

# Third programming Assignment

Peter Monk

December 2, 2009

You may work in pairs on this assignment (or alone if you prefer).

1. Writing the Navier Stokes solver was intended to suggest some of the features often found in scientific computing projects:
  - (a) A scientific problem (in this case fluid flow)
  - (b) The need for a combination of algorithms to solve the basic problem:
    - The special finite difference method to help handle the incompressibility of the fluid
    - Special time stepping (implicit in the pressure, explicit in the diffusion)
    - An appropriate Poisson solver
  - (c) Once we have the basic solver we often want to add extra features. In our case:
    - We added convection
  - (d) Going forward we could easily add more scientific features (for example particle transport, temperature effects) or more flexibility to the solver (more complex geometry, moving boundaries and different boundary conditions).
  - (e) Another possible direction would be to address efficiency. We might try to speed up the slowest parts of the algorithm (a better Poisson solver for example) or parallelize the code.
  - (f) However, another area which always consumes a large amount of time is writing code to understand and visualize the solution. This is the main subject of this assignment.

Since the vector field  $(u, v)$  is divergence free it is the curl of a potential called the *stream function*, so that there is a function  $\psi(x, y)$  such that

$$\frac{\partial \psi}{\partial x} = -v \text{ and } \frac{\partial \psi}{\partial y} = u$$

This function has the physical interpretation as follows. At a given time  $t$ , a *streamline* is a curve whose tangent is parallel to the velocity vector  $(u, v)$  at each point  $(x, y)$ . The streamlines are contours of  $\psi$ . Contour plots of  $\psi$  are easier to interpret than the random quiver plots (or at least complement them).

We will compute an approximate discrete stream function  $\psi_{i,j}$  and for simplicity we shall associate this value with the upper right hand corner of the  $(i, j)$ th cell. So using the zero based indices of your scanned pages we will compute  $\psi_{i,j}$  for  $0 \leq i \leq imax$  and  $0 \leq j \leq jmax$ . We will do this in a crude way using only the second equation of the definition (and hope the first one is OK....). We discretize

$$\frac{\partial \psi}{\partial y} = u$$

by

$$\left[ \frac{\partial \psi}{\partial y} \right]_{i,j} = \frac{\psi_{i,j} - \psi_{i,j-1}}{\delta y}$$

Fixing arbitrarily  $\psi_{i,0} = 0$  we get

$$\psi_{i,j} = \psi_{i,j-1} + u_{i,j} \delta y, \text{ for } j = \dots, jmax$$

and each  $i$ . This is what you need to implement. Write a function

```
function [x,y,psi]=comp_psi(u,v,imax,jmax,dely)
```

that computes **psi** as above and also returns the positions of  $x_i$  and  $y_j$ . You can plot the contour map by

```
contour(x,y,psi)
```

Rerun the examples for  $re = 100, 1000$  and  $5000$  from the previous Computing Assignment and plot the stream functions at the times requested. For  $re = 100$  you should observe a single rotating eddy. For  $re = 1000$  secondary eddies will start in the lower two corners, whereas for  $re = 5000$  a third eddy should appear in the upper left hand corner.

2. Perform challenges 18.1, 18.2 and 18.3 in the book (see Chapter 18).