2D S_N Radiation Transport with Diffusion Acceleration

MATH 676 – Milestone 1 Presentation

Logan H. Harbour

Department of Nuclear Engineering
Texas A&M University

NUCLEAR ENGINEERING
TEXAS A&M UNIVERSITY

One-group Linear Boltzmann Equation

Begin with the one-group S_N transport equation for a single direction d (neglecting boundary conditions for simplicity), as

$$\mathbf{\Omega}_d \cdot \nabla \psi_d(\mathbf{x}) + (\sigma_a(\mathbf{x}) + \sigma_s(\mathbf{x}))\,\psi_d(\mathbf{x}) - \frac{\sigma_s(\mathbf{x})}{2\pi} \sum_{d=1}^{N_\Omega} \omega_d \psi_d(\mathbf{x}) = q(\mathbf{x})\,, \quad (1)$$

where σ_a represents a probability of particle absorption and σ_s represents a probability of radiation scattering. Let $\mathbb{T}_h$ be the set of all cells of the triangulation in a discontinuous approximation space. The DG weak form with test function v_d is

$$\sum_{K \in \mathbb{T}_h} \Big[(-\mathbf{\Omega}_d \cdot \nabla v_d, \psi_d)_K + \left(\psi_d^+ \mathbf{\Omega}_d \cdot \mathbf{n}, v_d\right)_{\delta K} + (\sigma_t \psi_d, v_d)_K - (\sigma_s \phi, v_d)_K = (q, v_d)_K \Big]\,, \quad (2)$$

where ϕ is the *scalar flux*, $\phi = \frac{1}{2\pi} \sum_d^{N_\Omega} \omega_d \psi_d$, and ψ_d^+ is the upwind value of ψ_d (the value from the side of the face in which $\mathbf{\Omega} \cdot \mathbf{n} \geq 0$).

Source Iteration

We commonly solve the transport equation by *source iteration*, a form of Richardson iteration. Cast Eq. (1) with iterative index ℓ as

$$\boldsymbol{\Omega}_d \cdot \nabla \psi_d^{(\ell+1)} + \sigma_t \psi_d^{(\ell+1)} = \sigma_s \phi^{(\ell)} + q\,, \tag{3}$$

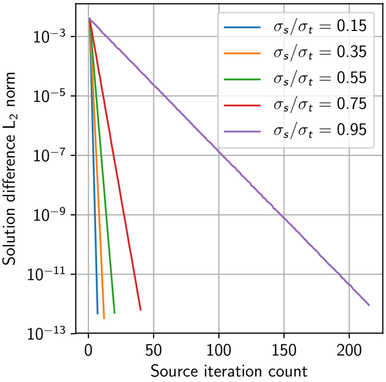
where ℓ is the iterative index, $\psi_d^{(0)} = \phi^{(0)} = \vec{0}$. After solving each direction, d, for an iteration ℓ in Eq. (3), update the scalar flux with

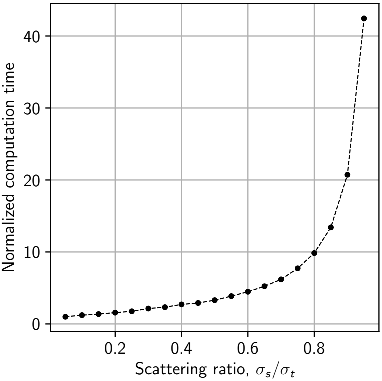
$$\phi^{(\ell+1)} = \frac{1}{2\pi} \sum_{d=1}^{N_\Omega} w_d \psi_d^{(\ell+1)}\,.$$

$\psi^{(\ell+1)}$ is the particles that have scattered at most ℓ times. As $\sigma_s/\sigma_t \to 1$, particles scatter more before they are absorbed $\to$ **the number of source iterations becomes significant!** This problem becomes the goal of this work: introduce a diffusion problem as a preconditioner for Eq. (3).

Lots of Scattering

Introduce $\mathcal{D} = [0, 10]^2$, $N_\Omega = 20$, $q = 1$, $\sigma_a + \sigma_s = \sigma_t = 100$, and 64^2 elements. Increase the scattering ratio, σ_s/σ_t and observe results.

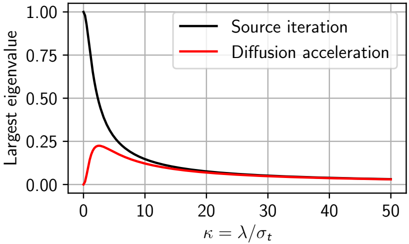




Diffusion Acceleration

- Simple algebraic manipulations can show that the error in $\psi^{\ell+1}$ satisfies the transport equation with a source equal to:

$$R^{\ell+1} = \frac{\sigma_s}{2\pi}(\phi^{\ell+1} - \phi^{\ell})\,.$$



- A Fourier analysis shows (above and to the right for $\sigma_s/\sigma_t = 1$) that the transport equation with a diffusion approximation attenuates the errors most poorly attenuated by the transport sweep.
- **Project goal:** cast the transport problem for the error in $\psi^{\ell+1}$ using the diffusion approximation as an acceleration to source iteration.
  - This requires an S_N solver (complete) and a diffusion solver (starting).

So far...

Completed

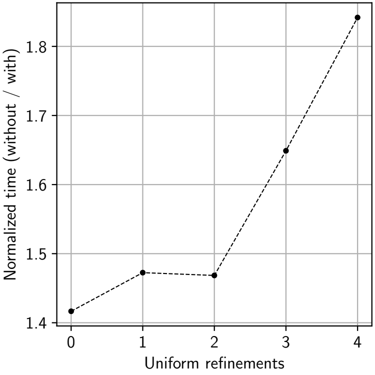
- A one-group, 2D neutron transport code using the S_N approximation has been developed using linear discontinuous finite elements in Deal.ii.
- Verified using known constant source solutions and MMS.
- Primarily uses the `MeshWorker` interface as discussed in step-12.
- `downstream_renumbering` preconditions the within-direction solve.

Frustrations (fewer than expected)

- Discontinuous finite element is new to me. Not like the majority of Deal.ii, `MeshWorker` took some getting used to.
- I really need to quit spending so much time on architecture...
- S_N transport has specified directions of travel, therefore the discontinuous weak form is somewhat intuitive. This is not the case with diffusion transport, therefore the DFEM weak form isn't as fun.

Aside: Downstream renumbering

- The S_N problem with upwind approximation $\rightarrow$ matrix solves can be local if solved in the upstream direction.
- Mixing `downstream()` with a Gauss-Seidel preconditioner mimics this behavior. Otherwise, solve with Richardson iteration.
- Consider a $[0,10]^2$ domain, $N_\Omega = 20, q = 1, \sigma_s = 10, \sigma_t = 100, 25^2$ elements. Refine the mesh uniformly and observe effect of renumbering DoFs based on direction.



Coming soon (maybe? hopefully?)