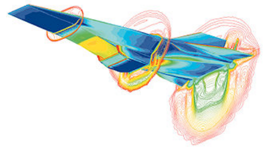
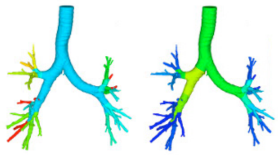
Hyper-X at Mach 7, NASA
image in the public domain

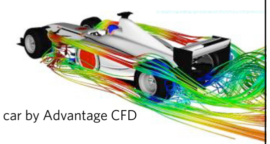


ME 702
Computational Fluid Dynamics, CFD

Prof. Lorena A. Barba



Human airways, by
FuiDA nv

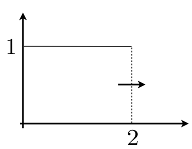


F1 car by Advantage CFD

BU

Burgers' equation —Debrief

- test problem—moving shock



$$u(x,0) = \begin{cases} 1 & : \quad x \in (0,2) \\ 0 & : \quad x \in (2,4) \end{cases}$$

$$\frac{\partial u}{\partial t} + u\,\frac{\partial u}{\partial x} = 0$$

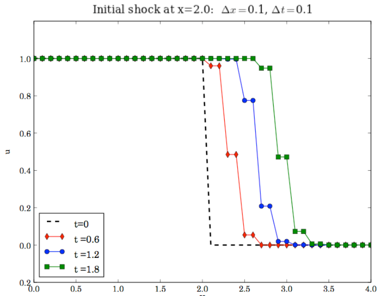
inviscid Burgers (hyperbolic):
a simple analog of the Euler equations

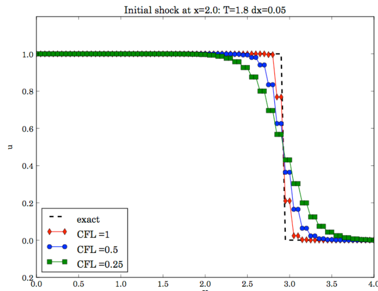
The discontinuity travels at $\frac{1}{2}(u_L + u_R)$

See Tannehill, Anderson & Pletcher, 2nd ed. (1997), p. 176 ff.

Burgers' equation **Lax-Friedrichs**

- What happens as time advances?
- What happens as you decrease CFL?

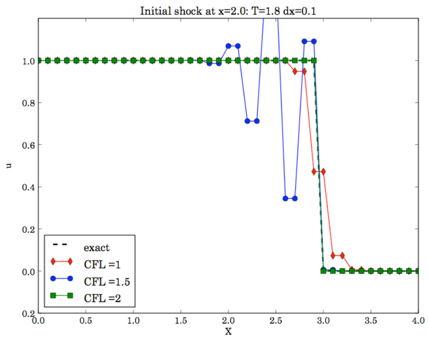




Burgers' equation **Lax-Friedrichs**

- Stability: $\left|\frac{\Delta t}{\Delta x} u_{\max}\right| < 1$

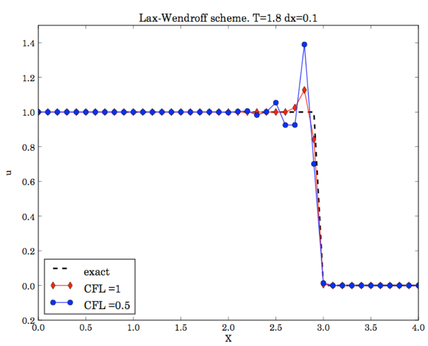
- What happens with CFL = 2 ?



Burgers' equation **Lax-Wendroff**

- Stability: $\left|\frac{\Delta t}{\Delta x} u_{\max}\right| < 1$

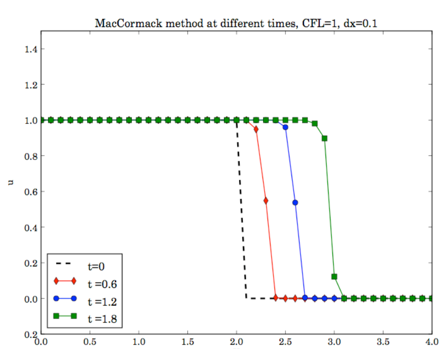
- What happens as you decrease CFL?



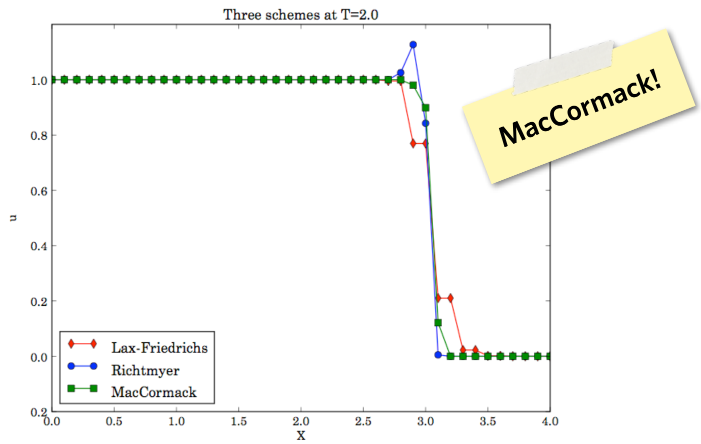
Burgers' equation **Mac-Cormack**

- Stability: $\left|\frac{\Delta t}{\Delta x} u_{\max}\right| < 1$

The solution is very well-behaved with CFL=1



Burgers' equation **Which scheme is best?**



Burgers' equation **Beam-Warming implicit**

- This implicit method is unconditionally stable.
- For nonlinear convection:
  - the scheme is very dispersive / it has no inherent dissipation
- What is the effect of explicit damping?
  - Note: 4th-order damping term does not alter overall order of accuracy

Burgers' equation **Beam-Warming implicit**

- recall, the FD equation is:

$$-\frac{\Delta t}{4\Delta x}u^n_{i-1}u^{n+1}_{i-1}+u^{n+1}_i+\frac{\Delta t}{4\Delta x}u^n_{i+1}u^{n+1}_{i-1}=u^n_i$$

- write as $a_i\, u^{n+1}_{i-1}+b_i\, u^{n+1}_i+c_i\, u^{n+1}_{i-1}=d_i$
  - Thomas algorithm for tri-diagonal systems ...

Burgers' equation **Beam-Warming implicit**

Thomas algorithm $a_i\, u_{i-1}^{n+1} + b_i\, u_i^{n+1} + c_i\, u_{i-1}^{n+1} = d_i$

- **Note**: wikipedia entry for Thomas algorithm assumes $a_1 = 0$ and $c_n = 0$ then writes the system as:

$$\begin{bmatrix} b_1 & c_1 & & & 0 \\ a_2 & b_2 & c_2 & & \\ & a_3 & b_3 & \ddots & \\ & & \ddots & \ddots & c_{n-1} \\ 0 & & & a_n & b_n \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \\ d_3 \\ \vdots \\ d_n \end{bmatrix}.$$

forward elimination adjusts the upper diagonals:

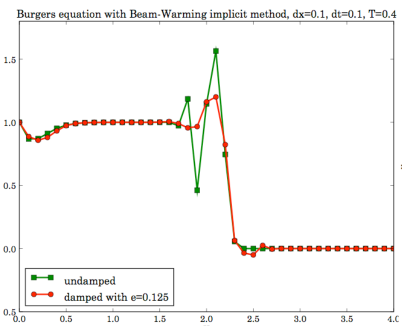
$$c'_i = \begin{cases} \dfrac{c_1}{b_1} & ; \quad i = 1 \\ \dfrac{c_i}{b_i - c'_{i-1} a_i} & ; \quad i = 2, 3, \dots, n-1 \end{cases}$$

$$d'_i = \begin{cases} \dfrac{d_1}{b_1} & ; \quad i = 1 \\ \dfrac{d_i - d'_{i-1} a_i}{b_i - c'_{i-1} a_i} & ; \quad i = 2, 3, \dots, n. \end{cases}$$

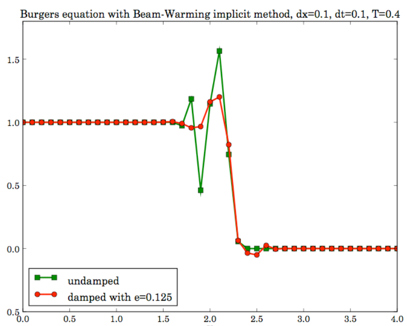
- ... and backward elimination gives the solution.

Burgers' equation **Beam-Warming implicit**

- but what if $a_1 \neq 0$ due to boundary conditions?
  Say, $u_0 = B_0$



Without modifying the algorithm for left-BC.



Modifying the RHS with $d_1 - a_1 B_0$

Euler equations

in conservation form (1D):

$$\frac{\partial \rho}{\partial t} + \frac{\partial (\rho u)}{\partial x} = 0 \qquad \text{Continuity}$$

$$\frac{\partial (\rho u)}{\partial t} + \frac{\partial}{\partial x}(\rho u^2 + p) = 0 \qquad \text{Conservation of Momentum}$$

$$\frac{\partial}{\partial t}(\rho e_T) + \frac{\partial}{\partial x}(\rho u e_T + pu) = \frac{\partial}{\partial t}(\rho e_T) + \frac{\partial}{\partial x}(\rho u h_T) = 0 \qquad \text{Energy}$$

Euler equations **Vector form**

Also called "short form."

- Column vector of conserved quantities $\vec{u} = \begin{bmatrix} \rho \\ \rho u \\ \rho e_T \end{bmatrix}$
- and a flux vector

$$\vec{f} = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ (\rho e_T + p)u \end{bmatrix} = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho h_T u \end{bmatrix}$$

- Euler equations in conservation form: $\dfrac{\partial \vec{u}}{\partial t} + \dfrac{\partial \vec{f}}{\partial x} = 0$

- Note that the flux vector can be written as a function of the conserved quantities, so:

$$\frac{\partial \vec{f}}{\partial x} = \frac{\partial \vec{f}}{\partial \vec{u}} \frac{\partial \vec{u}}{\partial x}$$

where,

$$\frac{\partial \vec{f}}{\partial \vec{u}} = \begin{pmatrix} \frac{\partial f_1}{\partial u_1} & \frac{\partial f_1}{\partial u_2} & \frac{\partial f_1}{\partial u_3} \\ \frac{\partial f_2}{\partial u_1} & \frac{\partial f_2}{\partial u_2} & \frac{\partial f_2}{\partial u_3} \\ \frac{\partial f_3}{\partial u_1} & \frac{\partial f_3}{\partial u_2} & \frac{\partial f_3}{\partial u_3} \end{pmatrix}$$

is called the Jacobian matrix, A

- which allows us to write:

... representing the full Euler eqns.

$$\boxed{\frac{\partial \vec{u}}{\partial t} + A\frac{\partial \vec{u}}{\partial x} = 0}$$

Looks a lot like the scalar convection equation!

Euler equations **Discretization**

- e.g., using a CD formula for the vector function:

$$\frac{\partial \vec{f}}{\partial x}(x_i) = \frac{\vec{f}(x_{i+1}) - \vec{f}(x_{i-1})}{2\Delta x} + \mathcal{O}(\Delta x^2)$$

- simply replaced scalar u by the vector $\vec{f}$ in a well-known FD formula.
- Similarly, replace scalar derivative by Jacobian matrix ...

$$a = \frac{\mathrm{d}f}{\mathrm{d}u} \quad \longrightarrow A = \frac{\mathrm{d}\vec{f}}{\mathrm{d}\vec{u}}$$

Euler equations **Lax-Friedrichs**

- Use your notes for LF scheme on Burgers equation to develop the scheme for Euler equations!