2D RTE Volume Integral Solver

LaTeX by Zhongming Qu

January 20, 2014

Contents

1 Intro: From RTE to VIE

1.1 RTE

This paper follows the notations of Liemert. The radiative transport equation (RTE) reads

$$\hat{\boldsymbol{s}} \cdot \nabla \psi(\boldsymbol{r}, \hat{\boldsymbol{s}}) + \mu_t(\boldsymbol{r})\psi(\boldsymbol{r}, \hat{\boldsymbol{s}}) = \mu_s(\boldsymbol{r}) \int d\hat{\boldsymbol{s}}' f(\hat{\boldsymbol{s}} \cdot \hat{\boldsymbol{s}}')\psi(\boldsymbol{r}', \hat{\boldsymbol{s}}') + q(\boldsymbol{r}, \hat{\boldsymbol{s}}) \tag{1.1.1}$$

with

$\boldsymbol{r}$	$[L]^{+1}$	position vectorsymtor
$\hat{\boldsymbol{s}}$	$[L]^{0}$	unit direction vectorsymtor
$\psi(\boldsymbol{r}, \hat{\boldsymbol{s}})$	$[L]^{0}$	de-dimensionalized radiance
$q(\boldsymbol{r}, \hat{\boldsymbol{s}})$	$[L]^{-1}$	source term corresponding to the de-dimensionalized radiance
$f(\hat{\boldsymbol{s}} \cdot \hat{\boldsymbol{s}}')$	$[L]^{0}$	scattering phase function as a function of $\hat{\boldsymbol{s}} \cdot \hat{\boldsymbol{s}}'$, $f_m = g^{|m|}$
$\mu_s(\boldsymbol{r})$	$[L]^{-1}$	scattering cross-section, independent of $\hat{\boldsymbol{s}}$
$\mu_a(\boldsymbol{r})$	$[L]^{-1}$	absorption cross-section, independent of $\hat{\boldsymbol{s}}$
$\mu_t = \mu_s + \mu_a$	$[L]^{-1}$	total cross-section, independent of $\hat{\boldsymbol{s}}$

1.2 VIE

For a single scatterer, the RTE can transform into the following volume integral equation (VIE)

$$\begin{aligned}\psi(\boldsymbol{r}, \hat{\boldsymbol{s}}) + \int d\boldsymbol{r}' d\hat{\boldsymbol{s}}' g(\boldsymbol{r}, \hat{\boldsymbol{s}}; \boldsymbol{r}', \hat{\boldsymbol{s}}')\mu_t(\boldsymbol{r}')\psi(\boldsymbol{r}', \hat{\boldsymbol{s}}') \\ - \int d\boldsymbol{r}' d\hat{\boldsymbol{s}}' g(\boldsymbol{r}, \hat{\boldsymbol{s}}; \boldsymbol{r}', \hat{\boldsymbol{s}}')\mu_s(\boldsymbol{r}') \int d\hat{\boldsymbol{s}}'' f(\hat{\boldsymbol{s}} \cdot \hat{\boldsymbol{s}}'')\psi(\boldsymbol{r}', \hat{\boldsymbol{s}}'') \\ = \int d\boldsymbol{r}' d\hat{\boldsymbol{s}}' g(\boldsymbol{r}, \hat{\boldsymbol{s}}; \boldsymbol{r}', \hat{\boldsymbol{s}}')q(\boldsymbol{r}', \hat{\boldsymbol{s}}')\end{aligned} \tag{1.2.1}$$

where $g(\boldsymbol{r}, \hat{\boldsymbol{s}}; \boldsymbol{r}', \hat{\boldsymbol{s}}')$ is the free space, i.e., $\mu_s = \mu_a = 0$, Green's function. Define two linear functionals:

$$\psi^I[q](\boldsymbol{r}, \hat{\boldsymbol{s}}) = \int d\boldsymbol{r}' d\hat{\boldsymbol{s}}' g(\boldsymbol{r}, \hat{\boldsymbol{s}}; \boldsymbol{r}', \hat{\boldsymbol{s}}')q(\boldsymbol{r}', \hat{\boldsymbol{s}}') \tag{1.2.2}$$

$$\begin{aligned}\psi^S[\psi](\boldsymbol{r}, \hat{\boldsymbol{s}}) = &- \int d\boldsymbol{r}' d\hat{\boldsymbol{s}}' g(\boldsymbol{r}, \hat{\boldsymbol{s}}; \boldsymbol{r}', \hat{\boldsymbol{s}}')\mu_t(\boldsymbol{r}')\psi(\boldsymbol{r}', \hat{\boldsymbol{s}}') \\ &+ \int d\boldsymbol{r}' d\hat{\boldsymbol{s}}' g(\boldsymbol{r}, \hat{\boldsymbol{s}}; \boldsymbol{r}', \hat{\boldsymbol{s}}')\mu_s(\boldsymbol{r}') \int d\hat{\boldsymbol{s}}'' f(\hat{\boldsymbol{s}} \cdot \hat{\boldsymbol{s}}'')\psi(\boldsymbol{r}', \hat{\boldsymbol{s}}'')\end{aligned} \tag{1.2.3}$$

Rewrite Eq.(1.2.1) as

$$\psi - \psi^S \psi = \psi^I q \tag{1.2.4}$$

The explicit physical meaning of Eq.(1.2.4) is: *The total field is the sum of the incident field due to the source $q(\boldsymbol{r}, \hat{\boldsymbol{s}})$ in free space and the scattering field due to the excited sources in the scattering volume.*

By definition, ψ^S is a linear functional acting on ψ. $\psi^I(\boldsymbol{r}', \hat{\boldsymbol{s}}')$ is a scalar field determined by the free space Green's function $g(\boldsymbol{r}, \hat{\boldsymbol{s}}; \boldsymbol{r}', \hat{\boldsymbol{s}}')$ and the source $q(\boldsymbol{r}', \hat{\boldsymbol{s}}')$. The free space Green's function is

$$g_t(\boldsymbol{r}, \hat{\boldsymbol{s}}; \boldsymbol{r}', \hat{\boldsymbol{s}}') = \frac{1}{|\boldsymbol{r} - \boldsymbol{r}'|}\delta(\hat{\boldsymbol{s}} - \hat{\boldsymbol{s}}_{\boldsymbol{r}-\boldsymbol{r}'})\delta(\hat{\boldsymbol{s}}' - \hat{\boldsymbol{s}}_{\boldsymbol{r}-\boldsymbol{r}'}) \tag{1.2.5}$$

Discretization of Eq.(1.2.1) will result in a set of linear equations. The solution to the resulting linear equations approximates the radiance field everywhere inside the scattering volume Ω. The radiance field everywhere outside of Ω can be calculated using Eq.(1.2.3)

3

The goal of this paper is to calculate the scattering field due to *plane wave incidence* coming in a given direction $\hat{s}$, i.e., this paper will use

$$\psi^I(\boldsymbol{r}', \hat{s}') = \delta(\hat{s} - \hat{s}^I) \tag{1.2.6}$$

as the incident radiance field. The use of Eq.(1.2.6) sidestepped the explicit expression of the incident source $q(\boldsymbol{r}, \hat{s})$, which has an infinitely large strength but is located at infinity to give a finite impact on the volume of interest.

2   2D and 3D Formalism

2.1   3D Formalism

The equations in section 1 are general. We need to specialize them into 2D and 3D. This paper focuses on 2D. Make the following substitution to equations in section 1:

$$\boldsymbol{r} \to (r, \theta_{\boldsymbol{r}}, \phi_{\boldsymbol{r}}) \qquad\qquad d\boldsymbol{r} = r^2 \sin\theta_{\boldsymbol{r}} dr d\theta_{\boldsymbol{r}} d\phi_{\boldsymbol{r}}$$
$$\hat{s} \to (\theta, \phi) \qquad\qquad d\hat{s} = \sin\theta d\theta d\phi$$
$$\hat{s} \cdot \hat{s}' \to \cos(\theta - \theta') + \cos(\phi - \phi')\sin\theta\sin\theta'$$

2.2   2D Formalism

Make the following substitution to equations in section 1:

$$\boldsymbol{r} \to (r, \phi_{\boldsymbol{r}}) \qquad\qquad d\boldsymbol{r} = r dr d\phi_{\boldsymbol{r}}$$
$$\hat{s} \to \phi \qquad\qquad d\hat{s} = d\phi$$
$$\hat{s} \cdot \hat{s}' \to \cos(\phi - \phi'), \phi - \phi'$$

The 2D RTE, VIE, and related equations are:

$$\hat{s} \cdot \nabla\psi(\boldsymbol{r}, \phi) + \mu_t(\boldsymbol{r})\psi(\boldsymbol{r}, \phi) = \mu_s(\boldsymbol{r}) \int d\phi' f(\phi - \phi')\psi(\boldsymbol{r}, \phi') + q(\boldsymbol{r}, \phi) \tag{2.2.1}$$

$$\psi(\boldsymbol{r}, \phi) - \psi^S[\psi](\boldsymbol{r}, \phi) = \psi^I[q](\boldsymbol{r}, \phi) \tag{2.2.2}$$

$$\begin{aligned}
\psi^S[\psi](\boldsymbol{r}, \phi) = &-\int d\boldsymbol{r}' d\phi' g(\boldsymbol{r}, \phi; \boldsymbol{r}', \phi')\mu_t(\boldsymbol{r}')\psi(\boldsymbol{r}', \phi') \\
&+ \int d\boldsymbol{r}' d\phi' g(\boldsymbol{r}, \phi; \boldsymbol{r}', \phi')\mu_s(\boldsymbol{r}') \int d\phi'' f(\phi' - \phi'')\psi(\boldsymbol{r}', \phi'')
\end{aligned} \tag{2.2.3}$$

$$\psi^I[q](\boldsymbol{r}, \phi) = \int d\boldsymbol{r}' d\phi' g(\boldsymbol{r}, \phi; \boldsymbol{r}', \phi')q(\boldsymbol{r}', \phi') \tag{2.2.4}$$

$$g(\boldsymbol{r}, \phi; \boldsymbol{r}', \phi') = \frac{1}{|\boldsymbol{r} - \boldsymbol{r}'|}\delta(\phi - \phi_{\boldsymbol{r} - \boldsymbol{r}'})\delta(\phi' - \phi_{\boldsymbol{r} - \boldsymbol{r}'}) \tag{2.2.5}$$

$$\psi^I(\boldsymbol{r}, \phi) = \delta(\phi - \phi^I) \tag{2.2.6}$$

In Eq.(2.2.6), ϕ^I is the incidence angle.

4

2.3 2D Heyney-Greenstein Phase Function

This paper uses the 2D Heyney-Greenstein (HG) phase function

$$
\begin{aligned}
f_g(\phi, \phi') &= f_g(\phi - \phi') \\
&= \frac{1}{2\pi} \frac{1 - g^2}{1 + g^2 - 2g\cos(\phi - \phi')} \\
&= \frac{1}{2\pi} \sum_{m=-\infty}^{+\infty} g^{|m|} e^{+im(\phi - \phi')}
\end{aligned}
\tag{2.3.1}
$$

The discrete Fourier components of the 2D HG phase function are

$$
f_m = (f_g)_m = \int_0^{2\pi} d\phi' e^{+im(\phi' - \phi)} f_g(\phi - \phi') = g^{|m|}
\tag{2.3.2}
$$

The convolution of two 2D HG phase functions is another 2D HG phase function

$$
\begin{aligned}
f_{g_1} * f_{g_2} &= (f_{g_1} * f_{g_2})(\phi - \phi') \\
&= \int d\phi'' f_{g_1}(\phi - \phi'') f_{g_2}(\phi'' - \phi') \\
&= \int d\phi'' \frac{1}{2\pi} \sum_{m_1=-\infty}^{+\infty} g_1^{|m_1|} e^{im_1(\phi - \phi'')} \frac{1}{2\pi} \sum_{m_2=-\infty}^{+\infty} g_2^{|m_2|} e^{im_2(\phi'' - \phi')} \\
&= \frac{1}{2\pi} \sum_{m_1=-\infty}^{+\infty} \sum_{m_2=-\infty}^{+\infty} \frac{1}{2\pi} \int d\phi'' e^{(m_2 - m_1)\phi''} g_2^{|m_2|} g_1^{|m_1|} e^{+im_1\phi} e^{-im_2\phi'} \\
&= \frac{1}{2\pi} \sum_{m_1=-\infty}^{+\infty} \sum_{m_2=-\infty}^{+\infty} \delta_{m_1,m_2} g_2^{|m_2|} g_1^{|m_1|} e^{+im_1\phi} e^{-im_2\phi'} \\
&= \frac{1}{2\pi} \sum_{m=-\infty}^{+\infty} (g_1 g_2)^{|m|} e^{+im(\phi - \phi')} \\
&= f_{g_1 g_2}(\phi - \phi')
\end{aligned}
\tag{2.3.3}
$$

2.4 Things to Do

To solve Eq.(2.2.2), the basic procedure is to

1. discretize the scatterer

2. apply test procedure

3. calculate matrix elements

4. solve the linear equations

Possible accelerating techniques are

5. matrix compression using low rank tensors

6. iterative linear solver using M-V multiplication

All the rest of this paper are devoted to each and all of the procedure and techniques mentioned above.

3   Discretizing the VIE

3.1   Finite Element Expansion

Grid the volume of interest using N_s triangles, numbered as $n = 1, 2, 3, \ldots, N_S$. Use the following finite series to approximate the true $\psi(\boldsymbol{s}, \phi)$:

$$\psi(\boldsymbol{r}, \phi) = \sum_{n=1}^{N_s} \sum_{m=-N_d}^{N_d} X_{nm} \xi_{nm}(\boldsymbol{r}, \phi) \tag{3.1.1}$$

$\{\xi_{nm}\}$ is a set of chosen basis functions. This paper uses the following one:

$$\xi_{nm}(\boldsymbol{r}, \phi) = S_n(\boldsymbol{r}) e^{im\phi} \tag{3.1.2a}$$

$$S_n(\boldsymbol{r}) = \begin{cases} 1, & \boldsymbol{r} \in S_n \\ 0, & \boldsymbol{r} \notin S_n \end{cases} \tag{3.1.2b}$$

Eq.(3.1.1) is hereafter called *the finite expansion* of $\psi(\boldsymbol{r}, \phi)$. The number of the degree of freedom (d.o.f.) of this finite expansion is $N_s(2N_d + 1)$.

This paper will use the following notations:

$S_n(\boldsymbol{r})$	the pulse function
S_n	the n-th triangle
$S(n)$	the area of the n-th triangle
$R(n)$	$\sqrt{S(n)/\pi}$
$\sum_{n,m}$	$\sum_{n=1}^{N_s} \sum_{m=-N_d}^{N_d}$
$\xi_{nm}(\boldsymbol{r}, \phi)$	$S_n(\boldsymbol{r}) e^{im\phi}$
N_n	N_s
N_m	$2N_d + 1$
N_g	$N_n N_m = N_s(2N_d + 1)$

3.2   Garlekin Test Procedure

We have, in short-hand notations, the finite expansion by Eq.(3.1.1):

$$\psi(\boldsymbol{r}, \phi) = \sum_{n,m} X_{nm} \xi_{nm}(\boldsymbol{r}, \phi) \tag{3.2.1}$$

Plug the finite expansion Eq.(3.1.1) into Eq.(2.2.2)

$$\sum_{n',m'} X_{n'm'} \{ \xi_{n'm'}(\boldsymbol{r}, \phi) - \psi^S[\xi_{n'm'}](\boldsymbol{r}, \phi) \} = \psi^I[q](\boldsymbol{r}, \phi) \tag{3.2.2}$$

To find out such a set of X_{nm} that the finite expansion (Eq.(3.1.1)) best approximates the true $\psi(\boldsymbol{r}, \phi)$, multiply $\xi_{nm}^*(\boldsymbol{r}, \phi)$ on both sides of Eq.(3.2.2) and integrate over $d\boldsymbol{r}d\phi$ over the volume of the scatterer.

$$\sum_{n',m'} X_{n'm'} \{ \int d\boldsymbol{r}d\phi \xi_{nm}^*(\boldsymbol{r}, \phi) \xi_{n'm'}(\boldsymbol{r}, \phi)$$
$$- \int d\boldsymbol{r}d\phi \xi_{nm}^*(\boldsymbol{r}, \phi) \psi^S[\xi_{n'm'}](\boldsymbol{r}, \phi) \} \tag{3.2.3}$$
$$= \int d\boldsymbol{r}d\phi \xi_{nm}^*(\boldsymbol{r}, \phi) \psi^I(\boldsymbol{r}, \phi)$$

Plug Eq.(2.2.3) in Eq.(3.2.3), define the following matrices:

$$Z_{(nm)(n'm')} = A_{(nm)(n'm')} + B_{(nm)(n'm')} \tag{3.2.4}$$

$$A_{(nm)(n'm')} = \int d\boldsymbol{r}d\phi \xi_{nm}^*(\boldsymbol{r},\phi)\xi_{n'm'}(\boldsymbol{r},\phi) \tag{3.2.5}$$

$$B_{(nm)(n'm')} = -\int d\boldsymbol{r}d\phi \xi_{nm}^*(\boldsymbol{r},\phi)\psi^S[\xi_{n'm'}](\boldsymbol{r},\phi) \tag{3.2.6}$$

$$V_{(nm)(n'm')} = \int d\boldsymbol{r}d\phi \xi_{nm}^*(\boldsymbol{r},\phi)\psi^I(\boldsymbol{r},\phi) \tag{3.2.7}$$

Let

$$B_{(nm)(n'm')} = B_{(nm)(n'm')}^t + B_{(nm)(n'm')}^s \tag{3.2.8a}$$

$$
\begin{aligned}
B_{(nm)(n'm')}^t = &\int d\boldsymbol{r}d\phi \xi_{nm}^*(\boldsymbol{r},\phi) \\
&\times \int d\boldsymbol{r}'d\phi' g(\boldsymbol{r},\phi;\boldsymbol{r}',\phi')\mu_t(\boldsymbol{r}')\xi_{n'm'}(\boldsymbol{r}',\phi')
\end{aligned}
\tag{3.2.8b}
$$

$$
\begin{aligned}
B_{(nm)(n'm')}^s = &-\int d\boldsymbol{r}d\phi \xi_{nm}^*(\boldsymbol{r},\phi) \\
&\times \int d\boldsymbol{r}'d\phi' g(\boldsymbol{r},\phi;\boldsymbol{r}',\phi')\mu_s(\boldsymbol{r}') \\
&\times \int d\phi'' f(\phi'-\phi'')\xi_{n'm'}(\boldsymbol{r}',\phi'')
\end{aligned}
\tag{3.2.8c}
$$

The VIE Eq.(2.2.2) is converted to a a set of linear equations:

$$Z \cdot X = V \tag{3.2.9}$$

The structure of Z is:

- $N_s \times N_s$ block matrices
- each block is a $N_d \times N_d$ matrix

4   Calculating Matrix Elements

4.1   Identity Term - $A_{(nm)(n'm')}$

$$
\begin{aligned}
A_{(nm)(n'm')} &= \int d\boldsymbol{r}d\phi \xi_{nm}^*(\boldsymbol{r},\phi)\xi_{n'm'}(\boldsymbol{r},\phi) \\
&= \int d\boldsymbol{r} S_n(\boldsymbol{r})S_{n'}(\boldsymbol{r}) \int d\phi e^{-im\phi}e^{+im'\phi}
\end{aligned}
\tag{4.1.1}
$$

Use the 1-point quadrature rule to do the integral over $\boldsymbol{r}$ and the fact that the integral over ϕ is a discrete delta function:

$$\int d\boldsymbol{r} S_n(\boldsymbol{r})S_{n'}(\boldsymbol{r}) = S(n)\delta_{nn'}$$

$$\int d\phi e^{-im\phi}e^{+im'\phi} = 2\pi\delta_{mm'}$$

The 1-point result is

$$A_{(nm)(n'm')} = 2\pi S(n)\delta_{nn'}\delta_{mm'} \tag{4.1.2}$$

4.2   Interaction Term - $B_{(nm)(n'm')}$

4.2.1 $B^t_{(nm)(n'm')}$ $n \neq n'$

$$
\begin{aligned}
B^t_{(nm)(n'm')} &= \int d\boldsymbol{r}\, d\phi\, S_n(\boldsymbol{r}) e^{-im\phi} \int d\boldsymbol{r}'\, d\phi'\, g(\boldsymbol{r}, \phi; \boldsymbol{r}', \phi') \mu_t(\boldsymbol{r}') S_{n'}(\boldsymbol{r}') e^{+im'\phi'} \\
&= \int d\phi\, S(n) e^{-im\phi} \int d\phi'\, S(n') g(\boldsymbol{r}_n, \phi; \boldsymbol{r}_{n'}, \phi') \mu_t(n') e^{+im'\phi'} \\
&= \mu_t(n') S(n) S(n') \int d\phi\, d\phi'\, e^{-im\phi} e^{+im'\phi'} g(\boldsymbol{r}_n, \phi; \boldsymbol{r}_{n'}, \phi') \\
&= \mu_t(n') S(n) S(n') \int d\phi\, d\phi'\, e^{-im\phi} e^{+im'\phi'} \frac{1}{|\boldsymbol{r}_n - \boldsymbol{r}_{n'}|} \delta(\phi - \phi_{\boldsymbol{r}_n - \boldsymbol{r}_{n'}}) \delta(\phi' - \phi_{\boldsymbol{r}_n - \boldsymbol{r}_{n'}}) \\
&= \mu_t(n') S(n) S(n') e^{-im\phi_{\boldsymbol{r}_n - \boldsymbol{r}_{n'}}} e^{+im'\phi_{\boldsymbol{r}_n - \boldsymbol{r}_{n'}}} \frac{1}{|\boldsymbol{r}_n - \boldsymbol{r}_{n'}|} \\
&= \mu_t(n') \frac{S(n) S(n')}{|\boldsymbol{r}_n - \boldsymbol{r}_{n'}|} e^{-i(m-m')\phi_{\boldsymbol{r}_n - \boldsymbol{r}_{n'}}}
\end{aligned}
\tag{4.2.1}
$$

4.2.2 $B^s_{(nm)(n'm')}$ $n \neq n'$

$$
\begin{aligned}
B^s_{(nm)(n'm')} &= - \int d\boldsymbol{r}\, d\phi\, S_n(\boldsymbol{r}) e^{-im\phi} \int d\boldsymbol{r}'\, d\phi'\, g(\boldsymbol{r}, \phi; \boldsymbol{r}', \phi') \mu_s(\boldsymbol{r}') \int d\phi''\, f(\phi' - \phi'') S_{n'}(\boldsymbol{r}') e^{+im'\phi''} \\
&= - \int d\boldsymbol{r}\, d\phi\, S_n(\boldsymbol{r}) e^{-im\phi} \int d\boldsymbol{r}'\, d\phi'\, S_{n'}(\boldsymbol{r}') e^{+im'\phi'} g(\boldsymbol{r}, \phi; \boldsymbol{r}', \phi') \mu_s(\boldsymbol{r}') \int d\phi''\, f(\phi' - \phi'') e^{+im'(\phi'' - \phi')} \\
&= - \int d\boldsymbol{r}\, d\phi\, S_n(\boldsymbol{r}) e^{-im\phi} \int d\boldsymbol{r}'\, d\phi'\, S_{n'}(\boldsymbol{r}') e^{+im'\phi'} g(\boldsymbol{r}, \phi; \boldsymbol{r}', \phi') \mu_s(\boldsymbol{r}') g^{|m'|}
\end{aligned}
\tag{4.2.2}
$$

Compare the last line of Eq.(4.2.2) to the first line of Eq.(4.2.1). Make the following replacement in the calculation of B^t:

$$
\begin{aligned}
-(minus) &\to + (plus) \\
\mu_t(n') &\to \mu_s(n') g^{|m'|}
\end{aligned}
$$

The final result for B^s is just

$$
B^s_{(nm)(n'm')} = -\mu_s(n') g^{|m'|} \frac{S(n) S(n')}{|\boldsymbol{r}_n - \boldsymbol{r}_{n'}|} e^{-i(m-m')\phi_{\boldsymbol{r}_n - \boldsymbol{r}_{n'}}}
\tag{4.2.3}
$$

4.2.3 $B^t_{(nm)(n'm')}$ $n = n'$

Still do the 1 point quadrature for the field point $\boldsymbol{r}$, but treat the source point $\boldsymbol{r}'$ with greater care:

$$
\begin{aligned}
B^t_{(nm)(n'm')} &= \int d\boldsymbol{r}\, d\phi\, S_n(\boldsymbol{r}) e^{-im\phi} \int d\boldsymbol{r}'\, d\phi'\, g(\boldsymbol{r}, \phi; \boldsymbol{r}', \phi') \mu_t(\boldsymbol{r}') S_n(\boldsymbol{r}') e^{+im'\phi'} \\
&= \mu_t(n) S(n) \int d\boldsymbol{r}' \int d\phi\, d\phi'\, e^{-im\phi} e^{+im'\phi'} \frac{1}{|\boldsymbol{r}_n - \boldsymbol{r}'|} \delta(\phi - \phi_{\boldsymbol{r}_n - \boldsymbol{r}'}) \delta(\phi' - \phi_{\boldsymbol{r}_n - \boldsymbol{r}'}) S_n(\boldsymbol{r}') \\
&= \mu_t(n) S(n) \int d\boldsymbol{r}'\, e^{-i(m-m')\phi_{\boldsymbol{r}_n - \boldsymbol{r}'}} \frac{1}{|\boldsymbol{r}_n - \boldsymbol{r}'|} S_n(\boldsymbol{r}')
\end{aligned}
\tag{4.2.4}
$$

Change variable

$$
\begin{aligned}
\boldsymbol{r}_n - \boldsymbol{r}' &= \boldsymbol{R} \\
|\boldsymbol{r}_n - \boldsymbol{r}'| &= |\boldsymbol{R}| = R \\
\phi_{\boldsymbol{r}_n - \boldsymbol{r}'} &= \phi_{\boldsymbol{R}} \\
d\boldsymbol{r}' &= d\boldsymbol{R} = R \; dR \; d\phi_{\boldsymbol{R}}
\end{aligned}
\tag{4.2.5}
$$

So that

$$
\begin{aligned}
B^t_{(nm)(n'm')} &= \mu_t(n) S(n) \int \phi_{\boldsymbol{R}} \; e^{-i(m-m')\phi_{\boldsymbol{r}_n - \boldsymbol{r}'}} \int R \; dR \frac{1}{R} S_n(\boldsymbol{r}') \\
&= \mu_t(n) S(n) 2\pi \delta_{mm'} \int dR \; S_n(\boldsymbol{r}')
\end{aligned}
\tag{4.2.6}
$$

Approximate the triangle domain S_n by a circle of radius $R(n) = \sqrt{S(n)/\pi}$

$$
B^t_{(nm)(n'm')} = 2\pi^{1/2} S(n)^{3/2} \mu_t(n) \delta_{mm'}
\tag{4.2.7}
$$

4.2.4 $B^s_{(nm)(n'm')} \; n = n'$

Similar to the derivation of off-diagonal B^s, the result is easily gotten

$$
B^s_{(nm)(n'm')} = -2\pi^{1/2} S(n)^{3/2} \mu_s(n) g^{|m|} \delta_{mm'}
\tag{4.2.8}
$$

4.3 Input Vector - V_{nm}

Use the plane wave incidence field in Eq.(2.2.6) to calculate the input vectorsymtor

$$
\begin{aligned}
V_{nm} &= \int d\boldsymbol{r} d\phi \, \xi^*_{nm}(\boldsymbol{r}, \phi) \psi^I(\boldsymbol{r}, \phi) \\
&= \int d\boldsymbol{r} d\phi \, S_n(\boldsymbol{r}) e^{-im\phi} \delta(\phi - \phi^I) \\
&\approx S(n) e^{-im\phi^I}
\end{aligned}
\tag{4.3.1}
$$

Used 1 point quadrature rule in the last step.

4.4 $Z = A + B$

For easy reference, list the result of section 4:

$$
A_{(nm)(n'm')} = 2\pi S(n) \delta_{nn'} \delta_{mm'}
\tag{4.4.1}
$$

$$
B^t_{(nm)(n'm')} = \mu_t(n') \frac{S(n)S(n')}{|\boldsymbol{r}_n - \boldsymbol{r}_{n'}|} e^{-i(m-m')\phi_{\boldsymbol{r}_n - \boldsymbol{r}_{n'}}}, \forall n \neq n'
\tag{4.4.2}
$$

$$
B^s_{(nm)(n'm')} = -\mu_s(n') g^{|m'|} \frac{S(n)S(n')}{|\boldsymbol{r}_n - \boldsymbol{r}_{n'}|} e^{-i(m-m')\phi_{\boldsymbol{r}_n - \boldsymbol{r}_{n'}}}, \forall n \neq n'
\tag{4.4.3}
$$

$$
B^t_{(nm)(n'm')} = 2\pi^{1/2} S(n)^{3/2} \mu_t(n) \delta_{mm'}, \forall n = n'
\tag{4.4.4}
$$

$$
B^s_{(nm)(n'm')} = -2\pi^{1/2} S(n)^{3/2} \mu_s(n) g^{|m|} \delta_{mm'}, \forall n = n'
\tag{4.4.5}
$$

$$
V_{nm} = S(n) e^{-im\phi^I}
\tag{4.4.6}
$$

So that

$$
\begin{aligned}
Z_{(nm)(n'm')} &= A_{(nm)(n'm')} + B_{(nm)(n'm')} \\
&= \begin{cases} 2\pi S(n)\{1 + [\mu_t(n) - \mu_s(n) g^{|m|}] S(n)^{1/2} \pi^{-1/2}\} & n = n' \\ \frac{S(n)S(n')}{|\boldsymbol{r}_n - \boldsymbol{r}_{n'}|} [\mu_t(n') - \mu_s(n') g^{|m'|}] e^{-i(m-m')\phi_{\boldsymbol{r}_n - \boldsymbol{r}_{n'}}} & n \neq n' \end{cases}
\end{aligned}
\tag{4.4.7}
$$

9

5 Accelerations

5.1 Acceleration on N_d

By the virtue of Eq.(4.4.7) and Eq.(4.3.1), in principle the system can be solved by direcitly inverting Z. But even for small N_s and N_d, the size of matrix Z could become excessively large for computers as of the year 2013. This section explains a method for accelerating N_d related calculation. The number of floating point operations (flop) is reduced by a factor of $N_m = 2N_d + 1$. That said, we achieved $O(N_d)$ scaling in both CPU and memory.

Treat $Z_{(nm)(n'm')}$ as $N_s \times N_s$ block matrices of size $N_d \times N_d$. The diagonal Z block matrices are

$$
\begin{aligned}
Z(n,n)_{mm'} &\equiv Z_{(nm)(n'm')} \\
&= \delta_{mm'}[2\pi S(n) + 2\pi^{1/2}\mu_t(n)S(n)^{3/2} - 2\pi^{1/2}\mu_s(n)S(n)^{3/2}g^{|m|}] \\
&= \delta_{mm'}[Zon1(n) + Zon2(n)g^{|m|}]
\end{aligned}
\tag{5.1.1}
$$

with two 1D vectorsymtors of length N_s

$$
Zon1(n) = 2\pi S(n) + 2\pi^{1/2}\mu_t(n)S(n)^{3/2} \tag{5.1.2a}
$$

$$
Zon2(n) = -2\pi^{1/2}\mu_s(n)S(n)^{3/2} \tag{5.1.2b}
$$

The off-diagonal Z block matrices are

$$
\begin{aligned}
Z(n,n')_{mm'} &\equiv Z_{(nm)(n'm')} \\
&= \frac{S(n)S(n')}{|\boldsymbol{r}_n - \boldsymbol{r}_{n'}|}\mu_t(n')e^{-im\phi_{\boldsymbol{r}_n - \boldsymbol{r}_{n'}}}e^{+im'\phi_{\boldsymbol{r}_n - \boldsymbol{r}_{n'}}} \\
&\quad - \frac{S(n)S(n')}{|\boldsymbol{r}_n - \boldsymbol{r}_{n'}|}\mu_s(n')e^{-im\phi_{\boldsymbol{r}_n - \boldsymbol{r}_{n'}}}g^{|m'|}e^{+im'\phi_{\boldsymbol{r}_n - \boldsymbol{r}_{n'}}} \\
&= Zoff(n,n')\mu_t(n')[v_{nn'} \times v_{nn'}^*]_{mm'} \\
&\quad - Zoff(n,n')\mu_s(n')[v_{nn'} \times (g\, v_{nn'}^*)]_{mm'} \\
&\quad - Zoff(n,n')[v_{nn'} \times u_{nn'}]_{mm'}
\end{aligned}
\tag{5.1.3}
$$

with the following pre-computed quantities

$$
Zoff(n,n') = \begin{cases} \frac{S(n)S(n')}{|\boldsymbol{r}_n - \boldsymbol{r}_{n'}|} & n \neq n' \\ 0 & n = n' \end{cases}
\tag{5.1.4a}
$$

$$
(v_{nn'})_m = e^{-m\phi_{\boldsymbol{r}_n - \boldsymbol{r}_{n'}}} \tag{5.1.4b}
$$

$$
(u_{nn'})_m = [\mu_t(n')v_{nn'}^* - \mu_s(n')gv_{nn'}^*]_m \tag{5.1.4c}
$$

$$
(g)_m = g^{|m|} \tag{5.1.4d}
$$

$$
(g\, v^*)_m = (g)_m(v_{nn'}^*)_m \tag{5.1.4e}
$$

5.2 Acceleration on N_s (not yet)

5.3 Matrix-Vector Multiplication without Assembling the Matrix

So far, all discussions focus on *how to calculate the full matrix faster.* However, when N_g is greater than a few thousands, direct linear solvers fail. In that regime, iterative solvers do the job. An iterative solver requires not the full matrix, but only the matrix-vectorsymtor (M-V) multiplication rule as the input. Factors affecting the performance of an iterative solver include, but not limited to:

- the time per call of the M-V multiplication

- the condition number of the matrix

- the iterative algorithm

- possible use of preconditioner

The performance, i.e., the time per call, of the M-V multiplication to a very large extent determines the performance of the iterative solver. In this subsection, we develop a fast M-V multiplication method for the 2D VIE.

Using Einstein's convention to sum over repeated indices, the input vectorsymtor X and the output vectorsymtor $Y = Z \cdot X$ can be expressed as

$$X(n')_{m'} \equiv X_{n'm'} \tag{5.3.1a}$$

$$Y(n)_m \equiv Y_{nm} = Z_{(nm)(n'm')} X_{n'm'} \tag{5.3.1b}$$

Use the $Z(n, n')$ defined in Eq.(5.1.1) and Eq.(5.1.3) to rewrite Eq.(5.3.1b) as

$$Y(n)_m = Z(n,n)_{mm'} X(n)_{m'} + \sum_{n \neq n'} Z(n,n')_{mm'} X(n')'_m \tag{5.3.2}$$

First, do the diagonal term:

$$
\begin{aligned}
Z(n,n)_{mm'} X(n)_{m'} &= \sum_{m'} \delta_{mm'} [Zon1(n) + Zon(2)g^{|m|}] X(n)_{m'} \\
&= [Zon1(n) + Zon(2)g^{|m|}] X(n)_m \\
&= Zon1(n) X(n)_m + Zon2(n) g^{|m|} X(n)_m \\
&= [Zon1(n) X(n) + Zon2(n)g \ X(n)]_m
\end{aligned}
\tag{5.3.3}
$$

Second, do each of the off diagonal terms separately:

$$
\begin{aligned}
Z(n,n')_{mm'} X(n')'_m &= \sum_{m'} Zoff(n,n')(v_{nn'} \times u_{nn'})_{mm'} X(n)_{m'} \\
&= Zoff(n,n')(v_{nn'})_m \sum_{m'} (u_{nn'})_{mm'} X(n)_{m'} \\
&= Zoff(n,n')(v_{nn'})_m [u_{nn'} \cdot X(n)]
\end{aligned}
\tag{5.3.4}
$$

Plug Eq.(5.3.3) and Eq.(5.3.4) into Eq.(5.3.2) while noting Eq.(5.1.4a) that $Zoff(n,n) = 0$, and write the result in the vectorsymtorial form:

$$
\begin{aligned}
Y(n) = &Zon1(n) X(n) + Zon2(n)g \ X(n) \\
&+ \sum_{m=-N_d}^{+N_d} Zoff(n,n')[X(n) \cdot u_{nn'}] v_{nn'}
\end{aligned}
\tag{5.3.5}
$$

6   Better Convergence

6.1   Convergence Issue

The true, unapproximated plane wave incidence

$$\psi^I(\boldsymbol{r}, \phi) = \delta(\phi - \phi^I) = \lim_{N_d \to \infty} \frac{1}{2\pi} \sum_{m=-N_d}^{N_d} e^{im(\phi - \phi^I)} \tag{6.1.1}$$

has an infnitely extended discrete Fourier spectrum of equal strength $\frac{1}{2\pi}$. Thus, the leading order term in the solution also has the Dirac-delta function. Any truncation at finite N_d would miss out the high frequency components. Any effort with a finite N_d would run into the trouble of reconstructing a Dirac-delta function, which failed us in terms of accuracy (how close were the numerical solutions to the true solution?) and precision

(did the numericl solutions convergence to a common value?).

Numerical experiment (awaiting completion) showed some sign of psuedo-convergence in the range $N_d = 10 \sim 20$. But the mechanism broke down as N_d went higher - not a sign of convergence. This seemingly weird phenamenon is understood in detail (not detailed here) as a result of the singularity of the incidence wave. The way out is to extract whatever singularities in the true solution by hand before feeding the rest, expectedly smoother part into the numerical mechanism.

Section (6) notes down the attempt to eliminate the singular terms and achieve better accuracy and precision. Each following subsections consists of two parts. The first parts are general formalism. The second parts are 2D formalism.

6.2 Extracting ψ_b

6.2.1 General

The RTE reads, in its general form, as in Eq.(1.1.1)

$$\hat{s} \cdot \nabla \psi(\boldsymbol{r}, \hat{s}) + \mu_t(\boldsymbol{r})\psi(\boldsymbol{r}, \hat{s}) = \mu_s(\boldsymbol{r}) \int d\hat{s}' f(\hat{s} \cdot \hat{s}')\psi(\boldsymbol{r}', \hat{s}') + q(\boldsymbol{r}, \hat{s})$$

Let

$$\psi(\boldsymbol{r}, \hat{s}) = \psi_b(\boldsymbol{r}, \hat{s}) + \psi_{sc}(\boldsymbol{r}, \hat{s}) \tag{6.2.1}$$

The subscript b stands for **ballistic**. The subscript sc stands for **scattering**. Plug Eq.(6.2.1) into Eq.(1.1.1)

$$\hat{s} \cdot \psi_b(\boldsymbol{r}, \hat{s}) + \mu_t(\boldsymbol{r})\psi_b(\boldsymbol{r}, \hat{s}) + \hat{s} \cdot \psi_{sc}(\boldsymbol{r}, \hat{s}) + \mu_t(\boldsymbol{r})\psi_{sc}(\boldsymbol{r}, \hat{s})$$
$$= q(\boldsymbol{r}, \hat{s}) + \mu_s(\boldsymbol{r}) \int d\hat{s}' f(\hat{s} \cdot \hat{s}')\psi_{sc}(\boldsymbol{r}, \hat{s}') + \mu_s(\boldsymbol{r}) \int d\hat{s}' f(\hat{s} \cdot \hat{s}')\psi_b(\boldsymbol{r}, \hat{s}') \tag{6.2.2}$$

Let

$$\hat{s} \cdot \psi_b(\boldsymbol{r}, \hat{s}) + \mu_t(\boldsymbol{r})\psi_b(\boldsymbol{r}, \hat{s}) = q(\boldsymbol{r}, \hat{s}) \tag{6.2.3a}$$

$$\hat{s} \cdot \psi_{sc}(\boldsymbol{r}, \hat{s}) + \mu_t(\boldsymbol{r})\psi_{sc}(\boldsymbol{r}, \hat{s}) = \mu_s(\boldsymbol{r}) \int d\hat{s}' f(\hat{s} \cdot \hat{s}')\psi_{sc}(\boldsymbol{r}, \hat{s}') + q_b(\boldsymbol{r}, \hat{s}) \tag{6.2.3b}$$

$$q_b(\boldsymbol{r}, \hat{s}) = \mu_s(\boldsymbol{r}) \int d\hat{s}' f(\hat{s} \cdot \hat{s}')\psi_b(\boldsymbol{r}, \hat{s}') \tag{6.2.3c}$$

Eq.(6.2.3) is completely equivalent to Eq.(6.2.2). The solution to Eq.(6.2.3a), $\psi_b(\boldsymbol{r}, \hat{s})$, can be calculated using the Green's function for Eq.(6.2.3a)

$$g_t(\boldsymbol{r}, \hat{s}; \boldsymbol{r}', \hat{s}') = \frac{e^{-\tau(\boldsymbol{r}' \to \boldsymbol{r})}}{|\boldsymbol{r} - \boldsymbol{r}'|} \delta(\hat{s} - \hat{s}_{\boldsymbol{r} - \boldsymbol{r}'})\delta(\hat{s}' - \hat{s}_{\boldsymbol{r} - \boldsymbol{r}'}) \tag{6.2.4}$$

The $\tau(\boldsymbol{r}' \to \boldsymbol{r})$ in Eq.(6.2.4) is the *optical length* from $\boldsymbol{r}'$ to $\boldsymbol{r}$. The optical length is defined as the following path integrals:

$$\tau(\boldsymbol{r}' \to \boldsymbol{r}) \equiv \int_{C_{\boldsymbol{r}' \to \boldsymbol{r}}} \mu_t dl \tag{6.2.5a}$$

$$\tau(\boldsymbol{r}, -\hat{s}^I) \equiv \int_{C_{\boldsymbol{r}, -\hat{s}^I}} \mu_t dl \tag{6.2.5b}$$

In Eq.(6.2.5),

$$\begin{array}{ll} C_{\boldsymbol{r}' \to \boldsymbol{r}} & \text{is the straight line from } \boldsymbol{r}' \text{ to } \boldsymbol{r} \\ C_{\boldsymbol{r}, -\hat{s}^I} & \text{is the ray that starts from } \boldsymbol{r}' \text{ and in the direction } -\hat{s}^I \end{array}$$

List Eq.(6.2.4), Eq.(1.2.5), Eq.(1.2.2), and Eq.(1.2.6) together:

$$g(\boldsymbol{r}, \hat{\boldsymbol{s}}; \boldsymbol{r}', \hat{\boldsymbol{s}}') = \frac{1}{|\boldsymbol{r} - \boldsymbol{r}'|} \delta(\hat{\boldsymbol{s}} - \hat{\boldsymbol{s}}_{\boldsymbol{r}-\boldsymbol{r}'}) \delta(\hat{\boldsymbol{s}}' - \hat{\boldsymbol{s}}_{\boldsymbol{r}-\boldsymbol{r}'}) \tag{6.2.6a}$$

$$g_t(\boldsymbol{r}, \hat{\boldsymbol{s}}; \boldsymbol{r}', \hat{\boldsymbol{s}}') = e^{-\tau(\boldsymbol{r}' \to \boldsymbol{r})} g(\boldsymbol{r}, \hat{\boldsymbol{s}}; \boldsymbol{r}', \hat{\boldsymbol{s}}') \tag{6.2.6b}$$

$$\psi^I[q](\boldsymbol{r}, \hat{\boldsymbol{s}}) = \int d\boldsymbol{r}' d\hat{\boldsymbol{s}}'\, g(\boldsymbol{r}, \hat{\boldsymbol{s}}; \boldsymbol{r}', \hat{\boldsymbol{s}}') q(\boldsymbol{r}', \hat{\boldsymbol{s}}') \tag{6.2.6c}$$

$$= \delta(\hat{\boldsymbol{s}} - \hat{\boldsymbol{s}}^I) \tag{6.2.6d}$$

The ballistic term ψ_b is

$$\psi_b(\boldsymbol{r}, \hat{\boldsymbol{s}}) = \int d\boldsymbol{r}' d\hat{\boldsymbol{s}}'\, g_t(\boldsymbol{r}, \hat{\boldsymbol{s}}; \boldsymbol{r}', \hat{\boldsymbol{s}}') q(\boldsymbol{r}', \hat{\boldsymbol{s}}')$$
$$= e^{-\tau(\boldsymbol{r}, -\hat{\boldsymbol{s}}^I)} \delta(\hat{\boldsymbol{s}} - \hat{\boldsymbol{s}}^I) \tag{6.2.7}$$

6.2.2   2D

Write all the equations in section 6.2.1 in 2D:

$$\hat{\boldsymbol{s}} \cdot \psi_b(\boldsymbol{r}, \phi) + \mu_t(\boldsymbol{r}) \psi_b(\boldsymbol{r}, \phi) = q(\boldsymbol{r}, \phi) \tag{6.2.8a}$$

$$\hat{\boldsymbol{s}} \cdot \psi_{sc}(\boldsymbol{r}, \phi) + \mu_t(\boldsymbol{r}) \psi_{sc}(\boldsymbol{r}, \phi) = \mu_s(\boldsymbol{r}) \int d\phi'\, f(\phi - \phi') \psi_{sc}(\boldsymbol{r}, \phi') + q_b(\boldsymbol{r}, \phi) \tag{6.2.8b}$$

$$q_b(\boldsymbol{r}, \phi) = \mu_s(\boldsymbol{r}) \int d\phi'\, f(\phi - \phi') \psi_b(\boldsymbol{r}, \phi') \tag{6.2.8c}$$

$$\psi_b(\boldsymbol{r}, \phi) = \int d\boldsymbol{r}' d\phi'\, g_t(\boldsymbol{r}, \phi; \boldsymbol{r}', \phi') q(\boldsymbol{r}', \phi')$$
$$= e^{-\tau(\boldsymbol{r}, -\hat{\boldsymbol{s}}^I)} \delta(\phi - \phi^I) \tag{6.2.8d}$$

$$\psi_{sc}^I[q_b](\boldsymbol{r}, \phi) = \int d\boldsymbol{r}' d\phi'\, g(\boldsymbol{r}, \phi; \boldsymbol{r}', \phi') q_b(\boldsymbol{r}', \phi') \tag{6.2.8e}$$

$$g_t(\boldsymbol{r}, \phi; \boldsymbol{r}', \phi') = \frac{e^{-\tau(\boldsymbol{r}' \to \boldsymbol{r})}}{|\boldsymbol{r} - \boldsymbol{r}'|} \delta(\phi - \phi_{\boldsymbol{r}-\boldsymbol{r}'}) \delta(\phi' - \phi_{\boldsymbol{r}-\boldsymbol{r}'}) \tag{6.2.8f}$$

$$g(\boldsymbol{r}, \phi; \boldsymbol{r}', \phi') = \frac{1}{|\boldsymbol{r} - \boldsymbol{r}'|} \delta(\phi - \phi_{\boldsymbol{r}-\boldsymbol{r}'}) \delta(\phi' - \phi_{\boldsymbol{r}-\boldsymbol{r}'}) \tag{6.2.8g}$$

Only need to calculate the new input vectorsymtor, $(V_{sc})_{nm}$, to re-use the mechanism developed in the previous sections.

$$(V_{sc})_{nm} = \int d\boldsymbol{r} d\phi\, \xi_{nm}^*(\boldsymbol{r}, \phi) \psi_{sc}^I[q_b](\boldsymbol{r}, \phi) \tag{6.2.9}$$

Calculate in the following order

$$\psi_b(\boldsymbol{r}, \phi) \to q_b(\boldsymbol{r}, \phi) \to \psi_{sc}^I(\boldsymbol{r}, \phi) \to (V_{sc})_{nm} \tag{6.2.10}$$

$$
\begin{aligned}
q_b(\boldsymbol{r}, \phi) &= \mu_s(\boldsymbol{r}) \int d\phi' f(\phi - \phi') \psi_b(\boldsymbol{r}, \phi') \\
&= \mu_s(\boldsymbol{r}) \int d\phi' f(\phi - \phi') \delta(\phi - \phi^I) e^{-\tau(\boldsymbol{r}, -\hat{\boldsymbol{s}}^I)} \\
&= \mu_s(\boldsymbol{r}) f(\phi - \phi^I) e^{-\tau(\boldsymbol{r}, -\hat{\boldsymbol{s}}^I)}
\end{aligned}
\tag{6.2.11a}
$$

$$
\begin{aligned}
\psi_{sc}^I(\boldsymbol{r}, \phi) &= \int d\boldsymbol{r}' d\phi' g(\boldsymbol{r}, \phi; \boldsymbol{r}', \phi') q_b(\boldsymbol{r}', \phi') \\
&= \int d\boldsymbol{r}' d\phi' \frac{1}{|\boldsymbol{r} - \boldsymbol{r}'|} \delta(\phi - \phi_{\boldsymbol{r} - \boldsymbol{r}'}) \delta(\phi' - \phi_{\boldsymbol{r} - \boldsymbol{r}'}) \mu_s(\boldsymbol{r}') f(\phi' - \phi^I) e^{-\tau(\boldsymbol{r}', -\hat{\boldsymbol{s}}^I)} \\
&= \int d\boldsymbol{r}' \frac{1}{|\boldsymbol{r} - \boldsymbol{r}'|} \delta(\phi - \phi_{\boldsymbol{r} - \boldsymbol{r}'}) \mu_s(\boldsymbol{r}') f(\phi_{\boldsymbol{r} - \boldsymbol{r}'} - \phi^I) e^{-\tau(\boldsymbol{r}', -\hat{\boldsymbol{s}}^I)}
\end{aligned}
\tag{6.2.11b}
$$

$$
\begin{aligned}
(V_{sc})_{nm} &= \int d\boldsymbol{r} d\phi \; \xi_{nm}^*(\boldsymbol{r}, \phi) \psi_{sc}^I(\boldsymbol{r}, \phi) \\
&= \int d\boldsymbol{r} d\phi \; S_n(\boldsymbol{r}) e^{-im\phi} \int d\boldsymbol{r}' \frac{1}{|\boldsymbol{r} - \boldsymbol{r}'|} \delta(\phi - \phi_{\boldsymbol{r} - \boldsymbol{r}'}) \mu_s(\boldsymbol{r}') f(\phi_{\boldsymbol{r} - \boldsymbol{r}'} - \phi^I) e^{-\tau(\boldsymbol{r}', -\hat{\boldsymbol{s}}^I)} \\
&= \int d\boldsymbol{r} S_n(\boldsymbol{r}) \int d\boldsymbol{r}' \frac{e^{-im\phi_{\boldsymbol{r} - \boldsymbol{r}'}}}{|\boldsymbol{r} - \boldsymbol{r}'|} \mu_s(\boldsymbol{r}') f(\phi_{\boldsymbol{r} - \boldsymbol{r}'} - \phi^I) e^{-\tau(\boldsymbol{r}', -\hat{\boldsymbol{s}}^I)} \\
&= S(n) \int d\boldsymbol{r}' \frac{e^{-im\phi_{\boldsymbol{r}_n - \boldsymbol{r}'}}}{|\boldsymbol{r}_n - \boldsymbol{r}'|} \mu_s(\boldsymbol{r}') f(\phi_{\boldsymbol{r}_n - \boldsymbol{r}'} - \phi^I) e^{-\tau(\boldsymbol{r}', -\hat{\boldsymbol{s}}^I)}
\end{aligned}
\tag{6.2.11c}
$$

The integral over $\boldsymbol{r}'$ shall be carried out over the entire volume of interest.

$$
\int d\boldsymbol{r}' = \sum_{n'=1}^{N_s} \int d\boldsymbol{r}' S_{n'}(\boldsymbol{r}') = \int d\boldsymbol{r}' S_n(\boldsymbol{r}') + \sum_{n' \neq n} \int d\boldsymbol{r}' S_{n'}(\boldsymbol{r}')
\tag{6.2.12}
$$

The first term in Eq.(6.2.12) is the self-interaction/ diagonal term. Use the same change of variable as in Eq.(4.2.5)

$$
\begin{aligned}
&S(n) \int d\boldsymbol{r}' e^{-im\phi_{\boldsymbol{r}_n - \boldsymbol{r}'}} \frac{1}{|\boldsymbol{r}_n - \boldsymbol{r}'|} \mu_s(\boldsymbol{r}') f(\phi_{\boldsymbol{r}_n - \boldsymbol{r}'} - \phi^I) e^{-\tau(\boldsymbol{r}', -\hat{\boldsymbol{s}}^I)} \\
=&S(n) \int d\boldsymbol{R} \; e^{-im\phi_{\boldsymbol{R}}} \frac{1}{R} \mu_s(n) f(\phi_{\boldsymbol{R}} - \phi^I) e^{-\tau(\boldsymbol{r}_n, -\hat{\boldsymbol{s}}^I)} \\
=&S(n) \mu_s(n) e^{-im\phi^I} e^{-\tau(\boldsymbol{r}_n, -\hat{\boldsymbol{s}}^I)} \int dR \int d\phi_{\boldsymbol{R}} \; e^{-im(\phi_{\boldsymbol{R}} - \phi^I)} f(\phi_{\boldsymbol{R}} - \phi^I) \\
=&S(n) \mu_s(n) g^{|m|} e^{-im\phi^I} e^{-\tau(\boldsymbol{r}_n, -\hat{\boldsymbol{s}}^I)} \sqrt{S(n)/\pi} \\
=&\pi^{-1/2} S(n)^{3/2} \mu_s(n) g^{|m|} e^{-im\phi^I} e^{-\tau(\boldsymbol{r}_n, -\hat{\boldsymbol{s}}^I)}
\end{aligned}
\tag{6.2.13}
$$

The second terms in Eq.(6.2.12) are the other-interaction/ off-diagonal terms are each:

$$
\begin{aligned}
&S(n) \int d\boldsymbol{r}' e^{-im\phi_{\boldsymbol{r}_n - \boldsymbol{r}'}} \frac{1}{|\boldsymbol{r}_n - \boldsymbol{r}'|} \mu_s(\boldsymbol{r}') f(\phi_{\boldsymbol{r}_n - \boldsymbol{r}'} - \phi^I) e^{-\tau(\boldsymbol{r}', -\hat{\boldsymbol{s}}^I)} \\
=&\frac{S(n) S(n')}{|\boldsymbol{r}_n - \boldsymbol{r}_{n'}|} \mu_s(n') e^{-im\phi_{\boldsymbol{r}_n - \boldsymbol{r}_{n'}}} f(\phi_{\boldsymbol{r}_n - \boldsymbol{r}_{n'}} - \phi^I) e^{-\tau(\boldsymbol{r}', -\hat{\boldsymbol{s}}^I)}
\end{aligned}
\tag{6.2.14}
$$

For numerical computation, pre-compute the following two quantities:

$$
damp(n) = \exp\left[-\tau(\boldsymbol{r}_n, -\hat{\boldsymbol{s}}^I)\right]
\tag{6.2.15a}
$$

$$
r2rF(n, n') = f(\phi_{\boldsymbol{r}_n - \boldsymbol{r}_{n'}} - \phi^I)
\tag{6.2.15b}
$$

Recall other pre-computed quantities defined in Eq.(5.1.4), the new $(V_{sc})_{nm}$ can be expressed solely by the pre-computed quantities.

$$
\begin{aligned}
(V_{sc})_{nm} =& \pi^{-1/2} S(n)^{3/2} \mu_s(n) g^{|m|} e^{-im\phi^I} e^{-\tau(\boldsymbol{r}_n, -\hat{\boldsymbol{s}}^I)} \\
& + \sum_{n' \neq n} \frac{S(n)S(n')}{|\boldsymbol{r}_n - \boldsymbol{r}_{n'}|} \mu_s(n') e^{-im\phi_{\boldsymbol{r}_n - \boldsymbol{r}_{n'}}} f(\phi_{\boldsymbol{r}_n - \boldsymbol{r}_{n'}} - \phi^I) e^{-\tau(\boldsymbol{r}_{n'}, -\hat{\boldsymbol{s}}^I)} \\
=& \pi^{-1/2} S(n)^{1/2} \mu_s(n) damp(n) g_m V_{nm} \\
& + \sum_{n' \neq n} Zoff(n, n') r2rF(n, n') damp(n') \mu_s(n') (v_{nn'})_m \\
=& \pi^{-1/2} S(n)^{1/2} \mu_s(n) damp(n) g_m V_{nm} \\
& + \sum_{n'=1}^{N_s} Zoff(n, n') r2rF(n, n') damp(n') \mu_s(n') (v_{nn'})_m
\end{aligned}
\tag{6.2.16}
$$

The last line in Eq.(6.2.16) used the fact that $Zoff(n, n) = 0$.

Solve

$$
Z \cdot X_{sc} = V_{sc}
\tag{6.2.17}
$$

for X_{sc}. Also, considering the contribution from ψ_b,

$$
\begin{aligned}
(X_b)_{nm} &= \int d\boldsymbol{r} d\phi \ \xi_{nm}^*(\boldsymbol{r}, \phi) \psi_b(\boldsymbol{r}, \phi) \\
&= \int d\boldsymbol{r} d\phi \ S_n(\boldsymbol{r}) e^{-im\phi} e^{-\tau(\boldsymbol{r}, -\hat{\boldsymbol{s}}^I)} \delta(\phi - \phi^I) \\
&= S(n) e^{-im\phi^I} e^{-\tau(\boldsymbol{r}_n, -\hat{\boldsymbol{s}}^I)}
\end{aligned}
\tag{6.2.18}
$$

The final solution vectorsymtor X is

$$
X = X_b + X_{sc}
\tag{6.2.19}
$$

7   Quadrature Rules

This section documents the equations good for code implementation.

7.1   Gaussian Quadrature Rules (not finished yet)

A general 1D integral

$$
\int_a^b f(x) dx
\tag{7.1.1}
$$

can be transformed ($x = \frac{a+b}{2} + \frac{b-a}{2} t$) into

$$
\frac{b-a}{2} \int_{-1}^{+1} f(\frac{a+b}{2} + \frac{b-a}{2} t) dt
\tag{7.1.2}
$$

So that we only need to consider the folloing standard integral

$$
\int_{-1}^{+1} f(x) dx
\tag{7.1.3}
$$

Gaussian quadrature rules are well-known. Gauss-Legenre quadrature rule is used for smooth integrand. (not finished yet)

7.2   Singular and Near-Singular Integrations

Given an arbitrary triangle with nodes

$$P_0(x_0, y_0)$$
$$P_1(x_1, y_1)$$
$$P_2(x_2, y_2)$$

The integrand is singular at P_0

$$\frac{f(x, y)}{\sqrt{(x - x_0)^2 + (y - y_0)^2}} \tag{7.2.1}$$

Usually $\sqrt{(x - x_0)^2 + (y - y_0)^2}$ is denoted as R. The above integrand is called R^{-1} singular.
Shift the origin to P_0, then rotate the axes (passive perspective of rotation) so that the y'-axis align with the normal direction of $\overrightarrow{P_1 P_2}$, namely $\hat{\boldsymbol{n}}$. The transformed coordinates are

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \frac{1}{|\overrightarrow{P_1 P_2}|} \begin{pmatrix} x_2 - x_1 & y_2 - y_1 \\ -y_2 + y_1 & x_2 - x_1 \end{pmatrix} \begin{pmatrix} x - x_0 \\ y - y_0 \end{pmatrix} \tag{7.2.2}$$

P_1 and P_2 have the following transormed coordinates:

$$x_1' = [(x_2 - x_1)(x_1 - x_0) + (y_2 - y_1)(y_1 - y_0)]/|\overrightarrow{P_1 P_2}| \tag{7.2.3a}$$
$$x_2' = [(x_2 - x_1)(x_2 - x_0) + (y_2 - y_1)(y_2 - y_0)]/|\overrightarrow{P_1 P_2}| \tag{7.2.3b}$$
$$y_1' = y_2' = [(x_1 - x_2)y_0 + (x_2 - x_0)y_1 + (x_0 - x_1)y_2]/|\overrightarrow{P_1 P_2}| \tag{7.2.3c}$$
$$h = y_1' = y_2' \tag{7.2.3d}$$

Define Do the following $\sinh^{-1}$ transform

$$u = \sinh^{-1}\left(\frac{x'}{y'}\right)$$
$$v = y' \tag{7.2.4}$$

The inverse transform is:

$$x' = v \sinh u$$
$$y' = v \tag{7.2.5}$$

u and v are called the transverse and the radial coordinates, respectively.
The Jacobian cancels the R^{-1} divergence in the original integrand:

$$J = \left| \frac{\partial(u, v)}{\partial(x', y')} \right| = \sqrt{x'^2 + y'^2} = R \tag{7.2.6}$$

The triangular area defined by $P_0 P_1 P_2$ in $x - y$, $x' - y'$ plane is thus transformed into a rectangular area in $u - v$ plane.

$$\begin{aligned}
I &= \iint_{\Delta P_0 P_1 P_2} dx\, dy \frac{f(x, y)}{\sqrt{(x - x_0)^2 + (y - y_0)^2}} \\
&= \iint_{\Delta P_0 P_1 P_2} \frac{dx'\, dy'}{\sqrt{x'^2 + y'^2}} f(x(x', y'), y(x', y')) \\
&= \int_{u_<}^{u_>} du \int_{v_<}^{v_>} dv\, f(x, y)
\end{aligned} \tag{7.2.7}$$

16

where $u_<(u_>)$ and $v_<(v_>)$ are the smaller(larger) one of u_1, u_2 and v_1, v_2, respectively:

$$\begin{cases} u_1 = \sinh^{-1}(x_1'/h) \\ u_2 = \sinh^{-1}(x_2'/h) \end{cases} \quad \begin{cases} v_1 = 0 \\ v_2 = h \end{cases} \tag{7.2.8}$$

Use Eq.(7.2.2) and Eq.(7.2.5) to express x, y in terms of u, v:

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} x_0 \\ y_0 \end{pmatrix} + \frac{1}{|\overrightarrow{P_1 P_2}|} \begin{pmatrix} x_2 - x_1 & -y_2 + y_1 \\ y_2 - y_1 & x_2 - x_1 \end{pmatrix} \begin{pmatrix} v \sinh u \\ v \end{pmatrix} \tag{7.2.9}$$

The standard Gaussian quadrature rules are often given for $(0,1)$ as $\{\xi^i, w^i\}$, where ξ^i is the i-th abscissas an the w^i is the i-th weight.

As an example, the n-th order Gauss-Legendre rule

$$\int_0^1 f(x)dx = \sum_i^n w^i f(\xi^i) \tag{7.2.10}$$

is exact up to order $2n + 1$.

For an integral from a to b, similar to Eq.(7.1.2).

$$\begin{aligned} \int_a^b f(x)dx &= (b - a) \int_{-1}^{+1} f[a + (b - a)t]dt \\ &= (b - a) \sum_i^n w^i f[a + (b - a)\xi^i] \end{aligned} \tag{7.2.11}$$

For a rectangular area in $u-v$ plane, as in Eq.(7.2.7), simply use the product of two independent Gauss-Legendre rules. Construct two standard Gaussian quadrature rules, i.e., for $(0,1)$, for u- and v- axis, respectively.

$$\begin{pmatrix} u^i \\ v^j \end{pmatrix} = \begin{pmatrix} u_1 + (u_2 - u_1) \; \xi_u^i \\ h \; \xi_v^j \end{pmatrix} \qquad \text{with} \qquad \begin{array}{l} \{\xi_u^i, w_u^i\}_{i=1}^{N_u} \\ \{\xi_v^j, w_v^j\}_{j=1}^{N_v} \end{array} \tag{7.2.12}$$

Eq.(7.2.7) becomes

$$\begin{aligned} I &= (v_> - v_<)(u_> - u_<) \sum_{i,j}^{N_u, N_v} w_u^i w_v^j f(x, y)\big|_{(i,j)} \\ &= |h(u_1 - u_2)| \sum_{i,j}^{N_u, N_v} w_u^i w_v^j f(x, y)\big|_{(i,j)} \end{aligned} \tag{7.2.13}$$

The (x, y) coordinates can be obtained by:

$$\begin{pmatrix} x \\ y \end{pmatrix}\bigg|_{(i,j)} = \begin{pmatrix} x_0 \\ y_0 \end{pmatrix} + \frac{1}{|\overrightarrow{P_1 P_2}|} \begin{pmatrix} x_2 - x_1 & -y_2 + y_1 \\ y_2 - y_1 & x_2 - x_1 \end{pmatrix} \begin{pmatrix} v^j \sinh u^i \\ v^j \end{pmatrix} \tag{7.2.14a}$$

Note that Eq.(7.2.14) made no assumption that $h > 0$ or $u_1 < u_2$. This is very suitable for numerical implementation.

Summary of equations for the $\sinh^{-1}$ transform method:

$$I = \iint_{\Delta P_0 P_1 P_2} dxdy \frac{f(x,y)}{\sqrt{(x-x_0)^2 + (y-y_0)^2}} \tag{7.2.15a}$$

$$= |h(u_1 - u_2)| \sum_{i,j}^{N_u, N_v} w_u^i w_v^j f(x,y)\big|_{(i,j)}$$

$$x_1' = [(x_2 - x_1)(x_1 - x_0) + (y_2 - y_1)(y_1 - y_0)]/|\overrightarrow{P_1 P_2}| \tag{7.2.15b}$$

$$x_2' = [(x_2 - x_1)(x_2 - x_0) + (y_2 - y_1)(y_2 - y_0)]/|\overrightarrow{P_1 P_2}| \tag{7.2.15c}$$

$$y_1' = y_2' = [(x_1 - x_2)y_0 + (x_2 - x_0)y_1 + (x_0 - x_1)y_2]/|\overrightarrow{P_1 P_2}| \tag{7.2.15d}$$

$$h = y_1' = y_2' \tag{7.2.15e}$$

$$\begin{cases} u_1 = \sinh^{-1}(x_1'/h) \\ u_2 = \sinh^{-1}(x_2'/h) \end{cases} \qquad \begin{cases} v_1 = 0 \\ v_2 = h \end{cases} \tag{7.2.15f}$$

$$\begin{pmatrix} x \\ y \end{pmatrix}\bigg|_{(i,j)} = \begin{pmatrix} x_0 \\ y_0 \end{pmatrix} + \frac{1}{|\overrightarrow{P_1 P_2}|} \begin{pmatrix} x_2 - x_1 & -y_2 + y_1 \\ y_2 - y_1 & x_2 - x_1 \end{pmatrix} \begin{pmatrix} v^j \sinh u^i \\ v^j \end{pmatrix} \tag{7.2.15g}$$

$$\begin{pmatrix} u^i \\ v^j \end{pmatrix} = \begin{pmatrix} u_1 + (u_2 - u_1)\,\xi_u^i \\ h\,\xi_v^j \end{pmatrix} \tag{7.2.15h}$$

7.3   Radial-Angular Method

We may also use the radial-angular method proposed by Khayat and Wilton [Khayat, Wilton, 2008]. The R is general 3D. The observation point is (x_0, y_0, z) with possible non-vanishing z.
Use cylindrical coordinates:

$$x' = x - x_0 = r' \cos \varphi' \tag{7.3.1a}$$

$$y' = y - y_0 = r' \sin \varphi' \tag{7.3.1b}$$

The radial-angular method used the following transorm for singularity cancellation:

$$u = -\ln \tan \frac{\varphi'}{2}$$

$$= \ln \left(\sqrt{1 + \left(\frac{x'}{y'}\right)^2} + \frac{x'}{y'} \right) \tag{7.3.2a}$$

$$= \sinh^{-1}\left(\frac{x'}{y'}\right)$$

$$v = R = \sqrt{x'^2 + y'^2 + z^2} \tag{7.3.2b}$$

$$J = \frac{R}{\cosh u} \tag{7.3.2c}$$

Note that

$$\tan \frac{\varphi'}{2} = e^{-u}$$

$$x'^2 + y'^2 = r'^2 \tag{7.3.3}$$

The inverse transform is:

$$x' = r'\frac{1 - \tan\frac{\varphi'}{2}^2}{1 + \tan\frac{\varphi'}{2}^2} = \frac{\sqrt{v^2 - z^2}}{\cosh u}\sinh u \tag{7.3.4a}$$

$$y' = r'\frac{2\tan\frac{\varphi'}{2}}{1 + \tan\frac{\varphi'}{2}^2} = \frac{\sqrt{v^2 - z^2}}{\cosh u} \tag{7.3.4b}$$

With z, the integral domain in the $u - v$ plane is no longer rectangular.

$$u_L = -\ln\tan\frac{\varphi'_L}{2} = \ln\left(\sqrt{1 + \left(\frac{x_L}{h}\right)^2} + \frac{x_L}{h}\right)$$

$$u_U = -\ln\tan\frac{\varphi'_U}{2} = \ln\left(\sqrt{1 + \left(\frac{x_U}{h}\right)^2} + \frac{x_U}{h}\right) \tag{7.3.5}$$

$$v_L = |z|$$

$$v_U(u) = \sqrt{z^2 + (h\cosh u)^2}$$

However, the integral I can still be evaluated using the same set of basic 1D quadrature rules as in Eq.(7.2.12)

$$I = \iint_{\Delta P_0 P_1 P_2} dx\, dy \frac{f(x,y)}{\sqrt{(x - x_0)^2 + (y - y_0)^2 + z^2}}$$

$$= (u_U - u_L)\sum_{i,j}^{N_u, N_v} \frac{v_U - v_L}{\cosh u^i} w_u^i w_v^j f(x,y)\big|_{(u^i, v^{i,j})} \tag{7.3.6}$$

$$= (u_U - u_L)\sum_{i,j}^{N_u, N_v} \left[\sqrt{h^2 + \left(\frac{|z|}{\cosh u^i}\right)^2} - \frac{|z|}{\cosh u^i}\right] w_u^i w_v^j f(x,y)\big|_{(u^i, v^{i,j})}$$

The resulted summary of equations is:

$$
I = \iint\limits_{\Delta P_0 P_1 P_2} dx\,dy \frac{f(x,y)}{\sqrt{(x-x_0)^2 + (y-y_0)^2 + z^2}}
\tag{7.3.7a}
$$

$$
= (u_U - u_L) \sum_{i,j}^{N_u, N_v} W^{i,j} f(x,y)\big|_{(u^i, v^{i,j})}
$$

$$
W^{i,j} = \left[\sqrt{h^2 + (|z|/\cosh u^i)^2} - |z|/\cosh u^i\right] w_u^i w_v^j
\tag{7.3.7b}
$$

$$
x_1' = [(x_2 - x_1)(x_1 - x_0) + (y_2 - y_1)(y_1 - y_0)]/|\overrightarrow{P_1 P_2}|
\tag{7.3.7c}
$$

$$
x_2' = [(x_2 - x_1)(x_2 - x_0) + (y_2 - y_1)(y_2 - y_0)]/|\overrightarrow{P_1 P_2}|
\tag{7.3.7d}
$$

$$
y_1' = y_2' = [(x_1 - x_2)y_0 + (x_2 - x_0)y_1 + (x_0 - x_1)y_2]/|\overrightarrow{P_1 P_2}|
\tag{7.3.7e}
$$

$$
h = |y_1'| = |y_2'|
\tag{7.3.7f}
$$

$$
x_L = \min\{x_1', x_2'\}
\tag{7.3.7g}
$$

$$
x_U = \max\{x_1', x_2'\}
\tag{7.3.7h}
$$

$$
u_L = \ln\left[\sqrt{1 + (x_L/h)^2} + x_L/h\right]
\tag{7.3.7i}
$$

$$
u_U = \ln\left[\sqrt{1 + (x_U/h)^2} + x_U/h\right]
\tag{7.3.7j}
$$

$$
v_L = |z|
\tag{7.3.7k}
$$

$$
v_U(u) = \sqrt{z^2 + (h\cosh u)^2}
\tag{7.3.7l}
$$

$$
\begin{pmatrix} x \\ y \end{pmatrix}\bigg|_{(u^i, v^{i,j})} = \begin{pmatrix} x_0 \\ y_0 \end{pmatrix} + \frac{1}{|\overrightarrow{P_1 P_2}|} \begin{pmatrix} x_2 - x_1 & -y_2 + y_1 \\ y_2 - y_1 & x_2 - x_1 \end{pmatrix} \frac{\sqrt{{v^{i,j}}^2 - z^2}}{\cosh u^i} \begin{pmatrix} \sinh u^i \\ 1 \end{pmatrix}
\tag{7.3.7m}
$$

$$
\begin{pmatrix} u^i \\ v^{i,j} \end{pmatrix} = = \begin{pmatrix} u_L + u_U\ \xi_u^i \\ v_L + \sqrt{z^2 + (h\cosh u^i)^2}\ \xi_v^j \end{pmatrix}
\tag{7.3.7n}
$$

7.4   Duffy Transform

8   Quadrature Result

8.1   Quadrature Rules on Triangle Domain

David Dunavant and Stephen Wandzura are the two important names. There are well developed C++ codes available:

http://people.sc.fsu.edu/~jburkardt/cpp_src/wandzura/wandzura.html

http://people.sc.fsu.edu/~jburkardt/cpp_src/dunavant/dunavant.html

I have made the Dunavant codes compilable by C compilers and linked it through LibraryLink into Mathematica. One can dynamically compute Dunavant rules by calling from Mathematica.

The standard quadrature rules for triangles are expressed as the abscissas and weights for the standard triangle:

$$
\{(\xi_i, \zeta_i), w_i\}_{i=1}^N
\tag{8.1.1}
$$

The standard triangle is the triangle formed by joining the three nodes

$$
\begin{aligned}
&(0,0) \\
&(1,0) \\
&(0,1)
\end{aligned}
\tag{8.1.2}
$$

The standard weights, w_i, are normalized to unity:

$$\sum_{i=1}^{M} w_i = 1 \tag{8.1.3}$$

The integration on general triangle $P_0(x_0, y_0), P_1(x_1, y_1), P_2(x_2, y_2)$ is approximated using the quadrature rule as:

$$\int dx dy f(x, y) = \text{Area} \sum_{i=1}^{M} w_i f(x_i, y_i) \tag{8.1.4}$$

The abscissas are given by

$$\begin{pmatrix} x_i \\ y_i \end{pmatrix} = \begin{pmatrix} x_1 - x_0 & x_2 - x_0 \\ y_1 - y_0 & y_2 - y_0 \end{pmatrix} \begin{pmatrix} \xi_i \\ \zeta_i \end{pmatrix} + \begin{pmatrix} x_0 \\ y_0 \end{pmatrix} \tag{8.1.5}$$

8.2 Calculating Matrix Elements - Quadrature

For the identity term, $A_{(nm)(n'm')}$, Eq.(4.1.2) is exact:

$$A_{(nm)(n'm')} = 2\pi S(n)\delta_{nn'}\delta_{mm'} \tag{8.2.1}$$

The interaction term, $B_{(nm)(n'm')}$, needs greater care.

$$
\begin{aligned}
B^t_{(nm)(n'm')} &= \int d\boldsymbol{r} d\phi S_n(\boldsymbol{r}) e^{-im\phi} \int d\boldsymbol{r}' d\phi' g(\boldsymbol{r}, \phi; \boldsymbol{r}', \phi') \mu_t(\boldsymbol{r}') S_{n'}(\boldsymbol{r}') e^{+im'\phi'} \\
&= \int d\boldsymbol{r} d\phi S_n(\boldsymbol{r}) e^{-im\phi} \int d\boldsymbol{r}' d\phi' S_{n'}(\boldsymbol{r}') e^{+im'\phi'} \mu_t(\boldsymbol{r}') \frac{1}{|\boldsymbol{r} - \boldsymbol{r}'|} \delta(\phi - \phi_{\boldsymbol{r}-\boldsymbol{r}'})\delta(\phi' - \phi_{\boldsymbol{r}-\boldsymbol{r}'}) \\
&= \int d\boldsymbol{r} S_n(\boldsymbol{r}) \int d\boldsymbol{r}' S_{n'}(\boldsymbol{r}') \mu_t(\boldsymbol{r}') \frac{e^{+i(m'-m)\phi_{\boldsymbol{r}-\boldsymbol{r}'}}}{|\boldsymbol{r} - \boldsymbol{r}'|}
\end{aligned}
\tag{8.2.2}
$$

The result for $B^s_{(nm)(n'm')}$ is similar:

$$B^s_{(nm)(n'm')} = -\int d\boldsymbol{r} S_n(\boldsymbol{r}) \int d\boldsymbol{r}' S_{n'}(\boldsymbol{r}') \mu_s(\boldsymbol{r}') g^{|m'|} \frac{e^{+i(m'-m)\phi_{\boldsymbol{r}-\boldsymbol{r}'}}}{|\boldsymbol{r} - \boldsymbol{r}'|} \tag{8.2.3}$$

Add Eq.(8.2.2) and Eq.(8.2.3) to get

$$B_{(nm)(n'm')} = \int d\boldsymbol{r} S_n(\boldsymbol{r}) \int d\boldsymbol{r}' S_{n'}(\boldsymbol{r}') \left[\mu_t(\boldsymbol{r}') - \mu_s(\boldsymbol{r}') g^{|m'|} \right] \frac{e^{+i(m'-m)\phi_{\boldsymbol{r}-\boldsymbol{r}'}}}{|\boldsymbol{r} - \boldsymbol{r}'|} \tag{8.2.4}$$

In Eq.(8.2.4), there is a *testing* integral $\int d\boldsymbol{r}$, and a *source* integral $\int d\boldsymbol{r}'$. In previous sections, we dealt extensively with single-point testing and single-point source. In this section, we advance to multi-point testing and multi-point source.

$$
\begin{aligned}
B_{(nm)(n'm')} &= S(n)S(n') \sum_{j_n=1}^{M_n} \sum_{j_{n'}=1}^{M_{n'}} w_{j_n} w_{j_{n'}} \left[\mu_t(\boldsymbol{r}_{j_{n'}}) - \mu_s(\boldsymbol{r}_{j_{n'}}) g^{|m'|} \right] \frac{e^{+i(m'-m)\phi_{\boldsymbol{r}_{j_n} - \boldsymbol{r}_{j_{n'}}}}}{|\boldsymbol{r}_{j_n} - \boldsymbol{r}_{j_{n'}}|} \\
&= \mathrm{C}_t(n, n', m - m') + \mathrm{C}_s(n, n', m - m') g^{|m'|}
\end{aligned}
\tag{8.2.5}
$$

where

$$\mathrm{C}_t(n, n', \Delta m) = + S(n)S(n') \sum_{j_n=1}^{M_n} \sum_{j_{n'}=1}^{M_{n'}} w_{j_n} w_{j_{n'}} \mu_t(\boldsymbol{r}_{j_{n'}}) \frac{e^{-i\Delta m \phi_{\boldsymbol{r}_{j_n} - \boldsymbol{r}_{j_{n'}}}}}{|\boldsymbol{r}_{j_n} - \boldsymbol{r}_{j_{n'}}|} \tag{8.2.6a}$$

$$\mathrm{C}_s(n, n', \Delta m) = - S(n)S(n') \sum_{j_n=1}^{M_n} \sum_{j_{n'}=1}^{M_{n'}} w_{j_n} w_{j_{n'}} \mu_s(\boldsymbol{r}_{j_{n'}}) \frac{e^{-i\Delta m \phi_{\boldsymbol{r}_{j_n} - \boldsymbol{r}_{j_{n'}}}}}{|\boldsymbol{r}_{j_n} - \boldsymbol{r}_{j_{n'}}|} \tag{8.2.6b}$$

The M_n is the number of sampling points for S_n. The $M_{n'}$ is the number of sampling points for $S_{n'}$.

Eq.(8.2.5) and Eq.(8.2.6) indicate that each matrix block $B_{(nm)(n'm')}$ is Toeplitz-like. The matrix-vector (M-V) multiplication of $B(n, n')$ can be done at the cost of $O(N \log(N))$ in time and $O(N)$ in storage.

Use the quadrature rules to evaluate the V_{sc} in Eq.(6.2.11c):

$$
\begin{aligned}
(V_{sc})_{nm} &= \int d\boldsymbol{r}\, S_n(\boldsymbol{r}) \int d\boldsymbol{r}' \frac{e^{-im\phi_{\boldsymbol{r}-\boldsymbol{r}'}}}{|\boldsymbol{r}-\boldsymbol{r}'|} \mu_s(\boldsymbol{r}') f(\phi_{\boldsymbol{r}-\boldsymbol{r}'} - \phi^I) e^{-\tau(\boldsymbol{r}', -\hat{\boldsymbol{s}}^I)} \\
&= \sum_{n'=1}^{N_s} S(n) S(n') \sum_{j_n=1}^{M_n} \sum_{j_{n'}=1}^{M_{n'}} w_{j_n} w_{j_{n'}} \mu_s(\boldsymbol{r}_{j_{n'}}) \frac{e^{-im\phi_{\boldsymbol{r}_{j_n}-\boldsymbol{r}_{j_{n'}}}}}{|\boldsymbol{r}_{j_n}-\boldsymbol{r}_{j_{n'}}|} f(\phi_{\boldsymbol{r}_{j_n}-\boldsymbol{r}_{j'_n}} - \phi^I) e^{\tau(\boldsymbol{r}_{j'_n}, -\hat{\boldsymbol{s}}^I)}
\end{aligned}
\tag{8.2.7}
$$

Note that we are using the convention that the weights are normalized to unity.

9   Code Implementation

9.1   Mesh Generation

9.2   Pre-Computation

9.3   Fast M-V Multiplication

9.4   Iterative Solver

9.5   First Order - ψ_b

9.6   Second Order - ψ_b and ψ_{sb}

9.7   Quadrature Rules

10   What's Next?