Finite Volume Method

Praveen. C
Computational and Theoretical Fluid Dynamics Division
National Aerospace Laboratories
Bangalore 560 017
email: praveen@cfdlab.net

Workshop on Advances in Computational Fluid Flow and Heat Transfer
Annamalai University

October 17-18, 2005

Topics to be covered

1. Conservation Laws
2. Finite volume method
3. Types of finite volumes
4. Flux functions
5. Spatial discretization schemes
6. Higher order schemes
7. Boundary conditions
8. Accuracy and stability
9. Computational issues
10. References

Hyperbolic equations, Compressible flow, unstructured grid schemes

Conservation Laws and FVM

- Basic laws of physics are conservation laws - mass, momentum, energy
- Differential form

$$\frac{\partial U}{\partial t} + \frac{\partial f}{\partial x} + \frac{\partial g}{\partial y} + \frac{\partial h}{\partial z} = 0$$

  U - conserved variables

  f, g, h - flux vector

- Compressible flows - shocks and other discontinuities
- Classical solution may not exist
- Integral form (using divergence theorem)

$$\frac{\partial}{\partial t}\int_\Omega U \mathrm{d}x\mathrm{d}y\mathrm{d}z + \oint_{\partial\Omega}(fn_x + gn_y + hn_z)\mathrm{d}S = 0$$

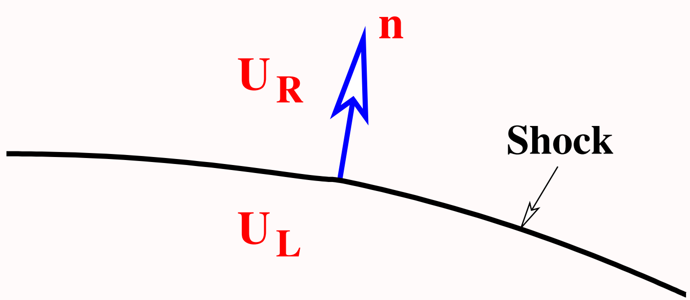
Rate of change of U in Ω = - (Net flux across the boundary of Ω)

$\Downarrow$

Starting point for finite volume method

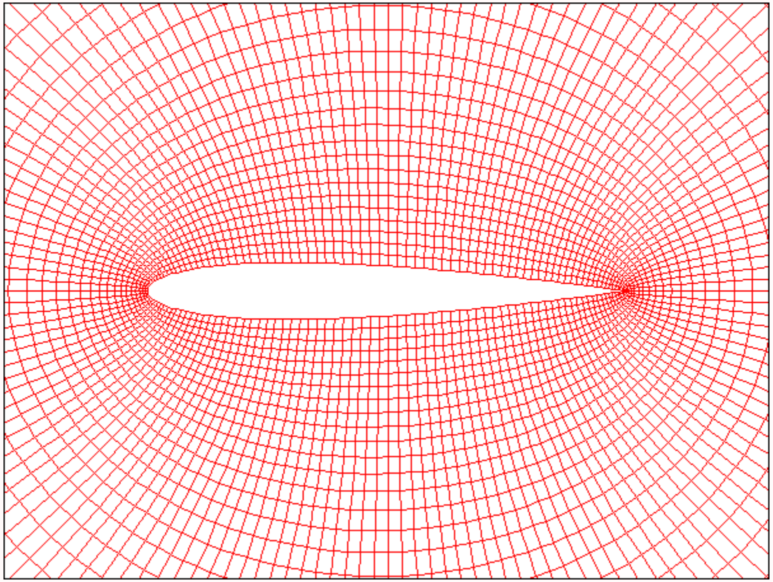
- Discontinuities are a consequence of conservation laws
- Rankine-Hugoniot jump conditions [9, 10]

$$(fn_x + gn_y + hn_z)_R - (fn_x + gn_y + hn_z)_L = s(U_R - U_L)$$

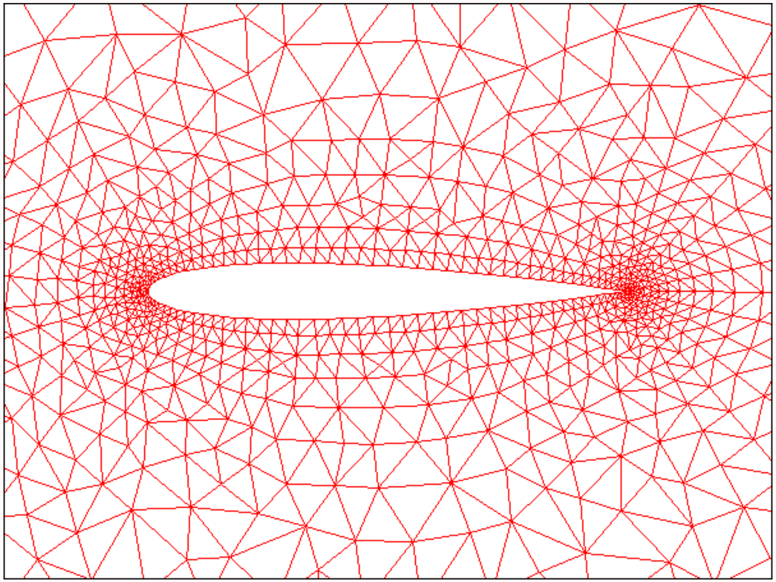


- Solution satisfying integral form - *weak solution*
- Definition (Weak solution)
  1. Satisfies the differential form in smooth regions
  2. Satisfies jump condition across discontinuities
- Hyperbolic conservation laws - non-uniqueness
- Limit of a dissipative model: Navier-Stokes → Euler

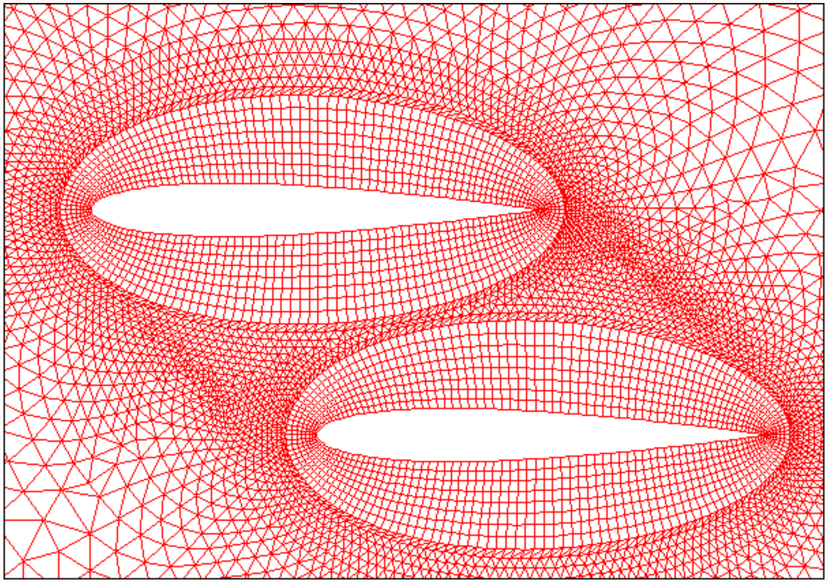
- Entropy condition - second law of thermodynamics
- Entropy satisfying weak solution - unique (Kruzkov)
- Conservative scheme (FVM) - correct shock location (Warnecke)
- Useful for solving equations with discontinuous coefficients
- FVM can be applied on arbitrary grids - structured and unstructured



- Entropy condition - second law of thermodynamics
- Entropy satisfying weak solution - unique (Kruzkov)
- Conservative scheme (FVM) - correct shock location (Warnecke)
- Useful for solving equations with discontinuous coefficients
- FVM can be applied on arbitrary grids - structured and unstructured



- Entropy condition - second law of thermodynamics
- Entropy satisfying weak solution - unique (Kruzkov)
- Conservative scheme (FVM) - correct shock location (Warnecke)
- Useful for solving equations with discontinuous coefficients
- FVM can be applied on arbitrary grids - structured and unstructured

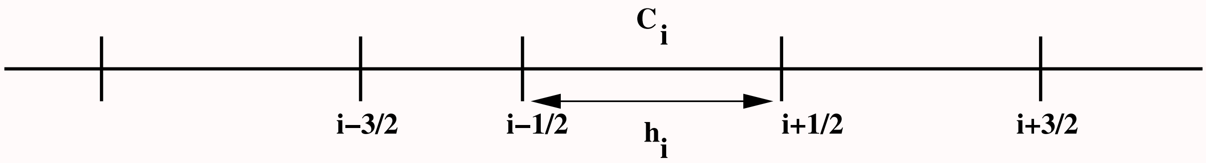


FVM in 1-D

- Divide computational domain $[a, b]$ into N cells

$$a = x_{1/2} < x_{3/2} < \ldots < x_{N+1/2} = b$$

$$C_i = [x_{i-1/2}, x_{i+1/2}]$$



- Conservation law for cell C_i

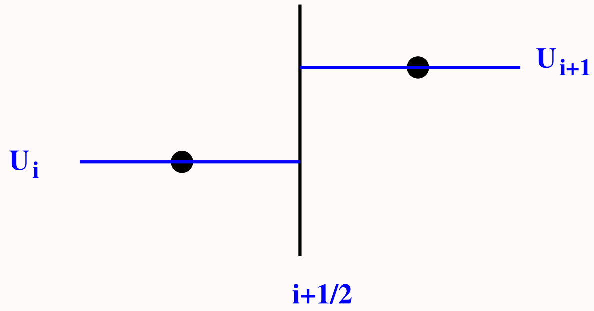
$$\frac{\partial}{\partial t} \int_{x_{i-1/2}}^{x_{i+1/2}} U \mathrm{d}x + f(x_{i+1/2}, t) - f(x_{i-1/2}, t) = 0$$

- Cell average value

$$U_i(t) = \frac{1}{h_i} \int_{x_{i-1/2}}^{x_{i+1/2}} U(x, t) \mathrm{d}x$$

- Conservation law for cell C_i

$$h_i \frac{\mathrm{d}U_i}{\mathrm{d}t} + f(x_{i+1/2}, t) - f(x_{i-1/2}, t) = 0$$



- *Riemann problem* at each interface
- *Numerical flux function* (Godunov approach)

$$F_{i+1/2}(t) = F(U_i(t), U_{i+1}(t))$$

- Semi-discrete update equation (ODE system)

$$\frac{\mathrm{d}U_i}{\mathrm{d}t} = -\frac{1}{h_i}[F_{i+1/2}(t) - F_{i-1/2}(t)]$$

- *Method of lines* approach
  - Discretize in space
  - Integrate the ODE system in time
- Explicit Euler scheme [$U_i^n \approx U(x_i, t^n)$]

$$\frac{U_i(t^{n+1}) - U_i(t^n)}{\Delta t} = -\frac{1}{h_i}[F_{i+1/2}(t^n) - F_{i-1/2}(t^n)]$$

$$\Downarrow$$

$$U_i^{n+1} = U_i^n - \frac{\Delta t}{h_i}[F_{i+1/2}^n - F_{i-1/2}^n]$$

- Conservation: Telescopic collapse of fluxes

$$\begin{aligned} \sum_i h_i \frac{\mathrm{d}U_i}{\mathrm{d}t} &= -\sum_i [F_{i+1/2}(t) - F_{i-1/2}(t)] \\ &= -[f(b,t) - f(a,t)] \end{aligned}$$

Numerical Flux Function

- Simple averaging

$$F_{i+1/2} = f((U_i + U_{i+1})/2) \quad \text{or} \quad F_{i+1/2} = (f_i + f_{i+1})/2$$

- Equivalent to central differencing

$$\frac{\mathrm{d}U_i}{\mathrm{d}t} + \frac{1}{h_i}(f_{i+1} - f_{i-1}) = 0 \quad \text{(unstable)}$$

- Two approaches

1. *Central* differencing with artificial dissipation [13]

$$F_{i+1/2} = \frac{1}{2}(f_i + f_{i+1}) - d_{i+1/2}$$

2. *Upwind* flux formula [9, 10, 13, 20, 22]

$$\begin{aligned} \text{FVS: } F_{i+1/2} &= f^+(U_i) + f^-(U_{i+1}) \\ \text{FDS: } F_{i+1/2} &= \frac{1}{2}(f_i + f_{i+1}) - \frac{1}{2}[(\Delta f)^-_{i+1/2} - (\Delta f)^+_{i+1/2}] \end{aligned}$$

- Example: convection-diffusion equation

$$\frac{\partial U}{\partial t} + \frac{\partial f}{\partial x} = 0, \quad f = aU - \nu \frac{\partial U}{\partial x}$$

$$F_{i+1/2} = aU_{i+1/2} - \nu \left. \frac{\partial U}{\partial x} \right|_{i+1/2}$$

- Upwind definition of interfacial state

$$U_{i+1/2} = \begin{cases} U_i & \text{if } a \geq 0 \\ U_{i+1} & \text{if } a < 0 \end{cases}$$

- Central-difference for viscous term

$$\left. \frac{\partial U}{\partial x} \right|_{i+1/2} = \frac{U_{i+1} - U_i}{x_{i+1} - x_i}$$

- Upwind numerical flux

$$F_{i+1/2} = \frac{1}{2}(aU_i + aU_{i+1}) - \frac{|a|}{2}(U_{i+1} - U_i) - \nu \frac{U_{i+1} - U_i}{x_{i+1} - x_i}$$

Significance of conservative scheme

- Inviscid Burgers equation

$$\frac{\partial U}{\partial t} + \frac{\partial}{\partial x}\left(\frac{U^2}{2}\right) = 0, \quad f(U) = \frac{U^2}{2}$$

- Rankine-Hugoniot condition

$$f_R - f_L = s(U_R - U_L) \Longrightarrow s = \frac{1}{2}(U_L + U_R)$$

- Non-conservative form

$$\frac{\partial U}{\partial t} + U\frac{\partial U}{\partial x} = 0$$

- *Upwind* scheme (assume $U \geq 0$)

$$\frac{U_i^{n+1} - U_i^n}{\Delta t} + U_i^n \frac{U_i^n - U_{i-1}^n}{h} = 0$$

or

$$U_i^{n+1} = U_i^n - \frac{\Delta t}{h} U_i^n (U_i^n - U_{i-1}^n)$$

- Initial condition

$$U(x,0) = \begin{cases} 1 & \text{if } x < 0 \\ 0 & \text{if } x > 0 \end{cases}$$

- Numerical solution

$$U_i^n = U_i^o \Longrightarrow \text{ stationary shock}$$

- Exact solution (shock speed $= 1/2$)

$$U(x,t) = \begin{cases} 1 & \text{if } x < t/2 \\ 0 & \text{if } x > t/2 \end{cases}$$

- Conservation form from physical considerations

$$U\frac{\partial U}{\partial t} + U\frac{\partial}{\partial x}\left(\frac{U^2}{2}\right) = 0$$

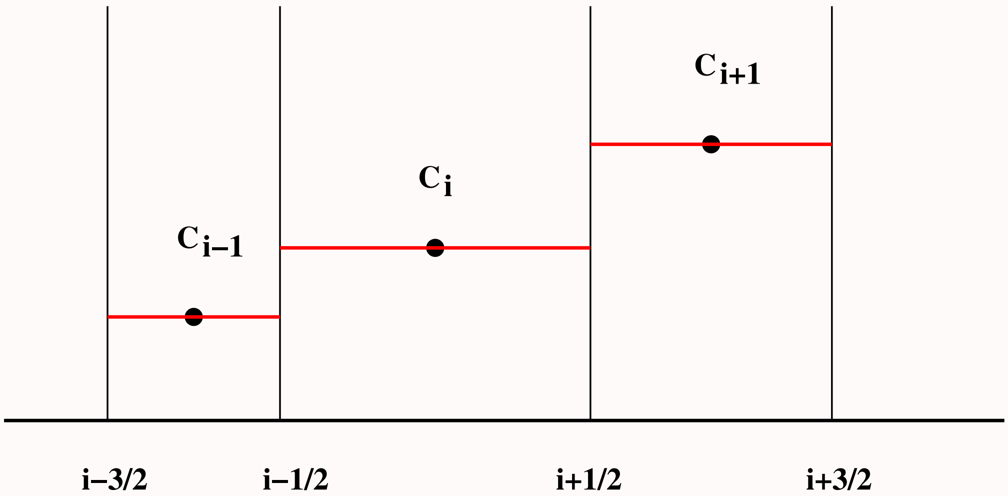
or

$$\frac{\partial}{\partial t}\left(\frac{U^2}{2}\right) + \frac{\partial}{\partial x}\left(\frac{U^3}{3}\right) = 0$$

- Jump conditions not identical: $s = \frac{2}{3}\left(\frac{U_L^2 + U_L U_R + U_R^2}{U_L + U_R}\right)$

Higher order scheme in 1-D

- Constant-in-cell representation



- First order accurate

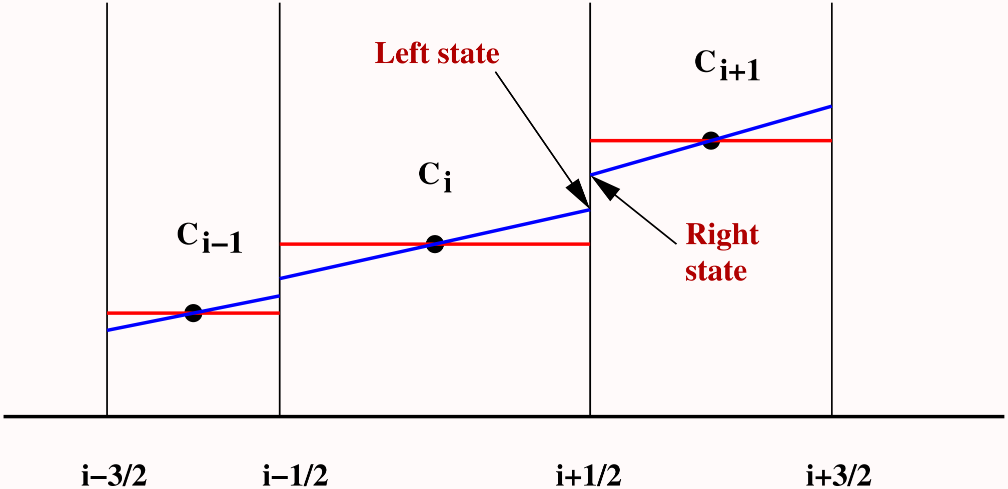
$$|U_i - U(x_i)| = O(h)$$

$$h = \max_i h_i$$

- Reconstruction - evolution - projection

Higher order scheme in 1-D

- Reconstruct the variation within a cell



- Linear reconstruction

$$\tilde{U}(x) = U_i + s_i(x - x_i), \quad x \in [x_{i-1/2}, x_{i+1/2}]$$

- Biased interpolant

$$U^L_{i+1/2} = U_i + s_i(x_{i+1/2} - x_i), \quad U^R_{i+1/2} = U_{i+1} + s_{i+1}(x_{i+1/2} - x_{i+1}),$$

- Flux for higher order scheme

$$F_{i+1/2} = F(U_i, U_{i+1})$$

- Reconstruction variables

    1. Conserved variables - conservative
    2. Characteristic variables - better upwinding but costly
    3. Primitive variables (ρ, u, p) - computationally cheap

- Unsteady flows - reconstruction must preserve conservation

$$\frac{1}{h_i}\int_{C_i} \tilde{U}(x)\mathrm{d}x = U_i$$

- Gradients for reconstruction: backward, forward, central difference

$$s_{i,b} = \frac{U_i - U_{i-1}}{x_i - x_{i-1}}, \quad s_{i,f} = \frac{U_{i+1} - U_i}{x_{i+1} - x_i}, \quad s_{i,c} = \frac{U_{i+1} - U_{i-1}}{x_{i+1} - x_{i-1}}$$

- Flux for higher order scheme

$$F_{i+1/2} = F(U^L_{i+1/2}, U^R_{i+1/2})$$

- Reconstruction variables

  1. Conserved variables - conservative
  2. Characteristic variables - better upwinding but costly
  3. Primitive variables (ρ, u, p) - computationally cheap

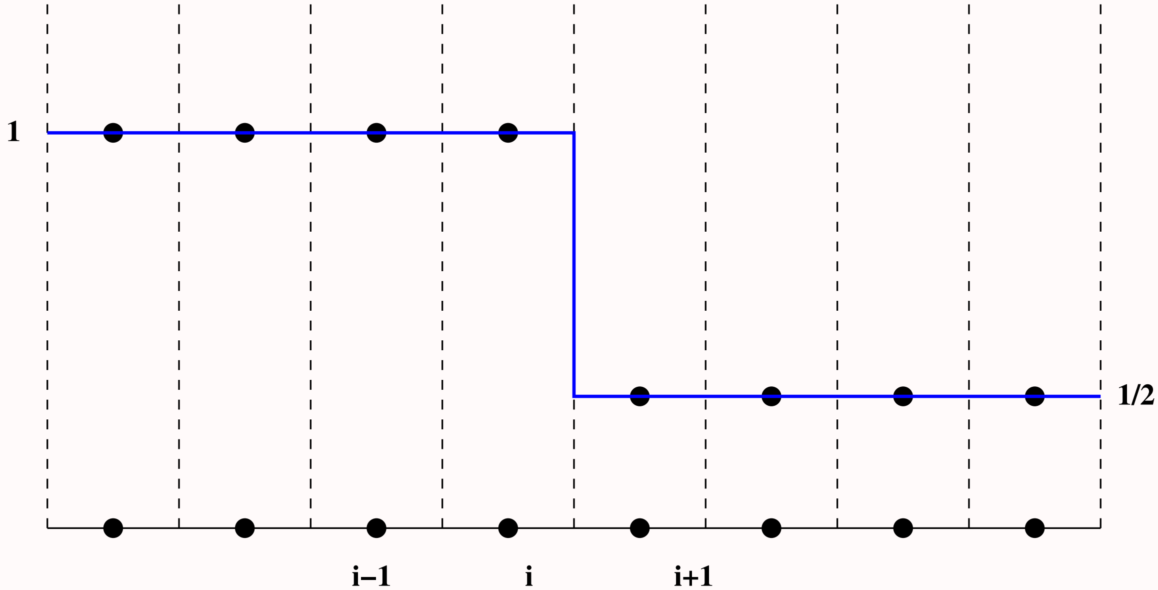
- Unsteady flows - reconstruction must preserve conservation

$$\frac{1}{h_i}\int_{C_i} \tilde{U}(x)\mathrm{d}x = U_i$$

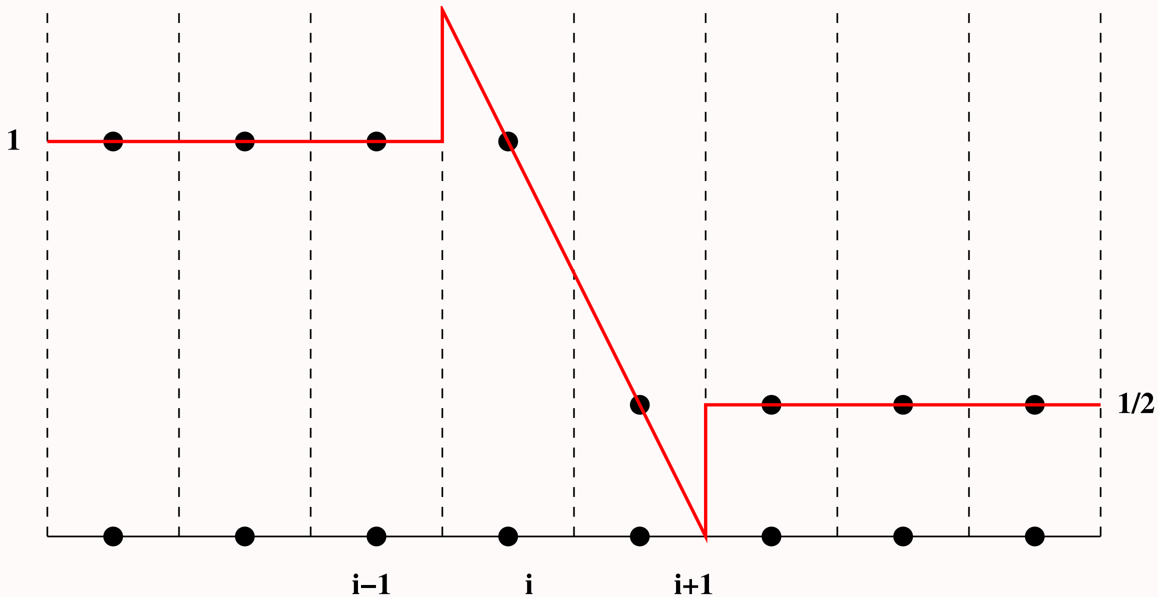
- Gradients for reconstruction: backward, forward, central difference

$$s_{i,b} = \frac{U_i - U_{i-1}}{x_i - x_{i-1}}, \quad s_{i,f} = \frac{U_{i+1} - U_i}{x_{i+1} - x_i}, \quad s_{i,c} = \frac{U_{i+1} - U_{i-1}}{x_{i+1} - x_{i-1}}$$

- Solution with discontinuity



- Central-difference: Non-monotone reconstruction



- Limited gradients [9, 12]

$$s_i = \text{Limiter}(s_{i,b}, s_{i,f}, s_{i,c})$$

FVM in 2-D

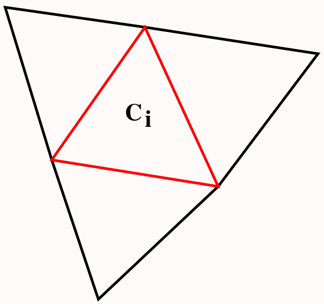
- Divide computational domain into disjoint polygonal cells, $\Omega = \cup_i C_i$
- Integral form for cell C_i

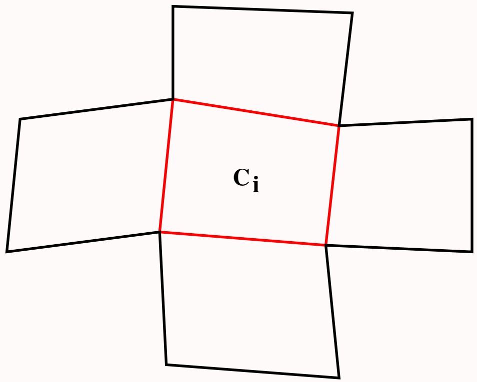
$$\frac{\partial}{\partial t}\int_{C_i} U \mathrm{d}x\mathrm{d}y + \oint_{\partial C_i} (f n_x + g n_y)\mathrm{d}S = 0$$

- Cell average value

$$U_i(t) = \frac{1}{|C_i|}\int_{C_i} U(x,y,t)\mathrm{d}x\mathrm{d}y, \quad |C_i| = \text{area of } C_i$$

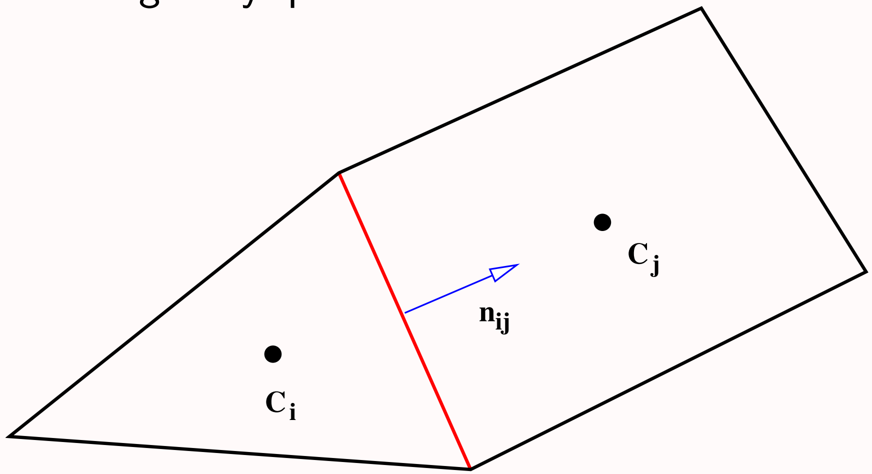
- Cell connectivity: $N(i) = \{j : C_j \text{ and } C_i \text{ share a common face}\}$





$$\oint_{\partial C_i}(fn_x + gn_y)\mathrm{d}S = \sum_{j\in N(i)} \int_{C_i \cap C_j}(fn_x + gn_y)\mathrm{d}S$$

- Approximate flux integral by quadrature



$$\int_{C_i \cap C_j}(fn_x + gn_y)\mathrm{d}S \approx F_{ij}\Delta S_{ij}$$

- Semi-discrete update equation

$$|C_i|\frac{\mathrm{d}U_i}{\mathrm{d}t} = -\sum_{j\in N(i)} F_{ij}\Delta S_{ij}$$

- Numerical flux function

$$F_{ij} = F(U_i, U_j, \hat{n}_{ij})$$

- Properties of flux function

1. Consistency

$$F(U, U, \hat{n}) = f(U) n_x + g(U) n_y$$

2. Conservation

$$F(V, U, -\hat{n}) = -F(U, V, \hat{n})$$

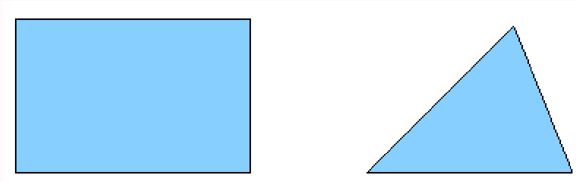
3. Continuity

$$\| F(U_L, U_R, \hat{n}) - F(U, U, \hat{n}) \| \le C \max \left(\| U_L - U \|, \| U_R - U \| \right)$$

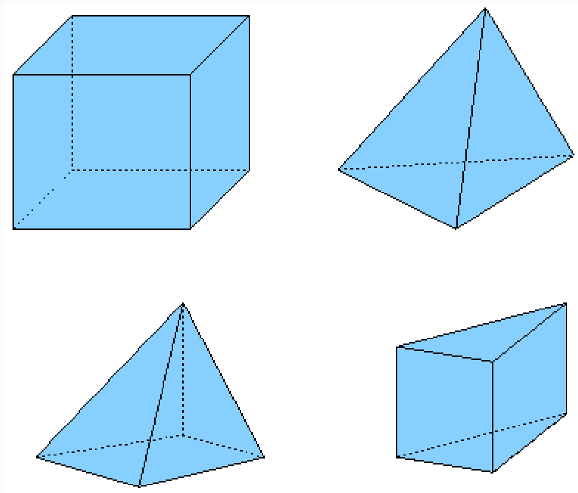
- Flux functions [10, 13, 20, 22]
  - FVS: Steger-Warming, Van Leer, KFVS, AUSM
  - FDS: Godunov, Roe, Engquist-Osher
- Integrate in time using a Runge-Kutta scheme [5, 12]

Grids and Finite Volumes

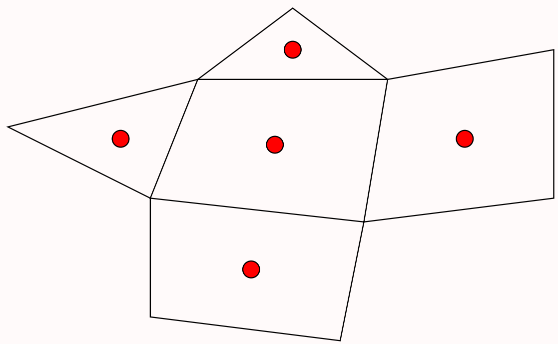
- Elements in 2-D

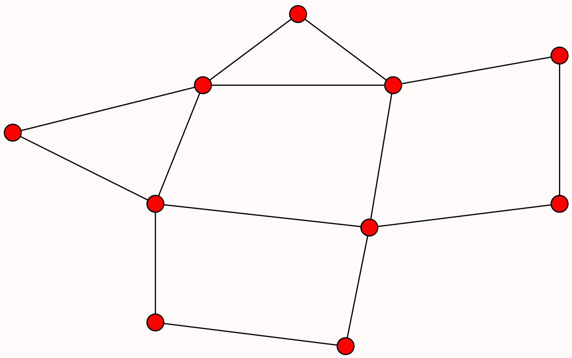


- Elements in 3-D

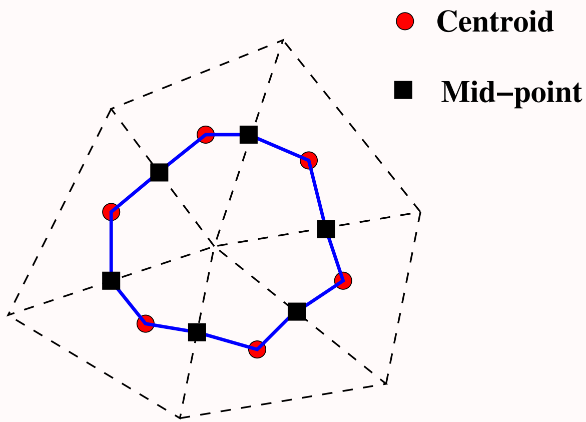


- Boundary layers - prism and hexahedra
- Cell-centered and vertex-centered scheme [5, 18, 21]

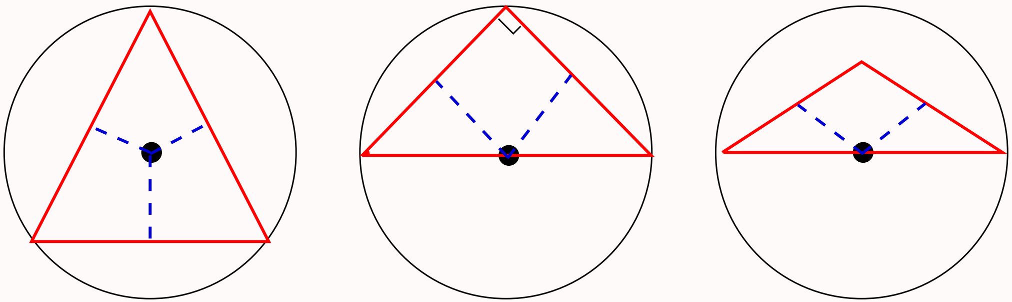




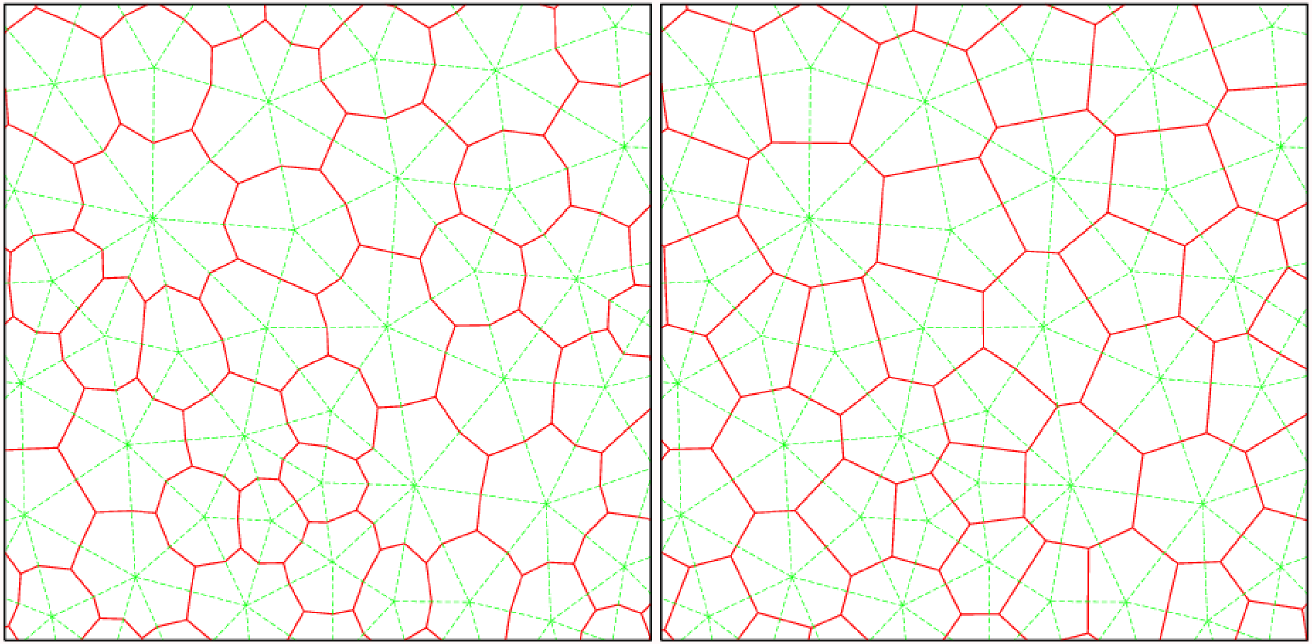
- Median (dual) cell
  - join centroid to mid-point of sides
  - well-defined for any triangulation



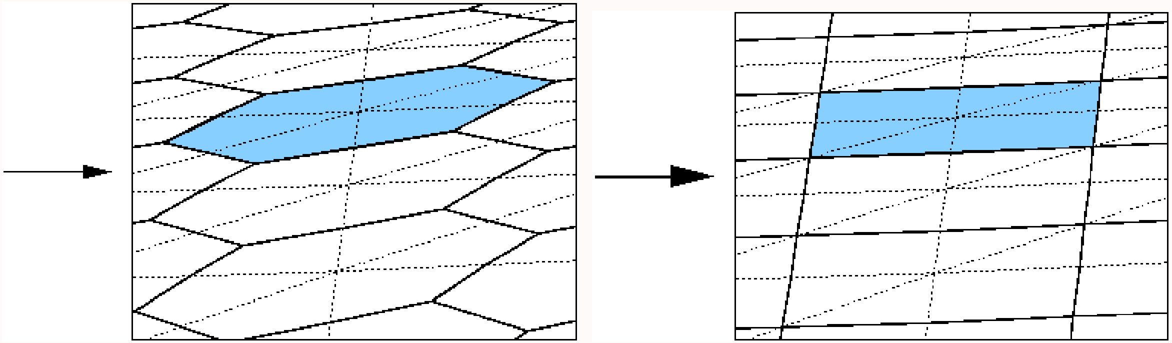
- Voronoi cell
  - join circum-center to mid-point of sides
  - smooth area variation
  - not defined for obtuse triangles
- Containment circle tessalation



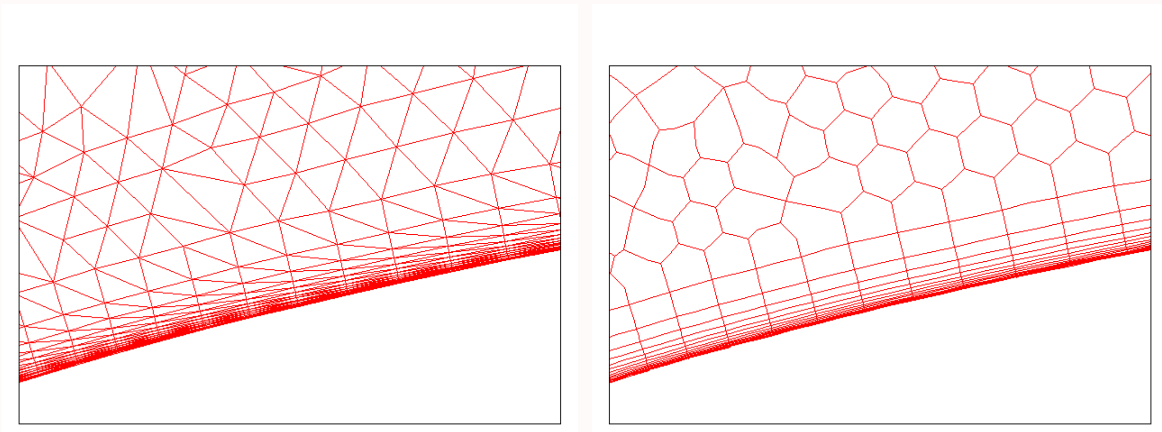
Median and containment-circle tessalation



- Stretched triangles - median dual and containment-circle

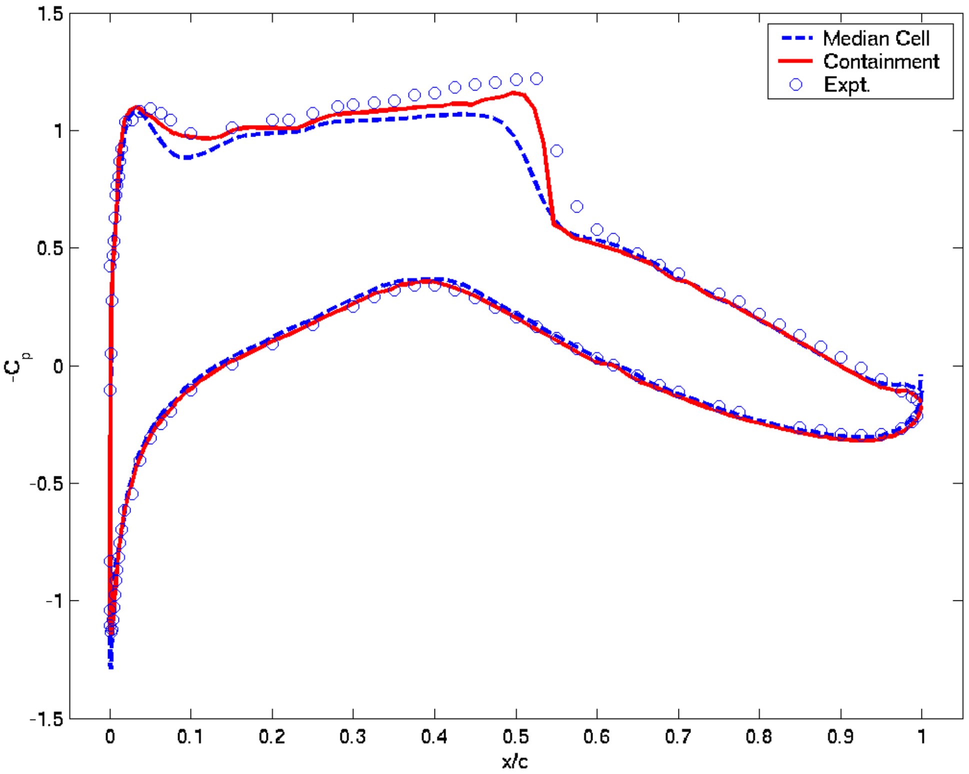


- Containment-circle finite volume



- Turbulent flow over RAE2822 airfoil: vertex-centered scheme

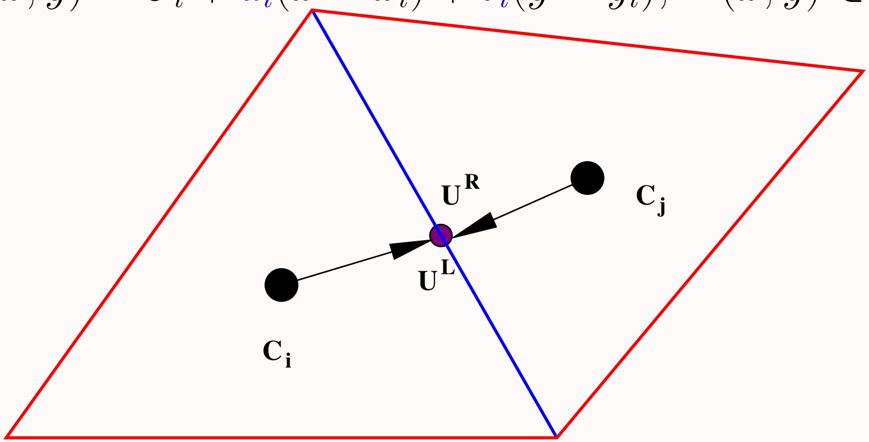
Mach $= 0.729$, $\alpha = 2.31$ deg, Re $= 6.5$ million



Higher order scheme in 2-D

- Bi-linear reconstruction in cell C_i

$$\tilde{U}(x,y) = U_i + a_i(x - x_i) + b_i(y - y_i), \quad (x,y) \in C_i$$



- Define left/right states

$$\begin{aligned} U^L &= U_i + a_i(x_{ij} - x_i) + b_i(y_{ij} - y_i) \\ U^R &= U_j + a_j(x_{ij} - x_j) + b_j(y_{ij} - y_j) \end{aligned}$$

- Flux for higher order scheme

$$F_{ij} = F(U^L, U^R, \hat{n}_{ij})$$

- Gradient estimation using
  1. Green-Gauss theorem
  2. Least squares fitting
- Green-Gauss theorem

$$\int_{C_i} \nabla U \mathrm{d}x\mathrm{d}y = \oint_{\partial C_i} U\hat{n}\mathrm{d}S$$

- Approximate surface integral by quadrature

$$\nabla U_i \approx \frac{1}{|C_i|}\sum_{\text{face}} \int_{\text{face}} U\hat{n}\mathrm{d}S$$

- Face value

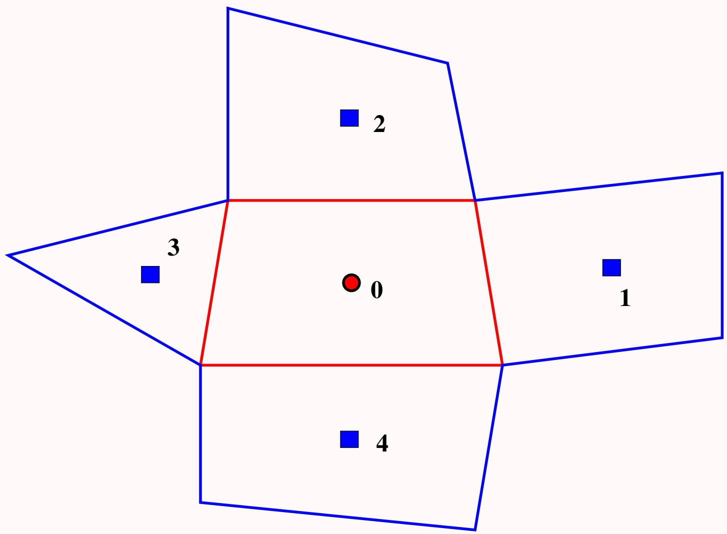
$$U_{\text{face}} = \frac{1}{2}(U_L + U_R)$$

- Non-uniform cells

$$U_{\text{face}} = \alpha U_L + (1-\alpha)U_R, \quad \alpha \in (0,1)$$

- Accuracy can degrade for non-uniform grids [4, 6, 8, 14]

- Least-squares reconstruction [3, 5]



$$U_o + a_o(x_j - x_o) + b_o(y_j - y_o) = U_j, \quad j = 1, 2, 3, 4$$

- Over-determined system of equations - solve by least-squares fit

$$\min \sum_j [U_j - U_o - a_o(x_j - x_o) - b_o(y_j - y_o)]^2, \quad \text{wrt } a_o, b_o$$

$$a_o = \sum_j \alpha_j (U_j - U_o), \quad b_o = \sum_j \beta_j (U_j - U_o)$$

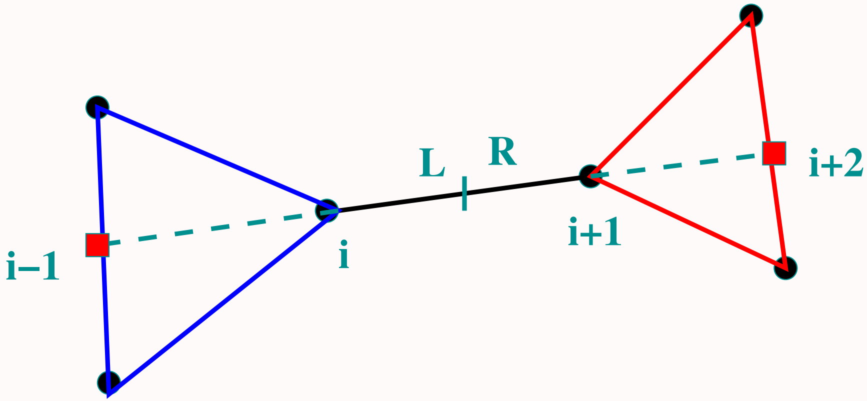
- Limited reconstruction
  - Cell-centered: Min-max [3, 5], Venkatakrishnan [5], ENO-type [1, 14]
  - Vertex-centered: edge-based limiter [17]
- Min-max limiter

$$U_{\min} \le U_o + a_o(x_j - x_o) + b_o(y_j - y_o) \le U_{\max}, \quad j = 1, 2, 3, 4$$

$$(a_o, b_o) \longleftarrow (\phi a_o, \phi b_o), \quad \phi \in [0, 1]$$

  - Very dissipative - smeared shocks
  - Performance degrades on coarse grids
  - Stalled convergence - limit cycle
  - Useful for flows with large discontinuities
- Venkatakrishnan limiter
  - Smooth modification of min-max limiter
  - Better control - depends on cell size
  - Better convergence properties

- Vertex-centered cell: Edge-based limiter



$$U^L = U_i + \frac{1}{2}\text{Limiter}\left[(U_{i+1} - U_i), \frac{|P_i P_{i+1}|}{|P_i P_{i-1}|}(U_i - U_{i-1})\right]$$

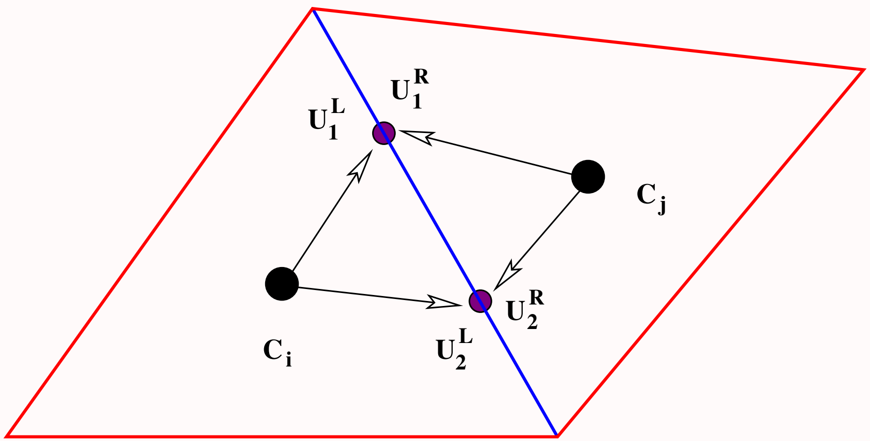
- Using vertex-gradients

$$U^L = U_i + \frac{1}{2}\text{Limiter}\left[(U_{i+1} - U_i), (\vec{P}_{i+1} - \vec{P}_i)\cdot\nabla U_i\right]$$

- Van-albada limiter

$$\text{Limiter}(a,b) = \frac{(a^2+\epsilon)b + (b^2+\epsilon)a}{a^2+b^2+2\epsilon}, \quad \epsilon \ll 1$$

Higher order flux quadrature



- Quadratic reconstruction in cell C_i

$$\tilde{U}(x,y) = \tilde{U}_i + a_i(x-x_i) + b_i(y-y_i) + c_i(x-x_i)^2 + d_i(x-x_i)(y-y_i) + e_i(y-y_i)^2$$

- 2-point Gauss quadrature for flux

$$F_{ij} = \omega_1 F(U_1^L, U_1^R, \hat{n}_{ij}) + \omega_2 F(U_2^L, U_2^R, \hat{n}_{ij})$$

Discretization of viscous flux

- Viscous terms

$$\nabla \cdot \mu \nabla u$$

- Finite volume discretization

$$\int_{C_i} (\nabla \cdot \mu \nabla u) \mathrm{d}V = \oint_{\partial C_i} (\mu \nabla u \cdot \hat{n}) \mathrm{d}S$$

- Simple averaging

$$\nabla u_{ij} = \frac{1}{2}(\nabla u_i + \nabla u_j)$$

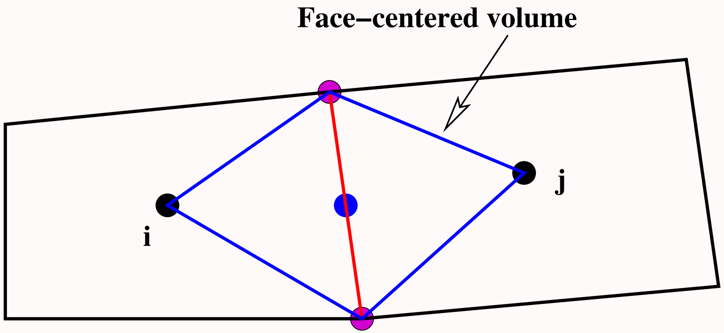
  - Odd-even decoupling on quadrilateral/hexahedral cells
  - Large stencil size

- 1-D case: $u_t = u_{xx}$

$$u_i^{n+1} = u_i^n + \frac{\Delta t}{2h}(u_{i-2}^n - 2u_i^n + u_{i+2}^n)$$

- Correction for decoupling problem [5]

- Green-Gauss theorem for auxiliary volume



- Least-squares gradients
  - Quadratic reconstruction: gradients and hessian [3]
  - Face-centered least-squares
- Vertex-centered scheme
  - Galerkin approximation on triangles/tetrahedra
  - Nearest neighbour stencil

Turbulence models

- Reynolds-average Navier-Stokes equations - need turbulence models
- Differential equation based models: $k-\epsilon$, $k-\omega$, Spalart-Allmaras
- Turbulence quantities must remain positive
- Discretize using first order upwind finite volume method
  Example: Spalart-Allmaras model

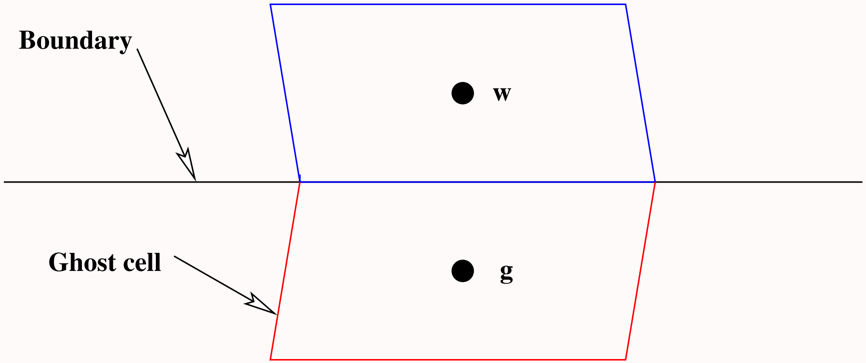
$$\int_{C_i} \nabla \cdot (\tilde{\nu} u) \mathrm{d}V \approx \sum_{j \in N(i)} [(u_{ij} \cdot \hat{n}_{ij})^+ \tilde{\nu}_i + (u_{ij} \cdot \hat{n}_{ij})^- \tilde{\nu}_j] \Delta S_{ij}$$

$$(\cdot)^{\pm} = \frac{(\cdot) \pm |(\cdot)|}{2}, \quad u_{ij} = \frac{1}{2}(u_i + u_j)$$

- Coupled or de-coupled approach
- Stiffness problem - positivity preserving implicit methods

Boundary conditions

- Cell-centered approach
  1. Ghost cell
  2. Flux boundary condition



- Inviscid flow (slip flow - zero normal velocity)

$$\rho_g = \rho_w, \quad p_g = p_w, \quad u_g = u_w, \quad v_g = -v_w$$

- Viscous flow (noslip flow - zero velocity)

$$\rho_g = \rho_w, \quad p_g = p_w, \quad u_g = -u_w, \quad v_g = -v_w$$

- Boundary flux depends on pressure only

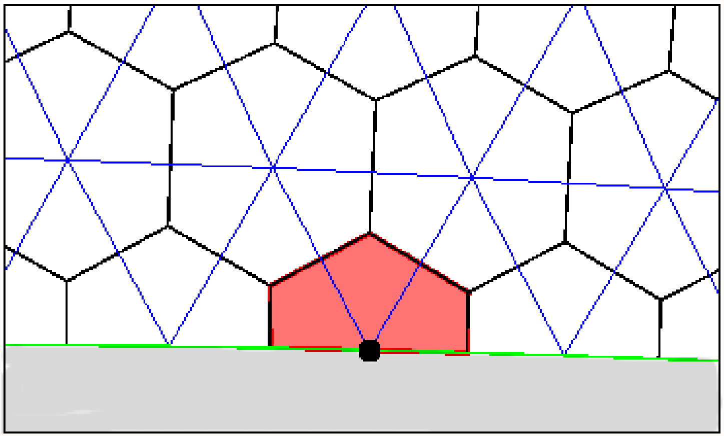
$$F(U_w, U_g, \hat{n}) = \text{function of } p \text{ only}$$

- Flux boundary condition

$$(\vec{F} \cdot \hat{n})_{\text{wall}} = p[0, n_x, n_y, 0]^\top$$

1. Extrapolate pressure from interior cells
2. Solve normal momentum equation [2]

- Vertex-centered approach - flux boundary condition
- Boundary cell in vertex-centered scheme



Accuracy and Stability

- FVM with linear reconstruction - second order accurate on uniform and smooth grids
- On non-uniform grids $\Longrightarrow$ formally first order accurate
- Local truncation error not a good indicator of global error [22]
- r'th order reconstruction and n_g Gaussian points for flux quadrature - accuracy is $\min(r, 2n_g)$ [19]
- Semi-discrete scheme

$$\frac{\mathrm{d}U_i}{\mathrm{d}t} = \sum_{j \in N(i)} a_{ij}(U_j - U_i), \quad a_{ij} \geq 0$$

- Local Extremum Diminishing (LED) property - maxima do not increase and minima do not decrease (Jameson)

- If U_i is a local maximum $\Longrightarrow U_j - U_i \le 0$

$$\frac{\mathrm{d}U_i}{\mathrm{d}t} = \sum_{j \in N(i)} a_{ij}(U_j - U_i) \le 0 \Longrightarrow U_i \text{ does not increase}$$

- Fully discrete scheme

$$U_i^{n+1} = (1 - \Delta t \sum_j a_{ij}) U_i^n + \sum_j a_{ij} U_j^n, \quad \Delta t \le \frac{1}{\sum_j a_{ij}}$$

- Convex linear combination

$$\min_{j \in N(i)} U_j^n \le U_i^{n+1} \le \max_{j \in N(i)} U_j^n$$

- Prevents oscillations (Gibbs phenomenon) near discontinuities
- Stable in maximum norm

$$\min_j U_j^n \le U_i^{n+1} \le \max_j U_j^n$$

- Elliptic equations - discrete maximum principle

$$\min_{j \in \partial\Omega} U_j \le U_i \le \max_{j \in \partial\Omega} U_j$$

Data structures and Programming

- Data structure for FVM
  - Coordinates of vertices
  - Indices of vertices forming each cell
- Cell-based updating

```
----------------------------------------------------
for cell = 1 to Ncell
      FluxDiv = 0
      for face = 1 to Nface(cell)
            cellNeighbour = CellNeighbour(cell, face)
            flux = NumFlux(cell, cellNeighbour)
            FluxDiv += flux
      end
      Unew(cell) = Uold(cell) - dt*FluxDiv
end
----------------------------------------------------
```

- Face-based updating

```
----------------------------------------------------
FluxDiv(:) = 0
for face = 1 to Nface
      LeftCell  = FaceCell(face,1)
      RightCell = FaceCell(face,2)
      flux      = NumFlux(LeftCell, RightCell)
      FluxDiv(LeftCell)  += flux
      FluxDiv(RightCell) -= flux
end
Unew(:) = Uold(:) - dt*FluxDiv(:)
----------------------------------------------------
```

- Flux computations reduced by half - speed-up of two
- Other geometric quantities - cell centroids, face areas, face normals, face centroids

References

[1] Abgrall R., "On essentially non-oscillatory schemes on unstructured meshes: analysis and implementation", J. Comp. Phys., Vol. 114, pp. 45-58, 1994.

[2] Balakrishnan N. and Fernandez, G., "Wall Boundary Conditions for Inviscid Compressible Flows on Unstructured Meshes", Int. Jl. for Num. Methods in Fluids, 28:1481-1501, 1998.

[3] Barth T. J., "Aspects of unstructured grids and solvers for the Euler and NS equations", Von Karman Institute Lecture Series, AGARD Publ. R-787, 1992.

[4] Barth T. J., "Recent developments in high order k-exact reconstruction on unstructured meshes", AIAA Paper 93-0668, 1993.

[5] Blazek J., *Computational Fluid Dynamics: Principles and Applications*, Elsevier, 2004.

[6] Delanaye M., Geuzaine Ph. and Essers J. A., "Development and application of quadratic reconstruction schemes for compressible flows on unstructured adaptive grids", AIAA-97-2120, 1997.

[7] Feistauer M., Felcman J. and Straskraba I., *Mathematical and Computational Methods for Compressible Flow*, Clarendon Press, Oxford, 2003.

[8] Frink N. T., "Upwind scheme for solving the Euler equations on unstructured tetrahedral meshes", AIAA J. 30(1), 70, 1992.

[9] Godlewski E. and Raviart P.-A., *Hyperbolic Systems of Conservation Laws*, Paris, Ellipses, 1991.

[10] Godlewski E. and Raviart P.-A., *Numerical Approximation of Hyperbolic Systems of Conservation Laws*, Springer, 1996.

[11] Carl Ollivier-Gooch and Michael Van Altena, "A high-order-accurate unstructured mesh finite-volume scheme for the advection-diffusion equation", JCP, 181, 729-752, 2002.

[12] Hirsch Ch., *Numerical Computation of Internal and External Flows*, Vol. 1, Wiley, 1988.

[13] Hirsch Ch., *Numerical Computation of Internal and External Flows*, Vol. 2, Wiley, 1989.

[14] Jawahar and Kamath H., "A High-Resolution Procedure for Euler and Navier-Stokes Computations on Unstructured Grids", Journal of Computational Physics, Vol. 164, No. 1, pp. 165-203, 2000

[15] Kallinderis Y. and Ahn H. T., "Incompressible Navier-Stokes method with general hybrid meshes", J. Comp. Phy., Vol. 210, pp. 75-108, 2005.

[16] LeVeque R. J., *Finite Volume Methods for Hyperbolic Equations*, Cambridge University Press, 2002.

[17] Lohner R., *Applied CFD Techniques: An Introduction based on FEM*, Wiley.

[18] Mavriplis D. J., "Unstructured grid techniques", Ann. Rev. Fluid Mech., 29:473-514, 1997.

[19] Sonar Th., "Finite volume approximations on unstructured grids", VKI Lecture Notes.

[20] Toro E., *Riemann Solvers and Numerical Methods for Fluid Dynamics*, Springer.

[21] Venkatakrishnan V, "Perspective on unstructured grid flow solvers", AIAA Journal, vol. 34, no. 3, 1996.

[22] Wesseling P., *Principles of Computational Fluid Dynamics*, Springer, 2001.

Thank You

These slides can be downloaded from

`http://pc.freeshell.org/pub/annamalai.pdf`