The **fixed-point** iterative process and associated error are (where $\rho(\mathbf{P})$ is the spectral radius of $\mathbf{P}$.):

$$\vec{x}^{(0)} = \text{ arbitrary ;} \qquad \vec{x}^{(k+1)} = \mathbf{P}\vec{x}^{(k)} + \tilde{\vec{b}}$$

$$\vec{e}^{(k+1)} = \mathbf{P}\vec{e}^{(k)} \; ; \qquad ||\vec{e}^{(k+1)}|| \leq ||\mathbf{P}^{k+1}|| \, ||\vec{e}^{(0)}|| \; ; \qquad ||\mathbf{P}^k|| \approx \rho^k(\mathbf{P})$$

To reduce error by a factor of ϵ, it takes $k \approx \frac{\log(\epsilon)}{\log(\rho(\mathbf{P}))}$ iterations.

Richardson	Jacobi
$(\mathbf{I} - \omega^{(k)}\mathbf{A})\vec{x}^{(k)} + \omega^{(k)}\vec{b}$	$\mathbf{D}^{-1}(\mathbf{D} - \mathbf{A})\vec{x}^{(k)} + \mathbf{D}^{-1}\vec{b}$
GS	SOR
$(\mathbf{D} + \mathbf{L})^{-1}\big[-\mathbf{U}\vec{x}^{(k)} + \vec{b}\big]$	$(\mathbf{D} + \omega\mathbf{L})^{-1}\Big([(1-\omega)\mathbf{D} - \omega\mathbf{U}]\vec{x}^{(k)} + \omega\vec{b}\Big)$

Table 1: $\vec{x}^{(k+1)}$ for several iterative methods

condition number of a matrix $\mathbf{A}$ is defined as $\kappa(\mathbf{A}) = ||\mathbf{A}|| \, ||\mathbf{A}^{-1}||$. If the 2-norm is used, then $||\mathbf{A}||_2 = \sigma_1$, $||\mathbf{A}^{-1}||_2 = 1/\sigma_m$, and $\kappa_2(\mathbf{A}) = \sigma_1/\sigma_m$; σ_m is the mth singular value of $\mathbf{A}$.

Let $\mathbf{G}$ be a non-singular **preconditioner**, then $\mathbf{A}\vec{x} = \vec{b}$ can be transformed as $\mathbf{G}^{-1}\mathbf{A}x = \mathbf{G}^{-1}b$.

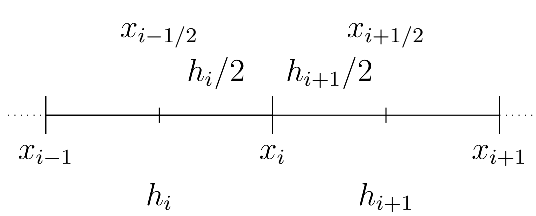
Finite Difference for the DE in 1D we use central difference for the derivative:

$$-\phi_{i-1} + \big(2 + \frac{h^2}{L^2}\big)\phi_i - \phi_{i+1} = h^2\frac{S_{0,i}}{D} \qquad i = 1, 2, \ldots, n-1 \qquad L \equiv \sqrt{\frac{D}{\Sigma_a}}$$

In 2D we use a 5-point stencil and use central in each dimension for the derivative, giving

$$\nabla^2\phi_{i,j} = \frac{\phi_{i+1,j} + \phi_{i-1,j} + \phi_{i,j+1} + \phi_{i,j-1} - 4\phi_{i,j}}{h^2}$$

Finite Volume for the DE in 1D uses this grid



The physics values are defined in each cell (cell-centered) and the flux and source are defined at the mesh points (edge-centered.
We integrate the DE in each cell from $x_{i-1/2}$ to $x_{i+1/2}$. At the boundaries we apply the BCs as appropriate to get the $i = 0$ and $i = n$ equations.

The 2D version uses a 2D stencil (not needed for exam). The physics values are again defined in each cell (cell-centered) and the flux and source are defined at the mesh points (edge-centered.

We again integrate over the partial cell, which is now in two dimensions: $x_{i-1/2}$ to $x_{i+1/2}$ and $y_{j-1/2}$ to $y_{j+1/2}$.
We use Gauss Theorem for the streaming term: $-\int_V d\vec{r}\left[\nabla\cdot\left(D(\vec{r})\nabla\phi(\vec{r})\right)\right] = \int_S d\vec{S}\, D(\vec{r})\frac{\partial}{\partial\hat{n}}\phi(\vec{r})$.

Critical/eigenvalue equations have the addition of k, allowing us to adjust the equation such that the rhs of the DE becomes $\frac{1}{k}\nu\Sigma_f(x)\phi(x)$.
This means the rhs of our FD or FVM matrix formulation contains flux instead of only a fixed source.

$$A = -\frac{d}{dx}D(x)\frac{d}{dx} + \Sigma_a(x) \qquad F = \nu\Sigma_f(x) \qquad k = \frac{\text{total production rate}}{\text{total loss rate}}$$

$$\mathbf{A}\vec{\phi}^{(m+1)} = \frac{1}{k^{(m)}}\mathbf{F}\vec{\phi^{(m)}} \qquad k^{(m+1)} = \frac{\int_0^{\tilde{a}} F\vec{\phi}^{(m+1)}(x)dx}{\frac{1}{k^{(m)}}\int_0^{\tilde{a}} F\vec{\phi}^{(m)}(x)dx}$$

We also add an eigenvalue solver method, like Power Iteration. This uses the idea that
$\vec{v}_0 = \gamma_1\vec{x}_1 + \gamma_2\vec{x}_2 + \cdots + \gamma_n\vec{x}_n$, where $\vec{x}_j$ is the jth eigenvector and γ_j is some constant.

Delayed neutrons let us deal with time variation. β fraction of the total neutrons released from fission are delayed. We use ρ = reactivity = $\frac{k-1}{k}$, and l is mean neutron lifetime.

$$n(t) = n_0 e^{\frac{(k-1)t}{l}}\,; \qquad T = \text{reactor period} = \frac{l}{k-1}$$

$$\phi(r,t) = vn(t)\psi_1(r)\,; \qquad \hat{C}_i(r,t) = C_i(t)\psi_1(r)$$

where ψ_1 is the fundamental mode solution of $\nabla^2\psi_n + B_g^2\psi_n = 0$

PRKE

$$\frac{dn(t)}{dt} = \frac{\rho(t)-\beta}{\Lambda}n(t) + \sum_{i=1}^{6}\lambda_i C_i(t)\,; \qquad \frac{dC_i(t)}{dt} = \frac{\beta_i}{\Lambda}n(t) - \lambda_i C_i(t)$$

$C_i(t)$ = delayed neutron concentration from ith precursor; Λ = effective n lifetime = $(v\nu\Sigma_F)^{-1}$.

Runge Kutta methods use stages to avoid higher order derivates to get better accuracy. Two stage:

$$U^{n+1/2} = U^n + \frac{1}{2}kf(U^n) \qquad U^{n+1} = U^n + kf(U^{n+1/2}) \qquad U^{n+1} = U^n + kf(U^n + \frac{1}{2}kf(U^n))$$

Monte Carlo uses random numbers, sampling rules, tallies, and <u>statistics</u>

$$\mu = E(x) = \int xf(x)dx\,; \qquad \bar{x} = \frac{1}{N}\sum_{i=1}^{N}x_i \quad \lim_{N\to\infty}\bar{x}\to\mu$$

$$\sigma^2 = E(x^2) - \mu^2\,; \qquad S_x^2 \approx \bar{x^2} - \bar{x}^2\,; \qquad S_{\bar{x}}^2 = \frac{S_x^2}{N}$$

$$p\{a \leq x \leq b\} = \int_a^b p(x)dx\ ; \qquad p(x) \geq 0\ ; \qquad \int_{-\infty}^{\infty} p(x)dx = 1$$

$$p(x = x_k) = p_k \equiv p(x_k)\ , k = 1, \ldots, N\ ; \qquad p_k \geq 0\ ; \qquad \sum_{k=1}^{N} p_k = 1$$

$$P\{x' \leq x\} = P(x) = \int_{-\infty}^{x} p(x')dx'\ \ ; \qquad P(-\infty) = 0\ , \quad P(\infty) = 1$$

$$P\{x' \leq x\} = P_k \equiv P(x_k) = \sum_{j=1}^{k} p_j\ , k = 1, \ldots, N\ ; \qquad P_0 = 0\ , \quad P_N = 1$$

Deal with normalization using $p(x) = \frac{g(x)}{G(\infty)}$, where $p(x)$ is the numerical PDF, $g(x)$ is the physical PDF, and $G(x)$ is the physical CDF.

With rejection sampling, create $g(x) \geq p(x)$ for all x; Generate pair of random variables, (ξ, η); $x' = G^{-1}(\xi)$; If $\eta < p(x')/g(x')$, accept x'; Else, reject x'.

Probability of distance to collision

$$\begin{aligned} p_c(s) &= \Sigma_t(s) e^{-\Sigma_t(s)s} ds \\ P_c(n) &= \int_0^s \Sigma_t(s) e^{-\Sigma_t(s)s'} ds' = -e^{-\Sigma_t(s)s'}|_0^s = 1 - e^{-\Sigma_t(s)s} \end{aligned}$$

After a collision, which nuclide from $p_i = \frac{\Sigma_{t,j}}{\Sigma_t}$ and which reaction from $p_x = \frac{\Sigma_{x,j}}{\Sigma_{t,j}}$.

In scattering, ihe new direction is $\big(\sin(\phi)\cos(\theta), \sin(\phi)\sin(\theta), \cos(\theta)\big) = \big(\sqrt{1-\mu^2}\cos(\theta), \sqrt{1-\mu^2}\sin(\theta), \mu\big)$.