Discrete Ordinates

Edward Norris

January 19, 2017

Abstract

The abstract text goes here.

Contents

1 CODE

Consider that we have the following variables, Ψ which is a function of x, y, z, E, and $\hat{\Omega}$. Further, we have the in-flux variables Ψ_{x-in}, Ψ_{y-in}, and $\Psi_{z-in]}$.

Given:

- A list of all energies, $\{E\}$
- A list of all directions in the quadrature, $\{\Omega\}$
- A mapping from r to zone number, $\mathcal{Z}(r)$
- Lists of direction cosines, $\mu(\Omega)$, $\xi(\Omega)$, and $\eta(\Omega)$
- A map of r to voxel volume, $V(r)$
- A mapping of weights for the quadrature, $\omega(\Omega)$
- A mapping of scatter cross sections in zone $\mathcal{Z}(r)$ from energy E' to E, $\Sigma_s(\mathcal{Z}(r), E' \rightarrow E)$
- A list of total cross sections in zone $\mathcal{Z}(r)$ at energy E, $\Sigma_T(\mathcal{Z}(r), E)$
- The surface area of all voxels, $A_{xy}(r)$, $A_{yz}(r)$, $A_{xz}(r)$

```
function sweep()

  # Knowns
  V(r)  # [cm^3] Volume of cell ijk
  A_xy(Ω,i,j), A_yz(Ω,j,k), A_xz(Ω,i,k)  # [cm^2] Area of each face of cell ijk
  S_x(E,r)  # [#] External source
  # Unknowns
  φ(r,E) = 0⃗     # Scalar flux
  Ψ(r,Ω,E) = 0⃗  # Angular flux

  for every E ∈ {E} in descending order
    while not converged
      φ^{i-1}(r) = φ^{i+1}(r)
      S(r) = 0⃗
      for every E' >= E
        for every cell ijk
          S(ijk) = S(ijk) + 1/4πφ(E',ijk) * Σ_s(Z(ijk),E') * V(ijk)
      for every cell ijk
        S(ijk) = S(ijk) + S_x(ijk)
      φ^{i+1} = 0⃗
      for every Ω ∈ Q in any order
        i_0 = f(μ(Ω))
        Δi = f(μ(Ω))
        j_0 = f(ξ(Ω))
        Δj = f(ξ(Ω))
        k_0 = f(η(Ω))
        Δk = f(η(Ω))
        for every cell ijk starting at (i_0,j_0,k_0) in direction <Δi,Δj,Δk>
          if i ∈ ∂
            φ_{x+} = 0
          else
            φ_{x+} = φ_{x-}(i ± 1,j,k)
          if j ∈ ∂
            φ_{y+} = 0
          else
            φ_{y+} = φ_{y-}(i,j ± 1,k)
          if k ∈ ∂
            φ_{z+} = 0
          else
            φ_{z+} = φ_{z-}(i,j,k ± 1)

          n = S(ijk) + A_yz(Ω,j,k)φ_{x+} + A_xz(Ω,i,k)φ_{y+} + A_xy(Ω,i,j)φ_{z+}
          d = V(ijk) * Σ_t(Z(ijk),E) + A_yz(Ω,j,k) + A_xz(Ω,i,k) + A_xy(Ω,i,j)
          Ψ_0 = n/d
          Ψ(E,Ω,ijk) = Ψ_0
          φ_{x-}(ijk) = 2Ψ_0 - φ_{x+}
          φ_{y-}(ijk) = 2Ψ_0 - φ_{y+}
          φ_{z-}(ijk) = 2Ψ_0 - φ_{z+}
          φ^{i+1}(ijk) = φ^{i+1} + ω(Ω)Ψ(e,Ω,ijk)
        for every cell ijk
          φ(E,ijk) = φ^{i+1}(ijk)
        EMIT φ(E,ijk)
      ϵ = max((φ^{i+1} - phi^{i-1})/φ^{i+1})

      for every cell ijk
        φ(E,ijk) = φ^{i+1}(ijk)
  output time to completion
```

```
function octant1()

  for x ∈ X in ascending order
    for y ∈ Y in ascending order
      for z ∈ Z in ascending order

        S = totalScatter(x, y, z, Ω, E)

        # Get the in-flux from the previous out-flux
        Ψ_{x-in}(x, y, z, E, Ω) ← Ψ_{x-out}(x − Δx, y, z, E, Ω)
        Ψ_{y-in}(x, y, z, E, Ω) ← Ψ_{x-out}(x, y − Δy, z, E, Ω)
        Ψ_{z-in}(x, y, z, E, Ω) ← Ψ_{x-out}(x, y, z − Δz, E, Ω)

        # Calculate the angular flux
        n ← S + 2μ(Ω)A_{yz}(y, z)Ψ_{x-in}(x, y, z, E, Ω)+
                2ξ(Ω)A_{xz}(x, z)Ψ_{y-in}(x, y, z, E, Ω)+
                2η(Ω)A_{xy}(x, y)Ψ_{z-in}(x, y, z, E, Ω)
        d ← 2μ(Ω)A_{yz}(y, z) + 2ξ(Ω)A_{xz}(x, z) + 2η(Ω)A_{xy}(x, y) + Σ_T(Z(r), E)
        Ψ(x, y, z, E, Ω) ← n/d

        # Calculate the out-flux
        Ψ_{x-out}(x, y, z, E, Ω) ← 2Ψ(x, y, z, E, Ω) − Ψ_{x-in}(x, y, z, E, Ω)
        Ψ_{y-out}(x, y, z, E, Ω) ← 2Ψ(x, y, z, E, Ω) − Ψ_{y-in}(x, y, z, E, Ω)
        Ψ_{z-out}(x, y, z, E, Ω) ← 2Ψ(x, y, z, E, Ω) − Ψ_{z-in}(x, y, z, E, Ω)

        # Increment the scalar flux
        φ(x, y, z, E) ← φ(x, y, z, E) + ω(Ω)Ψ(x, y, z, E, Ω)
```

```
function totalScatter(r, Ω, E)

  # Assume a point source at r_0 with energy E_0 and strength S_0 particles / sec
  S_x ← 0
  if r = r_0 and E = E_0
    S_x ← (1/4π) × S_0

  S_D ← 0
  for E' ∈ ℰ|E' ≥ E
    S_D = S_D + (1/4π) × φ(r, Ω, E') × Σ_s(𝒵(r), E', E) × V(r)

  S_T = S_x + S_D
  return S_T
```

The indexing scheme is as follows: E, A, x, y, z, l, m

2 Introduction

This document will teach you everything you need to know about the Boltzmann Equation from its derivation to the details of its implementation in computer code.

3 Mathematical Development

3.1 The Boltzmann Equation

The time dependent gamma transport equation can be written as

$$\begin{aligned}&\left[\frac{1}{v(E)}\frac{\partial}{\partial t}+\hat{\Omega}\cdot\nabla+\Sigma_t(\boldsymbol{r},E,t)\right]\psi(\boldsymbol{r},E,\hat{\Omega},t)=\\&\int_{4\pi}\int_0^\infty\Sigma_s(\boldsymbol{r},E'\to E,\hat{\Omega}'\to\hat{\Omega},t)\psi(\boldsymbol{r},E',\hat{\Omega}',t)dE'd\hat{\Omega}'+S(\boldsymbol{r},E,\hat{\Omega},t)\end{aligned}\tag{3.1.1}$$

However, we are typically not concerned with the transient case in medical diagnostic imaging. We are more interested in the steady state case. The time independent form of Eq. 3.1.1 is written as

$$\begin{aligned}&\left[\hat{\Omega}\cdot\nabla+\Sigma_t(\boldsymbol{r},E)\right]\psi(\boldsymbol{r},E,\hat{\Omega})=\\&\int_{4\pi}\int_0^\infty\Sigma_s(\boldsymbol{r},E'\to E,\hat{\Omega}'\to\hat{\Omega})\psi(\boldsymbol{r},E',\hat{\Omega}')dE'd\hat{\Omega}'+S(\boldsymbol{r},E,\hat{\Omega})\end{aligned}\tag{3.1.2}$$

3.2 Harmonic Approximation

Equation 3.1.2 has a number of terms that cannot be solved directly. Instead, some numerical approximation must be used. The macroscopic scattering cross section, Σ_s is typically expanded with a Legendre polynomial (for more information of Legendre polynomials, refer to Section 8.2). The expansion is as follows:

$$\Sigma_s(\boldsymbol{r},E'\to E,\hat{\Omega}'\to\hat{\Omega})\approx\sum_{l=0}^{L}\frac{2l+1}{4\pi}\Sigma_{s,l}(\boldsymbol{r},E'\to E)P_l(\hat{\Omega}'\to\hat{\Omega})\tag{3.2.1}$$

where $\Sigma_{s,l}$ is the expansion coefficients termed the "scattering moments." The Legendre polynomials $P_l(\hat{\Omega}'\to\hat{\Omega}$ are defined as

$$P_l(\hat{\Omega}' \to \hat{\Omega}) = \frac{1}{2l+1} \sum_{m=-l}^{l} Y^*_{l,m}(\hat{\Omega}') Y_{l,m}(\hat{\Omega}) \tag{3.2.2}$$

The angular fluence (ϕ) is also expanded as

$$\phi(\boldsymbol{r}, E', \hat{\Omega}') \approx \sum_{l=0}^{L} \sum_{m=-l}^{l} \phi_{lm}(\boldsymbol{r}, E') Y_{lm}(\hat{\Omega}') \tag{3.2.3}$$

A source term suitable for numeric integration is then

$$\begin{aligned} \int_0^\infty dE' \int_{4\pi} d\hat{\Omega}' \Sigma_s(\boldsymbol{r}, E' \to E, \hat{\Omega}' \to \hat{\Omega}) \phi(\boldsymbol{r}, E', \hat{\Omega}') \approx \\ \sum_{l=0}^{L} \frac{2l+1}{4\pi} \sum_{m=-l}^{l} \Sigma^{gg'}_{s,l} \phi^{g'}_{i,j,k,lm} Y_{l,m}(\hat{\Omega}_n) \end{aligned} \tag{3.2.4}$$

Finally, substituting Eqs. 3.2.4 and ¡¿ into ¡¿, we arrive at

$$\begin{aligned} \hat{\Omega}_n \cdot \nabla \psi^g_{i,j,k,n} + \sigma^g_{i,j,k} \psi^g_{i,j,k,n} = \\ \sum_{g'=0}^{G} \sum_{l=0}^{L} \frac{2l+1}{4\pi} \sum_{m=-l}^{l} \sigma^{g,g'}_{s,l} \psi^{g'}_{i,j,k,l,m} Y_{l,m}(\hat{\Omega}_n) + \frac{1}{4\pi} q^g_{i,j,k} \end{aligned} \tag{3.2.5}$$

We can calculate the scalar flux as

$$\phi^g_{i,j,k} = \sum_{n=1}^{|\Omega|} w_n \psi^g_{i,j,k,n} \tag{3.2.6}$$

3.3 Discretization

The continuous Boltzmann Equation Approximation has to be discretized in all dimensions to run on a computer.

The gradient of ψ is calculated as

$$\nabla \psi^g_{i,j,k,n} \approx \left\langle \frac{\psi^g_{i+1,j,k} - \psi^g_{i-1,j,k}}{\Delta x}, \frac{\psi^g_{i,j+1,k} - \psi^g_{i,j-1,k}}{\Delta y}, \frac{\psi^g_{i,j,k+1} - \psi^g_{i,j,k-1}}{\Delta z} \right\rangle \tag{3.3.1}$$

There are six faces on each parallelpiped with normals $< \pm 1, 0, 0 >$, $< 0, \pm 1, 0 >$, and $< 0, 0, \pm 1 >$. Taking the dot product of these normals and the evaluated gradient gives

$$\hat{\Omega} \cdot \nabla \psi^g_{i,j,k} \approx 2 \left(\frac{\psi^g_{i+1,j,k} - \psi^g_{i-1,j,k}}{\Delta x} + \frac{\psi^g_{i,j+1,k} - \psi^g_{i,j-1,k}}{\Delta y} + \frac{\psi^g_{i,j,k+1} - \psi^g_{i,j,k-1}}{\Delta z} \right) \tag{3.3.2}$$

3.4 Raytracer

Discrete ordinate solutions greatly benefit from the presence of a raytracing algorithm to provide the uncollided flux to the solver.

The raytracer was implemented as...

When the solver is changed from isotropic to anisotropic, the raytracer must be able to produce a Pn expansion needed by the anisotropic solver. We defined the spherical harmonic function as

Figure 1: Simulation Results

$$Y_l^m(\theta,\phi) = \begin{cases} \sqrt{2}K_l^m \cos(m\phi)P_l^m(\cos(\theta)), & m > 0 \\ K_l^0 P_l^0(\cos(\theta)), & m = 0 \\ \sqrt{2}K_l^m \sin(-m\phi)P_l^{-m}(\cos(\theta)), & m < 0 \end{cases} \tag{3.4.1}$$

where K_l^m is defined as

$$K_l^m = \sqrt{\frac{2l+1}{4\pi}\frac{(l-|m|)!}{(l+|m|)!}} \tag{3.4.2}$$

and the associated Legendre functions are computed by the recursion

$$\begin{aligned} P_0^0(x) &= 1 \\ P_m^m(x) &= (2m-1)!!(1-x^2)^{m/2} \\ P_{m+1}^m(x) &= x(2m+1)P_m^m(x) \\ P_l^m(x) &= \frac{x(2l-1)}{l-m}P_{l-1}^m(x) - \frac{l+m-1}{l-m}P_{l-2}^m(x) \end{aligned} \tag{3.4.3}$$

The spherical harmonic expansion is used to approximate a function f by

$$f(\theta,\phi) = \sum_{l=0}^{\infty}\sum_{m=-l}^{l} y_l^m(\theta,\phi)f_l^m \tag{3.4.4}$$

which is exact when the expansion continues to infinity. However, it is trunctated at L yielding

$$f(\theta,\phi) \approx \sum_{l=0}^{L}\sum_{m=-l}^{l} y_l^m(\theta,\phi)f_l^m \tag{3.4.5}$$

However, the value of each f_l^m must be computed before the approximation can be used. These values are computed by

$$f_l^m = \int_{4\pi} y_l^m(\theta,\phi) f(\theta,\phi) d\hat{\Omega} \tag{3.4.6}$$

In the case of the raytracer solution, since *only* the uncollided source is considered, the direction at any point is always precisely known, the particles are traveling away from the source point. Therefore, the function to be approximated is a Dirac delta function in the two dimensional surface of the unit sphere. Equation 3.4.6 can be reduced to

$$f_l^m = \int_{4\pi} y_l^m(\theta,\phi) f(\theta_0,\phi_0) d\hat{\Omega} \tag{3.4.7}$$

The property of the Dirac delta function

$$\int_{-\infty}^{\infty} \delta(x_0) f(x) dx = f(x_0) \tag{3.4.8}$$

can now be used which reduces Equation 3.4.6 to

$$f_l^m = y_l^m(\theta_0,\phi_0) \tag{3.4.9}$$

The moments are related to the angular flux by

$$\phi_l^m(\vec{r},E) = \int Y_l^m(\hat{\Omega}')\psi(\vec{r},\hat{\Omega}',E) d\hat{\Omega}' \tag{3.4.10}$$

which, for the uncollided flux, becomes

$$\phi_l^m(\vec{r},E) = \Psi Y_l^m(\hat{\Omega}_0) \tag{3.4.11}$$

where Ψ is equal to the magnitude of ψ_{unc} in the $\hat{\Omega}_0$ direction.

4 Implementation

At this point, we have successfully developed an algorithm that can be implemented on a computer. However, there are many challenges in actually implementing this algorithm.

4.1 Design

Yeah, it was designed.

4.2 Language Selection

Yeah, it was selected.

4.3 Salome

We chose to use the Salome framework for numerical pre/post-processing. Salome includes a built in geometry system and meshing capabilities.

4.4 Framework Integration

Yeah, it was integrated

4.5 C++ Implementation

Yeah, it was implemented.

```
#include <stdio.h>
#define N 10
/* Block
 * comment */

int main()
{
    int i;

    // Line comment.
    puts("Hello world!");

    for (i = 0; i < N; i++)
    {
        puts("LaTeX is also great for programmers!");
    }

    return 0;
}
```

5 MPI Implementation

MPI stands for Message Passing Interface and is the *de Facto* standard for implementing parallel algorithms across multiple physical machines.

6 GPU Implementation

GPU stands for Graphical Processing Unit and is a collection of SIMD (single instruction multiple data) processors. Each processor operates on its own piece of data but performs the same operation each other processor in its warp is executing.

7 Conclusion

Hooray! It works!

8 Appendix

8.1 Laplace's Equation

Many physical phenomena can be described by Poisson's Equation given below.

$$\nabla^2 f = \psi \tag{8.1.1}$$

Laplace's Equation is a special form of Poisson's Equation and is given by Eq. 8.1.2.

$$\nabla^2 f = 0 \tag{8.1.2}$$

Laplace's equation in cartesian coordinates expands to

Table 1: A list of solutions to Laplace's equation in different geometrical systems

Cartesian	Exponential, Circular (a class of trig function), hyperbolic
Cylindrical	Bessel, Exponential, Circular
Spherical	Legendre polynomial, Power, Circular

$$\nabla^2 f(x,y,z) = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} + \frac{\partial^2 f}{\partial z^2} = 0 \tag{8.1.3}$$

In cylindrical coordinates

$$\nabla^2 f(r,\phi,z) = \frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial f}{\partial r}\right) + \frac{1}{r^2}\frac{\partial^2 f}{\partial \phi^2} + \frac{\partial^2 f}{\partial z^2} \tag{8.1.4}$$

In spherical coordinates

$$\nabla^2 f(\rho,\theta,\phi) = \frac{1}{\rho^2}\frac{\partial}{\partial \rho}\left(\rho^2\frac{\partial f}{\partial \rho}\right) + \frac{1}{\rho^2\sin\theta}\frac{\partial}{\partial \theta}\left(\sin\theta\frac{\partial f}{\partial \theta}\right) + \frac{1}{\rho^2\sin^2\theta}\frac{\partial^2 f}{\partial \phi^2} \tag{8.1.5}$$

Any solution to Laplace's equation is known as a harmonic function. Further, if any two functions are a solution to Laplace's equation, then by the superposition principle, the summation of the two functions is also a solution to Laplace's equation.

Common solutions to Laplace's equation include

8.2 Legendre Polynomials

Legendre polynomials are solutions to

$$P_n(x) = \frac{1}{2\pi i}\oint \frac{1}{\sqrt{1-2\xi x+\xi^2}}\frac{1}{\xi^{n+1}}d\xi \tag{8.2.1}$$

The first few such solutions are
$P_0(x) = 1$
$P_1(x) = x$
$P_2(x) = \frac{1}{2}(3x^2-1)$
$P_3(x) = \frac{1}{2}(5x^3-3x)$
$P_4(x) = \frac{1}{8}(35x^4-30x^2+3)$
$P_5(x) = \frac{1}{8}(63x^5-70x^3+15x)$
$P_6(x) = \frac{1}{16}(231x^6-315x^4+105x^2-5)$

Legencre polynomials are useful for approximating functions truncated at some point. Legendre polynomials are mutually orthogonal which gives them the property

$$f(x) = \sum_{n=0}^{\infty} P_n(x) \tag{8.2.2}$$

where $f(x)$ is any arbitrary function. Often, like a Fourier series which uses sin and cos instead of Legendre functions, the series is cut off at some threshold as an approximation. As an example of this, consider the function $1.75x^3 + 0.6x^2 + .21x - .06$ over the domain $[2,5]$ using the first four Legendre polynomials.

The coefficients are computed

$$k_0 = \int_2^5 P_0(x)f(x)dx = \int_2^5 f(x)dx \tag{8.2.3}$$

$$k_1 = \int_2^5 P_1(x)f(x)dx = \int_2^5 xf(x)dx \tag{8.2.4}$$

$$k_2 = \int_2^5 P_2(x)f(x)dx = \int_2^5 \frac{1}{2}(3x^2-1)f(x)dx \tag{8.2.5}$$

$$k_3 = \int_2^5 P_3(x)f(x)dx = \int_2^5 \frac{1}{2}(5x^3-3x)f(x)dx \tag{8.2.6}$$

$$\tag{8.2.7}$$

These values are then used to reconstruct the original function by

$$f(x) \approx k_0P_0(x) + k_1P_1(x) + k_2P_2(x) + k_3P_3(x) \tag{8.2.8}$$

Associated Legendre polynomials are a generalization of the Legendre polynomial and solve

8.3 Associated Legendre Polynomials

The associated Legendre polynomials are solutions to the equation

$$P_l^m(x) = (-1)^m(1-x^2)^{m/2}\frac{d^mP_l(x)}{dx^m} \tag{8.3.1}$$

The associated Legendre polynomials can be computed recusively by

$$P_m^m(x) = (-1)^m(2m-1)!!(1-x^2)^(m/2) \tag{8.3.2}$$

$$P_{m+1}^m(x) = x(2m+1)P_m^m(x) \tag{8.3.3}$$

$$(l-m)P_l^m(x) = x(2l-1)P_{l-1}^m(x) - (l+m-1)P_{l-2}^m(x) \tag{8.3.4}$$

where $x!!$ is the double factorial function defined by:

$$x!! = \prod_{k=0}^{\lceil x/2\rceil-1} x-2k \tag{8.3.5}$$

For example, $5!! = 5\times3\times1 = 15$ and $6!! = 6\times4\times2 = 48$.

8.4 Spherical Harmonic Function

Combining the associated Legendre polynomials and the sin/cos series, the spherical harmonics equations can be derived. They are orthogonal in both directions.

$$Y_l^m(\theta,\phi) = N_l^{|m|}P_l^{|m|}(\cos\theta)e^{im\phi} \tag{8.4.1}$$

Using Euler's Equation (Eq. 8.4.2) to compute the imaginary exponent, the spherical harmonic can be expanded to Eq. 8.4.3.

$$e^{i\theta} = \cos(\theta) + i\sin(\theta) \tag{8.4.2}$$

$$Y_l^m(\theta,\varphi) = N_l^{|m|}P_l^m(\cos\theta)\left[\cos(m\phi) + i\sin(m\phi)\right] \tag{8.4.3}$$

The normalization constant N_l^m is defined as

$$N_l^m = \sqrt{\frac{2l+1}{4\pi}\frac{(l-m)!}{(l+m)!}} \tag{8.4.4}$$

In many cases, only the real part (y_l^m) of the complex spherical harmonics (Y_l^m) are necessary.

$$y_l^m = \begin{cases} \sqrt{2}N_l^m \cos(m\phi)P_l^m(\cos(\theta)) \\ N_l^0 P_l^0(\cos(\theta)) \\ \sqrt{2}N_l^{|m|} \sin(|m|\phi)P_l^{|m|}(\cos(\theta)) \end{cases} \tag{8.4.5}$$

$$\Sigma_s(\Omega \cdot \Omega') \approx \sum_{l=0}^{L} \sigma_l \sum_{m=-l}^{l} Y_l^m(\Omega)\bar{Y}_l^m(\Omega') \tag{8.4.6}$$

where the over-bar denotes the complex conjugate defined by Eq. 8.4.7.

$$\bar{Y}_l^m(\theta,\varphi) = (-1)^l\sqrt{\frac{2l+1}{4\pi}\frac{(l-m)!}{(l+m)!}}P_l^m(\cos\theta)\left[\cos(m\varphi) - i\sin(m\varphi)\right] \tag{8.4.7}$$

Expanding the inner summation of 8.4.6 with Eq. 8.4.3 and 8.4.7 yields:

$$\begin{aligned} Y_l^m(\theta,\varphi)\bar{Y}_l^m(\theta',\varphi') &= (-1)^{2l}\frac{2l+1}{4\pi}\frac{(l-m)!}{(l+m)!}P_l^m(\cos\theta)P_l^m(\cos\theta')\times \\ &\left[\cos(m\varphi) + i\sin(m\varphi)\right]\left[\cos(m\varphi') - i\sin(m\varphi')\right] \\ &= \frac{2l+1}{4\pi}\frac{(l-m)!}{(l+m)!}P_l^m(\cos\theta)P_l^m(\cos\theta')\times \\ &\left[\cos(m\varphi)\cos(m\varphi') + i\sin(m\varphi)\cos(m\varphi') - i\sin(m\varphi')\cos(m\varphi) - i^2\sin(m\varphi)\sin(m\varphi')\right] \end{aligned} \tag{8.4.8}$$

The real part of the
whose real part can be expanded to

$$Y_l^0 = \sqrt{\frac{2l+1}{4\pi}}P_l^0(1) \tag{8.4.9}$$

$$Y_l^{m,even} = (-1)^m\sqrt{\frac{2l+1}{2\pi}\frac{(l-m)!}{(l+m)!}}P_l^m(\cos(m\phi)) \tag{8.4.10}$$

$$Y_l^{m,odd} = (-1)^m\sqrt{\frac{2l+1}{2\pi}\frac{(l-m)!}{(l+m)!}}P_l^m(\sin(m\phi)) \tag{8.4.11}$$

Some useful recurrence relations used to calculate the Associated Legendre Polynomials are:

$$P_l^l(x) = (-1)^l(2l-1)!!(1-x^2)^(l/2) \tag{8.4.12}$$

$$P_{l+1}^l(x) = x(2l+1)P_l^l(x) \tag{8.4.13}$$

$$P_l^l(x) = (-1)^l(2l-1)!!(1-x^2)^(l/2) \tag{8.4.14}$$

where $x!!$ is the double factorial function defined by:

$$x!! = \prod_{k=0}^{\lceil x/2 \rceil - 1} x - 2k \tag{8.4.15}$$

For example, $5!! = 5 \times 3 \times 1 = 15$.

8.5 Klein-Nishina Formula

The Klein-Nishina formula gives the differential scattering cross section of photon incident on a single free electron.

$$\frac{\partial \sigma}{\partial \Omega} = \frac{1}{2}\alpha^2 r_c^2 P(E_\gamma, \theta)^2 \left(P(E_\gamma, \theta) + P(E_\gamma, \theta)^{-1} - 1 + \cos^2(\theta) \right) \tag{8.5.1}$$

$$P(E_\gamma, \theta) = \frac{1}{1 + (\frac{E_\gamma}{m_e c^2})(1 - \cos(\theta))} \tag{8.5.2}$$

where α is the fine structure constant ($\alpha \approx 7.2971E-3$), r_c is the reduced Compton wavelength ($r_c = \hbar$)

8.6 Numeric Integration

$$\tilde{H} \tag{8.6.1}$$