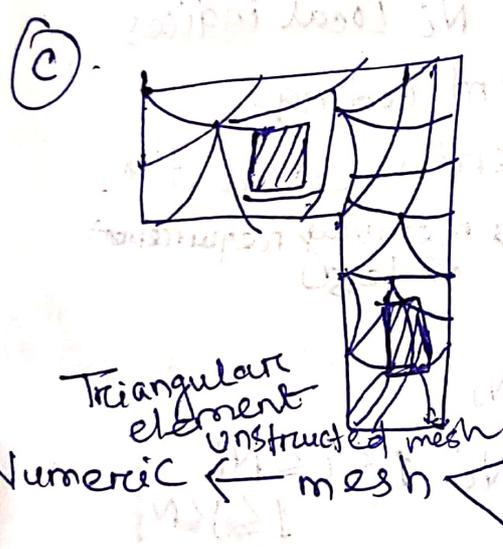
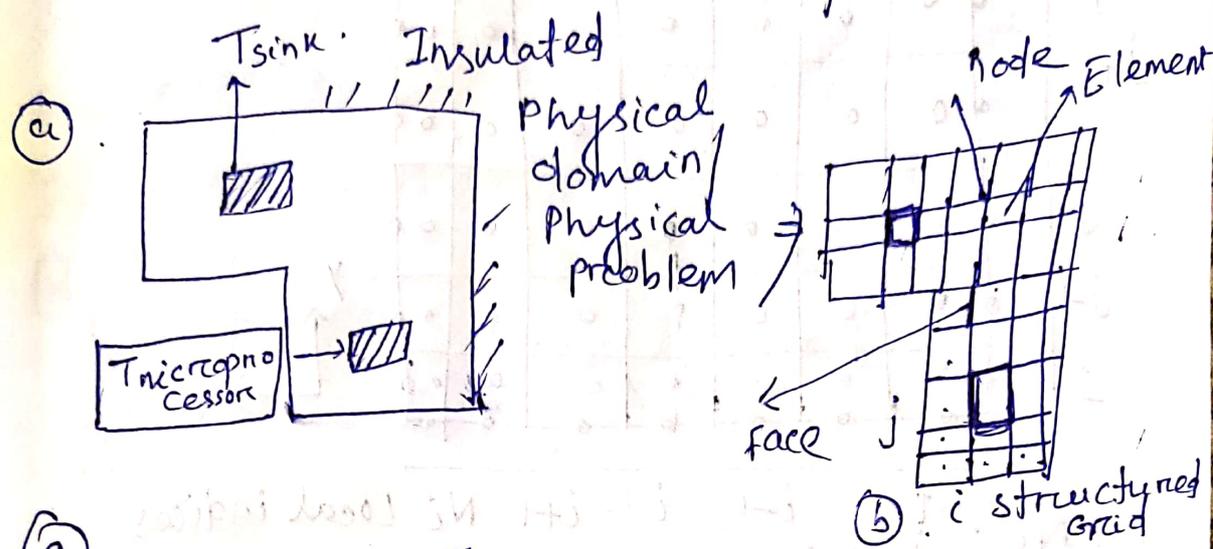


the equation are hyperbolic  
 speed is supersonic & elliptic for  
 subsonic speed.

### Finite Volume Mesh/Grid



- (b) Element handling is easier → data structure can be easier to handle.
  - (c) Element is complex type. It is not order. Complex geometry can be easily solved.
- structured } multi-block  
 unstructured } single-block

### Gradient computation:

$\phi_s = \text{outward pointing surface vector}$

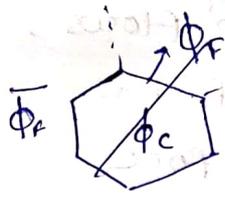
$$\nabla \phi_c = \frac{1}{V_c} \left[ \int_{V_c} \nabla \phi dV \right] \text{ using divergent theorem}$$

$$\nabla \phi_c = \frac{1}{V_c} \int_{V_c} \phi ds$$

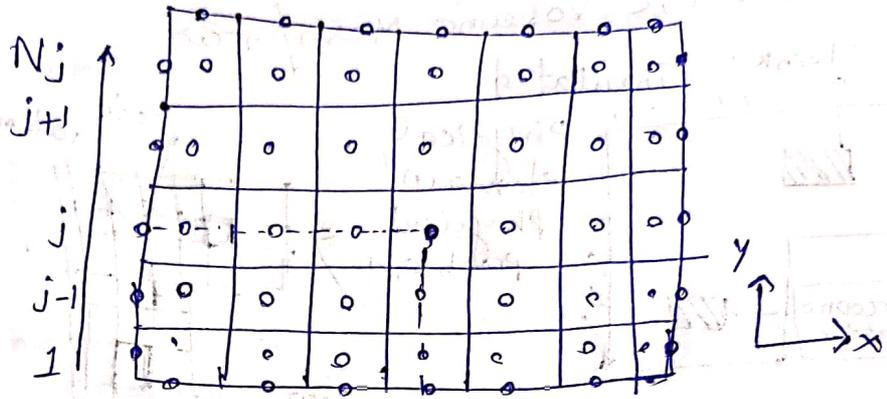
element 'c' of volume  $V_c$

Volume integral converted into surface integral.

Assuming same linear variation of  $\phi$  between  $\phi_c$  and  $\phi_f$



structure Grid - 2 Dimensional.



$i: Ni: 1 \quad i-1 \quad i \quad i+1 \quad Ni$  Local indices  
 $i, j: Ni \times Nj \rightarrow$  efficient coding  
 $i, j \rightarrow Ni \times Nj \rightarrow$  Efficient vectorization.  
 $=$  Total no. of cell.  $\rightarrow$  memory requirement is less  
 $i, j, k \rightarrow Ni \times Nj \times Nk$

2D - system  $(i, j): Ni \times Nj$

global indices  $(n) = i + (j-1)Ni \quad 1 \leq i \leq Ni$   
 $1 \leq j \leq Nj$

$(i, j) \rightarrow n$

$(i+1, j) \rightarrow n+1$

$(i, j-1) \rightarrow n-Ni$

3D  $n = i + (j-1)Ni + (k-1)Ni \times Nj$

$(i, j, k) \rightarrow n$

$(i+1, j, k) \rightarrow n+1$

$(i, j+1, k) \rightarrow n+Ni$

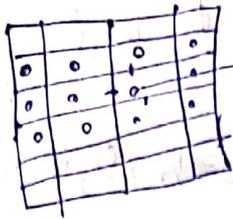
$(i, j, k+1) \rightarrow n+Ni \times Nj$

$(i-1, j, k) \rightarrow n-1$

$(i, j-1, k) \rightarrow n-Ni$

$(i, j, k-1) \rightarrow n-Ni \times Nj$

# Geometric information

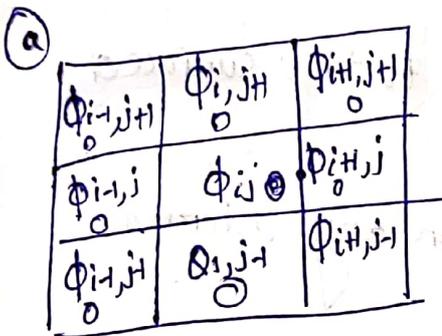


→ element  $(i,j)$ .

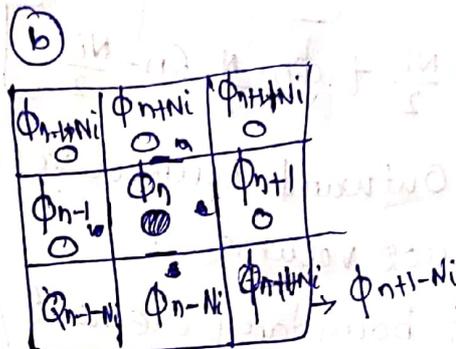
Faces are associated with element  $(i,j)$ .

4 Faces =  $s_1(i,j), s_2(i,j)$

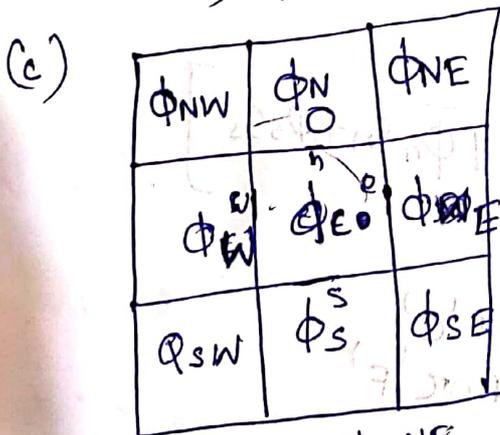
$$\begin{aligned}
 s_{i-1/2, j} &= -s_1(i,j) & s_3(i,j) & s_4(i,j) \\
 s_{i, j-1/2} &= -s_2(i,j) & s_1(i,j) & s_2(i,j) \\
 s_{i+1/2, j} &= s_1(i+1, j) & s_1(i+1, j) & s_2(i, j+1) \\
 s_{i, j+1/2} &= s_2(i, j+1)
 \end{aligned}$$



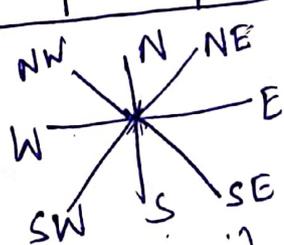
(a) Local indexing  
 $(i,j) \rightarrow n, n+1, \dots$



(b) global indexing.



It is a discretised indexing which gives directional value which is used in finite volume method. discretised cell nicely located.



discretised indexing

at face  $(i+1/2, j) \rightarrow$  interpolate the value by linear interpolation

$$\phi_{i+1/2, j} = g_{i+1/2, j} \phi_{i+1, j} + (1 - g_{i+1/2, j}) \phi_{i, j}$$

Using local indexing by figure (a) gradient calculation.

$$\nabla \phi_{ij} = \frac{1}{V_{ij}} \left[ \bar{\phi}_{i+\frac{1}{2},j} S_{i+\frac{1}{2},j} + \bar{\phi}_{i-\frac{1}{2},j} S_{i-\frac{1}{2},j} + \bar{\phi}_{i,j+\frac{1}{2}} S_{i,j+\frac{1}{2}} + \bar{\phi}_{i,j-\frac{1}{2}} S_{i,j-\frac{1}{2}} \right]$$

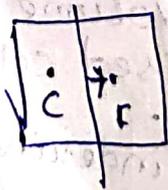
~~$\frac{1}{V_{ij}} \left[ \bar{\phi}_{i+\frac{1}{2},j} S_{i+\frac{1}{2},j} + \bar{\phi}_{i-\frac{1}{2},j} S_{i-\frac{1}{2},j} \right]$~~   
 Using global indexing (figure-b)

$$\nabla \bar{\phi}_n = \frac{1}{V_n} \left[ \bar{\phi}_{n+\frac{N_i}{2}} S_{n+\frac{N_i}{2}} + \bar{\phi}_{n-\frac{N_i}{2}} S_{n-\frac{N_i}{2}} + \bar{\phi}_{n+\frac{N_j}{2}} S_{n+\frac{N_j}{2}} + \bar{\phi}_{n-\frac{N_j}{2}} S_{n-\frac{N_j}{2}} \right]$$

$S =$  Outward normal vector to the surface of control volume  
 except boundary element / For any internal element  
 outward normal for one element  $\rightarrow$  represent inward normal direction for the other element.

Discretizing indexing

$$\nabla \bar{\phi}_e = \frac{1}{V_e} \left[ \bar{\phi}_e S_e + \bar{\phi}_w S_w + \bar{\phi}_n S_n + \bar{\phi}_s S_s \right]$$



outward for 'C'  
 = inward for 'F'

