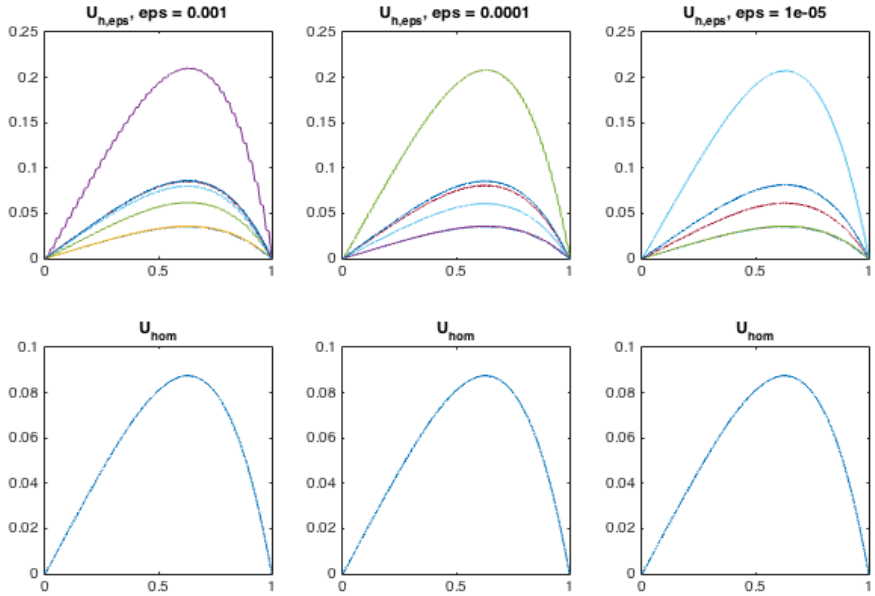
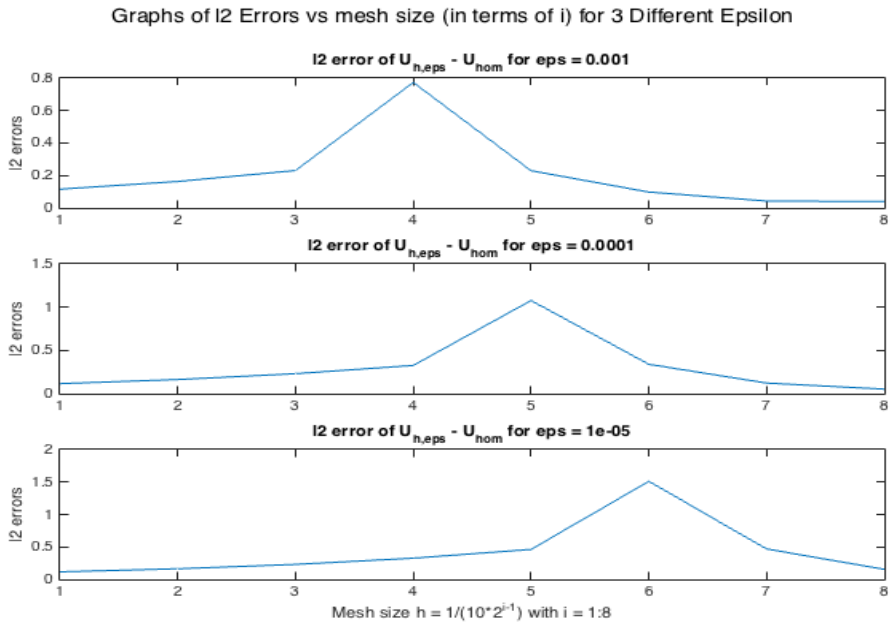
Dean Katsaros
Homework 4 Report

1. THe plots of the solutions to $U_{\epsilon,h}$ and U_{hom} for 3 different ϵ are included below. Below, the error plots for those 3 different ϵ are included as well. The l2 errors were calculated via taking the l2 norm of the difference between the solution from the finite difference scheme and the exact solution with homogenized coefficient at each of 8 iterations, giving the errors at 8 subsequently smaller mesh sizes.





From these graphs it appears that $\epsilon = 0.001$ gives the lowest error, but in all cases the errors tend to decrease as the mesh size gets larger, at least after a point.

2a) Shrinking the epsilon did not cause any slow down for the 2d finite difference using direct solver, as would be expected. Hence, *eps* was set to be 10^{-2}. Mesh sizes of N = 1024 and M = 1046529 took 5 seconds.

Mesh sizes of N = 2048 and M = 4190209 did not finish in a reasonable amount of time. The run was cancelled after about 8 minutes.

2 b) Using the 2D Finite Difference Solver with direct solve gave the following performance (highest 2 mesh sizes only reported). Tests were done with $a(x) = 1$ for simplicity.

N 512 M 261121
Build 0.201162
umfpack Solve 1.065037
L2 error direct 0.000016 Int 43.768346
N 1024 M 1046529
Build 0.918819
umfpack Solve 5.984899
L2 error direct 0.000004 Int 43.768346

Using the multigrid solver resulted in significantly slower performance. The finest meshes took about 5 minutes to finish. The L2 error was still low, getting as low as 0.00121 using multigrid solver.

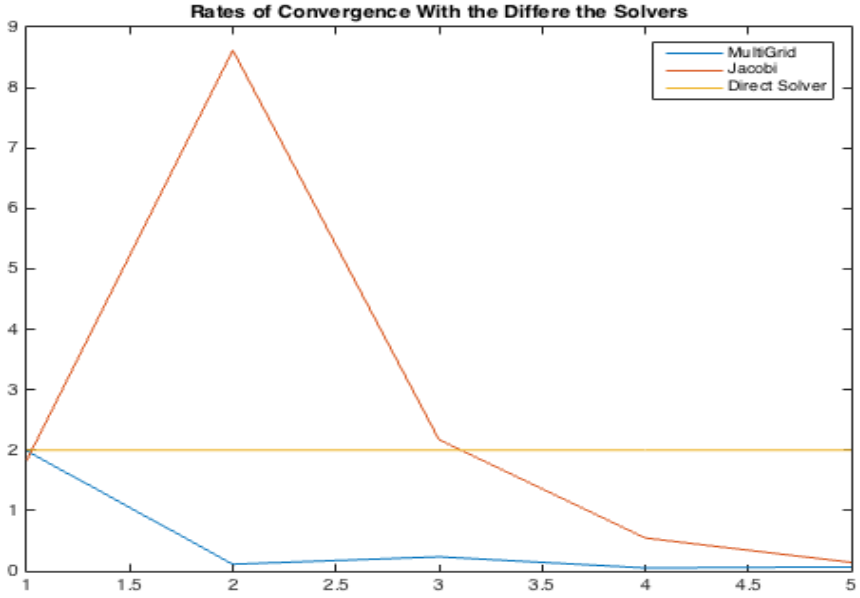
N 512 M 261121
Build 0.205040
umfpack Solve 73.769499
L2 error direct 0.001269 Int 43.768346
N 1024 M 1046529
Build 1.010152
umfpack Solve 298.716725
L2 error direct 0.001210 Int 43.768346

The Jacobi Solver resulted in significantly higher errors then the previous methods. The performance time was not as slow as multigrid, but significantly slower then the direct solve. Note: when plotting the rates of convergence, a LARGE outlier was ignored (see plot).

N 512 M 261121
Build 0.254777
umfpack Solve 49.924171
L2 error direct 37.778350 Int 43.768346
N 1024 M 1046529
Build 1.077271
umfpack Solve 202.715944
L2 error direct 41.580038 Int 43.768346

The rate of convergence was taken to be the last value plotted for each solver, the plot being of the log of error of the $i + 1th$ iteration divided by the error of the ith iteration (mesh size),

err^{i+1}/err^i for $i = 1:5$, then dividing by the log of the ratio of the mesh sizes at $i+1$ and i. That is, plot $r_i = \frac{\log(err^{i+1}/err^i)}{\log(2)}$ for $i = 1:5$.



The built matlab/direct solve gave a convergence rate of 2 whilst Jacobi and Multigrid presented a conergence rate of far less than 1. Consistent with the time data.

2c) and 2d) Numerically computing $\overline{A}$ returned the coefficient 0.9192 vs. the analytically computed coefficient 0.9298 using direct integration to calculate the harmonic average of A.

The elliptic PDE

$$-\nabla \cdot (A(x)\nabla u) = f$$

with $A(x) = 1 + a(x)x_1$ was then approached with $a(x)$ being the oscillatory function $a(x, x/\epsilon) = 1.1 + \sin(2\pi x_1)\sin\left(\frac{2\pi x_1}{\epsilon}\right)$ and $f = x^2$. This becomes an example of equation 12.2.1a:

$$(\dagger) \quad -\nabla \cdot (A^\epsilon(x)\nabla u^\epsilon) = f$$

for x in a domain Ω, and $u^\epsilon = 0$ on the boundary of Ω.

The Cell problem was solved Numerically, as mentioned above, to find the aforementioned homogenized coefficient which was substituted for $a(x)$. This is rationalized by Result 12.1 which says that the solution of $(\dagger)$ is approximately equal to the solution of the elliptic PDE with A^ϵ substituted for $\bar{A}$, where in our case $\bar{A}$ is $1 + \overline{a(x, x/\epsilon)}x_1$.

Precisely, the solution of

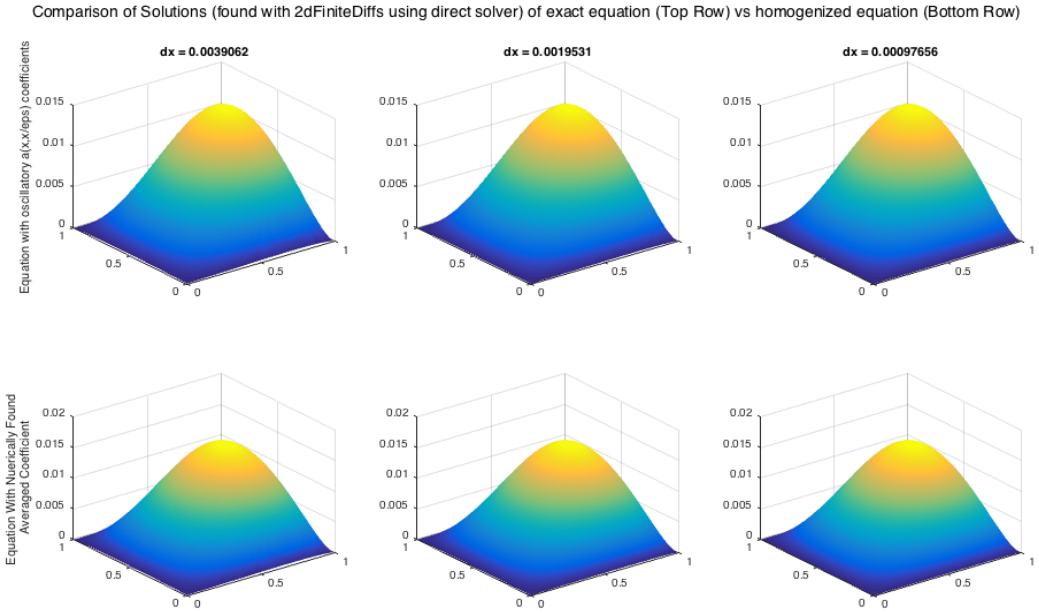
$$-\nabla \cdot ((1 + a(x, x/\epsilon)x_1)\nabla u) = x^2$$

using 2d finite differences was compared to to the solution of

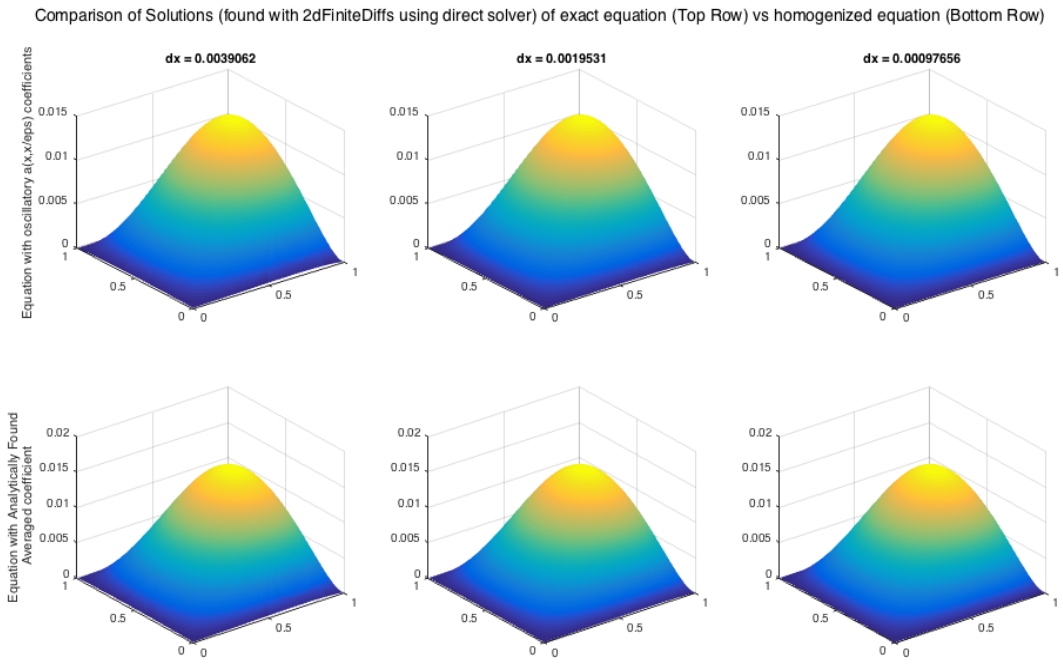
$$-\nabla \cdot ((1 + \overline{A}x_1)\nabla u) = -\nabla \cdot (1 + 0.9192x_1)\nabla u) = x^2$$

using 2d finite differences.

The following plot summarizes the findings, where surfaces for the 3 finest meshes are shown only. ϵ was taken to be 0.01 for all calculations for parts 2c) and 2d).

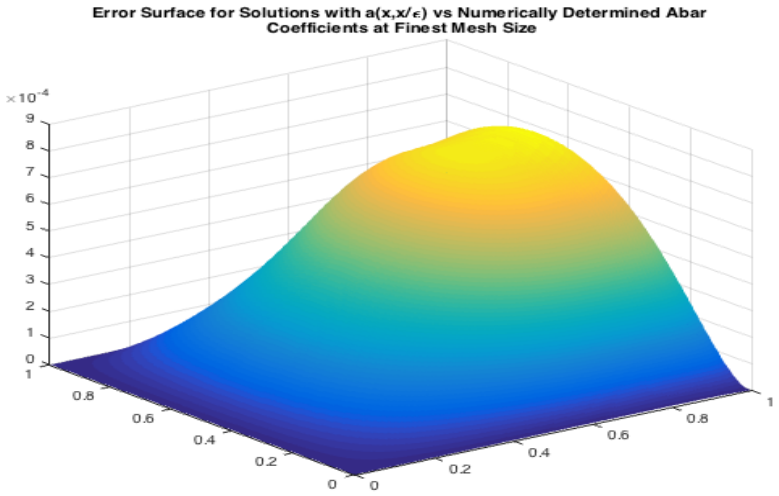


There is some error, as can be seen by the plotting scales. However, the error isn't enormous. The following comparison plot was created using the anlaytically determined coefficient, with all other parameters held the same.



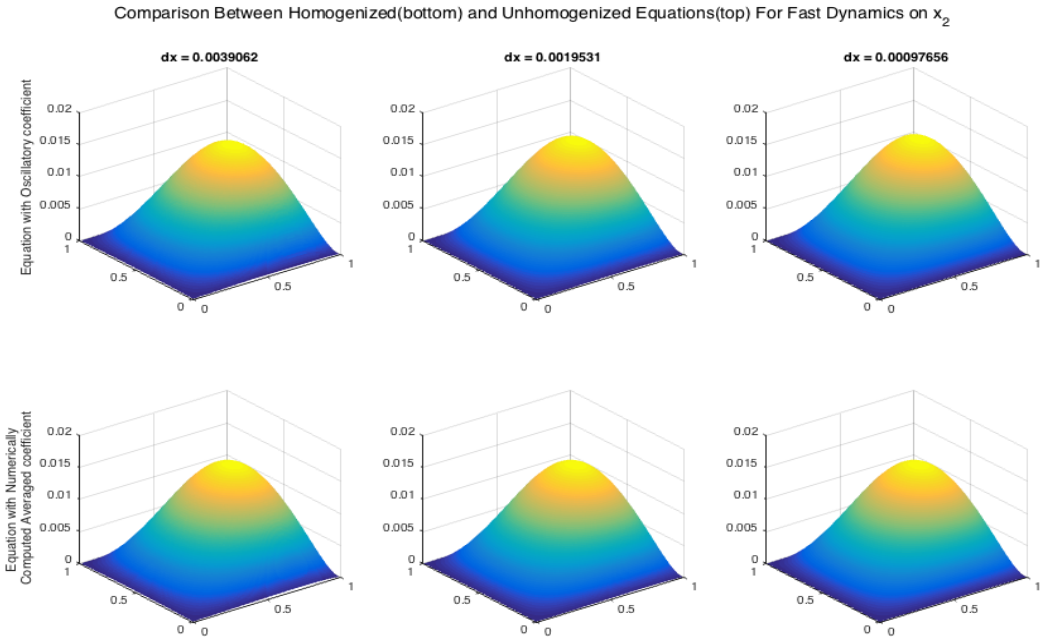
Comparing these two studies supports the idea that the differences between the homogenized and un-homogenized solutions are at least partially a result of the approximation introduced by homogenization itself, and not entirely the error associated with numerically finding the homogenized coefficient.

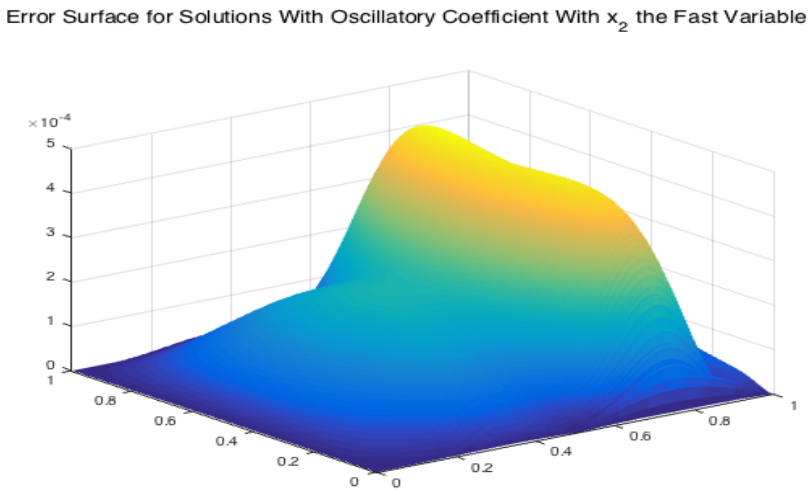
The following is an error surface for the homogenized (with numerically determined coefficient) solution surface and the solution surface with the exact coefficient.



This shows that the errors were small, but not zero.

The comparison plots and error surface plot was redone with the oscillatory coeffcient being moved to x_2. That is, $a(x, x/\epsilon)$ was set to $1.1 + \sin(2\pi x_1)\sin\left(\frac{2\pi x_2}{\epsilon}\right)$.





The errors were lower, so the approximation is better, but the errors weren't significantly lower. Ultimately, both these examples showed that Result 12.1 is indeed an approximation.