Math 226C Project 2

Theodore Kwan
tmkwan@uci.edu

May 24, 2016

Contents

1 Project Description

Consider the system partial differential equations on a domain Ω:

$$\begin{cases} -\Delta \mathbf{u} + \nabla p = \mathbf{f}, & (x,y) \in \Omega \\ \operatorname{div}(\mathbf{u}) = 0, & (x,y) \in \Omega \\ \mathbf{u} = \mathbf{g}, & (x,y) \in \partial\Omega \end{cases} \tag{1}$$

This system of partial differential equations can be written componentwise:

$$\begin{cases} -\Delta u + \frac{\partial p}{\partial x} = f_1(x,y), & (x,y) \in \Omega \\ -\Delta v + \frac{\partial p}{\partial y} = f_2(x,y), & (x,y) \in \Omega \\ \operatorname{div}(\mathbf{u}) = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0, & (x,y) \in \Omega \\ u(x,y) = g_1(x,y), & (x,y) \in \partial\Omega \\ v(x,y) = g_2(x,y), & (x,y) \in \partial\Omega \end{cases} \tag{2}$$

1.1 Problem 1

The first problem that will be analyzed on $\Omega = [-1,1] \times [-1,1]$ has the exact solution:

$$\mathbf{u} = \begin{pmatrix} 20xy^3 \\ 5(x^4 - y^4) \end{pmatrix} = \begin{pmatrix} u \\ v \end{pmatrix}, \quad p = 60x^2y - 20y^3 + c \tag{3}$$

To calculate the right hand side, and to ensure it satisfies the continuity equation, we have:

$$\nabla \mathbf{u} = \begin{pmatrix} \nabla u \\ \nabla v \end{pmatrix} = \begin{pmatrix} 20y^3 & 60xy^2 \\ 20x^3 & -20y^3 \end{pmatrix} \tag{4}$$

$$\Delta \mathbf{u} = \begin{pmatrix} 120xy \\ 60(x^2 - y^2) \end{pmatrix}, \quad \nabla p = \begin{pmatrix} 120xy \\ 60(x^2 - y^2) \end{pmatrix} \tag{5}$$

So:

$$-\Delta \mathbf{u} + \nabla p = \mathbf{f} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{6}$$

And:

$$\partial_x u + \partial_y v = 20y^3 - 20y^3 = 0 \tag{7}$$

1.2 Problem 2

The second problem to analyze is the lid driven cavity problem:

$$\begin{cases} -\Delta u + \frac{\partial p}{\partial x} = 0, & (x, y) \in \Omega \\ -\Delta v + \frac{\partial p}{\partial y} = 0, & (x, y) \in \Omega \\ \operatorname{div}(\mathbf{u}) = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0, & (x, y) \in \Omega \\ u(x, y) = g_1(x, y), & (x, y) \in \partial\Omega \\ v(x, y) = 0, & (x, y) \in \partial\Omega \end{cases} \tag{8}$$

Where:

$$g_1(x, y) = \begin{cases} 1, & (x, y) \in [-1, 1] \times \{1\} \\ 0 & (x, y) \notin [-1, 1] \times \{1\} \end{cases} \tag{9}$$

This problem has no known analytic solution, therefore the only way to measure the error is by measuring the residual.

2 Part 1 - Distributed Gauss-Seidel Method

The first method we use to solve equation (3) is the distributed Gauss-Seidel Relaxation (SSOR) method. The discretization being used is the finite difference method on a staggered grid known as the Marker and Cell (MAC) scheme.

From ([2]), the optimal relaxation parameter is given by:

$$\omega = \frac{2}{1 + \sin(\pi h)} \tag{10}$$

This will be used on the Distributed Gauss-Seidel method as well as on using the Distributed Gauss-Seidel method to solve the residual equation on the coarse grid in the multigrid methods.

The algorithm description of the Distributed Gauss-Seidel method used is given on the next page. It is the same Distributed Gauss-Seidel algorithm that is given in ([1]).

2.1 Algorithm

For the distributed Gauss Seidel method on the Marker and Cell (MAC) scheme, we are using a correction to solve the continuity equation. The algorithm is as follows:

Given the current approximations $\mathbf{u}^k,\ p^k$:

$$\left[\mathbf{u}^{k+1}, p^{k+1}\right] = DGS\left(\mathbf{u}^k, p^k\right) \tag{11}$$

1. Relax the momentum equations to find the intermediate velocity field by solving the poisson equation:

$$\begin{cases} \Delta \mathbf{u}^{k+\frac{1}{2}} = \nabla p^k - f(x,y), & (x,y) \in \Omega \\ \mathbf{u}^{k+\frac{1}{2}} = g(x,y) & (x,y) \in \partial\Omega \end{cases}$$

2. Relax the residual of the continuity equation by solving the poisson equation:

$$\begin{cases} \Delta \phi = \nabla \cdot \mathbf{u}^{k+\frac{1}{2}} - g(x,y), & (x,y) \in \Omega \\ \frac{\partial \phi}{\partial \mathbf{n}} = 0, & (x,y) \in \partial\Omega \end{cases}$$

3. Update the next step with the correction:

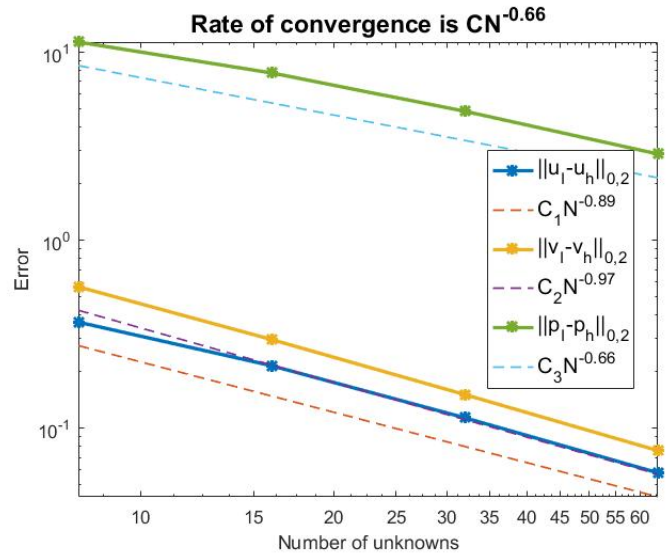
$$\begin{aligned} \mathbf{u}^{k+1} &= \mathbf{u}^{k+\frac{1}{2}} + \nabla \phi \\ p^{k+1} &= p^k - \Delta \phi \end{aligned}$$

2.2 First Problem

After approximating the solution to (3), we can look at the L_2 norm difference between the actual function and the approximations.

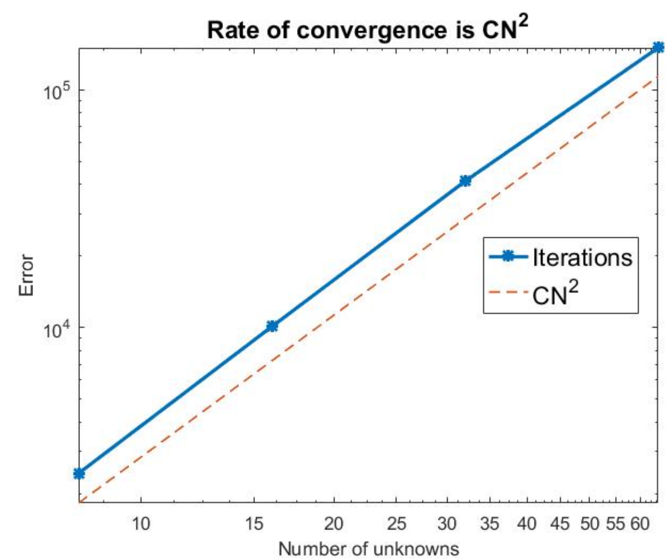
2.2.1 Error Convergence Rate

The convergence rate for the error is given below. In this, we vary h from 2^{-3} to 2^{-5}. The rate of convergence is given by:



2.2.2 Iterations

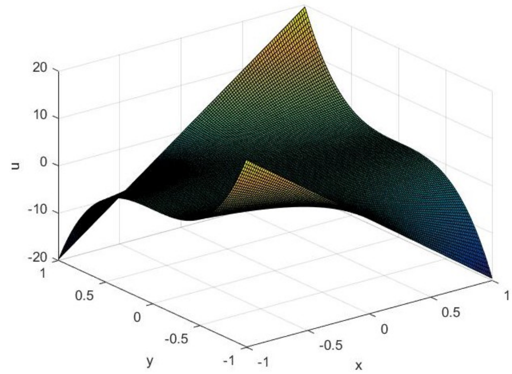
The total number of iterations required for each h is given in the plot below:



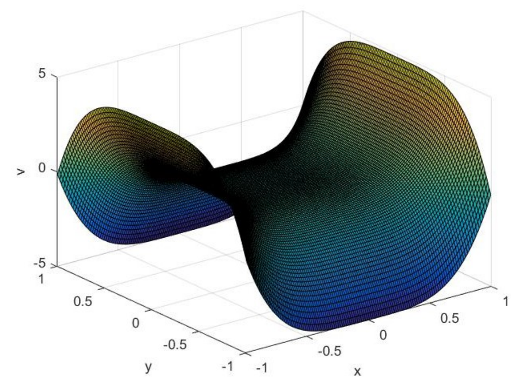
2.2.3 Solution Plots

We can plot the velocity fields and the pressure. The plots are given by:

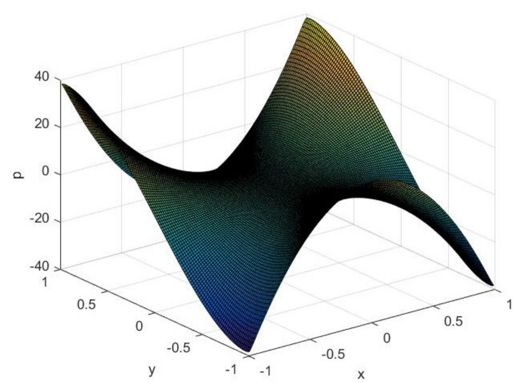
- For u



- For v



- For p

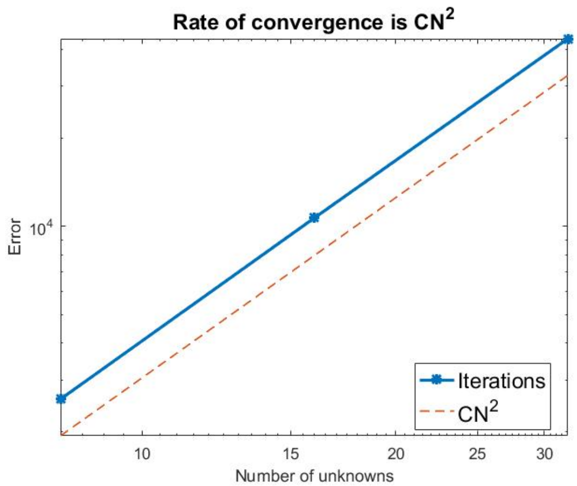


2.3 Second Problem

The second problem which will be used to test the Distributed Gauss-Seidel method is the lid-driven cavity problem (8).

2.3.1 Iterations

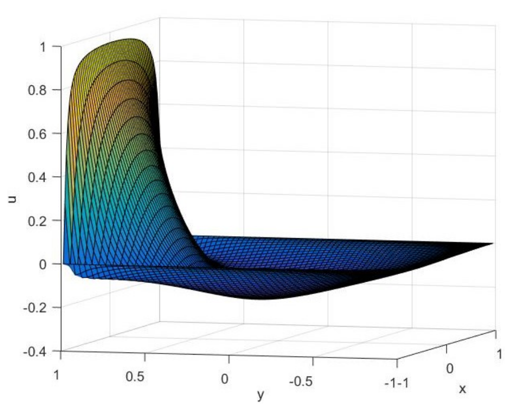
The total number of iterations required for each h is given in the plot below:



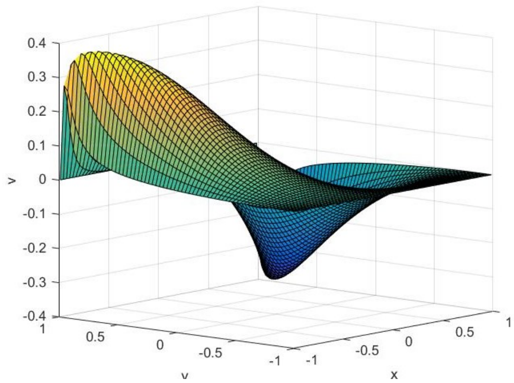
2.3.2 Solution Plots

We can plot the velocity fields and the pressure since there is no error to plot. The plots are given by:

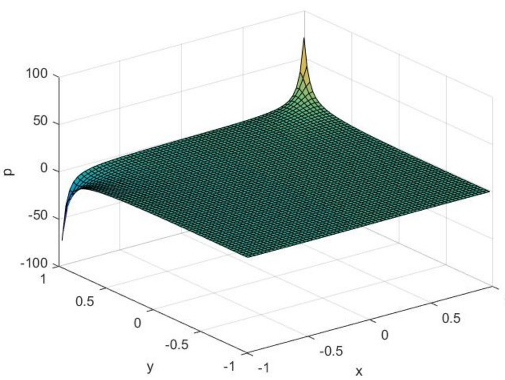
- For u



- For v



- For p



3 Part 2 - Two-Grid Method

This section will analyze using a two grid method combined with Distributed Gauss-Seidel to approximate a solution to (3) and (8). The mesh used is still the one given by the Marker and Cell (MAC) scheme.

The description of the algorithm is given on the next page.

3.1 Algorithm

For the two-grid algorithm, we use a recursive method so that it can be easily converted to a multigrid method. The alogirthm is as follows:

$$\left[\mathbf{u}^{k+1}, p^{k+1}\right] = \text{TwoGrid}\left(\mathbf{u}^k, p^k\right) \tag{12}$$

1. Pre-smooth with Distributed Gauss-Seidel (11):

$$\left[\mathbf{u}^k, p^k\right] = \text{DGS}\left(\mathbf{u}^k, p^k\right)$$

2. Calculate the Residual for the momentum equations and the continuity equation:

$$[\mathbf{r}_u, r_p] = \left(f(x,y) - A\mathbf{u}^k - B^T p^k, \nabla \cdot \mathbf{u}^k\right)$$

3. Restrict the residual equations to the coarse grid:

$$\left[\mathbf{r}_u^c, r_p^c\right] = \text{Restriction}\left(\mathbf{r}_u, r_p\right)$$

4. Solve the residual equations in the coarse grid:

$$\begin{cases} A_u \mathbf{e}_u^c = \mathbf{r}_u^c, \\ A e_p^c = r_p^c \end{cases}$$

5. Prolongate the error corrections back to the fine grid:

$$[\mathbf{e}_u, e_p] = \text{Prolongation}\left(\mathbf{e}_u^c, e_p^c\right)$$

6. Update the corrections to the original variables:

$$\begin{aligned} \mathbf{u}^{k+1} &= \mathbf{u}^k + \mathbf{e}_u \\ p^{k+1} &= p^k + e_p \end{aligned}$$

7. Post-smooth with Distributed Gauss-Seidel (11):

$$\left[\mathbf{u}^{k+1}, p^{k+1}\right] = \text{DGS}\left(\mathbf{u}^k, p^k\right)$$

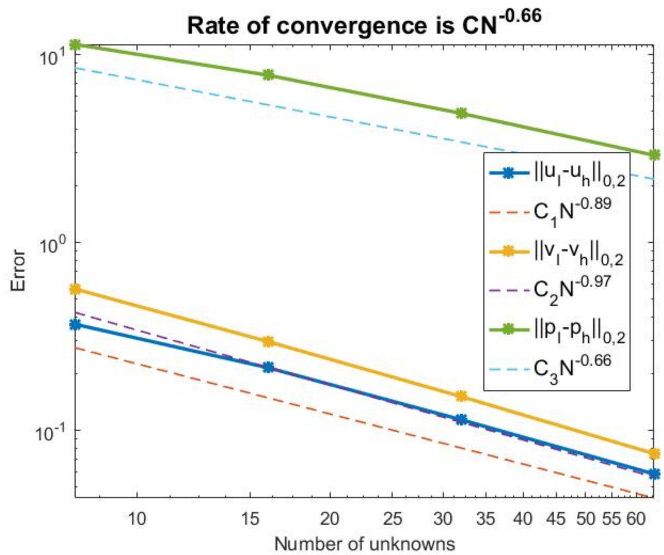
8. Return the final values of $\left[\mathbf{u}^{k+1}, p^{k+1}\right]$

3.2 First Problem

After approximating the solution to (3), we can look at the L_2 norm difference between the actual function and the approximations.

3.2.1 Error Convergence Rate

The convergence rate for the error is given below. For the figure below, we vary h from 2^{-3} to 2^{-6}. The rate of convergence is given by:



A tabular representation of the number of iterations and the error is given by:

h on fine grid	Smoothing Steps	Tolerance	Iterations	Error
0.125	3	0.0090689	10	0.56604
0.0625	3	0.050275	10	0.29686
0.03125	3	0.27785	10	0.15135
0.015625	3	1.5459	9	0.075115

So using the finite difference method on all of the different size meshes take 10 iterations, which is expected.

3.3 Second Problem

This section analyzes the number of iterations required to solve the lid-driven cavity problem (8) using the two grid method.

3.3.1 Iterations

The total number of iterations required for each h is given in the table below:

h on fine grid	Smoothing Steps	Tolerance	Iterations
0.125	3	0.00049574	8
0.0625	3	0.0028507	8
0.03125	3	0.016255	8
0.015625	3	0.092319	8

Which is the same regardless of the value of h used as expected.

4 Part 3 - Multigrid Method

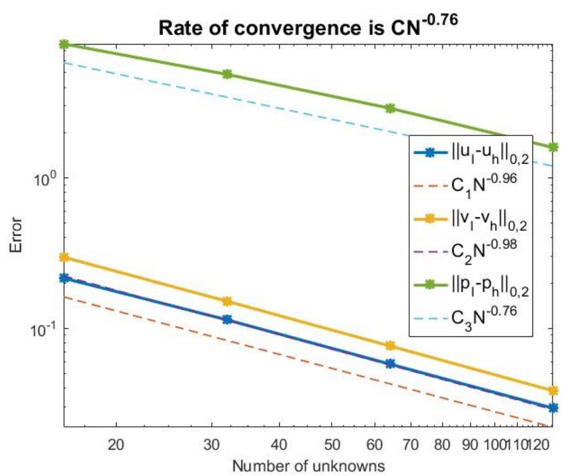
The multigrid algorithm is just a slight modification of the two-grid algorithm (12). The difference is that instead of solving the residual equation in step 3, the residual equation is passed recursively to the multigrid method until the desired number of levels has been reached. Then the residual equation is solved on the coarse grid and the correction is prolongated back to the fine grid, where the original variables on the fine grid are updated.

4.1 Problem 1

This section analyzes the convergence rates of the multigrid (J level V-cycle) method on the problem (3) varying both h as well as J

4.1.1 Convergence Rate For Error

When we vary h, as well as J with a relative tolerance of 10^{-6}, we get the following results for convergence:



The parameters used in the above simulation are given by:

h on fine grid	J	Smoothing Steps	Tolerance	Iterations	Error
0.0625	2	3	0.050275	10	0.29686
0.03125	3	3	0.27785	12	0.15136
0.015625	4	3	1.5459	13	0.076245
0.0078125	5	3	8.6583	14	0.03843

4.1.2 Number of Iterations

When we are just testing the number of iterations for $h = 2^{-7}$ and different levels, we find:

h on fine grid	J	Smoothing Steps	Tolerance	Iterations	Error
0.0078125	3	3	8.6583	11	0.037456
0.0078125	4	3	8.6583	13	0.038563
0.0078125	5	3	8.6583	14	0.03843
0.0078125	6	3	8.6583	15	0.03836

Which is not completely uniform as expected, but still under 20 iterations. This could be due to not solving to a low tolerance on the fine grid. When only 3 smoothing steps are used, the method does not converge within the same number of steps. So the extra smoothing step was required for the convergence to be correct.

4.2 Second Problem

This section analyzes the number of iterations required to solve the lid-driven cavity problem (8) using the two grid method.

4.2.1 Iterations

The total number of iterations required for each h is given in the table below:

h on fine grid	J	Smoothing Steps	Tolerance	Iterations
0.0625	3	3	0.0028507	8
0.03125	4	3	0.016255	10
0.015625	5	3	0.092319	12
0.0078125	6	3	0.52326	13

When we vary the level without varying h, the method converges within a relatively close number of steps. This is shown in the table below.

h on fine grid	J	Smoothing Steps	Tolerance	Iterations
0.0078125	3	3	0.52326	10
0.0078125	4	3	0.52326	12
0.0078125	5	3	0.52326	13
0.0078125	6	3	0.52326	13

References

[1] Ming Wang and Long Chen. "Multigrid Methods for the Stokes Equations Using Distributive Gauss—Seidel Relaxations Based on the Least Squares Commutator". In: *J. Sci. Comput.* 56.2 (Aug. 2013), pp. 409–431. ISSN: 0885-7474. DOI: `10.1007/s10915-013-9684-1`. URL: `http://dx.doi.org/10.1007/s10915-013-9684-1`.

[2] Shiming Yang and Matthias K. Gobbert. "The optimal relaxation parameter for the {SOR} method applied to the Poisson equation in any space dimensions". In: *Applied Mathematics Letters* 22.3 (2009), pp. 325–331. ISSN: 0893-9659. DOI: `http://dx.doi.org/10.1016/j.aml.2008.03.028`. URL: `http://www.sciencedirect.com/science/article/pii/S0893965908001523`.