## Numerical Formulation of IFV for Unstructured Grids

The partial differential equation of the general form

$$\frac{\partial A}{\partial t} + \boldsymbol{\nabla} \cdot \boldsymbol{F} \;=\; \mathcal{S}, \tag{1}$$

with accumulation term $A$, source/sink term $\mathcal{S}$, and flux term $\boldsymbol{F}$ of the form

$$\boldsymbol{F} \;=\; \boldsymbol{q}\rho X - \phi D \rho \boldsymbol{\nabla} X, \tag{2}$$

can be solved numerically through discretized integrated finite volume equations.

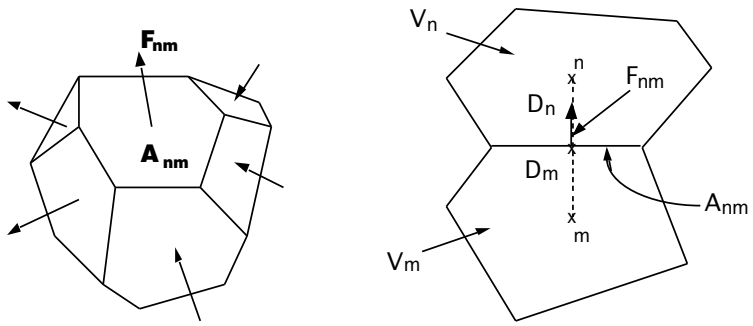


Figure 1: General unstructured grid showing control volume, and interfacial areas and fluxes.

Partitioning the computational domain into a set of finite volumes $V_n$ (see Figure **??**) and integrating the partial differential equations over each volume yields a discretized form of the mass conservation equations. The following results are obtained:

$$\int_{V_n} \frac{\partial}{\partial t} A \, dV \simeq \frac{A_n^{t+\Delta t} - A_n^t}{\Delta t} V_n, \tag{3}$$

for the accumulation term,

$$\int_{V_n} \mathcal{S} \, dV \simeq \mathcal{S}_n V_n, \tag{4}$$

for the source term, and

$$\int_{V_n} \boldsymbol{\nabla} \cdot \boldsymbol{F} \, dV \;=\; \int_{\partial V_n} \boldsymbol{F} \cdot \boldsymbol{dS} \simeq \sum_{n'} F_{nn'} A_{nn'}, \tag{5}$$

for the flux term, where $\partial V_n$ denotes the surface of $V_n$, and the sum is over the neighboring volumes connected to $V_n$. The flux $F_{nn'}$ across the $n-n'$ interface connecting volumes $V_n$ and $V_{n'}$ is defined by

$$F_{nn'} \;=\; (q\rho)_{nn'} X_{nn'} - (\phi D \rho)_{nn'} \frac{X_n - X_{n'}}{d_n + d_{n'}}, \tag{6}$$

where the subscript $nn'$ indicates that the quantity is evaluated at the interface, and the quantities $d_n$, $d_{n'}$ denote the distances from the centers of the control volumes $V_n$, $V_{n'}$ to the their common interface with interfacial area $A_{nn'}$. Combining these results gives the residual equation for the discretized form of the partial differential equations

$$R_n \;=\; \big(A_n^{k+1} - A_n^k\big)\frac{V_n}{\Delta t} + \sum_{n'} F_{nn'} A_{nn'} - \mathcal{S}_n V_n, \tag{7}$$

where, in general, $R_n$ is a nonlinear function of the independent field variables and superscript $k$ denotes the $k$th time step. These equations may be solved using a Newton-Raphson iteration technique in which the discretized equations are first linearized resulting in the Newton-Raphson equations

$$\sum_{n'} J_{nn'}^i \, \delta x_{n'}^{i+1} \;=\; -R_n^i, \tag{8}$$

for the $i$th iteration, with the Jacobian matrix $J_{nn'}^i$ defined by

$$J_{nn'}^i \;=\; \frac{\partial R_n^i}{\partial x_{n'}^i}. \tag{9}$$

Typically, for solving the flow equations $\delta x_n = \delta p_n$, where $p$ denotes the fluid pressure, whereas for the reactive transport equations $\delta x_n = \delta \ln C_{jn}$, where $C_{jn}$ denotes the concentration of the $j$th chemical species.

An explicit method is used to solve the mineral mass transfer equations

$$\frac{\partial \varphi_m}{\partial t} \;=\; \overline{V}_m I_m, \tag{10}$$

given by:

$$\phi_m(\boldsymbol{r},\, t + \Delta t) \;=\; \phi_m(\boldsymbol{r},\, t) + \Delta t \overline{V}_m I_m(\boldsymbol{r},\, t), \tag{11}$$

with mineral volume fraction $\varphi_m$ and molar volume $\overline{V}_m$, where the mineral reaction rate $I_m(\boldsymbol{r},\, t)$ is taken from the previous time step.

**Pseudo code illustrating implementation of the IFV method.** Pseudo is given below for the flow equation

$$\frac{\partial}{\partial t}\varphi\rho + \boldsymbol{\nabla}\cdot\rho\boldsymbol{u} \;=\; 0, \tag{12}$$

with $\boldsymbol{u}$ given by Darcy's law

$$\boldsymbol{u} \;=\; -\frac{\kappa}{\mu}\nabla\big(p - W\rho g z\big), \tag{13}$$

with permeability $\kappa$, viscosity $\mu$, molar fluid density $\rho$, formula weight of water $W$, and acceleration of gravity $g$.

```
!accumulation term
  do n = 1, grid%nlmax  ! For each local node do...
    ng = grid%nL2G(n)   ! corresponding ghost index
    voldt = porosity_loc_p(ng) * volume_p(n) / grid%dt
```

```
    r_p(jn) = (ddensity_loc_p(ng) - density_p(n)) * voldt
  enddo

!flux terms
  do nc = 1, grid%nconn  ! For each interior connection...
    m1 = grid%nd1(nc) ! node indices other either side of face nc
    m2 = grid%nd2(nc)

    n1 = grid%nG2L(m1) ! local node indices
    n2 = grid%nG2L(m2)

    dd1 = grid%dist1(nc) ! distances to interface
    dd2 = grid%dist2(nc)

    ip1 = grid%iperm1(nc) ! permeability direction
    ip2 = grid%iperm2(nc)

    if (ip1 == 1) then
      perm1 = perm_xx_loc_p(m1) ! permeability in x-direction
    else if (ip1 == 2) then
      perm1 = perm_yy_loc_p(m1) ! permeability in y-direction
    else
      perm1 = perm_zz_loc_p(m1) ! permeability in z-direction
    endif

    if (ip2 == 1) then
      perm2 = perm_xx_loc_p(m2)
    else if (ip2 == 2) then
      perm2 = perm_yy_loc_p(m2)
    else
      perm2 = perm_zz_loc_p(m2)
    endif

    dd = dd1 + dd2
    f1 = dd1/dd
    f2 = dd2/dd

    gravity = grid%fmwh2o * grid%gravity * grid%delz(nc)

    D1 = perm1 / viscosity_loc_p(m1)
    D2 = perm2 / viscosity_loc_p(m2)

    D = (D1 * D2) / (dd2*D1 + dd1*D2)

    density_ave = f2 * ddensity_loc_p(m1) + f1* ddensity_loc_p(m2)

    v_darcy = -D * (ppressure_loc_p(m2) - ppressure_loc_p(m1) &
                - gravity * density_ave)
```

```
  q = v_darcy * grid%area(nc)
  flux = density_ave * q

    ! Now add the flux contributions for this phase.
    ! Note that fluxes through a downstream face should be added to the
    ! residual component at the cell, while fluxes through an upstream face
    ! should be subtracted.  (The divergence gives the net OUTFLOW rate per
    ! unit volume.)  Thus, when working with pressure differences,
    ! (ppressure(jm2) - ppressure(jm1)) should be *subtracted* at the
    ! upstream node n1 because q = -D*div(P).

  if (n1 > 0) then  ! If the upstream node is not a ghost node...
    r_p(n1) = r_p(n1) + flux
  endif

  if (n2 > 0) then ! If the downstream node is not a ghost node...
    r_p(n2) = r_p(n2) - flux
  endif
enddo
```