Section 3: Finite Volume Method for Diffusion Problems

Governing Differential Equation for a Diffusion Problem

$$\frac{\partial}{\partial x}\left(\Gamma\frac{\partial\phi}{\partial x}\right)+\frac{\partial}{\partial y}\left(\Gamma\frac{\partial\phi}{\partial y}\right)+S=0$$

or, in vector form:

$\nabla\cdot(\Gamma\nabla\phi)$+S=0

Integrate Over Control Volume

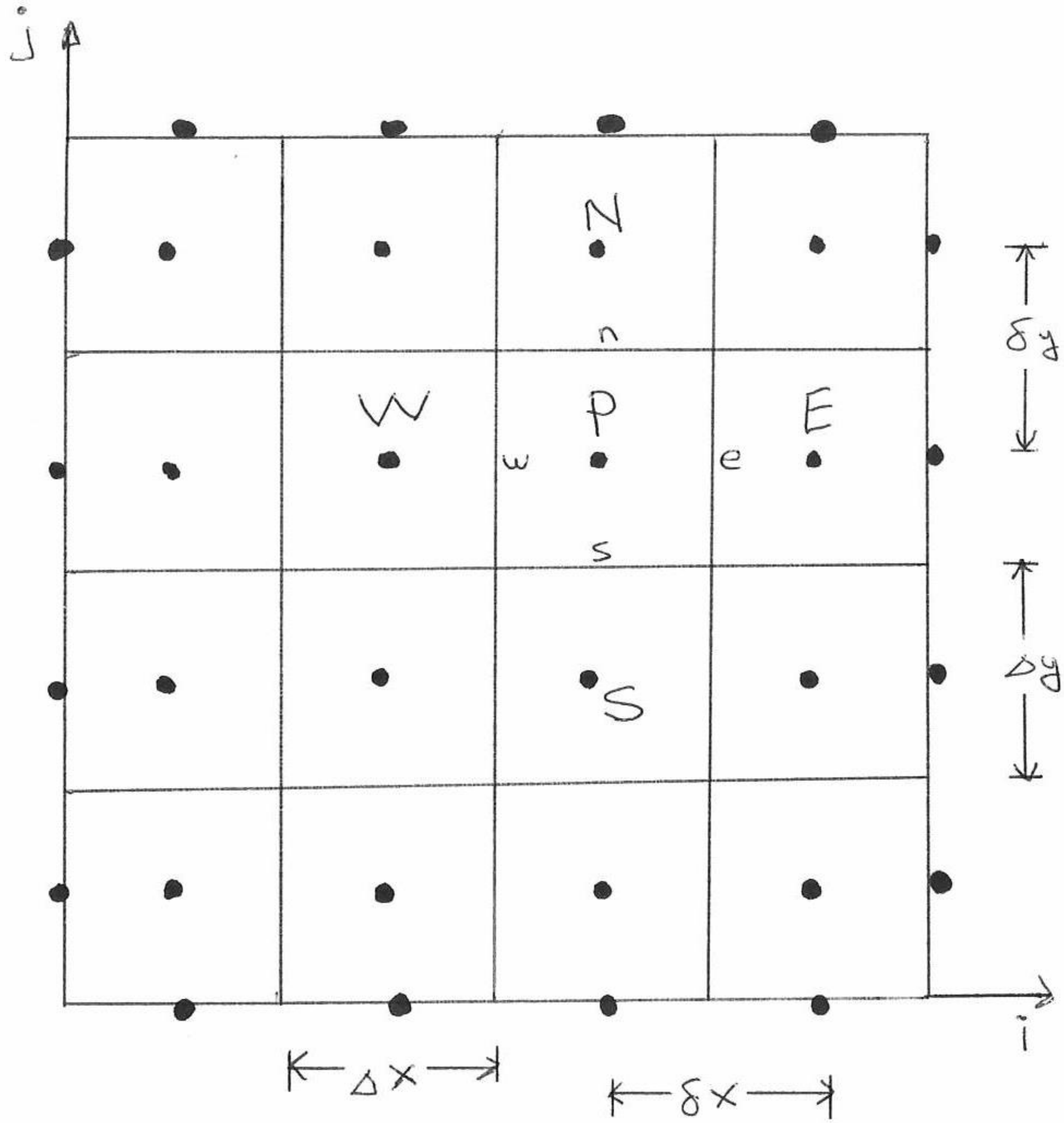
$$\int\nabla\cdot(\Gamma\nabla\phi)dV+\int SdV=0$$

Apply the Gauss Divergence Theorem, which converts the volume integral to a surface integral, and is given as

$$\int\nabla\cdot\overline{a}\,dV=\int\overline{a}\cdot\hat{n}\,dS$$

Applied to our integral we find

$$\int\Gamma\nabla\phi\cdot\hat{n}\,dS+\int SdV=0$$



Typical Finite Volume Mesh and Terminology

$$A_e\Gamma_e\frac{\partial\phi}{\partial x}\bigg|_e - A_w\Gamma_w\frac{\partial\phi}{\partial x}\bigg|_w + A_n\Gamma_n\frac{\partial\phi}{\partial y}\bigg|_n - A_s\Gamma_s\frac{\partial\phi}{\partial y}\bigg|_s + \bar{S}\Delta x\Delta y = 0$$

Finite-difference approximation for derivatives

$$\frac{\partial\phi}{\partial x}\bigg|_e = \frac{\phi_E - \phi_P}{(\delta x)_e} + O(\delta x)^2$$

and

source term linearization

$$\bar{S}\Delta x\Delta y = S_u + S_p\phi_p$$

results in

$$\Delta y\left(\Gamma_e\frac{\phi_E - \phi_P}{(\delta x)_e} - \Gamma_w\frac{\phi_P - \phi_W}{(\delta x)_w}\right) + \Delta x\left(\Gamma_n\frac{\phi_N - \phi_P}{(\delta y)_n} - \Gamma_s\frac{\phi_P - \phi_S}{(\delta y)_s}\right) + S_u + S_P\phi_P = 0$$

Rearrange as

$$\left(\frac{\Gamma_w\Delta y}{(\delta x)_w} + \frac{\Gamma_e\Delta y}{(\delta x)_e} + \frac{\Gamma_s\Delta x}{(\delta y)_s} + \frac{\Gamma_n\Delta x}{(\delta y)_n} - S_p\right)\phi_P = \frac{\Gamma_w\Delta y}{(\delta x)_w}\phi_W + \frac{\Gamma_e\Delta y}{(\delta x)_e}\phi_E + \frac{\Gamma_n\Delta x}{(\delta y)_n}\phi_N + \frac{\Gamma_s\Delta x}{(\delta y)_s}\phi_S + S_u$$

$$A_P\phi_P = A_W\phi_W + A_E\phi_E + A_S\phi_S + A_N\phi_N + S_u$$

$$A_W = \frac{\Gamma_w\Delta y}{(\delta x)_w}$$

$$A_E = \frac{\Gamma_e\Delta y}{(\delta x)_e}$$

$$A_S = \frac{\Gamma_s \Delta x}{(\delta y)_s}$$

$$A_N = \frac{\Gamma_n \Delta x}{(\delta y)_n}$$

$$A_P = A_W + A_E + A_S + A_N - S_P$$

****

Boundaries

A) Fixed (Dirichlet) boundary condition (i.e., set ϕ on a boundary)

Look at west boundary as an example

Find

$$S_p = \frac{-\Gamma_w \Delta y}{((\delta x)_w/2)}$$

$$S_u = \frac{\Gamma_w \Delta y}{((\delta x)_w/2)}\phi_{BC}$$

$$A_W = 0$$

What did we do? "Cut the link" to west boundary and utilized source terms to implement boundary condition.

B) Derivative (Neumann) boundary condition on the west boundary

$$S_P = 0$$

$$S_u = -A_w \Gamma_w \left.\frac{\partial \phi}{\partial x}\right|_w$$

$$A_W = 0$$

****

Solution algorithm

Successive over relaxation iterative scheme

$$\phi_P = \phi_P + \frac{\Omega}{A_P}(A_W\phi_W + A_E\phi_E + A_S\phi_S + A_N\phi_N + S_u - A_P\phi_P)$$

$$0 < \Omega < 2$$

****

Example Problem

Solve the following problem using the finite-volume method:

$\nabla^2\phi$=0

$0 \leq x \leq 1 \qquad 0 \leq y \leq 1$

Set the boundary conditions as:

$\phi(0, y) = 0;\ \ \phi(1, y) = y;\ \ \phi(x, 0) = 0;\ \ \phi(x, 1) = x$

This problem is of the form:

$\nabla \cdot (\Gamma \nabla \phi) = 0$

with $\Gamma = 1$.

The problem has the analytic solution $\phi(x, y) = xy$.

Consequently, we can determine our numerical errors.

Simple Code in Fortran90

```fortran
!     PROGRAM TO SOLVE LAPLACE EQUATION OVER DOMAIN
!     0<=x<=1,  0<=y<=1
!     with boundary conditions phi=x*y
!*************************************************
      implicit none
      integer, parameter :: nx=25, ny=25
      integer :: i,j,iterations
      real :: x,y,z,dx,dy,omega
      real, dimension(1:nx,1:ny) :: AE,AW,AN,AS,AP,SP,SU
      real, dimension(0:nx+1,0:ny+1) :: phi
!*************************************************
      dx=1./float(nx);  dy=1./float(ny)
!     initialize all SU and SP to interior cell values of zero
      SU=0;  SP=0
      phi=0.0
!     initialize AE,AW,AN,AS,AP over all cells
!     to the values over the interior cells
      AE=1.; AW=1.; AN=1.; AS=1.
!************************************************
```

```
!     set wall boundary values (overwrite initialization values)
!     west boundary
      AW(1,:)=0.    !cut link
      SU(1,:)=0.
      SP(1,:)=-2.

!     east boundary
      AE(NX,:)=0.    !cut link
      SP(NX,:)=-2.
      do j=1,NY
         SU(NX,j)=2.*( (dy/2.) + dy*float(j-1) )
      end do

!     south boundary
      AS(:,1)=0.    !cut link
      SU(:,1)=0.
      SP(:,1)=-2.
```

```
!     north boundary
      AN(:,NY)=0.    ! cut link
      SP(:,NY)=-2.
     do i=1,NX
        SU(i,NY)=2.*( (dx/2.) + dx*float(i-1) )
     end do

      corner cell boundary values
!     southwest
      AS(1,1)=0.    ! cut links
      AW(1,1)=0.
      SU(1,1)=0.
      SP(1,1)=-4.

!     northwest
      AN(1,ny)=0.    ! cut links
      AW(1,ny)=0.
      SU(1,ny)=2.*(dx/2.)
      SP(1,ny)=-4.
```

```
!    southeast
     AS(nx,1)=0.    ! cut links
     AE(nx,1)=0.
     SU(nx,1)=2.*(dy/2.)
     SP(nx,1)=-4.

!    northeast
     AN(nx,ny)=0.    ! cut links
     AE(nx,ny)=0.
     SU(nx,ny)=2.*((dx/2.) + dx*float(nx-1)) + 2.*((dy/2.) + dy*float(ny-1))
     SP(nx,ny)=-4.
     AP=AE+AW+AN+AS-SP

!    Successive Over-Relaxation  ITERATIVE SOLVER
     omega=1.7   ! relaxation factor
     do iterations=1,500   ! set iterations over interior cells to 500
     do i=1,nx
     do j=1,ny
             phi(i,j)=phi(i,j) + omega/AP(i,j)*(AE(i,j)*phi(i+1,j)+ &
                        AW(i,j)*phi(i-1,j)+AN(i,j)*phi(i,j+1)+ &
                        AS(i,j)*phi(i,j-1)+SU(i,j)-AP(i,j)*phi(i,j))
     end do; end do; end do
!*********************************************
```

```
!     WRITE comma separated DATA FILE FOR PARAVIEW
      open(unit=10,file='results.csv')
      write(10,*)'x coord, y coord, z coord, phi'

!     write out Interior data points
      do i=1,nx
      do j=1,ny
         x=dx/2+float(i-1)*dx
         y=dy/2+float(j-1)*dy
         z=0.
        write(10,*)x,',',y,',',z,',',phi(i,j)
      end do;  end do
```

```
!    write west boundary values
     do j=1,ny
        y=dy/2+float(j-1)*dy
        write(10,*)0.0,',',y,',',0.0,',',0.0
     end do;

!    write east boundary values
     do j=1,ny
        y=dy/2+float(j-1)*dy
        write(10,*)1.0,',',y,',',0.0,',',y
     end do

!    write south boundary values
     do i=1,nx
        x=dx/2+float(i-1)*dx
        write(10,*)x,',',0.0,',',0.0,',',0.0
     end do;
!    write north boundary values
     do i=1,nx
        x=dx/2+float(i-1)*dx
        write(10,*)x,',',1.0,',',0.0,',',x
     end do;
```

```
!     write corner boundary values
      write(10,*)0.0,',',0.0,',',0.0,',',0.0     ! southwest

      write(10,*)0.0,',',1.0,',',0.0,',',0.0     ! northwest

      write(10,*)1.0,',',0.0,',',0.0,',',0.0     ! southeast

      write(10,*)1.0,',',1.0,',',0.0,',',1.0     ! northeast

      end
```