Computational Methods for the Euler Equations

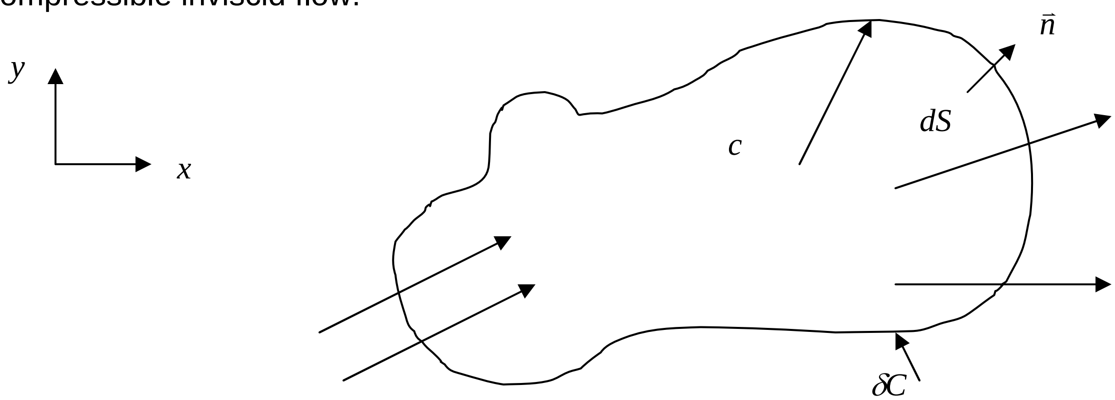
Before discussing the Euler Equations and computational methods for them, let's look at what we've learned so far:

Method	Assumptions/Flow type
2-D panel	2-D, Incompressible, Irrotational Inviscid
Vortex lattice	3-D, Incompressible, Irrotational Inviscid, Small disturbance
Potential method Prandtl-Glauert	3-D, Subsonic compressible, Irrotational, Inviscid, Small disturbance
Euler CFD	3-D, Compressible (no M_∞ limit), Rotational, Shocks, Inviscid

The only major effect missing after this week will be viscous-related effects.

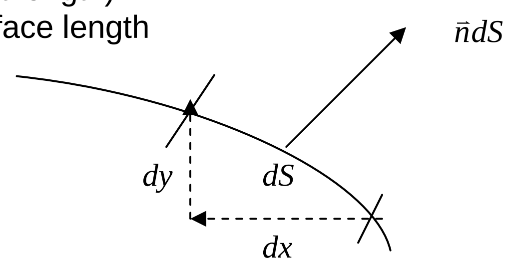
2-D Euler Equations in Integral Form

Consider an arbitrary area (i.e. a fixed control volume) through which flows a compressible inviscid flow:



$\vec{n} \equiv$ outward pointing normal (unit length)
$dS \equiv$ elemental (differential) surface length

$\vec{n}dS = dy\vec{i} - dx\vec{j}$



Note: Path around surface is taken so that interior of control volume is on left.

Conservation of Mass

$$\begin{pmatrix} rate\,of\ change \\ of\ mass\ in\,C \end{pmatrix} + \begin{pmatrix} rate\,of\ mass\ flow \\ out\,of\ C \end{pmatrix} = 0$$

$$Mass\ in\,C = \iint_C \rho dA$$

where $\rho \equiv destiny\,of\ \ fluid$

$$\Rightarrow \underset{of\ mass\,in\,C}{rate\,of\ change} = \frac{d}{dt}\iint_C \rho dA$$

Now, the rate of mass flowing out of C:

Mass flow out of $C = \oint_{\delta C} \rho \vec{u} \bullet \vec{n} dS$ $\qquad$ $\vec{u} =$ velocity vector

$$\Rightarrow \quad \boxed{\frac{d}{dt}\iint_C \rho dA + \oint_{\delta C} \rho \vec{u} \bullet \vec{n} dS = 0}$$

Conservation of x-momentum

Recall that: total rate of change momentum $= \sum$ forces

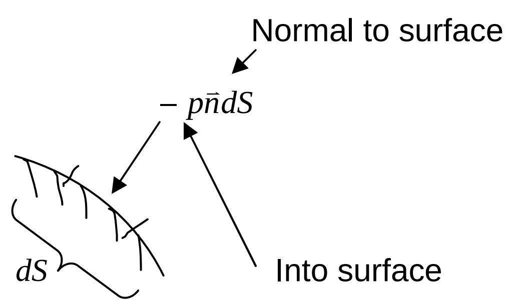
For x-momentum this gives:

$$\begin{pmatrix} rate\,of\ change\,of \\ x - momentum\ in\,C \end{pmatrix} + \begin{pmatrix} rate\,of\ x - momflow \\ out\,of\ C \end{pmatrix} = \sum \text{ Forces in x-direction}$$

$$\frac{d}{dt}\iint_C \rho u dA + \oint_{\delta C} \rho u \vec{u} \bullet \vec{n} dS = \sum \text{ Forces in x-direction}$$

Now, looking closer at x-forces, for an inviscid compressible flow we only have pressure (ignoring gravity).

Recall pressure acts normal to the surface



$$\Rightarrow \sum Force\ s\,in\,x = -\oint_{\delta C} p\vec{n} \bullet \vec{i}\,dS$$

Gives x-direction

$$\boxed{\frac{d}{dt}\iint_C \rho u dA + \oint_{\delta C} \rho u \vec{u} \bullet \vec{n} dS = -\oint_{\delta C} p\vec{n} \bullet \vec{i}\, dS}$$

Conservation of y-momentum

This follows exactly the same as the x-momentum:

$$\boxed{\frac{d}{dt}\iint_C \rho v dA + \oint_{\delta C} \rho v \vec{u} \bullet \vec{n} dS = -\oint_{\delta C} p\vec{n} \bullet \vec{j} dS}$$

Conservation of Energy

Recalling your thermodynamics:

$$\begin{pmatrix} \text{total rate of change} \\ \text{of energy in } C \end{pmatrix} = \begin{pmatrix} \text{work done on} \\ \text{fluid in } C \end{pmatrix} + \begin{pmatrix} \text{heat added} \\ \text{to } C \end{pmatrix}$$

For the Euler equations, we ignore the possibility of heat addition.

$$\begin{pmatrix} \text{total rate of change} \\ \text{of energy in } C \end{pmatrix} = \begin{pmatrix} \text{rate of change of} \\ \text{energy in } C \end{pmatrix} + \begin{pmatrix} \text{rate of energy} \\ \text{flow out of } C \end{pmatrix}$$

The total energy of the fluid is:

$$\underset{\text{Total energy}}{\rho E} = \underset{\text{Internal energy}}{\rho e} + \underbrace{\frac{1}{2}\rho(u^2 + v^2)}_{\text{Kinetic energy}}$$

Note: $\underset{\text{Static temperature}}{e = c_v T}$ where $c_v \equiv$ specific heat at constant volume

So,

$$\begin{pmatrix} \text{total rate of change} \\ \text{of energy in } C \end{pmatrix} = \frac{d}{dt}\iint_C \rho E dA + \oint_{\delta C} \rho E \vec{u} \bullet \vec{n} dS$$

The work done on the fluid is through pressure forces and is equal to the pressure forces multiplied by (i.e. acting in) the velocity direction:

$$(work) = \oint_{\delta C} \underset{\text{Pressure force}}{(-p\vec{n})} \bullet \vec{u} dS$$

$$\Rightarrow \quad \boxed{\frac{d}{dt}\iint_C \rho E dA + \oint_{\delta C} \rho E \vec{u} \bullet \vec{n} dS = -\oint_{\delta C} p\vec{n} \bullet \vec{u} dS}$$

Summary of 2-D Euler Equations

$$\frac{d}{dt}\iint_C \rho dA + \oint_{\delta C} \rho \vec{u} \bullet \vec{n} dS = 0$$

$$\frac{d}{dt}\iint_C \rho u dA + \oint_{\delta C} \rho u \vec{u} \bullet \vec{n} dS = -\oint_{\delta C} p\vec{n} \bullet \vec{i}\, dS$$

$$\frac{d}{dt}\iint_C \rho v dA + \oint_{\delta C} \rho v \vec{u} \bullet \vec{n} dS = -\oint_{\delta C} p\vec{n} \bullet \vec{j}\, dS$$

$$\frac{d}{dt}\iint_C \rho E dA + \oint_{\delta C} \rho E \vec{u} \bullet \vec{n} dS = -\oint_{\delta C} p\vec{n} \bullet \vec{n} dS$$

These are often written very compactly as:

$$\frac{d}{dt}\iint_c U dA + \oint_{sc} \left(F\vec{i} + G\vec{j}\right) \bullet \vec{n} ds = 0$$

$$U \equiv \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho E \end{bmatrix} \qquad F \equiv \begin{bmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uH \end{bmatrix} \qquad G \equiv \begin{bmatrix} \rho v \\ \rho vu \\ \rho v^2 + p \\ \rho vH \end{bmatrix}$$

Conservative state vector | Flux vector for x-direction | Flux vector for y-direction

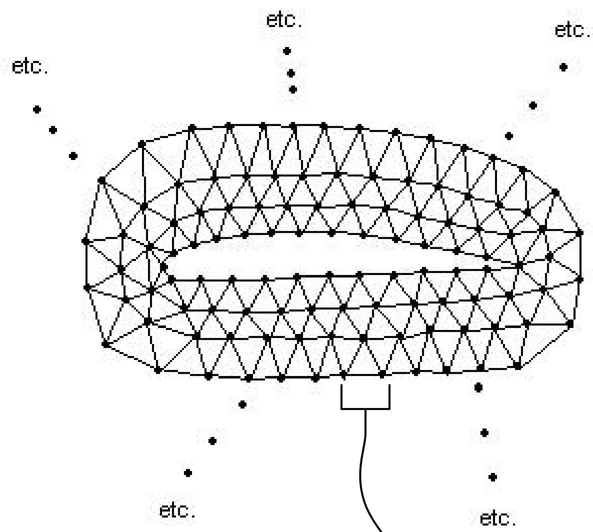
$$H \equiv total\, enthalpy \equiv E + p/\rho$$

Ideal gas: $p = \rho RT = (\gamma - 1)\left[\rho E - \frac{1}{2}\rho(u^2 + v^2)\right]$

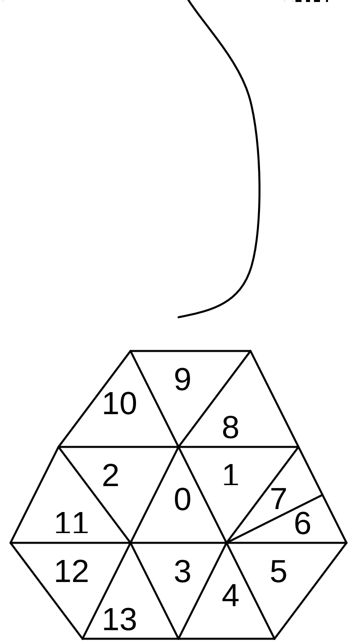
A Finite Volume Scheme for the 2-D Euler Eqns.

Here's the basic idea:

(1) Divide up (i.e. discretize) the domain into simple geometric shapes (triangles and quads)



Looking at this small region:

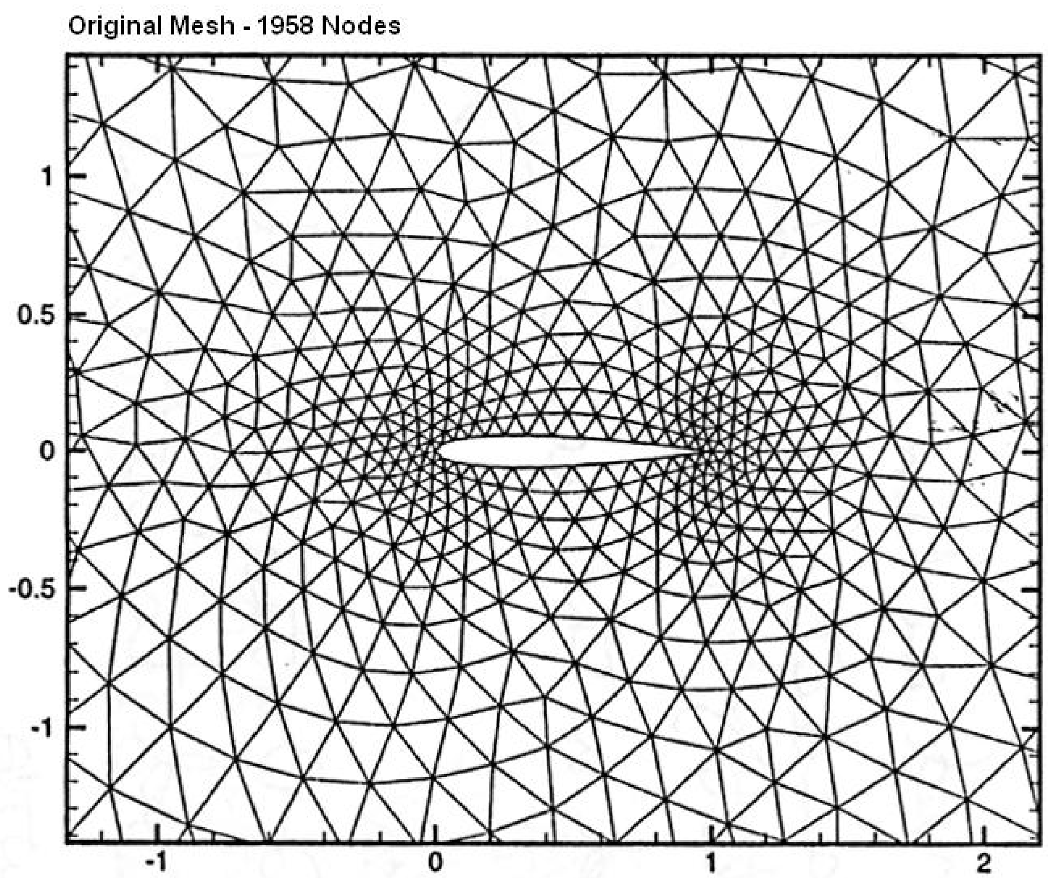


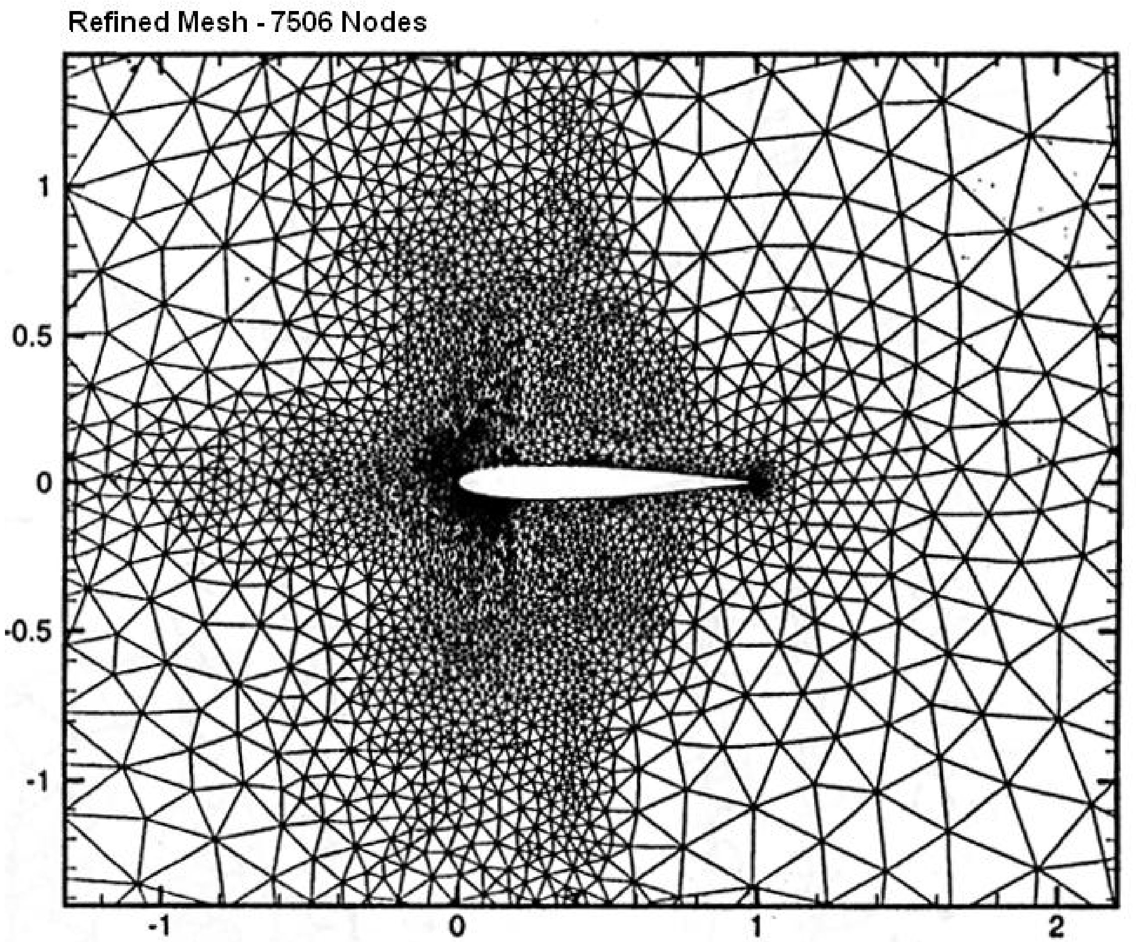
Cell 0 is surrounded by cells 1, 2, & 3.
i.e. cell 0 has $\underbrace{\text{3 neighbors:}}_{\text{Nearest neighbors}}$ cell 1, 2, & 3.

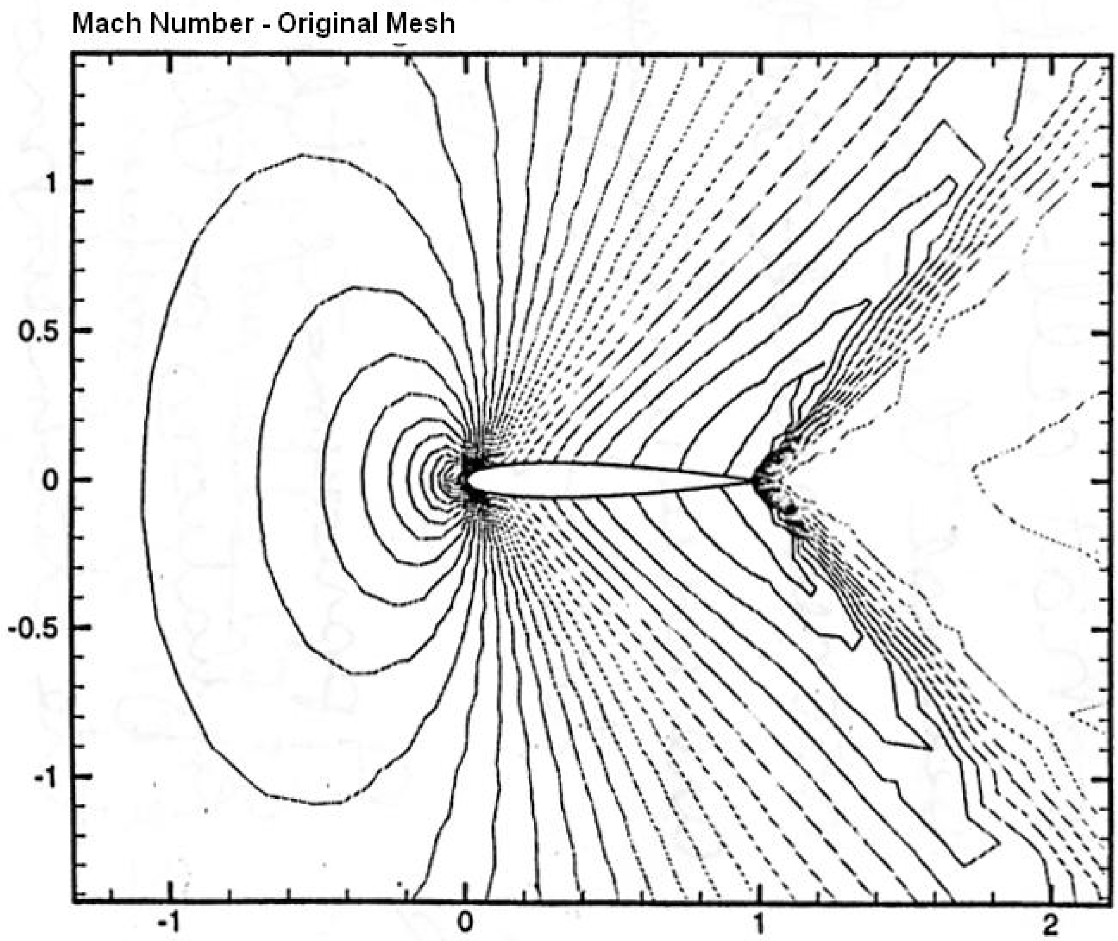
(2) Decide how to place the unknowns in the grid.
   (a) Cell-centered: cell-average values of the conservative state vector are stored for each cell.
   (b) Node-based: point values of the conservative state vector are stored at each node.

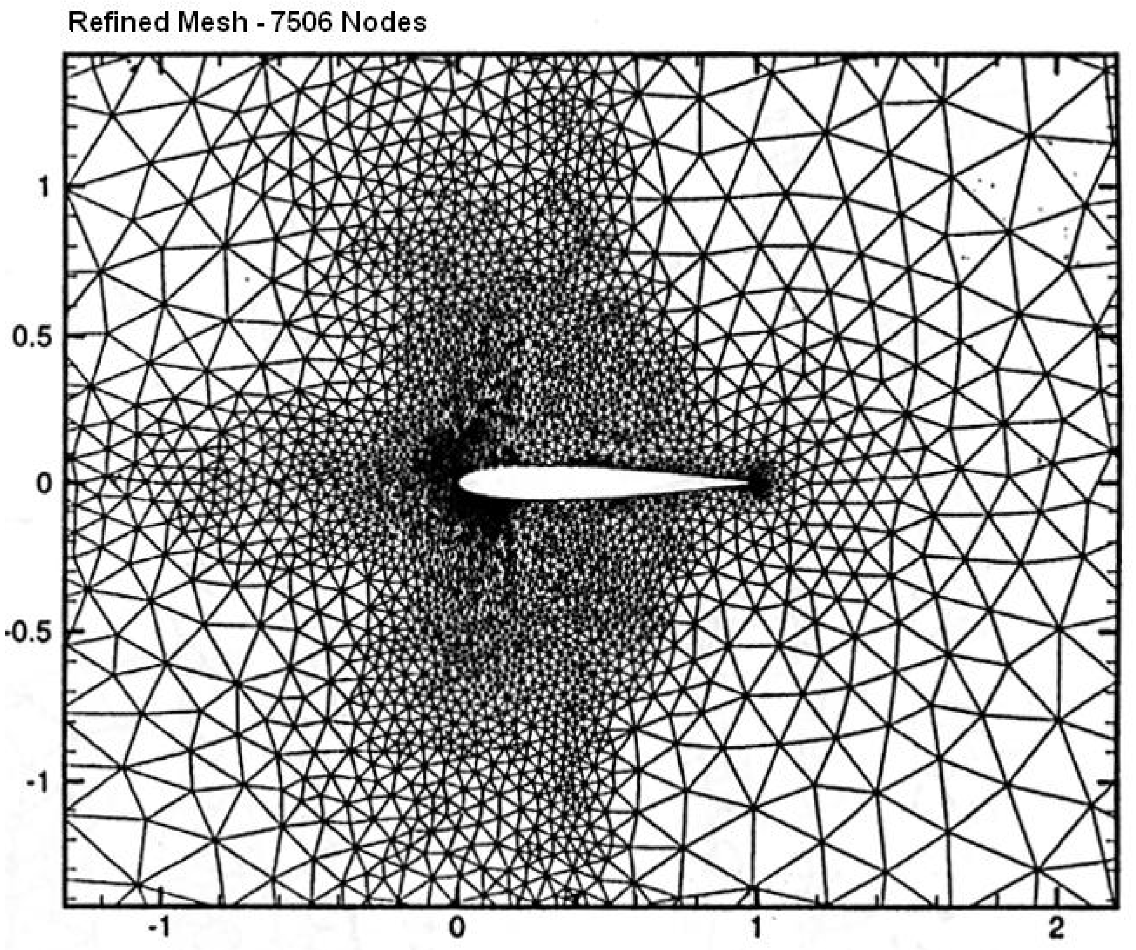
The debate still rages about which of these options is best. We will look at cell-centered schemes because these are easiest (although not necessarily the best). Also, they are very widely used in the aerospace industry.

(3) Approximate the 2-D integral Euler equation on the grid to determine the chosen unknowns.

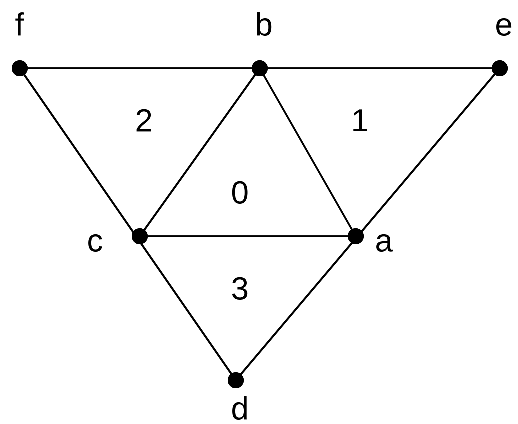



Refined Mesh - 7506 Nodes
1
0.5
0
-0.5
-1
-1
0
1
2


Mach Number - Original Mesh
1
0.5
0
-0.5
-1
-1
0
1
2


Refined Mesh - 7506 Nodes
1
0.5
0
-0.5
-1
-1
0
1
2

Let's look in detail at step (3):



Cells: 0,1, 2, 3
Nodes: a, b, c, d, e, f

Cell-average unknowns:

$$U_0 = \begin{bmatrix} \rho_0 \\ (\rho u)_0 \\ (\rho v)_0 \\ (\rho E)_0 \end{bmatrix} \qquad U_1 = \begin{bmatrix} \rho_1 \\ (\rho u)_1 \\ (\rho v)_1 \\ (\rho E)_1 \end{bmatrix} \qquad U_2 = \text{............} \qquad U_3 = \text{...........}$$

Specifically, we define U_0 as:

$$U_0 \equiv \frac{1}{A_0}\iint_{C_0} U dA \qquad \text{where } \begin{cases} C_0 \equiv cell\ 0 \\ A_0 \equiv area\, of\ cell\ 0 \end{cases}$$

Now, we apply conservation eqns:

$$\frac{d}{dt}\iint_{C_0} U dA + \oint_{\partial C_0} \left(F\vec{i} + G\vec{j}\right) \bullet \vec{n} dS = 0$$

The time-derivative term can be simplified a little:

$$\frac{d}{dt}\iint_{C_0} U dA = A_0 \frac{dU_0}{dt}$$

The surface flux integral can also be simplified a little:

$$\oint_{\partial C_0} \left(F\vec{i} + G\vec{j}\right) \bullet \vec{n} ds = \int_a^b \left(F\vec{i} + G\vec{j}\right) \bullet \vec{n} dS$$

$$+ \int_b^c \left(F\vec{i} + G\vec{j}\right) \bullet \vec{n} dS$$

$$+ \int_c^a \left(F\vec{i} + G\vec{j}\right) \bullet \vec{n} dS$$

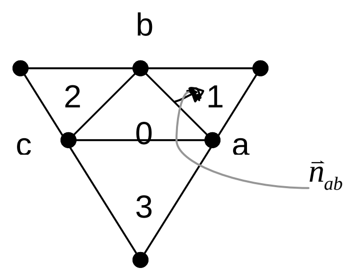
Combining these expressions:

$$\boxed{A_0 \frac{dU_0}{dt} + \int_a^b \left(F\vec{i} + G\vec{j}\right) \bullet \vec{n} dS + \int_b^c \left(F\vec{i} + G\vec{j}\right) \bullet \vec{n} dS + \int_c^a \left(F\vec{i} + G\vec{j}\right) \bullet \vec{n} dS = 0}$$

No approximations so far!

Now, we make some approximations. Let's look at the surface integral from $a \to b$:

$$\int_a^b \left(F\vec{i} + G\vec{j}\right) \bullet \vec{n} ds$$



The normal can easily be calculated since the <u>face</u> is a straight line between nodes a & b. Recall, the unknowns are stored at all centers. So, what would be a logical approximation for :

$$\int_a^b \left(F\vec{i} + G\vec{j}\right) \bullet \vec{n}_{ab} dS = ???$$

Option #1=

Option #2=

Note: Option #1 $\neq$ option #2 in general.

There is very little difference in practice between these options. Let's stick with:

$$\Im_{ab} \equiv \int_a^b \left(F\vec{i} + G\vec{j}\right) \bullet \vec{n}_{ab} dS = \left[\frac{1}{2}(F_0 + F_1)\vec{i} + \frac{1}{2}(G_0 + G_1)\vec{j}\right] \bullet \vec{n}_{ab} \Delta S_{ab}$$

$$\Im_{bc} \equiv \int_b^c \left(F\vec{i} + G\vec{j}\right) \bullet \vec{n}_{bc} dS = \left[\frac{1}{2}(F_0 + F_2)\vec{i} + \frac{1}{2}(G_0 + G_2)\vec{j}\right] \bullet \vec{n}_{bc} \Delta S_{bc}$$

$$\Im_{ca} \equiv \int_c^a \left(F\vec{i} + G\vec{j}\right) \bullet n_{ca} dS = \left[\frac{1}{2}(F_0 + F_3)\vec{i} + \frac{1}{2}(G_0 + G_3)\vec{j}\right] \bullet \vec{n}_{ca} \Delta s_{ca}$$

Where

$$F_0 \equiv F(U_0) \qquad G_0 \equiv G(U_0)$$
$$F_1 \equiv F(U_1) \qquad G_1 \equiv G(U_1)$$
$$F_2 \equiv F(U_2) \qquad G_2 \equiv G(U_2)$$
$$F_3 \equiv F(U_3) \qquad G_3 \equiv G(U_3)$$

Finally, we have to approximate $A_0 \frac{dU_0}{dt}$ somehow. The simplest approach is forward Euler:

$$A_0 \frac{dU_0}{dt} + \Im_{ab} + \Im_{bc} + \Im_{ca} = 0$$

$$\boxed{\begin{array}{c} A_0 \dfrac{U_0^{n+1} - U_0^n}{\Delta t} + \Im_{ab}^n + \Im_{bc}^n + \Im_{ca}^n = 0 \\ \text{Where } U_o^n \equiv U_o\left(t^n\right) \text{ and } t^n \equiv n\Delta t,\ n \equiv iteration \end{array}}$$

And $\Im_{ab}^n$ etc. are defined as:

$$\Im_{ab}^n \equiv \left[\frac{1}{2}\left(F_0^n + F_1^n\right)\vec{i} + \frac{1}{2}\left(G_0^n + G_1^n\right)\vec{j}\right] \bullet \vec{n}_{ab} \Delta S_{ab}$$

$$F_0^n \equiv F\left(U_0^n\right) \qquad etc.$$

$$F_1^n \equiv F\left(U_1^n\right)$$

For steady solution, basic procedure is to make a guess of U at $t = 0$ and then iterate until the solution no longer changes. This is called time marching.

<u>Question</u>
What assumptions have we made in developing our 2-D Euler Equation Finite Volume Method?