Solving the Navier-Stokes Equations in Primitive Variables

Grétar Tryggvason
Spring 2011

The projection method—review
Methods for the Navier-Stokes Equations
Moin and Kim
Bell, et al
Colocated grids
Boundary conditions

Summary of discrete vector equations

\[
\frac{u_{i,j}^{n+1} - u_{i,j}^{n}}{\Delta t} = -\nabla \cdot (\nabla P_{i,j}) + D_{i,j}
\]

Evolution of the velocity

\[
\nabla \cdot u_{i,j}^{n+1} = 0
\]

Constraint on velocity

No explicit equation for the pressure!

To derive an equation for the pressure we take the divergence of

\[
u_{i,j}^{n+1} = u_{i,j}^{n} - \Delta t \nabla \cdot (\nabla P_{i,j})
\]

and use the mass conservation equation

\[
\nabla \cdot u_{i,j}^{n+1} = 0
\]

The result is

\[
\nabla P_{i,j}^{n+1} = \frac{1}{\Delta t} \nabla \cdot u_{i,j}^{n+1}
\]

Projection Method

1. Find a temporary velocity using the advection and the diffusion terms only:

\[
u_{i,j}^{n+1} = u_{i,j}^{n} + \Delta t (-\nabla \cdot (\nabla P_{i,j}))
\]

2. Find the pressure needed to make the velocity field incompressible

\[
\nabla P_{i,j}^{n+1} = \frac{1}{\Delta t} \nabla \cdot u_{i,j}^{n+1}
\]

3. Correct the velocity by adding the pressure gradient:

\[
u_{i,j}^{n+1} = u_{i,j}^{n} - \Delta t \nabla \cdot (\nabla P_{i,j})
\]
Computational Fluid Dynamics
Algorithm

**Determine u, v boundary conditions**

Advect $u_i' = u_i + \Delta t \left(-A_{ij}D'_{ij} + D_{ij}'\right)$

Poisson equation for $p_{ij}$ (iteration)

Projection $u_{ij}' = u_{ij} - \Delta t \nabla_p p_{ij}$

$t = t + \Delta t$

---

**Forward in time, centered in space (summary):**

$u^* - u^n = -\frac{\Delta t}{\nu} \nabla^2 u^*$

$\nabla^2 p = \frac{1}{\Delta t} \nabla \cdot u^*$

$u^{n+1} = u^* - \Delta t \nabla p$

**Time step limitations**

$\Delta t \leq \frac{2\nu}{U^2}$ & $\Delta t \leq \frac{1}{4} \frac{h^3}{\nu}$

**Increasing h**

$\Delta t^* = \frac{2}{Re}$

$\Delta t^* = \frac{1}{4} \frac{h^3}{L^2}$

**What is the maximum timestep?**

$Re_{max} = \frac{L}{h}$

And $\Delta t \rightarrow 0$ for $Re \rightarrow 0$ and $Re \rightarrow \infty$

---

Advanced Solvers

- For low Re, use implicit methods for diffusion term
- For high Re, use stable advection schemes
- Combine both for schemes intended for all Re

---

**Fully Implicit**

$$\frac{u^{n+1} - u^n}{\Delta t} = \frac{1}{2} \left(-\left(A(u^n) + A(u^{n+1})\right) + \nabla \left[\nabla^2 u^n + \nabla^2 [u^{n+1}]\right]\right) - \nabla p$$

$\nabla \cdot u^{n+1} = 0$

Solve by iteration

Rarely used due to the complications of the nonlinear system that must be solved for the advection terms
**Predictor-Corrector**

A second order method can be developed by first taking a forward step, then a backward step and average the results:

\[
\begin{align*}
\frac{\mathbf{u}^n - \mathbf{u}^{n+1}}{\Delta t} &= -\mathbf{V}\mathbf{u}^n + \frac{1}{2}\mathbf{V}\mathbf{u}^{n+1} \\
\mathbf{V}^\phi &= \frac{1}{\Delta t} \mathbf{V}\mathbf{\tilde{u}}
\end{align*}
\]

Backward step using the predicted velocity:

\[
\begin{align*}
\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} &= -\mathbf{V}\mathbf{u}^n' + \frac{1}{2}\mathbf{V}\mathbf{u}^{n+1}' \\
\mathbf{V}^\phi &= \frac{1}{\Delta t} \mathbf{V}\mathbf{\tilde{u}}
\end{align*}
\]

Then average the results

\[
\begin{align*}
\mathbf{u}^{n+1/2} &= \frac{1}{2} (\mathbf{u}^n + \mathbf{u}^{n+1}) \\
\mathbf{V}^\phi &= \frac{1}{2\Delta t} (\mathbf{V}\mathbf{\tilde{u}} + \mathbf{V}\mathbf{\tilde{u}}')
\end{align*}
\]

**Adam-Bashford/Crank-Nicolson**

\[
\begin{align*}
\mathbf{u}^{n+1} - \mathbf{u}^n &= \left( \frac{3}{2} A(\mathbf{u}^n) - \frac{1}{2} A(\mathbf{u}^{n+1}) \right) \mathbf{V} \left( \mathbf{V}^\phi + \mathbf{V}^\phi_2 \right) - \mathbf{V} p \\
\mathbf{V} \cdot \mathbf{u}^{n+1} &= 0
\end{align*}
\]

Split:

\[
\begin{align*}
\mathbf{u}^{n+1} - \mathbf{u}^n &= \left( \frac{3}{2} A(\mathbf{u}^n) - \frac{1}{2} A(\mathbf{u}^{n+1}) \right) \mathbf{V} \mathbf{V}^\phi_2 - \mathbf{V} p \\
\mathbf{u}^{n+1} - \mathbf{u}^n &= \mathbf{V}^\phi_2
\end{align*}
\]

The correction equation is implicit and must be solved by an iteration in the same way as the pressure equation.

**Method of Kim and Moin (JCP 59 (1985), 8-29)**

\[
\begin{align*}
\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} &= \left( \frac{3}{2} A(\mathbf{u}^n) - \frac{1}{2} A(\mathbf{u}^{n+1}) \right) \mathbf{V} \left( \mathbf{V}^\phi + \mathbf{V}^\phi_2 \right) - \mathbf{V} p \\

\end{align*}
\]

Notice that \( \Phi \) is not exactly p. Adding the first two equations gives

\[
\begin{align*}
\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} &= \left( \frac{3}{2} A(\mathbf{u}^n) - \frac{1}{2} A(\mathbf{u}^{n+1}) \right) \mathbf{V} \left( \mathbf{V}^\phi + \mathbf{V}^\phi_2 \right) - \mathbf{V} p \\
\mathbf{V} \cdot \mathbf{u}^{n+1} &= 0
\end{align*}
\]

Where we have added and subtracted an implicit diffusion term.

Using

\[
\begin{align*}
\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} &= \mathbf{V} \mathbf{\phi} \\
\mathbf{V} \cdot \mathbf{u}^{n+1} &= 0
\end{align*}
\]

we can rewrite the last terms as:

\[
\begin{align*}
\frac{\mathbf{V}^\phi (\mathbf{V}^\phi - \mathbf{V}^\phi_2)}{2} = \frac{\mathbf{V}^\phi (\mathbf{V}^\phi - \mathbf{V}^\phi_2)}{2}
\end{align*}
\]

**Method of Bell, Colella and Glaz (JCP 85 (1989), 7-83)**

\[
\begin{align*}
\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} &= -A(\mathbf{u}^{n+1}) + \frac{3}{2} \mathbf{V} \left( \mathbf{V}^\phi + \mathbf{V}^\phi_2 \right) - \mathbf{V} p \\
\mathbf{V} \cdot \mathbf{u}^{n+1} &= 0
\end{align*}
\]

A Godunov method is used for the advection terms.

**A complete Runge-Kutta time integration (Weinan E.)**

First a half step:

\[
\begin{align*}
\frac{\mathbf{u}^{n+1/2} - \mathbf{u}^n}{\Delta t} &= -\mathbf{V}\mathbf{u}^n + \frac{3}{2} \mathbf{V} \frac{1}{2} \mathbf{V} \mathbf{u}^{n+1} - \mathbf{V} p \\
\mathbf{V} \cdot \mathbf{u}^{n+1/2} &= 0
\end{align*}
\]

continue for the second step:

\[
\begin{align*}
\frac{\mathbf{u}^{n+1} - \mathbf{u}^{n+1/2}}{\Delta t} &= -\mathbf{V}\mathbf{u}^{n+1/2} + \frac{3}{2} \mathbf{V} \frac{1}{2} \mathbf{V} \mathbf{u}^{n+1} - \mathbf{V} p \\
\mathbf{V} \cdot \mathbf{u}^{n+1} &= 0
\end{align*}
\]
A complete Runge-Kutta time integration (continued)

Take a full step using the predicted velocity

\[
\begin{align*}
\frac{u^{n+1} - u^n}{\Delta t} &= -\nabla^2 u^{n+1} + \nu \nabla^2 u^n \\
\nabla^2 p^{n+1} &= \frac{1}{\Delta t} \nabla^2 u^{n+1} \\
u^{n+1} &= u^n - \Delta \nabla p^n
\end{align*}
\]

Then compute

\[k = \Delta t (-\nabla^2 u^n + \nu \nabla^2 u^n)\]

And finally

\[
\begin{align*}
u^{n+1} &= \frac{1}{2} (-u^n + u^{n+1} + 2u^n + u^{n+1}) + \frac{k}{\Delta t} \\
\nabla^2 p^n &= \frac{1}{\Delta t} \nabla^2 u^n \\
u^n &= u^n - \Delta \nabla p^n
\end{align*}
\]

Colocated grids

Although staggered grids have been very successful, in some cases it is desirable to use co-located (or colocated) grids where all variables are located at the same physical point.

Staggered grids

Colocated grids

All variables are stored at the same location

\[
\begin{align*}
u_{i,j}^n &= u_{i,j}^n - \frac{\Delta x}{2} (u_{i-1,j}^n - u_{i+1,j}^n) \\
u_{i,j}^{n+1} &= u_{i,j}^n - \frac{\Delta t}{\Delta x} \left( \frac{1}{2} (p_{i+1,j}^n + p_{i,j}^n) - \frac{1}{2} (p_{i-1,j}^n + p_{i,j}^n) \right) \\
u_{i,j}^{n+2} &= u_{i,j}^n - \frac{\Delta t}{2\Delta x} (p_{i+1,j}^n - p_{i-1,j}^n)
\end{align*}
\]

Split equations for the \( u \) velocity

First idea: use averaging for the variables on the edges:

\[
\begin{align*}
p_i^n &= \frac{1}{2} (p_{i+1}^n + p_i^n) \\
u_{i,j}^n &= \frac{1}{2} (u_{i-1,j}^n + u_{i+1,j}^n)
\end{align*}
\]
Substituting
\[ u_i = \frac{1}{2}(u_{i+1} + u_{i-1}) \]
and
\[ u_i^{*\text{nl}} = u_i^{*\text{nl}} = \frac{\Delta t}{\Delta x} (p_{i+1} - p_{i-1}) \]
into
\[ u_i^{*\text{nl}} - u_i^{*\text{nl}} + v_i^{*\text{nl}} - v_i^{*\text{nl}} = 0 \]
yields
\[ p_{i+1} + p_{i-1} + p_{i+2} + p_{i-2} - 4p_i = \frac{2\Delta t}{\Delta x} (u_i^{*\text{nl}} - u_i^{*\text{nl}} + v_i^{*\text{nl}} - v_i^{*\text{nl}}) \]

The remedy is to find the pressures that make the edge velocities incompressible.


A straightforward application discretization on colocated grids results in a very wide stencil for the pressure field, and the pressure points are also uncoupled and the pressure field can develop oscillations.

The Rhie and Chow method

Instead of interpolating (the final velocity)
\[ u_i^{*\text{nl}} = \frac{1}{2}(u_{i+1}^{*\text{nl}} + u_{i-1}^{*\text{nl}}) \]
interpolate (the intermediate velocity)
\[ u_i^{*\text{nl}} = \frac{1}{2}(u_{i+1} + u_{i-1}) \]
and then find
\[ u_i^{*\text{nl}} = u_i^{*\text{nl}} = \frac{\Delta t}{\Delta x} (p_{i+1} - p_i) \]
In effect, “pretend” we are using a staggered grid.

Substitute
\[ u_i^{*\text{nl}} = u_i - \frac{\Delta t}{\Delta x} (p_{i+1} - p_{i-1}) \]
where
\[ u_i^{*\text{nl}} = \frac{1}{2}(u_{i+1} + u_{i-1}) \]
into
\[ u_i^{*\text{nl}} - u_i^{*\text{nl}} + v_i^{*\text{nl}} - v_i^{*\text{nl}} = 0 \]
giving
\[ \left( u_i - \frac{\Delta t}{\Delta x} (p_{i+1} - p_{i-1}) \right) \left( v_i - \frac{\Delta t}{\Delta x} (p_{i+1} - p_{i-1}) \right) + 0 = 0 \]
Rearrange:
\[ p_{i+1} + p_{i-1} + p_{i+2} + p_{i-2} - 4p_i = \frac{\Delta t}{\Delta x} (u_i^{*\text{nl}} - u_i^{*\text{nl}} + v_i^{*\text{nl}} - v_i^{*\text{nl}}) \]

For the correction of the momentum equation we still use the average of the pressures
\[ p_i = \frac{1}{2} (p_{i+1} + p_{i-1}) \]
giving
\[ u_i^{*\text{nl}} = u_i^{*\text{nl}} = \frac{\Delta t}{\Delta x} (p_{i+1} - p_{i-1}) \]
The algorithm is therefore:

1. First find predicted velocities:
   \[ u_j^* = u_j - \Delta t \Delta x (u_j) \]  and  \[ v_i^* = v_i - \Delta t \Delta y (v_i) \]

2. Find pressure by solving:
   \[ p_{i,j} + p_{i+1,j} + p_{i,j+1} - 4 p_{i,j} = \frac{h}{2\Delta t} \left( u_{i+1,j} - u_{i,j} + v_{i,j+1} - v_{i,j} \right) \]
   suitably modified at the boundaries

3. Correct the velocities:
   \[ u_j^{**} = u_j^* - \frac{\Delta t}{2h} (p_{i+1,j} - p_{i-1,j}) \]  and  \[ v_i^{**} = v_i^* - \frac{\Delta t}{2h} (p_{i,j+1} - p_{i,j-1}) \]

Find the pressure gradient by applying the Navier-Stokes equations at a point at the boundary
\[
\frac{\partial}{\partial x} \left( \frac{\partial p}{\partial x} \right) + \frac{\partial}{\partial y} \left( \frac{\partial p}{\partial y} \right) = -\frac{\partial}{\partial x} \left( \mu \frac{\partial u}{\partial x} \right) - \frac{\partial}{\partial y} \left( \mu \frac{\partial v}{\partial y} \right)
\]
At the wall, most of the terms are zero
\[
\frac{\partial p}{\partial y} = \frac{\partial v}{\partial y} \Rightarrow \frac{p_2 - p_1}{h} = \mu \frac{\partial v}{\partial y}
\]
Evaluated by one-sided differences

Write the pressure equation for \( j = 2 \)
\[
p_{i,2} + p_{i,1} + p_{i+1,2} - 4 p_{i,2} = \frac{h}{2\Delta t} \left( u_{i+1,2} - u_{i,2} + v_{i,3} - v_{i,1} \right)
\]
And use
\[
p_{i,1} - p_{i,2} = \frac{h}{2\Delta t} v_i
\]
For the pressure at \( j = 1 \)
Colocated grids

Why colocated grids:
- Sometimes simpler for body fitted grids
- Easy to use methods for hyperbolic equations
- Easier to implement AMR
- Some people just don’t like staggered grids!

Boundary conditions

Inflow and outflow

Boundary conditions

- Fully enclosed flow
- Driven cavity
- Internal flow (inflow & outflow)
- External flow
- Flow over bodies

For fully enclosed domains, such as in the driven cavity problem, the application of the boundary conditions is relatively straightforward.

For domains with inflows it is usually reasonable to specify the velocity profile at the inlet.

The major problem is how to handle outflow boundaries in such a way that the fluid leaves in a “physically plausible” way.

The Backward Facing Step Problem

\[ u = u(y) \]
\[ v = 0 \]
\[ \frac{\partial v}{\partial x} = 0 \]
\[ \frac{\partial u}{\partial x} = 0 \]
Example

For the pressure, obtain an equation by applying conservation of mass to the control volume next to the boundary

\[ p(x; j = (1/2)(p(x, j+1) + p(x, j-1) - h(\partial v/\partial y)(v(x+1, j+1) - v(x+1, j))) \]
Periodic Boundary conditions
In many cases it is possible to use periodic boundary conditions, where what flows out through one boundary reappears flowing in through the opposite boundary. Such conditions are particularly suitable for theoretical studies of idealized flows. For such boundaries it is easiest to specify the pressure drop, but by adjusting the pressure gradient it is possible to specify the volume flux.

Other ways to deal with free-stream boundaries
- Include potential flow perturbation
- Compute flow from vorticity distribution
- Map the boundary at infinity to a finite distance

Fundamentally, the specification of the boundary conditions does not have a unique solution and is also faced by experimentalists. However, by taking the boundaries far away and checking the solution for the effect of moving the boundaries, good results can be obtained.

The two-dimensional programs developed in the project and shown here can be extended to fully three-dimensional flows in a relatively straightforward way, replacing \( u(i,j) \) by \( u(i,j,k) \), etc. The time required to run the code increases significantly and visualizing the output becomes more challenging.