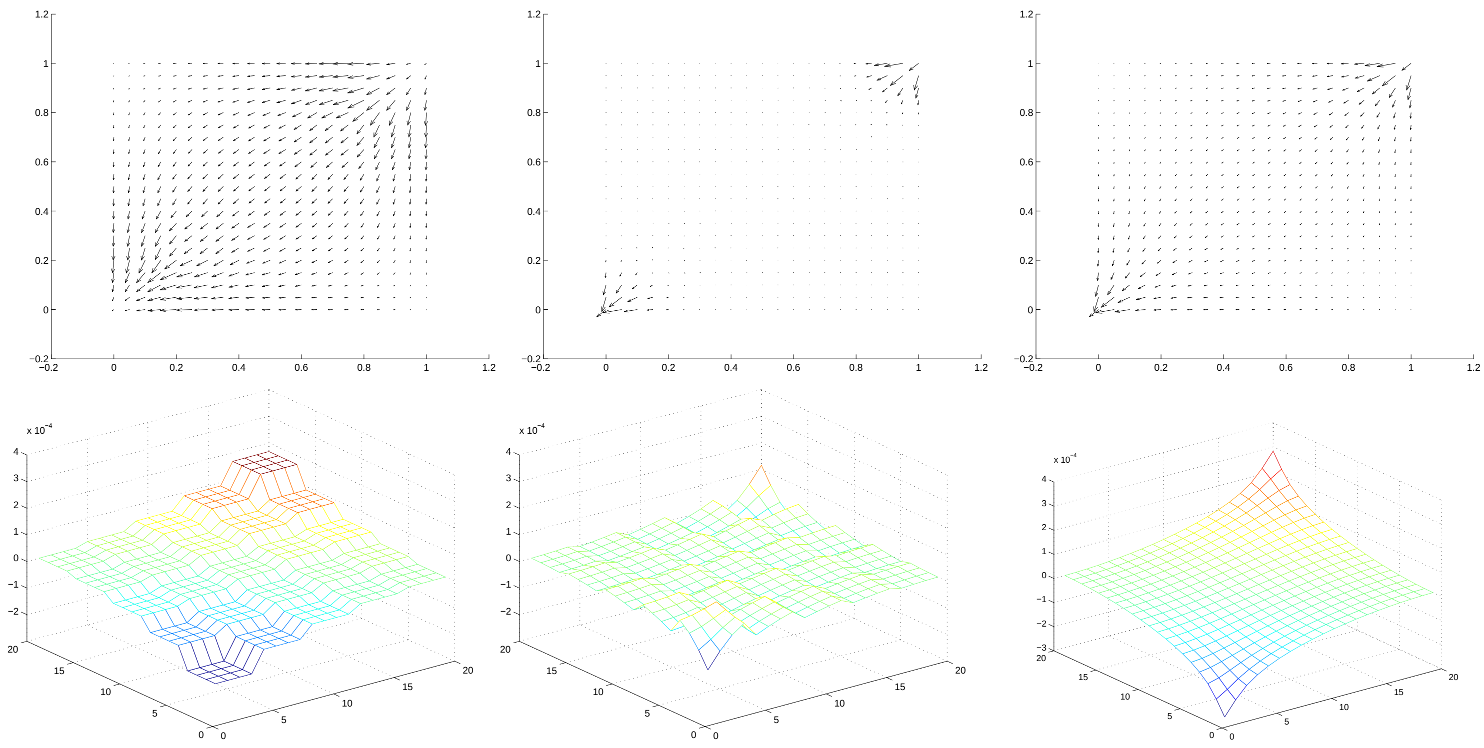


Figure 11: $\boldsymbol{\sigma}$ and u on coarse and fine scale.

Orthogonality

We use an hierarchical basis of Raviart-Thomas element for the flux and the piecewise constants for the pressure.

$$(w_c, \nabla \cdot \boldsymbol{v}_f) = \sum_K w_c^K \int_{\partial K} \boldsymbol{n} \cdot \boldsymbol{v}_f \, dx = 0$$

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 $w_f \in W_f$.

Partition of unity

If we let ϕ_i be a coarse Raviart-Thomas base function,

$$\boldsymbol{\phi}_i = \begin{bmatrix} \phi_i^x \\ \phi_i^y \end{bmatrix}, \qquad \phi_i = \begin{bmatrix} \phi_i^x & 0 \\ 0 & \phi_i^y \end{bmatrix},$$

will be a partition of unity,

$$\sum_{i \in \mathcal{N}} \begin{bmatrix} \phi_i^x & 0 \\ 0 & \phi_i^y \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

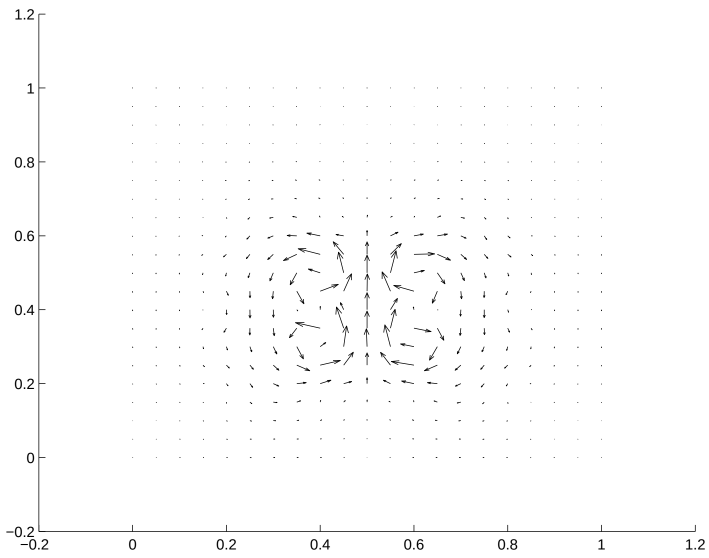
The multiscale method

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,

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

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

Local Solutions



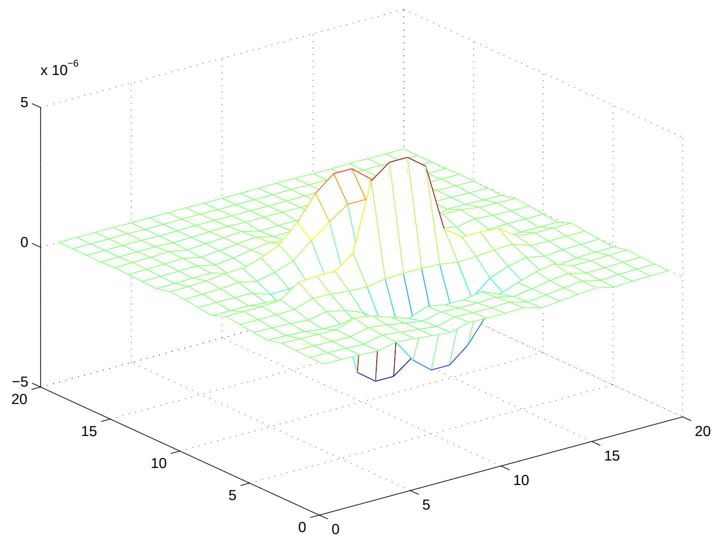


Figure 12: The local solutions $\boldsymbol{\sigma}_{f,i}$ and $u_{f,i}$

In this simple example $a = 1$.

Motivation for Introducing Patches

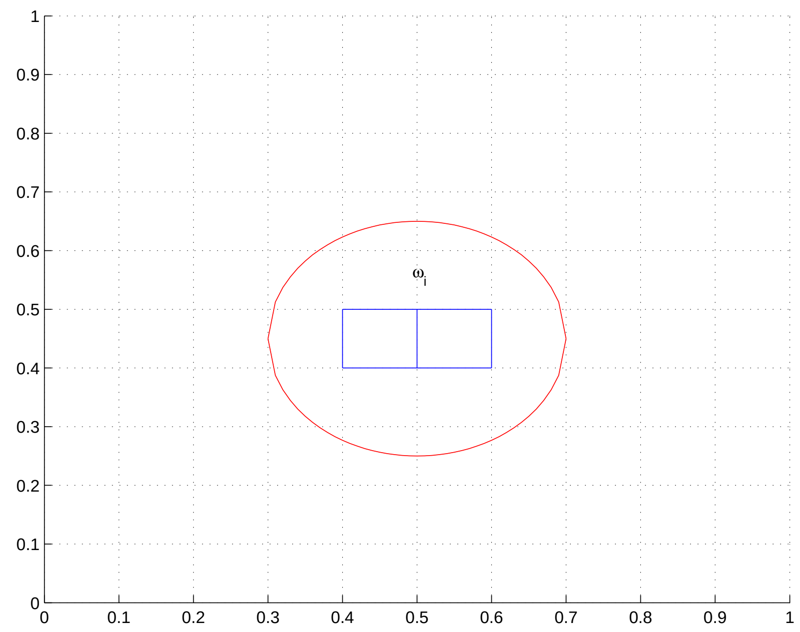
- The right hand side has support on $supp(\boldsymbol{\phi}_i) = supp(\psi_i)$.
- The equations are solved in a slice space where solutions decay rapidly.

$$\int_E \boldsymbol{n} \cdot \boldsymbol{\sigma}_{f,i} \, dx = 0$$

and

$$\int_K u_{f,i} \, dx = 0.$$

The Patch



The patch ω_i typically consists of coarse elements but could have any shape.

Solving Local Neumann Problems

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

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

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

Example of Local Solutions $U_{f,i}$

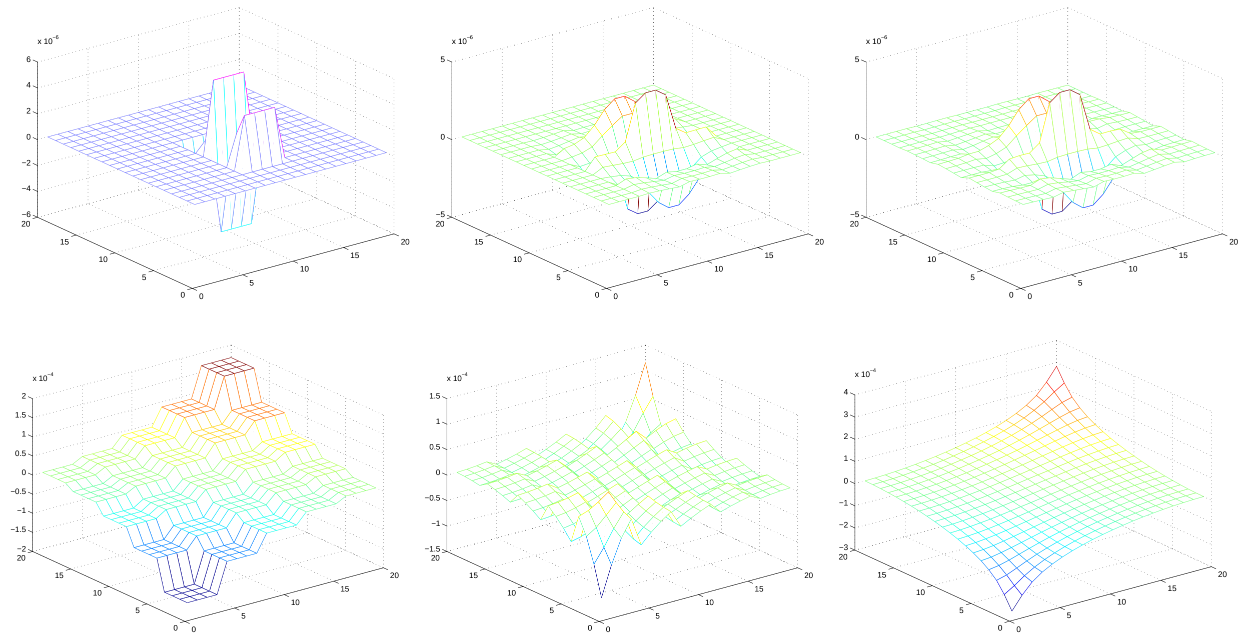


Figure 13: Above: 1, 2, and 3 layer patches, below: U_c, U_f, and, U using 3 layers.

Error estimation: Postprocessing of U

A posteriori error estimates in energy norm using RT or BDM elements. (Stenberg)

$$\|\boldsymbol{\sigma} - \boldsymbol{\Sigma}\|_0^2 \leq C \sum_{K \in \mathcal{K}} h_K^2 \|f + \nabla \cdot \boldsymbol{\Sigma}\|_{0,K}^2 + \|\boldsymbol{\Sigma} - \nabla U^*\|_{0,K}^2$$

$$+ C \sum_{K \in \mathcal{K}} h_K^{-1} \| \left[U^* \right] \|_{0,\partial K}^2,$$

where, $P_0 U_K^* = U_K$, and

$$(\nabla U^*, \nabla v)_K = (\boldsymbol{\Sigma}, \nabla v)_K \quad \text{for all } v \in (I - P_0) P^1(K).$$

Energy Norm Estimate $\|v\|_a^2 = (\frac{1}{a}v, v)$

Next we present an estimate of the error.

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

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

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

U^* is a post processed version of U.

Adaptive Strategy

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

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

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

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

Adaptive Strategy

- 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

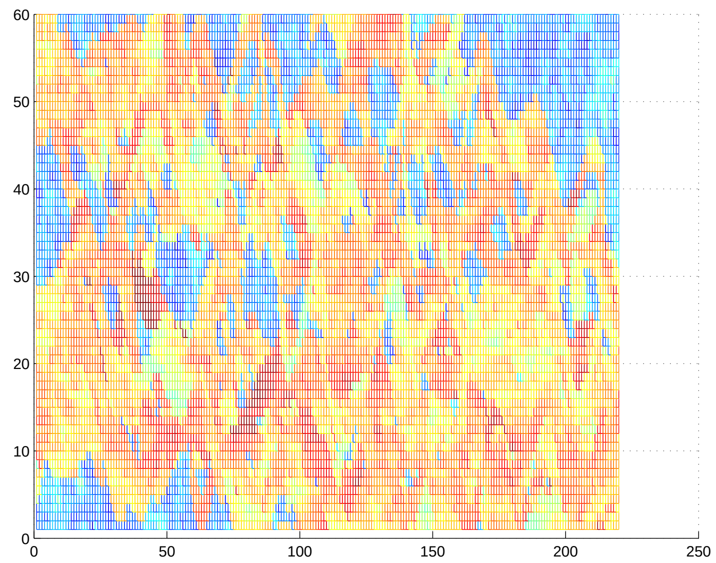
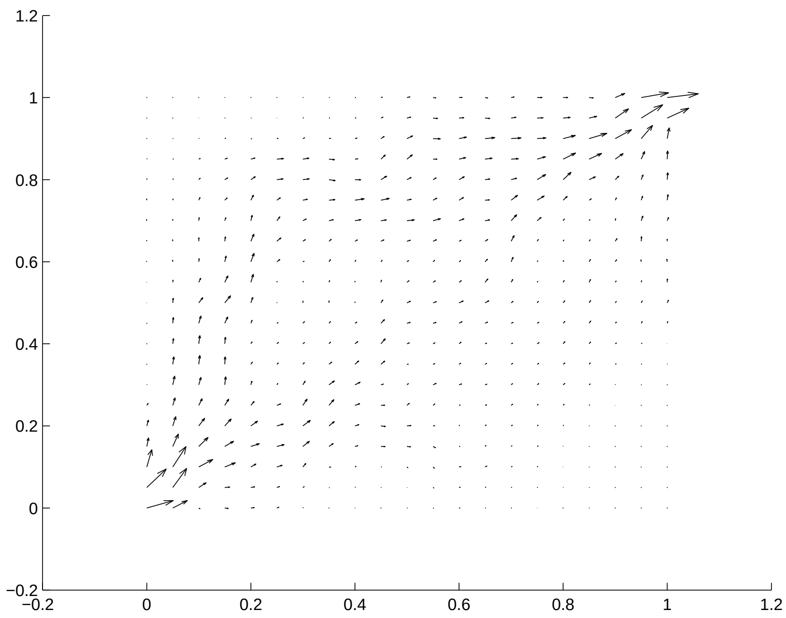


Figure 14: 2D slice of the permeability a (in log scale) taken from the tenth SPE comparative solution project.

Reference solutions



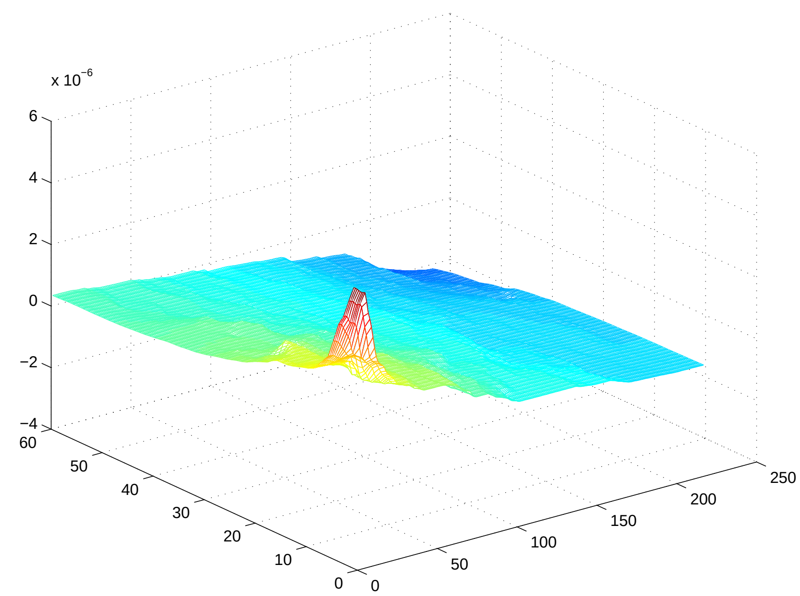


Figure 15: Above we see the reference solution, (left) flux $-\mathbf{\Sigma}$ and (right) pressure u.

Convergence

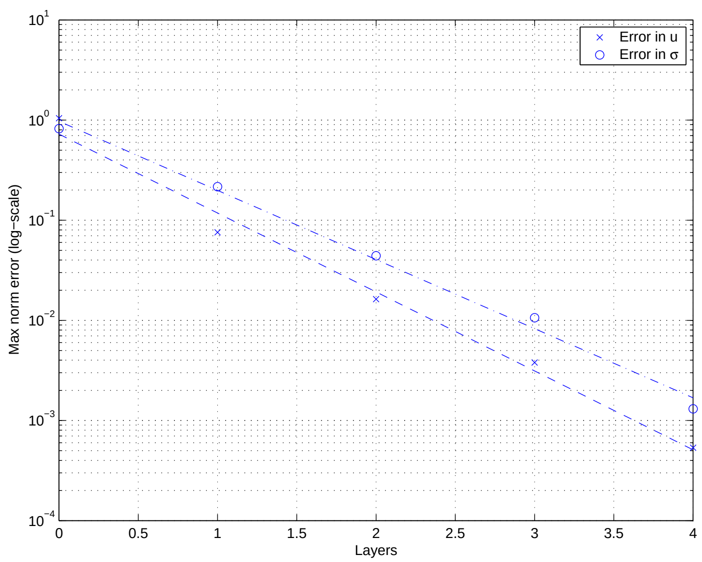


Figure 16: Max norm error (compared to reference solution) in log scale versus number of layers.

Example using the adaptive algorithm

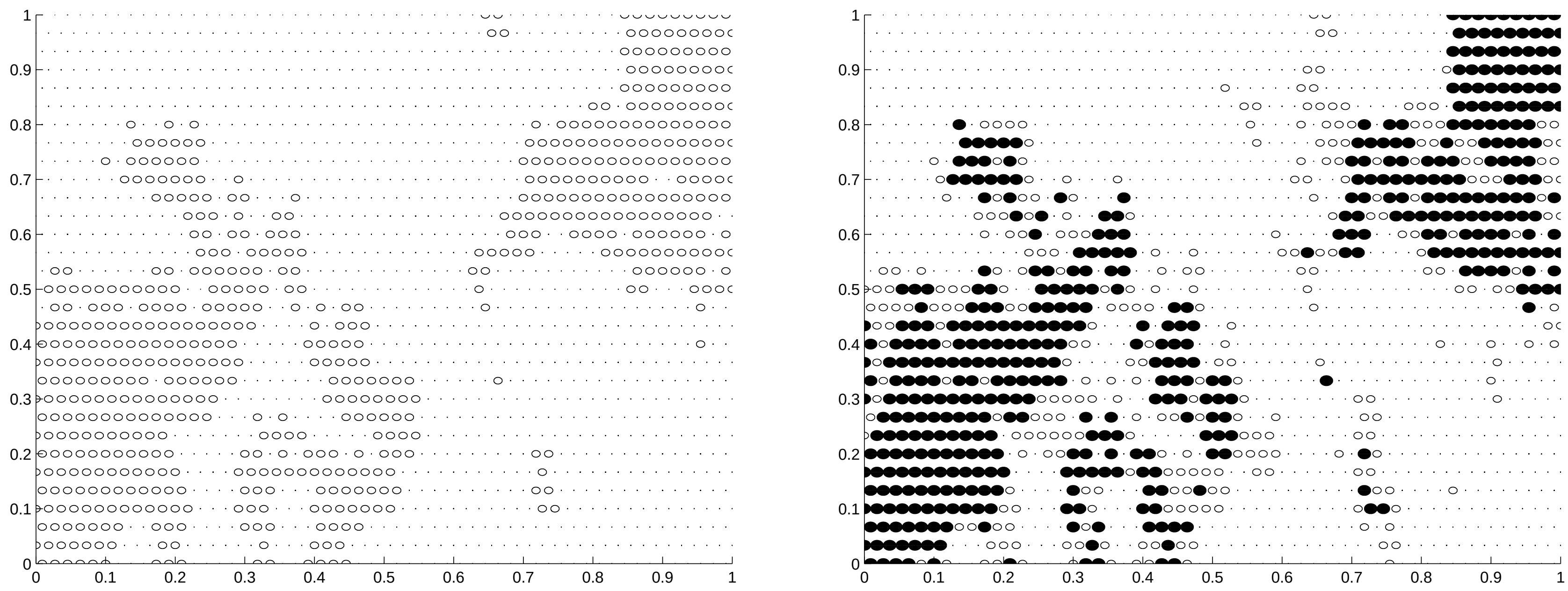
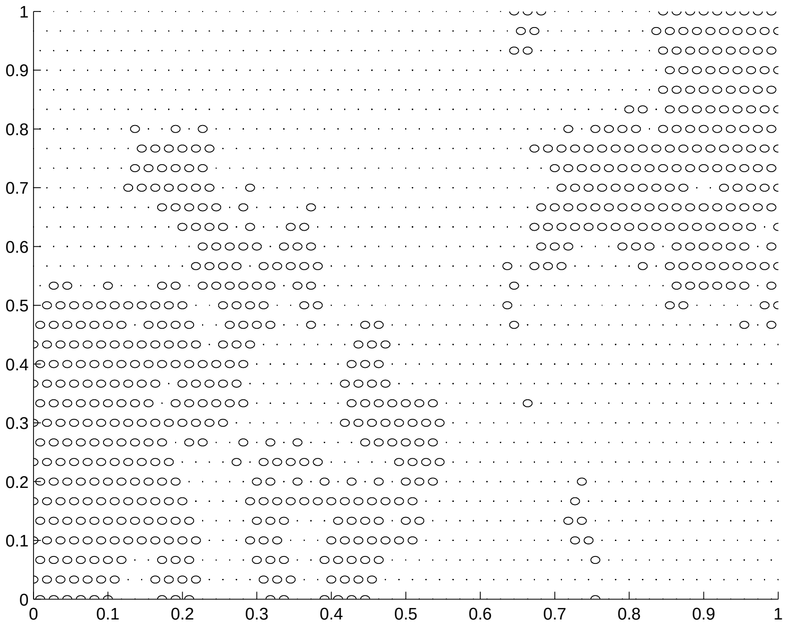


Figure 17: 35% of the patches increased in each iteration.

Example using the adaptive algorithm



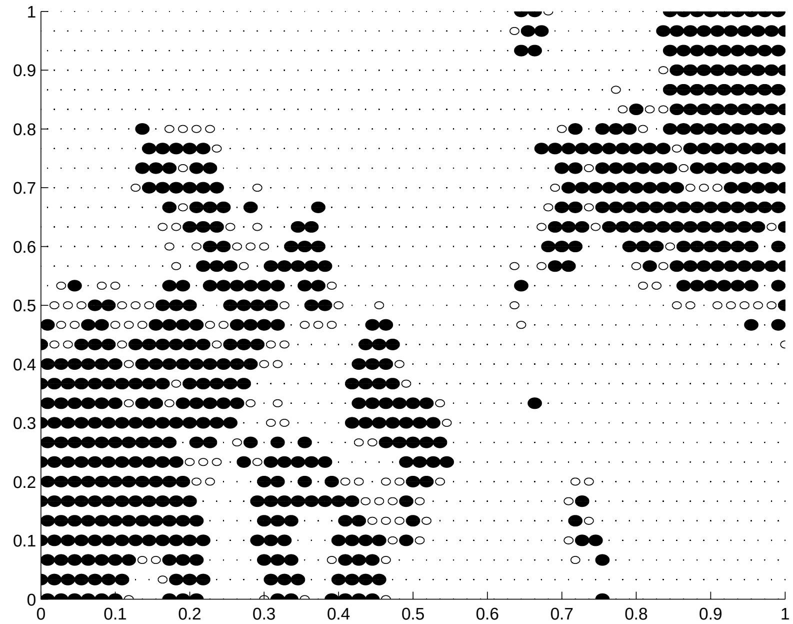


Figure 18: 35% of the fine scale meshes refined in each iteration.

Relative error in energy norm

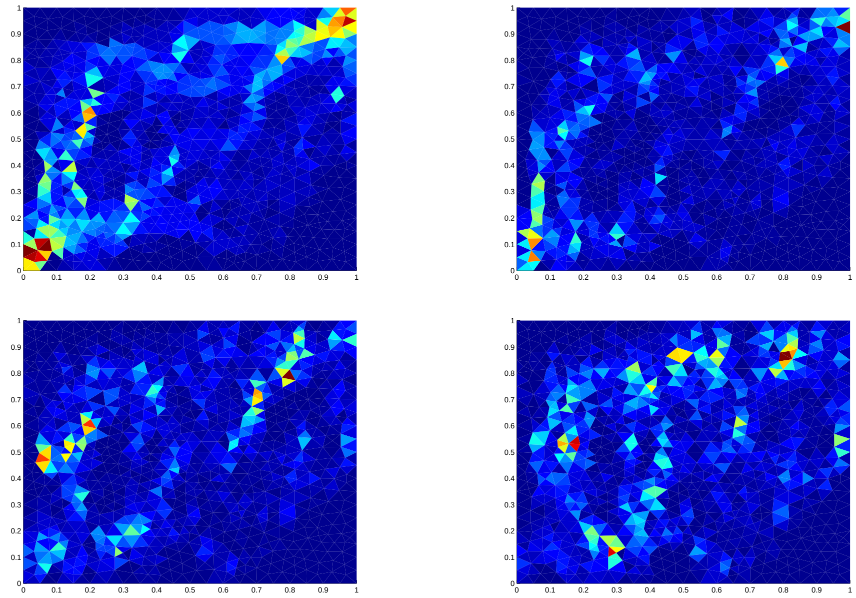


Figure 19: Relative error in energy norm: 106%, 16%, 10%, and 8%.

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.
- Extend the multiscale method to convection-diffusion equations and to time dependent problems. Extension to 3D.