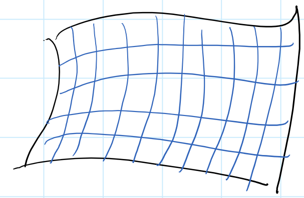
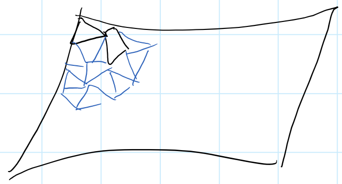
Grid generation

Overall approaches:

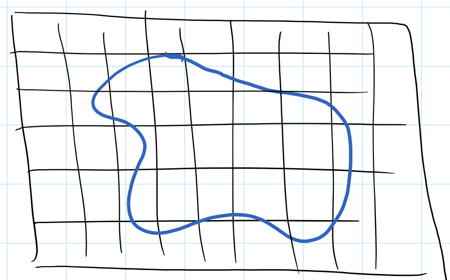
1) Body-fitted grids



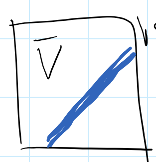
- Cartesian curvilinear
- simplicial grids (e.g. triangles in 2D, tetrahedra in 3D)



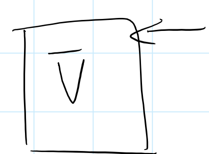
2) Embedded boundaries



- replace boundary "with its effect", typically source or doublet singularity
- cut cells



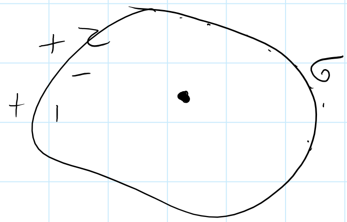
set up $\Longleftrightarrow$ equivalent cell



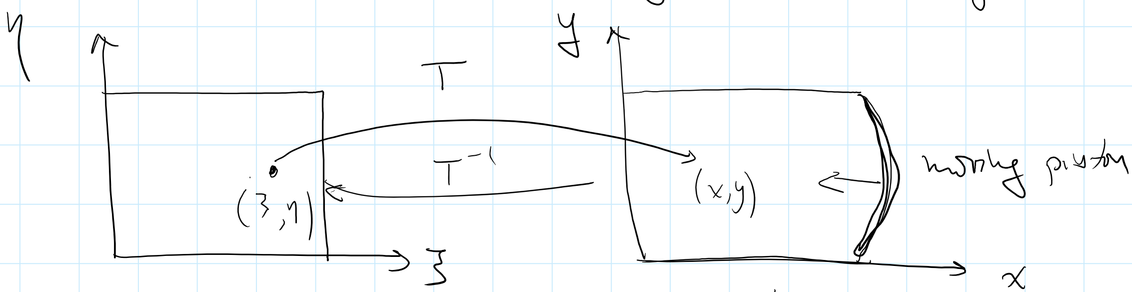
Recall harmonic representation theorem

$$\begin{cases}\nabla^2\varphi = \rho & \text{in } \Omega \\ \varphi\big|_\Sigma = f & \text{on } \Sigma = \partial\Omega\end{cases} \qquad \text{in 3D}$$

$$\varphi(\vec{x}) = \frac{1}{2\pi}\int_V \frac{f}{|\vec{y}-\vec{x}|}\,d\vec{y} + \frac{1}{2\pi}\int \frac{\sigma}{|\vec{y}-\vec{x}|}\,d\vec{y} + \frac{1}{2\pi}\int_{\Sigma} \tau \frac{\partial}{\partial n}\left(\frac{1}{|\vec{y}-\vec{x}|}\right) d\vec{y}$$



Conservation laws in moving, curvilinear grids



$$T: \begin{cases} x = X(\xi,\eta,t) \\ y = Y(\xi,\eta,t) \end{cases} \qquad \exists\, T^{-1}$$

$$J = \begin{vmatrix} X_\xi & X_\eta \\ Y_\xi & Y_\eta \end{vmatrix} \neq 0$$

$$(1)\quad \boxed{\rho_t + f(\rho)_x + g(\rho)_y = 0}$$ Conservation law in physical coordinates

typically solved locally through linearization

$$\rho_t + A\rho_x + B\rho_y = 0 \qquad A = \frac{\partial f}{\partial \rho},\quad B = \frac{\partial g}{\partial \rho}$$

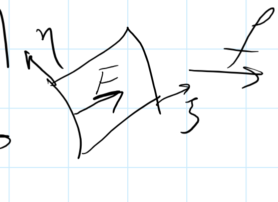
A, B = flux Jacobians

J = (geometric) Jacobian

$$\tilde{\rho}(t,\xi,\eta) = \rho\left(t, X(t,\xi,\eta), Y(t,\xi,\eta)\right)$$

$$(2)\quad \left| (J\tilde{\rho})_t + F(\tilde{\rho})_\xi + G(\tilde{\rho})_\eta = 0 \right.$$

(2) $\boxed{(J\tilde{q})_t + F(\tilde{q})_\xi + G(\tilde{q})_\eta = 0}$



$$F = \begin{vmatrix} f & g \\ x_\eta & y_\eta \end{vmatrix} - \begin{vmatrix} x_t & y_t \\ x_\eta & y_\eta \end{vmatrix} q$$

$$G = \begin{vmatrix} x_\xi & y_\xi \\ f & g \end{vmatrix} - \begin{vmatrix} x_\xi & y_\xi \\ x_t & y_t \end{vmatrix} q$$

in general $\begin{cases} x(\xi,\eta) \\ y(\xi,\eta) \end{cases}$ $\qquad \dfrac{\partial f}{\partial x} = \dfrac{\begin{vmatrix} f_\xi & f_\eta \\ y_\xi & y_\eta \end{vmatrix}}{\begin{vmatrix} x_\xi & x_\eta \\ y_\xi & y_\eta \end{vmatrix}}$

(2) is the expression of conservation law in logical Cartesian coordinates "computational" coordinates

$$\frac{\partial F}{\partial q} = \begin{vmatrix} f_q & g_q \\ x_\eta & y_\eta \end{vmatrix} - \begin{vmatrix} x_t & y_t \\ x_\eta & y_\eta \end{vmatrix} \cdot I$$

Eigenvectors & eigenvalues of the flux Jacobians in computational coordinates are simply linear combinations of those in physical coordinates!

Example: Moving piston (bearclaw/applications/2A/moving mesh/piston)

Auxiliary variables $\begin{pmatrix} x_\xi \\ x_\eta \\ y_\xi \end{pmatrix}$ 7 aux fields

$$\begin{pmatrix} \vdots \\ Y_3 \\ Y_4 \\ X_t \\ Y_t \\ J \end{pmatrix}$$

7 aux fields