Prof. Benoit Forget
w/ the help
of his minions

22.106

Neutron Interactions and Applications

Lecture Notes and Problems

Massachusetts Institute of Technology

Contents

Part I

Nuclear Data

Lecture 1 | Microscopic Interactions

Life is made up of small pleasures. Happiness is made up of those tiny successes. The big ones come too infrequently. And if you don't collect all these tiny successes, the big ones don't really mean anything.

Norman Lear

Lecture 2 | Macroscopic Interactions

> The whole is more than the sum of its parts.
>
> Aristotle

Lecture 3 | Shielding and Moderation

> A little more moderation would be good. Of course, my life hasn't exactly been one of moderation.
>
> Donald Trump

Lecture 4 | Neutron Detection and Spectroscopy

Lecture 5 | Nuclear Data: a Black Box?

Lecture 6 | R-Matrix Theory

Lecture 7 | Neutron Thermalization

Lecture 8 | Scattering Laws and S(α,β)

Part II

Monte Carlo Transport

Lecture 9 | Probability and Statistics

Lecture 10 | Monte Carlo Methods

Lecture 11 | Collision Physics and Sampling 'Em

Lecture 12 | Tallies and Uncertainties

The only thing that makes life possible is permanent, intolerable uncertainty; not knowing what comes next.

Ursula Le Guin

Listing 12.1: Main program for a Monte Carlo slab problem

```
program MonteCarloSlab
!-------------------------------------------------------------------------------
! MonteCarloSlab -- A Fortran Monte Carlo Code for Neutron Transport in Slabs
!-------------------------------------------------------------------------------
    use MonteCarloSlabRoutines
    implicit none
    type(Input)     :: Inpt   ! Problem input
    type(Output)    :: Outp ! Problem output
    integer         :: i
    !---------------------------------------------------------------------------
    ! SAMPLE PROBLEM:  (A crafty student will add input file handling!)
    !     vacuum | 4 cm, mat2 | 4 cm, mat1, source=1 | 4 cm, mat2 | vacuum
    ! We divide the domain into 24 cells of with 0.5 cm:
    Inpt%NumCells = 24
    ! Allocate the cell boundary and material id arrays
    allocate( Inpt%CellBounds(Inpt%NumCells+1), Inpt%MatId(Inpt%NumCells) )
    ! Set the boundaries and assign materials
    Inpt%CellBounds = 0.0
    do i = 2, Inpt%NumCells+1
        Inpt%CellBounds(i) = Inpt%CellBounds(i-1) + 0.5
    end do
    Inpt%MatId = (/2,2,2,2,2,2,2,2,1,1,1,1,1,1,1,1,2,2,2,2,2,2,2,2/)
    ! We have two materials, each with a SigmaT, SigmaA, and SigmaS
    Inpt%NumMats = 2
    allocate( Inpt%Sigma(Inpt%NumMats,3) )
    ! Sigma =     material 1  ( SigmaT | SigmaA | SigmaS )
    !             material 2  ( ...
    Inpt%Sigma(1,1) = 1.0d0; Inpt%Sigma(1,2) = 0.5d0; Inpt%Sigma(1,3) = 0.5d0
    Inpt%Sigma(2,1) = 1.5d0; Inpt%Sigma(2,2) = 1.2d0; Inpt%Sigma(2,3) = 0.3d0
    ! Let's do a million histories
    Inpt%NumHist = 1e6;
    ! Allocate the tallies
    allocate( Outp%TrakEst(Inpt%NumCells), Outp%CollEst(Inpt%NumCells) )
    call PlayGame(Inpt,Outp)
        !**** INSERT TALLY POST-PROCESSING HERE (i.e. flux computation, etc.)
    ! Deallocate input
```

```
    deallocate( Inpt%CellBounds, Inpt%MatId, Inpt%Sigma )
    ! Deallocate output
    deallocate( out%TrakEst, out%CollEst )
end program MonteCarloSlab
```

Listing 12.2: Subroutines for a Monte Carlo slab problem

```
module MonteCarloSlabRoutines
    implicit none
    type Input ! Data structure for holding all relevant problem information
        ! cell boundaries
        double precision, allocatable, dimension(:)   :: CellBounds
        ! material id's in each cell
        integer, allocatable, dimension(:)            :: MatId
        ! cross-sections
        double precision, allocatable, dimension(:,:) :: Sigma
        integer :: NumCells, & ! number of cells
                   NumMats,  & ! number of different materials
                   NumHist     ! number of histories
    end type Input
    type Output ! Data structure for holding the relevant tally data
        double precision, allocatable, dimension(:) :: TrakEst, CollEst
    end type Output
contains

subroutine PlayGame(in,out)
!-------------------------------------------------------------------------------
! subroutine PlayGame(in,out)
! Play the game, or do the simulation, or let particles roam free.
!   Inputs:
!       in          -- data structure containing relevant input data
!   Outputs:
!       out         -- data structure of output indomation
!       Outp%trakest -- track length estimator tally
!       Outp%collest -- collision estimator tally
    type(Input), intent(in)         :: Inpt
    type(Output), intent(inout)     :: Outp
!       Local Variables
    double precision, allocatable, dimension(:) :: tmptrak,& ! temporary
                                                   tmpcoll   ! tallies
    integer ::        alive,        & ! Whether or not I'm alive
                      coll,         & ! Whether or not I'm exiting a collision
                      mycell,       & ! My current cell
                      neighbor,     & ! My neighbor
                      n,m=0

    double precision  mu,           & ! My direction
                      x,            & ! My location
                      SigT,         & ! SigmaT of mycell
                      mfps,         & ! Mean free paths to travel
                      dist,         & ! Distance along x axis to travel
                      d2neighbor      ! Distance (along x) to neighbors)
    double precision, parameter :: one = 1.0d0 ! for use in "sign" function
!-------------------------------------------------------------------------------
    print *, 'beginning...'
    ! Initialize built-in random number generator
    call random_seed()
    ! Set the estimator tallies
    Outp%TrakEst = 0
    Outp%CollEst = 0
    ! Allocate the temporary tally arrays
    allocate( tmptrak(Inpt%NumCells), tmpcoll(Inpt%NumCells) )

    !--------------------------------------------------------------------------
    ! Let the histories begin!
    !--------------------------------------------------------------------------
```

```fortran
histories: do n = 1, Inpt%NumHist
    ! Get my starting location and direction
    call Source(in,x,mu)
    ! Reset temporary tallies
    tmptrak = 0.0
    tmpcoll = 0.0
    alive   = 1
    coll    = 0
    call GetMyCell(x,in,mycell) ! Get my cell

    !----------------------------------------------------------------------
    ! Welcome to the life of a neutron!
    !----------------------------------------------------------------------
    do while ( alive .eq. 1 )
        if ( coll .eq. 1 ) then
            ! I need a new direction in life
            mu = 2*rand()-1
        end if
        ! Determine my neighbor and how far to her
        call GetMyNeighbor(mycell,x,mu,in,neighbor,d2neighbor)
        ! Sample how many mfp's I will go
        mfps    = -log(rand())
        ! Determine my total cross-section
        SigT = Inpt%Sigma(Inpt%MatId(mycell),1)
        ! Determine how far along x-axis I go
        dist = mfps*mu/SigT
        ! Determine whether I reach surface or collide
        if ( abs(dist) .gt. d2neighbor) then
            ! Then I pass to the next cell, so update track length tally
            tmptrak(mycell) = tmptrak(mycell) + abs(d2neighbor/mu)
            if  ( (neighbor.eq.0) .or. (neighbor.eq.(Inpt%NumCells+1)) ) then
                ! I leaked out of left or right
                alive = 0
            else
                ! I get to the boundary
                mycell  = neighbor
                x       = x + sign(one,mu)*d2neighbor
                coll    = 0
            end if
        else
            ! Otherwise, I collide, so update both tallies
            tmpcoll(mycell) = tmpcoll(mycell) + 1
            tmptrak(mycell) = tmptrak(mycell) + mfps/SigT
            ! Update my location
            x = x + dist
            ! Sample the actual collision
            if ( rand() .lt. Inpt%Sigma(Inpt%MatId(mycell),2)/SigT ) then
                ! I am absorbed
                alive = 0
            else
                ! I am scattered
                coll = 1
            end if
        end if
    end do ! while
    !----------------------------------------------------------------------
    ! R.I.P.
    !----------------------------------------------------------------------

    Outp%TrakEst = Outp%TrakEst + tmptrak ! Update tallies; these are "S1"
    Outp%CollEst = Outp%CollEst + tmpcoll ! counters; S2 would be here too!

end do histories
!--------------------------------------------------------------------------
! Game Over.
!--------------------------------------------------------------------------
print *, '...ending'
```

```fortran
end subroutine PlayGame

subroutine Source(Inpt,x,mu)
!------------------------------------------------------------------------------
! subroutine Source(in,x,mu)
! Defines the source distribution.
!   Inputs:
!       in  -- data structure containing relevant input data
!   Outputs:
!       x   -- my current position
!      mu   -- my current direction (i.e. cosine w/r to x axis).
    type(Input), intent(in)         :: Inpt
    double precision, intent(out)   :: x, mu
!------------------------------------------------------------------------------
    ! Example: We are born in the middle 4 slabs unidomly and
    !          isotropically (in the LAB!)
    x  = rand()*4.0+4.0
    mu = 2.0*rand()-1.0
end subroutine Source

subroutine GetMyNeighbor(mycell,x,mu,Inpt,neighbor,d2neighbor)
!------------------------------------------------------------------------------
! subroutine GetMyNeighbor(mycell,x,mu,Inpt,neighbor,d2neighbor)
! Finds out who my neighboring cell is.
!   Inputs:
!       mycell      -- my current home
!       x           -- my current x coordinate
!       mu          -- my current direction
!       in          -- data structure containing relevant input data
!   Outputs:
!       neighbor        -- the next cell over (in my direction)
!       d2neighbor  -- distance along x axis bedoe I enter neighbor
    integer, intent(in)             :: mycell
    double precision, intent(in)    :: x, mu
    type(Input), intent(in)         :: Inpt
    integer, intent(out)            :: neighbor
    double precision, intent(out)   :: d2neighbor
!------------------------------------------------------------------------------
    if ( mu .gt. 0.0 ) then
        d2neighbor = Inpt%CellBounds(mycell+1) - x
        neighbor   = mycell + 1
    else
        d2neighbor = x - Inpt%CellBounds(mycell)
        neighbor   = mycell - 1;
    end if
end subroutine GetMyNeighbor

subroutine GetMyCell(x,Inpt,mycell)
!------------------------------------------------------------------------------
! subroutine GetMyCell(x,in,mycell)
! Given my x coordinate, find out my cell.
!   Inputs:
!       x           -- my current x coordinate
!       in          -- data structure containing relevant input data
!   Outputs:
!       mycell      -- my current home
    double precision, intent(in)    :: x
    type(Input), intent(in)         :: Inpt
    integer, intent(out)            :: mycell
!------------------------------------------------------------------------------
    do mycell = 1, Inpt%NumCells
        if ( x < Inpt%CellBounds(mycell+1) ) then
            exit
        end if
    end do
end subroutine GetMyCell
```

```
double precision function rand()
!-----------------------------------------------------------------------------
! Use a Matlab-like call for random numbers
!-----------------------------------------------------------------------------
    call random_number(rand)
end function rand

end module MonteCarloSlabRoutines
```

Lecture 13 | Variance Reduction

Lecture 14 | Criticality: Computing with Monte Caro and Safety Aspects

Exercises

1. Write a short Monte Carlo program to determine the eigenvalue k for an infinite slab of width 5 with total cross-section $\Sigma = 1.0$, $\Sigma_S = 0.5$, and $\nu\Sigma_F$ = 0.8. Compare your solution to an analytic diffusion solution using extrapolated conditions (i.e. $\tilde{a} = a + a_0$, where a_0 = 0.7104/Σ).

Part III

Deterministic Transport

Lecture 15 | The Transport Equation

In this lecture, we introduce general *transport equations* and finish by developing the form used in neutron transport theory. In the lectures to follow, we shall describe other aspects of the neutron transport equation and methods by which it can be solved both analytically and numerically.

Transport Theory

Transport theory aims to describe mathematically the movement (i.e. "transport") of particles as they traverse a medium. For example, we might describe the transport of high energy gammas through a lead shield, or the movement of neutrons through uranium dioxide pellets. We might also describe the movement of particles in a dense gas as they navigate through a medium consisting of the gas itself.

In all cases, transport theory describes such processes in an *average* sense. For instance, we do not compute the individual trajectories of neutrons in a reactor via transport theory. That, instead, would require molecular dynamics, in which Newton's equation of force is solved for the many-bodied problem of all neutrons in the vicinity of interest (an essentially impossible problem), or perhaps the Monte Carlo methods described in previous lectures, where a sample of individual particles are tracked to approximate ensemble averages (a difficult, but as we've seen, tractable problem). Hence, the quantities we shall compute using the equations of transport theory should be recognized as expected and not exact values.

Fundamental Quantities

We begin by defining several fundamental quantities. It should be noted that the forms introduced at first are likely different from what you might have seen in a previous reactor physics course. The purpose here is two-fold. First, we wish to introduce the quantities and eventually the equations in a general way to make clear that transport theory is not restricted to the neutron transport equation. Second, for those who might be familiar with e.g. the Boltzmann equation of gas dynamics (and not neutron transport), the notation will be familiar and lead smoothly into our more familiar form.

We first define the *phase space density function*, the knowledge which we can use to compute essentially all quantities of interest:

$n(\mathbf{r}, \mathbf{v}, t) d^3r d^3v \equiv$ expected number of particles in d^3r about $\mathbf{r}$ with velocity dv about $\mathbf{v}$ at time t.

It is often most convenient to break the velocity into its scalar (speed) and vector (direction) components. The scalar component is recast in the energy variable via $E = mv^2/2$, and the direction vector is defined $\mathbf{\Omega} = \mathbf{v}/|\mathbf{v}|$. The phase space density can then be rewritten as

$n(\mathbf{r}, \mathbf{\Omega}, E, t) d^3r d\Omega dE \equiv$ expected number of particles in d^3r about $\mathbf{r}$ going in the directions $d\Omega$ about $\mathbf{\Omega}$ with energy dE about E at time t.

In this form, $n(\mathbf{r}, \mathbf{\Omega}, E, t)$ is often referred to as the angular density.

We can relate the phase space densities in terms of $\mathbf{v}$ and $(E, \mathbf{\Omega})$ via the relations

$$
\begin{aligned}
n(\mathbf{r}, E, \mathbf{\Omega}, t) &= (v/m) n(\mathbf{r}, \mathbf{v}, t) \\
n(\mathbf{r}, E, \mathbf{\Omega}, t) &= (1/mv) n(\mathbf{r}, v, \mathbf{\Omega}, t) \\
n(\mathbf{r}, v, \mathbf{\Omega}, t) &= v^2 n(\mathbf{r}, \mathbf{v}, t) \,,
\end{aligned}
\tag{15.1}
$$

proofs of which are left as exercises.

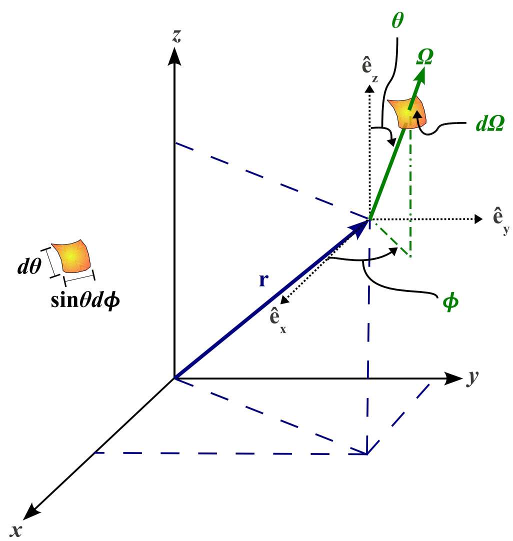

Figure 15.1: Schematic of Phase Space.

Figure 15.1 depicts a schematic of the phase space used in terms of the position $\mathbf{r}$ and direction $\mathbf{\Omega}$. The position vector is further broken down into the polar angle θ and azimuthal angle ϕ. The differential solid angle element $d\Omega$ is also shown, and can be expressed in terms of θ and ϕ via

$$d\Omega = \sin\theta d\theta d\phi \,.$$

One might envision such solid angle elements as the small bumps on basketballs.

A quantity closely related to the phase space density (or current density) is the *angular current density*:

$$\mathbf{j}(\mathbf{r},\mathbf{v},t)\cdot d\mathbf{S}d^3v = \mathbf{v}n(\mathbf{r},\mathbf{v},t)\cdot d\mathbf{S}d^3v \equiv$$ expected number of particles that cross an area dS per second with velocity d^3v about $\mathbf{v}$ at time t.

We can also define *partial currents* with respect to a particular surface S defined by an outward normal vector $\hat{\mathbf{e}}_s$:

$$J_{\pm}(\mathbf{r},t) = \pm\int_{\pm} d^3v\hat{\mathbf{e}}_s\cdot\mathbf{j}(\mathbf{r},\mathbf{v},t) \equiv$$ the rate at which particles flow through S in the outward (+) or inward (-) direction.

The *current density* $\mathbf{J}(\mathbf{r},t)$ is defined by integrating the angular current density over all velocities. Then for our surface S, the net rate of particles passing outward through S is just $\mathbf{J}(\mathbf{r},t)\cdot\hat{\mathbf{e}}_s$. From our definition of partial currents, the net current passing outward must also be $J_+ - J_-$, yielding the useful identity

$$\mathbf{J}(\mathbf{r},t)\cdot\hat{\mathbf{e}}_s = J_+(\mathbf{r},t) - J_-(\mathbf{r},t)\,. \tag{15.2}$$

Of particular interest to us in the next section will be reaction rates, which can most easily be described using the *angular flux*

$$\psi(\mathbf{r},\mathbf{\Omega},E,t) = vn(\mathbf{r},\mathbf{\Omega},E,t) \equiv \quad \text{angular flux}$$

and *scalar flux*

$$\phi(\mathbf{r},E,t) = \int_{4\pi} d\Omega\psi(\mathbf{r},\mathbf{\Omega},E,t) \equiv \quad \text{scalar flux.}$$

A General Transport Equation

Consider an arbitrary volume V with a surface S. Our goal is to represent the time rate of change of the particle density $n(\mathbf{r},\mathbf{v},t)$ within the volume. Neglecting external forces, the only factors affecting the density are collisions within the volume that change a particle's velocity, the streaming of particles into and out of the surface, and any internal source of particles. This simple balance can be expressed mathematically as

$$
\begin{aligned}
\overbrace{\frac{\partial}{\partial t}\left(\int_V d^3r n(\mathbf{r},\mathbf{v},t)\right)}^{\text{total rate of change of } n \text{ in } V} &= -\overbrace{\int_S dS\hat{\mathbf{e}}_S\cdot\mathbf{j}(\mathbf{r},\mathbf{v},t)}^{\text{streaming rate}} + \overbrace{\int_V d^3r\Big(\frac{\partial n}{\partial t}\Big)_{\text{coll}}}^{\text{collision rate}} \\
&\quad + \underbrace{\int_V d^3r s(\mathbf{r},\mathbf{v},t)}_{\text{source emission rate}}\,,
\end{aligned}
\tag{15.3}
$$

where s represents a source inside the volume and $(\partial n/\partial t)_{\text{coll}}$ is the time rate of change due to collisions, specific forms of which are application-dependent and will be discussed below. Note the minus sign on the surface integral, i.e. the streaming term. Since the integral describes the net rate of neutrons going *out* of the surface, we negate it so that a positive net rate directed inward is a positive contribution to the total time rate of change of n in V.

Eq. 20.24 gives us a simple relation in terms of both volume and surface integrals. Our life is always easiest if we have the same integration on both sides. By the divergence (or Gauss') theorem, we can rewrite the streaming term

$$
\int_S dS\hat{\mathbf{e}}_S\cdot\mathbf{j}(\mathbf{r},\mathbf{v},t) = \int_V d^3r\nabla\cdot\mathbf{j}(\mathbf{r},\mathbf{v},t)\,. \tag{15.4}
$$

Since ∇ acts on $\mathbf{r}$ and not $\mathbf{v}$, we note $\nabla\cdot\mathbf{j} = \nabla\cdot(\mathbf{v}n) = \mathbf{v}\cdot\nabla n + \overbrace{n\nabla\cdot\mathbf{v}}^{\to 0} = \mathbf{v}\cdot\nabla n$. Hence, the streaming term becomes

$$
\int_V d^3r\nabla\cdot\mathbf{j}(\mathbf{r},\mathbf{v},t) = \int_V d^3r\mathbf{v}\cdot\nabla n(\mathbf{r},\mathbf{v},t)\,. \tag{15.5}
$$

For a constant volume, $(\partial/\partial t)\int_V d^3rn = \int_V d^3(\partial n/\partial t)$, and so our balance equation can be rewritten as

$$
\begin{aligned}
\overbrace{\left(\int_V d^3r\frac{\partial}{\partial t}n(\mathbf{r},\mathbf{v},t)\right)}^{\text{total rate of change of } n \text{ in } V} &= -\overbrace{\int_V d^3r\mathbf{v}\cdot\nabla n(\mathbf{r},\mathbf{v},t)}^{\text{streaming rate}} + \overbrace{\int_V d^3r\Big(\frac{\partial n}{\partial t}\Big)_{\text{coll}}}^{\text{collision rate}} \\
&\quad + \underbrace{\int_V d^3r s(\mathbf{r},\mathbf{v},t)}_{\text{source emission rate}}\,.
\end{aligned}
\tag{15.6}
$$

For an arbitrary volume V, the integrands of Eq. 15.6 must vanish, yielding a general transport equation:

$$
\frac{\partial}{\partial t}n(\mathbf{r},\mathbf{v},t) = -\mathbf{v}\cdot\nabla n(\mathbf{r},\mathbf{v},t) + \Big(\frac{\partial n}{\partial t}\Big)_{\text{coll}} + s(\mathbf{r},\mathbf{v},t)\,. \tag{15.7}
$$

Even More Generality

We can skip the differential volume formulation by considering the material derivative of n (using Cartesian coordinates):

$$\begin{aligned}\frac{Dn}{Dt} &\equiv \frac{\partial n}{\partial t} + \frac{\partial x}{\partial t}\frac{\partial n}{\partial x} + \frac{\partial y}{\partial t}\frac{\partial n}{\partial y} + \frac{\partial z}{\partial t}\frac{\partial n}{\partial z} + \frac{\partial v_x}{\partial t}\frac{\partial n}{\partial v_x} + \frac{\partial v_y}{\partial t}\frac{\partial n}{\partial v_y} + \frac{\partial v_z}{\partial t}\frac{\partial n}{\partial v_z} \\ &= \frac{\partial n}{\partial t} + \mathbf{v}\cdot\nabla n + \mathbf{a}\cdot\nabla_{\mathbf{v}} n \\ &= \frac{\partial n}{\partial t} + \mathbf{v}\cdot\nabla n + \frac{\mathbf{F}}{m}\cdot\nabla_{\mathbf{v}} n \,,\end{aligned} \tag{15.8}$$

where $\nabla_{\mathbf{v}}$ is the gradient operator with respect to velocity components (rather than spatial coordinates). The material derivative is the total time rate of change, accounting for convective (streaming) effects as well as the influence of an external force $\mathbf{F}$, a factor we did not account for above.

This total rate of change must be balanced by sources and sinks, which are the collision and internal source terms. Hence, an even more general transport equation can be written

$$\frac{\partial n}{\partial t} + \mathbf{v}\cdot\nabla n + \frac{\mathbf{F}}{m}\cdot\nabla_{\mathbf{v}} n = \left(\frac{\partial n}{\partial t}\right)_{\text{coll}} + s \,. \tag{15.9}$$

Neutron Transport

To arrive at the neutron transport equation, we bring back the macroscopic cross-sections studied in Lecture 2. Using our definition for the scalar flux, the volumetric collision rate at a particular point in phase space and time is simply

$$R_{\text{coll}}(\mathbf{r},\boldsymbol{\Omega},E,t) = \psi(\mathbf{r},\boldsymbol{\Omega},E,t)\Sigma_t(\mathbf{r},E)\,. \tag{15.10}$$

However, we know that neutrons at one energy and angle can scatter into another energy and angle, and so in general, the rate at which neutrons at any angle and energy are scattered into a particular energy and angle is

$$R_{\text{in-scatter}}(\mathbf{r},\boldsymbol{\Omega},E,t) = \int_0^\infty dE' \int_{4\pi} d\Omega' \Sigma_s(\mathbf{r},\boldsymbol{\Omega}\cdot\boldsymbol{\Omega}',E'\to E)\psi(\mathbf{r},\boldsymbol{\Omega}',E',t)\,. \tag{15.11}$$

The time rate of change due to collisions is thus

$$\begin{aligned}\left(\frac{\partial n}{\partial t}\right)_{\text{coll}} &= -\psi(\mathbf{r},\boldsymbol{\Omega},E,t)\Sigma_t(\mathbf{r},E) \\ &\quad + \int_0^\infty dE' \int_{4\pi} d\Omega' \Sigma_s(\mathbf{r},\boldsymbol{\Omega}'\to\boldsymbol{\Omega},E'\to E)\psi(\mathbf{r},\boldsymbol{\Omega}',E',t)\,.\end{aligned} \tag{15.12}$$

Substituting this into the general transport equation, using $\psi = vn$, and neglecting any external forces, we find the neutron transport equation:

$$\begin{aligned}\frac{1}{v}\frac{\partial\psi}{\partial t} + \hat{\Omega}\cdot\nabla\psi + \Sigma_t\psi(\mathbf{r},\mathbf{\Omega},E,t) = \\ + \int_0^\infty dE' \int_{4\pi} d\Omega' \Sigma_s(\mathbf{r},\mathbf{\Omega}\cdot\mathbf{\Omega}',E'\to E)\psi(\mathbf{r},\mathbf{\Omega}',E',t) + s\,.\end{aligned} \tag{15.13}$$

Note, the source term s has been used to represent internal sources but could also account for other sources such as fission (though the specific form is, of course, hidden).

Assumptions for the Neutron Transport Equation

In writing down Eq. 15.13 as we have, a number of assumptions have been made explicitly or implicitly. These include:

1. The neutron density is large so that it makes sense to be computing for mean values (which is all transport equations can provide)

2. The neutrons are point particles, meaning that wave effects are insignificant

3. Collisions are well-defined, two-body interactions that occur instantaneously (delayed neutrons from fission, which are not covered here, are a notable exception and deserve special treatment)

4. Between collisions, neutrons stream with a constant velocity

5. Neutron-neutron interactions are negligible

6. The properties of the medium are assumed known and time-independent (burnup in a reactor is another exception)

7. The medium is taken to be isotropic (i.e. no directional-dependence)

Further Reading

This lecture follows quite closely the treatment of transport theory in Chapter 1 of Duderstadt and Martin [10]. The reader is encouraged to read that chapter (uploaded to Stellar) and others (MIT libraries should have a copy for the eager beaver). Bell and Glasstone [3] give a more traditional derivation, as do Duderstadt and Hamilton [9].

Exercises

1. Prove the relations given in Eq. 15.1.
2. In Lecture 14, the eigenvalue problem, i.e. a problem without an external source, was introduced in operator form. You probably also know the eigenvalue diffusion equation from reactor physics.
   (a) Write down the 1-d, one group transport equation for an eigenvalue problem in slab geometry (you need to determine the fission source term)
   (b) Assuming isotropic scattering, and an infinite homogeneous, derive a simple expression for k in terms of the cross-sections; what does this expression represent?

Lecture 16 | Boundary and Initial Conditions, and Some Other Transport Equations

The last lecture introduced both general transport equations and the neutron transport equation. In this lecture, we discuss several boundary and interface conditions for constraining the transport equation. We finish by briefly describing two additional transport equations that help us recognize special features of the neutron transport equation.

Boundary and Initial Condtions

In the last lacture, we finished with the Eq. 15.13 neutron transport equation:

$$\frac{1}{v}\frac{\partial \psi}{\partial t} + \hat{\Omega}\cdot\nabla\psi + \Sigma_t\psi(\mathbf{r},\mathbf{\Omega},E,t) = \\ + \int_0^\infty dE' \int_{4\pi} d\Omega' \Sigma_s(\mathbf{r},\mathbf{\Omega}\cdot\mathbf{\Omega}',E'\to E)\psi(\mathbf{r},\mathbf{\Omega}',E',t) + s\,.$$

This is an integro-differential equation in 7 variables: 3 in space, 2 in angle, 1 in energy, and time. Like all differential equations, the transport equation requires initial and boundary conditions.

Initial Conditions

Initial conditions for the transport equation are relatively straightforward. At some initial time t_0, an initial condition is expressed as

$$\psi(\mathbf{r},\mathbf{\Omega},E,t)|_{t=t_0} = f(\mathbf{r},\mathbf{\Omega},E)\,, \tag{16.1}$$

where f represents a known function of space, angle, and energy.

Time-dependent problems in neutron transport are often quite challenging due to the wide range of time scales involved. A good example comes from the study of reactor kinetics, where the time scales range from prompt neutron lifetimes (on the order of 10^{-5} seconds) to the delayed neutrons of longest lifetime (on the order of tens of seconds). Any numerical scheme is effectively limited by the smallest time scale, leading to a "stiff" problem.

Other neutron transport problems exhibit even more diverse time scales. The time scale for nuclear weapons is perhaps most easily quantified with "shakes", that is 10^{-8} seconds. Isotopic changes in nuclear reactors due to irradiation can have profound effects on time scales ranging from hours (xenon production) up to months or years (burnup).

Frequently, we are interested in steady state values, so that $\frac{\partial\psi}{\partial t} = 0$. If the source s in Eq. 15.13 includes an external source, then the problem is a *fixed source problem*. If s does not include an external source (and includes only e.g. a fission source), then the problem becomes an *eigenvalue problem*, which are studied in Lecture 14.

Boundary Conditions

The most straightforward boundary condition to enforce is the *free surface* or *vacuum* condition. Physically, the condition represents the situation where no neutrons enter a volume from the outside. In other words, the volume of interest can be thought to exist in a void. Mathematically, the condition is expressed

$$\psi(\mathbf{r}, \mathbf{\Omega}, E, t) = 0\,, \quad \hat{\mathbf{n}} \cdot \mathbf{\Omega} < 0\,, \tag{16.2}$$

where $\hat{\mathbf{n}}$ is the unit *outward normal* vector to the surface of interest. Since $\hat{\mathbf{n}} \cdot \mathbf{\Omega}$ is just the cosine of the angle between the incident neutrons and the normal vector, we see the flux vanishes whenever that cosine is negative, or whenever the neutron direction is inward.

A point of warning: reentrant geometries must be avoided when using vacuum conditions. Unless treated with special care, reentrant geometries lead to inconsistency. Neutrons leaving one portion of the geometry could, in theory, reenter another portion, but since vacuum conditions disallow this, the true problem is not modeled correctly. A common example of this occurs when "squaring" an exterior cylindrical boundary, as exhibited in Figure 16.1.

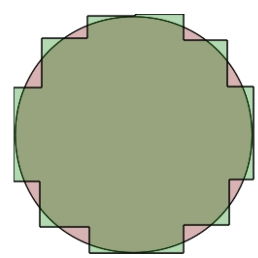

Figure 16.1: A reentrant square cyclinder.

Another useful boundary condition is simply to specify the incident flux when it is known:

$$\psi(\mathbf{r}_s, \mathbf{\Omega}, E, t) = f(\mathbf{r}_s, \mathbf{\Omega}, E, t)\,. \tag{16.3}$$

In this way, boundary sources can be defined.

A *reflective* or *specular* condition is such that

$$\psi(\mathbf{r}_s, \mathbf{\Omega}, E, t) = \psi(\mathbf{r}_s, \mathbf{\Omega}_R, E, t)\,, \qquad \hat{\mathbf{n}} \cdot \mathbf{\Omega} < 0\,, \tag{16.4}$$

where $\mathbf{\Omega}_R$ is the (mirror) reflection of $\mathbf{\Omega}$. Reflective conditions are widely used in lattice physics, where pin cells or assemblies are modeled in an infinite array; the "infinite" is captured by the reflective conditions. See Figure 16.2.

A variation on reflective conditions is an *albedo* condition, where

$$\psi(\mathbf{r}_s, \mathbf{\Omega}, E, t) = \alpha\psi(\mathbf{r}_s, \mathbf{\Omega}_R, E, t)\,, \qquad \hat{\mathbf{n}} \cdot \mathbf{\Omega} < 0\,. \tag{16.5}$$

Here, α is the "albedo" and quantifies the strength with which neutrons stream back into the system after streaming out. Historically, albedo conditions were highly useful since they can often capture the physics of reflectors with minimum computational cost. The albedos for many materials were precomputed (or found experimentally), effectively eliminating a significant portion of phase space in e.g. reactor analysis.

Another approach is to use *periodic* conditions, such that

$$\psi(\mathbf{r}_1, \mathbf{\Omega}, E, t) = \psi(\mathbf{r}_2, \mathbf{\Omega}, E, t)\,, \tag{16.6}$$

Figure 16.2 illustrates both reflective and periodic boundary conditions. Periodic conditions work well in infinite arrays that have assymetric unit cells (for which reflective conditions would represent an infinite but incorrect array).

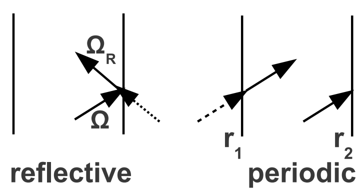

Figure 16.2: Reflective and periodic boundary conditions.

The final boundary condition we mention is the *white boundary condition*, a condition where all neutrons incident on a boundary reflect back isotropically in angle. For this case,

$$\begin{aligned}\psi(\mathbf{r}_s, \mathbf{\Omega}, E, t) &= \frac{\int_{\hat{\mathbf{n}}\cdot\mathbf{\Omega}'>0} \hat{\mathbf{n}} \cdot \mathbf{\Omega}' \psi(\mathbf{r}, \mathbf{\Omega}', E, t) d\Omega'}{\int_{\hat{\mathbf{n}}\cdot\mathbf{\Omega}'>0} \hat{\mathbf{n}} \cdot \mathbf{\Omega}' d\Omega'} \\ &= \frac{J_+(\mathbf{r}_s, E, t)}{\int_{\hat{\mathbf{n}}\cdot\mathbf{\Omega}'>0} \hat{\mathbf{n}} \cdot \mathbf{\Omega}' d\Omega'}\,, \qquad \hat{\mathbf{n}} \cdot \mathbf{\Omega}' < 0\,.\end{aligned} \tag{16.7}$$

Note the conditions on $\mathbf{\Omega}$ and $\mathbf{\Omega}'$. The first corresponds to the left hand side and is limited to $\hat{\mathbf{n}} \cdot \mathbf{\Omega}' < 0$, i.e. incoming directions. Contrarily, $\mathbf{\Omega}'$ is the dummy variable on the right hand side, and is always integrated over the domain where $\hat{\mathbf{n}} \cdot \mathbf{\Omega}' > 0$, i.e. outgoing directions. This is so because we integrate the entire outgoing neutron population (which is proportional to the outgoing partial current) and then redistribute that number uniformly over all incident directions, i.e. isotropically.

White boundary conditions have had use in lattice physics where an isotropic angular distribution is sometimes relatively accurate. In particular, the white boundary condition provides a useful fix for reflective conditions in Wigner-Seitz cells, which convert square pin cells into equivalent cylindrical cells, since cylindrical cells can be treated with 1-d methods. However, while in square cells the reflective conditions work fine, they do not work well in cylindrical geometries (see Figure 16.3), since neutrons entering at certain angles can spend too much time in the moderator before colliding. This consequently leads to overprediction of the moderator flux, an artifact known as the Newmarch effect. As a result, white conditions are used.

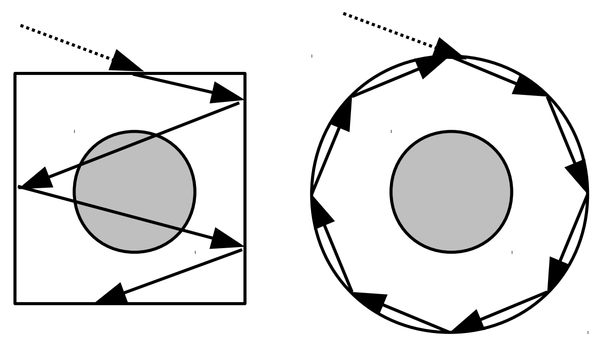

Figure 16.3: Square pin cell and equivalent Wigner-Seitz cell. Same incident direction and location.

Other Transport Equations

We finish this lecture by presenting in brief two other important transport equations.

Photon Transport

Photon transport is fundamental to radiation hydrodynamics (an integral aspect of "bomb" physics) and astrophysics. Photon transport can largely be divided into two classes of problems: *radiative transfer*, which consists of the propogation of soft (low energy) x-rays, and *high energy* photon transport, which can largely be treated as we do neutrons. We briefly describe the former.

The quantity of interest is the intensity, essentially an "energy angular flux", and is defined

$$I_\nu(\mathbf{r}, \mathbf{\Omega}, t) = (h\nu)cn(\mathbf{r}, \mathbf{\Omega}, E, t)\,, \tag{16.8}$$

where $h\nu$ is the photon energy. The "radiative transfer equation" is

$$\frac{1}{c}\frac{\partial I_\nu}{\partial t} + \Omega \cdot \nabla I_\nu = \rho(\varepsilon_\nu - \kappa_\nu I_\nu)\,, \tag{16.9}$$

where ε is a mass emission coefficient (a source term), and κ is a mass attenuation coefficient (a loss term). The radiative transfer equations are nonlinear due to the temperature-dependence of the underlying interaction coefficients (the "cross-sections), particularly the emission term (which is not explicitly represented in Eq. 16.9).

For the case of local thermodynamic equilibrium (LTE), Eq. 16.9 is simplified somewhat. Local thermodynamic equilibrium exists when the quantity $S_\nu = \varepsilon_\nu/\kappa_\nu = B_\nu$, where B_ν is the Planck distribution (i.e. the black body spectrum). In this case, Eq. 16.9 takes the form

$$\frac{1}{c}\frac{\partial I_\nu}{\partial t} + \Omega \cdot \nabla I_\nu = \rho\kappa_\nu(B_\nu - I_\nu)\,. \tag{16.10}$$

Radiative transfer is inherently a frequency-dependent process: the coefficients depend on the frequency, and a medium emits photons of a wide range of frequencies. A frequently used approximation is to neglect this dependence in what is called the *grey approximation*, similar to the one-speed studies in neutron transport we will study in the next several lectures. "Black bodies" are also often used; these are pure absorbers whose emission spectrum is the Planck distribution.

To determine the temperature (dependence on which is implicit in all the quantities of Eq. 16.9), an energy conservation equation is used. As an example, using the grey approximation and assuming temperature- and spatially-independent coefficients, Eq. 16.9 becomes

$$\frac{1}{c}\frac{\partial I}{\partial t} + \Omega \cdot \nabla I(\mathbf{r}, \mathbf{\Omega}, t) = \rho\kappa(acT^4(\mathbf{r}, t) - I)\,, \tag{16.11}$$

where a is the emissivity (as in the Stefan-Boltzmann law) and T is the temperature. The corresponding energy conservation equation is

$$\overbrace{c_v\frac{\partial T}{\partial t}}^{\text{energy rate of change}} = \overbrace{\rho\kappa}^{\text{abs. coef.}} \overbrace{\int_{4\pi} I(\mathbf{r}, \mathbf{\Omega}, t)d\Omega}^{\text{energy flux}} - \overbrace{\rho\kappa acT^4(\mathbf{r}, t)}^{\text{loss due to emission}} + \overbrace{Q(\mathbf{r}, t)}^{\text{gains from outside}}\,, \tag{16.12}$$

where c_v is the specific heat and Q represents any external energy source.

Plasma Transport

Another area of interest for nuclear engineers is plasma physics. Let us apply Eq. 15.9 to electrons in a plasma, where we substitute in the Lorentz force for $\mathbf{F}$:

$$\frac{\partial n}{\partial t} + \mathbf{v} \cdot \nabla n + \frac{e}{m}\Big(\mathbf{E} + (\mathbf{v} \times \mathbf{B})\Big) \cdot \nabla_{\mathbf{v}} n = \left(\frac{\partial n}{\partial t}\right)_{\text{coll}} + s \,. \tag{16.13}$$

If we neglect sources and collisions, we arrive at the Vlasov equation:

$$\frac{\partial n}{\partial t} + \mathbf{v} \cdot \nabla n + \frac{e}{m}\Big(\mathbf{E} + (\mathbf{v} \times \mathbf{B})\Big) \cdot \nabla_{\mathbf{v}} n = 0 \,. \tag{16.14}$$

Augmented with Maxwell's equations, the Vlasov equation gives a rather complete description of collisionless plasmas.

Further Reading

The treatment of boundary conditions is rather straightforward, but the student may wish to consult e.g. Duderstadt and Martin [10] or Lewis and Miller [16]. The discussion of white boundary conditions and the Wigner-Seitz dilemma follows that of Hébert [12], and the Newmarch effect is identified by Stamm'ler and Abbate [24].

The discussion of photon and plasma transport largely follows that of Duderstadt and Martin [10]. The example grey approximation equations are given in a paper by Miller and Lewis [18], and there is a wealth of literature on the subject. For those interested in radiative transfer as it applies to atmospheres, see MIT course 12.815.

Lecture 17 | Analytical Solutions

In this lecture, we analyze the neutron transport equation analytically for several simple cases. In particular, we investigate neutron streaming in a vacuum and in a purely absorbing slab. The next lecture offers further analytical and semi-analytical treatments using the integral form of the transport equation. These two lectures ultimately show the difficulty with which realistic problems can be addressed by "pen and paper" and serve to motivate our later discussions of deterministic numerical methods.

One-Speed Transport

Before we consider solutions to the transport equation, we first eliminate the energy dependence. The reason for this is simple: *the energy is simply too hard to deal with directly*. The dependence of the various cross-sections on the energy is erratic, and, as we have seen in previous lectures, there are isotopes whose dependence on energy in certain energy ranges cannot even be resolved!

We can eliminate E in two ways. First, we can assume that ψ and the cross-sections are constant in energy within an energy range $E_g < E < E_{g-1}$; this is the multi-group method, which has been the workhorse of deterministic transport methods for decades*. A second, somewhat superficial approach is to multiply the energy-dependent transport equation by $\delta(E - E_0)$. Since $f(x)\delta(x - x_0) = f(x_0)$, we have for E_0 (or a groups g) the time- and energy-independent or *one-speed transport equation*:

$$\hat{\Omega} \cdot \nabla\psi(\mathbf{r}, \mathbf{\Omega}) + \Sigma_t\psi(\mathbf{r}, \mathbf{\Omega}) = \int_{4\pi} d\Omega' \Sigma_s(\mathbf{r}, \mathbf{\Omega} \cdot \mathbf{\Omega}')\psi(\mathbf{r}, \mathbf{\Omega}') + s(\mathbf{r}, \mathbf{\Omega})\,. \quad (17.1)$$

*A new methodology being developed here at MIT is a generalization of the multigroup method where instead of flat fluxes within groups (a "zeroth" order representation), the fluxes can have higher order dependences (linear, parabolic, etc.) using discrete Legendre polynomial expansions.

Streaming in Vacuum

Perhaps the easiest class of problems to consider are those whose medium is vacuum. In this case, there are no particle interactions, and all we need to do is follow particles along trajectories from sources. These trajectories are called *characteristics*, and in general, the *method of characteristics* is the mathematical technique we can use to find ψ. Note, a more general use of the *method of characteristics* applies to arbitrary media and is fast becoming the standard transport method in lattice physics.

The Streaming Term

Consider the streaming of neutrons in a sourceless vacuum:

$$\hat{\Omega} \cdot \nabla\psi(\mathbf{r}, \mathbf{\Omega}) = 0\,. \tag{17.2}$$

What form does the streaming term $\hat{\Omega} \cdot \nabla\psi$ have? It depends crucially on the underlying coordinate systems.

It helps to note that $\hat{\mathbf{\Omega}} \cdot \nabla\psi$ is just the spatial rate of change of ψ along the direction of travel, i.e. along the characteristic. Suppose we have a particle originally at a location $\mathbf{r}_0$ going in a direction $\mathbf{\Omega}$. Then upon traveling a distance s, the location $\mathbf{r} = \mathbf{r}_0 + s\mathbf{\Omega}$. Accordingly, the spatial rate of change of ψ can be written

$$\frac{d}{ds}\psi(\mathbf{r}_0 + s\mathbf{\Omega}, \mathbf{\Omega}) = 0\,. \tag{17.3}$$

We can also show more explicitly that $\frac{d\psi}{ds} = \hat{\Omega} \cdot \nabla\psi$. In general, the direction vector $\mathbf{\Omega}$ has the basic form

$$\mathbf{\Omega} = \mu\hat{\mathbf{e}}_\mu + \eta\hat{\mathbf{e}}_\eta + \xi\hat{\mathbf{e}}_\xi\,, \tag{17.4}$$

where μ, η, and ξ are directional cosines and the $\hat{\mathbf{e}}$'s are corresponding coordinate vectors. In general, $\mathbf{\Omega}$ depends on three spatial coordinates p_1, p_2, and p_3 and two angular coordinates, usually parameterized as the cosine of the polar angle $\chi = \cos(\theta)^\dagger$ and the azimuthal angle ϕ. Hence,

$$\frac{d}{ds} = \frac{dp_1}{ds}\frac{\partial}{\partial p_1} + \frac{dp_2}{ds}\frac{\partial}{\partial p_2} + \frac{dp_3}{ds}\frac{\partial}{\partial p_3} + \frac{d\mu}{ds}\frac{\partial}{\partial \mu} + \frac{d\phi}{ds}\frac{\partial}{\partial \phi}\,. \tag{17.5}$$

The various derivatives in Eq. 17.5 may or may not vanish, depending on the geometry. Consider Cartesian geometry where $p_1 = x$ and so on. The Cartesian spatial

†We use χ to represent the polar angle cosine in general rather than μ, since μ here will be the directional cosine with respect to the x axis.

and angular system was given in Figure 15.1, where the polar angle was defined with respect to the z axis and the azimuth with respect to the x axis. Any incremental movement ds along the direction $\mathbf{\Omega}$ can be seen to change neither θ (nor its cosine ξ) nor ϕ, since the angular coordinate system is invariant as the particle moves. All this means is that $\mathbf{\Omega}$ at $\mathbf{r}_0$ is the same as the $\mathbf{\Omega}$ at $\mathbf{r}_0 + ds\mathbf{\Omega}$. Hence, $d\xi/ds = d\phi/ds = 0$ and

$$\frac{d}{ds} = \frac{dx}{ds}\frac{\partial}{\partial x} + \frac{dy}{ds}\frac{\partial}{\partial y} + \frac{dz}{ds}\frac{\partial}{\partial z}\,, \tag{17.6}$$

but dx/ds is just the directional cosine with respect to the x axis, μ, and likewise $dx/ds = \eta$ and $dz/ds = \xi$ so that

$$\frac{d}{ds} = \mu\frac{\partial}{\partial x} + \eta\frac{\partial}{\partial y} + \xi\frac{\partial}{\partial z} = \mathbf{\Omega}\cdot\nabla\,. \tag{17.7}$$

For other geometries, the streaming term is not as simple, since the angular coordinate system does depend on the position $\mathbf{r}$. The spherical spatial and angular system is given if Figure 17.1. The three spatial coordinates are r, θ_r and ϕ_r. The angular coordinate system is such that the polar angle is defined with respect to $\mathbf{r}$. The azimuth and secondary coordinates are defined somewhat arbitrarily.

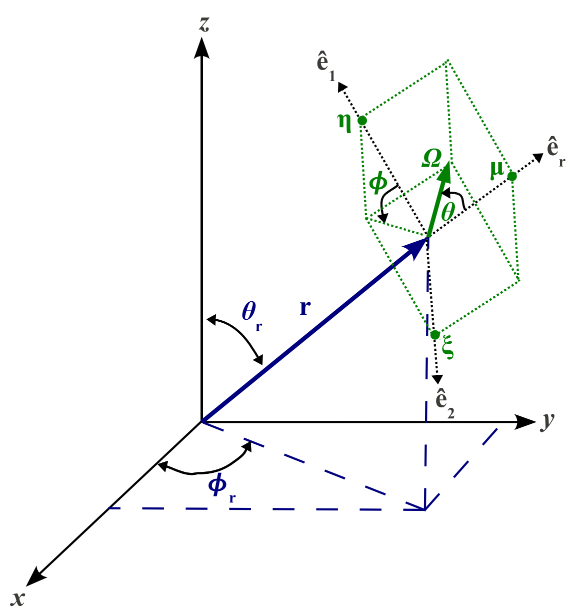

Figure 17.1: Spherical phase space.

The streaming operator in spherical coordinates is defined generally as

$$\frac{d}{ds} = \frac{dr}{ds}\frac{\partial}{\partial r} + \frac{d\theta_r}{ds}\frac{\partial}{\partial \theta_r} + \frac{d\phi_r}{ds}\frac{\partial}{\partial \phi_r} + \frac{d\mu}{ds}\frac{\partial}{\partial \mu} + \frac{d\phi}{ds}\frac{\partial}{\partial \phi}\,. \tag{17.8}$$

As a simple example, consider the case of 1-d transport in spherical coordinates, for which we assume the flux is independent of the spatial coordinates θ_r and ϕ_r, which eliminates the derivatives with respect to θ_r and ϕ_r. Moreover, if we look at the angular coordinates, we see that if r is the only spatial variable, then there should be dependence only on μ. A dependence on the azimuthal angle would require a non-uniform particle distribution in the other spatial directions. Hence, the derivative with respect to ϕ also vanishes, leaving

$$\frac{d}{ds} = \frac{dr}{ds}\frac{\partial}{\partial r} + \frac{d\mu}{ds}\frac{\partial}{\partial \mu}\,. \tag{17.9}$$

From Figure 17.1, we can immediately see that

$$\frac{dr}{ds} = \mu\,. \tag{17.10}$$

Less obvious is $d\mu/ds$. First note that $r d\theta = -ds \sin\theta$, which we can see directly from Figure 17.2. The negative sign arises because a positive ds leads to a decrease θ. Then, noting that $\mu = \cos\theta$ so that $d\mu = -\sin\theta d\theta$, we can show

$$\frac{d\mu}{ds} = \frac{1-\mu^2}{r}\,, \tag{17.11}$$

and

$$\frac{d\psi}{ds} = \mathbf{\Omega}\cdot\nabla_{\mathrm{1d}}\psi = \mu\frac{\partial\psi}{\partial r} + \frac{1-\mu^2}{r}\frac{\partial\psi}{\partial\mu}\,. \tag{17.12}$$

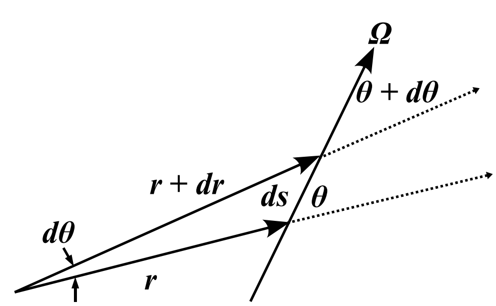

Figure 17.2: Change in r and θ as a particle streams.

Example 1: Meandering from a Plane Source in a Vacuum Slab

As a first example, consider the case of neutrons streaming in a 1-d slab for $x > 0$ where the source is an isotropic planar source at $x = 0$ which we will use as a boundary condition rather than a source term. The Cartesian streaming term above simplifies in 1-d to

$$\mu \frac{\partial \psi}{\partial x} = 0 \,, \tag{17.13}$$

subject to

$$\psi(0, \mu) = \frac{S_0}{2} \,. \tag{17.14}$$

In 1-d slab geometry, we essentially integrate out the 2π associated with the azimuthal angle. Hence, isotropic sources of strength S become $S/2$ instead of $S/4\pi$. The units of ψ are slightly different, and one must divide by 2π [rad] in order to obtain units appropriate in 3-d. Integrating the equations shows that

$$\psi(x, \mu) = c \,, \quad \mu > 0 \,, \tag{17.15}$$

for some constant c. At $x = 0$, we must have $\psi(0, \mu) = S/2$, and so for all $x > 0$, $\psi(x, \mu) = S/2$. The same holds for negative x and μ.

Example 2: Playing in Vacuum Outside a Spherical Shell Source

We now consider a neutrons streaming in a vacuum due to an isotropic spherical shell source of radius r_0. We focus only on $r > r_0$. The transport equation can be written

$$\mu \frac{\partial \psi}{\partial r} + \frac{1 - \mu^2}{r} \frac{\partial \psi}{\partial \mu} = \frac{S_0 \delta(r - r_0)}{4\pi r_0} \,. \tag{17.16}$$

From Eqs. 17.10 and 17.11, we have

$$ds = \frac{d\mu}{(1 - \mu^2)/r} = \frac{dr}{\mu} \,, \tag{17.17}$$

or

$$\frac{dr}{r} = \frac{\mu d\mu}{1 - \mu^2} \,. \tag{17.18}$$

We integrate from initial coordinates r_s and μ_s, as shown in Figure 17.3, and rearrange to obtain

$$\mu = \sqrt{1 - (1 - \mu_s^2) \left(\frac{r_s}{r} \right)^2} \,. \tag{17.19}$$

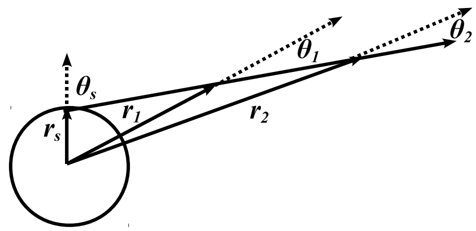

Figure 17.3: Streaming of particles from spherical shell source.

Eq. 17.19 shows explicitly how μ changes as a function of r. This phenomenon is often called angular redistribution and leads to collimation of the flux away from the source. Since $d\psi/ds$ must equal the source, we can write

$$ds = \frac{d\mu}{(1-\mu^2)/r} = \frac{d\psi}{\frac{S_0\delta(r-r_0)}{4\pi r_0}}\,, \tag{17.20}$$

or

$$\frac{d\psi}{dr} = \frac{S_0\delta(r-r_0)}{4\pi r_0}\,. \tag{17.21}$$

Then

$$\int_{r_s,\mu_s}^{r,\mu}\frac{d\psi}{dr} = \int_{r_s}^{r} dr\frac{S_0\delta(r-r_0)}{4\pi r_0}\,, \tag{17.22}$$

or

$$\psi(r,\mu) - \psi(r_s,\mu_s) = \frac{S_0}{8\pi r_s^2}\frac{1}{\sqrt{1-(1-\mu_s^2)}}\,. \tag{17.23}$$

We take the boundary flux to be $\psi(r_s,\mu_s) = 0$, which implies that μ_s is limited to $\mu_s \le 0 \le \pi/2$. If μ_s spanned through π, then particles could be born into the sphere and stream out of another location, a complication we care to avoid in this example. From Eq. 17.19, we have $(1-\mu_s^2) = (r/r_s)^2(1-\mu^2)$. We note that for $\mu_s = 0$, $\mu = \sqrt{1-(r_s/r)^2}$ and when $\mu_s = 1$, $\mu = 1$. Finally, we have

$$\psi(r,\mu) = \frac{S_0}{8\pi}\frac{1}{\sqrt{r_s^2 - r^2(1-\mu^2)}}\,, \quad \sqrt{1-\left(\frac{r_s}{r}\right)^2} \le \mu \le 1\,. \tag{17.24}$$

To illustrate the behavior of ψ, we have taken $r_s = 1$ and $S_0 = 8\pi$. Figure 17.4 shows ψ as a function of radius for several μ values. Of course, we see that as neutrons move farther from the source, the flux at larger θ values (smaller μ values) diminishes, as we expect due to angular redistribution. Figure 17.5 shows ψ as a function of μ for several values of r. We see effects of the same phenomenon, in that the angular distribution becomes more collimated about $\mu = 1$ for larger r.

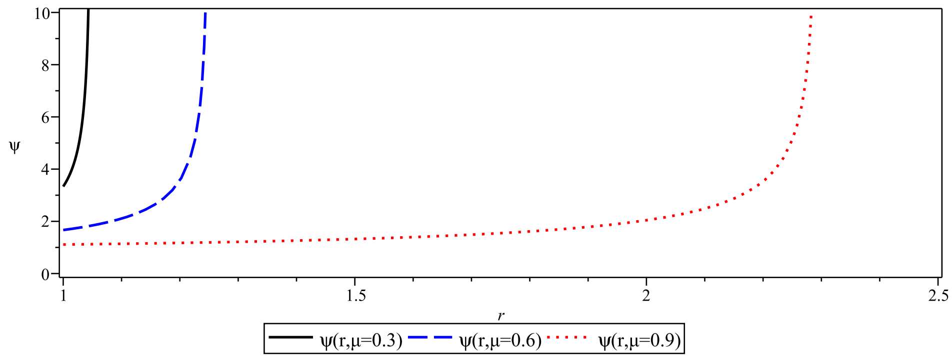

Figure 17.4: Angular flux as a function of radius for certain μ values.

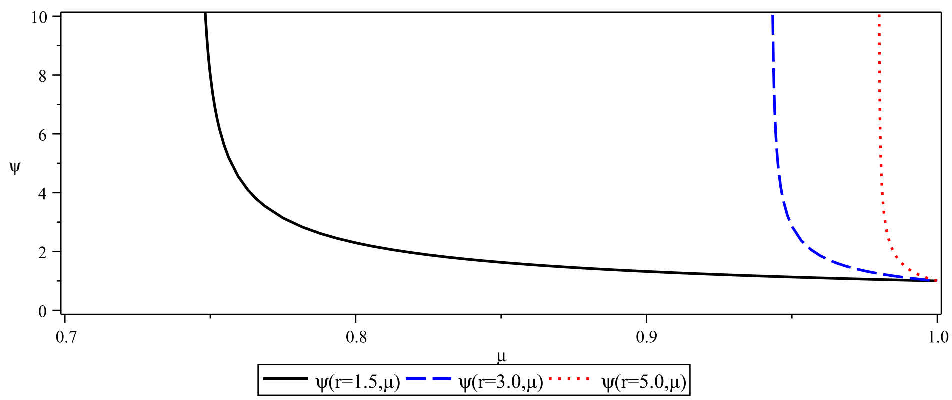

Figure 17.5: Angular flux as a function of μ at certain r values.

Example 3: A Purely Absorbing Slab

As a final example, we apply what we've learned in vacuum to transport in a purely absorbing slab. In this case, the 1-d transport equation is

$$\mu \frac{\partial \psi}{\partial x} + \Sigma_a(x)\psi(x,\mu) = S(x,\mu)\,. \tag{17.25}$$

As an example, we study a uniform slab of length L, subject to vacuum boundaries, and with a uniform, isotropic source of volumetric strength[‡] S_0. Our equation sim-

[‡]By volumetric strength, we mean the units are neutrons per unit volume per second.

plifies to

$$\mu \frac{\partial \psi}{\partial x} + \Sigma_a \psi(x, \mu) = \frac{S_0}{2} \,. \tag{17.26}$$

To solve the problem, we decompose ψ into ψ_+ for $\mu > 0$ and ψ_- for $\mu < 0$. In this way, we can start at one end of the slab and work our way across, essentially as would a neutron.

For $\mu > 0$, we divide Eq. 17.26 through by μ and compute the integrating factor

$$if = \exp \int_0^x dx \Sigma_a \mu = \exp \Sigma_a x/\mu \,. \tag{17.27}$$

Then we have

$$\frac{d}{dx}\left(\psi_+ e^{\Sigma_a x}\right) = \frac{S_0}{2\mu} e^{\Sigma_a x/\mu} \,, \tag{17.28}$$

and integrating from 0 to x yields

$$\psi_+ = \frac{S_0}{2\Sigma_a}\left(1 - e^{-\Sigma_a x/\mu}\right) , \tag{17.29}$$

where we note $\psi_+ 0, \mu = 0$ from the given boundary condition.

For $\mu < 0$, we do similarly. It helps in this case to use $-|\mu|$ in place of μ, as it can be easy to lose negative signs. Using the integrating factor $\exp{(L-x)/|\mu|}$, we find after integrating from L to x that

$$\psi_- = \frac{S_0}{2\Sigma_a}\left(1 - e^{-\Sigma_a (L-x)/|\mu|}\right) . \tag{17.30}$$

For the case of $L = 10$ and $\Sigma_a = 1$, Figure 17.6 shows ψ for several values of ψ. Note the symmetry, as should be expected.

Our next goal is to compute the scalar flux. By definition, the scalar flux in 1-d is

$$\phi(x) = \int_{-1}^{1} d\mu \psi(x, \mu) \,, \tag{17.31}$$

which can be broken into

$$\phi(x) = \int_{-1}^{0} d\mu \psi_-(x, \mu) + \int_{0}^{1} d\mu \psi_+(x, \mu) \,. \tag{17.32}$$

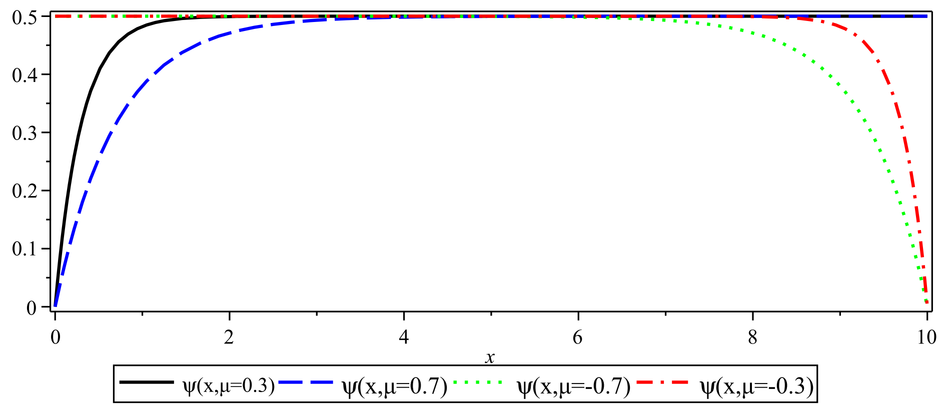

Figure 17.6: Angular flux at specific angles for the absorbing slab.

Inserting our expressions above, we have

$$\begin{aligned}\phi(x) &= \frac{S_0}{2\Sigma_a}\left(\int_{-1}^{1} d\mu - \int_{-1}^{0} d\mu e^{-\Sigma_a(L-x)/|\mu|} - \int_0^1 d\mu e^{-\Sigma_a x/\mu}\right) \\ &= \frac{S_0}{2\Sigma_a}\left(2 - \int_0^1 d\mu' e^{-\Sigma_a(L-x)/\mu} - \int_0^1 d\mu e^{-\Sigma_a x/\mu}\right) \\ &= \frac{S_0}{2\Sigma_a}\Big(2 - E_2(\Sigma_a(L-x)) - E_2(\Sigma_a x)\Big)\,.\end{aligned} \tag{17.33}$$

For the same numerical example, $\phi(x)$ is shown in Figure 17.7 along with the current density $\mathbf{J}(x)$, computation of which is left as an exercise.

Exponential Integrals

The functions $E_n(x)$ are called "exponential integrals" and are characteristic of slab problems. They are defined by

$$E_n(x) \equiv \int_0^1 d\mu \mu^{n-2} e^{-x/\mu}\,. \tag{17.34}$$

They also satisfy

$$E_n(x) = -\int dx E_{n-1}(x)\,. \tag{17.35}$$

Several of the E_n functions are shown in Figure 17.8.

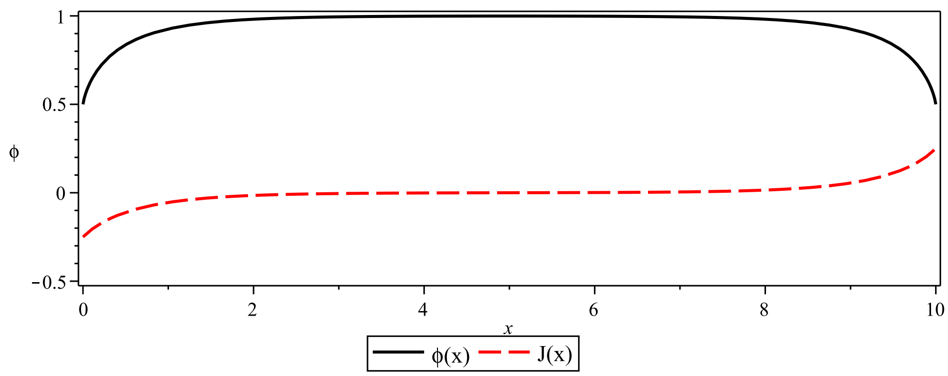

Figure 17.7: Scalar flux and current density.

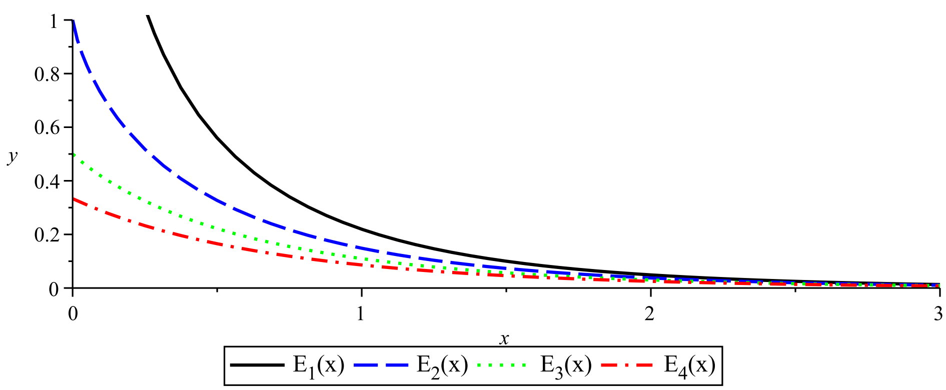

Figure 17.8: The first four exponential integral functions.

Further Reading

The problems discussed in this lecture have been rather elementary in nature. To explore more realistic problems—even just including scattering in slab geometry—would take us into a much more complicated mathematical domain involving integral transforms (with inverses using complex analysis), singular eigenfunction expansions, and more. Duderstadt and Martin [10] cover most of what has been discussed, and much more. The motivated student should look to that work (specifically Chapter 2), along with Bell and Glasstone (Chapter 2), Case and Zweifel [5], and Davison [8]. These are listed in reverse chronological order, and perhaps, in increasing order of difficulty. Davison in particular uses older notation, so the reader be aware!

Exercises

1. **Cylindrical Coordinates**. Derive the streaming term for the neutron transport equation in 1-d cylindrical geometry. Include diagrams to help explain your approach.
2. **Angular Current from Plane Source**. In Example 1, what is the angular current **j**?
3. **Isotropic Current Conditions**. For Example 1, find ψ and **j** when the boundary condition is an isotropic angular current (rather than an isotropic angular flux).
4. **Flux Inside a Sphere**. Solve for the angular flux on the interior of the spherical shell from Example 2.
5. **Angular Redistribution in 1-d**. Explain in your own words why ψ in a 1-d spherical problem depends only on the radius r and cosine of the polar angle μ. Specifically, why is there no dependence on a second angular variable?
6. **Scalar Flux from Spherical Shell Source**. Solve for the scalar flux on the outside of the sphere in Example 2, i.e. integrate Eq. 17.24 over μ. What happens at $r = r_s$ and why? What is the limiting behavior of $\phi(r)$ as $r \to \infty$? In other words, what does the spherical source "look like" far from its surface?
7. **Current Density in a Slab**. Derive an expression for the current density **J** for a purely absorbing slab. Using the values from the example, generate the curves in Figure 17.7. Compute the total absorption rate in the slab and the leakage rate at the boundaries. Is it what you expect?

Lecture 18 | The Integral Form

In this lecture we consider the integral form of the transport equation in both general coordinates and the specific case of slab geometry. The integral form is useful in situations where only the scalar flux is required and is the foundation for the collision probability method, which we cover in the next lecture, as well as the method of characteristics, fast becoming the technique of choice in reactor analysis. We consider some analytical aspects of the integral equation and finish by discussing a simple numerical approach based on Neumann series expansions.

The Integral Transport Equation

You may have noticed from the previous two lectures that the angular dependence of the particle density is a relatively unique aspect of transport processes. Often, this angular dependence is the hardest aspect that we deal with directly, either analytically or numerically* It would be nice to eliminate the angular variable completely, and for certain problems, the *integral transport equation* allows us to do with a minimum of approximation.

Before we derive the integral form of the transport equation, it helps define a new quantity called the *emission density*,

$$Q(\mathbf{r}, \mathbf{\Omega}) = \int_{4\pi} d\Omega' \Sigma_s(\mathbf{r}, \mathbf{\Omega} \cdot \mathbf{\Omega}') \psi(\mathbf{r}, \mathbf{\Omega}') + S(\mathbf{r}, \mathbf{\Omega}) \,. \tag{18.1}$$

Essentially, Q is a generalized source containing any external source S and scattering source; of course, fission could also be included.

Recall the discussion of Eq. 17.3, where we consider a distance s from some reference point $\mathbf{r}_0$ along the characteristic $\mathbf{r}_0 + s\mathbf{\Omega}$. The full transport equation in

*Energy is by far more complex, as we've noted, and there are very few analytical problems where energy can be handled directly (slowing down in an infinite homogeneous medium problem is one example). Consequently, we typically use only the crudest representation possible in discretizing the energy variable, though much physics goes into generating the discrete data.

terms of s is

$$\frac{d}{ds}\psi(\mathbf{r}_0 + s\mathbf{\Omega}, \mathbf{\Omega}) + \Sigma_t(\mathbf{r}_0 + s\mathbf{\Omega})\psi = Q(\mathbf{r}_0 + s\mathbf{\Omega}, \mathbf{\Omega})\,, \tag{18.2}$$

which can be called the *forward characteristic* equation, as we follow neutrons forward from the reference point. Similar, and which we use below, is the *backward characteristic* form, which represents following neutrons backward along the characteristic from a current point $\mathbf{r}$. Let $p = -s$. Then $d/dp = -d/ds$ and Eq. 18.2 becomes (after dropping the 0 subscript from $\mathbf{r}$)

$$-\frac{d}{dp}\psi(\mathbf{r} - p\mathbf{\Omega}, \mathbf{\Omega}) + \Sigma_t(\mathbf{r} - p\mathbf{\Omega})\psi = Q(\mathbf{r} - p\mathbf{\Omega}, \mathbf{\Omega})\,. \tag{18.3}$$

We wish to integrate from $p = 0$ to some maximum distance (eventually to be either infinity or a global boundary). Introducing the integrating factor

$$if = e^{-\int_0^p \Sigma_t(\mathbf{r}-p'\mathbf{\Omega})dp'} \tag{18.4}$$

into Eq. 18.3, we have

$$-\frac{d}{dp}\Big(\psi(\mathbf{r}_0 - p\mathbf{\Omega}, \mathbf{\Omega})e^{-\int_0^p \Sigma_t(\mathbf{r}-p'\mathbf{\Omega})dp'}\Big) = Q(\mathbf{r} - p\mathbf{\Omega})e^{-\int_0^p \Sigma_t(\mathbf{r}-p'\mathbf{\Omega})dp'}\,, \tag{18.5}$$

and integrating

$$\begin{aligned} -\int_{\psi(p=0)}^{\psi(p'=p)} \left(d\Big(\psi(\mathbf{r} - p'\mathbf{\Omega}, \mathbf{\Omega})e^{-\int_0^{p'} \Sigma_t(\mathbf{r}-p''\mathbf{\Omega})dp''}\Big)\right) = \\ \int_0^p Q(\mathbf{r} - p'\mathbf{\Omega}, \mathbf{\Omega})e^{-\int_0^{p'} \Sigma_t(\mathbf{r}-p''\mathbf{\Omega})dp''}dp' \end{aligned} \tag{18.6}$$

yields

$$\begin{aligned} \psi(\mathbf{r}, \mathbf{\Omega}) - \psi(\mathbf{r} - p\mathbf{\Omega}, \mathbf{\Omega})e^{-\int_0^p \Sigma_t(\mathbf{r}-p''\mathbf{\Omega})dp''} = \\ \int_0^p Q(\mathbf{r} - p'\mathbf{\Omega}, \mathbf{\Omega})e^{-\int_0^{p'} \Sigma_t(\mathbf{r}-p''\mathbf{\Omega})dp''}dp'\,. \end{aligned} \tag{18.7}$$

We can simplify the notation somewhat by defining the *optical pathlength* τ, such that

$$\tau(\mathbf{r}, \mathbf{r} - p\mathbf{\Omega}) = \int_0^p \Sigma_t(\mathbf{r} - p'\mathbf{\Omega})dp'\,. \tag{18.8}$$

Then, the integral equation for the angular flux becomes

$$\psi(\mathbf{r}, \mathbf{\Omega}) = \int_0^p Q(\mathbf{r} - p'\mathbf{\Omega}, \mathbf{\Omega})e^{-\tau(\mathbf{r},\mathbf{r}-p'\mathbf{\Omega})}dp' + \psi(\mathbf{r} - p\mathbf{\Omega}, \mathbf{\Omega})e^{-\tau(\mathbf{r},\mathbf{r}-p\mathbf{\Omega})}\,. \tag{18.9}$$

In many cases, the emission density Q is assumed to be isotropic in the lab system. In this case, we can integrate out the angular dependence and arrive at an integral equation for the scalar flux often referred to as Peierl's equation. To do so, we let the integration bound p of Eq. 18.9 go to infinity. We assume the second term on the right hand side to vanish (i.e. either ψ vanishes at infinity, or equivalently, the optical path length $\tau(\mathbf{r}, \infty)$ goes to ∞, and so the exponential vanishes).

Letting $Q(\mathbf{r}, \mathbf{\Omega}) = Q(\mathbf{r} - p\mathbf{\Omega})/4\pi$, and substituting $\mathbf{r}' = \mathbf{r} - p\mathbf{\Omega}$, we integrate Eq. 18.9 over all angles to get

$$\phi(\mathbf{r}) = \int_{4\pi} d\Omega \int_0^\infty \frac{Q(\mathbf{r}')}{4\pi} e^{-\tau(\mathbf{r},\mathbf{r}')} dp\,. \tag{18.10}$$

Now note that $p = |\mathbf{\Omega} p| = |\mathbf{r} - \mathbf{r}'|$, and consequently $p^2 = |\mathbf{r} - \mathbf{r}'|^2$. Multiplying within the integrand by $1 = p^2/|\mathbf{r} - \mathbf{r}'|^2$ yields

$$\phi(\mathbf{r}) = \int_{4\pi} d\Omega \int_0^\infty \frac{Q(\mathbf{r}')}{4\pi} e^{-\tau(\mathbf{r},\mathbf{r}')} \frac{p^2 dp}{|\mathbf{r} - \mathbf{r}'|^2}\,, \tag{18.11}$$

and if the reader thinks this looks suspiciously like a volume integral in spherical coordinates, she would be right. Letting $dV' = 4\pi d\Omega dp p^2$, we have

$$\phi(\mathbf{r}) = \int_{V'} \frac{Q(\mathbf{r}') e^{-\tau(\mathbf{r},\mathbf{r}')} dV'}{4\pi |\mathbf{r} - \mathbf{r}'|^2}\,. \tag{18.12}$$

Integral Transport in Slab Geometry

We now look at the special case of Eq. 18.12 in slab geometry, for which the emission density is a function only of x, i.e. $Q(\mathbf{r}) = Q(x)$. We make use of cylindrical coordinates, with the axis taken to be x. Our goal will be to integrate out the radial (ρ) and azimuthal (ω) spatial components, leaving just the x dependence. The differential volume is then

$$dV' = \rho d\rho d\omega dx', \tag{18.13}$$

and Eq. 18.12 becomes

$$\begin{aligned}\phi(x) &= \int_{-\infty}^{\infty} dx' \int_0^\infty d\rho\rho \int_0^{2\pi} \frac{d\omega' Q(x') e^{-\tau(\mathbf{r},\mathbf{r}')}}{4\pi |\mathbf{r} - \mathbf{r}'|^2} \\ &= 2\pi \int_{-\infty}^{\infty} dx' \int_0^\infty d\rho\rho \frac{Q(x') e^{-\tau(\mathbf{r},\mathbf{r}')}}{4\pi |\mathbf{r} - \mathbf{r}'|^2}\,.\end{aligned} \tag{18.14}$$

We now need to express $\mathbf{r}$ and ρ in terms of x. Since the cross-sections (as quantified by τ) are really dependent only on x, we can relate the full distance $|\mathbf{r}' - \mathbf{r}|$

with its projection along the x axis via a directional cosine λ^{-1} such that

$$\lambda = \frac{|\mathbf{r}' - \mathbf{r}|}{|x' - x|} \tag{18.15}$$

and $\tau(\mathbf{r}', \mathbf{r}) = \lambda\tau(x', x)$. Moreover,

$$|\mathbf{r}' - \mathbf{r}|^2 = \rho^2 + |x' - x|^2 \tag{18.16}$$

which, using Eq. 18.15, can be rewritten as

$$\rho^2 = (\lambda^2 - 1)|x' - x|^2\,, \tag{18.17}$$

and differentiating, we find

$$\rho d\rho = \lambda d\lambda |x' - x|^2\,. \tag{18.18}$$

Noting that $\rho = 0$ corresponds to $\lambda = 1$, Eq. 18.14 can be written in terms of λ to give

$$\begin{aligned}
\phi(x) &= 2\pi \int_{-\infty}^{\infty} dx' \int_1^{\infty} \lambda d\lambda \frac{|x' - x|^2}{4\pi|\mathbf{r}' - \mathbf{r}|^2} Q(x') e^{-\tau(x,x')} \\
&= 2\pi \int_{-\infty}^{\infty} dx' \int_1^{\infty} \lambda d\lambda \frac{1}{4\pi\lambda^2} Q(x') e^{-\tau(x,x')} \\
&= \int_{-\infty}^{\infty} dx' \frac{1}{2} \int_1^{\infty} \lambda d\lambda \frac{1}{\lambda} Q(x') e^{-\tau(x,x')}
\end{aligned} \tag{18.19}$$

or

$$\phi(x) = \int_{-\infty}^{\infty} dx' \frac{1}{2} E_1(\tau(x, x')) Q(x')\,, \tag{18.20}$$

where we have used the E_1 function defined at the end of Lecture 17.

First-Flight Kernels

Eq. 18.20 gives us an example of the use of a *first-flight kernel* for the scalar flux, the general use of which takes the form

$$\phi(\mathbf{r}) = \int d^3\mathbf{r}' k(\mathbf{r}, \mathbf{r}') Q(\mathbf{r}') \tag{18.21}$$

for a kernel $k(\mathbf{r}, \mathbf{r}')$. For slab geometry, the first-flight kernel is seen to be

$$k_{\text{slab}}(x, x') = \frac{1}{2} E_1(\tau(x, x'))\,. \tag{18.22}$$

First-flight kernels have a particularly easy (and important!) physical interpretation. Consider Eq. 18.20 for the case of a purely absorbing medium. Then the emissivity Q consists only of external sources. To help visualize the problem, take Q to be a delta function at x_0, i.e. $Q(x) = Q_0\delta(x - x_0)$. Substituting this into Eq. 18.20 gives

$$\phi(x) = \frac{1}{2}E_1(\tau(x, x_0))Q_0 \,. \tag{18.23}$$

Thus, the kernel $k(x, x')$ can be seen to give the contribution of the source particles born at x' to the flux at x. In other words, it gives to us the *uncollided flux*. For many systems, having the uncollided flux can be a good approximation for the total flux, and in some numerical schemes, it can be a good initial guess to help reduce computational time and numerical artifacts (e.g. the discrete ordinates method, discussed in Lecture 20).

If we look back at the Peierl's equation (Eq. 18.12), we find the fundamental first-flight kernel of the point source,

$$k_{\text{point}}(\mathbf{r}, \mathbf{r}') = \frac{e^{-\tau(\mathbf{r},\mathbf{r}')}}{4\pi|\mathbf{r} - \mathbf{r}'|^2} \,. \tag{18.24}$$

Two things are worth noting about Eq. 18.24. First, the first-flight kernels for all other geometrical configurations can be derived from this kernel. A second point, related to the first, is that the point kernel is closely related to the *Green's function* for the transport equation.

Green's Functions

A Green's function $G(x, x')$ for a linear differential operator† $L = L(x)$ is defined

$$LG(x, x') = \delta(x - x') \,. \tag{18.25}$$

A linear differential operator is any linear combination of basic differentiation operators. L could be d/dx or d^2/dx^2 or $d^2/dx^2 + d/dx$, and so on. The utility of G arises when we wish to solve the inhomogeneous differential equation

$$Lu(x) = f(x) \,. \tag{18.26}$$

If we multiply both sides of Eq. 18.25 by $f(x')$ and integrate over x', we find

$$\int LG(x, x')f(x')dx' = \int dx'\delta(x - x')f(x') = f(x) \,, \tag{18.27}$$

†We'll discuss the linearity of the transport equation in Lecture 22, and we'll use operator notation extensively in Lecture 23.

but this suggests that

$$Lu(x) = \int LG(x,x')f(x')dx' = L\int G(x,x')f(x')dx'\,, \tag{18.28}$$

or

$$u(x) = \int G(x,x')f(x')dx'\,. \tag{18.29}$$

Hence, if we know $G(x,x')$, then we can solve the inhomogeneous equation for u.

What about the transport equation? Consider again Eq. 18.9, neglecting the second term, and letting $p \to \infty$, i.e.

$$\psi(\mathbf{r},\mathbf{\Omega}) = \int_0^\infty Q(\mathbf{r}-p\mathbf{\Omega},\mathbf{\Omega})e^{-\tau(\mathbf{r},\mathbf{r}-p\mathbf{\Omega})}dp\,. \tag{18.30}$$

Note that this is still integrating along the characteristic. It is more convenient to cast this as volume integral, similar to what we did above for Eq. 18.12. However, even in volume form, we still want the integration confined to the characteristic. By defining

$$\delta(\mathbf{\Omega}\cdot\mathbf{\Omega}') \equiv \delta(\mu-\mu')\delta(\phi-\phi')\,, \tag{18.31}$$

and

$$\mathbf{\Omega}_R \equiv \frac{\mathbf{r}-\mathbf{r}'}{|\mathbf{r}-\mathbf{r}'|}\,, \tag{18.32}$$

and recalling $p = |\mathbf{r}-\mathbf{r}'|$, we can rewrite Eq. 18.30 as

$$\psi(\mathbf{r},\mathbf{\Omega}) = \int_{4\pi} d\Omega_R\delta(\mathbf{\Omega}\cdot\mathbf{\Omega}_R)\int_0^\infty Q(\mathbf{r}',\mathbf{\Omega}_R)e^{-\tau(\mathbf{r},\mathbf{r}')}dp\,. \tag{18.33}$$

Using $dV' = p^2dpd\Omega_R$, this becomes

$$\psi(\mathbf{r},\mathbf{\Omega}) = \int_{V'} dV'\frac{1}{|\mathbf{r}-\mathbf{r}'|^2}\delta(\mathbf{\Omega}\cdot\mathbf{\Omega}_R)Q(\mathbf{r}',\mathbf{\Omega}_R)e^{-\tau(\mathbf{r},\mathbf{r}')}\,. \tag{18.34}$$

Now let Q be a delta source at $\mathbf{r}_0$ emitting particles in direction $\mathbf{\Omega}_0$, or $Q = \delta(\mathbf{r}-\mathbf{r}_0)\delta(\Omega\cdot\Omega_0)$, similar to the right hand side of Eq. 18.25. Then

$$\begin{aligned}\psi(\mathbf{r},\mathbf{\Omega}) &= \int_{V'} dV'\frac{e^{-\tau(\mathbf{r},\mathbf{r}')}}{|\mathbf{r}-\mathbf{r}'|^2}\delta(\mathbf{\Omega}\cdot\mathbf{\Omega}_R)\delta(\mathbf{r}-\mathbf{r}_0)\delta(\mathbf{\Omega}\cdot\mathbf{\Omega}_0)\\ &= \frac{e^{-\tau(\mathbf{r},\mathbf{r}_0)}}{|\mathbf{r}-\mathbf{r}_0|^2}\delta\left(\mathbf{\Omega}\cdot\frac{\mathbf{r}-\mathbf{r}_0}{|\mathbf{r}-\mathbf{r}_0|}\right)\delta(\mathbf{\Omega}\cdot\mathbf{\Omega}_0)\\ &= G_{\text{point}}(\mathbf{r},\mathbf{\Omega};\mathbf{r}_0,\mathbf{\Omega}_0)\,.\end{aligned} \tag{18.35}$$

This is the Green's function for the angular flux, with which we can find ψ for any emission density $Q(\mathbf{r}, \mathbf{\Omega})^{\ddagger}$. We can use G_{point} to recover the scalar flux point kernel by letting Q be a unit isotropic point source, i.e. $Q = \delta(\mathbf{r} - \mathbf{r}_0)/4\pi$. Then

$$\begin{aligned}\psi(\mathbf{r}, \mathbf{\Omega}) &= \int_V dV' \int_{4\pi} d\Omega' G_{\text{point}}(\mathbf{r}, \mathbf{\Omega}; \mathbf{r}', \mathbf{\Omega}') \frac{\delta(\mathbf{r}' - \mathbf{r}_0)}{4\pi} \\ &= \frac{e^{-\tau(\mathbf{r}, \mathbf{r}_0)}}{4\pi |\mathbf{r} - \mathbf{r}_0|^2} \delta\left(\mathbf{\Omega} \cdot \frac{\mathbf{r} - \mathbf{r}_0}{|\mathbf{r} - \mathbf{r}_0|} \right).\end{aligned} \tag{18.36}$$

Integrating ψ over all angles yields

$$\begin{aligned}\phi(\mathbf{r}) &= \int_{4\pi} d\Omega \frac{e^{-\tau(\mathbf{r}, \mathbf{r}_0)}}{4\pi |\mathbf{r} - \mathbf{r}_0|^2} \delta\left(\mathbf{\Omega} \cdot \frac{\mathbf{r} - \mathbf{r}_0}{|\mathbf{r} - \mathbf{r}_0|} \right) \\ &= \frac{e^{-\tau(\mathbf{r}, \mathbf{r}_0)}}{4\pi |\mathbf{r} - \mathbf{r}_0|^2} ,\end{aligned} \tag{18.37}$$

which is indeed our point kernel for the scalar flux.

Neumann Series

Lecture 19 will cover the widely-used collision probability method that is based on the integral transport equation. Here, we investigate a less versatile yet quite enlightening method based on expansion of the scalar flux in a so-called Neumann[§] series for slab problems that include scattering.

Consider the integral equation in a slab of length L subject to vacuum boundary conditions:

$$\phi(x) = \int_0^L dx' \frac{1}{2} E_1(\tau(x, x'))(S(x') + \phi(x')\Sigma_s(x')) . \tag{18.38}$$

The integration bounds are 0 and L, since all sources must vanish in the outside vacuum. Defining the operator K,

$$K\phi = \int_0^L dx' \frac{1}{2} E_1(\tau(x, x'))\phi(x)\Sigma_s(x) , \tag{18.39}$$

we can rewrite the integral equation as

$$(I - K)\phi = K \frac{S}{\Sigma_s} . \tag{18.40}$$

[‡]Careful! If Q includes scattering, then it depends on ψ, and so a direct solution in this form is not possible

[§]Carl Neumann, not to be confused with John von Neumann.

Without going into formal detail (though it should seem "reasonable"), Eq. 18.40 can be solved by expanding

$$(I - K)^{-1} = I + K + K^2 + \dots\,, \tag{18.41}$$

where I is the identity operator, so that

$$\phi(x) = \sum_{n=0}^{\infty} K^n \frac{S(x)}{\Sigma_s(x)}\,. \tag{18.42}$$

Suppose we define

$$\phi_n(x) \equiv K^n \frac{S(x)}{\Sigma_s(x)}\,. \tag{18.43}$$

Then we see that $\phi_0 = K(S/\Sigma_s)$, $\phi_1 = K^2(S/\Sigma_s) = K(\phi_0)$ and so on. We recognize $\phi_0(x)$ as the uncollided flux, which is then used as input for ϕ_1. We recognize ϕ_1 as those neutrons already having undergone a single collision (and no more). In general, we call ϕ_n the nth collided flux. The Neumann series Eq. 18.42 just adds up all neutrons that have not collided, those that have collided once, those that have twice, and so on, thus capturing the entire population of neutrons in the system. Defining (and computing) ϕ_n in terms of the previous term ϕ_{n-1} sequence is called *Neumann iteration*.

A Numerical Example

We illustrate the use of Neumann iteration in a simple MATLAB code for a homogeneous slab problem with a uniform isotropic source. The integrals involved are approximated using the trapezoid rule, though the exercises explore other schemes. Please refer to the code comments in Listing 18.1 to understand the exact implementation of the algorithm.

One aspect of the algorithm that should be pointed out is use of the method of *subtraction of singularity*. In the numerical integration, we select some set of points x_i at which we compute the scalar flux. In a straight forward integration, this would require several evaluations of the term $E_1(\Sigma|x_i - x_i|)$, which blows up. As an alternative, one can write

$$\begin{aligned} \phi(x) &= \int_0^L k_{\text{slab}}(x, x')Q(x')dx' \\ &= \int_0^L k_{\text{slab}}(x, x')(Q(x') - Q(x))dx' + \int_0^L k_{\text{slab}}(x, x')Q(x)dx'\,. \end{aligned} \tag{18.44}$$

The second term of the final line can be integrated analytically, and you should do this to obtain the form used in the code. The first term vanishes identically for $x = x'$, which is why the code has the logical evaluation ***if****(i∼=0)* that if unsatisfied leaves $\phi_i = 0$.

Listing 18.1: Solution of Slab Problem via Neumann Series

```
function [phi,phiL] = neumann_slab(L,SigT,SigS,Q,N,M)
% function phi = neumann_slab(L,SigT,SigS,Q,N,M)
%   This function solves the one-speed transport equation in slab geometry
%   for the scalar flux using a Neumann series solution.  The integrals
%   are computed using the trapezoid rule.
%
%   The problem is a slab of width L, with uniform total cross-section
%   SigT, uniform (isotropic) scattering cross-section SigS, and
%   uniform isotropic source of volumetric strength Q.
%
%   Inputs:
%       L       -- slab width [cm]
%       SigT    -- total cross-section [1/cm]
%       SigS    -- scattering cross-section [1/cm]
%       Q       -- uniform isotropic source [1/cm^3-s]
%       N       -- number of points for evaluating flux
%       M       -- number of Neumann terms to compute
%   Output
%       phi     -- scalar flux [N,1]
%       phiL    -- Neumann terms [N,M+1] (M=0 yields uncollided only)
phi  = zeros(N,1);      % scalar flux
phiL = zeros(N,M+1);    % the Lth collided fluxes
QL   = Q*ones(N,1);     % source vector
% uncollided flux
phiL(:,1) = ithcollided(L,N,QL,SigT);
% collided fluxes
for i = 1:M
   QL          = SigS*phiL(:,i);              % compute new source
   phiL(:,i+1) = ithcollided(L,N,QL,SigT);    % get ith collided flux term
end
% scalar flux is just the sum
for i = 1:M+1
    phi = phi + phiL(:,i);
end
end % function neumann_slab

function phi = ithcollided(L,N,Q,SigT)
% function phi = ithcollided(L,N,Q,SigT)
%   This function computes a Neumann term for the given source.  The
%   exponential integral is computed using Matlab's symbolic function Ei,
%   which is pretty slow!  The method of subtraction of singularities has
%   been used.  The flux is computed as follows:
%     phi(x)   = int( k(x,x')*Q(x')dx', x=0,L )
%     phi(x_i) = h * ( 0.5*k(x_i,x_0)*Q(x_0) + 1.0*k(x_i,x_1) + ... )
%
%   Inputs:
%       L       -- slab width [cm]
%       N       -- number of points for evaluating flux
%       Q       -- source
%       SigT    -- total cross-section [1/cm]
%   Output
%       phi     -- the current Neumann term
phi = zeros(N,1);
h = L / (N-1);
% loop through all N
for i = 1:N
    xi = h*(i-1);
    phi(i) = 0;
```

```
    for j = 1:N % begin trapezoid integration
        if (i~=j) % if i=j, Q(i)=Q(j) and the term vanishes; avoids NaN!
            if (j==0 || j==N) % end points get coefficient of 0.5
                A = 0.5;
            else
                A = 1.0;
            end
            xj      = h*(j-1);
            phi(i)  = phi(i) + ...
                      A*h*mfun('Ei',1,(SigT*abs(xi-xj)))*(Q(j)-Q(i));
        end
    end
    % extra part from subtraction of the singularity
    phi(i) = phi(i) + 0.5*Q(i)/SigT * ...
             (2-mfun('Ei',2,SigT*(L-xi))-mfun('Ei',2,SigT*(xi)));
end
end % function ithcollided
```

Further Reading

The derivation of the integral equations generally follows that of Lewis and Miller [16]. First-flight kernels and Green's functions are covered in Duderstadt and Martin [10] and Case and Zweifel [5]. Solving for the scalar flux via Neumann iterations is discussed in Duderstadt and Martin [10], while its use as a general technique for integral equations is described e.g. by Arfken and Weber [2]. Implementation of the code and use of the subtraction of singularities method was inspired by notes of Prof. D. Henderson at Wisconsin. Any good numerical analysis textbook will provide more information on numerical integration, and the reader is encouraged to explore this fundamental topic.

Exercises

1. **Point-to-slab**. Show how k_{slab} can be generated using k_{point}.
2. **Line source**. Show that the first-flight kernel for an infinite isotropic line source is given by

$$k_{\text{line}}(r, r') = \frac{1}{2\pi r} Ki_1(\Sigma_t |r - r'|)\,, \tag{18.45}$$

where

$$Ki_n(x) = \int_0^{\pi/2} \cos^{n-1}(\theta) e^{-x/\cos(\theta)} d\theta = \int_0^{\infty} \frac{e^x \cosh(u)}{\cosh^n(u)} \tag{18.46}$$

are the Bickley-Naylor function of order n, and where $Ki_0(r) = K_0(r)$, the zeroth order modified Bessel function.

3. **Spherical shell**. Show that the first-flight kernel for a spherical shell is given by
$$k_{\text{sph}}(r, r') = \frac{r'}{2r} Ki_1(\Sigma_t |r - r'| - \Sigma_t |r + r'|) , \tag{18.47}$$
4. **Current density kernels**. In the lecture found the flux kernel for a plane source, and the kernels for line and spherical sources are given in the first two exercises. We can also derive kernels for the current density. For example, in a slab, there is a current kernel $\gamma_{\text{slab}}(x, x')$ such that
$$\mathbf{J}(x) = \int_0^L \gamma_{\text{slab}}(x, x') Q(x') dx' . \tag{18.48}$$
Derive this kernel $\gamma_{\text{slab}}(x, x')$.
5. **Using kernels**. Use k_{slab} to find the scalar flux for the example of Lecture 17, and use γ_{slab} to find the corresponding current density.
6. **Neumann Series**. Use the Neumann series code for the slab in Listing 18.1 to plot the scalar flux, uncollided flux, and the first five collided fluxes on the same graph using 20 spatial divisions for $L = 10$, $\Sigma_t = 1.0$, and (a) $\Sigma_s = 0.2$ and (b) $\Sigma_s = 0.8$. What changes for the case with higher scattering?
7. **Convergence**. For the same problem, modify the Neumann series code to compute as many collided fluxes are necessary so that
$$\frac{|\phi_n(x_i) - \phi_n(x_{i-1})|}{\phi_n(x_{i-1}} < \epsilon = 1 \times 10^{-6} \quad \forall\, x_i \, ,$$
and comment on the results. (Hint, which norm on the x vector is appropriate?)
8. **Nonuniform medium**. Could the Neumann series code be easily modified to handle a nonuniform slab? Suggest an approach for this.
9. **Numerical integration**. Modify the Neumann series code to handle both Simpson's rule and Gaussian quadrature. Overviews of both can be found in numerical analysis texts. For the purely absorbing slab example of Lecture 17, compare the trapezoid, Simpson's, and Gaussian schemes to the analytical solution. Comment on which method appears to get it "right" with the least effort.
10. **Efficiency**. The Neumann code is slow due to the symbolic computation of the E_n functions. Modify the code to precomputing the various coefficients (indexed by i and j), and comment on any improvement. For even greater efficiency, find a way to evaluate the exponential functions numerically (via some form of approximation) that allows you compute $E_n(x)$ rapidly; see for example Hebert's *Applied Reactor Physics*. How does the numerical evaluation affect the accuracy of the result?

Lecture 19 | Collision Probability Method

In the last lecture, we discussed the integral transport equation, derived its form in slab geometry, and provided a simple numerical approach based on Neumann iteration and numerical quadrature for solving homogeneous slab problems. In this lecture, we discuss the rather versatile *collision probability method* in slab geometry, though the method is certainly applicable to higher dimensions. We finish by providing a simple code that serves as the basis for several exercises.

Collision Probabilities

Recall Eq. 18.20, the integral equation in slab geometry:

$$\phi(x) = \int_{-\infty}^{\infty} dx' \frac{1}{2} E_1(\tau(x, x'))Q(x') \,, \tag{18.20}$$

where the emission density Q contains any external and scattering sources, all assumed to be isotropic. Suppose we apply this equation to a finite slab of length L with vacuum boundaries, much as we did toward the end of Lecture 18, or

$$\phi(x) = \int_{0}^{L} dx' \frac{1}{2} E_1(\tau(x, x'))Q(x') \,. \tag{19.1}$$

Let us assume the slab can be divided into a number of regions in which all cross-sections and external sources uniform, as in Figure 19.1. Within a region i spanning $x_{i-1/2}$ to $x_{i+1/2}$, we define an average flux

$$\phi(x_i) = \phi_i \equiv \frac{1}{\Delta_i} \int_{x_{i-1/2}}^{x_{i+1/2}} \phi(x) dx \,, \tag{19.2}$$

where $\Delta_i \equiv x_{i+1/2} - x_{i-1/2}$. Substituting the expression for ϕ in Eq. 19.1 into Eq. 19.2 yields

$$\phi_i = \frac{1}{\Delta_i} \int_{x_{i-1/2}}^{x_{i+1/2}} dx \int_{0}^{L} dx' \frac{1}{2} E_1(\tau(x, x'))Q(x') \,. \tag{19.3}$$

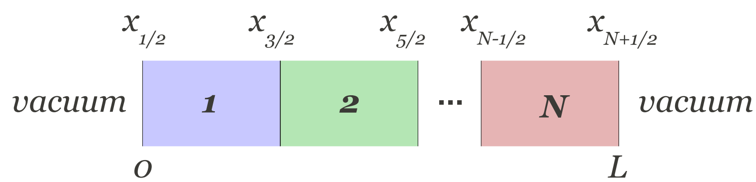

Figure 19.1: Slab discretization for CPM.

If we assume the flux within the cell is its average (a constant), then the emission density is also constant, and we find

$$\int_0^L dx' \frac{1}{2} E_1(\tau(x,x')) Q(x') dx \approx \sum_{i=1}^{N} Q_i \int_{x_{i-1/2}}^{x_{i+1/2}} dx' \frac{1}{2} E_1(\tau(x,x')) \,. \tag{19.4}$$

Substituting this into Eq. 19.3 (and being mindful of primes) yields

$$\begin{aligned} \phi_i &= \frac{1}{\Delta_i} \int_{x_{i-1/2}}^{x_{i+1/2}} dx \left(\sum_{i'=1}^{N} Q_{i'} \int_{x_{i'-1/2}}^{x_{i'+1/2}} dx' \frac{1}{2} E_1(\tau(x,x')) \right) \\ &= \frac{1}{\Delta_i} \sum_{i'=1}^{N} Q_{i'} \int_{x_{i-1/2}}^{x_{i+1/2}} dx \int_{x_{i'-1/2}}^{x_{i'+1/2}} dx' \frac{1}{2} E_1(\tau(x,x')) \end{aligned} \tag{19.5}$$

The collision probability method (as the name might imply) actually deals in terms of reactions rates rather than just the flux. If we multiply both sides of 19.5 by $\Sigma_i \Delta_i$, where Σ_i is the total cross-section in region i, we get an equation for $\phi_i \Sigma_i \Delta_i$, the total collision rate within region i. This equation is

$$\begin{aligned} \phi_i \Sigma_i \Delta_i &= \Sigma_i \sum_{i'=1}^{N} Q_{i'} \int_{x_{i-1/2}}^{x_{i+1/2}} dx \int_{x_{i'-1/2}}^{x_{i'+1/2}} dx' \frac{1}{2} E_1(\tau(x,x')) \\ &= \sum_{i'=1}^{N} Q_{i'} \Delta_{i'} \frac{\Sigma_i}{\Delta_{i'}} \int_{x_{i-1/2}}^{x_{i+1/2}} dx \int_{x_{i'-1/2}}^{x_{i'+1/2}} dx' \frac{1}{2} E_1(\tau(x,x')) \\ &= \sum_{i'=1}^{N} Q_{i'} \Delta_{i'} P_{ii'} \,, \end{aligned} \tag{19.6}$$

where

$$P_{ii'} \equiv \frac{\Sigma_i}{\Delta_{i'}} \int_{x_{i-1/2}}^{x_{i+1/2}} dx \int_{x_{i'-1/2}}^{x_{i'+1/2}} dx' \frac{1}{2} E_1(\tau(x,x')) \tag{19.7}$$

is the *first-flight collision probability*. Notice we rearranged terms such that $P_{ii'}$ is unitless. Moreover, $P_{ii'}$ can be interpreted as the probability that a neutron born uniformly and isotropically in a region i' makes its first collision in region i.

Solving the Equations

Eq. 19.6 provides a set of equation for the scalar flux (or rather, the collision rates) in terms of the region emission densities Q_i and collision probabilities $P_{ii'}$. In our case of slab geometry, we can compute the collision probabilities analytically. We simply list the results, leaving the derivations as exercises:

$$\begin{aligned} P_{ii} &= 1 - \frac{1}{2\Sigma_i \Delta_i}\Big(1 - 2E_3(\Sigma_i \Delta_i)\Big) \\ P_{ii'} &= \frac{1}{2\Sigma_i' \Delta_i'}\Big(E_3(\tau_{ii'}) - E_3(\tau_{ii'} + \Delta_i \Sigma_i) - E_3(\tau_{ii'} + \Delta_{i'} \Sigma_{i'}) \\ &\quad + E_3(\tau_{ii'} + \Delta_i \Sigma_i + \Delta_{i'} \Sigma_{i'})\Big)\,, \;\; i \neq i'\,, \end{aligned} \tag{19.8}$$

where

$$\tau_{ii'} \equiv \begin{cases} \tau(x_{i+1/2}, x_{i'-1/2}) & i' > i \\ \tau(x_{i'+1/2}, x_{i-1/2}) & i' < i \end{cases} . \tag{19.9}$$

So how do we solve the equations once we have the $P_{ii'}$? We have a system of equations of the form

$$\Delta_i \Sigma_i \phi_i = \sum_{i'=1}^{N} (\Delta_{i'} \Sigma_{si'} \phi_{i'} + S_{i'} \Delta_{i'}) P_{ii'}\,. \tag{19.10}$$

If we rearrange terms, bringing all ϕ's to the left hand side, we can write for $i = 1$

$$\Delta_1(\Sigma_1 - P_{11}\Sigma_{s1})\phi_1 - P_{12}\Delta_2\Sigma_{s2}\phi_2 + \ldots = S_1\Delta_1 P_{11} + S_2 \Delta_2 P_{12} + \ldots \tag{19.11}$$

If we define $f_i = \Delta_i \Sigma_i \phi_i$ and $s_i = \sum_{i'} S_{i'} \Delta_{i'} P_{ii'}$, this becomes

$$\Big(1 - \frac{\Sigma_{s1}}{\Sigma_1} P_{11}\Big) f_1 + \Big(-\frac{\Sigma_{s2}}{\Sigma_2} P_{12}\Big) f_2 + \ldots = s_i\,. \tag{19.12}$$

Generalizing to matrix form, we have

$$\mathbf{H}\mathbf{f} = \mathbf{s}\,, \tag{19.13}$$

where

$$\mathbf{H} = \begin{bmatrix} 1 - \frac{\Sigma_{s1}}{\Sigma_1} P_{11} & -\frac{\Sigma_{s2}}{\Sigma_2} P_{12} & -\frac{\Sigma_{s3}}{\Sigma_2} P_{13} & \cdots & \\ -\frac{\Sigma_{s1}}{\Sigma_1} P_{21} & 1 - \frac{\Sigma_{s2}}{\Sigma_2} P_{22} & -\frac{\Sigma_{s3}}{\Sigma_2} P_{23} & \cdots & \\ & & & \ddots & \end{bmatrix} . \tag{19.14}$$

For a purely absorbing case, Eq. 19.13 is quite simple, as $\mathbf{H}$ reduces to the identity matrix. Suppose we have a four region problem, with a source in just the third region. The resulting set of linear equations is simply

$$\mathbf{H} = \left[\begin{array}{cccc} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{array} \right] \left[\begin{array}{c} \Delta_1 \Sigma_1 \phi_1 \\ \Delta_2 \Sigma_2 \phi_2 \\ \Delta_3 \Sigma_3 \phi_3 \\ \Delta_4 \Sigma_4 \phi_4 \end{array} \right] = \left[\begin{array}{c} S_3 P_{13} \Delta_3 \\ S_3 P_{23} \Delta_3 \\ S_3 P_{33} \Delta_3 \\ S_3 P_{43} \Delta_3 \end{array} \right] , \qquad (19.15)$$

the solution of which is simply $\phi_1 = (S_3 P_{13} \Delta_3)/(\Delta_1 \Sigma_1)$ and so forth.

Benefits and Limitations

Eq. 19.13 is powerful (as are integral methods in general) in that no angular approximation in the flux are made. However, the sources and scattering are assumed to be isotropic, which does limit to an extent the accuracy in angle.

Furthermore, the collision probability suffers from two rather significant deficiencies. First, the matrix $\mathbf{H}$ is in general a dense matrix. Hence, for large problems, the memory requirements are significant. Furthermore, the flat flux approximation within a region is only first-order accurate (meaning errors are proportional to Δ_i).

More Advanced Aspects

In the discussion above, we considered only vacuum conditions. The collision probability method has been most heavily used in lattice physics calculations, where the domain is often represented as an infinite array of identical unit cells via periodic boundary conditions.

To derive these conditions would be beyond our intended scope, but the reader might conclude the derivation would, theoretically, involve infinite sums over $P_{ii'}$, and this is indeed the case. The interested reader is encouraged to read Section 3.8.3 of Hebert [12] for discussion of a suitable numerical approach.

A Simple Code

Here, we provide a simple code for computing the scalar flux in a slab consisting of several regions of uniform composition, given in Listing 19.1. The fundamental components of the code are just those that compute τ and $P_{ii'}$ and should be readily understood. As in the Neumann iteration code of the last lecture, MATLAB's built in symbolic function for the exponential integrals is used. This is very slow, and so the student is encouraged to find efficient and accurate numerical techniques to evaluate these functions.

Listing 19.1: Solution of Slab Problem via CPM

```
function phi = cpm(Delta,SigT,SigS,Source)
% function phi = cpm(Delta,SigT,SigS,Source)
%   This function solves the one-speed transport equation in slab geometry
%   for the scalar flux using a the collision probability method.
%
%   Inputs: (all must be column vectors, i.e dimension [n,1]
%       Delta   -- region widths [cm]
%       SigT    -- region total cross-section [1/cm]
%       SigS    -- region scattering cross-section [1/cm]
%       Source  -- region uniform isotropic source [1/cm^3-s]
%   Output
%       phi     -- scalar flux in each region
tau = comptau(Delta,SigT);  % compute tau
P = ffcp(Delta,SigT,tau);   % compute collision probabilities
H = zeros(length(Delta));   % set up H and right hand side
for i = 1:length(Delta)
    for j = 1:length(Delta)
        H(i,j) = -SigS(j)/SigT(j)*P(i,j);        % add diagonal 1's below
        s(i) = sum( Source(i)*Delta(i)*P(i,:) ); % right hand side
    end
end
H = H + eye(length(Delta));     % add 1's to the diagonal
f = ( H \ s');                  % solve for the collision rates
phi = f ./ ( SigT .* Delta );   % solve for the flux
end

function tau = comptau(Delta,SigT)
% function tau = comptau(Delta,SigT)
%   This function computes the optical pathlengths between regions.
%
%   Inputs:
%       Delta   -- region widths [cm]
%       SigT    -- region total cross-section [1/cm]
%   Output:
%       tau     -- optical path length
tau = zeros(length(Delta));
DeltaSigT = Delta.*SigT;    % length of regions in mfp's
for i = 1:length(Delta)
    % tau(i,i) remains 0
    for ip = i+1:length(Delta) % only ip > i, since tau(x,x')=tau(x',x)
        % adding optical distance between x(i+1/2) and x(ip-1/2)
        tau(i,ip) = sum( DeltaSigT(i+1:ip-1) );
        tau(ip,i) = tau(i,ip);
    end
end
end

function P = ffcp(Delta,SigT,tau)
% function P = ffcp(Delta,SigT,tau)
%   This function computes the first-flight collision probabilities.  Note,
%   it uses MATLAB's symbolic function for E3, which is REALLY SLOW.  The
%   student is strongly encouraged to look up fast and accuracte numerical
%   approximations to the En functions.  See Hebert's textbook.
%
%   Inputs:
%       Delta   -- region widths [cm]
%       SigT    -- region total cross-section [1/cm]
%       tau     -- optical path length
%   Outputs:
%       P       -- first-flight collision probabilities
P = zeros(length(Delta));
```

```
for i = 1:length(Delta)
    P(i,i) = 1 - 0.5/SigT(i)/Delta(i) * ...
            ( 1 - 2*mfun('Ei', 3, SigT(i)*Delta(i)) );
    for j = 1:i-1
        P(i,j) = 0.5/SigT(j)/Delta(j) * ...
            ( mfun('Ei', 3, tau(i,j)) - ...
              mfun('Ei', 3, tau(i,j)+Delta(i)*SigT(i)) - ...
              mfun('Ei', 3, tau(i,j)+Delta(j)*SigT(j)) + ...
              mfun('Ei', 3, tau(i,j)+Delta(j)*SigT(j)+Delta(i)*SigT(i) ));
        P(j,i) = P(i,j); % reciprocity in action!
    end
end
end
```

Further Reading

This lecture has largely followed Section 5-3 of Lewis and Miller [16]. There, the student will find some more details regarding periodic conditions for slab geometry as well as some information regarding a 2-d implementation of CPM in later sections. Hebert [12] also discusses CPM in several geometries and provides sample code relevant for both slab and a 2-d pin cell problems.

Exercises

1. **CPM and Scattering**. Use the CPM code to repeat Exercise 6 of Lecture 18. Does increase scattering have any effect on the computational time? Why or why not?
2. **Source Iteration**. In our (relatively standard) formulation of the collision probability method, we end up with a dense matrix $\mathbf{H}$. Reformulate the approach such that the scattering term is *not* brought to the left hand side. In this case, $\mathbf{H}$ is just the identity. However, the right hand side now contains an additional scattering source contribution, which depends on the (unknown) fluxes. Develop an approach such that the flux is initially guessed, the right hand side is formed, new fluxes are computed, and the right hand side is updated, yielding an iterative process called *source iteration*.
3. **CPM with Source Iteration**. Implement the *source iteration* you developed in the last exercise in the given CPM code. Repeat Exercise 6 from Lecture 18, using a stopping criterion of

$$\frac{|\phi_i^n - \phi_i^{n-1}|}{\phi_i^{n-1}} < \epsilon = 1 \times 10^{-6} \quad \forall\, i\,,$$

   where n is the iteration index. Comparing to Exercise 1, is there any benefit to using source iteration in place of solving a dense matrix system directly? If not for this particular problem, underwhat conditions would you expect source iteration to be better?

4. **Understanding Periodic Conditions**. Read Section 3.8.3 of Hebert and provide a short synopsis of period conditions can be treated in slab geometry.
5. **Implementing Periodic Conditions**. Implement periodic conditions in the given CPM code, following the method described by Hebert.

Lecture 20 | Discrete Ordinates Method

In the last two lectures, we dealt exclusively with the integral form of the transport equation and ways to solve it numerically. Now, we return to the integro-differential form of the equation and introduce the *discrete ordinates (S_N) method*, one of the most widely-used deterministic transport methods. We subsequently introduce the method of *source iteration* for treating the scattering source and finish with a brief overview of acceleration via the methods of coarse mesh rebalance (CMR) and coarse mesh finite difference (CMFD). As in previous lectures, we provide a simple code that implements some of the concepts discussed.

A Discrete Angle Domain

The integral methods of the last two lectures are useful in many applications because they effectively eliminate the angular variable. If we are to treat the transport equation in its integro-differential form, we need a way to handle the angles. For the sake of brevity, we limit our discussion to transport in slab geometry with isotropic sources and scattering, for which the transport equation reduces to

$$\begin{aligned} \mu\frac{\partial\psi}{\partial x} + \Sigma_t(x)\psi(x,\mu) &= \frac{\Sigma_s(x)}{2}\int_{-1}^{1} d\mu'\psi(x,\mu') + \frac{S(x)}{2} \\ &= \frac{\Sigma_s(x)}{2}\phi(x) + \frac{S(x)}{2} \end{aligned} \tag{20.1}$$

The discrete ordinates method consists of requiring Eq. 20.1 to hold only for discrete values μ. In other words, we require

$$\mu_n\frac{\partial\psi}{\partial x} + \Sigma_t(x)\psi(x,\mu_n) = \frac{\Sigma_s(x)}{2}\phi(x) + \frac{S(x)}{2}\,. \tag{20.2}$$

What about ϕ? This is where the approximation is actually made. Because we have ψ only at discrete angles μ_n, we cannot perform the continuous integral. We do the next best thing and compute a weighted sum:

$$\phi(x) \approx \sum_{n=1}^{N} w_n \psi_n(x)\,, \tag{20.3}$$

where we have introduced the notation $\psi_n(x) \equiv \psi(x, \mu_n)$.

The only actual requirement on w_n is that they satisfy

$$\sum_{n=1}^{N} w_n = 2\,, \tag{20.4}$$

since this sum represents $\int_{-1}^{1} d\mu = 2$. Of course, we don't choose the weights arbitrarily, nor do we choose the cosines μ_n arbitrarily. The choice of both is the art of numerical quadrature.

Intermission: Numerical Quadrature (a.k.a. Numerical Integration)

You saw application of the *trapezoid rule* in the integral transport code of Lecture 18. You have probably seen the trapezoid rule in previous classes, and if not that, then certainly the even simpler *Riemann sum* approximation to integrals, here in the form of the *midpoint rule*:

$$\int_a^b f(x)dx \approx \Delta \sum_{n=0}^{N-1} f\Big(a + (n + 1/2)\Delta\Big)\,, \tag{20.5}$$

where

$$\Delta = \frac{b-a}{N}\,, \tag{20.6}$$

and N is the number of divisions in the center of which f is evaluated.

The general form of a quadrature rule is

$$\int_a^b f(x)dx \approx \sum_{n=0}^{N-1} w_n f(x_n)\,, \tag{20.7}$$

and we see that the midpoint rule chooses equal weights $w_n = w = \Delta$ and evaluates f at equally spaced points $x_n = a + (n + 1/2)\Delta$. The trapezoid rule also takes evenly spaced points, $x_n = a + n\Delta$, for $n = 0 \ldots N$, and with $w_0 = w_N = 0.5\Delta$, and $w_n = \Delta$, $0 < n < N$.

Other quadrature sets using evenly spaced points also exist. One general class consists of the *Newton-Cotes formulas*, which interpolate $f(x)$ using various order polynomials and thus allows for analytic integration. The midpoint rule and the trapezoid rule are the simplest examples, using zeroth order and first order polynomials. Another example is *Simpson's rule*, defined

$$\begin{aligned}\int_a^b f(x)dx \approx &\frac{\Delta}{3}\Big(f(a)+4f(a+\Delta)+2f(a+2\Delta)+4f(a+3\Delta)\\ &+2f(a+4\Delta)+\ldots+4f(b-\Delta)+f(b)\Big)\,.\end{aligned} \tag{20.8}$$

Simpson's rule is based on a piece-wise quadratic interpolation of sets of three points and can often be a very good "quick and dirty" way to integrate numerically. The reader is encouraged to consult a numerical analysis text for more details.

In many cases, it is possible to use fewer, non-equally spaced points (with unique weights) to yield accurate integral approximations that would otherwise take many equally spaced points. One such scheme is Gauss-Legendre quadrature, which is the standard quadrature used to define ϕ in Eq. 20.3. The n-point Gauss-Legendre scheme is constructed so that polynomials of degree less than or equal to $2n-1$ are integrated exactly over the domain of interest.

Consider a 2-point Gauss-Legendre scheme, i.e. $\int_{-1}^1 f(x)dx = w_1 f(x_1) + w_2 f(x_2)$. We have four unknowns, so we need four equations. The simplest case is to integrate the monomials 1, x, x^2, and x^3 over the desired range. Doing so, we get the set of equations

$$\begin{aligned}\int_{-1}^1 1dx &= 2 = w_1+w_2\\ \int_{-1}^1 xdx &= 0 = w_1x_1+w_2x_2\\ \int_{-1}^1 x^2dx &= \frac{2}{3} = w_1x_1^2+w_2x_2^2\\ \int_{-1}^1 x^3dx &= 0 = w_1x_1^3+w_2x_2^3\,.\end{aligned} \tag{20.9}$$

This system is solved for $w_1 = w_2 = 1$, $x_1 = -1/\sqrt{3}$, and $x_2 = 1/\sqrt{3}$. Interestingly, x_1 and x_2 are the roots of the second-order Legendre polynomial $P_2(x) = x^2 - 1/3$. In fact, the x_i for any n-point Gauss-Legendre scheme are the n roots of the nth order Legendre polynomial—hence the name of the quadrature rule*. The

*It should be noted that an entire class of Gaussian quadratures exists based on use of other sets of polynomials to approximate integrals where f is weighted by some weighting function associated with the polynomials. An important example is the Gauss-Chebyshev quadrature, where we approximate the integral $\int_{-1}^1 \frac{f(x)}{\sqrt{1-x^2}}dx$. The x_i are found to be roots of the Chebyshev polynomials.

weights are defined

$$w_i = \frac{2(1 - x_i^2)}{(n+1)^2 (P_{n+1}(x_i))^2} \,. \tag{20.10}$$

As a reference, the Gauss-Legendre abscissa and weights are given in Table 20.1 for $N = 2$, 4, and 6. Note, odd orders are not used because the abscissa would then include $x = 0$. Since our goal is ultimately to integrate over μ, and since we shall encounter terms with $1/\mu$, we simply avoid the singularity by choosing even N.

Table 20.1: Gauss-Legendre quadrature parameters.

N	μ_i	w_i
2	± 0.5773502691	1
4	± 0.8611363115	0.3478548451
	± 0.3399810435	0.6521451549
6	± 0.9342695142	0.1713244924
	± 0.6612093864	0.3607615730
	± 0.2386191860	0.4679139346

A Discrete Spatial Domain

We next deal with the spatial variable, using the finite volume method. Consider the discretization of Figure 20.1. The slab is discretized into I regions, each of which is assumed to have constant cross-sections and sources, denoted Σ_{ti} and so forth. The cell centers are given by x_i, and the cell edges by $x_{i\pm 1/2}$. For a slab defined over $0 \leq x \leq L$, $x_{1/2} = 0$ and $x_{I+1/2} = L$.

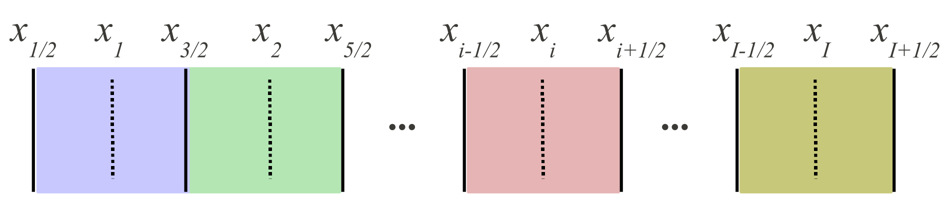

Figure 20.1: Discrete ordinates spatial discretization.

Now, we integrate Eq. 20.2 over cell i:

$$\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} dx \Big(\mu_n \frac{\partial \psi_n}{\partial x} + \Sigma_t(x) \psi_n(x) \Big) = \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} dx \Big(\frac{\Sigma_s(x)}{2} \phi(x) + \frac{S(x)}{2} \Big) \,, \tag{20.11}$$

so that

$$\mu_n(\psi_{i+\frac{1}{2},n} - \psi_{i-\frac{1}{2},n}) + \Delta_i \Sigma_{ti} \psi_{i,n} = \frac{\Delta_i}{2}\Big(\Sigma_{is}\phi_i + S_i\Big) \,, \tag{20.12}$$

where $\Delta_i \equiv x_{i+\frac{1}{2}} - x_{i-\frac{1}{2}}$ and where the integrals have been approximated via the midpoint rule where needed. For the sake of brevity, we switch to the emission density, defining

$$Q_i = (\Sigma_{is}\phi_i + S_i)/2 \,, \tag{20.13}$$

so that

$$\mu_n(\psi_{i+\frac{1}{2},n} - \psi_{i-\frac{1}{2},n}) + \Delta_i \Sigma_{ti} \psi_{i,n} = \Delta_i Q_i \,. \tag{20.14}$$

Eq. 20.14 is a good first step, but we're not done. Imagine we know the angular flux entering the left side of slab i. We would like to be able to compute the angular flux leaving i to the right. Doing exactly this over the entire domain is referred to as "sweeping" and is a common technique for hyperbolic problems, as it allows us to propagate boundary conditions along the direction of neutron paths. The problem with Eq. 20.14 is that we have another unknown: $\psi_{i,n}$, the cell-centered angular flux. There are several approximations we can use, and we develop two: the diamond difference (DD) and step difference (SD) methods.

For both DD and SD, we define the cell-centered angular flux by

$$\psi_{i,n} = \frac{1+\alpha_n}{2}\psi_{i+\frac{1}{2},n} + \frac{1-\alpha_n}{2}\psi_{i-\frac{1}{2},n} \,, \tag{20.15}$$

where

$$\alpha_n = \begin{cases} 0 & \text{for DD} \\ |\mu_n|/\mu_n & \text{for SD} \end{cases} \,. \tag{20.16}$$

The DD method is probably the most ubiquitous of the simple schemes available. It essentially says the angular flux varies linearly within a cell, and it provides second-order accuracy. Unfortunately, the DD method can lead to negative fluxes for Δ_i too large, demonstration of which is left as an exercise. On the other hand, the SD method cannot yield negative fluxes, but is only first-order accurate. For the SD method, note that the cell-center angular flux is equal to the incident angular flux at the edge.

Sweeping Formulas

We hinted above that we can solve the discrete ordinates equations via use of sweeps across the domain. In slab geometry, this consists of sweeps to the right for $\mu_n > 0$ and sweeps left for $\mu_n < 0$. If we have vacuum or some other known boundary flux, we start with that condition, as reflective conditions require information from a sweep having come from the opposite direction. If all conditions are reflective, a guess must be made for one, which will likely increase the number of iterations required (we'll get to iterations below).

Substituting Eq. 20.15 into Eq. 20.14 and rearranging, we find for sweeping right

$$\psi_{i+\frac{1}{2},n} = A^+_{i,n}\psi_{i-\frac{1}{2},n} + B^+_{i,n}Q_i\,, \tag{20.17}$$

and sweeping left

$$\psi_{i-\frac{1}{2},n} = A^-_{i,n}\psi_{i+\frac{1}{2},n} + B^-_{i,n}Q_i\,, \tag{20.18}$$

where

$$A^{\pm}_{i,n} = \frac{2\mu_n \mp (1 \mp \alpha_m)\Sigma_{ti}\Delta_i}{2\mu_n \pm (1 \pm \alpha_m)\Sigma_{ti}\Delta_i}\,, \tag{20.19}$$

and

$$B^{\pm}_{i,n} = \frac{\pm 2\Delta_i}{2\mu_n \pm (1 \pm \alpha_m)\Sigma_{ti}\Delta_i}\,. \tag{20.20}$$

Source Iteration

When we introduce scattering into the emission density, the right hand side becomes dependent on the solution. A common way around this is the so-called source iteration method. The source iteration method is related to the Neumann iteration scheme discussed in Lecture 18, and a connection between the two is left as an exercise.

Source iteration is characterized by a right hand side that lags behind the left hand side. More explicitly, we solve the equation

$$\mu_n(\psi^m_{i+\frac{1}{2},n} - \psi^m_{i-\frac{1}{2},n}) + \Delta_i\Sigma_{ti}\psi^m_{i,n} = \Delta_i Q^{m-1}_i\,, \tag{20.21}$$

where m is the iteration index and

$$Q^{m-1} = \frac{1}{2}(\Sigma_{is}\phi^{m-1}_i + S_i)\,. \tag{20.22}$$

For $m = 1$, we need q^0. Often, ϕ^0 is simply set to zero, unless a better initial guess is easily made.

The iterations continue until the updated fluxes change insignificantly. This is usually quantified in a relative sense for the scalar flux by

$$\frac{\max|\phi^m - \phi^{m-1}|}{\phi^{m-1}} < \epsilon_\phi \tag{20.23}$$

for some small ϵ_ϕ.

Acceleration

The discrete ordinates method is in theory a very fast, memory-efficient technique. Unless the angular flux is needed explicitly, only one set of edge angular fluxes is needed at a time (e.g. the left edge flux for computing the right edge flux, which then can overwrite the left edge flux).

Despite the algorithmic simplicity of the method, the underlying source iteration scheme can be obnoxiously slow for problems with high scattering ratios. Several methods have been introduced over the years to accelerate the source convergence. Of these, diffusion synthetic acceleration (DSA) is probably the most powerful and widespread. The method essentially involves performing a diffusion solve to update the scalar flux, and has shown spectacular success for a wide variety of problems. However, DSA can be difficult if not impossible to implement for certain geometries and/or discretization schemes, as it requires a very strict consistency between the transport and diffusion mesh. Instead, we look at two other relatively simple methods that have been quite successful: the *coarse mesh rebalance* (CMR) and *coarse mesh finite difference* (CMFD) methods.

Coarse Mesh Rebalance

CMR was probably the first wide-spread method for acceleration discrete ordinates calculations. Its foundation rests in the *neutron balance equation*, which we obtain in 1-d by integrating Eq. 20.1 over angle, yielding

$$\frac{\partial}{\partial x}J(x) + \Sigma_t(x)\phi(x) = \Sigma_s(x)\phi(x) + S(x)\,, \tag{20.24}$$

where $J = \mathbf{J}\cdot\hat{i}$ is the net current in the x direction. Now suppose we divide the domain into a number of coarse meshes, indexed by j, with cell edges at $x_{j-\frac{1}{2}}$ and $x_{j+\frac{1}{2}}$. Subtracting the scattering source from both sides and integrating over the jth coarse mesh, we obtain

$$J_{j+\frac{1}{2}} - J_{j-\frac{1}{2}} + \int_{x_i} \Sigma_r(x)\phi(x)dx = \int_{x_i} S(x)dx\,. \tag{20.25}$$

Expressing Eq. 15.2 as $J = J^+ - J^-$, where $+$ indicates to the right and $-$ to the left, we have

$$J^+_{j+\frac{1}{2}} - J^-_{j+\frac{1}{2}} - J^+_{j-\frac{1}{2}} + J^-_{j-\frac{1}{2}} + R_j = S_j\,, \tag{20.26}$$

where the partial currents are defined

$$J^{\pm}_{j+\frac{1}{2}} = \sum_{\mu_n \gtrless 0} \mu_n \psi_{j+\frac{1}{2},n}\,, \tag{20.27}$$

R_j is the coarse mesh-integrated removal rate, and S_j is the coarse mesh-integrated source.

Eq. 20.26 must be satisfied by a fully converged fine mesh solution. For an unconverged iterate, which we denote $\psi^{m+\frac{1}{2}}$, Eq. 20.26 is generally not satisfied. To force the flux to satisfy neutron balance, we introduce multiplicative rebalance factors f_j such that

$$\psi_{i,n}^{m+1} = \begin{cases} f_j \psi_{i,n}^{m+\frac{1}{2}} & x_{j-\frac{1}{2}} < x_i < x_{j+\frac{1}{2}} \\ f_{j-1} \psi_{i,n}^{m+\frac{1}{2}} & x_i = x_{j+\frac{1}{2}} \text{ and } \mu_n > 0 \\ f_{j+1} \psi_{i,n}^{m+\frac{1}{2}} & x_i = x_{j+\frac{1}{2}} \text{ and } \mu_n < 0 \end{cases} . \tag{20.28}$$

Note the appropriate factor's index denotes from which coarse mesh the neutrons originate. For the case of vacuum boundaries, the corresponding incident partial current vanishes. For a reflective boundary at the left, $f_0 = f_1$, and similarly for the right boundary.

To illustrate the method, consider a slab with vacuum boundaries divided into three coarse meshes. Substituting the modified fluxes into the balance equation yields a set of three equations:

$$\begin{aligned} f_1 J_{3/2}^+ - f_2 J_{3/2}^- + f_1 J_{1/2}^- + f_1 R_1 &= f_1 S_1 \\ f_2 J_{5/2}^+ - f_3 J_{5/2}^- - f_1 J_{3/2}^+ + f_2 J_{3/2}^- + f_2 R_2 &= f_2 S_2 \\ f_3 J_{7/2}^+ - f_2 J_{5/2}^+ + f_3 J_{5/2}^- + f_3 R_3 &= f_3 S_3 \, , \end{aligned} \tag{20.29}$$

which yields a tridiagonal system written in condensed form as

$$\mathbf{A}\mathbf{f} = \mathbf{S} \, . \tag{20.30}$$

Solving this equation for $\mathbf{f}$ and updating to ψ^{m+1} allows us to recompute the scattering source, which—hopefully—converges faster than without the rebalancing scheme. Look to the Further Reading section for references that investigate stability of coarse mesh rebalance. Algorithm 1 shows how CMR (or any low order acceleration scheme) can be used within source iteration.

Coarse Mesh Finite Difference

The coarse mesh finite difference method has its origins in a nonlinear acceleration technique developed by K. Smith that makes use of discontinuity factors. With discontinuity factors, any region-integrated high order solution (e.g. from fine mesh transport) can be represented by a low order method (e.g. coarse mesh diffusion).

initialization;
while *ψ^m or ϕ^m not converged* **do**
  compute scattering source ;
  $\psi^{m+\frac{1}{2}} \leftarrow$ sweep(ψ^m) ;
  if *using CMR* **then**
    $\psi^{m+1} \leftarrow$ cmr$(\psi^{m+\frac{1}{2}})$;
  else
    $\psi^{m+1} \leftarrow \psi^{m+\frac{1}{2}}$;
  end
end

Algorithm 1: Accelerated Source Iteration

CMFD makes use of this idea to represent partially converged high order solutions in low order form, followed by subsequent iterations in the low order domain. This approach is a physical analog of multigrid methods, and its main effect is to dampen low order errors. What this means is that the high order method will quickly get the "shape" right, and a low order acceleration will quickly get the "magnitude" right.

We sketch the implementation. Given a partially converged fine mesh solution, homogenized diffusion coefficients and cross-sections are found for each coarse mesh, as are volume-averaged sources and fluxes. A coarse mesh diffusion equation is to be solved, but due to an incompletely converged fine mesh solution, discontinuity factors—additive corrections to the effective diffusion coefficients—must be computed to enforce net current continuity between coarse meshes. Once these are defined, the coarse mesh diffusion equations can be solved. The ratio of the updated average flux to the original average flux of a coarse mesh is used to scale the original fine mesh solution. The interested student should see the references and exercises for further details.

A Simple Code

Here, we provide a simple discrete ordinates code for computing the angular and scalar flux in a slab, given in Listing 20.1. The input requires region (coarse mesh) edge locations, within which the number of fine meshes are defined. Each region is assigned a material index and volumetric source strength. The total and scattering cross-sections represent values for different materials; each material can be placed in a region using the appropriate index. This quick input format allows one to investigate fairly easily a wide range of slab compositions.

Currently, the code is limited to the DD and S_4 approximations and vacuum boundaries. Convergence is tested for ϕ only. The code returns the mesh-centered scalar flux and mesh-edge angular flux. Within the solver, the notation follows the lecture notation relatively closely.

Listing 20.1: Solution of Slab Problem via Sn

```
function [phi,psi] = sn(EDGE,NFM,SigT,SigS,RegMat,Source)
% function [phi,psi] = sn(EDGE,NFM,Sigma,RegMat,Source)
%   This function solves the one-speed transport equation in slab geometry
%   using the discrete ordinates method.  The S4 and diamond difference
%   approximations are implemented, and only vacuum boundaries are treated.
%
%   Inputs:
%       EDGE    -- region edges
%       NFM     -- number of fine meshes per region
%       SigT    -- total cross-section for each material
%       SigS    -- scattering cross-section for each material
%       RegMat  -- which material goes in each region
%       Source  -- region uniform isotropic source [1/cm^3-s]
%   Output
%       phi     -- fine mesh-centered scalar flux
%       psi     -- fine mesh-edge angular flux

% assign angles and weights
mu = [ -0.8611363115 -0.3399810435  0.3399810435  0.8611363115 ];
wt = [  0.3478548451  0.6521451549  0.6521451549  0.3478548451 ];
% allocate
totNFM = sum(NFM);            % total number of fine meshes
psi    = zeros(totNFM+1,4);   % current angular flux
phi    = zeros(totNFM,1);     % current scalar flux
S      = zeros(totNFM,4);     % fine mesh source
Q      = S;                   % emission density
fmmid  = zeros(totNFM,1);     % fine mesh material id
% compute discretization
j = 0;
for i = 1:length(NFM)
    Delta( (j+1):(j+NFM(i)) )     = ( EDGE(i+1) - EDGE(i) ) / NFM(i);
    S( (j+1):(j+NFM(i)), : )      = Source(i)/2;
    fmmid( (j+1):(j+NFM(i))    )  = RegMat(i);
    j = sum(NFM(1:i));
end
% precompute coefficients, following Eqs. 20.16-20.20.
alpha = zeros(4,1);
A     = zeros(totNFM,4);
B     = A;
for i = 1:totNFM
    m = fmmid(i);
    for n = 1:4  % use the sign of mu to choose appropriate form
        smu    = sign(mu(n));
        denom  = 2*mu(n)+smu*(1+smu*alpha(n))*SigT(m)*Delta(i);
        A(i,n) = (2*mu(n)-smu*(1-smu*alpha(n))*SigT(m)*Delta(i)) / denom;
        B(i,n) = smu * 2 * Delta(i) / denom;
    end
end
% convergence parameters
eps_phi = 1e-5; max_it  = 200;
err_phi = 1;    it = 0;
% Begin source iterations
while (err_phi > eps_phi && it <= max_it )
    % Save old scalar flux
    phi0 = phi;
    % Update source
    for i = 1:totNFM
        Q(i,:) = S(i,:) + 0.5*SigS(fmmid(i))*phi(i);
    end
```

```
    % Perform sweeps
    for i = totNFM:-1:1  % right-to-left
        psi(i,1:2) = A(i,1:2).*psi(i+1,1:2) + B(i,1:2).*Q(i,1:2);
    end
    for i = 1:totNFM     % left-to-right
        psi(i+1,3:4) = A(i,3:4).*psi(i,3:4) + B(i,3:4).*Q(i,3:4);
    end
    % --- Insert Acceleration Here ---
    % Update phi (need cell-centered psi, so use Eq. 20.15)
    for i = 1:totNFM
         phi(i) = sum( wt(:) .* (0.5*( (1+alpha(:)).*psi(i+1,:)' + ...
                                       (1-alpha(:)).*psi(i,:)'    ) ));
    end
    % Update error and iteration counter
    err_phi =  max(  abs(phi-phi0)./phi0 );
    it = it + 1;
end
if (it <= max_it)
    disp(['Converged in ',num2str(it), ' iterations.'])
else
    disp('Failed to converge.')
end

end
```

Further Reading

A good introduction to the discrete ordinates method in 1-d can be found in Chapter 3 in Lewis and Miller [16]; higher dimensions are covered in Chapter 4 of the same work. The foundation of the method is usually credited to Chandrasekhar (in the context of stellar radiation) and can be found in his monograph [7].

A review paper by Adams and Larsen [1] provides a survey of the many acceleration techniques available for the discrete ordinates equations, and with its several hundred references, is the place to look for further information. Lewis and Miller provides more information on coarse mesh rebalance, and Cefus and Larsen have assessed its stability [6]. Park and Cho [19] have suggested angular-dependent rebalance schemes, and their work is a good place to start to find references to other CMR variations. The discontinuity factors used in CMFD were first proposed by Smith in the context of homogenization [21], and he later proposed their use for acceleration [22]. A recent paper by Zhong et al. [29] provides a modern overview and advanced use of the approach along with a relatively detailed set of equations for implementation.

Recent advances in the discrete ordinates method include employing advanced Krylov subspace methods, outlined by Warsa, Wareing, and Morel [25], variations of which are implemented in the state-of-the-art code Denovo [11]. Moreover, the discrete ordinates equations can be parallelized; the most popular approach is that of Koch, Baker, and Alcouffe [14] (also implemented in Denovo).

For more information on numerical quadrature, see any number of numerical analysis books. Also, MIT course 18.335 is also highly recommended, as it covers numerical quadrature and many other aspects of numerical methods (including a heavy focus on numerical linear algebra).

Exercises

1. **Negative fluxes**. Using the DD method, express the sweeping step relation in the form
$$\psi_{3/2,n} = A\psi_{1/2,n} + B \,,$$
where $\mu_n > 0$ and A and B are constants to be determined, and eliminate any α terms. Under what conditions can $\psi_{3/2,n}$ be negative? Under what conditions, if any, can $\phi_{1,n}$ be negative? While negative fluxes are not inherently bad numerically, they don't make much physical sense. Suggest a method for "fixing" negative fluxes.
2. **Step Difference**. Implement the step difference method.†
3. **Step Characteristic**. The step characteristic (SC) scheme is another differencing approach based on analytical integration of the transport equation over the fine mesh cell, yielding for example
$$\psi_{i+\frac{1}{2},n} = \psi_{i-\frac{1}{2},n} e^{-\frac{\Delta_i \Sigma_{ti}}{\mu}} + \frac{Q_i}{2\Sigma_{ti}}\left(1 - e^{-\frac{\Delta_i \Sigma_{ti}}{\mu}}\right) .$$
   (a) Derive an expression for α_{mi} as in Eq. 20.28 for use with the SC method. Note that α in this case is also indexed by the fine mesh index i.
   (b) Implement the SC method.
4. **Accuracy**. Here, we want to investigate the accuracy of various difference methods. To do so, we will use a known analytical solution for as a reference. See e.g. Appendix A of LeVeque [15] for more on error analysis.
   (a) Using the S_6 approximation, solve *analytically* the discrete ordinates equations for the sample slab problem of Lecture 17. Implement this reference solution as a function in MATLAB (or your language of choice), so that $\phi(x)$ can be evaluated for any value of x.
   (b) Using the DD approximation, solve the S_6 equations using 10, 20, 40, 80, and 160, and 320 meshes. For each case, compute the absolute value of the maximum error between the DD and analytical solution for ϕ, i.e. find
$$e_I^{\text{DD}} = \max_{1\leq i \leq I} \left|\phi_i^{\text{DD}} - \phi_i^{\text{ref}}\right| .$$

†Note, in this and other exercises you are asked to use or modify the given S_N code. It is also acceptable to use your own code, either written from scratch or via modification of the one given.

(c) Do the same for the SD method.

(d) We say a method is pth order accurate if $e(\Delta) \propto \Delta^p$. Estimate p for both methods.

(e) Plot the errors for both methods as a function of Δ. Include also the functions $a\Delta^1$ and $b\Delta^2$, where a and b are constants chosen to yield a nice plot. Hint: use a log-log plot.

5. **Accuracy: Part Deux**. Here, we want to investigate the accuracy of the Gauss-Legendre quadrature and hence S_N order.
   (a) Compute $\phi(x)$ analytically for the sample slab problem of Lecture 17, and make this available as a function.
   (b) Compute ϕ analytically using the S_N method for orders 2, 4, 8, 16, 32, and 64. You will have to look up the quadrature parameters for $N > 4$.
   (c) Compute the errors in ϕ as in the last problem for $x = 5.0$ and plot as a function of N.
   (d) Estimate the "p" value as in the last problem and comment.
6. **Reflective Conditions**. Modify the given S_N code to handle reflective conditions, and solve the following problems...
7. **CMR**. Implement CMR in the given S_N code. Test it on the following problems, using a variety of coarse mesh sizes...
8. **CMFD**. Read the paper by Zhong et al. [29] and derive the CMFD equations in 1-d. Implement the method in the S_N code and compare it to CMR for the slab configuration given in the last problem.
9. **Discrete Ordinates Matrix**. The discrete ordinates equations are simply a set of coupled differential equations discretized in space and angle. Consequently, we can express the equations in matrix form. Consider a uniform slab (Σ_t and Σ_s) with a uniform isotropic source of volumetric strength S. Suppose we discretize this slab into three cells of equal width Δ. Using an S_2 approximation with the DD method, and assuming vacuum conditions:
   (a) Cast the sweep-based source iteration scheme in matrix form, expressing the right hand vector in terms of Q, the emission density.
   (b) Reformulate the equations so that the source term is brought to the left hand side, thus enabling a direct solution of the equations without source iteration.

   For $\Sigma_t = 1.0$, $\Sigma_s = 0.5$, $\Delta = 1.0$, and $S = 1.0$, verify your expressions from parts (a) and (b). For more on part (b), see the paper by Patton and Holloway [20].

Lecture 21 | $\mathrm{P_N}$ Method and Diffusion

We continue in this lecture our treatment of the integro-differential form of the transport equation. Here, we turn to expansion of the angular variable in an orthogonal basis, the well-known Legendre polynomials in slab geometry and the spherical harmonics in multidimensional settings. This approach is typically referred to generically as the P_N method, which is the common symbol for the Legendre polynomials. We'll first start with a brief discussion of anisotropic scattering and how it can be handled using orthogonal expansions. We'll then derive the P_N equations for slab geometry, show how one formally arrives at the *neutron diffusion equation*, and provide a simple code that illustrates some concepts.

Anisotropic Scattering

In the past several lectures, we presented equations appropriate for scattering that is isotropic in the laboratory system. While it is often appropriate to assume isotropic scattering in the center of mass system (i.e. s-wave scattering), this corresponds to a forward peaked distribution in the laboratory system. This is especially true for light scatterers, which are important moderators. Since the neutron transport equation itself lives in the laboratory system, to account even for isotropic center of mass scattering requires we devise a way to include angular dependence in our scattering cross-section.

Recall that the macroscopic scattering cross-section can be expressed by $\Sigma_s(\hat{\Omega}' \to \hat{\Omega})$, which roughly quantifies the probability that a neutron going in direction $\hat{\Omega}'$ will collide and end up in the direction $\hat{\Omega}$. However, if the medium is isotropic, recall that this probability depends only on the cosine of the angle between the two direction vectors, i.e

$$\Sigma_s(\hat{\Omega}' \to \hat{\Omega}) = \Sigma_s(\hat{\Omega}' \cdot \hat{\Omega}) = \Sigma_s(\mu_0)\,, \tag{21.1}$$

where μ_0 is the cosine of the scattering angle. Note, this is not valid for a moving medium (perhaps a molten, flowing fuel) or single crystals.

The expression $\Sigma_s(\mu_0)$ is often expressed as an expansion in Legendre polynomials. This not only provides a satisfactory way to represent experimental data points in function form, but it also facilitates many analytic and numerical treatments of anisotropic scattering. Using the Legendre polynomials $P_l(\mu_0)$, we write

$$\Sigma_s(\mu_0) = \sum_{l=0}^{\infty} \frac{2l+1}{4\pi} \Sigma_{sl} P_l(\mu_0)\,, \tag{21.2}$$

where the Legendre coefficients (or *Legendre momements*) are defined

$$\Sigma_{sl} = 2\pi \int_{-1}^{1} \Sigma_s(\mu_0) P_l(\mu_0) d\mu_0\,. \tag{21.3}$$

Legendre Polynomials

So what are the Legendre polynomials? There are many ways to express them, including as the solution of the Legendre differential equation,

$$(1-x^2)\frac{d^2 P_l}{dx^2} - 2x\frac{dP_l}{dx} + l(l+1)P_l(x) = 0\,. \tag{21.4}$$

This equation is related to Laplace's equation in spherical coordinates, for which full solutions take the form of the *spherical harmonics*. Additionally, the Legendre polynomials are orthogonal over $-1 < x < 1$, satisfying

$$\int_{-1}^{1} P_n(x) P_m(x) dx = \frac{2}{2n+1}\delta_{mn}\,, \tag{21.5}$$

where δ_{mn} is the Kronecker delta. In our derivation of the P_n equations, we'll use the Legendre recurrence relation,

$$(l+1)P_{l+1}(x) - (2l+1)xP_l(x) + lP_{l-1}(x) = 0\,. \tag{21.6}$$

For reference, the Legendre polynomials through $N = 5$ are given in Figure 21.1, and are defined

$$\begin{aligned}
P_0(x) &= 1 \\
P_1(x) &= x \\
P_2(x) &= \frac{1}{2}(3x^2 - 1) \\
P_3(x) &= \frac{1}{2}(5x^3 - 3x) \\
P_4(x) &= \frac{1}{8}(35x^4 - 30x + 3) \\
P_5(x) &= \frac{1}{8}(64x^5 - 70x^3 + 15x)\,.
\end{aligned} \tag{21.7}$$

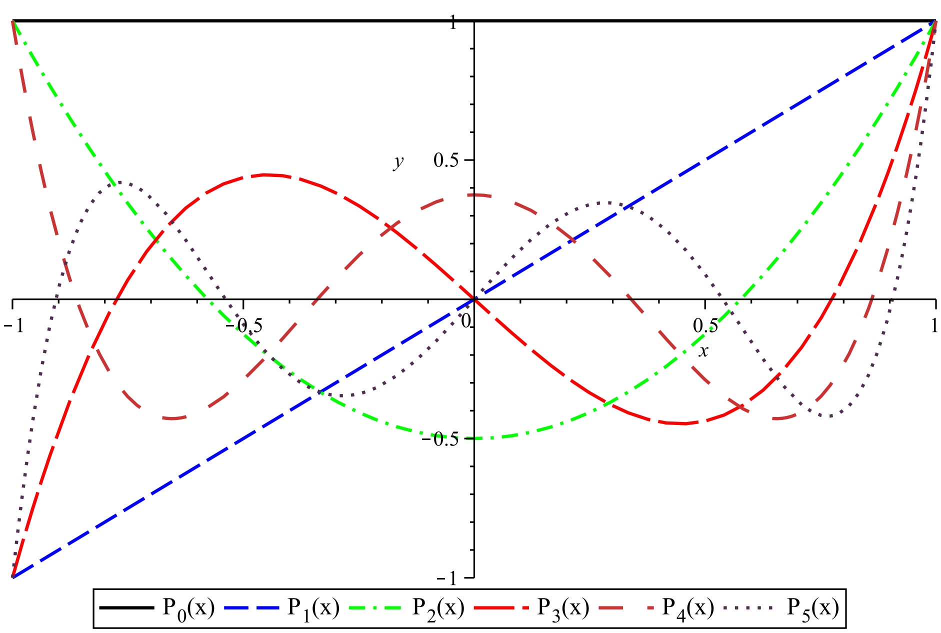

Figure 21.1: The first several Legendre polynomials, $P_N(x)$.

The Legendre polynomials are useful for expanding the scattering cross-section because they are orthogonal over the range $-1 \leq \mu \leq 1$. Moreover, expansions in the Legendre polynomials are unique in that the zeroth moment Σ_{s0} preserves the integral properties of the underlying distribution (i.e the mean), whereas the other moments merely provide shape. By looking at the P_N in Figure 21.1, this becomes more obvious when one notes only P_0 has a non-vanishing integral over the domain.

Expanding the Angular Dependence

To derive the P_N equations, we must be careful to choose a starting point consistent with our expansion of the scattering kernel defined by Eqs. 21.2 and 21.3. If we're not, it's easy to encounter a phantom 2π! Take as our starting point the monoenergetic transport equation in slab geometry with arbitary angular dependence of the scattering and source terms,

$$\mu \frac{\partial \psi}{\partial x} + \Sigma_t(x)\psi(x,\mu) = \int_0^{2\pi} d\phi' \int_{-1}^{1} d\mu' \Sigma_s(x,\mu_0)\psi(x,\mu') + S(x,\mu). \quad (21.8)$$

We expand the source

$$S(x,\mu) = \sum_{n=0}^{\infty} \frac{2n+1}{4\pi} S_n(x) P_n(\mu)\,, \tag{21.9}$$

where

$$S_n(x) = 2\pi \int_{-1}^{1} S(x,\mu) P_n(\mu) d\mu\,, \tag{21.10}$$

and the angular flux

$$\psi(x,\mu) = \sum_{n=0}^{\infty} \frac{2n+1}{4\pi} \psi_n(x) P_n(\mu)\,, \tag{21.11}$$

where

$$\psi_n(x) = 2\pi \int_{-1}^{1} \psi(x,\mu) P_n(\mu) d\mu\,. \tag{21.12}$$

Before substituting these expressions, it helps to simplify the scattering expression. Using Eq. 21.2, we can write the scattering term of Eq. 21.8

$$\begin{aligned} &\int_0^{2\pi} d\phi' \int_{-1}^{1} d\mu' \Sigma_s(x,\mu_0) \psi(x,\mu') \\ &\qquad = \int_0^{2\pi} d\phi' \int_{-1}^{1} d\mu' \sum_{l=0}^{\infty} \frac{2l+1}{4\pi} \Sigma_{sl}(x) P_l(\mu_0) \psi(x,\mu')\,. \end{aligned} \tag{21.13}$$

Here, we make use of the *Legendre addition theorem*, which states

$$P_l(\mu_0) = P_l(\mu)P_l(\mu') + 2\sum_{m=1}^{l} \frac{(l-m)!}{(l+m)!} P_l^m(\mu) P_l^m(\mu') \cos(m(\phi-\phi'))\,, \tag{21.14}$$

where $P_l^m(\mu)$ are the *associated Legendre polynomials*, defined

$$P_l^m(\mu) = \sqrt{(1-\mu^2)^m} \frac{d^m P_l}{d\mu^m}\,. \tag{21.15}$$

Substituting this in, we find

$$\begin{aligned} &\int_0^{2\pi} d\phi' \int_{-1}^{1} d\mu' \sum_{l=0}^{\infty} \frac{2l+1}{4\pi} \Sigma_{sl}(x) P_l(\mu_0) \psi(x,\mu') \\ &\quad = \sum_{l=0}^{\infty} \frac{2l+1}{4\pi} \Sigma_{sl}(x) \int_{-1}^{1} d\mu' \psi(x,\mu') \int_0^{2\pi} d\phi' \Bigg(P_l(\mu)P_l(\mu') + \\ &\qquad\qquad 2\sum_{m=1}^{l} \frac{(l-m)!}{(l+m)!} P_l^m(\mu) P_l^m(\mu') \cos(m(\phi-\phi')) \Bigg)\,. \end{aligned} \tag{21.16}$$

Noting that $\int_0^{2\pi} \cos(m(\phi - \phi'))d\phi' = 0$, we find

$$\begin{aligned}
&\int_0^{2\pi} d\phi' \int_{-1}^{1} d\mu' \sum_{l=0}^{\infty} \frac{2l+1}{4\pi} \Sigma_{sl}(x) P_l(\mu_0) \psi(x, \mu') \\
&\quad = \sum_{l=0}^{\infty} \frac{2l+1}{4\pi} \Sigma_{sl}(x) \int_{-1}^{1} d\mu' \psi(x, \mu') \int_0^{2\pi} d\phi' \Big(P_l(\mu) P_l(\mu') \Big) \\
&\quad = \sum_{l=0}^{\infty} \frac{2l+1}{4\pi} \Sigma_{sl}(x) P_l(\mu) \left(2\pi \int_{-1}^{1} d\mu' \psi(x, \mu') P_l(\mu') \right) \\
&\quad = \sum_{l=0}^{\infty} \frac{2l+1}{4\pi} \Sigma_{sl}(x) P_l(\mu) \psi_l(x) \,.
\end{aligned} \tag{21.17}$$

Substituting the simplified scattering term of Eq. 21.17 and the expansions of Eqs. 21.9 and 21.11 into Eq. 21.8 yields

$$\begin{aligned}
&\sum_{n=0}^{\infty} \frac{2n+1}{4\pi} P_n(\mu) \Big(\mu \frac{\partial \psi_n(x)}{\partial x} + \Sigma_t(x) \psi_n(x) \Big) = \\
&\qquad \sum_{n=0}^{\infty} \frac{2n+1}{4\pi} \Sigma_{sn}(x) P_n(\mu) \psi_n(x) + \sum_{n=0}^{\infty} \frac{2n+1}{4\pi} S_n(x) P_n(\mu) \,.
\end{aligned} \tag{21.18}$$

By rearranging the Legendre recurrence relation of Eq. 21.6, we can express the product $\mu P_n(\mu)$ as

$$\mu P_n(\mu) = \frac{1}{2n+1} \Big((n+1) P_{n+1}(\mu) + n P_{n-1}(\mu) \Big) \,. \tag{21.19}$$

Substituting this in for μP_n in the first term on the left of Eq. 21.18 and cancelling a few like terms yields

$$\begin{aligned}
&\sum_{n=0}^{\infty} \left(\Big((n+1) P_{n+1}(\mu) + n P_{n-1}(\mu) \Big) \frac{\partial \psi_n(x)}{\partial x} + (2n+1) P_n(\mu) \Sigma_t(x) \psi_n(x) \right) = \\
&\qquad \sum_{n=0}^{\infty} (2n+1) \Sigma_{sn}(x) P_n(\mu) \psi_n(x) + \sum_{n=0}^{\infty} (2n+1) S_n(x) P_n(\mu) \,.
\end{aligned} \tag{21.20}$$

To exploit the orthogonality property defined in Eq. 21.5, we multiply both sides of

Eq. 21.20 by $\frac{2m+1}{2}P_m$ and integrate the result over $-1 \leq \mu \leq 1$. Then we find

$$\int_{-1}^{1} d\mu \left\{ \sum_{n=0}^{\infty} \frac{2m+1}{2} P_m(\mu) \left(\overbrace{(n+1)P_{n+1}(\mu)\frac{\partial \psi_n}{\partial x}}^{\delta_{n+1,m} \to n=m-1} + \overbrace{nP_{n-1}(\mu)\frac{\partial \psi_n}{\partial x}}^{\delta_{n-1,m} \to n=m+1} \right) \right\} = m\frac{\partial \psi_{m-1}}{\partial x} + (m+1)\frac{\partial \psi_{m+1}}{\partial x}\,, \tag{21.21}$$

and

$$\int_{-1}^{1} d\mu \frac{2m+1}{2} P_m(\mu)(2n+1)P_n(\mu)\Sigma_t(x)\psi_n(x) = (2m+1)\Sigma_t(x)\psi_m(x)\,, \tag{21.22}$$

and similarly for the scattering and source terms. Together, we have

$$\begin{aligned} \frac{m+1}{2m+1}\frac{\partial \psi_{m+1}}{\partial x} + \frac{m}{2m+1}\frac{\partial \psi_{m-1}}{\partial x} + \Sigma_t(x)\psi_m(x) = \\ \Sigma_{sm}(x)\psi_m(x) + S_m(x)\,,\ m = 0 \ldots \infty\,, \end{aligned} \tag{21.23}$$

where we take $\psi_{-1} = 0$.

The P_N Equations

Eq. 21.23 is an infinite set of coupled differential equations that represents an exact treatment in angle. To yield a tractable problem, we make the P_N approximation:

$$\frac{d\psi_{N+1}}{dx} = 0\,. \tag{21.24}$$

As an example, the P_1 approximation yields

$$\begin{aligned} n = 0: \quad & \frac{d\psi_1}{dx} + \Sigma_t(x)\psi_0(x) = \Sigma_{s0}\psi_0 + S_0 \\ n = 1: \quad & \frac{1}{3}\frac{d\psi_0}{dx} + \Sigma_t(x)\psi_1(x) = \Sigma_{s1}\psi_1 + S_1\,, \end{aligned} \tag{21.25}$$

which is a set of two equations in the two unknown Legendre moments ψ_0 and ψ_1. In general, the P_N approximation gives $N+1$ equations in an equal number of unknowns.

An interesting fact is that the P_N approximation is equalivalent to the S_{N+1} approximation in slab geometry if the Gauss-Legendre quadrature set is used, proof of which is left as an exercise for the simple case of the P_1 and S_2 equations. A similar equivalence can me made in certain cases for higher dimensions between the spherical harmonics equations and multidimensional S_N approximations. Doing so is one way to mitigate the numerical artifacts known as ray effects common to the S_N approximation. See the references in the Further Reading.

Diffusion Theory

The P_N equations give us one formal way to derive the diffusion equation. Suppose in Eq. 21.25 that we take the source to be isotropic so that $S_1 = 0$. Moreover, we define the first scattering momement to be $\Sigma_{s1} = \Sigma_{s0}\bar{\mu}$, where $\bar{\mu} = \int_{-1}^{1} d\mu\mu\Sigma_s(\mu) / \int_{-1}^{1} d\mu\Sigma_s(\mu)$.

Then, we rearrange Eq. 21.25 to find

$$\psi_1(x) = \frac{-1}{3(\Sigma_t - \Sigma_{s0}\mu)} \frac{d\psi_0}{dx} \equiv -D(x)\frac{\psi_0}{dx} \,, \tag{21.26}$$

so that

$$-\frac{d}{dx}D(x)\frac{\psi_0}{dx} + (\Sigma_t - \Sigma_{s0})\psi_0(x) = S_0(x) \,. \tag{21.27}$$

This is the diffusion equation, and we recognize $\psi_0(x)$ as the scalar flux $\phi(x)$, $\psi_1(x)$ as the current density $J(x)$, and D as the diffusion coefficient. Note that while diffusion theory explicitly assumes an isotropic source, it does treat linearly anisotropic scattering exactly.

Boundary Condition

In general, there are two approaches to defining boundary conditions for the P_N (or diffusion) equations:

1. conserve net current (Marshak)

2. obtain correct boundary flux for a finite number of μ values (Mark)

The P_N approximation consists of $N + 1$ equations, which requires $N + 1$ boundary conditions. Typically, it's desirable to spread these conditions evenly among surfaces. For a slab system, there are just two surfaces, and for odd N, there are an even number of conditions to be satisfied. Hence, we typically only use odd N approximations.

Marshak Conditions

The *Marshak* condition places a limit on the odd moments of the P_N expansion, as the odd moments drive net flows in angular space. Suppose we are given an incident boundary condition $B_L(\mu)$ on the left hand side, represented as

$$\psi(x_L, \mu) = B_L(\mu)\,, \quad \mu > 0\,. \tag{21.28}$$

Then the Marshak condition states

$$\int_0^1 \psi(x_L, \mu) P_l(\mu) d\mu = \int_0^1 B_L(\mu) P_l(\mu) d\mu\,, \tag{21.29}$$

for $l = 1,\, 3,\, \ldots, N$. In general, the Marshak conditions give the best results for the P_N equations. Moreover, they capture the physically appealing notion of conservation of particles across the boundary.

Mark Condition

The Mark condition can be expressed as

$$\psi_l(x_L) = \int_{-1}^1 B_L(\mu)\delta(\mu - \mu_n) P_l(\mu) d\mu = B(\mu_n) P_l(\mu_n)\,, \tag{21.30}$$

for a desired set of μ_n. Typically, these μ_n are chosen to be symmetric for an equal number of μ_n per half-space. A common choice are the μ_n such that $P_{N+1}(\mu_n) = 0$, which should look strikingly similar to the condition for generating abscissa for the Gauss-Legendre quadrature set.

A Simple Code

To be continued.

Further Reading

To be continued.

Exercises

1. $\mathbf{P_1}$ **Boundary Conditions**. Derive the P_1 equations for isotropic scattering and an isotropic source. Derive the Marshak vacuum conditions for arbitrary left and right boundaries in a slab. Do the same for the Mark conditions.

2. **$\mathbf{P_2}$ Equations**. Show for the case of linearly anisotropic scattering and isotropic source that the P_2 equations can be written as a second order partial differential equation similar in form to the typical neutron diffusion equation.
3. **Numerical Solution of the $\mathbf{P_1}$ Equations**. Consider a slab of width 10 cm with $\Sigma_t = 1.0$ [cm^{-1}], and $c = \Sigma_s/\Sigma_t = 0.5$ (isotropic scattering in the lab system). A uniform, isotropic source of 1 [n/cm^2-s] is located in the first half of the slab, and both slab edges are subject to vacuum conditions. Write a code to solve the P_1 approximation to this problem using Marshak conditions. Plot $\psi(x, \mu)$ at $x = 0$, 2.5, 5.0, 7.5, and 10 [cm]. Plot $\phi(x)$ over the whole slab.
4. **Numerical Solution of the $\mathbf{P_3}$ Equations**. For the same problem, write a code to solve the P_3 approximation using Marshak conditions.
5. **Diffusion via asymptotics**. Consider the following rescaling of the 1-d, mono-energetic transport equation with isotropic scattering. Finish me.
6. **Legendre Addition Theorem**. Prove the Legendre polynomial addition theorem. You may use any resource you want, but make sure you understand all steps of the proof. A particularly straightforward approach begins as follows. Start with an expansion of an arbitrary function in the full spherical harmonics. Then, substitute the definition of the expansion coefficients back into the expansion. Noting that the integral and summation can be switched, what function must the summation be?
7. **Defining the Scattering Angle**. Prove $\mu_0 = \cos(\theta)\cos(\theta') + \sin(\theta)\sin(\theta')\cos(\phi - \phi')$, where μ_0 is the cosine of the scattering angle and (θ,ϕ) and (θ', ϕ') are the original and final angles, respectively. Include a diagram.

Lecture 22 | Linearity of the Transport Equation and Reciprocity Relations

Lecture 23 | The Adjoint and Perturbation Theory

In this lecture, we introduce the *adjoint* form of the transport equation and describe what it represents physically. We then apply the adjoint equation in a useful technique known as *first order* or *linear perturbation theory*. In the next lecture, we make further use of the adjoint in the context of variational approximations.

The Adjoint Function

We define the *inner product* of two functions $\phi(x)$ and $\psi(x)$ to be

$$\langle \phi, \psi \rangle \equiv \int \phi(x)\psi(x)dx \, , \tag{23.1}$$

where ϕ and ψ satisfy appropriate continuity and boundary conditions. A *self-adjoint* operator M satisfies

$$\langle \phi, M\psi \rangle \equiv \langle M\phi, \psi \rangle \, . \tag{23.2}$$

As an example, the operator corresponding to one-speed diffusion can be shown to be self-adjoint, an exercise left to the reader. Note, self-adjoint and *hermitian* are synonymous. You may be familiar with the latter term from quantum physics, and you might recall that such operators (say the Hamiltonian) have real eigenvalues (like the energy) and orthogonal eigenfunctions (such as the nice sines and cosines of the infinite well).

If an operator L is not self-adjoint, it is possible to define an operator L^* that is adjoint to L. Then L^* will operate on adjoint functions $\psi^*(x)$ that may satisfy different boundary conditions than those of $\psi(x)$ on which L operates. The adjoint operator L^* must satisfy

$$\langle \psi^*, L\psi \rangle = \langle \psi, L^*\psi^* \rangle \, , \tag{23.3}$$

which we refer to as the *adjoint identity* (and which is actually a special case of a generalized Green's theorem).

Transport Operator

We now define the transport equation in operator form. Defining the operator

$$\begin{aligned} L\psi \equiv & -\hat{\Omega}\cdot\nabla\psi(\mathbf{r},\mathbf{\Omega},E) \\ & -\Sigma_t\psi + \int_0^\infty dE' \int_{4\pi} d\Omega' \Sigma_s(\mathbf{r},\mathbf{\Omega}\cdot\mathbf{\Omega}',E'\to E)\psi(\mathbf{r},\mathbf{\Omega}',E')\,, \end{aligned} \tag{23.4}$$

the transport equation is simply $L\psi = -Q$, for some source Q; note the sign of the right hand side. The transport operator L is *not self-adjoint*. Convince yourself that this is indeed the case by evaluating the adjoint identity and paying specific attention to the terms corresponding to streaming and scattering. For convenience, we assume ψ is also subject to vacuum conditions on all external surfaces, i.e. $\psi(\mathbf{r},\mathbf{\Omega},E) = 0$ for $\hat{n}\cdot\hat{\Omega} < 0$ where $\hat{n}$ is the outward normal vector.

The adjoint transport operator L^* will satisify $\langle\psi^*, L\psi\rangle = \langle\psi, L^*\psi^*\rangle$, if and only if we define it such that

$$\begin{aligned} L^*\psi^* \equiv & \,\hat{\Omega}\cdot\nabla\psi^*(\mathbf{r},\mathbf{\Omega},E) \\ & -\Sigma_t\psi^* + \int_0^\infty dE' \int_{4\pi} d\Omega' \Sigma_s(\mathbf{r},\mathbf{\Omega}\cdot\mathbf{\Omega}',E\to E')\psi^*(\mathbf{r},\mathbf{\Omega}',E')\,, \end{aligned} \tag{23.5}$$

with the further restriction that $\psi^*(\mathbf{r},\mathbf{\Omega},E) = 0$ for $\hat{n}\cdot\hat{\Omega} > 0$. It's worth noting we could have chosen conditions other than vacuum boundaries for ψ, which would yield different conditions for ψ^*; see the exercises for the case of reflecting boundaries.

Interpreting the Adjoint

Let's consider a subcritical, time-independent system containing an arbitrary source. Suppose we are interested in a detector response with an associated cross-section $\Sigma_d(\mathbf{r},E)$. Then we have the forward equation

$$\begin{aligned} & \hat{\Omega}\cdot\nabla\psi(\mathbf{r},\mathbf{\Omega},E) + \Sigma_t\psi = \\ & \int_0^\infty dE' \int_{4\pi} d\Omega' \Sigma_s(\mathbf{r},\mathbf{\Omega}\cdot\mathbf{\Omega}',E'\to E)\psi(\mathbf{r},\mathbf{\Omega}',E') + Q(\mathbf{r},\mathbf{\Omega},E)\,, \end{aligned} \tag{23.6}$$

subject to vacuum conditions, and the adjoint equation

$$\begin{aligned} & -\hat{\Omega}\cdot\nabla\psi^*(\mathbf{r},\mathbf{\Omega},E) + \Sigma_t\psi^* = \\ & \int_0^\infty dE' \int_{4\pi} d\Omega' \Sigma_s(\mathbf{r},\mathbf{\Omega}\cdot\mathbf{\Omega}',E\to E')\psi^*(\mathbf{r},\mathbf{\Omega}',E')\,, \end{aligned} \tag{23.7}$$

subject to the appropriate conditions. Here, the detector cross-section, which can be thought of as a "detector response function", is the adjoint source. Now, we multiply Eq. 23.6 by ψ^* and Eq. 23.7 by ψ, subtract the latter from the former, and integrate over all variables to get (operator and inner-product notation)

$$\langle \psi^*, L\psi \rangle - \langle \psi, L^*\psi^* \rangle = -\langle \psi^*, Q \rangle + \langle \psi, \Sigma_d \rangle \,, \tag{23.8}$$

but by the adjoint identity, the left hand side vanishes, and we are left with a most important result:

$$\langle \psi^*, Q \rangle = \langle \psi, \Sigma_d \rangle \,. \tag{23.9}$$

Suppose our forward source is a unit monoenergetic, unidirectional point source, i.e. $Q(\mathbf{r}, \mathbf{\Omega}, E) = \delta(\mathbf{r} - \mathbf{r}_0)\delta(\mathbf{\Omega} - \mathbf{\Omega}_0)\delta(E - E_0)$. From Eq. 23.9, we find

$$\psi^*(\mathbf{r}_0, \mathbf{\Omega}_0, E_0) = \int_V d^3r \int_E dE \int_{4\pi} d\Omega \Sigma_d(\mathbf{r}, E')\psi(\mathbf{r}, \mathbf{\Omega}, E) = R \,, \tag{23.10}$$

where R is the total detector response. In this case, $\psi^*(\mathbf{r}_0, \mathbf{\Omega}_0, E_0)$ is the expected contribution to the detector response due to a unit delta source located at $(\mathbf{r}_0, \mathbf{\Omega}_0, E_0)$ in phase space. More broadly, ψ^* represents the importance of neutrons in a particular region of phase space to a given detector response.

Perturbation Theory — Fixed Source

In this and the next section, we apply the adjoint in determining the change to an integral system parameter* due to a small perturbation in the system. We begin with a fixed source example and follow with an eigenvalue example.

Suppose we have a system described by†

$$L_0\psi_0 = Q_0 \tag{23.11}$$

with vacuum boundaries. Our goal is to evaluate a detector response that takes the form

$$R_0 = \int\int\int d^3r d\Omega dE \Sigma_d(\mathbf{r}, E) Q(\mathbf{r}, E, \mathbf{\Omega}) \,. \tag{23.12}$$

Suppose we now introduce some small change. Perturbation theory can allow us to determine the detector response, to first order accuracy, for a "small" perturbation to the system. Let us define a new, perturbed system to be

$$L\psi = Q \,, \tag{23.13}$$

*Here, an "integral" parameter is any integrated quantity, e.g. a reaction rate in a volume, or k_{eff} in a reactor.

†Note, the operators in this section have absorbed the minus sign of the right hand side sources above.

also subject to vacuum conditions, and where $L = L_0 + \delta L_0$, $Q = Q_0 + \delta Q_0$, and $\psi = \psi_0 + \delta\psi_0$. We assume we know L and Q (we're making the perturbation), and we want to know ψ (and eventually, its effect on R). Rewriting the system, we have

$$\begin{aligned}(L_0 + \delta L_0)(\psi_0 + \delta\psi_0) &= Q_0 + \delta Q_0 \\ L_0\psi_0 + \delta L_0\psi_0 + L_0\delta\psi_0 + \theta(\delta^2) &= Q_0 + \delta Q_0\,,\end{aligned} \tag{23.14}$$

but noting our original equation is contained, we are left with two separate equations,

$$L_0\psi_0 = Q_0\,, \tag{23.15}$$

and

$$\delta L_0\psi_0 + L_0\delta\psi_0 = \delta Q_0\,, \tag{23.16}$$

where we have neglected terms of order δ^2; hence the theory is known as "first order" or "linear" perturbation theory.

We now introduce the adjoint to the *original equation*, with the adjoint source being our detector response function Σ_d, i.e.

$$L_0^*\psi_0^* = \Sigma_d\,, \tag{23.17}$$

and as before, we have $\langle\psi_0^*, Q_0\rangle = \langle\psi_0, \Sigma_d\rangle = R_0$. Our goal is to determine δR_0, or $R - R_0 = \delta R_0 = \langle\delta\psi_0, \Sigma_d\rangle$. To do so, we multiply Eq. 23.16 by ψ_0^* and Eq. 23.17 by $\delta\psi_0$, subtract the latter from the former, and integrate over all phase space, yielding

$$\langle\psi_0^*, \delta L_0\psi_0\rangle + \langle\psi_0^*, L_0\delta\psi_0\rangle - \langle L_0^*\psi_0^*, \delta\psi_0\rangle = \langle\psi_0^*, \delta Q_0\rangle - \langle\delta\psi_0, \Sigma_d\rangle\,. \tag{23.18}$$

The second and third terms on the left hand side cancel by way of the adjoint identity, and the second term on the right hand side is δR_0. Thus, we have

$$\delta R_0 = \langle\psi_0^*, \delta Q_0\rangle - \langle\psi_0^*, \delta L_0\psi_0\rangle\,. \tag{23.19}$$

From Eq. 23.19, we see that an increased source gives rise to a *greater* response, and an increase in L, corresponding to greater leakage or interaction, produces a *lower* response, as is expected.

Perturbation Theory — Eigenvalue

As another example of perturbation theory, consider the unperturbed eigenvalue problem

$$H_0\psi_0 = \lambda_0 F_0\psi_0 \tag{23.20}$$

subject to vacuum conditions. As above, suppose we introduce some small change, and let a new, perturbed system be

$$H\psi = \lambda F\psi\,. \tag{23.21}$$

where $H = H_0 + \delta H_0$, $F = F_0 + \delta F_0$, $\psi = \psi_0 + \delta\psi_0$, and $\lambda = \lambda_0 + \delta\lambda_0$. Our goal is to find $\delta\lambda_0$ due to the perturbation.

We rewrite the perturbed system

$$(H_0 + \delta H_0)(\psi_0 + \delta\psi_0) = (\lambda_0 + \delta\lambda_0)(F_0 + \delta F_0)(\psi_0 + \delta\psi_0) \tag{23.22}$$

and expand

$$\begin{aligned} &H_0\psi_0 + \delta H_0\psi_0 + H_0\delta\psi_0 + \theta(\delta^2) \\ &\qquad = \lambda_0 F_0\psi_0 + \lambda_0 F_0\delta\psi_0 + \lambda_0\delta F_0\psi_0 + \delta\lambda_0 F_0\psi_0 + \theta(\delta^2)\,. \end{aligned} \tag{23.23}$$

Again, we recognize our original equation and a second equation with first-order perturbations,

$$(H_0 - \lambda_0 F_0)\delta\psi_0 = (\lambda_0\delta F_0\psi_0) - (\delta H_0 - \delta\lambda_0 F_0)\psi_0\,. \tag{23.24}$$

We define the adjoint problem

$$H_0^*\psi_0^* = \lambda_0 F_0^*\psi_0^* \tag{23.25}$$

subject to the appropriate boundary conditions. Similar to our treatment in the fixed source example, we multiply Eq. 23.24 by ψ_0^* and Eq. 23.25 by $\delta\psi_0$, subtract the latter from the former, and integrate over all phase space. After a bit of rearranging, we find

$$\delta\lambda_0 = \frac{\langle\psi_0^*, (\delta H_0 - \lambda_0\delta F_0)\psi_0\rangle}{\langle\psi_0^*, F_0\psi_0\rangle}\,. \tag{23.26}$$

Further Reading

Most of the development here follows that of Bell and Glasstone [3], Chapter 6. Duderstadt and Hamilton [9] develop the adjoint within the diffusion framework and apply it to problems of reactor physics in Chapters 5 and 7. A particularly appealing description of the physical interpretation of the adjoint, albeit with a reactor physics flavor, is given by Henry [13].

It's worth noting that the adjoint was developed first by Lagrange as a mathematical construct; however, its physical utility was first realized much later in the context of quantum mechanical perturbation theory, and later yet in reactor physics. This history and more is to be found in Marchuk's treatise on adjoint methods [17]. That first application of the adjoint in reactor physics was due to the "father of nuclear engineering," Eugene Wigner [26].

The available literature on perturbation theory is quite large. One important recent effort has been to couple sensitivities defined by perturbation theory to cross-section uncertainties in order to estimate the uncertainty of integral system parameters including the eigenvalue [4] and various worths [28] due to the underlying data uncertainty.

Exercises

1. **Self-adjointness**. Prove the one-speed diffusion operator, i.e. $L\phi = D\phi_{xx} - \Sigma_a\phi(x) = -S$ is self-adjoint. You may assume a homogeneous medium with zero-flux boundary conditions, neglecting extrapolation distances.
2. **Adjoint Transport Equation**. Demonstrate that the adjoint operator L^* defined by Eq. 23.5 really is the adjoint to the forward operator L for the given boundary conditions.
3. **Adjoint Boundary Conditions**. (a) For the case that ψ satisfies vacuum conditions, we found that $\psi^*(\mathbf{r}, \mathbf{\Omega}, E) = 0$ for $\hat{n} \cdot \hat{\Omega} > 0$. What does this mean physically? (b) For the one-speed transport equation, derive the boundary conditions for ψ^* when ψ satisfies reflecting conditions.
4. **Using the Adjoint**. (a) Briefly describe the physical meaning of the adjoint flux. (b) Suppose we have a known shield with a known detector on one side. Suppose further that the particle source on the opposite side of the shield is not known *a priori* and can take widely varying forms. (An example of this might be a shielding analysis for a fusion reactor, where we think we have a good shield and then we try using it for several possible sources). How could the adjoint be used so that only one "transport" calculation would be needed to compute the detector dose given an arbitrary source?
5. **A Point Detector**. Repeat the process used to obtain $\psi^* = R$ but for the case of a point detector, $\Sigma_d = \delta(\mathbf{r} - \mathbf{r}_0)\delta(\mathbf{\Omega} - \mathbf{\Omega}_0)\delta(E - E_0)$. Clearly, one obtains an expression for $\psi(\mathbf{r}_0, \mathbf{\Omega}_0, E_0)$. What does $\psi^*(\mathbf{r}, \mathbf{\Omega}, E)$ represent in this case? This result is a generalization of reciprocity relations previously discussed for one-speed transport.
6. **The "Contributon" Flux**. In 1-d and one-speed, define the quantity

$$C(x, \mu) = \psi(x, \mu)\psi^*(x, \mu) \,,$$

where ψ and ψ^* are the forward and adjoint angular fluxes. Take the forward problem to have vacuum boundaries. (a) Mathematically, express the vacuum boundary condition for ψ at a general external boundary x_b. (b) With your knowledge of the corresponding adjoint boundary condition, write down the mathematical expression for the boundary condition of $C(x, \mu)$. (c) Taking the adjoint source to be a flux-to-dose factor, what are the units of ψ, ψ^*, and C? Can you interpret C physically? (For more information on this mysterious quantity, see the paper by Williams [27].)

7. **Eigenvalue Perturbation**. Prove Eq. 23.26. Also, describe what possible changes in the system coincide with the perturbations δH_0 and δF_0 and how such changes impact the eigenvalue perturbation. Remember that k_{eff} is λ^{-1}.
8. **Eigenvalue Sensitivity**. Defining the sensitivity of k_{eff} to a cross-section Σ_x to be
$$S_{k,\Sigma_x} = \frac{\Sigma_x}{k_{eff}} \frac{\partial k_{eff}}{\partial \Sigma_x} \,,$$
find an expression for S_{k,Σ_x} in terms of the partial derivatives of H_0 and F_0 with respect to Σ_x.

Lecture 24 | Variational Methods

In this lecture, we investigate use of *variational principles* in the context of nuclear engineering. An entire course could be devoted to this subject; here, we focus on the same two quantities analyzed in Lecture 23, namely the reaction rate for a fixed source problem and the eigenvalue of an eigenvalue problem. We demonstrate that variational principles can be used to estimate these quantities using approximate inputs (i.e. fluxes). We finish by showing how our first order perturbation theory can be derived directly from variational principles.

Variational Principles, Functionals, and Stationary Points

A variational principle casts a particular function, usually a problem's solution, as the stationary point of some *functional*. Often, the functional itself is called the variational principle. A functional is simply a function that takes another function as its argument and returns a scalar as its value. Consider a function $f(x) = Ax + B$. A possible functional would be $F[f(x)] = \int_{x_1}^{x_2} f(x)dx$, which certainly yields a scalar value. Typically, the value of the functional represents a quantity of interest (e.g. a reaction rate).

We see that functionals are quite like functions; how exactly then do we define a *stationary point*, and what does it mean in the context of a variational principle? Recall from elementary calculus that a stationary point is the value of the independent variables such that the function reaches a local extremum (or saddle-point), i.e. the function's derivative (or gradient) vanishes. The same idea applies to functionals. Defining the "weak derivative" of F at a point $f(x)$ in the direction $g(x)$ as

$$\delta F[f, g] = \lim_{\epsilon \to 0} = \frac{F[f + \epsilon g] - F[f]}{\epsilon} \,, \tag{24.1}$$

the *first variation* of F is defined

$$\delta F[f, g] = \left(\frac{d}{d\epsilon} F[f + \epsilon g] \right) \Big|_{\epsilon=0} \,, \tag{24.2}$$

for arbitrary g. When $\delta F[\tilde{f}, g] = 0$ for all g, $\tilde{f}$ is called a stationary point of F, and the expression

$$\delta F[\tilde{f}, g] = 0\,, \quad \forall g \tag{24.3}$$

defines the variational principle for $\tilde{f}$. For $F[f]$ that represents a quantity of interest, $F[\tilde{f}]$ represents the true value for that quantity. Moreover, very near the stationary point, $\delta F \to 0$ by construction, and the errors in F (and hence the quantity of interest) are second order, which gives rise to the utility of variational approximations.*

A Simple, Illustrative Example

It is easiest to understand these ideas through a simple example (unrelated to nuclear engineering). Suppose we wish to find the curve giving us the shortest difference between two points in a plane. Of course, this is intuitive: the solution should be a line. We show this using variational techniques.

Let the curve be $y(x)$; from any calculus book, we know the differential arc length is then

$$dl = \sqrt{dx^2 + dy^2} = \sqrt{1 + (y')^2}\,. \tag{24.4}$$

We take as our functional the arc length,

$$F[y] = \int_{x_1}^{x_2} \sqrt{1 + (y')^2} dx\,, \tag{24.5}$$

where (x_1,y_1) and (x_2,y_2) are the points of interest.

Taking the first variation of F,

$$\begin{aligned}
\delta F[y, g] &= \Big(\frac{d}{d\epsilon}\int_{x_1}^{x_2} \sqrt{1 + (y' + \epsilon g')^2} dx\Big)\Big|_{\epsilon=0} \\
&= \left(\int_{x_1}^{x_2} \frac{(y' + \epsilon g')g'}{\sqrt{1 + (y' + \epsilon g')^2}}\right)\Bigg|_{\epsilon=0} \\
&= \int_{x_1}^{x_2} \frac{(y')g'}{\sqrt{1 + (y')^2}}\,,
\end{aligned} \tag{24.6}$$

we find that the arc length is minimized when

$$\delta F[y, g] = 0 = \int_{x_1}^{x_2} \frac{(y')g'}{\sqrt{1 + (y')^2}}\,. \tag{24.7}$$

*When our interest is a quantity (e.g. a reaction rate) rather than the solution (e.g. a flux), we often refer to $F[f]$ as a variational principle for that quantity of interest; in the next sections, we look at variational principles for the reaction rate and eigenvalue.

By inspection, the variational principle is satisfied for arbitrary g only if $y' = 0$. This relation is known as the *Euler equation*†. Of course, to satisfy $y' = 0$ requires our solution $\tilde{y}(x)$ to be of the form $Ax + B$, i.e. linear, as expected.

A Variational Principle for Fixed Source Problems

Suppose we are interested in a linear functional of the flux, such as $G_{fs}[\psi] = \langle \psi, \Sigma_d \rangle$, a reaction rate. An appropriate variational principal is represented by the *generalized Roussopolos functional*

$$F_{fs}[\psi, \psi^*] = G_{fs}[\psi] + \langle \psi^*, (Q - L\psi) \rangle \, . \tag{24.8}$$

Note, $\delta F_{fs}[\psi, \psi^*] = 0$ is a variational principle for G_{fs} if the corresponding solution ψ yields $F_{fs}[\psi, \psi^*] = G_{fs}[\psi]$. We see this is so when the second term of F vanishes when ψ solves $L\psi = Q$, i.e. when ψ is the solution to the transport equation.

To determine the variational principle, we take the first variation of F_{fs},

$$\begin{aligned} \delta F_{fs}[\phi, \phi^*] &= \left(\frac{d}{d\epsilon} \Big(\langle (\phi + \epsilon \delta\phi) \Sigma_d \rangle + \langle (\phi^* + \epsilon \delta \phi^*), (Q - L(\phi + \epsilon\delta\phi)) \rangle \Big) \right) \Bigg|_{\epsilon = 0} \\ &= \langle \delta\phi, \Sigma_d \rangle - \langle \phi^*, L\delta\phi \rangle + \langle \delta\phi^*, Q \rangle - \langle \delta\phi^*, L\phi \rangle \, . \end{aligned} \tag{24.9}$$

Noting $\langle \phi^*, L\delta\phi \rangle = \langle L^*\phi^*, \delta\phi \rangle$, we have for our principle

$$0 = \langle \delta\phi, (\Sigma_d - L^*\phi^*) \rangle + \langle \delta\phi^*, (Q - L\phi) \rangle \, , \tag{24.10}$$

which is satisfied when $L\phi = Q$ and $L^*\phi^* = \Sigma_d$. These are the corresponding Euler equations, and we see they are just the original forward and adjoint transport equations.

The importance of this variational principle (and others in general) is that it gives us an estimate for G_{fs} accurate to second order for approximate fluxes ϕ and ϕ^*. To demonstrate this, suppose the true solutions to the Euler equations are ψ and ψ^*. Let $\phi = \psi + \delta\psi$ and $\phi^* = \psi^* + \delta\psi^*$. We subsitute these expressions into F and find

$$\begin{aligned} F_{fs}[\phi, \phi^*] &= \langle \Sigma_d, (\psi + \delta\psi) \rangle + \langle (\psi^* + \delta\psi^*), (Q - L(\psi + \delta\psi^*)) \rangle \\ &= \langle \Sigma_d, \psi \rangle + \langle \Sigma_d, \delta\psi \rangle + \langle \psi^*, Q \rangle + \langle \delta\psi, Q \rangle \\ &\quad - \langle \psi^*, L\psi \rangle - \langle \psi^*, L\delta\psi \rangle - \langle \delta\psi^*, L\psi \rangle - \langle \delta\psi^*, L\delta\psi \rangle \\ &= \langle \Sigma_d, \psi \rangle - \langle \delta\psi^*, L\delta\psi \rangle \\ &= G_{fs}[\psi] + \theta(\delta^2) \, . \end{aligned} \tag{24.11}$$

†In general, setting the first variation to zero gives rise to a set of partial differential equations collectively known as the Euler equations.

Hence, F provides a first order accurate (i.e. good through first order) estimate of the reaction rate given approximate forward and adjoint fluxes.

A Variational Principle for Eigenvalue Problems

For eigenvalue problems, an appropriate functional is the *Rayleigh quotient*

$$F_{ev}[\psi, \psi^*] = \frac{\langle \psi^*, L\psi \rangle}{\langle \psi^*, F\psi \rangle} \,. \tag{24.12}$$

Here, the quantity of interest is $G_{fs} = \lambda$, and, unlike the fixed source case, we have $G_{fs} = F_{ev}$. You should show that F_{ev} is in fact a valid expression for λ. The associated Euler equations are just the forward and adjoint eigenvalue equations, which can be shown by setting the first variation of F_{ev} to zero, an exercise left to the reader.

For approximate fluxes ϕ and ϕ, it can be shown that

$$F_{ev}[\phi, \phi^*] = \lambda + \theta(\delta^2) \,, \tag{24.13}$$

the proof of which is left as an exercise.

Perturbation Theory from Variational Principles

In general, it is possible to find first order perturbation estimates using the expression

$$\delta = \bar{F}[\psi, \psi^*] - F[\psi, \psi^*] \,, \tag{24.14}$$

where F is the appropriate functional with nominal operators and $\bar{F}$ is the function evaluated with perturbed operators. It must be stressed that ψ and ψ^* are assumed to be the exact fluxes for the unperturbed problem.

As an example, we re-derive the first order perturbation to a detector response. Suppose our perturbations to our system include $L + \delta L$, $Q + \delta Q$, and $\Sigma_d + \delta\Sigma_d$. Then Eq. 24.14 gives

$$\begin{aligned} \delta_{fs} &= \bar{F}[\psi, \psi^*] - F[\psi, \psi^*] \\ &= \langle \psi, (\Sigma_d + \delta\Sigma_d) \rangle + \langle \psi^*, (Q + \delta Q - (L + \delta L)\psi) \rangle - \langle \psi, \delta\Sigma_d \rangle \\ &= \langle \psi, \delta\Sigma_d \rangle + \langle \psi^*, \delta Q \rangle - \langle \psi^*, \delta L\psi \rangle \,. \end{aligned} \tag{24.15}$$

The last line is equivalent to Eq. 23.19 with the addition of the first term, which explicitly accounts for changes in the detector response function.

Eq. 24.14 can also be applied to the Rayleigh quotient, yielding Eq. 23.26, proof of which is also left as an exercise.

Further Reading

Most of the material presented here is contained in Duderstadt and Martin [10] Chapter 7 in addition to Chapter 1 of Stacey [23]. The former contains several examples and a variational derivation of the diffusion equation. The reader is also encouraged to look up Pomraning's large body of work on variational methods[‡].

Exercises

1. **Second Order Accuracy**. Show that the Roussopoulos functional $F[\phi, \phi^*]$ gives a second order accurate value for $\langle \psi, \Sigma_d \rangle$ when ϕ and ϕ^* are approximate values of ψ and ψ^*, i.e. prove Eq. 24.11.
2. **Fixed Source Perturbation**. Prove Eq. 24.15.
3. **Rayleigh Quotient**. (a) Take the first variation of the Rayleigh quotient with respect to the forward and adjoint fluxes, and show the stationarity conditions are just the forward and adjoint eigenvalue problems. (b) Demonstrate that the Rayleigh quotient is a second order estimator for λ.
4. **Losses-to-Gains**. Directly from the eigenvalue equation, we find
$$\lambda = (L\psi)/(F\psi)\,,$$
i.e. losses-to-gains. Show why this is not a variational principle for λ. Consider again Eq. 24.12. How does the adjoint change the physical interpretation of gains-to-losses?
5. **Eigenvalue Perturbation**. Prove that $\bar{F}_R[\psi, \psi^+] - F_R[\psi, \psi^+]$ yields the first order perturbation estimate for $\delta\lambda$ given by Eq. 23.26.
6. **Applying Roussopolos**. Consider a 1-d, 1-speed diffusion problem in a slab of width $2a$, $\Sigma_a = 0.022$, and $D = 0.14$. (a) Solve for the exact scalar flux, using the conditions $\phi(\pm a) = 0$ (neglect extrapolated boundaries), assuming a uniform source $Q(x) = 1$ and $a = 1$. (b) Compute the total absorption rate in the slab. (c) Now, approximate the solution as $\tilde{\phi}(x) = Ax^2 + Bx + C$ with the same maximum (at $x = 0$) and the same boundary conditions. Compute the absorption rate using $\tilde{\phi}(x)$. (d) Now, recalling the 1-d, 1-speed diffusion equation is self-adjoint, use the Roussopolos principle to compute the absorption rate. What can you conclude?
7. **More Roussopolos** For same problem, estimate the total absorption rate if something causes $\Sigma_a = 1.1$. Be careful to consider the impact of the change in Σ_a on both Σ_d and L.

[‡] It is worth noting that Stacey and Pomraning, both with prolific work in variational methods, did their graduate work in this department.

Part IV

Bibliography and Appendices

Bibliography

[1] M.L. Adams and E.W. Larsen. Fast Iterative Methods for Discrete-Ordinates Particle Transport Calculations. *Progress in Nuclear Energy*, 40(1):3–159, 2002.

[2] G.B. Arfken and H.J. Weber. *Mathematical Methods for Physicists*. Academic Press, 1995.

[3] G.I. Bell and S. Glasstone. *Nuclear Reactor Theory*. Van Nostrand-Reinhold, New York, 1970.

[4] B.L. Broadhead, B.T. Rearden, C.M. Hopper, J.J. Wagschal, and C.V. Parks. Sensitivity-and Uncertainty-Based Criticality Safety Validation Techniques. *Nucl. Sci. Eng*, 146(3):340–366, 2004.

[5] K.M. Case and P.F. Zweifel. *Linear Transport Theory*. Addison-Wesley, Reading,MA, 1967.

[6] G.R. Cefus and E.W. Larsen. Stability Analysis of Coarse-Mesh Rebalance. *Nucl. Sci. Eng*, 105:31, 1990.

[7] S. Chandrasekhar. *Radiative Transfer*. Oxford University Press, 1950.

[8] B. Davison. *Neutron Transport Theory*. Oxford, 1957.

[9] J.J. Duderstadt and L.J. Hamilton. *Nuclear Reactor Analysis*. John Wiley and Sons, Inc., New York, 1976.

[10] J.J. Duderstadt and W.R. Martin. *Transport Theory*. John Wiley and Sons, Inc., New York, 1979.

[11] T.M. Evans, A.S. Stafford, R.N. Slaybaugh, and K.T. Clarno. Denovo: A New Three-Dimensional Parallel Discrete Ordinates Code in SCALE. *Nuclear Technology*, 171(2):171–200, 2010.

[12] A. Hébert. *Applied Reactor Physics*. Presses Internationales Polytechnique, 2009.

[13] A.F. Henry. *Nuclear Reactor Analysis*. MIT Press, Cambridge, MA, 1975.

[14] K.R. Koch, R.S. Baker, and R.E. Alcouffe. Solution of the First-Order Form of the 3-D Discrete Ordinates Equation on a Massively Parallel Processor. *Trans. Amer. Nucl. Soc.*, 65(108):198–199, 1992.

[15] R.J. LeVeque. *Finite Difference Methods for Ordinary and Partial Differential Equations*. SIAM, 2007.

[16] E.E. Lewis and W.F. Miller Jr. *Computational Methods of Neutron Transport*. American Nuclear Society, Lagrange Park, IL, 1993.

[17] G.I. Marchuk. *Adjoint Equations and Analysis of Complex Systems*. Springer, 1995.

[18] W.F. Miller Jr and E.E. Lewis. Nonlinear Response Matrix Methods for Radiative Transfer. In *International Topical Meeting on Advances in Reactor Physics, Mathematics and Computation*, Paris, France, April 1987. American Nuclear Society.

[19] Y.R. Park and N.Z. Cho. Coarse-Mesh Angular Dependent Rebalance Acceleration of the Discrete Ordinates Transport Calculations. *Nuclear Science and Engineering*, 148(3), 2004.

[20] B.W. Patton and J.P. Holloway. Application of Preconditioned GMRES to the Numerical Solution of the Neutron Transport Equation. *Annals of Nuclear Energy*, 29(2):109–136, 2002.

[21] K.S. Smith. *Spatial Homogenization Methods for Light Water Reactor Analysis*. PhD thesis, 1980.

[22] K.S. Smith. Nodal Method Storage Reduction by Nonlinear Iteration. *Trans. Am. Nucl. Soc.*, 44, 1983.

[23] W.M. Stacey. *Variational Methods in Nuclear Reactor Analysis*. Academic Press, 1974.

[24] R.J.J. Stamm'ler and M.J. Abbate. *Methods of Steady-State Reactors Physics in Nuclear Design*. Academic Press, New York, 1983.

[25] J.S. Warsa, T.A. Wareing, and J.E. Morel. Krylov Iterative Methods and the Degraded Effectiveness of Diffusion Synthetic Acceleration for Multidimensional SN Calculations in Problems with Material Discontinuities. *Nuclear Science and Engineering*, 147(3):218–248, 2004.

[26] E.P. Wigner. Effect of Small Perturbations on Pile Period. *Chicago Report CP-G-3048; see also The Collected Works of Eugene Paul Wigner, Part A.*, 1945.

[27] M.L. Williams. Generalized Contributon Response Theory. *Nuclear Science and Engineering*, 108:355–383, 1991.

[28] M.L. Williams. Sensitivity and Uncertainty Analysis for Eigenvalue-Difference Responses. *Nucl. Sci. Eng.*, 155(1):18–36, 2007.

[29] Z. Zhong, T.J. Downar, Y. Xu, M.D. DEHART, and K.T. CLARNO. Implementation of Two-Level Coarse-Mesh Finite Difference Acceleration in an Arbitrary Geometry, Two-Dimensional Discrete Ordinates Transport Method. *Nuclear Science and Engineering*, 158(3):289–298, 2008.

Appendix A

To facilitate understanding of the various terms used throughout the lecture materials, we provide here a list of variables, short definitions, and common units where applicable. In very few cases, symbols are used more than once due to convention (e.g. ϕ for both flux and azimuthal angle). Bold symbols indicate a vector quantity.

Table A.1: Fundamental quantities

Symbol	Description	Units
$\psi(\vec{r}, \hat{\Omega}, E, t)$	angular flux	$\frac{\mathrm{n}}{\mathrm{cm^2{-}s{-}eV{-}ster}}$
$\psi^+(\vec{r}, \hat{\Omega}, E, t)$	adjoint angular flux	$\frac{\mathrm{n}}{\mathrm{cm^2{-}s{-}eV{-}ster}}$
$\phi(\vec{r}, E, t)$	scalar flux	$\frac{\mathrm{n}}{\mathrm{cm^2{-}s{-}eV}}$
$\phi^+(\vec{r}, E, t)$	adjoint scalar flux	$\frac{\mathrm{n}}{\mathrm{cm^2{-}s{-}eV}}$
$\mathbf{j}(\vec{r}, \hat{\Omega}, E, t)$	angular current density	$\frac{\mathrm{n}}{\mathrm{cm^2{-}s{-}eV{-}ster}}$
$\mathbf{J}(\vec{r}, E, t)$	current density	$\frac{\mathrm{n}}{\mathrm{cm^2{-}s{-}eV}}$
$J_{\pm}(\vec{r}, E, t)$	partial current density	$\frac{\mathrm{n}}{\mathrm{cm^2{-}s{-}eV}}$

Appendix B

We covered the P_N equations and how the diffusion equation is formally derived in Lecture 21. In this lecture, we revisit the diffusion equation for slab geometry and develop a straightforward differencing scheme for its numerical solution. We provide a simple code that performs multigroup diffusion for fixed source and eigenvalue (reactor) problems.

The Diffusion Equation

Albedo Conditions

Difference Schemes

Further Reading

Exercises

1. **Boundary Sources**. Modify the albedo condition given in Eq. ?? to include a boundary source.
2. **Diffusion in 2-d**. Derive the difference equations for diffusion in 2-d Cartesian geometry following the procedure used for 1-d above.