

MECH 510

Final Project

Prepared for:

Prof. Carl Ollivier-Gooch

CEME 2050, UBC

Prepared by:

Nicholas Earle

UBC # 21943600

Date:

December 19th, 2018

# Introduction

The objective of this assignment was to write a program to solve the incompressible, laminar Navier-Stokes equations in two dimensions, on a rectangular Cartesian grid. An artificial compressibility discretization was use for the coupled equations, using a second order discretization in space, and an implicit Euler time advance. A block Thomas algorithm and approximate factorization scheme was then used to solve each step of the implicit time advance.

The project was broken into many parts to make the task more manageable and provide numerous checks along the way. The first part was to ensure that the flux integral and Jacobian matrices were correct. This was followed by building the approximate factorization scheme for the implicit solver. Testing for this included looking at the pressure oscillations in the solution, as well as tuning the value of beta, and the over-relaxation. Now on to the basic solution of a box with a moving top. First, the code was again checked for stability, then run with the top of the box moving to the right. Then for our own sanity, it was run with the top moving at the same speed to the left. These results should be almost exactly opposite. And finally, we looked at the effects of increasing the height of the box.

# Formulation

First things first, the Navier-Stokes equations:

Where equation 1 is the conservation of mass, and equation 2 is the conservation of momentum. Expanded into two dimensions these look like:

Conservation of mass:

Conservation of x-momentum:

Conservation of y-momentum;

Now applying the artificial compressibility method for solving these equations, we get a non-dimensionalized form of:

Where:

Applying the implicit Euler time advance and space discretization to equation 6, we get:

Where:

And likewise for the other discretization of F and G. Now, if we Taylor expand our terms for n+1, we obtain the following equation for

Applying this expansion to the discrete form in equation 8, we obtain a fully discretized form that can be solved:

Where:

And likewise for G and at other locations. Collecting like terms from equation 11, we can write it in the form:

Where each coefficient A, B, and C, correspond to the partial derivatives found in equation 11, with subscript x for F, and y for G. From here we can expand this equation into a sparse matrix with five diagonal entries of size N2, or using the approximate factorization scheme we can represent it as:

Where Dx and Dy are simply a sparse matrix of the A, B, and C derivatives in x and y, respectively. From here, these equations can be solved using the tri-diagonal block Thomas algorithm. The last step in the formulation is what to do about boundary conditions. Because this is a viscous flow, the velocity boundary conditions are no-slip conditions. Simply saying that the velocity at the wall equals the velocity of the wall, introducing two Dirichlet boundary conditions.

Pressure is not as simple, where from the Navier-Stokes equations we get that it is

Which is not exactly zero, but it is very close, so for the sake of the solver we will approximate the pressure gradient at the wall to be zero, introducing our final Neumann boundary condition.