SIAM J. SCI. COMPUT.
Vol. 42, No. 4, pp. B869–B893
© 2020 Society for Industrial and Applied Mathematics

Downloaded 09/01/20 to 138.195.214.230. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

LOW-MEMORY, DISCRETE ORDINATES, DISCONTINUOUS GALERKIN METHODS FOR RADIATIVE TRANSPORT*

ZHENG SUN† AND CORY D. HAUCK‡

Abstract. The discrete ordinates discontinuous Galerkin (S_N-DG) method is a well-established and practical approach for solving the radiative transport equation. In this paper, we study a low-memory variation of the upwind S_N-DG method. The proposed method uses a smaller finite element space that is constructed by coupling spatial unknowns across collocation angles, thereby yielding an approximation with fewer degrees of freedom than the standard method. Like the original S_N-DG method, the low-memory variation still preserves the asymptotic diffusion limit and maintains the characteristic structure needed for mesh sweeping algorithms. While we observe second-order convergence in the scattering dominated, diffusive regime, the low-memory method is in general only first-order accurate. To address this issue, we use upwind reconstruction to recover second-order accuracy. For both methods, numerical procedures based on upwind sweeps are proposed to reduce the system dimension in the underlying Krylov solver strategy.

Key words. radiative transport, discrete ordinates, discontinuous Galerkin, diffusion limit

AMS subject classifications. 65N35, 65N22, 65F50, 35J05

DOI. 10.1137/19M1271956

1. Introduction.

Radiative transport equations [2, 8, 9, 12, 25, 26, 27] describe the flows of particles, such as photons, neutrons, and electrons, as they pass through and interact with a background medium. These equations are used in various applications, including astrophysics and nuclear reactor analysis.

In this paper, we consider the scaled, steady-state, linear transport equation

$$\Omega \cdot \nabla \Psi(\Omega, x) + \left(\frac{\sigma_{\mathrm{s}}(x)}{\varepsilon} + \varepsilon \sigma_{\mathrm{a}}(x) \right) \Psi(\Omega, x) = \frac{\sigma_{\mathrm{s}}(x)}{\varepsilon} \overline{\Psi}(x) + \varepsilon q(x), \quad (\Omega, x) \in S \times D, \tag{1.1a}$$

$$\Psi(\Omega, x) = g(\Omega, x), \quad (\Omega, x) \in \Gamma^{-}. \tag{1.1b}$$

*Submitted to the journal's Computational Methods in Science and Engineering section July 2, 2019; accepted for publication (in revised form) April 14, 2020; published electronically July 2, 2020.
https://doi.org/10.1137/19M1271956

Funding: This material was based, in part, upon work supported by the DOE Office of Advanced Scientific Computing Research. ORNL is operated by UT-Battelle, LLC, for the U.S. Department of Energy (DOE) under contract DE-AC05-00OR22725. This research is supported in part by an appointment with the NSF Mathematical Sciences Summer Internship Program sponsored by the National Science Foundation, Division of Mathematical Sciences (DMS). This program is administered by the Oak Ridge Institute for Science and Education (ORISE) through an interagency agreement between DOE and NSF. ORISE is managed by ORAU under DOE contract DE-SC0014664. The United States Government retains and the publisher, by accepting the article for publication, acknowledges that the United States Government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this manuscript, or allow others to do so, for United States Government purposes. The DOE will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan (http://energy.gov/downloads/doe-public-access-plan).

†Department of Mathematics, The Ohio State University, Columbus, OH 43210 (sun.2516@osu.edu).

‡Computational and Applied Mathematics Group, Oak Ridge National Laboratory, Oak Ridge, TN 37831, and Department of Mathematics, University of Tennessee, Knoxville TN 37996 (hauckc@ornl.gov).

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 09/01/20 to 138.195.214.230. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Here $D \subset \mathbb{R}^3$ is an open, bounded, and Lipschitz domain; S is the unit sphere in $\mathbb{R}^3$; and $\Gamma^- = \{(\Omega, x) \in S \times \partial D : \Omega \cdot n(x) < 0\}$ corresponds to the inflow boundary, where $n(x)$ is the outward unit normal vector at any point $x \in \partial D$ that the boundary is C^1. In cases that D possesses special symmetries, the angular flux Ψ may depend on only one or two spatial variables. In such cases, (1.1) may be reformulated with lower-dimensional domain $D_d \subset \mathbb{R}^d$ with $d = 1, 2$.

The *angular flux* Ψ is the flux of particles at the location x moving with unit speed in the direction Ω, and the *scalar flux* $\overline{\Psi} = \frac{1}{|S|}\int_S \Psi d\Omega$ is the average of Ψ over S.[1] The functions σ_{s} and σ_{a} are (known) nondimensionalized scattering and absorption cross sections, respectively, and q is a (known) nondimensionalized source. The function $g(\Omega, x)$ is the (known) incoming flux at $x \in \partial D$ moving in the direction Ω. The constant $\varepsilon > 0$ is a scaling parameter which characterizes the relative strength of scattering.

Designing effective numerical methods for (1.1) is a serious challenge, and the intent of this paper is to address two of the main issues. First, for a three-dimensional problem, the unknown intensity Ψ is a function of three spatial and two angular variables; the discretization of this five-dimensional phase space usually requires significant computational resources. Second, when the parameter ε is small, Ψ is nearly independent of Ω and can be approximated by the solution of a diffusion equation in the variable x only [5, 6, 16]. That is, away from the boundary, $\Psi(\Omega, x) = \Psi^{(0)}(x) + O(\varepsilon)$ as $\varepsilon \to 0$, where $\Psi^{(0)}$ satisfies

$$-\nabla \cdot \left(\frac{1}{3\sigma_{\mathrm{s}}}\nabla\Psi^{(0)}(x)\right) + \sigma_{\mathrm{a}}\Psi^{(0)}(x) = q(x), \quad x \in D, \tag{1.2}$$

along with appropriate boundary conditions. A numerical method for (1.1) should preserve this asymptotic limit without having to resolve the length scales associated with ε [19]. In other words, in the limit $\varepsilon \to 0$, a discretization of the transport equation (1.1) should become a consistent and stable discretization of the diffusion equation (1.2). Otherwise a highly refined mesh is needed to approximate the solution accurately [22].[2]

Classical approaches for discretizing (1.1) often involve separate treatment of the angular and spatial variables, and a variety of options are available. Among them, the discrete ordinates discontinuous Galerkin (S_N-DG) method [1, 17, 21] has received significant attention due to its robustness, computational efficiency, and convenient implementation. The discrete ordinates (S_N) method (see [23] for a substantial review and additional references) is a collocation method in which the angular variable Ω is discretized into a finite number of directions and a quadrature rule is used to evaluate $\overline{\Psi}$. The S_N discretization preserves nonnegativity of Ψ and can incorporate the boundary conditions from (1.1) in a straightforward way. It also preserves the characteristic structure of the advection operator in (1.1), which allows for the use of fast sweeping techniques for inverting the discrete form of the operator on the left-hand side of (1.1).

Discontinuous Galerkin (DG) methods are a class of finite element methods that construct numerical solutions using piecewise polynomial spaces. The DG approach was introduced in [29] for the express purpose of solving equations like (1.1), followed shortly thereafter by a rigorous analysis in [24]. Since then, DG methods have been applied to nonlinear hyperbolic conservation laws and convection-dominated

[1] Often the quantity $\Phi = 4\pi\overline{\Psi}$ is referred to as the scalar flux. The difference is simply a normalization factor from integration of the sphere. Here, we borrow the convention used in [25].

[2] This issue is also known as "locking" in the elliptic literature [4].

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

TABLE 1.1

Memory costs of standard S_N-DG and the low-memory variation, for both triangles and rectangles, for reduced spatial dimension d. The first two rows give the number of unknowns per spatial cell for each approach. The last row is the asymptotic ratio of the memory costs by two methods when n_Ω becomes large.

Unknowns per cell	Triangles (P^1)	Rectangles (Q^1)
Standard S_N-DG	$(d+1)n_\Omega$	$2^d n_\Omega$
Low-memory S_N-DG	$n_\Omega + d$	$(n_\Omega - 1) + 2^d$
Memory cost ratio as $n_\Omega \gg 1$	$d+1$	2^d

problems [7], elliptic problems [3], and equations with higher-order derivatives [31, 32]. When used with upwind fluxes, DG methods preserve the characteristic structure of (1.1) that enables sweeps. Moreover, if the approximation space can support globally continuous linear polynomials, then DG methods with upwind fluxes will yield accurate numerical solutions for Ψ without the need to resolve ε with the spatial mesh [1, 13, 21]. However, this condition on the approximation space means that at least P^1 elements must be used for a triangular mesh and Q^1 elements for a rectangular mesh.[3]

In order to reduce memory costs in the upwind S_N-DG method, while still preserving the asymptotic diffusion limit and maintaining the characteristic structure needed for sweeps, we propose in this paper to couple the finite element spaces between different collocation angles in the discrete ordinate approximation. Since the solution becomes isotropic in the diffusion limit ($\varepsilon \to 0$), we hypothesize that only a P^1 (for triangles) or Q^1 (for rectangles) approximation of the angular average is necessary. Thus, instead of using a tensor product finite element space for the S_N-DG system, we seek the solution in a proper subspace, in which all the elements have isotropic slopes. This choice of finite element space yields a significant reduction in memory per spatial cell, as illustrated in Table 1.1.

In the diffusion limit, the low-memory approach typically displays second-order accuracy. However, because the finite element representation of each ordinate is coupled to all the other ordinates, the overall accuracy of the low-memory approach for fixed ε is only first-order. To address this drawback, we propose a modification of the low-memory scheme that uses local reconstruction to improve accuracy. As long as the reconstruction uses upwind information, the resulting transport operator can still be inverted with sweeps. While rigorous theoretic properties of this modified scheme are still under investigation, we observe numerically that it recovers second-order accuracy for arbitrary fixed ε and captures the asymptotic diffusion limit. However, the method does generate some small numerical artifacts at the discontinuity of the cross section, which we point out in the numerical results of section 4.

The rest of the paper is organized as follows. In section 2, we introduce the background and revisit the S_N-DG method. Low-memory methods, including the original first-order approach and the second-order reconstructed scheme, are detailed in section 3. Numerical tests are provided in section 4 to illustrate the behavior of both approaches. Finally, conclusions and future work are discussed in section 5.

[3]This condition can be circumvented for nonupwind methods. In [28], the authors made the piecewise constant DG method asymptotic-preserving with parameters adjusting numerical fluxes under different regimes. Similar techniques were introduced in finite volume contexts [20] as well and were recently used in [15] to develop a positive, asymptotic-preserving method.

Downloaded 09/01/20 to 138.195.214.230. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 09/01/20 to 138.195.214.230. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

2. The S_N-DG method. In this section, we review the S_N-DG scheme and discuss its asymptotic properties and implementation. Throughout the paper, we consider the case $\inf_{x\in D}\sigma_{\rm s}(x)=\delta_{\rm s}>0$ and $\inf_{x\in D}\sigma_{\rm a}(x)=\delta_{\rm a}>0$, unless otherwise stated. In general, the well-posedness of (1.1) also holds for $\sigma_{\rm a}\geq 0$ [30]. In some places, we will also assume that the cross section is piecewise constant, either to simplify the exposition or to make connections between first- and second-order forms of the diffusion limit. In the numerics, we often consider nonzero boundary conditions. However, in proofs we often assume that $g=0$. When g is nonzero but isotropic, many of the results still hold. However, when g is anisotropic, the diffusion equation requires a boundary layer correction in order to be uniformly accurate [16]. At the discrete level, this situation requires more sophisticated analysis [1, 13, 21] than is presented here.

2.1. Formulation. Consider a quadrature rule with points $\{\Omega_j\}_{j=1}^{n_\Omega}$ and positive weights $\{\alpha_j\}_{j=1}^{n_\Omega}$ such that

$$\frac{1}{|S|}\int_S f(\Omega)d\Omega\approx\sum_{j=1}^{n_\Omega}\alpha_j f(\Omega_j)\quad\forall f\in C(S). \tag{2.1}$$

We assume the quadrature is exact for polynomials in Ω up to degree two,[4] that is,

$$\text{(i)}\ \sum_{j=1}^{n_\Omega}\alpha_j=1,\quad \text{(ii)}\ \sum_{j=1}^{n_\Omega}\alpha_j\Omega_j=0,\quad\text{and}\quad \text{(iii)}\ \sum_{j=1}^{n_\Omega}\alpha_j\Omega_j\otimes\Omega_j=\frac{1}{3}\,\mathrm{Id}\,. \tag{2.2}$$

The S_N method approximates the angular flux Ψ at the quadrature points $\{\Omega_j\}_{j=1}^{n_\Omega}$ by a vector-valued function $\psi(x)=(\psi_1(x),\psi_2(x),\dots,\psi_{n_\Omega}(x))$ whose components satisfy a coupled system with n_Ω equations

$$\Omega_j\cdot\nabla\psi_j(x)+\left(\frac{\sigma_{\rm s}}{\varepsilon}+\varepsilon\sigma_{\rm a}\right)\psi_j(x)=\frac{\sigma_{\rm s}}{\varepsilon}\overline{\psi}(x)+\varepsilon q(x),\qquad \overline{\psi}(x)=\sum_{j=1}^{n_\Omega}\alpha_j\psi_j(x). \tag{2.3}$$

To formulate the upwind DG discretization of the S_N system (2.3), let $\mathcal{T}_h=\{K\}$ be a quasi-uniform partition of the domain D. We assume $D=\cup_{K\in\mathcal{T}_h}\mathrm{cl}(K)$ to avoid unnecessary technicalities. Let $\mathcal{F}_h=\cup_{K\in\mathcal{T}_h}\partial K$ be the collection of cell interfaces and let $\mathcal{F}_h^\partial$ be the collection of boundary faces. Given a cell K, we denote by ν_K the outward normal on ∂K and for any $x\in\partial K$, let $v^{\rm int}(x)=\lim_{\delta\to0^+}v(x-\delta\nu_K)$ and $v^{\rm ext}(x)=\lim_{\delta\to0^+}v(x+\delta\nu_K)$. Given a face F, we denote by ν_F a prescribed normal (chosen by convention) and, for any $x\in F$, let $v^\pm=\lim_{\delta\to0^+}v(x\pm\delta\nu_F)$. For convenience, we assume trace values are identically zero when evaluated outside of D.

The standard S_N-DG method uses the tensor product finite element space

$$\mathcal{V}_h=\prod_{j=1}^{n_\Omega}V_h,\qquad V_h=\{v_j:v_j|_K\in Z_1(K)\}, \tag{2.4}$$

where for triangular or tetrahedral meshes, $Z_1(K)$ is the space $P^1(K)$ of linear polynomials on K and for Cartesian meshes $Z_1(K)$ is the space $Q^1(K)$ of multilinear

[4] Level symmetric quadratures of moderate size will satisfy these properties. See, e.g., [25] and references therein.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 09/01/20 to 138.195.214.230. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

polynomials on K. The space $\mathcal{V}_h$ can be equipped with an inner product $(\cdot,\cdot)$ and associated norm $\|\cdot\|$ given by

$$(u,v) = \sum_{K\in\mathcal{T}_h}\sum_{j=1}^{n_\Omega}\alpha_j\int_K u_j v_j dx \qquad \text{and} \qquad \|v\| = \sqrt{(v,v)}. \tag{2.5}$$

The seminorm induced by jumps at the cell interfaces is given by

$$[\![v]\!] = \left(\sum_{F\in\mathcal{F}_h}\sum_{j=1}^{n_\Omega}\alpha_j\int_F |\Omega_j\cdot\nu_F|(v_j^- - v_j^+)^2 dx\right)^{1/2}. \tag{2.6}$$

To construct the S_N-DG method, define the local operators

$$L_{j,K}(u,v) = -\int_K u_j\Omega_j\cdot\nabla v_j dx + \int_{\partial K}\widehat{u}_j\Omega_j\cdot\nu_K v_j^{\text{int}} dx + \int_K\left(\frac{\sigma_{\text{s}}}{\varepsilon}+\varepsilon\sigma_{\text{a}}\right)u_j v_j dx, \tag{2.7a}$$

$$S_{j,K}(u,v) = \int_K \frac{\sigma_{\text{s}}}{\varepsilon}\overline{u}v_j dx, \quad \text{with } \overline{u} = \sum_{j=1}^{n_\Omega}\alpha_j u_j, \tag{2.7b}$$

$$Q_{j,K,g}(v) = \int_K \varepsilon q v_j dx - \int_{\partial K\cap\mathcal{F}_h^\partial} g\Omega_j\cdot\nu_K v_j^{\text{int}} dx, \tag{2.7c}$$

where $\widehat{u}_j(x) = \lim_{\delta\to 0^+} u_j(x-\delta\Omega_j)$ is the upwind trace at $x\in\partial K$ and is defined as zero when the limit is taken outside of D. Then set

$$B(u,v) = L(u,v) - S(u,v), \tag{2.8}$$

where

$$L(u,v) = \sum_{K\in\mathcal{T}_h}\sum_{j=1}^{n_\Omega}\alpha_j L_{j,K}(u,v) \quad \text{and} \quad S(u,v) = \sum_{K\in\mathcal{T}_h}\sum_{j=1}^{n_\Omega}\alpha_j S_{j,K}(u,v), \tag{2.9}$$

and let

$$Q_g(v) = \sum_{K\in\mathcal{T}_h}\sum_{j=1}^{n_\Omega}\alpha_j Q_{j,K,g}(v). \tag{2.10}$$

The S_N-DG method can then be written as follows: *find* $\psi_h = (\psi_{h,1},\dots,\psi_{h,n_\Omega})\in\mathcal{V}_h$ *such that*

$$B(\psi_h, v) = Q_g(v) \qquad \forall v\in\mathcal{V}_h. \tag{2.11}$$

2.2. Implementation. Recall that n_Ω is the number of discrete ordinates in the S_N discretization. Let $n_x = |\mathcal{T}_h|$ be the number of mesh cells in $\mathcal{T}_h$ and let n_P be the dimension of $Z_1(K)$. Then the dimension of $\mathcal{V}_h$ is $n_\Omega\cdot n_x\cdot n_P$.

Let $\{b^{p,r} : p = 1,\dots,n_x, r = 0,\dots,n_P-1\}$ be a set of basis functions for V_h, with $b^{p,r}$ locally supported on $K_p\in\mathcal{T}_h$. Then the set $\mathbb{B} = \{\xi^{l,p,r} : l = 1,\dots,n_\Omega, p = 1,\dots,n_x, r = 0,\dots,n_P-1\}$, where $\xi_j^{l,p,r}(x) = \delta_{lj}b^{p,r}(x)$ $(j = 1,\dots n_\Omega)$ and δ is the Kronecker delta, gives a complete set of basis functions for $\mathcal{V}_h$. With this choice of basis functions, the variational formulation in (2.11), written as

$$L(\psi_h, v) = S(\psi_h, v) + Q_g(v) \qquad \forall v\in\mathcal{V}_h, \tag{2.12}$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 09/01/20 to 138.195.214.230. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

can be assembled into a linear system (detailed in Appendix A)

$$\mathbf{L}\boldsymbol{\Psi} = \mathbf{MP}\boldsymbol{\Psi} + \mathbf{Q}. \tag{2.13}$$

In the above equation, $\mathbf{L}$ is an $(n_\Omega \cdot n_x \cdot n_P) \times (n_\Omega \cdot n_x \cdot n_P)$ block diagonal matrix, where the jth block ($j = 1, \dots n_\Omega$) corresponds to the discretization of the operator $\psi_j \to \Omega_j \cdot \nabla\psi_j + \left(\frac{\sigma_s}{\varepsilon} + \varepsilon\sigma_a\right)\psi_j$; $\mathbf{M}$ is an injective $(n_\Omega \cdot n_x \cdot n_P) \times (n_x \cdot n_P)$ matrix, and $\mathbf{P}$ is an $(n_x \cdot n_P) \times (n_\Omega \cdot n_x \cdot n_P)$ matrix; $\mathbf{Q}$ is an $(n_\Omega \cdot n_x \cdot n_P)$ vector assembled from the source q and the inflow boundary g; and $\boldsymbol{\Psi} = (\psi^{l,p,r})$ is an $(n_\Omega \cdot n_x \cdot n_P)$ vector such that $\psi_h = \sum_{l,p,r} \psi^{l,p,r}\xi^{l,p,r}$.

If upwind values are used to evaluate the numerical trace $\widehat{u}_j$, each block of $\mathbf{L}$ can be inverted efficiently with a sweep algorithm. The system in (2.13) can be solved numerically with a Krylov method by first solving the reduce system

$$\boldsymbol{\Phi} - \mathbf{PL}^{-1}\mathbf{M}\boldsymbol{\Phi} = \mathbf{PL}^{-1}\mathbf{Q} \tag{2.14}$$

for the $n_x \cdot n_P$ vector $\boldsymbol{\Phi} := \mathbf{P}\boldsymbol{\Psi}$. This equation is derived by applying $\mathbf{L}^{-1}$ and then $\mathbf{P}$ to (2.13). In a second step $\boldsymbol{\Psi}$ is recovered from the relation

$$\boldsymbol{\Psi} = \mathbf{L}^{-1}\mathbf{M}\boldsymbol{\Phi} + \mathbf{L}^{-1}\mathbf{Q}. \tag{2.15}$$

The following theorem is proven in Appendix B.

THEOREM 2.1. *The matrix* $\mathbf{I}_{n_x \cdot n_P} - \mathbf{PL}^{-1}\mathbf{M}$ *is invertible.*

Remark 2.1 (Sherman–Morrison formula). According to the Sherman–Morrison formula (see, for example, [11, section 2.1.3]), given invertible matrices $\mathbf{B} = \mathbf{A} + \mathbf{UV}$ and $\mathbf{I} + \mathbf{VA}^{-1}\mathbf{U}$,

$$\mathbf{B}^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1}\mathbf{U}(\mathbf{I} + \mathbf{VA}^{-1}\mathbf{U})^{-1}\mathbf{VA}^{-1}. \tag{2.16}$$

The direct application of (2.16) with $\mathbf{A} = \mathbf{L}$, $\mathbf{U} = -\mathbf{M}$, and $\mathbf{V} = \mathbf{P}$ yields the formula in (2.15) with $\boldsymbol{\Phi}$ given by (2.14).

2.3. Asymptotic scheme. As $\varepsilon \to 0$, the S_N-DG scheme gives a consistent approximation to the asymptotic diffusion problem. For simplicity, we focus here on the zero inflow boundary condition $g = 0$. The analysis of more general boundary conditions can be found in [1, 13, 14, 21].

We use an overline to represent isotropic subspaces. In other words, a function in an overlined space has the same component for each angle. For example,

$$\overline{\mathcal{V}}_h = \{v = (v_1, \dots, v_{n_\Omega}) \in \mathcal{V}_h : v_j = \overline{v} \in V_h, j = 1, \dots, n_\Omega\} \tag{2.17}$$

is defined as the isotropic subspace of $\mathcal{V}_h$. Since the vector-valued space $\overline{\mathcal{V}}_h$ is isomorphic with the scalar-valued space V_h, one can identify $\overline{\mathcal{V}}_h$ with V_h. Throughout the paper, all overlined spaces are independent of angles and depend on the spatial variable only, including the spaces $\overline{\mathcal{V}}_h^3$, defined below, and $\overline{\Omega\mathcal{W}_h}$, defined in (3.3).

We let $\mathcal{C}_{h,\text{zero}}$ be the space of continuous functions in $\overline{\mathcal{V}}_h$ that vanish on ∂D and

$$\overline{\mathcal{V}}_h^3 = \{(\varphi_1, \varphi_2, \varphi_3) : \varphi_i \in \overline{\mathcal{V}}_h\} \tag{2.18}$$

be the tensor product space of $\overline{\mathcal{V}}_h$, with an induced norm still denoted by $\|\cdot\|$. To facilitate the discussion, we also define

$$J_h = \frac{1}{\varepsilon}\sum_{j=1}^{n_\Omega} \alpha_j\Omega_j\psi_{h,j} = \sum_{j=1}^{n_\Omega} \alpha_j\Omega_j\frac{\psi_{h,j} - \overline{\psi}_h}{\varepsilon}, \tag{2.19}$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 09/01/20 to 138.195.214.230. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

which is a vector field in $\mathbb{R}^3$. The following result is proved in [13];[5] see also [1] and Theorem 3.2 in this paper.

THEOREM 2.2 (asymptotic scheme). *Suppose $g = 0$. Then as $\varepsilon \to 0$, $(\psi_h)_{\varepsilon>0}$ and $(J_h)_{\varepsilon>0}$ converge to $\psi_h^{(0)} = \overline{\psi}_h^{(0)} \in \mathcal{C}_{h,\mathrm{zero}}$ and $J_h^{(0)} \in \overline{\mathcal{V}}_h^3$, respectively, that are the unique solution to the mixed problem:*

$$\sum_{K\in\mathcal{T}_h} \int_K \left(-J_h^{(0)} \cdot \nabla\varphi + \sigma_{\mathrm{a}} \psi_h^{(0)} \varphi\right) dx = \int_D q\varphi dx, \tag{2.20a}$$

$$\sum_{K\in\mathcal{T}_h} \int_K \left(\frac{1}{3}\nabla\psi_h^{(0)} + \sigma_{\mathrm{s}} J_h^{(0)}\right) \cdot \zeta dx = 0 \tag{2.20b}$$

$\forall\varphi \in \mathcal{C}_{h,\mathrm{zero}}$ and $\forall\zeta \in \overline{\mathcal{V}}_h^3$.

3. Low-memory strategies. In this section, we generalize the statement of Theorem 2.2 slightly to allow for proper subspaces of $\mathcal{V}_h$ in the finite element formulation. Based on the analysis, a first-order low-memory scheme is constructed. We then apply the reconstruction technique to lift the accuracy of the method to second-order.

3.1. Asymptotic schemes with subspaces of $\mathcal{V}_h$. The results of Theorem 2.2 suggest that, rather than ψ_h, it is the approximation of the integrated quantities $\overline{\psi}_h$ and J_h that play an important role in the diffusion limit. In particular, the continuity requirement on $\overline{\psi}_h^{(0)}$ is crucial. Indeed, as is well known [1], if the space V_h is constructed from piecewise constants, then (2.20) implies that $\psi_h^{(0)}$ is a global constant and $J_h^{(0)} = 0$. This solution is clearly inconsistent with the diffusion limit. However, it is possible to construct a DG method: *find $\psi_h = (\psi_{h,1}, \ldots, \psi_{h,n_\Omega}) \in \mathcal{W}_h$ such that*

$$B(\psi_h, v) = Q_g(v) \qquad \forall v \in \mathcal{W}_h \tag{3.1}$$

based on a proper subspace $\mathcal{W}_h \subset \mathcal{V}_h$ that maintains the diffusion limit but requires fewer unknowns for a given mesh $\mathcal{T}_h$.

THEOREM 3.1. *For each $\varepsilon > 0$ and linear subspace $\mathcal{W}_h \subset \mathcal{V}_h$, (3.1) has a unique solution. In particular, if $g = 0$, the solution satisfies the energy estimate*

$$\frac{1}{\varepsilon}\left\|\sigma_{\mathrm{s}}^{\frac{1}{2}}(\psi_h - \overline{\psi}_h)\right\|^2 + \frac{\varepsilon}{2}\left\|\sigma_{\mathrm{a}}^{\frac{1}{2}}\psi_h\right\|^2 + \frac{1}{2}[\![\psi_h]\!]^2 \le \frac{\varepsilon}{2\delta_{\mathrm{a}}}\|q\|^2. \tag{3.2}$$

The proof is based on coercivity of $B(\cdot,\cdot)$ and we refer to [17] and [13] for details. Here, $g = 0$ is assumed for simplicity. Energy estimates with general inflow boundary condition can be found in [13, Lemma 4.2]. In [17], the case $\varepsilon = 1$ is studied and error estimates are derived using the coercivity with respect to a modified norm.

We next characterize sufficient conditions for $\mathcal{W}_h$. Define the spaces

$$\overline{\Omega\mathcal{W}_h} := \left\{\sum_{j=1}^{n_\Omega} \alpha_j \Omega_j v_j : v \in \mathcal{W}_h\right\} \subset \overline{\mathcal{V}}_h^3 \quad \text{and} \quad \Omega\cdot\overline{\Omega\mathcal{W}_h} := \{\Omega\cdot\zeta : \zeta \in \overline{\Omega\mathcal{W}_h}\} \subset \mathcal{V}_h, \tag{3.3}$$

[5] The result in [13] is actually stated for more general settings. In particular it allows g to be nonzero and possibly anisotropic.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 09/01/20 to 138.195.214.230. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

where $\Omega \cdot \zeta := (\Omega_1 \cdot \zeta, \dots, \Omega_{n_\Omega} \cdot \zeta)$. According to (2.19), $J_h \in \overline{\Omega \mathcal{W}_h}$. Theorem 2.2 can now be generalized to the space $\mathcal{W}_h$.

THEOREM 3.2. *Suppose $g = 0$ and $\mathcal{W}_h \subset \mathcal{V}_h$ is a linear space such that $\Omega \cdot \overline{\Omega \mathcal{W}_h} \subset \mathcal{W}_h$. Then as $\varepsilon \to 0$, $(\psi_h)_{\varepsilon>0}$ and $(J_h)_{\varepsilon>0}$ converge to $\psi_h^{(0)} = \overline{\psi}_h^{(0)} \in \mathcal{C}_{h,\mathrm{zero}} \cap \mathcal{W}_h$ and $J_h^{(0)} \in \overline{\Omega \mathcal{W}_h}$, respectively, that are the unique solution to the mixed problem* (2.20), *$\forall \varphi \in \mathcal{C}_{h,\mathrm{zero}} \cap \mathcal{W}_h$ and $\forall \zeta \in \overline{\Omega \mathcal{W}_h}$.*

Proof. Because the proof follows the arguments in [13, section 4] closely, we provide only a brief outline, emphasizing where the condition on the space $\mathcal{W}_h$ plays a role.

1. The stability estimate in (3.2) provides the following three bounds:

   (3.4)
   $$\text{(i) } \|\psi_h\|^2 \le \frac{1}{\delta_{\mathrm{a}}^2}\|q\|^2, \quad \text{(ii) } \|\psi_h - \overline{\psi}_h\|^2 \le \frac{\varepsilon^2}{\delta_{\mathrm{a}}\delta_{\mathrm{s}}}\|q\|^2, \quad \text{and} \quad \text{(iii) } [\![\psi_h]\!]^2 \le \frac{\varepsilon}{\delta_{\mathrm{a}}}\|q\|^2.$$

   Bounds (i) and (ii) imply that ψ_h converges (via a subsequence) to a function $\psi_h^{(0)} \in \overline{\mathcal{V}}_h$. Bound (iii) implies that $\psi_h^{(0)} \in \mathcal{C}_{h,\mathrm{zero}} \cap \mathcal{W}_h = \mathcal{C}_{h,\mathrm{zero}} \cap \overline{\mathcal{W}}_h$.
2. Since, from the definition in (2.19),
   $$\|J_h\| \le \sum_{j=1}^{n_\Omega} \alpha_j \frac{\|\psi_h - \overline{\psi}_h\|}{\varepsilon}, \tag{3.5}$$
   where $\|J_h\|$ is the tensor product norm of J_h in $\overline{\mathcal{V}}_h^3$, the bound (ii) implies further that $(J_h)_{\varepsilon>0} \subset \overline{\Omega \mathcal{W}_h}$ is uniformly bounded and hence converges subsequentially to a limit $J_h^{(0)} \in \overline{\Omega \mathcal{W}_h}$.
3. The equation in (2.20a) is derived by testing (3.1) with $v = \varphi \in \mathcal{C}_{h,\mathrm{zero}} \cap \mathcal{W}_h$ and using the fact that φ is independent of Ω and continuous in x.
4. It is the derivation of (2.20b) which uses the condition $\Omega \cdot \overline{\Omega \mathcal{W}_h} \subset \mathcal{W}_h$. Specifically, if $v = \Omega \cdot \zeta$ with $\zeta \in \overline{\Omega \mathcal{W}_h}$, then this condition implies that $v \in \mathcal{W}_h$. Therefore, we can test (3.1) with this choice of v to find that

   (3.6)
   $$\begin{aligned} L(\psi_h, \Omega\cdot\zeta) - S(\psi_h, \Omega\cdot\zeta) &= -\sum_{j=1}^{n_\Omega} \alpha_j \sum_{K\in\mathcal{T}_h} \int_K \psi_{h,j}\Omega_j \cdot \nabla(\Omega_j\cdot\zeta)dx \\ &\quad + \sum_{j=1}^{n_\Omega} \alpha_j \sum_{K\in\mathcal{T}_h} \int_{\partial K} \widehat{\psi}_{h,j}(\Omega_j\cdot\nu_K)(\Omega_j\cdot\zeta^{\mathrm{int}})dx \\ &\quad + \sum_{j=1}^{n_\Omega} \alpha_j \sum_{K\in\mathcal{T}_h} \int_K \Big(\Big(\frac{\sigma_{\mathrm{s}}}{\varepsilon} + \varepsilon\sigma_{\mathrm{a}}\Big)\psi_{h,j} - \frac{\sigma_{\mathrm{s}}}{\varepsilon}\overline{\psi}_h\Big)(\Omega_j\cdot\zeta)dx \\ &=: I + II + III. \end{aligned}$$

   We combine I and II, using the fact that $\overline{\psi}_h^{(0)} \in \mathcal{C}_{h,\mathrm{zero}}$ and invoking (2.2). This gives

   (3.7)
   $$\begin{aligned} \lim_{\varepsilon\to 0}(I+II) &= \sum_{j=1}^{n_\Omega} \alpha_j(\Omega_j\otimes\Omega_j) : \sum_{K\in\mathcal{T}_h}\Big(-\int_K \overline{\psi}_h^{(0)}\nabla\zeta dx + \int_{\partial K}\overline{\psi}_h^{(0)}\nu_K\otimes\zeta^{\mathrm{int}}dx\Big) \\ &= \frac{1}{3}\,\mathrm{Id} : \sum_{K\in\mathcal{T}_h}\int_K \nabla\overline{\psi}_h^{(0)}\otimes\zeta dx = \sum_{K\in\mathcal{T}_h}\int_K \frac{1}{3}\nabla\overline{\psi}_h^{(0)}\cdot\zeta dx. \end{aligned}$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 09/01/20 to 138.195.214.230. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Since $\sum_{j=1}^{n_\Omega} \alpha_j \overline{\psi}_h \Omega_j = 0$, it can be shown that

(3.8)
$$\lim_{\varepsilon\to 0} III = \lim_{\varepsilon\to 0} \sum_{K\in\mathcal{T}_h} \int_K \Big(\frac{\sigma_{\rm s}}{\varepsilon} + \varepsilon\sigma_{\rm a}\Big) \sum_{j=1}^{n_\Omega} (\alpha_j \psi_{h,j} \Omega_j)\cdot\zeta dx = \int_K \sigma_{\rm s} J_h^{(0)}\cdot\zeta dx.$$

Finally, the right-hand side of (3.1) is (for $g = 0$)

$$Q_0(v) = \sum_{j=1}^{n_\Omega} \alpha_j \sum_{K\in\mathcal{T}_h} \int_K \Omega_j\cdot\zeta q dx = 0. \tag{3.9}$$

Combining (3.7), (3.8), and (3.9) recovers (2.20b).

5. Uniqueness of the subsequential limits $\psi_h^{(0)}$ and $J_h^{(0)}$ follows from the unisolvency of (2.20). Indeed if $(\widetilde{\psi}_h, \widetilde{J}_h)$ is the difference between any two solutions of (2.20), then

$$3\sigma_{\rm s}\|\widetilde{J}_h\|^2 + \sigma_{\rm a}\|\widetilde{\psi}_h\|^2 = 0. \tag{3.10}$$

Since $\sigma_{\rm s}$ and $\sigma_{\rm a}$ are assumed positive, it follows that $\widetilde{\psi}_h$ and $\widetilde{J}_h$ are identically zero. ∎

We next discuss different choices of $\mathcal{W}_h$ and the corresponding pair, $\mathcal{S}_h := \mathcal{C}_{h,\rm zero} \cap \mathcal{W}_h$ and $\mathcal{J}_h := \overline{\Omega\mathcal{W}_h}$, which arise in the diffusion limit. Let $Z_0(K)$ be the space spanned by constants on K, and define the piecewise constant space $\mathcal{V}_{h,0}$ and its (cellwise) orthogonal complement $\mathcal{V}_{h,1}$,

$$\mathcal{V}_{h,0} = \{v\in\mathcal{V}_h \colon v_j|_K \in Z_0(K) \quad \forall K\in\mathcal{T}_h\}, \tag{3.11a}$$

$$\mathcal{V}_{h,1} = \Big\{v\in\mathcal{V}_h \colon \int_K v_j dx = 0 \quad \forall K\in\mathcal{T}_h\Big\} = \mathcal{V}_{h,0}^{\perp}, \tag{3.11b}$$

so that $\mathcal{V}_h = \mathcal{V}_{h,0} + \mathcal{V}_{h,1}$. The isotropic subspace of $\mathcal{V}_{h,r}$ is denoted by $\overline{\mathcal{V}}_{h,r}$ and the subsequent product space is denoted by $\overline{\mathcal{V}}_{h,r}^3$, $r = 0, 1$.

1. When $\mathcal{W}_h = \mathcal{V}_{h,0}$ or $\mathcal{W}_h = \{v\in\mathcal{V}_h : v_j|_K \in P_1(K), \forall K\in\mathcal{T}_h\}$, we have $\mathcal{S}_h = \{0\}$, which implies $\psi_h^{(0)} = 0$ and $J_h^{(0)} = 0$.
2. When $\mathcal{W}_h = \mathcal{V}_{h,0} + \overline{\mathcal{V}}_{h,1} + \Omega\cdot\overline{\mathcal{V}}_{h,1}^3$, it can be shown that $\mathcal{S}_h = \mathcal{C}_{h,\rm zero}$, $\mathcal{J}_h = \overline{\mathcal{V}}_h^3$,[6] and $\Omega\cdot\mathcal{J}_h \subset \mathcal{W}_h$. The asymptotic scheme is the same as that of the original S_N-DG method. If $\sigma_{\rm s}$ and $\sigma_{\rm a}$ are both piecewise constant, then the asymptotic scheme has the primal form, *Find* $\psi_h^{(0)} \in \mathcal{C}_{h,\rm zero}$*, such that*

$$\sum_{K\in\mathcal{T}_h} \int_K \left(\frac{1}{3\sigma_{\rm s}} \nabla\psi_h^{(0)}\cdot\nabla\varphi + \sigma_{\rm a}\psi_h^{(0)}\varphi\right) dx = \int_D q\varphi dx \tag{3.12}$$

$\forall\varphi\in\mathcal{C}_{h,\rm zero}$. This is the classical continuous Galerkin approximation, which is stable and second-order accurate.
3. When $\mathcal{W}_h = \mathcal{V}_{h,0} + \overline{\mathcal{V}}_{h,1}$, then $\mathcal{S}_h = \mathcal{C}_{h,\rm zero}$, $\mathcal{J}_h = \overline{\mathcal{V}}_{h,0}^3$, and $\Omega\cdot\mathcal{J}_h \subset \mathcal{W}_h$. With P^1 elements and triangular meshes, the asymptotic scheme is essentially the P_N scheme suggested by Egger and Schlottbom in [10] with $N = 1$. If

[6] Since $\overline{\mathcal{V}}_h^3 \supset \mathcal{J}_h = \overline{\Omega\mathcal{W}_h} \supset \overline{\Omega(\Omega\cdot\overline{\mathcal{V}}_h^3)} = \overline{\mathcal{V}}_h^3$, which forces $\mathcal{J}_h = \overline{\mathcal{V}}_h^3$. Here we have used (iii) in (2.2) for the last equality.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Q^1 elements and Cartesian meshes are used, the scheme yields the same variational form as that in [10], while the space pair no longer satisfies the condition $\nabla \mathcal{S}_h \subset \mathcal{J}_h$.

From another point of view, suppose σ_s and σ_a are piecewise constant, then the primal form can be reduced as follows: *find* $\psi_h^{(0)} \in \mathcal{C}_{h,\mathrm{zero}}$*, such that*

$$\sum_{K\in\mathcal{T}_h}\int_K \left(\frac{1}{3\sigma_\mathrm{s}}\Pi_0\left(\nabla\psi_h^{(0)}\right)\cdot\Pi_0(\nabla\varphi)+\sigma_\mathrm{a}\psi_h^{(0)}\varphi\right)dx=\sum_{K\in\mathcal{T}_h}\int_K q\varphi dx \tag{3.13}$$

$\forall\varphi\in\mathcal{C}_{h,\mathrm{zero}}$. For P^1 elements on triangular meshes, (3.13) is identical to (3.12). For Q_1 elements on Cartesian meshes, one can show that (3.13) is unisolvent. Furthermore, $\|\Pi_0(\nabla\psi_h^{(0)})\|^2+\|\psi_h^{(0)}\|^2\le\max(\frac{3\sigma_\mathrm{s}}{2\sigma_\mathrm{a}},\sigma_\mathrm{a}^{-2})\|q\|^2$ if $\sigma_\mathrm{a}\ge\delta_\mathrm{a}>0$. While the accuracy is hard to analyze under the finite element framework. Assume a uniform square mesh with cell length h. Let σ_s and σ_a be globally constant. Then (3.13) can be rewritten as a finite difference scheme under the Lagrange basis functions:

$$-\frac{\psi_{i-1,j-1}+\psi_{i-1,j+1}-4\psi_{i,j}+\psi_{i+1,j-1}+\psi_{i+1,j+1}}{3\sigma_\mathrm{s}\cdot 2h^2}+\sigma_\mathrm{a}A[\psi_{i,j}]=A[q_{i,j}], \tag{3.14}$$

$$\begin{aligned}A[\psi_{i,j}]:=&\frac{1}{36}\left(\psi_{i-1,j-1}+\psi_{i-1,j+1}+\psi_{i+1,j-1}+\psi_{i+1,j+1}\right)\\&+\frac{1}{9}\left(\psi_{i-1,j}+\psi_{i,j-1}+\psi_{i,j+1}+\psi_{i+1,j}\right)+\frac{4}{9}\psi_{i,j}.\end{aligned} \tag{3.15}$$

The truncation error of the method is $\mathcal{O}(h^2)$.

At first glance, $\mathcal{W}_h=\mathcal{V}_{h,0}+\overline{\mathcal{V}}_{h,1}+\Omega\cdot\overline{\mathcal{V}}_{h,1}^3$ seems to be the natural choice for constructing the low-memory scheme that preserves the correct diffusion limit. However, coupling between angles requires special treatment for reducing the system dimension. The extra moments $\Omega\cdot\overline{\mathcal{V}}_{h,1}^3$ will make the resulting system even larger than that of the original S_N-DG method. Although it may be worth while to include extra moments for problems with anisotropic scattering, for which a large system has to be solved anyway, we avoid this option for solving (1.1). We therefore explore the other choice $\mathcal{W}_h=\mathcal{V}_{h,0}+\overline{\mathcal{V}}_{h,1}$ in the rest of the paper.

3.2. Low-memory scheme. Based on the analysis and discussion of subsection 3.1, we propose a scheme that uses the finite element space

$$\mathcal{V}_h^{\mathrm{lm}}=\mathcal{V}_{h,0}+\overline{\mathcal{V}}_{h,1}. \tag{3.16}$$

Then with $\mathcal{W}_h=\mathcal{V}_h^{\mathrm{lm}}$, the low-memory S_N-DG scheme is written as follows: *find* $\psi_h\in\mathcal{V}_h^{\mathrm{lm}}$ *such that*

$$B(\psi_h,v)=Q_g(v)\qquad\forall v\in\mathcal{V}_h^{\mathrm{lm}}, \tag{3.17}$$

where B *and* Q_g *are defined in* (2.8) *and* (2.10)*, respectively.*

We now show that this scheme can be implemented using sweeps, i.e., a strategy analagous to the (2.14) and (2.15), which relies heavily on the fast inversion of the operator $\mathbf{L}$. For simplicity, we only consider the case σ_s being piecewise constant. The implementation is based on the block matrix formulation (2.13) of the S_N-DG method, with the sizes of blocks displayed below.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 09/01/20 to 138.195.214.230. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Downloaded 09/01/20 to 138.195.214.230. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

- Matrix $\mathbf{L}$:

$$\mathbf{L} = \left[\begin{array}{c:c} \underbrace{\mathbf{L}_{00}}_{n_\Omega \cdot n_x} & \underbrace{\mathbf{L}_{01}}_{n_\Omega \cdot n_x \cdot (n_P-1)} \\ \underbrace{\mathbf{L}_{10}}_{n_\Omega \cdot n_x} & \underbrace{\mathbf{L}_{11}}_{n_\Omega \cdot n_x \cdot (n_P-1)} \end{array}\right] \begin{array}{l} \} \, n_\Omega \cdot n_x \\ \} \, n_\Omega \cdot n_x \cdot (n_P-1) \end{array}. \tag{3.18}$$

Here $\mathbf{L}_{rr'}$ is the matrix block associated to $L(u,v)$ defined in (2.9) with $u \in \mathcal{V}_{h,r'}$ and $v \in \mathcal{V}_{h,r}$.

- Matrix $\mathbf{MP}$:

$$\mathbf{M} = \left[\begin{array}{c:c} \mathbf{M}_0 & \\ \underbrace{\phantom{\mathbf{M}_0}}_{n_x} & \underbrace{\mathbf{M}_1}_{n_x \cdot (n_P-1)} \end{array}\right] \begin{array}{l} \} \, n_\Omega \cdot n_x \\ \} \, n_\Omega \cdot n_x \cdot (n_P-1) \end{array} \tag{3.19}$$

$$\mathbf{P} = \left[\begin{array}{c:c} \mathbf{P}_0 & \\ \underbrace{\phantom{\mathbf{P}_0}}_{n_\Omega \cdot n_x} & \underbrace{\mathbf{P}_1}_{n_\Omega \cdot n_x \cdot (n_P-1)} \end{array}\right] \begin{array}{l} \} \, n_x \\ \} \, n_x \cdot (n_P-1) \end{array}, \tag{3.20}$$

which implies that

$$\mathbf{S} := \mathbf{MP} = \left[\begin{array}{c:c} \underbrace{\mathbf{M}_0\mathbf{P}_0}_{n_\Omega \cdot n_x} & \\ & \underbrace{\mathbf{M}_1\mathbf{P}_1}_{n_\Omega \cdot n_x \cdot (n_P-1)} \end{array}\right] \begin{array}{l} \} \, n_\Omega \cdot n_x \\ \} \, n_\Omega \cdot n_x \cdot (n_P-1) \end{array}. \tag{3.21}$$

The block $\mathbf{S}_{rr'} = \mathbf{M}_r\mathbf{P}_{r'}$ is associated to $S(u,v)$ defined in (2.9) with $u \in \mathcal{V}_{h,r'}$ and $v \in \mathcal{V}_{h,r}$.

- Vector $\mathbf{Q}$:

$$\mathbf{Q} = \left[\begin{array}{c} \mathbf{Q}_0 \\ \hdashline \mathbf{Q}_1 \end{array}\right] \begin{array}{l} \} \, n_\Omega \cdot n_x \\ \} \, n_\Omega \cdot n_x \cdot (n_P-1) \end{array}. \tag{3.22}$$

The vector block $\mathbf{Q}_r$ is associated to $Q_g(v)$ defined in (2.10) with $v \in \mathcal{V}_{h,r}$.

We now need to specify the basis for assembling the matrix blocks. Recall from subsection 2.2 that for each p, $\{b^{p,r}\}_{r=0}^{n_P-1}$ forms a basis for $Z_1(K_p)$, and $\xi_j^{l,p,r} = \delta_{lj} b^{p,r}$. We further assume $\{b^{p,r}\}_{r=0}^{n_P-1}$ is an orthogonal set and $\{b^{p,0}\}_{p=1}^{n_x}$ is a set of constant functions on K_p. Then the sets of basis functions for subspaces, denoted by $\mathbb{B}$ with corresponding subscripts, can be characterized as follows:

- $\mathbb{B}_0$ forms the basis of $\mathcal{V}_{h,0}$:

$$\mathbb{B}_0 = \left\{\xi^{l,p,0} : l = 1, \ldots, n_\Omega, p = 1, \ldots, n_x\right\}.$$

- $\mathbb{B}_1$ forms the basis of $\mathcal{V}_{h,1}$:

$$\mathbb{B}_1 = \left\{\xi^{l,p,r} : l = 1, \ldots, n_\Omega, p = 1, \ldots, n_x, r = 1, \ldots, n_P - 1\right\}.$$

- $\mathbb{B}_1^{\mathrm{lm}}$ forms the basis of $\overline{\mathcal{V}}_{h,1}$:

$$\mathbb{B}_1^{\mathrm{lm}} = \left\{\eta^{p,r} : \eta_j^{p,r} = b^{p,r}, j = 1, \ldots, n_\Omega, p = 1, \ldots, n_x, r = 1, \ldots, n_P - 1\right\}.$$

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 09/01/20 to 138.195.214.230. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

- $\mathbb{B}^{\text{lm}}$ forms the basis of $\mathcal{V}_h^{\text{lm}}$. Since $\mathcal{V}_h^{\text{lm}} = \mathcal{V}_{h,0} + \overline{\mathcal{V}}_{h,1}$, we can take
$$\mathbb{B}^{\text{lm}} = \mathbb{B}_0 \cup \mathbb{B}_1^{\text{lm}}.$$
The dimension of $\mathcal{V}_h^{\text{lm}}$ is then $n_\Omega \cdot n_x + n_x \cdot (n_P - 1)$.

Note there exists a mapping from $\mathbb{B}_1$ to $\mathbb{B}_1^{\text{lm}}$. Since $\eta_j^{p,r} = b^{p,r} = \sum_{l=1}^{n_\Omega} \delta_{lj} b^{p,r} = \sum_{l=1}^{n_\Omega} \xi_j^{l,p,r}$, it can be seen that

$$\eta^{p,r} = \sum_{l=1}^{n_\Omega} \xi^{l,p,r} = \sum_{l',p',r'=1}^{n_\Omega} \Sigma^{(p,r),(l',p',r')} \xi^{l',p',r'}, \tag{3.23}$$

where $\mathbf{\Sigma} = (\Sigma^{(p,r),(l',p',r')})$ is an $(n_x \cdot (n_P - 1)) \times (n_\Omega \cdot n_x \cdot (n_P - 1))$ matrix with components $\Sigma^{(p,r),(l',p',r')} = \delta_{pp'}\delta_{rr'}$. The matrix $\mathbf{\Sigma}$ corresponds to a summation operator that maps an angular flux to a scalar flux, while $\mathbf{\Sigma}^T$ copies the scalar flux to each angular direction.

Let the solution of the low-memory method be represented by $\mathbf{\Psi} = \left[\mathbf{\Psi}_0, \mathbf{\Sigma}^T \mathbf{\Phi}_1\right]^T$. Using the fact $\mathbf{P}_1 \mathbf{\Sigma}^T = \mathbf{I}_{n_x \cdot (n_P - 1)}$, one can show $\mathbf{\Psi}$ satisfies the equations

$$\mathbf{L}_{00}\mathbf{\Psi}_0 + \mathbf{L}_{01}\mathbf{\Sigma}^T\mathbf{\Phi}_1 = \mathbf{M}_0\mathbf{P}_0\mathbf{\Psi}_0 + \mathbf{Q}_0, \tag{3.24a}$$

$$\mathbf{\Sigma}\mathbf{L}_{10}\mathbf{\Psi}_0 + \mathbf{\Sigma}\mathbf{L}_{11}\mathbf{\Sigma}^T\mathbf{\Phi}_1 = \mathbf{\Sigma}\mathbf{M}_1\mathbf{\Phi}_1 + \mathbf{\Sigma}\mathbf{Q}_1. \tag{3.24b}$$

As that in the original S_N-DG method, the system dimension of (3.24) can be reduced with the following procedure:

1. Solve for $\mathbf{\Phi}_1$ in terms of $\mathbf{\Psi}_0$ through (3.24b):
$$\mathbf{\Phi}_1 = \mathbf{B}_{11}^{-1}\mathbf{\Sigma}\left(-\mathbf{L}_{10}\mathbf{\Psi}_0 + \mathbf{Q}_1\right), \qquad \mathbf{B}_{11} = \mathbf{\Sigma}\mathbf{L}_{11}\mathbf{\Sigma}^T - \mathbf{\Sigma}\mathbf{M}_1. \tag{3.25}$$
2. Substitute $\mathbf{\Phi}_1$ from (3.25) into (3.24a) to obtain a closed equation for $\mathbf{\Psi}_0$:
$$\mathbf{\Psi}_0 - \mathbf{L}_{00}^{-1}\mathbf{M}_0(\mathbf{P}_0\mathbf{\Psi}_0) - \mathbf{L}_{00}^{-1}\mathbf{L}_{01}\mathbf{\Sigma}^T(\mathbf{B}_{11}^{-1}\mathbf{\Sigma}\mathbf{L}_{10}\mathbf{\Psi}_0) = \mathbf{L}_{00}^{-1}(\mathbf{Q}_0 - \mathbf{L}_{01}\mathbf{\Sigma}^T\mathbf{B}_{11}^{-1}\mathbf{\Sigma}\mathbf{Q}_1). \tag{3.26}$$
3. Apply $\mathbf{P}_0$ and $\mathbf{\Sigma}\mathbf{L}_{10}$ to (3.26) to obtain a closed system for $\mathbf{X}_0 = \mathbf{P}_0\mathbf{\Psi}_0$ and $\mathbf{X}_1 = \mathbf{B}_{11}^{-1}\mathbf{\Sigma}\mathbf{L}_{10}\mathbf{\Psi}_0$:
$$\mathbf{K}\begin{bmatrix}\mathbf{X}_0\\ \mathbf{X}_1\end{bmatrix} = \begin{bmatrix}\mathbf{P}_0\\ \mathbf{\Sigma}\mathbf{L}_{10}\end{bmatrix}\mathbf{L}_{00}^{-1}(\mathbf{Q}_0 - \mathbf{L}_{01}\mathbf{\Sigma}^T\mathbf{B}_{11}^{-1}\mathbf{\Sigma}\mathbf{Q}_1), \tag{3.27}$$
where
$$\mathbf{K} = \begin{bmatrix}\mathbf{I}_{n_x} - \mathbf{P}_0\mathbf{L}_{00}^{-1}\mathbf{M}_0 & -\mathbf{P}_0\mathbf{L}_{00}^{-1}\mathbf{L}_{01}\mathbf{\Sigma}^T\\ -\mathbf{\Sigma}\mathbf{L}_{10}\mathbf{L}_{00}^{-1}\mathbf{M}_0 & \mathbf{B}_{11} - \mathbf{\Sigma}\mathbf{L}_{10}\mathbf{L}_{00}^{-1}\mathbf{L}_{01}\mathbf{\Sigma}^T\end{bmatrix}. \tag{3.28}$$
4. Solve for $\mathbf{X}_0$ and $\mathbf{X}_1$ in (3.27). Then use (3.26) and (3.25) to obtain $\mathbf{\Psi}$:
$$\mathbf{\Psi}_0 = \mathbf{L}_{00}^{-1}\mathbf{M}_0\mathbf{X}_0 + \mathbf{L}_{00}^{-1}\mathbf{L}_{01}\mathbf{\Sigma}^T\mathbf{X}_1 + \mathbf{L}_{00}^{-1}(\mathbf{Q}_0 - \mathbf{L}_{01}\mathbf{\Sigma}^T\mathbf{B}_{11}^{-1}\mathbf{\Sigma}\mathbf{Q}_1), \tag{3.29a}$$
$$\mathbf{\Phi}_1 = \mathbf{B}_{11}^{-1}\mathbf{\Sigma}\left(-\mathbf{L}_{10}\mathbf{\Psi}_0 + \mathbf{Q}_1\right). \tag{3.29b}$$

Only step 4 above is needed to implement the algorithm. If one solves for $\mathbf{\Psi}_0$ directly from (3.26), then an $(n_\Omega \cdot n_x) \times (n_\Omega \cdot n_x)$ matrix should be inverted, while with (3.27), the matrix dimensions are reduced to $(n_x \cdot n_P) \times (n_x \cdot n_P)$. Typically n_P is much smaller than n_Ω.

We state the following theorems on the invertibility of $\mathbf{B}_{11}$ and $\mathbf{K}$, whose proofs can be found in Appendices C and D, respectively.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 09/01/20 to 138.195.214.230. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

THEOREM 3.3. $\mathbf{B}_{11}$ *is invertible. Furthermore, if the quadrature rule is central symmetric, then* $\mathbf{B}_{11}$ *is symmetric positive definite. Here, central symmetry means* Ω_j *and* $-\Omega_j$ *are both selected in the quadrature rule and their weights are equal, i.e.,* $\alpha_j = \alpha_{-j}$.

THEOREM 3.4. $\mathbf{K}$ *is invertible.*

Remark 3.1. Typically, the linear system in such a context is solved using the Krylov method, in which one needs to evaluate the multiplication of a vector with $\mathbf{K}$ in each iteration. We can use the following formula to avoid repeated evaluation in the left multiplication of $\mathbf{K}$:

$$\mathbf{K}\begin{bmatrix}\mathbf{X}_0\\ \mathbf{X}_1\end{bmatrix} = \begin{bmatrix}\mathbf{I}_{n_x} & \\ & \mathbf{B}_{11}\end{bmatrix}\begin{bmatrix}\mathbf{X}_0\\ \mathbf{X}_1\end{bmatrix} - \begin{bmatrix}\mathbf{P}_0\\ \mathbf{\Sigma}\mathbf{L}_{10}\end{bmatrix}\mathbf{L}_{00}^{-1}\left(\mathbf{M}_0\mathbf{X}_0 + \mathbf{L}_{01}\mathbf{\Sigma}^T\mathbf{X}_1\right). \tag{3.30}$$

Remark 3.2. As demonstrated in [18], the inversion of the block $\mathbf{L}_{00}$ in (3.27), rather than the full matrix $\mathbf{L}$ in (2.14), results in significant savings in terms of floating point operations (and hence time-to-solution). This savings will be partially offset by the need to invert the matrix $\mathbf{B}_{11}$ in (3.25). However, since the overall effect on time-to-solution depends heavily on the details of implementation, we do not investigate this aspect of the low-memory method in the numerical results but instead leave such an investigation to future work.

3.3. Reconstructed low-memory scheme. Because the low-memory scheme couples the angular components of $\mathcal{V}_{h,1}$, it is only first-order for fixed $\varepsilon > 0$. To recover second-order accuracy (formally), we introduce a spatial reconstruction procedure to approximate the anisotropic parts of $\mathcal{V}_{h,1}$.

3.3.1. Numerical scheme. We denote by Π_i the orthogonal projection from $\mathcal{V}_h$ to $\mathcal{V}_{h,i}$, $i = 0, 1$. The only information from the low-memory space $\mathcal{V}_h^{\mathrm{lm}}$ retains from $v \in \mathcal{V}_{h,1}$ is $\overline{\Pi_1(v)}$; the information contained in $\Pi_1(v) - \overline{\Pi_1(v)}$ is missing. We therefore introduce an operator $R_g^* v = R_g\Pi_0(v) - \overline{R_g\Pi_0(v)}$, where $R_g\Pi_0$ is an operator that returns the reconstructed slopes using piecewise constants and the boundary condition g, to rebuild the difference. Then the reconstructed scheme is written as follows: *find* $\psi_h \in \mathcal{V}_h^{\mathrm{lm}}$ *such that*

$$B(\psi_h + R_g^*\psi_h, v) = Q_g(v) \qquad \forall v \in \mathcal{V}_h^{\mathrm{lm}}. \tag{3.31}$$

The reconstruction $\psi_h + R_g^*\psi_h$ then gives a more accurate approximation to Ψ.

Equivalently, by assembling all boundary terms into the right-hand side, the reconstructed scheme can also be formulated as a Petrov–Galerkin method with trial function space

$$\mathcal{V}_h^{\mathrm{rlm}} = \{v + R_0^* v : v \in \mathcal{V}_h^{\mathrm{lm}}\}. \tag{3.32}$$

Since $R_0^* 0 = 0$, $\mathcal{V}_h^{\mathrm{rlm}}$ is in fact a linear space. With this formulation, the reconstructed method solves the following problem: *find* $\psi_{h,R_0} \in \mathcal{V}_h^{\mathrm{rlm}}$, *such that*

$$B(\psi_{h,R_0}, v) = \widetilde{Q}_g(v) \qquad \forall v \in \mathcal{V}_h^{\mathrm{lm}}. \tag{3.33}$$

The use of different trial and test function spaces makes the analysis of this scheme less transparent. Currently, we have no theoretical guarantee of unisolvency or the numerical diffusion limit. We observe, however, that the method recovers second-order convergence for several different test problems across a wide range of ε.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 09/01/20 to 138.195.214.230. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

In this paper, we apply the reconstruction suggested in [18] to recover slopes for simplicity, although in general other upwind approaches can also be used.[7] For illustration, we consider a uniform Cartesian mesh on $[0,1]\times[0,1]\times[0,1]$. The grid points are labeled from $\frac{1}{2}$ to $n+\frac{1}{2}$, respectively. We denote by $u^0_{i,j,k}$ the cell average of u on the cell $K_{i,j,k}$ that centers at (x_i, y_j, z_k). Along each direction $\Omega = (\Omega_x, \Omega_y, \Omega_z)$,

$$(R_g\Pi_0(u))|_{K_{i,j,k}} = (\delta_x^{s_x} u^0_{i,j,k})(x-x_i) + (\delta_y^{s_y} u^0_{i,j,k})(y-y_j) + (\delta_z^{s_z} u^0_{i,j,k})(z-z_k), \tag{3.34}$$

where $s_x = -\text{sign}(\Omega_x)$,

$$\delta_x^- u^0_{i,j,k} = \begin{cases} \dfrac{u^0_{i,j,k} - u^0_{i-1,j,k}}{h}, & 2 \le i \le n, \\ \dfrac{u^0_{1,j,k} - g(\Omega, (0, y_j, z_k))}{h/2}, & i = 1, \end{cases} \tag{3.35}$$

$$\delta_x^+ u^0_{i,j,k} = \begin{cases} \dfrac{u^0_{i+1,j,k} - u^0_{i,j,k}}{h}, & 1 \le i \le n-1, \\ \dfrac{g(\Omega, (1, y_j, z_k)) - u^0_{n,j,k}}{h/2}, & i = n. \end{cases} \tag{3.36}$$

$\delta_y^\pm$ and $\delta_z^\pm$ are defined similarly. For numerical results in the next section, we only reconstruct the P^1 slopes to recover the second-order accuracy; Q^1 type reconstruction gives similar results in terms of the convergence rate.

3.3.2. Implementation. Let $\mathbb{B}_0^{\text{rlm}} = \{\xi^{l,p,0} + R_0^*\xi^{l,p,0} : l = 1,\dots,n_\Omega, p = 1,\dots,n_x\}$ and $\mathcal{V}_h^{\text{rlm}} = \text{span}\{\mathbb{B}_0^{\text{rlm}}, \mathbb{B}_1^{\text{lm}}\}$. As in the first-order method, the total degrees of freedom is $n_\Omega \cdot n_x + n_x \cdot (n_P - 1)$. The boundary terms are assembled into a vector $\mathbf{r}_g$. Here we use $\mathbf{\Psi} = [\mathbf{\Psi}_0, \mathbf{\Psi}_1]^T$ to represent the solution of the reconstructed method, where $\mathbf{\Psi}_1 = \mathbf{\Sigma}^T\mathbf{\Phi}_1 + (\mathbf{I}_{n_\Omega\cdot n_x\cdot(n_P-1)} - \mathbf{\Sigma}^T\mathbf{P}_1)(\mathbf{R}\mathbf{\Psi}_0 + \mathbf{r}_g)$. Note $\mathbf{P}_1\mathbf{\Sigma}^T = \mathbf{I}_{n_x\cdot(n_P-1)}$, which implies $\mathbf{P}_1\mathbf{\Psi}_1 = \mathbf{\Phi}_1$. The block matrix form can then be written as follows:

$$\mathbf{L}_{00}\mathbf{\Psi}_0 + \mathbf{L}_{01}\mathbf{\Psi}_1 = \mathbf{M}_0\mathbf{P}_0\mathbf{\Psi}_0 + \mathbf{Q}_0, \tag{3.37a}$$

$$\mathbf{\Sigma}\mathbf{L}_{10}\mathbf{\Psi}_0 + \mathbf{\Sigma}\mathbf{L}_{11}\mathbf{\Psi}_1 = \mathbf{\Sigma}\mathbf{M}_1\mathbf{\Phi}_1 + \mathbf{\Sigma}\mathbf{Q}_1. \tag{3.37b}$$

With

$$\widetilde{\mathbf{L}}_{00} = \mathbf{L}_{00} + \mathbf{L}_{01}\mathbf{R}, \tag{3.38}$$

$$\widetilde{\mathbf{L}}_{10} = \mathbf{L}_{10} + \mathbf{L}_{11}(\mathbf{I}_{n_\Omega\cdot n_x\cdot(n_P-1)} - \mathbf{\Sigma}^T\mathbf{P}_1)\mathbf{R}, \tag{3.39}$$

$$\widetilde{\mathbf{Q}}_0 = \mathbf{Q}_0 - \mathbf{L}_{01}(\mathbf{I}_{n_\Omega\cdot n_x\cdot(n_P-1)} - \mathbf{\Sigma}^T\mathbf{P}_1)\mathbf{r}_g, \tag{3.40}$$

$$\widetilde{\mathbf{Q}}_1 = \mathbf{Q}_1 - \mathbf{L}_{11}(\mathbf{I}_{n_\Omega\cdot n_x\cdot(n_P-1)} - \mathbf{\Sigma}^T\mathbf{P}_1)\mathbf{r}_g, \tag{3.41}$$

one can rewrite (3.37) as

$$\widetilde{\mathbf{L}}_{00}\mathbf{\Psi}_0 + \mathbf{L}_{01}\mathbf{\Sigma}^T\left(\mathbf{\Phi}_1 - \mathbf{P}_1\mathbf{R}\mathbf{\Psi}_0\right) = \mathbf{M}_0\mathbf{P}_0\mathbf{\Psi}_0 + \widetilde{\mathbf{Q}}_0, \tag{3.42a}$$

$$\mathbf{\Sigma}\widetilde{\mathbf{L}}_{10}\mathbf{\Psi}_0 + \mathbf{\Sigma}\mathbf{L}_{11}\mathbf{\Sigma}^T\mathbf{\Phi}_1 = \mathbf{\Sigma}\mathbf{M}_1\mathbf{\Phi}_1 + \mathbf{\Sigma}\widetilde{\mathbf{Q}}_1. \tag{3.42b}$$

We follow the procedure as before to reduce the system dimension.

[7] For example, one can apply upwind reconstruction with wider stencils to improve the accuracy with an increased computational cost. Furthermore, the reconstruction can also be different at different spatial cells along different collocation angles, which may lead to an adaptive version of the reconstructed method. We postpone the discussion on numerical efficiency with different reconstruction methods to future work.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 09/01/20 to 138.195.214.230. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

1. Solve for $\mathbf{\Phi}_1$ in terms of $\mathbf{\Psi}_0$ through (3.42b):

$$\mathbf{\Phi}_1 = \mathbf{B}_{11}^{-1}\mathbf{\Sigma}\left(-\widetilde{\mathbf{L}}_{10}\mathbf{\Psi}_0 + \widetilde{\mathbf{Q}}_1\right), \qquad \mathbf{B}_{11} = \mathbf{\Sigma}\mathbf{L}_{11}\mathbf{\Sigma}^T - \mathbf{\Sigma}\mathbf{M}_1. \tag{3.43}$$

2. Substitute $\mathbf{\Phi}_1$ from (3.43) into (3.42a) to obtain a closed equation for $\mathbf{\Psi}_0$:

$$\begin{aligned}\mathbf{\Psi}_0 - \widetilde{\mathbf{L}}_{00}^{-1}\mathbf{M}_0(\mathbf{P}_0\mathbf{\Psi}_0) - \widetilde{\mathbf{L}}_{00}^{-1}\mathbf{L}_{01}\mathbf{\Sigma}^T(\mathbf{B}_{11}^{-1}\mathbf{\Sigma}\widetilde{\mathbf{L}}_{10} + \mathbf{P}_1\mathbf{R})\mathbf{\Psi}_0 \\ = \widetilde{\mathbf{L}}_{00}^{-1}(\widetilde{\mathbf{Q}}_0 - \mathbf{L}_{01}\mathbf{\Sigma}^T\mathbf{B}_{11}^{-1}\mathbf{\Sigma}\widetilde{\mathbf{Q}}_1).\end{aligned} \tag{3.44}$$

3. Applying $\mathbf{P}_0$ and $\mathbf{\Sigma}\widetilde{\mathbf{L}}_{10}$ to (3.44), to obtain a closed system for $\mathbf{X}_0 = \mathbf{P}_0\mathbf{\Psi}_0$ and $\mathbf{X}_1 = (\mathbf{B}_{11}^{-1}\mathbf{\Sigma}\widetilde{\mathbf{L}}_{10} + \mathbf{P}_1\mathbf{R})\mathbf{\Psi}_0$:

$$\widetilde{\mathbf{K}}\begin{bmatrix}\mathbf{X}_0\\ \mathbf{X}_1\end{bmatrix} = \begin{bmatrix}\mathbf{P}_0\\ \mathbf{\Sigma}\mathbf{L}_{10}\end{bmatrix}\widetilde{\mathbf{L}}_{00}^{-1}(\widetilde{\mathbf{Q}}_0 - \mathbf{L}_{01}\mathbf{\Sigma}^T\mathbf{B}_{11}^{-1}\mathbf{\Sigma}\widetilde{\mathbf{Q}}_1), \tag{3.45}$$

where

$$\widetilde{\mathbf{K}} = \begin{bmatrix}\mathbf{I}_{n_x} - \mathbf{P}_0\widetilde{\mathbf{L}}_{00}^{-1}\mathbf{M}_0 & -\mathbf{P}_0\widetilde{\mathbf{L}}_{00}^{-1}\mathbf{L}_{01}\mathbf{\Sigma}^T\\ -\mathbf{\Sigma}\widetilde{\mathbf{L}}_{10}\widetilde{\mathbf{L}}_{00}^{-1}\mathbf{M}_0 & \mathbf{B}_{11} - \mathbf{\Sigma}\widetilde{\mathbf{L}}_{10}\widetilde{\mathbf{L}}_{00}^{-1}\mathbf{L}_{01}\mathbf{\Sigma}^T\end{bmatrix}. \tag{3.46}$$

4. Solve for $\mathbf{X}_0$ and $\mathbf{X}_1$ in (3.45). Use (3.44) and (3.43) to recover $\mathbf{\Psi}$:

$$\mathbf{\Psi}_0 = \widetilde{\mathbf{L}}_{00}^{-1}\mathbf{L}_{01}\mathbf{X}_0 + \widetilde{\mathbf{L}}_{00}^{-1}\mathbf{M}_0\mathbf{X}_1 + \mathbf{L}_{00}^{-1}(\widetilde{\mathbf{Q}}_0 - \mathbf{L}_{01}\mathbf{\Sigma}^T\mathbf{B}_{11}^{-1}\mathbf{\Sigma}\widetilde{\mathbf{Q}}_1), \tag{3.47}$$

$$\mathbf{\Phi}_1 = \mathbf{B}_{11}^{-1}\mathbf{\Sigma}\left(-\widetilde{\mathbf{L}}_{10}\mathbf{\Psi}_0 + \widetilde{\mathbf{Q}}_1\right). \tag{3.48}$$

As the first-order method, only step 4 is used in the implementation. Since only upwind information is used, $\widetilde{\mathbf{L}}_{00}$ is invertible and can be inverted with sweeps along each angular direction. Note that $\mathbf{B}_{11}$ is invertible, as has been pointed out in Appendix C. One can follow the argument in Appendix D to show $\widetilde{\mathbf{K}}$ is invertible if the scheme (3.31) is unisolvent.

4. Numerical tests. In this section, we present numerical tests to examine the performance of the methods. While the preceding analysis is presented specifically in the full three-dimensional setting ($d = 3$), we restrict ourselves to simulations in reduced geometries for which $d = 1$ or $d = 2$. The analysis for these cases proceeds in the same way.

4.1. One-dimensional tests (slab geometry). In slab geometries, the radiative transport equation takes the form (see, e.g., [25, p. 28])

$$\mu\partial_x\Psi(\mu,x) + \left(\frac{\sigma_{\rm s}}{\varepsilon} + \varepsilon\sigma_{\rm a}\right)\Psi(\mu,x) = \frac{\sigma_{\rm s}}{2\varepsilon}\int_{-1}^{1}\Psi(\mu',x)d\mu' + \varepsilon q(x), \tag{4.1}$$

$$\Psi(\mu,x_a) = \Psi_l(\mu), \text{ if } \mu \geq 0, \quad \text{and} \quad \Psi(\mu,x_b) = \Psi_r(\mu), \text{ if } \mu < 0, \tag{4.2}$$

where $\mu \in [-1,1]$ and the spatial domain is $D_1 = [x_a, x_b]$. We will compare the S_N-P^0-DG scheme, S_N-P^1-DG scheme, low-memory scheme (LMDG), and reconstructed scheme (RLMDG). Numerical error is evaluated in L^1 norm.

Example 4.1. We first examine convergence rates of the methods using fabricated solutions. Let $\varepsilon = 1$, $\sigma_{\rm s} = 1$, $\sigma_{\rm a} = 1$, and $D_1 = [0,1]$. Assuming the exact solution Ψ, we compute the source term q and the inflow boundary conditions Ψ_l and Ψ_r

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 09/01/20 to 138.195.214.230. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

TABLE 4.1
Accuracy test for Example 4.1.

		P^0-DG		P^1-DG		LMDG		RLMDG	
Ψ	h	Error	Order	Error	Order	Error	Order	Error	Order
$\cos x$	1/20	3.79e-3	-	2.48e-5	-	2.24e-5	-	9.14e-5	-
	1/40	1.91e-3	0.99	6.27e-6	1.98	5.62e-6	2.00	2.31e-5	1.99
	1/80	9.55e-4	1.00	1.58e-6	1.99	1.41e-6	2.00	5.80e-6	1.99
	1/160	4.78e-4	1.00	3.96e-7	2.00	3.52e-7	2.00	1.45e-6	2.00
$\cos(x+\mu)$	1/20	4.70e-3	-	2.11e-5	-	3.20e-3	-	7.74e-5	-
	1/40	2.36e-3	0.99	5.36e-6	1.98	1.60e-3	1.00	1.95e-5	1.99
	1/80	1.18e-3	1.00	1.35e-6	1.99	8.01e-4	1.00	4.89e-6	1.99
	1/160	5.91e-4	1.00	3.39e-7	1.99	4.01e-4	1.00	1.23e-6	2.00

accordingly. With this approach, it may happen that q depends on μ. We use the 32 points Gauss quadrature on $[-1, 1]$ for S_N discretization.

We consider the case $\Psi = \cos x$ and $\Psi = \cos(x + \mu)$. The results are documented in Table 4.1. When Ψ is isotropic, the low-memory scheme exhibits second-order convergence. For the anisotropic case, the LMDG scheme degenerates to first-order accuracy, while the RLMDG scheme remains second-order accurate.

To better understand numerical efficiency, we analyze results in Table 4.1 by plotting L^1 error versus number of mesh cells, total degrees of freedom of the solution (memory costs), and number of equations in the reduced linear system (either (2.14), (3.27), or (3.45)).

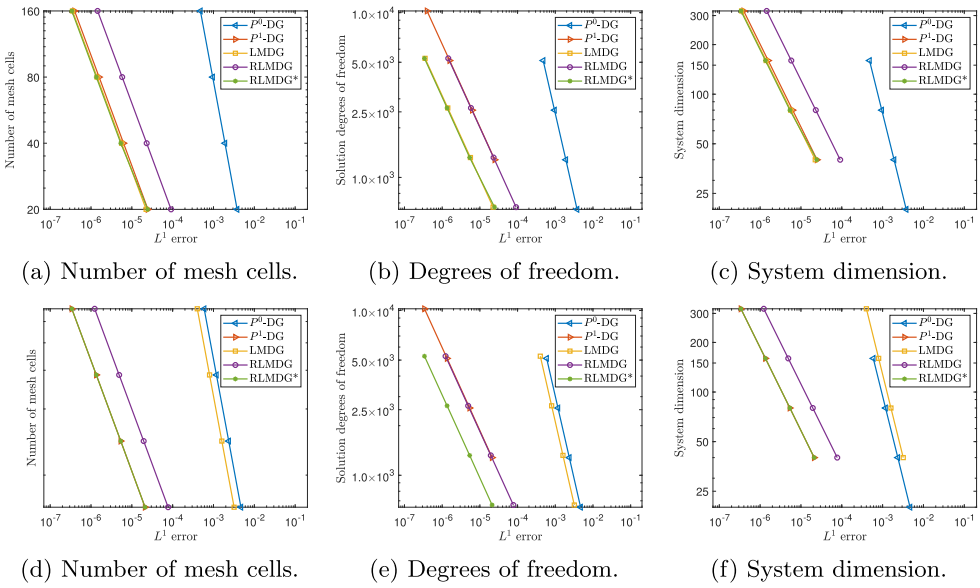
For the LMDG method, when the solution is isotropic, the method uses similar numbers of mesh cells as the P^1-DG method to reach the same accuracy. As a result, a reduced linear system of similar size is solved, but the degrees of freedom is smaller. For the anisotropic case, the LMDG method is first-order accurate. Compared with the P^0-DG method, it is able to reach similar accuracy on a coarser mesh. The reduced linear system is larger, but the number of degrees of freedom is indeed smaller.

For the RLMDG method, it seems to be less accurate compared with the P^1-DG method, and a finer mesh has to be used to achieve the same accuracy. As a result, the solution degrees of freedom is similar to that of the P^1-DG method but the reduced system is even larger. However, we point out that a more accurate reconstruction may solve this problem. For example, instead of using two cells, one can recover slopes in the interior region with a three-cell upwind reconstruction (which we call RLMDG*). This new method is still second-order accurate, but its error is comparable to the P^1-DG method and significantly smaller than the current RLMDG method. Efficiency results for RLMDG* are depicted by green lines in Figure 4.1 (they overlap with red lines in (d) and (f)). These results show that RLMDG* yields reduced systems of similar size to those of the P^1-DG method, but it uses less overall memory.

Example 4.2. In the second numerical test, we examine the convergence rate and asymptotic-preserving property of the methods. Let $\sigma_s = \sigma_a = 1$ in (4.1). The computational domain is set as $D_1 = [0, \pi]$. We take $\Psi_l = \Psi_r = 0$ and $q = \frac{4}{3}\sin(x)$. The 32-point Gauss quadrature is used for S_N discretization.

Numerical error at $\varepsilon = 10^{-5}$ and $\varepsilon = 1$ is listed in Table 4.2, respectively. The reference solutions are set as the numerical solutions with the P^1-DG scheme on a mesh with 1280 cells. One can see from Table 4.2 that the LMDG scheme exhibits a second-order convergence rate at $\varepsilon = 10^{-5}$, when the solution is almost isotropic, while it converges at a first-order rate when $\varepsilon = 1$. The RLMDG method is second-order in both cases.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.



(a) Number of mesh cells. (b) Degrees of freedom. (c) System dimension.

(d) Number of mesh cells. (e) Degrees of freedom. (f) System dimension.

FIG. 4.1. *Numerical efficiency in Example* 4.1. *The first row is for isotropic test* $\Psi = \cos(x)$ *and the second row is for anisotropic test* $\Psi = \cos(x + \mu)$.

TABLE 4.2
Accuracy test for Example 4.2.

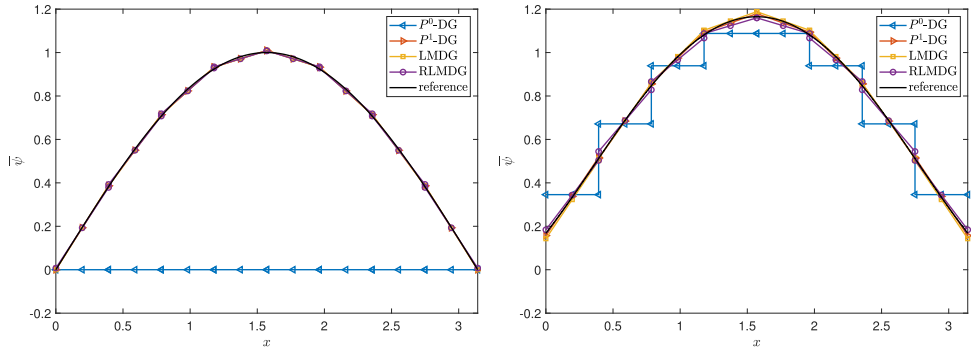
		P^0-DG		P^1-DG		LMDG		RLMDG	
ε	h	Error	Order	Error	Order	Error	Order	Error	Order
10^{-5}	1/20	2.00e-0	-	1.89e-3	-	1.89e-3	-	7.52e-3	-
	1/40	2.00e-0	0.00	4.70e-4	2.01	4.71e-4	2.00	1.88e-3	2.00
	1/80	2.00e-0	0.00	1.17e-4	2.01	1.16e-4	2.03	4.70e-4	2.00
	1/160	1.99e-0	0.00	2.91e-5	2.00	3.06e-5	1.92	1.17e-4	2.00
1	1/20	1.06e-1	-	2.91e-3	-	3.08e-2	-	9.55e-3	-
	1/40	5.38e-2	0.98	7.72e-4	1.92	1.59e-2	0.95	2.60e-3	1.88
	1/80	2.71e-2	0.99	1.99e-4	1.95	8.09e-3	0.98	6.90e-4	1.91
	1/160	1.35e-2	1.00	5.03e-5	1.99	4.08e-3	0.99	1.80e-4	1.94

Solution profiles of different schemes on a sparse uniform mesh, with $h = \pi/8$, are shown in Figure 4.2. When $\varepsilon = 10^{-5}$, both LMDG and RLMDG methods preserve the correct diffusion limit, unlike the P^0-DG method. When $\varepsilon = 1$, all schemes give valid approximations.

Example 4.3. We then consider a test from [28] with discontinuous cross-sections. The problem is defined on $[0, 1]$ and is purely scattering, i.e., $\sigma_a \equiv 0$. The cross-section is $\sigma_s = \sigma_{s,1} = 100$ on the left part of the domain $[0, 0.5]$ and is $\sigma_s = \sigma_{s,2} = 100, 1000,$ or 10000 on the right part $[0.5, 1]$. The source term is constant $q = 0.01$. In the numerical test, we set the mesh size to be $h = 0.1$ and $h = 0.02$, and solutions are depicted in Figures 4.3 and 4.4, respectively. As one can see, unlike the P^0-DG scheme, both LMDG and RLMDG schemes provide correct solution profiles. Since the problem is diffusive, the LMDG scheme gives accurate approximations that are almost indistinguishable from the P^1-DG solutions. The reconstructed scheme has

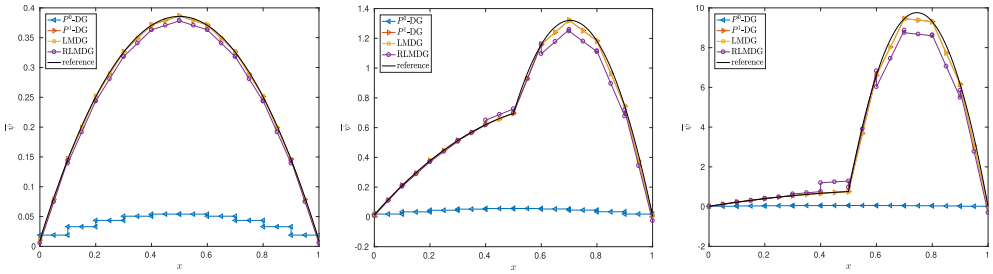
Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 09/01/20 to 138.195.214.230. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php



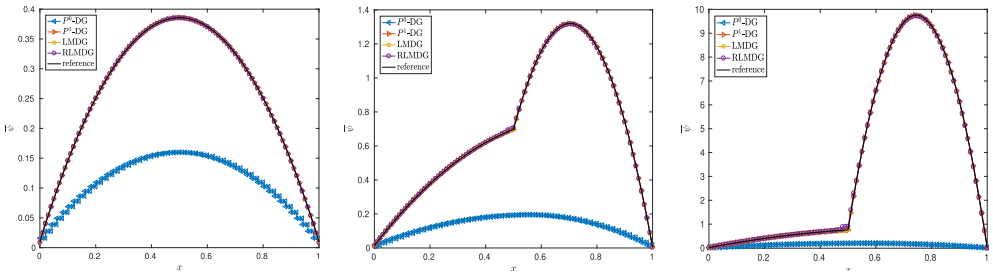
(a) $\varepsilon = 10^{-5}$. (b) $\varepsilon = 1$.

FIG. 4.2. *Profiles of numerical scalar fluxes in Example* 4.2.



(a) $\sigma_{s,2} = 100$. (b) $\sigma_{s,2} = 1000$. (c) $\sigma_{s,2} = 10000$.

FIG. 4.3. *Profiles of numerical scalar fluxes in Example* 4.3, $h = 0.1$.



(a) $\sigma_{s,2} = 100$. (b) $\sigma_{s,2} = 1000$. (c) $\sigma_{s,2} = 10000$.

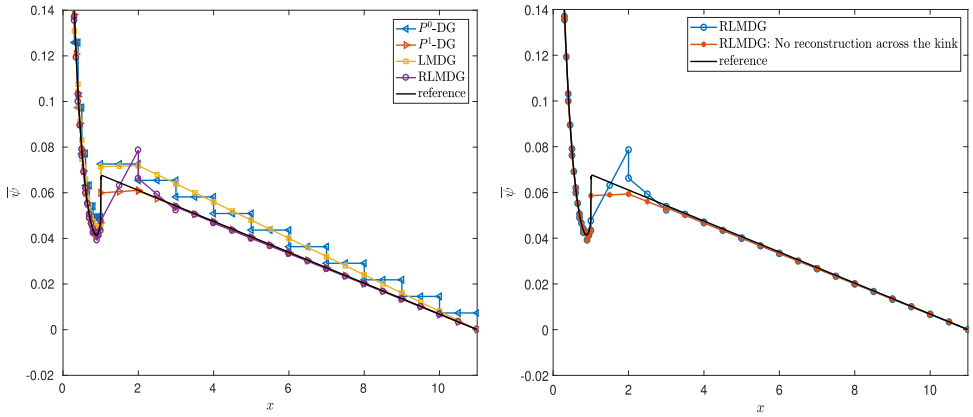
FIG. 4.4. *Profiles of numerical scalar fluxes in Example* 4.3, $h = 0.02$.

difficulty resolving the kink at $x = 0.5$, likely because the reconstruction is no longer accurate at this point. This artifact can indeed be alleviated as the mesh is refined comparing Figures 4.3 and 4.4.

Example 4.4. In this numerical test, we solve a test problem from [21] with discontinuous cross-sections. We take $q = 0$ with the left inflow $\Psi_l = 1$ at $x_a = 0$ and $\Psi_r = 0$ at $x_b = 11$. Let $\frac{\sigma_s}{\varepsilon} = \begin{cases} 0, & 0<x<1 \\ 100, & 1<x<11 \end{cases}$ and $\varepsilon\sigma_a = \begin{cases} 2, & 0<x<1 \\ 0, & 1<x<11 \end{cases}$. The 16-point

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 09/01/20 to 138.195.214.230. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php



(a) Numerical scalar fluxes. (b) With suppressed reconstruction.

FIG. 4.5. *Profiles of numerical scalar fluxes in Example* 4.4.

Gauss quadrature rule is used for angular discretization. The spatial mesh is set as $h = \begin{cases} 0.1, & 0<x<1 \\ 1, & 1<x<11 \end{cases}$.

Profiles of the scalar flux obtained with various schemes are depicted in Figure 4.5(a). The reference solutions are obtained with the P^1-DG scheme on a refined mesh. The solution of the LMDG scheme is satisfactory. As before, the RLMDG scheme gives an accurate approximation to the scalar flux, except for kinks near the discontinuity. However, this numerical artifact can also be alleviated by suppressing the reconstruction across the discontinuity; see Figure 4.5(b).

Example 4.5. This test is also from [21], with $D_1 = [0, 20]$ and $\Psi_l = \Psi_r = 0$. The cross-sections are $\frac{\sigma_s}{\varepsilon} = \begin{cases} 90, & 0<x<10 \\ 100, & 10<x<20 \end{cases}$ and $\varepsilon\sigma_a = \begin{cases} 10, & 0<x<10 \\ 0, & 10<x<20 \end{cases}$. We solve the problem using the 16-point Gauss quadrature rule and the spatial mesh is uniform with $h = 1$. For this numerical test, the system has smaller changes among different directions. Both the LMDG and RLMDG schemes give accurate approximations. Solution profiles are given in Figure 4.6.

4.2. Two-dimensional tests. In this section, we consider (1.1) with reduced spatial dimension $d = 2$. The problem is set up in the same form as (1.1), with the assumption that $\Psi = \Psi((\Omega_x, \Omega_y, \Omega_z), (x, y))$ is independent of the third spatial direction. The simulation can then be performed on the xy-domain D_2. Cartesian meshes are used for domain partitions.

Example 4.6. We set $\varepsilon = 1$ and $\sigma_s = \sigma_a = 1$ and test the accuracy with exact solutions $\Psi = \sin(x + y)$ and $\Psi = (\Omega_x - 3\Omega_y)^2 \sin(2x + y)$. As can be seen from Table 4.3, for $\Psi = \sin(x + y)$, both LMDG and RLMDG schemes are second-order accurate, while for the anisotropic problem with $\Psi = (\Omega_x - 3\Omega_y)^2 \sin(2x + y)$, the RLMDG scheme is still second-order accurate and the LMDG scheme is first-order accurate.

Example 4.7. To examine the asymptotic-preserving property, we consider the problem on $D_2 = [-1, 1] \times [-1, 1]$ with zero inflow boundary conditions. Let $\sigma_s = \sigma_a = 1$. We assume $q = (\frac{\pi^2}{6} + 1)\cos(\frac{\pi}{2}x)\cos(\frac{\pi}{2}y)$. The asymptotic solution is $\Psi^{(0)} =$

Downloaded 09/01/20 to 138.195.214.230. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

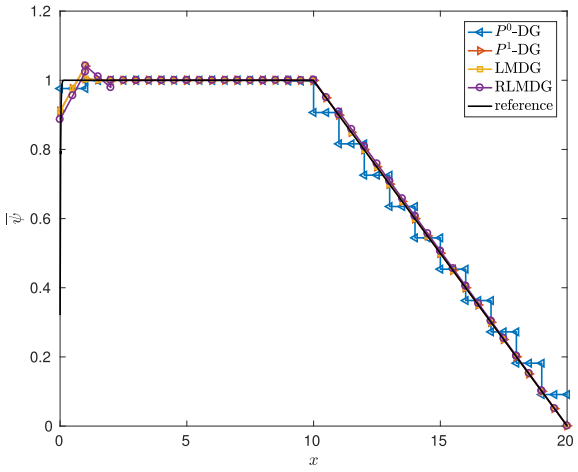


FIG. 4.6. *Profiles of numerical scalar fluxes in Example* 4.5.

TABLE 4.3
Two-dimensional accuracy test with fabricated solutions.

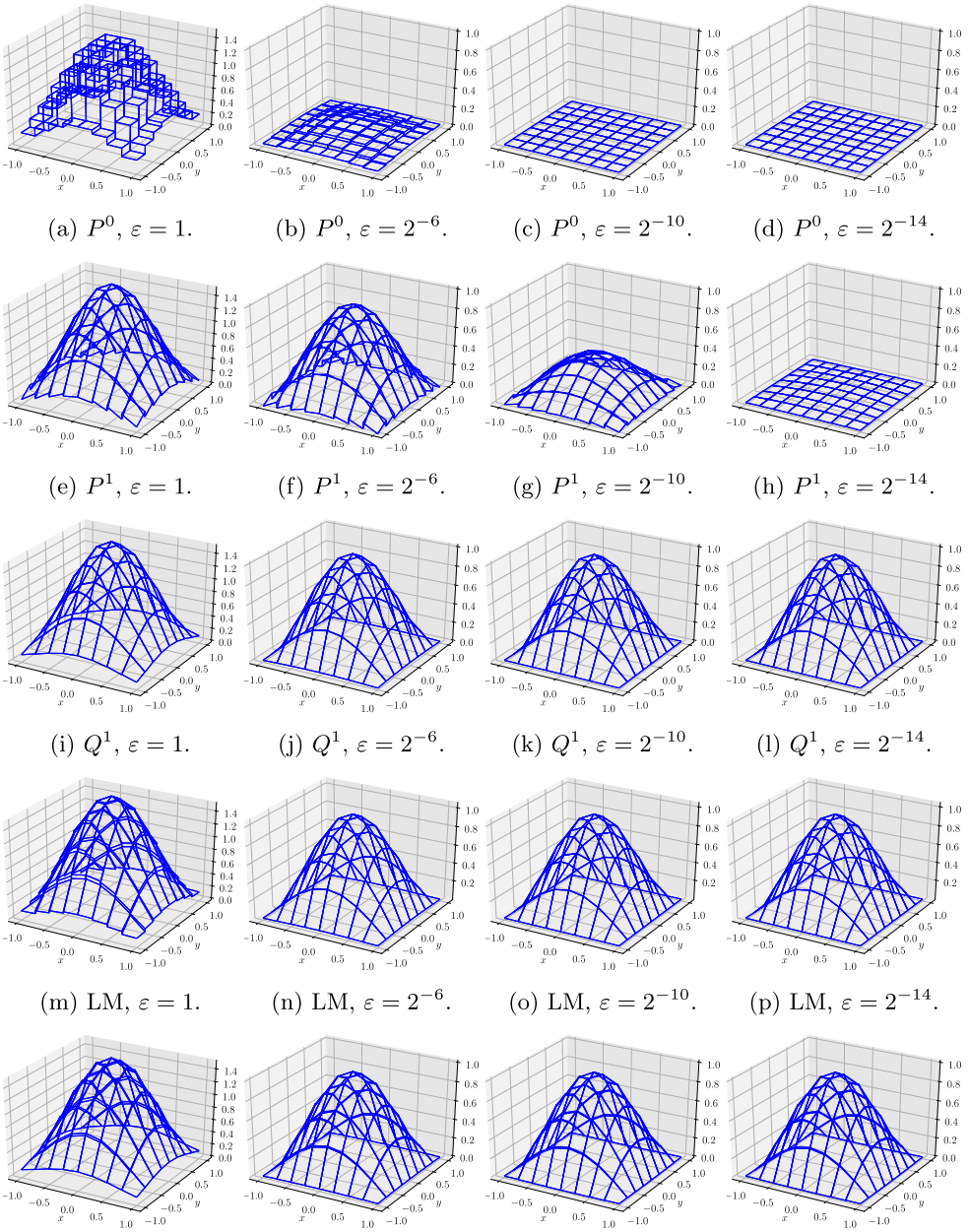
$\Psi = \sin(x+y)$										
	P^0-DG		P^1-DG		Q^1-DG		LMDG		RLMDG	
$h/\sqrt{2}$	Error	Order	Error	Order	Error	Order	Error	Order	Error	Order
1/20	2.04e-2	-	1.45e-4	-	1.40e-4	-	1.24e-4	-	4.59e-4	-
1/40	1.10e-2	0.89	3.42e-5	2.08	3.53e-5	1.98	3.12e-5	1.99	1.18e-4	1.96
1/80	5.77e-3	0.94	8.28e-6	2.04	8.88e-6	1.99	7.82e-6	2.00	2.98e-5	1.98
1/160	2.96e-3	0.96	2.04e-6	2.02	2.26e-6	2.00	1.96e-6	2.00	7.51e-6	1.99
$\Psi = (\Omega_x - 2\Omega_y)^2 \sin(2x+y)$										
	P^0-DG		P^1-DG		Q^1-DG		LMDG		RLMDG	
$h/\sqrt{2}$	Error	Order	Error	Order	Error	Order	Error	Order	Error	Order
1/20	7.84e-2		1.64e-3	-	1.39e-3	-	5.04e-2	-	4.81e-3	-
1/40	4.18e-2	0.91	4.12e-4	2.00	3.53e-4	1.98	2.57e-2	0.97	1.21e-3	1.99
1/80	2.12e-2	0.98	1.01e-4	2.03	8.87e-5	1.99	1.30e-2	0.99	3.05e-4	1.99
1/160	1.07e-2	0.97	2.52e-5	2.00	2.22e-5	2.00	6.51e-3	0.99	7.63e-5	2.00

$\cos(\frac{\pi}{2}x)\cos(\frac{\pi}{2}y)$. We test with $\varepsilon = 1, 2^{-6}, 2^{-10}, 2^{-14}$; the numerical results are given in Figure 4.7. For the P^0-DG and P^1-DG schemes, solutions become zero near the diffusion limit, while for the Q^1-DG scheme, LMDG scheme, and RLMDG scheme, the correct asymptotic profile is maintained.

5. Conclusions and future work. In this paper, we study a class of low-memory S_N-DG methods for the radiative transport equation. In our first method, we use the variational form of the original S_N-DG scheme with a smaller finite element space, in which functions have isotropic slopes. This method preserves the asymptotic diffusion limit and can still be solved with sweeps. It is first-order accurate and exhibits a second-order convergence rate near the diffusion limit. The second method is a correction of the first method with reconstructed slopes, which also preserves the diffusion limit and is second-order accurate in general settings (numerically). A summary of different methods and their properties can be found in Table 5.1.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 09/01/20 to 138.195.214.230. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php



(a) P^0, $\varepsilon = 1$. (b) P^0, $\varepsilon = 2^{-6}$. (c) P^0, $\varepsilon = 2^{-10}$. (d) P^0, $\varepsilon = 2^{-14}$.

(e) P^1, $\varepsilon = 1$. (f) P^1, $\varepsilon = 2^{-6}$. (g) P^1, $\varepsilon = 2^{-10}$. (h) P^1, $\varepsilon = 2^{-14}$.

(i) Q^1, $\varepsilon = 1$. (j) Q^1, $\varepsilon = 2^{-6}$. (k) Q^1, $\varepsilon = 2^{-10}$. (l) Q^1, $\varepsilon = 2^{-14}$.

(m) LM, $\varepsilon = 1$. (n) LM, $\varepsilon = 2^{-6}$. (o) LM, $\varepsilon = 2^{-10}$. (p) LM, $\varepsilon = 2^{-14}$.

(q) RLM, $\varepsilon = 1$. (r) RLM, $\varepsilon = 2^{-6}$. (s) RLM, $\varepsilon = 2^{-10}$. (t) RLM, $\varepsilon = 2^{-14}$.

FIG. 4.7. *Profiles of numerical scalar fluxes* $\overline{\psi}$ *in Example* 4.7.

Future work will focus on the efficiency boost of the low-memory methods. Possible directions include (i) further reducing degrees of freedom by enriching piecewise constant space only with continuous linear elements; (ii) developing preconditioners for linear systems; and (iii) comparing numerical efficiency of the methods with different reconstruction approaches, including adaptivity.

Downloaded 09/01/20 to 138.195.214.230. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

TABLE 5.1
Comparison of different methods.

		P^0-DG	P^1-DG	Q^1-DG	LMDG	RLMDG
Unisolvency when $\sigma_a \geq \delta_a > 0$		Yes				Unknown. Numerically: Yes
Preserves interior diffusion limit	1D	No	Yes			
	2D		Triangular: Yes Rectangular: No	Yes		
Order of accuracy	isotropic	1	2		2	2
	anisotropic				1	
System dimension	1D	n_x	$2n_x$			
	2D		$3n_x$	$4n_x$		
	3D		$4n_x$	$8n_x$		
Solution dimension	1D	$n_\Omega \cdot n_x$	$2n_\Omega \cdot n_x$		$n_\Omega \cdot n_x + n_x$	
	2D		$3n_\Omega \cdot n_x$	$4n_\Omega \cdot n_x$	$n_\Omega \cdot n_x + 3n_x$	
	3D		$4n_\Omega \cdot n_x$	$8n_\Omega \cdot n_x$	$n_\Omega \cdot n_x + 7n_x$	

Appendix A. Assembly of the matrices. From the variational form (2.12), we can derive a matrix system $\mathbf{L\Psi} = \mathbf{S\Psi} + \mathbf{Q}$. The matrices are defined as $\mathbf{L} = [L^{(l,p,r),(l',p',r')}]_{(n_\Omega \cdot n_x \cdot n_P)\times(n_\Omega \cdot n_x \cdot n_P)}$, $\mathbf{S} = [S^{(l,p,r),(l',p',r')}]_{(n_\Omega \cdot n_x \cdot n_P)\times(n_\Omega \cdot n_x \cdot n_P)}$, and $\mathbf{Q} = [Q^{(l,p,r)}]_{(n_\Omega \cdot n_x \cdot n_P)}$, where

$$L^{(l,p,r),(l',p',r')} = L\left(\xi^{l',p',r'}, \xi^{l,p,r}\right) = \delta_{ll'}\alpha_l \sum_{K\in\mathcal{T}_h}\left(-\int_K b^{p',r'}\Omega_l \cdot \nabla b^{p,r} dx + \int_{\partial K} \widehat{b}^{p',r'}\Omega_l \cdot \nu_K (b^{p,r})^{\text{int}} dx + \int_K \left(\frac{\sigma_s}{\varepsilon} + \varepsilon\sigma_a\right) b^{p',r'} b^{p,r} dx\right), \tag{A.1}$$

$$S^{(l,p,r),(l',p',r')} = S(\xi^{l',p',r'}, \xi^{l,p,r}) = \alpha_l\alpha_{l'} \sum_{K\in\mathcal{T}_h}\int_K \frac{\sigma_s}{\varepsilon} b^{p,r} b^{p',r'} dx, \tag{A.2}$$

$$Q^{(l,p,r)} = Q(\xi^{l,p,r}) = \alpha_l\left(\sum_{K\in\mathcal{T}_h}\int_K \varepsilon q b^{p,r} dx - \sum_{K\in\mathcal{T}_h}\int_{\partial K\cap F_h^\partial} g\Omega_l \cdot \nu_K b^{p,r} dx\right). \tag{A.3}$$

Note that $\mathbf{S}$ can be decomposed as the product of two matrices $\mathbf{S} = \mathbf{MP}$, where $\mathbf{M} = [M^{(l,p,r),(p'',r'')}]_{(n_\Omega \cdot n_x \cdot n_P)\times(n_x \cdot n_P)}$ with $M^{(l,p,r),(p'',r'')} = \alpha_l \sum_{K\in\mathcal{T}_h}\int_K \frac{\sigma_s}{\varepsilon} b^{p,r} b^{p'',r''} dx$, and $\mathbf{P} = [P^{(p'',r''),(l',p',r')}]_{(n_x \cdot n_P)\times(n_\Omega \cdot n_x \cdot n_P)}$ with $P^{(p'',r''),(l',p',r')} = \alpha_{l'}\delta_{p''p'}\delta_{r''r'}$. Hence the matrix equation becomes $\mathbf{L\Psi} = \mathbf{MP\Psi} + \mathbf{Q}$.

Appendix B. Proof of Theorem 2.1.

Proof. Since the variational problem of the S_N-DG method is unisolvent, $\mathbf{L} - \mathbf{MP}$ is invertible. To show $\mathbf{I}_{n_x \cdot n_P} - \mathbf{PL}^{-1}\mathbf{M}$ is invertible, one only needs to check

$$(\mathbf{I}_{n_x \cdot n_P} - \mathbf{PL}^{-1}\mathbf{M})\mathbf{X} = \mathbf{0} \Rightarrow \mathbf{X} = \mathbf{0}. \tag{B.1}$$

Indeed, with $(\mathbf{I}_{n_x \cdot n_P} - \mathbf{PL}^{-1}\mathbf{M})\mathbf{X} = \mathbf{0}$, we have

$$(\mathbf{L} - \mathbf{MP})(\mathbf{L}^{-1}\mathbf{MX}) = \mathbf{0} \Rightarrow \mathbf{L}^{-1}\mathbf{MX} = \mathbf{0} \Rightarrow \mathbf{MX} = \mathbf{0} \Rightarrow \mathbf{X} = \mathbf{0}. \tag{B.2}$$

Hence $\mathbf{I}_{n_x \cdot n_P} - \mathbf{PL}^{-1}\mathbf{M}$ is invertible. ∎

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 09/01/20 to 138.195.214.230. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Downloaded 09/01/20 to 138.195.214.230. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Appendix C. Proof of Theorem 3.3.

Proof. Note $\mathbf{B}_{11}$ corresponds to the variational problem (3.1) with $\mathcal{W}_h = \mathcal{V}_{h,1}$. Since the variational problem is unisolvent (even when $\sigma_a = 0$), $\mathbf{B}_{11}$ is invertible.

For all $u, v \in \overline{\mathcal{V}}_{h,1}$, since $u = \overline{u}$, we have

$$\sum_{j=1}^{n_\Omega} \alpha_j \int_K u_j \Omega_j \cdot \nabla v_j dx = \int_K \bar{u} \left(\sum_{j=1}^{n_\Omega} \alpha_j \Omega_j \right) \cdot \nabla \bar{v} dx = 0, \tag{C.1}$$

$$\sum_{j=1}^{n_\Omega} \sum_{K \in \mathcal{T}_h} \int_K \frac{\sigma_s}{\varepsilon} (u_j - \overline{u}) v_j dx = 0. \tag{C.2}$$

Therefore,

$$B_{11}(u,v) = \sum_{K \in \mathcal{T}_h} \int_K \varepsilon \sigma_a \overline{u}\ \overline{v} dx + \sum_{j=1}^{n_\Omega} \alpha_j \sum_{K \in \mathcal{T}_h} \int_{\partial K} \Omega_j \cdot \nu_K \widehat{u}_j v_j^{\text{int}} dx. \tag{C.3}$$

We would like to write the last term as a summation with respect to edges. ν_F^+ is defined as the unit normal of an edge F such that $e_1 \cdot \nu_F^+ > 0$. e_1 is the vector in $\mathbb{R}^3$, whose first component is 1 and others are 0. Suppose ν_F^+ is pointing from K^+ to K^-; we denote by $[v]_j = v_j|_{K^+} - v_j|_{K^-}$. Then

$$\begin{aligned} \sum_{j=1}^{n_\Omega} \alpha_j \sum_{K \in \mathcal{T}_h} \int_{\partial K} \Omega_j \cdot \nu \widehat{u}_j v_j dx &= \sum_{j=1}^{n_\Omega} \alpha_j \sum_{F \in \mathcal{F}_h} \int_F \Omega_j \cdot \nu_F^+ \widehat{u}_j [v]_j dx \\ &= \sum_{\substack{j=1,\dots,n_\Omega, \\ \Omega_j \cdot e_1 > 0}} \alpha_j \sum_{F \in \mathcal{F}_h} \int_F |\Omega_j \cdot \nu_F^+| [u]_j [v]_j dx. \end{aligned} \tag{C.4}$$

The last equality uses the central symmetry of the angular quadrature. Hence

$$B_{11}(u,v) = \sum_{K \in \mathcal{T}_h} \int_K \varepsilon \sigma_a \overline{u}\ \overline{v} d\Omega_N dx + \sum_{\substack{j=1,\dots,n_\Omega \\ \Omega_j \cdot e_1 > 0}} \alpha_j \sum_{F \in \mathcal{F}_h} \int_F |\Omega_j \cdot \nu_F^+| [u]_j [v]_j dx. \tag{C.5}$$

Here $\Omega_j \cdot e_1$ gives the first component of Ω_j. Since B_{11} is a symmetric and positive semidefinite bilinear form, $\mathbf{B}_{11}$ is then a symmetric and positive semidefinite matrix. The positive definiteness is implied by the fact $\mathbf{B}_{11}$ is invertible. ∎

Appendix D. Proof of Theorem 3.4. We first prove the following lemma.

LEMMA D.1. *Suppose* $\mathbf{I}_m - \mathbf{CA} - \mathbf{DB}$ *is invertible, where* $\mathbf{A}$ *is an* $n \times m$ *matrix,* $\mathbf{B}$ *is an* $n' \times m$ *matrix,* $\mathbf{C}$ *is an* $m \times n$ *matrix, and* $\mathbf{D}$ *is an* $m \times n'$ *matrix. Then* $\mathbf{I}_{n+n'} - \left[\begin{smallmatrix} \mathbf{A} \\ \mathbf{B} \end{smallmatrix}\right] \left[\begin{smallmatrix} \mathbf{C} & \mathbf{D} \end{smallmatrix}\right]$ *is invertible.*

Proof. It suffices to show that $(\mathbf{I}_{n+n'} - \left[\begin{smallmatrix} \mathbf{A} \\ \mathbf{B} \end{smallmatrix}\right] \left[\begin{smallmatrix} \mathbf{C} & \mathbf{D} \end{smallmatrix}\right]) \left[\begin{smallmatrix} \mathbf{X}_0 \\ \mathbf{X}_1 \end{smallmatrix}\right] = \mathbf{0}$ implies $\mathbf{X}_0$ and $\mathbf{X}_1$ are $\mathbf{0}$. Indeed, with $\mathbf{X} = \mathbf{CX}_0 + \mathbf{DX}_1$, the equality gives

$$\mathbf{AX} = \mathbf{ACX}_0 + \mathbf{ADX}_1 = \mathbf{X}_0 \quad \text{and} \quad \mathbf{BX} = \mathbf{BCX}_0 + \mathbf{BDX}_1 = \mathbf{X}_1, \tag{D.1}$$

which implies $(\mathbf{I}_m - \mathbf{CA} - \mathbf{DB})\mathbf{X} = \mathbf{X} - \mathbf{CX}_0 - \mathbf{DX}_1 = \mathbf{0}$. Since $\mathbf{I}_m - \mathbf{CA} - \mathbf{DB}$ is invertible, $\mathbf{X} = \mathbf{0}$. Using (D.1) we have $\mathbf{X}_0 = \mathbf{AX} = \mathbf{0}$ and $\mathbf{X}_1 = \mathbf{BX} = \mathbf{0}$. ∎

Then we show $\mathbf{K}$ is invertible.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

Downloaded 09/01/20 to 138.195.214.230. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Proof. One can see from Theorem 3.1 that the variational problem (3.17) is uniquely solvable. Therefore the associated system (3.26) is also unisolvent and $\mathbf{I}_{n_\Omega \cdot n_x} - \mathbf{L}_{00}^{-1}\mathbf{M}_0\mathbf{P}_0 - \mathbf{L}_{00}^{-1}\mathbf{L}_{01}\mathbf{\Sigma}^T\mathbf{B}_{11}^{-1}\mathbf{\Sigma}\mathbf{L}_{10}$ is invertible. Taking $\mathbf{A} = \mathbf{P}_0$, $\mathbf{B} = \mathbf{\Sigma}\mathbf{L}_{10}$, $\mathbf{C} = \mathbf{L}_{00}^{-1}\mathbf{M}_0$, and $\mathbf{D} = \mathbf{L}_{00}^{-1}\mathbf{L}_{01}\mathbf{\Sigma}^T\mathbf{B}_{11}^{-1}$ in Lemma D.1, one can show that the matrix $\mathbf{I}_{n_x \cdot n_P} - \left[\begin{smallmatrix}\mathbf{P}_0 \\ \mathbf{\Sigma}\mathbf{L}_{10}\end{smallmatrix}\right]\left[\mathbf{L}_{00}^{-1}\mathbf{M}_0 \ \mathbf{L}_{00}^{-1}\mathbf{L}_{01}\mathbf{\Sigma}^T\mathbf{B}_{11}^{-1}\right]$ is invertible. Hence $\mathbf{K} = \left(\mathbf{I}_{n_x \cdot n_P} - \left[\begin{smallmatrix}\mathbf{P}_0 \\ \mathbf{\Sigma}\mathbf{L}_{10}\end{smallmatrix}\right]\left[\mathbf{L}_{00}^{-1}\mathbf{M}_0 \ \mathbf{L}_{00}^{-1}\mathbf{L}_{01}\mathbf{\Sigma}^T\mathbf{B}_{11}^{-1}\right]\right)\left[\begin{smallmatrix}\mathbf{I}_{n_x} \\ \mathbf{B}_{11}\end{smallmatrix}\right]$ is also invertible. ∎

Acknowledgments. Zheng Sun would like to thank Oak Ridge National Laboratory for hosting his NSF internship and to thank the staff, post-docs, interns, and other visitors at ORNL for their warm hospitality. Cory Hauck would like to thank Marv Adams, Rob Lowrie, Ryan McClarren, and Jim Morel for the many discussions that contributed to the ideas in this paper.

REFERENCES

[1] M. L. Adams, *Discontinuous finite element transport solutions in thick diffusive problems*, Nuclear Sci. Engrg., 137 (2001), pp. 298–333.
[2] V. Agoshkov, *Boundary Value Problems for Transport Equations*, Springer, New York, 2012.
[3] D. N. Arnold, F. Brezzi, B. Cockburn, and L. D. Marini, *Unified analysis of discontinuous Galerkin methods for elliptic problems*, SIAM J. Numer. Anal., 39 (2002), pp. 1749–1779.
[4] I. Babuška and M. Suri, *On locking and robustness in the finite element method*, SIAM J. Numer. Anal., 29 (1992), pp. 1261–1293.
[5] C. Bardos, R. Santos, and R. Sentis, *Diffusion approximation and computation of the critical size*, Trans. Amer. Math. Soc., 284 (1984), pp. 617–649.
[6] A. Bensoussan, J. L. Lions, and G. C. Papanicolaou, *Boundary layers and homogenization of transport processes*, Publ. Res. Inst. Math. Sci., 15 (1979), pp. 53–157.
[7] B. Cockburn and C.-W. Shu, *Runge–Kutta discontinuous Galerkin methods for convection-dominated problems*, J. Sci. Comput., 16 (2001), pp. 173–261.
[8] R. Dautray and J. L. Lions, *Mathematical Analysis and Numerical Methods for Science and Technology, Volume* 6*: Evolution Problems* II, Spinger-Verlag, Berlin, 2000.
[9] B. Davison, *Neutron Transport Theory*, Oxford University Press, London, 1973.
[10] H. Egger and M. Schlottbom, *A mixed variational framework for the radiative transfer equation*, Math. Models Methods Appl. Sci., 22 (2012), 1150014.
[11] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 3rd ed., Johns Hopkins University Press, Baltimore, 2012.
[12] F. Graziani, *Computational Methods in Transport*, Lect. Notes Comput. Sci. Eng. 48, Springer, New York, 2006.
[13] J.-L. Guermond and G. Kanschat, *Asymptotic analysis of upwind discontinuous Galerkin approximation of the radiative transport equation in the diffusive limit*, SIAM J. Numer. Anal., 48 (2010), pp. 53–78.
[14] J.-L. Guermond, G. Kanschat, and J. C. Ragusa, *Discontinuous Galerkin for the radiative transport equation*, in Recent Developments in Discontinuous Galerkin Finite Element Methods for Partial Differential Equations, Springer, New York, 2014, pp. 181–193.
[15] J.-L. Guermond, B. Popov, and J. Ragusa, *Positive asymptotic preserving approximation of the radiation transport equation*, SIAM J. Numer. Anal., 58 (2020), pp. 519–540.
[16] G. J. Habetler and B. J. Matkowsky, *Uniform asymptotic expansions in transport theory with small mean free paths, and the diffusion approximation*, J. Math. Phys., 16 (1975), pp. 846–854.
[17] W. Han, J. Huang, and J. Eichholz, *Discrete-ordinate discontinuous Galerkin methods for solving the radiative transfer equation*, SIAM J. Sci. Comput., 32 (2010), pp. 477–497.
[18] V. Heningburg and C. D. Hauck, *Hybrid Solver for the Radiative Transport Equation Using Finite Volume and Discontinuous Galerkin*, arXiv:2002.02517, 2020.
[19] S. Jin, *Efficient asymptotic-preserving (AP) schemes for some multiscale kinetic equations*, SIAM J. Sci. Comput., 21 (1999), pp. 441–454.
[20] S. Jin and C. D. Levermore, *Numerical schemes for hyperbolic conservation laws with stiff relaxation terms*, J. Comput. Phys., 126 (1996), pp. 449–467.
[21] E. Larsen and J. Morel, *Asymptotic solutions of numerical transport problems in optically thick, diffusive regimes* II, J. Comput. Phys., 83 (1989), pp. 212–236.
[22] E. W. Larsen, J. Morel, and W. F. Miller, *Asymptotic solutions of numerical transport problems in optically thick, diffusive regimes*, J. Comput. Phys., 69 (1987), pp. 283–324.

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.

[23] E. W. Larsen and J. E. Morel, *Advances in discrete-ordinates methodology*, in Nuclear Computational Science, Springer, New York, 2010, pp. 1–84.
[24] P. Lesaint and P. A. Raviart, *On a finite element method for solving the neutron transport equation*, in Mathematical Aspects of Finite Elements in Partial Differential Equations, Proceedings of a Symposium Conducted by the Mathematics Research Center, University of Wisconsin-Madison, Madison, WI, 1974, pp. 1–3.
[25] E. E. Lewis and J. W. F. Miller, *Computational Methods in Neutron Transport*, John Wiley and Sons, New York, 1984.
[26] D. Mihalis and B. Weibel-Mihalis, *Foundations of Radiation Hydrodynamics*, Dover, Mineola, NY, 1999.
[27] G. C. Pomraning, *Radiation Hydrodynamics*, Pergamon Press, New York, 1973.
[28] J. C. Ragusa, J.-L. Guermond, and G. Kanschat, *A robust S_N-DG-approximation for radiation transport in optically thick and diffusive regimes*, J. Comput. Phys., 231 (2012), pp. 1947–1962.
[29] W. H. Reed and T. Hill, *Triangular Mesh Methods for the Neutron Transport Equation*, Tech. report, Los Alamos Scientific Laboratory, Los Alamos, NM, 1973.
[30] L. Wu and Y. Guo, *Geometric correction for diffusive expansion of steady neutron transport equation*, Commun. Math. Phys., 336 (2015), pp. 1473–1553.
[31] Y. Xu and C.-W. Shu, *Local discontinuous Galerkin methods for high-order time-dependent partial differential equations*, Commun. Comput. Phys., 7 (2010), pp. 1–46.
[32] J. Yan and C.-W. Shu, *Local discontinuous Galerkin methods for partial differential equations with higher order derivatives*, J. Sci. Comput., 17 (2002), pp. 27–47.

Downloaded 09/01/20 to 138.195.214.230. Redistribution subject to SIAM license or copyright; see http://www.siam.org/journals/ojsa.php

Copyright © by SIAM. Unauthorized reproduction of this article is prohibited.