A FAST FORWARD SOLVER OF RADIATIVE TRANSFER EQUATION

HAO GAO AND HONGKAI ZHAO

Abstract. In this paper we develop an efficient forward solver for steady-state or frequency-domain radiative transfer equation (RTE) on 2D and 3D structured and unstructured meshes with vacuum boundary condition or reflection boundary condition. In our algorithm we use a direct angular discretization and upwind type of spatial discretization that preserves properties of both scattering and differential operators of RTE. To solve the large linear system after discretization, we construct an efficient iterative scheme based on Gauss-Seidel and proper angular dependent ordering. With this iterative scheme as the relaxation we implement various multigrid methods in both angular and physical space. Our algorithm can deal with different scattering regimes efficiently. Although we emphasize applications in optical imaging, our method may be applied to more general RTEs, e.g., other types of scattering function. Efficiency and accuracy of our algorithm is demonstrated by comparison with both analytical solutions and Monte Carlo solutions, and various numerical tests in optical imaging.

1. Introduction

Many researchers have been studying the numerical solutions to the radiative transport equation (RTE) or the within-group neutron transport equation [4, 13] in the field of neutron transport [4], atmospheric radiative transfer [1], heat transfer [16] and optical imaging [2, 17, 9]. In this paper, we mainly study the steady-state RTE

(1.1)
$$\hat{s}\cdot\nabla\Phi(\vec{r},\hat{s})+\mu_t\Phi(\vec{r},\hat{s})=\mu_s\oint_{\hat{S}^{n-1}}f(\hat{s},\hat{s}')\Phi(\vec{r},\hat{s}')d\hat{s}'+q,\ \vec{r}\in\Omega\subset\Re^n,\hat{s}\in\hat{S}^{n-1}$$

where the quantities in the equation are the photon flux Φ which depends on both space $\vec{r}$ and angle $\hat{s}$, the light source q, the scattering function or the phase function f, the absorption coefficient μ_a, the scattering coefficient μ_s, the transport coefficient $\mu_t=\mu_a+\mu_s$, and $\hat{S}^{n-1}$, the unit circle when $n=2$ or the unit sphere when $n=3$.

Key words and phrases. radiative transfer equation (RTE), optical imaging, discrete ordinate method, source iteration, Henyey-Greenstein function, anisotropic scattering, multigrid, forward solver, discontinuous Galerkin method.

To simplify the discussion, we shall first consider the following vacuum boundary condition

$$\Phi(\vec{r},\hat{s})=0,\quad (\vec{r},\hat{s})\in\partial\Omega\times\hat{S}^{n-1},\ \text{if}\ \hat{s}\cdot\hat{n}<0 \tag{1.2}$$

where $\hat{n}$ is the outer normal to $\partial\Omega$ at $\vec{r}$. The reflection boundary condition will be addressed in Section 2.5.

The scattering function usually only depends on the angle θ between $\hat{s}$ and $\hat{s}'$. A popular model for the anisotropic scattering in optical imaging is the normalized Henyey-Greenstein scattering function (H-G)

$$f(\theta)=\begin{cases}\frac{1-g^2}{2\pi(1+g^2-2g\cos\theta)}, & n=2\\ \frac{1-g^2}{4\pi(1+g^2-2g\cos\theta)^{1.5}}, & n=3\end{cases}, \tag{1.3}$$

where g is the anisotropic factor that ranges from 0 to 1, indicating the strength of forward-peaking scattering. The larger the value of g is, the stronger forward-peaking the scattering is. Although we use this particular scattering function in our numerical tests, our algorithm works for other scattering functions as well.

In general numerical methods for solving RTE can be classified into two types, one is deterministic, i.e., based directly on the differential-integral equation; the other one is based on probabilistic formulation of RTE such as Monte Carlo methods. In this paper we are proposing an efficient deterministic method to solve RTE. We show some comparison of our method and Monte Carlo method in Section 3.7.

Disretization of RTE (1.1) involves discretization in both spatial and angular spaces. Popular spatial discretizations include the diamond scheme and variant versions of it [4, 11], the upwind scheme [2] and other finite difference schemes (FD), the finite element method (FEM) [17, 20], the finite volume method (FVM) [10], the Discontinuous Galerkin method (DG) [18, 7, 3]. The diamond scheme and FD are suitable for structured grids while FEM, FVM and DG can be used on unstructured grids. Also various angular discretizations exist: the P_n method [13], the FEM [17, 20], the discrete ordinate method (DOM) [4, 16, 2] and others [12, 15], and many of them use orthogonal bases to expand the integral term in RTE and represent it by summation in orthogonal bases, such as the Legendre polynomials, the spherical harmonics or the wavelets. However, these orthogonal base expansion methods are not local in angular space and may not be efficient to capture strong localized scattering between the radiances from neighboring angles

such as in the presence of strong forward-peaking scattering. Moreover, general boundary conditions on influx and outflux can be also difficult to cope with.

Among all the numerical schemes on structured grids, the DOM when combined with FD or the diamond scheme, which is then solved by source-iteration (SI) [4], is the most popular, because of its simplicity. However, there are two major issues:

(1) The conventional DOM presents difficulty dealing with strong forward-peaking scattering as mentioned above.
(2) The SI converges slowly in the scattering-dominating optical-thick regime. Acceleration techniques, such as diffusion synthetic acceleration and spatial multigrid [14], may be used to accelerate iterative convergence.

The mathematical theory about the RTE (1.1) is quite clear [13, 4]. The solution has both maximum principle and energy estimate due to nice properties of the transport operator and the integral operator. However, numerical computation is the only way to give accurate quantitative information of the solution in practice since there is no hope of analytical solution in general. The main challenge for solving RTE numerically is caused by its large dimension of independent variables. For the steady state RTE, which is a boundary value problem, the scattering (integral term) couples the solution at different location and different angle together. This means that we have to solve a large and not too sparse (due to the integral term) linear system after discretization. For this large linear system, iterative method is usually the only choice. Hence a faster iterative strategy is of crucial importance. On the other hand, RTE can behave quite differently in different regions if the computation domain is large compared with the mean free path ($1/\mu_t$), e.g., either more transport-like near the source, or more diffusion-like after a significant number of scattering. More importantly, these regions of different behaviors can not be clearly defined as a priori knowledge. This poses a major difficulty for designing a fast iterative solver in the whole domain. For example, diffusion synthetic acceleration method, which is based on diffusion approximation of RTE may not be effective in transport region, especially when the scattering is very anisotropic such as in forward-peaking case in optical imaging. In this paper, we would like to address these issues by

(1) introducing a direct angular discretization which preserves nice properties of the integral operator, captures localized scattering and deals with general boundary condition effectively,
(2) constructing an efficient iterative scheme (based on Gauss-Seidel iteration and angular dependent ordering) with proper spatial discretization (upwind type) and angular discretization which deals with transport effectively and has nice smoothing property,
(3) applying multigrid methods (with the above iterative scheme as relaxation) in both angular and physical space which can deal with diffusion effectively.

With all these ingredients, we can handle very general setup for RTE, e.g., structured or unstructured grids, geometry and boundary condition, and improve the conventional SI by 20-30 times consistently in all regimes.

Next we will describe our method in detail. Here is the outline. We introduce the angular discretization in Section 2.1. Then we present the efficient iterative method in Section 2.2 and acceleration by multigrid in Section 2.3. We describe our algorithm for unstructured meshes in Section 2.4 and treatment of reflection boundary condition in Section 2.5. Finally we show both efficiency and accuracy of our algorithm by extensive tests on analytical solutions as well as examples from optical imaging in Section 3.

2. Numerical algorithm

2.1. **Angular discretization.**

2.1.1. *Motivation.* In the conventional discrete ordinate method, the integral term in RTE is approximated by a finite summation of the weighted fluxes on the discrete ordinates through Legendre or spherical harmonic expansion of the phase function and the photon radiance. The weights come from the matching of the truncated leading-order coefficients in the expansion. There are a few drawbacks for this representation. Since the expansion is not local in angular space, this leads to both inaccuracy and inefficiency when dealing with highly forward-peaking phase function. Moreover, localized scattering interaction in angular space is not captured well in this representation. It actually causes strong coupling of different terms in the expansion which may make the linear system more difficult to solve. Moreover,

due to oscillatory nature of the expansion, non-physical oscillations, such as ray effects, may occur in the solution for forward peaking or strong transport regime [4]. Also boundary condition prescribing flux condition in partial angular space is difficult to implement. These difficulties are more eminent in optical imaging. since the commonly used phase function is H-G and g is usually around 0.8-0.95, which means highly forward-peaking.

To address the above issues, we triangulate the angular space (unit circle in 2D and unit sphere in 3D) and use a finite element basis to represent the solution in angular space piecewisely. So our representation in angular space is local.

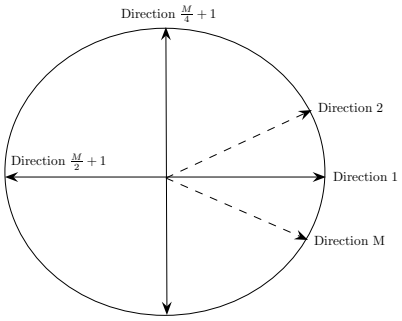

FIGURE 1. Angular discretization in 2D

2.1.2. *2D scheme.* We divide the angular space $[0, 2\pi)$ uniformly into M directions $\Phi_m, m = 1, \cdots, M$ with equal interval length $\triangle\theta$ as shown in Figure 1 and use the piecewise linear approximation of the photon radiance,

$$\Phi(x, y, \theta) \approx \sum_{m=1}^{M} \Phi_m(x, y) L_m(\theta) \tag{2.1}$$

where $L_m(\theta)$ is a piecewise linear function in θ and $L_m(\theta_{m'}) = \begin{cases} 1, & m = m' \\ 0, & otherwise \end{cases}$. With this discretization in angle, the RTE (1.1) is reduced to a set of partial

differential equations in space:

(2.2)
$$\cos\theta_m \frac{\partial \Phi_m}{\partial x} + \sin\theta_m \frac{\partial \Phi_m}{\partial y} + \mu_t \Phi_m = \mu_s \oint f(\theta_m - \theta')\Phi(\theta')d\theta' + q, m = 1, \cdots, M$$

Next we shall approximate the integral on the right hand by a summation through the following three steps. The first step is to approximate the phase function f by piecewise linear interpolation, $\tilde{f}$, on the same angular bases, i.e.,$\tilde{f}(\theta_m) = f(\theta_m)$. In the second step, we plug (2.1) in the integral and get

$$\oint \tilde{f}(\theta_m - \theta')\Phi(\theta')d\theta' = \sum_{m'=1}^{M} w_{mm'}\Phi_{m'} \tag{2.3}$$

where

$$w_{mm'} = \oint \tilde{f}(\theta_m - \theta')L_{m'}(\theta')d\theta' \tag{2.4}$$

Then we compute all $w_{mm'}$'s for a fixed direction m. Suppose $m = 1$ and we know $\tilde{f}(\theta_m, \theta_{m'})$ is symmetric with respect to m. (2.4) can be explicitly integrated to get

$$w_{1i} = \begin{cases} \frac{\Delta\theta}{3}(2f(\theta_1) + f(\theta_2)), & i = 1 \\ \frac{\Delta\theta}{6}(f(\theta_{i-1}) + 4f(\theta_i) + f(\theta_{i+1})), & i = 2, \cdots, \frac{M}{2} \\ \frac{\Delta\theta}{3}(2f(\theta_{\frac{M}{2}}) + f(\theta_{\frac{M}{2}+1})), & i = \frac{M}{2} + 1 \\ w_{1,M+2-i}, & i = \frac{M}{2} + 2, \cdots, M \end{cases} \tag{2.5}$$

To be consistent with the fact that the integration of phase function is one, we scale w_{1i} by the same constant so that $\sum_{i=1}^{M} w_{1i} = 1$. Since the angular mesh is symmetric and the phase function only depends on angles in between, the last step is to simply rotate the frame to align the direction m with direction 1 in order to get $w_{mm'}$'s for each m by shifting the $w_{1m'}$'s. Finally, we have a set of coupled transport equations for 2D RTE after angular discretization

$$\cos\theta_m \frac{\partial \Phi_m}{\partial x} + \sin\theta_m \frac{\partial \Phi_m}{\partial y} + \mu_t \Phi_m = \mu_s \sum_{m'=1}^{M} w_{mm'}\Phi_{m'} + q, m = 1, \cdots, M \tag{2.6}$$

2.1.3. *3D scheme.* For 3D unit sphere, we take eight triangular planes, each of which is in a different quadrant. The mesh on each triangular plane is shown in Figure 2 and then the mesh is projected onto the unit sphere as shown in Figure 2. This triangulation has the following nice properties: first, it is simple; second, it is quite uniform as we show below; third, it is easy to coarsen and refine for multigrid method. Let M denote the number of intervals on each side of the triangular plane, N denote the total number of vertices for the angular mesh. We have the following relation $N = 4M^2 + 2$. For example, $M = 4$ and $N = 66$ in Figure 2. Table 1

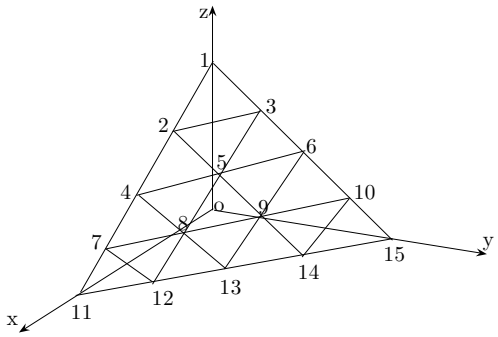

FIGURE 2. The projection of triangulation of a unit sphere in 1^{st} quadrant down to a triangular plane when $M = 4$

TABLE 1. Quantities for angular meshes. M_n: the angular mesh level with 2^n intervals, n: the index n in M_n, N_p: number of vertices, N_t: number of elements, A_i: area of the i^{th} element, $\bar{A}$: the average area, $\triangle$: standard deviation of element area, A_{min}: the minimum area, A_{max}: the maximum area, and $\delta = 1 - \sum_i \frac{A_i}{4\pi}$.

n	N_p	N_t	$\bar{A}$	Δ	$\Delta/\bar{A}$	$\frac{A_{min}}{A_{max}}$	δ
1	6	8	5.00e-1	0.00e0	0.000	1.000	6.82e-1
2	18	32	2.76e-1	4.48e-2	0.163	0.707	2.97e-1
3	66	128	8.95e-2	2.43e-2	0.272	0.367	8.82e-2
4	258	512	2.40e-2	7.49e-3	0.313	0.258	2.33e-2
5	1026	2048	6.10e-3	1.97e-3	0.323	0.221	5.92e-3
6	4098	8192	1.53e-3	4.99e-4	0.326	0.205	1.49e-3

shows quality of the mesh in term of area variation among elements. The area of each element is computed as the area of the triangle after projection of the vertices onto the unit sphere. We can see that the difference between the largest element and the smallest element as well as the the deviation with respect to the average become stabilized, as the mesh is further refined.

Given the above triangulation of the unit sphere, we can construct piecewise linear or quadratic approximation of the phase function and piecewise linear finite element basis in angular space for the photon flux. The standard assembly method in FEM is used to compute $w_{mm'}$'s defined as in (2.4), in which each curved element

on sphere is approximated by a triangular element and f is approximated by $\tilde{f}$. Also we normalize the sum $\sum_{m'} w_{mm'} = 1$ for all m.

For example, to compute $w_{mm'}$'s for $m = 10$, we first rotate the direction m to the z direction and let $(\hat{x}, \hat{y}, \hat{z})$ be the coordinates after the rotation. Note that the value of f only depends on $\hat{z}$ since the phase function only depends on the angle between two directions. In the case that two vertices in one element have the equal $\hat{z}$'s, the values of f at two different $\hat{z}$'s are used to linearly interpolate f in this element; otherwise, the values of f at three different $\hat{z}$'s are used to construct piecewise linear (Figure 3) or quadratic approximation $\tilde{f}$. Then the explicit integrations on each element can be carried out and the weights result from assembly.

Remark If piecewise linear approximation is used for f in all elements, all the weights $w_{mm'}$ are positive. On the other hand, the sum is normalized to one which preserves the basic property of the integral operator. With this angular discretization, we can capture the correct behavior in angular space up to the level of resolution. For example, in the case of extreme forward peaking, even if the mesh can not resolve the shape of the phase function well, we can still localize the interaction in the angular space and preserve the positivity of the weights. Later we will see that together with a proper spatial discretization for the transport operator, the fully discretized linear system preserves basic properties of the continuous RTE. If piecewise quadratic approximation is used for f, we can not guarantee the positivity of all weights. However we did not experience such a problem in our numerical test. In terms of accuracy, piecewise linear approximation is almost as good as quadratic one since the solution in angular space is approximated piecewise linearly.

2.2. **Improved Source-Iteration.** The conventional source-iteration is

$$\cos\theta_m \frac{\partial \Phi_m^{n+1}}{\partial x} + \sin\theta_m \frac{\partial \Phi_m^{n+1}}{\partial y} + \mu_t \Phi_m^{n+1} = \mu_s \sum_{m'=1}^{M} w_{mm'} \Phi_{m'}^{n} + q \tag{2.7}$$

in which the scattering term in the $(n+1)^{th}$ iteration is completely from n^{th} iteration, i.e., scattering effect is lagged behind transport. The physical interpretation of source iteration is that at $(n+1)^{th}$ iteration, those photons that have scattering events fewer than or equal to n are captured [4]. For each fixed direction, the above

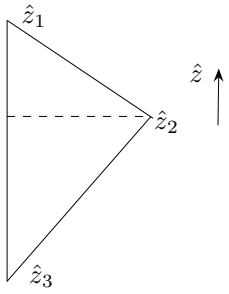

FIGURE 3. piecewise linear approximation: $\tilde{f}(\hat{z})$ is linear from $\hat{z}_1$ to $\hat{z}_2$ and linear from $\hat{z}_2$ to $\hat{z}_3$ in a triangle with $\hat{z}_1 > \hat{z}_2 > \hat{z}_3$.

transport equation can be solved efficiently. However, conventional SI does not utilize those available updated information in other directions effectively during the computation along each direction. In the following we improve the conventional SI method in several aspects and show improvement by numerical tests later.

2.2.1. *Angular Gauss-Seidel method (Method 1).* The first improvement is by using angular Gauss-Seidel iteration as

(2.8)
$$\cos\theta_m \frac{\partial \Phi_m^{n+1}}{\partial x} + \sin\theta_m \frac{\partial \Phi_m^{n+1}}{\partial y} + \mu_t \Phi_m^{n+1} = \mu_s \sum_{m'=1}^{m-1} w_{mm'} \Phi_{m'}^{n+1} + \mu_s \sum_{m'=m}^{M} w_{mm'} \Phi_{m'}^{n} + q.$$

Therefore, in the scattering term information from directions that have been updated is used. Hence interactions among different directions are captured more effectively which results in fewer number of iterations. Numerical simulations in Section 3.2 show that the iteration number of this improvement reduces the number of iterations to around one half to two thirds of that of conventional SI. On the other hand, the computational cost of one iteration does not increase.

2.2.2. *Combining the dominating scattering term (Method 2).* In the forward-peaking regime, after discretization of the scattering term, the most dominating scattering weight for each direction m is w_{mm}. We simply move it to the left-hand side to get

a full angular Gauss-Seidel iteration:

(2.9)
$$\cos\theta_m \frac{\partial \Phi_m^{n+1}}{\partial x} + \sin\theta_m \frac{\partial \Phi_m^{n+1}}{\partial y} + (\mu_t - \mu_s w_{mm})\Phi_m^{n+1} = \mu_s \sum_{m'=1}^{m-1} w_{mm'}\Phi_{m'}^{n+1} + \mu_s \sum_{m'=m+1}^{M} w_{mm'}\Phi_{m'}^{n} + q.$$

The third term on the left-hand side of (2.9) gives the correct decaying rate of the radiance Φ along each transport direction. In SI, the original decaying factor μ_t gives the false decaying rate, which causes a slow convergence rate. For example, in the highly forward-peaking scattering regime, the ratio of two decaying rates is $\frac{\mu_t - \mu_s w_{mm}}{\mu_t} \approx 1 - w_{mm}$, which can be quite different from 1 when $\mu_a \ll \mu_s$. We show that use of most updated information will accelerate convergence of the standard SI method. Numerical tests in section 3.2 show that the iteration number of Method 2 uses fewer number of iterations than Method 1. In particular the number of iterations is reduced more than half in strong forward-peaking scattering regime. At the same time the computational time of one iteration does not increase.

Remark With normalization of the sum of $w_{mm'}$ to be 1, $\mu_t - \mu_s w_{mm} > 0$.

2.2.3. *Spatial upwind discretization and ordering.* For a given θ_m, the transport equation has a fixed direction of characteristics in space. On a rectangular grid, simple upwind difference scheme can be used to follow the direction of characteristics. As an example in 2D, if $\theta_m \in (0, \frac{\pi}{2}]$ and a simple upwind scheme, i.e., backward difference, is used for $\frac{\partial}{\partial x}$ and $\frac{\partial}{\partial y}$, the fully discretized linear system for the corresponding transport equation looks like

(2.10)
$$(a + b + \mu_t - \mu_s w_{mm})\phi_{i,j,m} - (a\phi_{i-1,j,m} + b\phi_{i,j-1,m}) - \mu_s \sum_{m' \neq m} w_{mm'}\phi_{i,j,m'} = q_{i,j,m}$$

where

$$a = \frac{\cos\theta_m}{\triangle x} \geq 0,\ b = \frac{\sin\theta_m}{\triangle y} \geq 0,\ \text{for } 0 \leq \theta_m < \frac{\pi}{2}$$

Note that

$$\mu_t > \mu_s = \mu_s \sum_{m'} w_{mm'}$$

So with our angular discretization and proper upwind scheme for transport operator, the fully discretized linear system has the following nice property: the diagonal element has the opposite sign to all off-diagonal elements in each row and its absolute value is larger than the sum of absolute values of off-diagonal elements, i.e., we

get a M-matrix system. This implies the system (1) is non-degenerate, (2) has a discrete maximum principle, and (3) iterative methods will converge and use of most updated information, e.g., Gauss-Seidel method, will accelerate the convergence.

One can accelerate the convergence further by taking advantage of the hyperbolic nature of transport equation and use of upwind scheme through a proper angular dependent ordering of the spatial nodes according to the transport direction. For rectangular grids, the ordering is very simple. For example, in 2D, if $\theta_m \in (0, \frac{\pi}{2}]$, we sweep through the spatial index in the order of $i = 1, \cdots, N_x, j = 1, \cdots, N_y$, where N_x, N_y are the dimensions in x, y directions respectively, and we have the following update formula

(2.11)
$$\phi^n_{i,j,m} = \frac{a\phi^n_{i-1,j,m} + b\phi^n_{i,j-1,m} + \mu_{s_{ij}}(\sum_{m'=1}^{m-1} w_{mm'}\phi^n_{i,j,m'} + \sum_{m'=m+1}^{M} w_{mm'}\phi^{n-1}_{i,j,m'}) + q_{i,j,m}}{a + b + \mu_{t_{ij}} - \mu_{s_{ij}} w_{mm}}.$$

Note that with this spatial ordering, when we update $\phi^n_{i,j,m}$, both $\phi^n_{i-1,j,m}$ and $\phi^n_{i,j-1,m}$ have already been updated. Hence the information propagates correctly along the transport path. For simple transport equation, i.e., without the scattering term, the ordering will transform the linear system into a triangular system and one sweep will solve it. If $\theta_m \in (\frac{\pi}{2}, \pi]$ then the spatial ordering should be $i = N_x, \cdots, 1, j = 1, \cdots, N_y$ instead. In 2D there are four different orderings corresponding to four quadrants in which the direction $(\cos\theta_m, \sin\theta_m)$ belongs to. For 3D rectangular grids there are 8 orderings corresponding to 8 quadrants. We call this scheme (2.11) improved source-iteration (ISI) to distinguish it from the standard SI.

In Section 2.4 we will discuss in details about discontinuous Galerkin discretization and ordering for unstructured grids.

Remark In the above we show that Gauss-Seidel iteration with proper ordering can capture transport part of RTE effectively. For example, ISI alone can deal with forward peaking very efficiently. Moreover, due to our angular and spatial discretization, the iteration is a smoothing process in both space and angles. However, non-transport component of the solution or long range interaction due to multiple scattering is not captured effectively by the iterative method. Also it is known that the low frequency component of the solution converges slowly for iterative method in the case of isotropic scattering [4]. So multigrid strategy with the above ISI as

the relaxation should be a good combination.

2.3. **Multigrid.** In scattering-dominating and optical-thick regime, the photon trajectory is scattered many times and therefore the photon distribution becomes diffusive after multiple scattering. The photon flux is spreading out in both space and angle. We propose to use the above improved SI method as an efficient relaxation technique and combine with multigrid strategy [19] in spatial and angular space. We point out that our angular discretization can capture short range and long range interaction in angular space effectively. By alternating relaxations on coarse and fine mesh, multigrid strategy can effectively accelerate the RTE solver. In this section, we consider various multigrid methods on structured mesh. The goal is to demonstrate that multigrid methods, including angular multigrid (AMG), spatial multigrid (SMG), and angular and spatial multigrid (MG), indeed accelerate the RTE solver dramatically in the scattering-dominating and optical-thick regime. Numerical simulations show that multigrid methods accelerate ISI consistently in various regimes. We leave descriptions of more complicated multigrid algorithm for unstructured mesh to Section 2.4.

Our multigrid methods differ from those in the literature [14] in many ways, such as discretization, relaxation technique, interpolation and restriction operators, and multigrid implementation for both angular and spatial variables.

In the following we describe the AMG, SMG and MG sequentially.

2.3.1. *Angular Multigrid (AMG).* In [8], angular multigrid was studied with respect to spectral expansion of the angular space. Here we consider AMG based on multilevel angular mesh directly. First we restrict our discussion in two dimensions. With a fixed spatial mesh we define a sequence of meshes in angular space, Ω_{n_a}, which divides the unit circle in a total number of intervals 2^{n_a} as described in Section 2.1. n_a denotes the level for angular mesh which is ranging between the coarsest level n_{a0} and the finest level n_{af} We further define angular interpolation operator $I_{n_a}^{n_a+1}$ from Ω_{n_a} to Ω_{n_a+1}

$$I_{n_a}^{n_a+1}\Phi_{2m} = \Phi_m,\ I_{n_a}^{n_a+1}\Phi_{2m+1} = \frac{\Phi_m + \Phi_{m+1}}{2}, \tag{2.12}$$

and angular restriction operator $I^{n_a}_{n_a+1}$ from Ω_{n_a+1} to Ω_{n_a}

$$I^{n_a}_{n_a+1}\Phi_m = \Phi_{2m} \text{ or } I^{n_a}_{n_a+1}\Phi_m = \frac{1}{4}\Phi_{2m-1} + \frac{1}{2}\Phi_{2m} + \frac{1}{4}\Phi_{2m+1}. \tag{2.13}$$

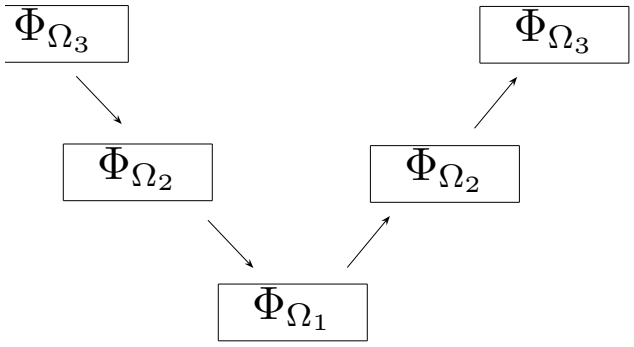

FIGURE 4. V-cycle on three layered meshes

We use the improved source iteration (ISI) (2.11) as the relaxation scheme in the V-cycle (Figure 4) AMG algorithm with an initial guess $[\Phi^0]_{n_a}$, which is usually a zero function:

$[\Phi]_{n_a} = AMGCYCLE(n_1, n_2, \Omega_{n_a}, [Q]_{n_a}, n_a, [\Phi^0]_{n_a})$

Step 1: relax $[\Phi]_{n_a}$ for n_1 times

Step 2: If $n_a \neq n_{a0}$

Compute the residual $[d]_{n_a}$

$[d]_{n_a-1} = I^{n_a-1}_{n_a}[d]_{n_a}$

$[\Phi]_{n_a-1} = AMGCYCLE(n_1, n_2, \Omega_{n_a-1}, [d]_{n_a-1}, n_a-1, 0)$

$[\Phi]_{n_a} = [\Phi]_{n_a} + I^{n_a}_{n_a-1}[\Phi]_{n_a-1}$

Step 3: relax $[\Phi]_{n_a}$ for n_2 times

Note that direct inversion of matrix is not used, even on the coarsest mesh. Our numerical experience shows that the convergence rate with direct inversion does not help the convergence, not even to say that it is more expensive.

Similarly, we define a sequence of 3D angular meshes on unit sphere using the proposed triangulation in Section 2.1: $M_{n_a},\ n_{a0} \leq n_a \leq n_{af}$. Again we define the

angular interpolation operator I_n^{n+1} from M_n to M_{n+1} by (Figure 5)

$$\begin{aligned} &I_n^{n+1}\Phi_i = \Phi_{i'}, i = 1,2,3, \\ &I_n^{n+1}\Phi_4 = \tfrac{\Phi_{1'}+\Phi_{2'}}{2},\ I_n^{n+1}\Phi_5 = \tfrac{\Phi_{1'}+\Phi_{3'}}{2},\ I_n^{n+1}\Phi_6 = \tfrac{\Phi_{2'}+\Phi_{3'}}{2}. \end{aligned} \tag{2.14}$$

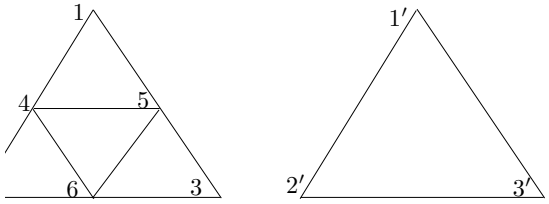

FIGURE 5. Vertices of t in M_{n+1} and t' in M_n

The angular restriction operator I_{n+1}^n from M_{n+1} to M_n is defined by

$$I_{n+1}^n\Phi_{i'} = \Phi_i,\ i = 1,2,3. \tag{2.15}$$

2.3.2. *Spatial Multigrid (SMG).* SMG algorithm for rectangular grid is really simple, and similar in 2D and 3D. Here we just specify the 2D spatial interpolation and restriction operators. Define a sequence of 2D meshes: $\Omega_{n_s} = (2^{n_{sx}}, 2^{n_{sy}}, N_{\hat{s}})$, $n_{sx0} \le n_{sx} \le n_{sxf}, n_{sy0} \le n_{sy} \le n_{syf}$. On these rectangular meshes we use the simplest piecewise constant DG approximation which is equivalent to first-order finite volume method with upwind scheme. Then, the spatial interpolation operator $I_{n_s}^{n_s+1}$ from Ω_{n_s} to Ω_{n_s+1} is

$$I_{n_s}^{n_s+1}\Phi_{2i-1,2j-1} = I_{n_s}^{n_s+1}\Phi_{2i-1,2j} = I_{n_s}^{n_s+1}\Phi_{2i,2j-1} = I_{n_s}^{n_s+1}\Phi_{2i,2j} = \Phi_{i,j} \tag{2.16}$$

and spatial restriction operator $I_{n_s+1}^{n_s}$ from Ω_{n_s+1} to Ω_{n_s} is

$$I_{n_s+1}^{n_s}\Phi_{i,j} = \frac{\Phi_{2i-1,2j-1} + \Phi_{2i-1,2j} + \Phi_{2i,2j-1} + \Phi_{2i,2j}}{4}. \tag{2.17}$$

2.3.3. *Multigrid in space and angle (MG).* Now we apply multigrid in both spatial and angular space. First, we define a sequence of meshes:

$$\Omega_{n_a,n_s} = (2^{n_{sx}}, 2^{n_{sy}}, 2^{n_a}),\ n_{a0} \le n_a \le n_{af},\ n_{sx0} \le n_{sx} \le n_{sxf},\ n_{sy0} \le n_{sy} \le n_{syf},$$

with $n_s = (n_{sx}, n_{sy})$.

There are at least four ways to combine AMG and SMG into a full multigrid algorithm (Figure 6).

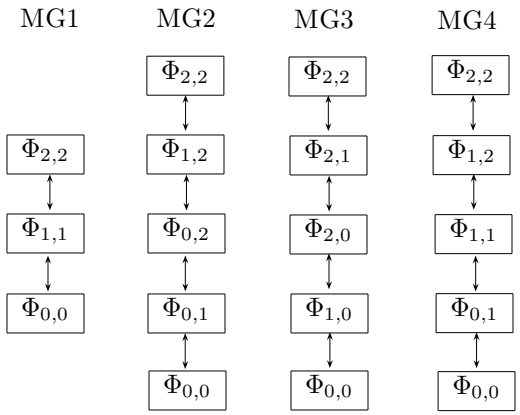

FIGURE 6. Multigrid methods

The first one, MG1, is to relax and interpolate solutions on two consecutive meshes Ω_{n_a,n_s} and Ω_{n_a+1,n_s+1}, i.e., the simultaneous angular and spatial multigrid. The second one, MG2, is to first apply AMG from $\Omega_{n_{af},n_{sf}}$ to $\Omega_{n_{a0},n_{sf}}$, and then apply SMG from $\Omega_{n_{a0},n_{sf}}$ to $\Omega_{n_{a0},n_{s0}}$. The third one, MG3, is to first apply SMG and then apply AMG. The fourth one, MG4, is to apply SMG and AMG alternatively. We show performance of different choices in Section 3.

2.4. **Algorithms on unstructured meshes.** After angular discretization of the steady state RTE (1.1), we get a system of coupled linear transport equations

$$\hat{s}_m \cdot \nabla \Phi_m + \mu_t \Phi_m = \mu_s \sum_{m'=1}^{M} w_{mm'} \Phi_{m'} + q, \ 1 \le m \le M, \tag{2.18}$$

where $\hat{s}_m := (s_1, s_2) = (\cos\theta_m, \sin\theta_m)$ in 2D and $\hat{s}_m := (s_1, s_2, s_3) = (\sin\theta_m \cos\phi_m, \sin\theta_m \sin\phi_m, \cos\theta_m)$ in 3D. DG method is a natural choice for transport equation on unstructured mesh which has many desirable properties including upwind nature and possible high order accuracy. For simplicity we start with piecewise constant DG.

2.4.1. *Spatial discretization: piecewise constant DG.* Let $T := \{\tau_i, i = 1, \cdots, N\}$ be a triangulation of the domain Ω, which is a triangular mesh in 2D or a tetrahedral

mesh in 3D, and $\{\varphi_i = 1_{\tau_i}, i = 1, \cdots, N\}$ be the bases. Then $\Phi_m \approx \sum_{i=1}^N \phi_{i,m}\varphi_i$, $\mu_t \approx \sum_{i=1}^N \mu_{t_i}\varphi_i$ and $\mu_s \approx \sum_{i=1}^N \mu_{s_i}\varphi_i$. In the following presentation, we shall suppress the mesh index i and angle index m to simplify the notations. First we integrate (2.18) against a test function φ_i on every τ_i to get

$$\int_{\tau_i} \hat{s}\cdot\nabla\Phi\varphi d\vec{x} + \int_{\tau_i} \mu_t\Phi\varphi d\vec{x} = \int_{\tau_i} (\mu_s \sum_{m'=1}^M w_{mm'}\Phi_{m'})\varphi d\vec{x} + \int_{\tau_i} q d\vec{x} \tag{2.19}$$

Then, applying Green's formula and splitting the flux through boundary, we have

$$-\int_{\tau_i} \Phi(\hat{s}\cdot\nabla\varphi)d\vec{x} + \int_{\Gamma^+} \Phi\varphi\hat{s}\cdot\hat{n}d\vec{S} + \int_{\tau_i}\mu_t\Phi\varphi d\vec{x} = \int_{\Gamma^-}\hat{\Phi}\varphi(-\hat{s}\cdot\hat{n})d\vec{S} + \int_{\tau_i}\mu_s\sum_{m'=1}^M w_{mm'}\Phi_{m'}\varphi d\vec{x} + \int_{\tau_i} q\varphi d\vec{x}, \tag{2.20}$$

where $\Gamma^+(\Gamma^-)$ denotes edges (in 2D) or faces (in 3D) of the element τ_i such that $\hat{s}\cdot\hat{n} \geq 0 (< 0)$ and $\hat{\Phi}$ is the upwind numerical flux. Finally, we have the following for piecewise constant DG

$$\phi\int_{\Gamma^+}\hat{s}\cdot\hat{n}d\vec{S} + \mu_t\phi\int_{\tau_i} d\vec{x} = \hat{\phi}\int_{\Gamma^-}(-\hat{s}\cdot\hat{n})d\vec{S} + \mu_s\sum_{m'=1}^M w_{mm'}\phi_{m'}\int_{\tau_i} d\vec{x} + \int_{\tau_i} q d\vec{x}, \tag{2.21}$$

where $\hat{\phi}$ is the value from the upwind neighboring element. So the corresponding explicit updating formula during an iteration is

$$\phi^n_{i,m} = \frac{\sum_{k=1}^{n_i}\hat{\phi}_{k,m}\int_{\Gamma^-}(-\hat{s}\cdot\hat{n})d\vec{S} + \mu_s(\sum_{m'=1}^{m-1} w_{mm'}\phi^n_{i,m'} + \sum_{m'=m+1}^M w_{mm'}\phi^{n-1}_{i,m'})A_i + \int_{\tau_i} q d\vec{x}}{\sum_{k=1}^{n_o}\int_{\Gamma^+}(\hat{s}\cdot\hat{n})d\vec{S} + (\mu_t - \mu_s w_{mm})A_i} \tag{2.22}$$

In 2D, A_i is the area of triangular element τ_i, n_i is the number of edges of τ_i with incoming flux, i.e., $\hat{s}\cdot\hat{n} < 0$, and n_o for that with $\hat{s}\cdot\hat{n} > 0$. On each edge Γ with two vertices (x_1, y_1) and (x_2, y_2), $\int_\Gamma(\hat{s}\cdot\hat{n})d\vec{S} = \cos\theta\cdot(y_2 - y_1) + \sin\theta\cdot(x_1 - x_2)$.

In 3D, A_i is the volume of tetrahedral element τ_i, n_i denotes the number of the faces of τ_i with incoming flux, i.e., $\hat{s}\cdot\hat{n} < 0$, and n_o for that with $\hat{s}\cdot\hat{n} > 0$. On each face τ with three counterclockwise oriented vertices $(x_i, y_i, z_i), i = 1, 2, 3$, $\int_\Gamma(\hat{s}\cdot\hat{n})d\vec{S} = (\vec{a}\times\vec{b})\cdot\hat{s}$, with $\vec{a} = (x_1 - x_2, y_1 - y_2, z_1 - z_2)$ and $\vec{b} = (x_1 - x_3, y_1 - y_3, z_1 - z_3)$.

Now we need to find the upwinding numerical flux $\hat{\phi}$. In order to determine where the upwind flux comes from, we need an assumption on spatial mesh, that is one edge or face is shared by at most two elements. For example, if Γ is shared by two triangles τ_1 and τ_2 in 2D, we know that the upwind flux comes from the other triangle τ_2 if it is incoming flux through Γ of τ_1 due to the assumption. In

implementation, the information for $\hat{\phi}$ is saved in the memory before running the RTE solver to save the computation time.

Another important issue is the ordering of the elements during the Gauss-Seidel iteration (2.22). A proper ordering of the elements, i.e., sweeping of all elements following the transport direction $\hat{s}$, will provide update in each cell using the most updated and correct information along characteristics. In this way information will propagate through all elements in very few iterations. For rectangular grid the ordering with respect to $\hat{s}$ is very simple and one sweep is enough for a fixed $\hat{s}$ (see Section 2.2.3). For unstructured mesh, the ordering is more complicated and varies with each $\hat{s}$. We will discuss it in more details in Section 2.4.3.

2.4.2. *Spatial discretization: piecewise linear DG.* Denote $\varphi_{ij}, i = 1, \ldots, N, j = 1, \ldots, n_d$ to be the basis for linear functions in element i, in which $n_d = 3$ for 2D triangular elements and $n_d = 4$ for 3D tetrahedral elements. In particular we use the nodal basis, i.e, $\varphi_{ij}(k) = \delta_{jk}$, where $j, k = 1, \ldots, n_d$ is the index for nodes in an element. Then for each angular direction $\hat{s}_m$, $\Phi_m \approx \sum_{i=1}^{N} \sum_{j=1}^{n_d} \phi_{ij,m} \varphi_{ij}$, and $\mu_t \approx \sum_{i=1}^{N} \sum_{j=1}^{n_d} \mu_{t_{ij}} \varphi_{ij}$ and $\mu_s \approx \sum_{i=1}^{N} \sum_{j=1}^{n_d} \mu_{s_{ij}} \varphi_{ij}$. In the following presentation, we shall suppress i, m to simplify notations whenever needed. Similarly, we integrate (2.18) against test function φ_{ij} on every element τ_i to get

$$\begin{array}{l} -\int_{\tau_i} \Phi(\hat{s} \cdot \nabla \varphi_j) d\vec{x} + \hat{s} \cdot \hat{n} \int_{\Gamma^+} \Phi \varphi_j d\vec{S} + \int_{\tau_i} (\mu_t - \mu_s w_{mm}) \Phi \varphi_j d\vec{x} \\ = (-\hat{s} \cdot \hat{n}) \int_{\Gamma^-} \hat{\Phi} \varphi_j d\vec{S} + \int_{\tau_i} (\mu_s \sum_{m' \neq m} w_{mm'} \Phi_{m'}) \varphi_j d\vec{x} + \int_{\tau_i} q \varphi_j d\vec{x}, \\ j = 1, 2, 3. \end{array} \tag{2.23}$$

The corresponding formula can be written in a linear system

$$(A + B^+ + C)\phi_m^n = B^- \hat{\phi} + D + Q \tag{2.24}$$

During the Guass-Seidel iterations a $n_d \times n_d$ linear system (2.24) is solved for each element.

What is left is to specify the matrices in (2.24), which are computed through a linear transformation (Figure 7), for example, $(x, y) = F(\hat{x}, \hat{y})$ on the triangular element in 2D as $\begin{bmatrix} x \\ y \end{bmatrix} = J \begin{bmatrix} \hat{x} \\ \hat{y} \end{bmatrix} + \begin{bmatrix} x_1 \\ y_1 \end{bmatrix}$ with $J = \begin{bmatrix} x_2 - x_1 & x_3 - x_1 \\ y_2 - y_1 & y_3 - y_1 \end{bmatrix}$. In the following, we list the results without deriving it.

In 2D,

$$\phi_m^n = \begin{bmatrix} \phi_{m,1}^n \\ \phi_{m,2}^n \\ \phi_{m,3}^n \end{bmatrix}, \; A = [-\int \varphi_i (\hat{s} \cdot \nabla \varphi_j) d\vec{x}] = \frac{1}{6} \begin{bmatrix} a+b & a+b & a+b \\ -a & -a & -a \\ -b & -b & -b \end{bmatrix}$$

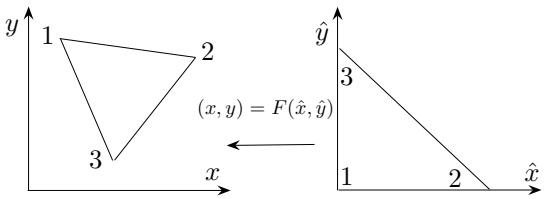

FIGURE 7. Linear transform in 2D

with $a = s_1(y_3 - y_1) + s_2(x_1 - x_3)$ and $b = s_1(y_1 - y_2) + s_2(x_2 - x_1)$,

$$C = [-\int(\mu_t - \mu_s w_{mm'})\varphi_i\varphi_j d\vec{x}] = \frac{|J|}{120}\begin{bmatrix} 2(3c_1+c_2+c_3) & 2c_1+2c_2+c_3 & 2c_1+c_2+2c_3 \\ 2c_1+2c_2+c_3 & 2(c_1+3c_2+c_3) & c_1+2c_2+2c_3 \\ 2c_1+c_2+2c_3 & (c_1+2c_2+2c_3) & 2(c_1+c_2+3c_3) \end{bmatrix}$$

with $c_i = \mu_{t_i} - \mu_{s_i} w_{mm}$ and $|J|$ representing the absolute value of the determinant of J, and

$$D = [\int(\mu_s \sum_{m'\neq m} w_{mm'}\phi_{m',i})\varphi_i\varphi_j d\vec{x}] = \frac{|J|}{120}\begin{bmatrix} 2d_1(3c_1+c_2+c_3)+d_2(2c_1+2c_2+c_3)+d_3(2c_1+c_2+2c_3) \\ d_1(c_1+2c_2+c_3)+2d_2(c_1+3c_2+c_3)+d_3(c_1+2c_2+2c_3) \\ d_1(2c_1+c_2+2c_3)+d_2(c_1+2c_2+2c_3)+2d_3(c_1+c_2+3c_3) \end{bmatrix}$$

with $c_i = \mu_{s_i}$ and $d_i = \sum_{m'=1}^{m-1} w_{mm'}\phi^n_{m',i} + \sum_{m'=m+1}^{M} w_{mm'}\phi^{n-1}_{m',i}$.

If the light source is a point source at (x_q, y_q) with unit intensity, $Q = \begin{bmatrix} 1-\hat{x}_q-\hat{y}_q \\ \hat{x}_q \\ \hat{y}_q \end{bmatrix}$ with a scaling factor, where $(\hat{x}_q, \hat{y}_q) = F^{-1}(x_q, y_q)$.

Now we specify B^+ and $B^-\hat{\phi}$. First let us define the following three matrices as the contribution from each edge,

$$B_1 = \frac{|a+b|}{6}\begin{bmatrix} 0&0&0 \\ 0&2&1 \\ 0&1&2 \end{bmatrix},\ B_2 = \frac{|a|}{6}\begin{bmatrix} 2&0&1 \\ 0&0&0 \\ 1&0&2 \end{bmatrix},\ B_3 = \frac{|b|}{6}\begin{bmatrix} 2&1&0 \\ 1&2&0 \\ 0&0&0 \end{bmatrix},$$

where B_1 is for edge $\overline{23}$, B_2 is for edge $\overline{13}$, and B_3 is for edge $\overline{12}$ Then, B^+ and B^- are determined by the sign of $\hat{s}\cdot\hat{n}$. For example, if we know in a triangle τ that $\hat{s}\cdot\hat{n}_{13} > 0$, $\hat{s}\cdot\hat{n}_{12} > 0$ and $\hat{s}\cdot\hat{n}_{23} < 0$, then $B^+ = B_2 + B_3$ and $B^-\hat{\phi} = B_1\begin{bmatrix} 0 \\ \hat{\phi}_2 \\ \hat{\phi}_3 \end{bmatrix}$, where $\hat{\phi}_2, \hat{\phi}_3$ comes from adjacent triangle sharing the edge $\overline{23}$.

In 3D, we have the following compact formulas.

$$[A]_{ij} = \frac{1}{24}\sum_{k=1}^{3}(2\delta_{i1}-1)max(\delta_{i1},\delta_{i,k+1})a_k \text{ with } \begin{bmatrix} a_1 & a_2 & a_3 \end{bmatrix}^T = |J|J^{-1}\hat{s}^T.$$

$$[C]_{ij} = \frac{|J|}{720}(1+\delta_{ij})(\delta_i c_i + \delta_j c_j + \sum_{k=1}^{4} c_k) \text{ with } c_i = \mu_{t_i} - \mu_{s_i} w_{mm}.$$

$$[D]_j = \sum_{k=1}^{4} C_{jk} d_k \text{ with } c_i = \mu_{s_i},\ d_i = \sum_{m'=1}^{m-1} w_{mm'}\phi^n_{m',i} + \sum_{m'=m+1}^{M} w_{mm'}\phi^{n-1}_{m',i}.$$

$$[B_{ijk}]_{mn} = \frac{1}{24}|(\vec{l_{ij}} \times \vec{l_{ik}}) \cdot \hat{s}| \max(\delta_{im},\delta_{jm},\delta_{km})(1+\delta_{mn})$$

with $\vec{l_{ik}} := (x_j - x_i, y_j - y_i, z_j - z_i)$, in which the index sets (i,j,k) have the values (1,2,3), (1,2,4), (1,3,4) and (2,3,4). Here B^+ and B^- are assembled from B_{ijk} according to the sign of $\hat{s}\cdot\hat{n}$.

2.4.3. *Spatial ordering for each directional sweeping.* A proper ordering of the elements is crucial for fast convergence of ISI. By following the transport direction information will propagate through all elements in very few iterations. For rectangular grid the ordering is very simple. A group of direction $\hat{s}$ shares the same ordering. Moreover one sweep is enough for a fixed $\hat{s}$ (see Section 2.2.3).

For unstructured mesh, the ordering is not as straightforward as for rectangular grid. First, the ordering varies with each $\hat{s}$. Second, even for a fixed direction and a fixed mesh there is no ordering that can follow the direction perfectly in general. We use the following simple method to determine the sweeping ordering for each direction. For example, in 2D, let us consider a fixed triangulation T with the transport direction $\hat{s}$ (Figure 8). First we rotate the coordinate to align the direction x with $\hat{s}$ axis to get x' and then order the spatial grids according to the x' coordinates in the rotated frame. In another word, the ordering is following the planar wavefront that is propagating with direction $\hat{s}$. In this way, when one element is being updated those elements that are in the upwinding direction are more likely to have been updated already and therefore the most updated information is used. 3D case can be done similarly. In implementation, the ordering for each direction is stored before iteratively solving RTE since they are unchanged once the angular and spatial meshes are specified.

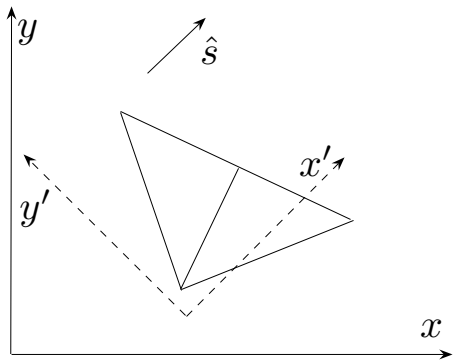

FIGURE 8. Alignment of the direction $\hat{s}$ with x axis

2.4.4. *Spatial interpolation and projection operators in multigrid.* Multigrid methods for unstructured grids are also more complicated than those for rectangular grids. In the following we define the spatial interpolation and projection operators for both piecewise constant DG and piecewise linear DG. Although we restrict our discussion in 2D for simplicity, the procedure can be carried over to 3D similarly.

First let us define a sequence of 2D spatial multigrid meshes $\Omega_{n_s} = (T_{n_s}, N_{\hat{s}}), n_{s0} \leq n_s \leq n_{sf}$, in which T_{n+1} is a refined spatial triangulation from T_n. Next we define the spatial mapping $P_{n\to n+1}$ from T_n to T_{n+1} by $P_{n\to n+1}(\tau_i) = \tau_j$ for $\tau_j \subseteq \tau_i$ with $\tau_i \in T_n$ and $\tau_j \in T_{n+1}$, which is stored in the memory before solving RTE. For piecewise constant DG, according to the above spatial mapping, we define the corresponding spatial interpolation operator I_n^{n+1} from T_n to T_{n+1} as $I_n^{n+1}(\Phi_{\tau_j}) = \Phi_{\tau_i}$ and spatial restriction operator I_{n+1}^n from T_{n+1} to T_n as $I_{n+1}^n(\Phi_{\tau_i}) = \sum_{j=1}^{N_i} \frac{A_{\tau_j}}{A_{\tau_i}} \Phi_{\tau_j}$, in which A represents area and N_i is the number of elements in T_{n+1} contained in the element τ_i in T_n.

Next, we consider the spatial interpolation and projection operators for 2D piecewise linear DG. Similarly, we first save the spatial mapping between the coarse mesh and the fine one. Then the spatial interpolation operator I_n^{n+1} from T_n to T_{n+1} is defined to be linear interpolation by

$$I_n^{n+1}(\phi_{i,j}) = \sum_{k=1}^{3} w_{ij,k}\phi_k, \tag{2.25}$$

where $\phi_{i,j}$ is the value at jth node of ith element on the fine mesh, ϕ_k is the value at kth node of the interpolated function on the coarse mesh and $w_{ij,k}$ is the value of the linear nodal basis φ_k at the vertex (i,j).

On the other hand, the spatial restriction operator I_{n+1}^n from T_{n+1} to T_n can be computed through standard L_2 projection technique by requiring

$$\int_{\tau_i} I_{n+1}^n(f)\varphi_k d\vec{x} = \sum_{\tau_j \subseteq \tau_i} \int_{\tau_j} f\varphi_k d\vec{x},\ k = 1,2,3 \tag{2.26}$$

in which f is a piecewise linear function on the fine mesh $\tau_j \subseteq \tau_i$ with $\tau_i \in T_n$ and $\tau_j \in T_{n+1}$, and φ_k is the nodal basis on the coarse mesh τ_i.

Then we have

$$\begin{bmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \end{bmatrix} = \sum_{j=1}^{N_i} \frac{A_{\tau_j}}{A_{\tau_i}} T^j \begin{bmatrix} \phi_{j,1} \\ \phi_{j,2} \\ \phi_{j,3} \end{bmatrix} \tag{2.27}$$

with

$$T^j = \begin{bmatrix} 1 - T_{21}^j - T_{31}^j & 1 - T_{22}^j - T_{32}^j & 1 - T_{23}^j - T_{33}^j \\ 2\hat{x}_1 + \hat{x}_2 + \hat{x}_3 - 1 & \hat{x}_1 + 2\hat{x}_2 + \hat{x}_3 - 1 & \hat{x}_1 + \hat{x}_2 + 2\hat{x}_3 - 1 \\ 2\hat{y}_1 + \hat{y}_2 + \hat{y}_3 - 1 & \hat{y}_1 + 2\hat{y}_2 + \hat{y}_3 - 1 & \hat{y}_1 + \hat{y}_2 + 2\hat{y}_3 - 1 \end{bmatrix} \tag{2.28}$$

where, ϕ_k is the value on kth node on a fixed element on the coarse mesh; $\phi_{j,k}$ is the value on kth node of jth element on the fine mesh contained in this fixed element which contains N_i elements of the fine mesh; $(\hat{x}_k, \hat{y}_k)$ are the new coordinates of kth node of jth element in the fixed element after transform (Figure 7) and A is for the area.

Similarly in 3D, we have

$$\begin{bmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \\ \phi_4 \end{bmatrix} = \sum_{j=1}^{N_i} \frac{V_{\tau_j}}{V_{\tau_i}} T^j \begin{bmatrix} \phi_{j,1} \\ \phi_{j,2} \\ \phi_{j,3} \\ \phi_{j,4} \end{bmatrix} \tag{2.29}$$

with

$$T^j = \begin{bmatrix} 1 - T_{21}^j - T_{31}^j - T_{41}^j & 1 - T_{22}^j - T_{32}^j - T_{42}^j & 1 - T_{23}^j - T_{33}^j - T_{43}^j & 1 - T_{24}^j - T_{34}^j - T_{44}^j \\ 2\hat{x}_1 + \hat{x}_2 + \hat{x}_3 + \hat{x}_4 - 1 & \hat{x}_1 + 2\hat{x}_2 + \hat{x}_3 + \hat{x}_4 - 1 & \hat{x}_1 + \hat{x}_2 + 2\hat{x}_3 + \hat{x}_4 - 1 & \hat{x}_1 + \hat{x}_2 + \hat{x}_3 + 2\hat{x}_4 - 1 \\ 2\hat{y}_1 + \hat{y}_2 + \hat{y}_3 + \hat{y}_4 - 1 & \hat{y}_1 + 2\hat{y}_2 + \hat{y}_3 + \hat{y}_4 - 1 & \hat{y}_1 + \hat{y}_2 + 2\hat{y}_3 + \hat{y}_4 - 1 & \hat{y}_1 + \hat{y}_2 + \hat{y}_3 + 2\hat{y}_4 - 1 \\ 2\hat{z}_1 + \hat{z}_2 + \hat{z}_3 + \hat{z}_4 - 1 & \hat{z}_1 + 2\hat{z}_2 + \hat{z}_3 + \hat{z}_4 - 1 & \hat{z}_1 + \hat{z}_2 + 2\hat{z}_3 + \hat{z}_4 - 1 & \hat{z}_1 + \hat{z}_2 + \hat{z}_3 + 2\hat{z}_4 - 1 \end{bmatrix}, \tag{2.30}$$

where V is for the volume.

With the spatial interpolation and projection operators defined above, we can implement various multigrid methods on unstructured grids to accelerate the RTE

solver. The numerical tests in Section 3 show that the multigrid methods accelerate ISI up to ten times.

2.5. **Boundary mismatch of refraction index.** Based on our angular discretization method, boundary condition can be treated easily. However, the implementation of boundary condition for finite difference scheme and DG method is a little different. For finite difference scheme, the boundary condition is enforced for point value while for DG boundary condition is enforced weakly by upwind flux through cell boundary. For example, the vacuum boundary condition for finite difference scheme on a rectangular grid in 2D is: at a grid point $(x_i, y_j) \in \partial\Omega$ on the boundary, we have

$$\Phi_{i,j,m} = 0 \tag{2.31}$$

for $\hat{s}_m \cdot \hat{n} \leq 0$, and

$$\Phi_{i,j,m} = \frac{a\Phi_{i-1,j,m} + b\Phi_{i,j-1,m} + \mu_s(\sum_{m'=1}^{m-1} w_{mm'}\Phi_{i,j,m'} + \sum_{m'=m+1}^{M} w_{mm'}\Phi_{i,j,m'}) + q_{i,j,m}}{a + b + \mu_t - \mu_s w_{mm}}, \tag{2.32}$$

for $\hat{s}_m \cdot \hat{n} > 0$, assuming that $\theta_m \in (0, \frac{\pi}{2})$. On the right hand side, fluxes $\Phi_{i-1,j,m}$, $\Phi_{i,j-1,m}$ come from interior neighbors and $\Phi_{i,j,m'} = 0$ for $\hat{s}_{m'} \cdot \hat{n} \leq 0$.

For piecewise constant DG discretization, at a boundary cell τ_k, we impose the influx through the cell boundary, which is also the domain boundary, $\hat{\phi}_{k,m} = 0$ in (2.22) for $\hat{s}_m \cdot \hat{n} \leq 0$. Similar flux condition is imposed for piecewise linear DG.

In the presence of boundary mismatch of the refraction index, one needs to consider reflection and refraction at the boundary (Figure 9). The relation between an outgoing flux ϕ^0 along direction $\hat{s}_0$, the reflected flux ϕ^1 along direction $\hat{s}_1$ and the refracted flux ϕ^2along direction $\hat{s}_2$, follows the Fresnel formula [6].

In the implementation (Figure 10), we reverse the above relation: given a direction at the boundary $\hat{s}_0$, we trace back from $\hat{s}_0$ and find direction $\hat{s}_1$, which is reflected into $\hat{s}_0$, and direction $\hat{s}_2$, which is refracted into $\hat{s}_0$, by Snell's law [6]. Then we can compute the flux contribution from ϕ^1 (photon flux) and ϕ^2 (light source) to ϕ^0 according to Fresnel formula as follow.

Let r be the reflection energy rate, then

$$r = \begin{cases} (\frac{n_i - n_o}{n_i + n_o})^2, & \text{if } \theta_i = 0 \\ 1, & \text{if } \frac{n_i}{n_o} sin\theta_i \geq 1 \\ \frac{1}{2}([\frac{tan(\theta_i - \theta_r)}{tan(\theta_i + \theta_r)}]^2 + [\frac{sin(\theta_i - \theta_r)}{sin(\theta_i + \theta_r)}]^2), & \text{otherwise} \end{cases} \tag{2.33}$$

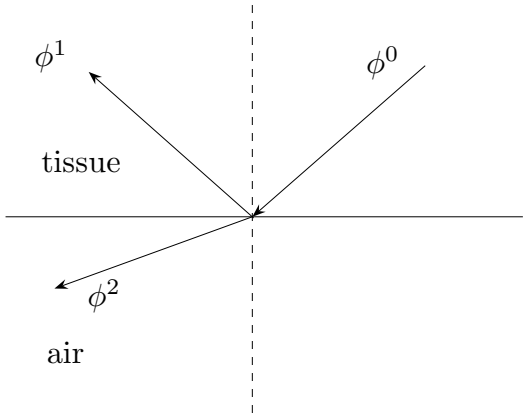

FIGURE 9. Reflection and refraction of an outgoing flux ϕ^0

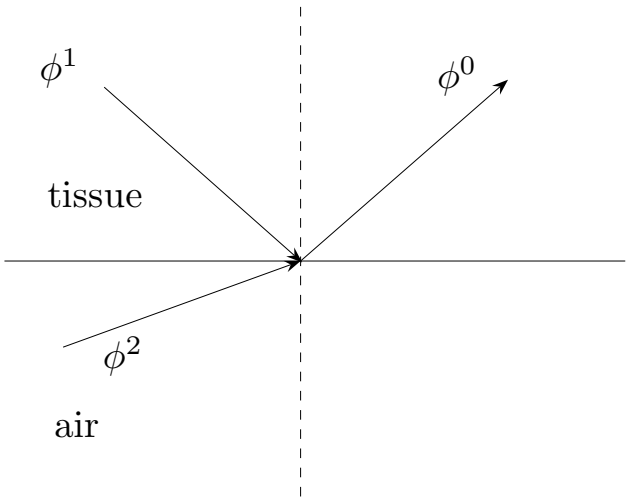

FIGURE 10. tracing back for the contribution from ϕ^1 and ϕ^2 to ϕ^0

where n_i for the refraction index of inside medium (tissue), n_o for that of outside medium (vacuum or air), θ_i for incident angle,and θ_r for refracted angle.

Then we have $\phi^0 = r_1\phi^1 + (1-r_2)\phi^2$ for the case with boundary mismatch of refraction index, in which r_1 represents the reflection energy rate for ϕ^1 and r_2 for ϕ^2. Note ϕ^2 is nonzero only in the presence of light source.

Since $\hat{s}_1$ and $\hat{s}_2$ may not coincide with the underlying discrete angular nodes, we use a linear interpolation of fluxes from neighboring angles to approximate ϕ^1 and ϕ^2 if necessary. In such a way, we establish a relationship among fluxes in different directions at the boundary. Again for finite difference scheme, at a grid point $(x_i, y_j) \in \partial\Omega$ on the boundary, for $\hat{s}_m \cdot \hat{n} \leq 0$,

$$\Phi_{i,j,m} = r_{m_1}\Phi_{i,j,m_1} + (1-r_{m2})\Phi_{i,j,m_2}, \tag{2.34}$$

where Φ_{i,j,m_1}, Φ_{i,j,m_2} are the corresponding traced-back reflected flux and refracted light source which may be linearly interpolated from neighboring fluxes on angular grids. This relation will replace (2.31) for vacuum boundary condition.

For piecewise constant DG discretization, at a boundary cell τ_k, the mismatch boundary condition is imposed through the influx at the cell boundary, which is also the domain boundary, $\hat{\phi}_{k,m} = r_{m_1}\phi_{k,m_1} + (1-r_{m_2})\phi_{k,m_2}$ in (2.22) for $\hat{s}_m \cdot \hat{n} \leq 0$, where ϕ_{k,m_1} is the corresponding traced-back reflected flux in the cell τ_k and ϕ_{k,m_2} is the corresponding traced-back refracted flux, e.g., a light source from exterior. Again these fluxes may be linearly interpolated from neighboring fluxes on angular grids. Similar flux condition is imposed for piecewise linear DG.

We see that the implementation difference between vacuum and non-vacuum boundary condition lies in the fact that the boundary fluxes for directions with $\hat{s} \cdot \hat{n} < 0$ are not zero and are coupled with outgoing fluxes through the boundary condition.

Remark In another word, the boundary condition provides a scattering relation among fluxes from different directions at boundary. More importantly, the above discretization at boundary will not destruct a M matrix system. It is also important that we use the trace back formulation and put $\hat{s}_0$ on the grid point of angular discretization. Otherwise if we put $\hat{s}_1$ or$\hat{s}_2$ on the grid point, it is not clear how to split the contribution from ϕ^1 and ϕ^2 to ϕ^0 since the reflection or refraction obeys Fresnel formula (2.33), which is nonlinear.

The coding includes two steps. First, for each $\hat{s}_0$ on the existing angular grids, we compute reflection direction $\hat{s}_1$ and refraction direction $\hat{s}_2$, and the corresponding

TABLE 2. The analytical solutions with $a = b = \mu_a/3$ for 2D and $a = b = c = \mu_a/6$ for 3D.

	Analytical solutions
Case 1	$\Phi = e^{-ax-by}$, f: H-G function with $g = 0.9$ $q = (\mu_a - a\cos\theta - b\sin\theta)e^{-ax-by}$
Case 2	$\Phi = e^{-ax-by}(1 + c_1\cos\theta)$, $f(\hat{s}, \hat{s}') = \frac{1}{2\pi}(1 + \frac{\hat{s}\cdot\hat{s}'}{2})$ $q = [(\mu_a - a\cos\theta - b\sin\theta)(1 + c_1\cos\theta) + \mu_s c_1 \frac{3\cos\theta}{4}]e^{-ax-by}$,$c_1 = \frac{\mu_a}{\mu_a + 6\mu_s}$
Case 3	$\Phi = e^{-ax-by-cz}$, f: H-G function with $g = 0.0$ $q = (\mu_a - a\sin\theta\cos\varphi - b\sin\theta\sin\varphi - c\cos\theta)e^{-ax-by-cz}$
Case 4	$\Phi = e^{-ax-by-cz}(1 + c_1\cos\theta)$, $f(\hat{s}, \hat{s}') = \frac{3}{16\pi}(1 + \hat{s}\cdot\hat{s}')^2$ $q = [(\mu_a - a\sin\theta\cos\varphi - b\sin\theta\sin\varphi - c\cos\theta)(1 + c_1\cos\theta) + \mu_s c_1 \frac{\cos\theta}{2}]e^{-ax-by-cz}$,$c_1 = \frac{\mu_a}{6\mu_s}$

reflection ratio, then we interpolate them linearly onto the existing angular grids. For each reflected or refracted direction, at most two directions on the existing grids contribute to it in 2D, while three directions do in 3D, based on our angular discretization. The weights and the directions for each ϕ^0 are stored in the memory before running the relaxation. Second, during the relaxation iteration, we trace back, that is, for each direction at boundary, we sum up the weighted contributions from the reflected fluxes and the refracted fluxes (e.g., boundary sources).

3. SIMULATIONS

In this section we present extensive numerical tests for our algorithm. Accuracy is tested on our piecewise linear DG method using analytic solution. The effects of several improvements, such as improved source iteration (ISI) and various multigrid methods are demonstrated. Tests in various scattering regimes, for heterogeneous media and in frequency domains show efficiency of our algorithms. Comparison with Monte Carlo Simulation is also provided.

3.1. **Verification of accuracy.** The analytical solutions in Table 2, on a homogeneous square or cube of side length $20mm$ with $\mu_a = 0.01mm^{-1}$ and $\mu_s = 1mm^{-1}$, are used to verify the accuracy of RTE solver.

The following quantities are used to measure the difference between the analytical solution Φ and numerical solution $\tilde{\Phi}$: $L_\infty^{(1)} = max_{\vec{x},\hat{s}}|\frac{\Phi-\tilde{\Phi}}{\Phi}|$, $L_2^{(1)} = \sqrt{\int(\Phi - \tilde{\Phi})^2 d\hat{s}d\vec{x}}$, $L_\infty^{(2)} = max_{\vec{x}}|\frac{\Psi-\tilde{\Psi}}{\Psi}|$, $L_2^{(2)} = \sqrt{\int(\Psi - \tilde{\Psi})^2 d\vec{x}}$, with photon density $\Psi = \int \Phi d\hat{s}$.
The numerical simulations in Table 3 and 4 show that the numerical solution to RTE with piecewise linear DG has second order accuracy as expected.

TABLE 3. Verification of 2^{nd} order accuracy of 2D RTE with piecewise linear DG on the mesh with five refinements, including: 13 nodes and 16 triangles, 41 nodes and 64 triangles, 145 nodes and 256 triangles, 545 nodes and 1024 triangles and 2113 nodes and 4096 triangles respectively.

	(n_a, n_s)	$L_\infty^{(1)}$	$L_\infty^{(2)}$	$L_2^{(1)}$	$L_2^{(2)}$
case 1	(2,1)	3.5e-3	2.1e-3	3.1e-2	7.1e-2
	(3,2)	1.0e-3	5.3e-4	1.0e-2	1.9e-2
	(4,3)	3.0e-4	1.3e-4	3.0e-3	4.8e-3
	(5,4)	1.0e-4	2.8e-5	8.6e-4	1.3e-3
	(6,5)	4.0e-5	7.3e-6	2.6e-4	3.4e-4
case 2	(2,1)	3.7e-3	2.8e-3	3.9e-2	9.5e-2
	(3,2)	9.7e-4	6.9e-4	1.0e-2	2.3e-2
	(4,3)	2.4e-4	1.7e-4	2.6e-3	5.7e-3
	(5,4)	8.3e-5	4.0e-5	7.3e-4	1.5e-3
	(6,5)	3.4e-5	1.0e-5	2.2e-4	3.9e-4

TABLE 4. Verification of 2^{nd} order accuracy of 3D RTE with piecewise linear DG on the mesh with three refinements, including: 172 nodes and 612 triangles, 1068 nodes and 4896 triangles, and 7483 nodes and 39168 triangles respectively.

	(n_a, n_s)	$L_\infty^{(1)}$	$L_\infty^{(2)}$	$L_2^{(1)}$	$L_2^{(2)}$
Case 3	(2,1)	1.3e-3	9.9e-4	5.4e-2	1.8e-1
	(3,2)	3.9e-4	2.1e-4	1.4e-2	4.2e-2
	(4,3)	1.1e-4	5.4e-3	3.8e-3	1.1e-2
Case 4	(2,1)	1.2e-3	8.7e-4	6.9e-2	1.9e-1
	(3,2)	3.9e-4	1.9e-4	1.8e-2	4.5e-2
	(4,3)	1.3e-4	4.8e-5	4.9e-3	1.2e-2

3.2. **Comparison of ISI and SI on structured grids.** In this study, each side of the square or cubic domain has $(N-1)$ mean free paths $(1/\mu_t)$. In Table 5 and 6, we vary parameters in the first four columns while fixing others. Upwind finite difference is used to discretize the spatial variables in this study. For all tests in 3D shown in Table 6, we use the residual error from 100 standard SI as the stopping criterion. From Table 5 and 6, we conclude that the improvement of Method 1, which utilizes updated information from other directions, degrades as the value of g increases since the coupling among fluxes in different direction becomes less. However, Method 2 performs better as the value of g increases since the dominant scattering term is properly involved. Without extra computational

TABLE 5. the number of iterations for 2D problems: the spatial mesh is $N \times N$ and the angular mesh has M angles.

g	$\mu_s - \mu_a$	N	M	SI	Method1	Method 2
0.7	1-0.01	101	36	484	278	207
0.9	1-0.01	101	60	389	246	144
0.95	1-0.01	101	72	314	218	98
0.7	10-0.01	101	36	1267	727	540
0.9	10-0.01	101	60	1437	908	529
0.95	10-0.01	101	72	1460	1013	447

TABLE 6. the number of iterations for 3D problems: the spatial mesh is $N \times N \times N$ and the angular mesh has n_a refinements as in Table 1.

g	$\mu_s - \mu_a$	N	n_a	SI	Method1	Method 2
0.7	1-0.01	101	4	100	58	42
0.9	1-0.01	101	4	100	76	24
0.95	1-0.01	101	4	100	80	21
0.0	5-0.01	101	4	100	50	49
0.7	5-0.01	101	4	100	58	41
0.9	5-0.01	101	4	100	76	23

cost, we can reduce the number of iterations by ISI to one third (1/3) of that by SI in the regime with highly forward-peaking scattering.

3.3. **Comparison of multigrid methods and ISI on structured grids.** The 2D simulations in this study are on a homogeneous square domain with side length L and corners $(0,0)$, $(0,L)$, $(L,0)$ and (L,L) with a isotropic point source at $(0.1L, 0.1L)$ with unit strength. $n_{af} = 6$, $n_{sf} = 7$, $n_{a0} = 2$, $n_{s0} = 1$ and each side L has length of 64 mean free paths.

3D numerical tests have the similar settings on a cubic domain with side length of 32 mean free paths and a isotropic source at $(0.1L, 0.1L, 0.1L)$ with $n_{af} = 4$, $n_{sf} = 5$. $n_{a0} = 1$, $n_{s0} = 1$ in Table 10-12.

In Table 7-12, the computational time t, the residual $\|d\|_1 := \int |d(\hat{s}, \vec{r})| d\hat{s} d\vec{r}$ as the stopping criterion, the iteration number N and the average convergence factor ρ according to $\rho^n = \|d^n\|_1 / \|d^{n-1}\|_1$ are compared; μ_a, μ_s has the unit mm^{-1}.

The conclusion is that multigrid methods can accelerate ISI significantly in all scattering regimes. Our algorithms are very effective for high forward peaking cases. The number of multigrid iterations increases (slowly) when scattering becomes stronger.

TABLE 7. The cases with $g = 0.9$

(μ_s, μ_a)		MG1	MG2	MG3	MG4	AMG	SMG	ISI
(1,0.01)	t	54	85	68	55	68	79	334
	$\|d\|$	1.1e-7	5.3e-7	1.0e-6	4.3e-8	9.1e-7	2.2e-6	6.1e-7
	N	6	7	6	5	6	7	280
	ρ	0.031	0.060	0.039	0.011	0.039	0.072	0.934
(10,0.01)	t	63	108	115	68	126	137	577
	$\|d\|$	1.5e-6	6.6e-6	1.5e-6	4.3e-7	8.3e-6	1.0e-5	3.5e-6
	N	7	9	10	6	11	12	480
	ρ	0.036	0.088	0.096	0.016	0.14	0.17	0.955
(1,0)	t	53	96	91	55	103	114	483
	$\|d\|$	5.0e-7	1.1e-6	3.2e-6	4.2e-7	1.0e-5	8.5e-6	3.7e-6
	N	6	8	8	5	9	10	400
	ρ	0.040	0.092	0.10	0.018	0.15	0.18	0.958

TABLE 8. The cases with $g = 0.7$

(μ_s, μ_a)		MG1	MG2	MG3	MG4	AMG	SMG	ISI
(1,0.01)	t	44	85	97	66	81	125	502
	$\|d\|$	9.0e-7	1.4e-5	2.0e-5	8.4e-7	7.0e-6	2.4e-5	1.0e-5
	N	6	9	9	6	7	11	420
	ρ	0.038	0.16	0.16	0.037	0.084	0.24	0.963
(10,0.01)	t	132	169	262	132	235	331	1298
	$\|d\|$	1.1e-5	1.7e-5	2.4e-5	1.9e-5	1.5e-5	3.3e-5	1.9e-5
	N	15	14	23	12	20	29	1080
	ρ	0.25	0.22	0.41	0.18	0.35	0.50	0.982
(1,0)	t	115	132	216	121	197	273	1102
	$\|d\|$	1.4e-5	2.5e-5	3.8e-5	7.7e-6	2.4e-5	5.4e-5	2.5e-5
	N	13	11	19	11	17	24	920
	ρ	0.29	0.23	0.45	0.22	0.40	0.54	0.984

TABLE 9. The cases with $g = 0.0$

(μ_s, μ_a)		MG1	MG2	MG3	MG4	AMG	SMG	ISI
(1,0.01)	t	92	73	137	95	151	145	623
	$\|d\|$	9.9e-6	5.0e-7	2.5e-5	5.8e-6	2.0e-5	2.8e-5	1.3e-5
	N	12	7	14	10	15	15	580
	ρ	0.24	0.057	0.31	0.17	0.34	0.34	0.975
(10,0.01)	t	367	261	506	331	612	611	2422
	$\|d\|$	7.2e-5	4.7e-5	7.1e-5	4.4e-5	7.3e-5	7.0e-5	6.1e-5
	N	48	25	52	35	61	63	2260
	ρ	0.67	0.46	0.69	0.57	0.73	0.74	0.992
(1,0)	t	368	261	498	321	614	600	2470
	$\|d\|$	6.6e-5	5.8e-5	7.6e-5	6.5e-5	6.9e-5	7.7e-5	6.6e-5
	N	48	25	51	34	61	62	2240
	ρ	0.74	0.56	0.75	0.65	0.79	0.79	0.994

TABLE 10. The 3D cases with $g = 0.9$

(μ_s, μ_a)		MG1	MG2	MG3	MG4	AMG	SMG	ISI
(1,0.01)	t	266	289	289	288	281	292	930
	$\|d\|$	2.2e-4	5.8e-5	4.6e-5	3.5e-5	1.6e-4	1.2e-4	6.4e-5
	N	4	4	4	4	4	4	90
	ρ	0.022	0.012	0.012	0.012	0.017	0.025	0.850
(1,0)	t	333	289	288	288	357	292	1029
	$\|d\|$	1.6e-5	3.6e-4	2.0e-4	1.0e-4	1.7e-5	8.2e-4	1.5e-4
	N	5	4	4	4	5	4	100
	ρ	0.025	0.020	0.018	0.016	0.020	0.044	0.872

TABLE 11. The 3D cases with $g = 0.7$

(μ_s, μ_a)		MG1	MG2	MG3	MG4	AMG	SMG	ISI
(1,0.01)	t	272	360	290	288	352	453	1641
	$\|d\|$	4.3e-4	9.2e-5	5.0e-4	1.4e-4	4.2e-4	1.4e-3	2.8e-4
	N	4	5	4	4	5	6	160
	ρ	0.029	0.055	0.029	0.028	0.073	0.18	0.926
(1,0)	t	272	362	290	288	423	656	2042
	$\|d\|$	7.2e-4	2.3e-4	9.7e-4	2.9e-4	9.6e-5	1.3e-3	6.9e-4
	N	4	5	4	4	6	9	200
	ρ	0.036	0.065	0.037	0.034	0.085	0.31	0.944

TABLE 12. The 3D cases with $g = 0.0$

(μ_s, μ_a)		MG1	MG2	MG3	MG4	AMG	SMG	ISI
(1,0.01)	t	475	289	453	369	564	666	2257
	$\|d\|$	3.7e-3	3.8e-4	3.3e-3	1.3e-3	1.1e-3	2.0e-3	9.4e-4
	N	7	4	7	5	8	9	220
	ρ	0.29	0.03	0.27	0.14	0.27	0.36	0.953
(1,0)	t	813	445	795	590	846	1035	3390
	$\|d\|$	4.5e-3	2.1e-3	4.1e-3	2.2e-3	3.5e-3	4.3e-3	2.9e-3
	N	12	6	11	8	12	14	330
	ρ	0.49	0.21	0.45	0.31	0.46	0.54	0.972

3.4. **Performance of multigrid acceleration in heterogeneous media on structured grids.** The simulations are on a square domain with side length $L = 20mm$ and corners $(0, 0)$, $(0, L)$, $(L, 0)$ and (L, L) with a square inclusion centered at $(0.5L, 0.5L)$ with side length l. MG2 is used with the following parameters: $n_{a0} = 3$, $n_{s0} = 3$, $n_{af} = 6$, $n_{sf} = 7$, $g = 0.7$ and $\mu_a = 0.01mm^{-1}$ with a isotropic point source at $(0.1L, 0.1L)$ with unit strength. μ_s^1 and μ_s^2 denote scattering coefficients for the background and the inclusion separately, which have the unit mm^{-1}. In

TABLE 13. convergence factor ρ for homogenous media

μ_s^1	1	2	5	10
ρ	0.032	0.052	0.37	0.69

TABLE 14. convergence factor ρ for the inclusion with $l = 5mm$

(μ_s^1, μ_s^2)	(1,2)	(1,5)	(1,10)	(2,5)	(2,10)	(5,10)
ρ	0.040	0.044	0.097	0.055	0.18	0.45

TABLE 15. convergence factor ρ for the inclusion with $l = 10mm$

(μ_s^1, μ_s^2)	(1,2)	(1,5)	(1,10)	(2,5)	(2,10)	(5,10)
ρ	0.052	0.088	0.46	0.15	0.49	0.58

TABLE 16. convergence factor ρ for the inclusion with random scattering coefficients with $l = 10mm$

$(\mu_s^1, \delta\mu_s)$	(1,1)	(1,4)	(1,9)	(2,3)	(2,8)	(5,5)
ρ	0.053	0.080	0.42	0.16	0.47	0.58

Table 16, the random scattering coefficients $\delta\mu_s$ are added to the background μ_s^1 in the inclusion, which is uniformly distributed between 0 and $\Delta\mu_s$.

The conclusion from Table 13-16 is that the acceleration of ISI by multigrid methods does not degrade in heterogeneous media or the convergence factor ρ for heterogeneous media is in between of those for the corresponding homogeneous media.

3.5. **Comparison of multigrid methods and ISI on unstructured grids.** The 2D simulations with piecewise linear DG are performed on a circular domain centered at $(0, 0)$ with radius $20mm$ and an isotropic source at $(-8mm, 0)$. Both of the angular mesh and the spatial mesh (Table 17) has six refinements with $n_{a0} = 2$, $n_{s0} = 1$, $n_{af} = 6$ and $n_{sf} = 6$ in multigrid acceleration.

The 3D simulations with piecewise linear DG are performed on a cylinder (Figure 11) centered at $(0, 0, 40mm)$ with radius $10mm$, height $80mm$ and an isotropic source at $(-5mm, 0, 40mm)$. The angular mesh has three refinements with $n_{a0} = 1$ and $n_{af} = 3$, and the spatial mesh (Table 20) has four refinements $n_{s0} = 1$ and $n_{sf} = 4$ in multigrid acceleration. Table 21 and 22 show the comparison results, in which the consideration of boundary mismatch of refraction index is in the second simulation as a typical setting in optical imaging.

TABLE 17. 2D spatial mesh with n_p vertices and n_t elements

n_s	1	2	3	4	5	6
n_p	9	25	81	289	1089	4225
n_t	8	32	128	512	2048	8192

TABLE 18. The 2D cases with $g = 0.0$ for piecewise linear DG

(μ_s, μ_a)		MG1	MG2	MG3	MG4	AMG	SMG	ISI
(1,0.01)	t	165	123	176	139	321	245	1340
	$\|d\|$	8.7e-4	1.2e-4	7.8e-5	6.3e-5	1.6e-3	1.2e-3	7.6e-4
	N	9	5	7	6	13	12	460
	ρ	0.20	0.037	0.11	0.060	0.35	0.31	0.971
(1,0)	t	385	295	310	303	742	553	2934
	$\|d\|$	3.8e-3	2.9e-3	3.5e-3	2.6e-3	4.5e-3	4.4e-3	3.5e-3
	N	21	12	13	13	30	27	1010
	ρ	0.54	0.34	0.38	0.36	0.65	0.63	0.988

TABLE 19. The 2D cases with $g = 0.9$ for piecewise linear DG

(μ_s, μ_a)		MG1	MG2	MG3	MG4	AMG	SMG	ISI
(1,0.01)	t	91	98	95	93	123	82	438
	$\|d\|$	8.5e-6	3.6e-4	2.1e-4	1.0e-4	4.8e-6	7.8e-5	5.0e-5
	N	5	4	4	4	5	4	150
	ρ	0.021	0.020	0.018	0.016	0.020	0.015	0.897
(10,0.01)	t	639	494	492	489	1184	1563	7710
	$\|d\|$	5.2e-3	4.5e-3	8.0e-3	3.8e-3	7.2e-3	8.5e-3	5.7e-3
	N	35	20	20	21	48	76	2640
	ρ	0.70	0.50	0.52	0.54	0.78	0.85	0.996

TABLE 20. 3D spatial mesh for piecewise linear DG with n_p vertices and n_t elements

n_s	1	2	3	4
n_p	353	1008	2944	8685
n_t	1146	4355	13600	43086

The conclusion from Table 18-22 is similar to those on structured grids.

3.6. **Comparison of multigrid methods and ISI for frequency-domain RTE on 2D unstructured grids.** The following simulations for frequency-domain RTE [10] are performed on a circular domain centered at $(0, 0)$ with radius $20mm$ and an isotropic source at $(-8mm, 0)$. Both of the angular mesh and the spatial mesh (Table 23) has five refinements with $n_{a0} = 2$, $n_{s0} = 1$, $n_{af} = 5$ and $n_{sf} = 5$ in

TABLE 21. The 3D cases with $g = 0.0$ for piecewise linear DG:$(1, 0.01)^*$ is with reflection boundary condition for $n_i = 1.5$ and $n_o = 1.0$.

(μ_s, μ_a)		MG1	MG2	MG3	MG4	AMG	SMG	ISI
(1,0.01)	t	1168	862	1201	861	1676	1697	4613
	$\|d\|$	1.7e-4	7.0e-5	1.2e-4	1.3e-4	3.5e-4	2.7e-4	1.3e-4
	N	9	6	7	6	12	10	260
	ρ	0.28	0.13	0.18	0.14	0.41	0.33	0.961
$(1, 0.01)^*$	t	1259	1102	1340	1116	1950	1867	5424
	$\|d\|$	3.8e-4	7.7e-5	9.3e-5	4.9e-5	3.6e-4	3.7e-4	1.9e-4
	N	10	8	8	8	14	11	290
	ρ	0.34	0.21	0.22	0.20	0.46	0.38	0.965

TABLE 22. The 3D cases with $g = 0.9$ for piecewise linear DG:$(10, 0.01)^*$ is with reflection boundary condition for $n_i = 1.5$ and $n_o = 1.0$.

(μ_s, μ_a)		MG1	MG2	MG3	MG4	AMG	SMG	ISI
(10,0.01)	t	4115	8753	4305	2987	6085	6811	17105
	$\|d\|$	5.8e-4	9.6e-4	4.8e-4	6.3e-4	7.2e-4	6.5e-4	5.9e-4
	N	30	59	24	20	41	38	910
	ρ	0.70	0.84	0.64	0.59	0.78	0.76	0.990
$(10, 0.01)^*$	t	4950	8638	5051	3604	7415	7891	20318
	$\|d\|$	6.9e-4	9.3e-4	5.5e-4	4.8e-4	7.3e-4	8.1e-4	6.7e-4
	N	36	58	28	24	50	44	1080
	ρ	0.75	0.84	0.68	0.64	0.81	0.79	0.992

TABLE 23. 2D spatial mesh with n_p vertices and n_t elements

n_s	1	2	3	4	5
n_p	9	25	81	289	1089
n_t	8	32	128	512	2048

multigrid acceleration. The conclusion from Table 24 and 25 is that the multigrid acceleration of ISI is not affected by modulation frequency within the normal range.

3.7. **Comparison of RTE solver and Monte Carlo simulation in 2D.** The following simulations are performed on a square domain centered at $(10mm, 10mm)$ with side length $20mm$ and an isotropic source at $(6mm, 3mm)$ with homogeneous background $\mu_a = 0.01mm^{-1}$, $\mu_s = 1mm^{-1}$, and $n_o = 1.0$. In our RTE solver, MG4 with piecewise linear DG is used with $n_{a0} = 2$, $n_{af} = 5$, $n_{s0} = 1$ and $n_{sf} = 3$ (Table 26) on a desktop with Intel CPU E6850 (3.00GHz). Monte Carlo simulation (MC) is performed by Tracepro [5] with five million photons on Dell X64 workstation

TABLE 24. The 2D cases with $g = 0.0, \mu_a = 0.01mm^{-1}, \mu_s = 1mm^{-1}$, with piecewise linear DG

ω		MG1	MG2	MG3	MG4	AMG	SMG	ISI
0	t	16	11	18	12	24	22	60
	$\|d\|$	1.7e-5	2.1e-6	9.0e-6	6.6e-6	4.2e-5	2.4e-5	1.5e-5
	N	11	6	10	7	13	14	420
	ρ	0.31	0.08	0.26	0.14	0.40	0.41	0.969
1e7	t	16	11	18	13	24	22	60
	$\|d\|$	1.7e-5	2.2e-6	9.0e-6	6.6e-6	4.2e-5	2.4e-5	1.5e-5
	N	11	6	10	7	13	14	420
	ρ	0.31	0.08	0.26	0.14	0.40	0.41	0.969
1e8	t	16	11	18	13	24	22	60
	$\|d\|$	1.7e-5	2.8e-6	9.8e-6	6.8e-6	4.2e-5	2.4e-5	1.5e-5
	N	11	6	10	7	13	14	420
	ρ	0.31	0.09	0.26	0.14	0.40	0.41	0.953
1e9	t	16	17	17	15	26	21	59
	$\|d\|$	2.7e-5	9.3e-6	1.7e-5	6.0e-6	2.1e-5	3.7e-5	1.6e-5
	N	11	9	9	7	14	13	410
	ρ	0.32	0.22	0.24	0.18	0.40	0.39	0.935
1e10	t	7	9	9	9	13	8	28
	$\|d\|$	5.4e-7	1.1e-7	1.1e-7	5.8e-7	4.4e-6	8.4e-7	5.8e-7
	N	5	5	5	5	7	5	200
	ρ	0.04	0.03	0.03	0.04	0.13	0.04	0.765

TABLE 25. The 2D cases with $g = 0.9, \mu_a = 0.01mm^{-1}, \mu_s = 10mm^{-1}$, with piecewise linear DG

ω		MG1	MG2	MG3	MG4	AMG	SMG	ISI
0	t	55	36	68	42	90	111	292
	$\|d\|$	6.1e-5	4.7e-5	6.9e-5	4.1e-5	7.6e-5	7.5e-5	5.5e-5
	N	39	20	38	24	50	72	2120
	ρ	0.74	0.55	0.73	0.60	0.79	0.85	0.994
1e7	t	57	38	70	44	93	113	304
	$\|d\|$	6.1e-5	4.7e-5	6.8e5	4.1e-5	7.6e-5	7.5e-5	5.5e-5
	N	39	20	38	24	50	72	2120
	ρ	0.74	0.55	0.73	0.60	0.79	0.85	0.994
1e8	t	56	38	70	44	93	113	303
	$\|d\|$	6.2e-5	4.8e-5	6.9e-5	4.2e-5	7.6e-5	7.5e-5	5.5e-5
	N	39	20	38	24	50	72	2120
	ρ	0.74	0.55	0.73	0.60	0.79	0.85	0.993
1e9	t	58	43	72	45	95	113	299
	$\|d\|$	7.1e-5	5.8e-5	6.9e-5	5.6e-5	7.3e-5	8.4e-5	5.5e-5
	N	40	23	38	25	51	71	2090
	ρ	0.75	0.60	0.73	0.62	0.80	0.85	0.999

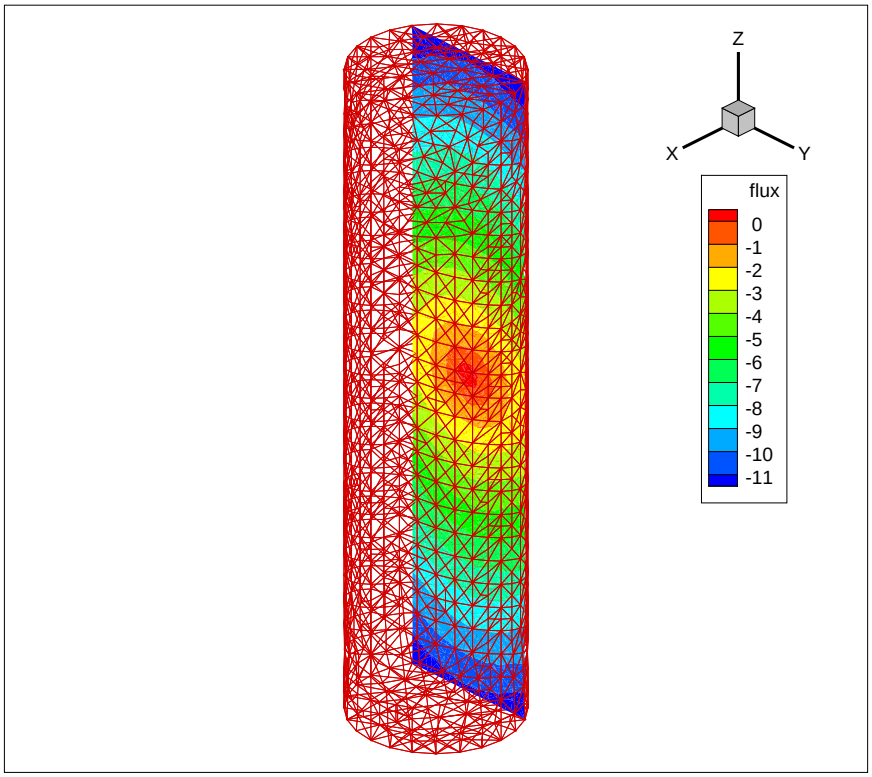

FIGURE 11. Plot of natural logarithm of 3D flux with $x = -5mm$ for Case $(10, 0.01)^*$ in Table 22

TABLE 26. 2D spatial mesh with n_p vertices and n_t elements

n_s	1	2	3
n_p	27	89	321
n_t	36	144	576

with a 2.80GHz CPU. In Figure 12 and 13, The boundary measure Φ from RTE and MC are plotted with $\Phi = \int_{\hat{s}\cdot\hat{n}>0} \hat{s}\cdot\hat{n}\phi(\vec{r},\hat{s})d\hat{s}$.

From Table 27, we conclude that, with either vacuum or reflection boundary condition, our RTE solver is much faster than MC, ranging from hundreds to thousands time, besides providing much smoother solution.

4. SUMMARY

In this paper, we present a efficient numerical algorithm for steady-state or frequency-domain radiative transport equation. The discretization is based on a direct triangulation in angular space and discontinuous Galerkin method in space.

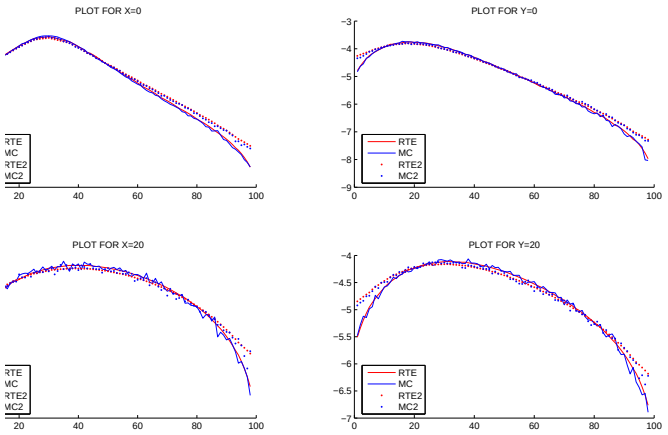

FIGURE 12. Plot of natural logarithm of normalized Φ for $g = 0.0$: RTE denotes the curve from our RTE solver with n=1,RTE2 with n=1.5; MC denotes the curve from MC with n=1, MC2 with n=1.5.

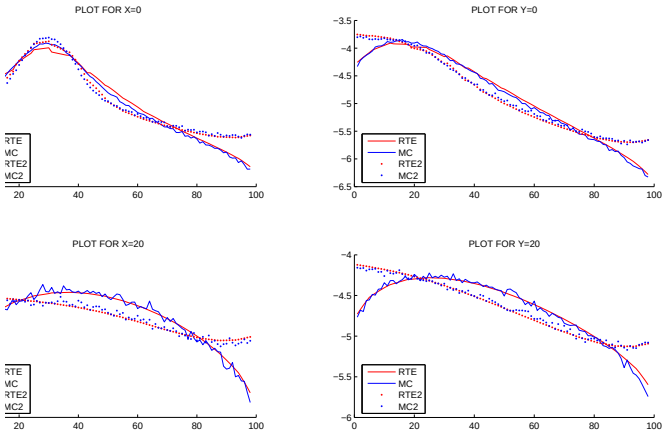

FIGURE 13. Plot of natural logarithm of normalized Φ for $g = 0.9$: RTE denotes the curve from our RTE solver with n=1,RTE2 with n=1.5; MC denotes the curve from MC with n=1, MC2 with n=1.5.

TABLE 27. Comparison of computation time (in seconds) of RTE and MC: g: anisotropic factor in H-G; n_i: refraction index of the square.

g	n_i	MC	RTE
0.0	1.0	700	1
0.0	1.5	1976	1
0.9	1.0	265	0.6
0.9	1.5	1484	0.8

Improved source iteration combined with multigrid method in both angle and space can dramatically improve the speed of convergence over standard source iteration method consistently in all scattering regimes. Our method works for general mesh in 2D and 3D and can treat various boundary conditions. Extensive numerical tests and comparison with Monte Carlo simulation demonstrate both accuracy and efficiency of our method.

Acknowledgement This work is partially supported by NSF grant DMS-0513073, DMS-0811254, and ONR grant N00014-06-1-0437. We would like to thank Dr. Ming Jiang and Yuanzheng Si in School of Mathematical Sciences at Peking University for providing the simulation results for Monte Carlo method.

References

1. Marshak A and Davis AB, *3d radiative transfer in cloudy atmospheres*, Springer-Verlag, Berlin (2005).
2. Klose AD, Netz U, Beuthan J, and Hielscher AH, *Optical tomography using the time-independent equation of radiative transfer - part 1: forward model*, Journal of Quantitative Spectroscopy & Radiative Transfer **72** (2002), 691–713.
3. Cockburn B, Karniadakis GE, and Shu CW, *Discontinuous galerkin methods: theory, computation and applications*, Springer, New York (2000).
4. Lewis EE and Miller WF, *Computational methods of neutron transport*, ANS Inc., La Grange Park, Illinois (1993).
5. Freniere ER, Gregory GG, and Hassler RA, *Edge diffraction in monte carlo ray tracing*, Optical Design and Analysis Software **3780** (1999), 151–157.
6. Jackson JD, *Classical electrodynamics, 3rd edition*, Wiley, New York (1999).
7. Morel JE and Warsa JS, *An sn spatial discretization scheme for tetrahedral meshes*, Nuclear science and engineering **151** (2005), 157–166.
8. Morel JE and Manteuffel TA, *An angular multigrid acceleration technique for s_n equations with highly forward-peaked scattering*, Nuclear Science and Engineering **107** (1991), 330–342.
9. Amit Joshi, John C Rasmussen, Eva M Sevick-Muraca, Todd A Wareing, and John McGhee, *Radiative transport-based frequency-domain fluorescence tomography*, Phys. Med. Biol. **53** (2008), 2069–2088.
10. Ren K, Abdoulaev GS, Bal G, and Hielscher AH, *Algorithm for solving the equation of radiative transfer in the frequency domain*, Optical letters **29** (2004), 578–580.
11. Lathrop KD, *Spatial differencing of the transport equation: positivity vs. accuracy*, Journal of computational physics **4** (1969), 475–498.
12. Evans KF, *The spherical harmonics discrete ordinate method for three-dimensional atmospheric radiative transfer*, J. Atmos. Sci. **55** (1998), 429–446.
13. Case KM and Zweifel PF, *Linear transport theory*, Addison-Wesley, Massachusetts (1967).
14. Adams ML and Larsen EW, *Fast iterative methods for discrete-ordinates particle transport calculations*, Progress in nuclear energy **40** (2002), 3–159.
15. Guven O and Bayazitoglu Y, *Radiative heat transfer solutions for anisotropically scattering media: the use of wavelets*, Journal of thermophysics and heat transfer **18** (2004), 172–177.
16. Koch R and Becker R, *Evaluation of quadrature schemes for the discrete ordinates method*, Journal of Quantitative Spectroscopy & Radiative Transfer **84** (2004), 423–435.
17. Tarvainen T, Vauhkonen M, Kolehmainen V, and Kaipio JP, *Hybrid radiative-transfer-diffusion model for optical tomography*, Applied Optics **44** (2005), 876–886.
18. Wareing TA, McGhee JM, Morel JE, and Pautz SD, *Discontinuous finite element sn methods on three-dimensional unstructured grids*, Nuclear science and engineering **138** (2001), 256–268.

19. Trottenberg U, Oosterlee CW, and Schuller A, *Multigrid*, Academic Press, San Diego (2001).
20. Martin WR, Yehnert CE, Lorence L, and Duderstadt JJ, *Phase-space finite element methods applied to the first-order form of the transport equation*, Annuals of nuclear energy **8** (1981), 633–646.

Department of Mathematics, University of California, Irvine, CA 92697-3875, USA
E-mail address: haog@uci.edu, zhao@math.uci.edu