Pseudospectral Methods

Vorticity: $\vec{w} = \nabla \times \vec{u}$

$$\vec{u}_t + (\vec{u}\cdot\nabla)\vec{u} = -\nabla p + \frac{1}{\text{Re}}\nabla^2\vec{u}$$

$$\Rightarrow \nabla\times\vec{u}_t + \nabla\times((\vec{u}\cdot\nabla)\vec{u}) = \underbrace{\nabla\times(-\nabla p)}_{=0} + \underbrace{\nabla\times(\frac{1}{\text{Re}}\nabla^2\vec{u})}_{\nabla\times\text{ and }\nabla^2\text{ commute}}$$

$$\Rightarrow \boxed{\vec{w}_t + \vec{u}\cdot\nabla\vec{w} = \frac{1}{\text{Re}}\nabla^2\vec{w}}$$

Navier-Stokes equations in vorticity formulation

How to find $\vec{u}$ from $\vec{w}$?

$\boxed{\text{2D}}$ $w = v_x - u_y$ scalar

Stream Function:

$\psi(\vec{x})$ s.t. $\vec{u} = \nabla^\perp\psi$, i.e. $\left\{\begin{array}{l} u = \psi_y \\ v = -\psi_x \end{array}\right\}$

$\Rightarrow \nabla^2\psi = \psi_{xx} + \psi_{yy} = -v_x + u_y = -w$

Thus: $w \overset{\nabla^2\psi=-w}{\longrightarrow} \psi \overset{\vec{u}=\nabla^\perp\psi}{\longrightarrow} \vec{u}$

Remark: Incompressibility guaranteed

$\nabla\cdot\vec{u} = u_x + v_y = \psi_{yx} - \psi_{xy} = 0$ ✓

Time Discretization

Semi-implicit:

- $\vec{u}\cdot\nabla w$ explicit
- $\nabla^2 w$ Crank-Nicolson

$$\frac{w^{n+1}-w^n}{\Delta t} + \underbrace{(\vec{u}^n\cdot\nabla w^n)}_{=N^n} = \frac{1}{\text{Re}}\cdot\frac{1}{2}\left(\nabla^2 w^n + \nabla^2 w^{n+1}\right)$$

$$\Rightarrow \left(\frac{1}{\Delta t}I - \frac{1}{2\text{Re}}\nabla^2\right)w^{n+1} = -N^n + \left(\frac{1}{\Delta t}I + \frac{1}{2\text{Re}}\nabla^2\right)w^n$$

$\uparrow$

If discretized in space → linear system

Here: want to use spectral methods

Fourier Transform Properties:

$$F(f) = \hat{f}(k) = \int_{-\infty}^{+\infty} f(x) e^{-2\pi i k x} dx$$

$$F^{-1}(\hat{f}) = f(x) = \int_{-\infty}^{+\infty} \hat{f}(k) e^{2\pi i k x} dk$$

$$F(f') = 2\pi i k F(f)$$

$$F(\nabla^2 f) = -4\pi^2 |\vec{k}|^2 F(f) \quad \leftarrow \text{ 2D FT } (|\vec{k}|^2 = {k_x}^2 + {k_y}^2)$$

$$F(f * g) = F(f) \cdot F(g)$$

$$F(f \cdot g) = F(f) * F(g)$$

Use Here:

$$\left(\frac{1}{\Delta t} + \frac{4\pi^2}{2\mathrm{Re}}|\vec{k}|^2\right) \hat{w}^{n+1} = -\vec{N}^n + \left(\frac{1}{\Delta t} - \frac{4\pi^2}{2\mathrm{Re}}|\vec{k}|^2\right) \hat{w}^n$$

$$\Rightarrow \hat{w}^{n+1} = \frac{-\hat{N}^n + \left(\frac{1}{\Delta t} - \frac{2\pi^2}{\mathrm{Re}}|\vec{k}|^2\right) \hat{w}^n}{\frac{1}{\Delta t} + \frac{2\pi^2}{\mathrm{Re}}|\vec{k}|^2}$$

Computation of $\hat{N}$:

$$N = \vec{u} \cdot \nabla w \Rightarrow \hat{N} = \hat{\vec{u}} * \widehat{\nabla w}$$

Inefficient in Fourier space.
So perform multiplication in physical space.

Numerical Method:

Time step $\hat{w}^n \rightarrow \hat{w}^{n+1}$:

1. $\hat{\psi}^n = -\frac{1}{|\vec{k}|} \hat{w}^n$
2. $\begin{cases} \hat{u}^n = -2\pi i k_y \hat{\psi}^n \\ \hat{v}^n = 2\pi i k_x \hat{\psi}^n \\ \hat{w_x}^n = -2\pi i k_x \hat{w}^n \\ \hat{w_y}^n = -2\pi i k_y \hat{w}^n \end{cases}$
3. $\boxed{\text{iFFT}} \rightarrow \begin{cases} u^n \\ v^n \\ {w_x}^n \\ {w_y}^n \end{cases}$
4. $N^n = u^n \cdot {w_x}^n + v^n \cdot {w_y}^n$
5. $\boxed{\text{FFT}} \rightarrow (\hat{N}^n)^*$
6. Truncate $k_x, k_y > \frac{2}{3} k_{\max}$ to prevent aliasing

$$\hat{N}^n = \begin{cases} (\hat{N})^* & \text{if } k_x, k_y \leq \frac{2}{3} k_{\max} \\ 0 & \text{else} \end{cases}$$

7. $$\hat{w}^{n+1} = \frac{-\hat{N}^{n+1} + \left(\frac{1}{\Delta t} - \frac{2\pi^2}{\mathrm{Re}}|\vec{k}|^2\right) \hat{w}^n}{\frac{1}{\Delta t} + \frac{2\pi^2}{\mathrm{Re}}|\vec{k}|^2}$$

Particle Methods

Linear advection

$$\begin{cases} u_t + cu_x = 0 \\ u(x,0) = u_0(x) \end{cases}$$

Solution:
$u(x,t) = u_0(x - ct)$
Need to work hard to get good schemes for advection, on a fixed grid.

Reason: Represent sideways motion by up and down motion.

Alternative:
Move computed nodes, rather than having a fixed grid.

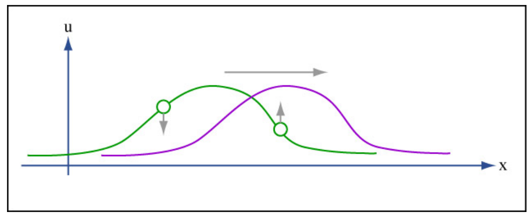

Image by MIT OpenCourseWare.

Ex.: 1D Advection

$$\begin{cases} u_t + cu_x = 0 \\ u(x,0) = u_0(x) \end{cases}$$

Particle method:
1. Sample initial conditions $x_j = j\Delta x, u_j = u_0(x_j)$
2. Move particles along characteristics:

$$\left\{ \begin{array}{l} \dot{x}_j = c \\ \dot{u}_j = 0 \end{array} \right\} \Rightarrow \begin{cases} x_j(t) = x_j(0) + ct \\ u_j(t) = u_0(x_j(0)) \end{cases} \rightarrow$$ Exact solution on particles.

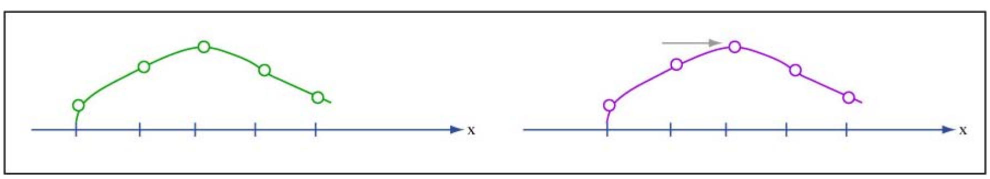

Image by MIT OpenCourseWare.

Ex.: 2D Advection

$$\begin{cases} \phi_t + \vec{v}(\vec{x}) \cdot \nabla\phi = 0 \\ \phi(\vec{x},0) = \phi_0(\vec{x}) \end{cases}$$

Particle method:

1. $\vec{x}_j, \phi_j = \phi_0(\vec{x}_j)$
2. $\left\{ \begin{array}{l} \dot{\vec{x}}_j = \vec{v}(\vec{x}_j) \\ \dot{\phi}_j = 0 \end{array} \right\}$

Solve characteristic ODE, e.g. by RK4; very accurate on particles.