A NON-NEGATIVE, NON-LINEAR, PETROV-GALERKIN METHOD FOR BILINEAR DISCONTINUOUS DIFFERENCING OF THE S_N EQUATIONS

Peter G. Maginot, Jean C. Ragusa[*]**, and Jim E. Morel**

Department of Nuclear Engineering

Texas A&M University

3133 TAMU, College Station, TX 77843

pmaginot@tamu.edu; jean.ragusa@tamu.edu; morel@tamu.edu

ABSTRACT

We have developed a new, non-negative, non-linear, Petrov-Galerkin bilinear discontinuous (BLD) finite element differencing of the 2-D Cartesian geometry S_N equations for quadrilaterals on an unstructured mesh. This work is an extension of a scheme we previously developed for use with linear discontinuous (LD) differencing of the 2-D S_N equations for rectangular mesh cells. We present the theory and equations that describe the new method. Additionally, we numerically compare the accuracy of our proposed method to the accuracy of BLD without lumping and the subcell corner balance method (equivalent to a "fully" lumped BLD scheme) for a test problem that causes BLD scheme to generate negative angular flux solutions.

Key Words: Radiation transport, DFEM, non-negative, bilinear, quadrilaterals

1   INTRODUCTION

Discontinuous finite element method (DFEM) spatial discretizations of the S_N neutron transport equation and S_N thermal radiative transfer equations can result in negative angular flux and negative angular intensity solutions. These negative solutions are non-physical, but inherent to the mathematics that define the radiation spatial differencing scheme. Several researchers have examined different methods (matrix lumping [1], fix-ups [2], and strictly non-negative solution representations [3]) that inhibit or eliminate the negativities of the linear discontinuous (LD) finite element scheme on a variety of spatial mesh cell types for the S_N neutron transport equation. However, Adams showed that LD does not maintain the neutronics thick diffusion limit on quadrilaterals [1]. We are interested in accurate methods for radiative transfer, therefore we seek methods that can maintain the equilibrium diffusion limit on quadrilaterals. If a radiative transfer spatial discretization is to maintain the equilibrium diffusion limit, its neutron transport analog must preserve the thick diffusion limit. As a first step towards accurate methods for radiative transfer on quadrilaterals, we seek a new, non-negative, finite element discretization that the maintains the bilinear spatial moments of the neutron transport equation on quadrilaterals.

The unlumped bilinear discontinuous (UBLD) spatial scheme yields negative angular flux solutions for quadrilateral cells with grazong radiation incidence and/or large optical thickness [1].

---

[*]Corresponding author

To our knowledge, only matrix lumping has been considered in an attempt to inhibit negative angular flux solutions of BLD spatial discretizations. While more common forms of matrix lumping, such as mass matrix or combination mass and surface matrix lumping, inhibit negative angular flux solutions, they do not guarantee a strictly non-negative BLD angular flux solution for source-free pure absorber problems [1]. Wareing, et al., derived the fully lumped BLD (FLBLD) scheme [4] that uses additional manipulations of the UBLD equations to yield angular flux solutions that are strictly non-negative for source-free pure absorber problems. Unfortunately, Adams demonstrated that FLBLD, which is equivalent to the subcell balance method on rectangles, is less accurate than unlumped BLD (UBLD) scheme for spatial mesh cells of thin and intermediate thicknesses [5].

In [3], we developed a non-negative, non-linear Petrov-Galerkin DFEM spatial differencing scheme that maintains the linear discontinuous spatial moments of the S_N neutron transport equations in slab and rectangular Cartesian geometry. In this paper, will extend the main idea of [3] to create a non-negative, non-linear Petrov-Galerkin DFEM scheme that will maintain the bilinear discontinuous spatial moments of the neutron transport equation on quadrilaterals. The remainder of this paper is divided as follows: a brief derivation common to all bilinear DFEM is given in Section 2, description and derivation of our new, bilinear consistent set-to-zero (BCSZ) Petrov-Galerkin DFEM scheme is given in Section 3, computational results demonstrating the strictly positive nature of BCSZ and its improved accuracy relative to UBLD and FLBLD are given in Section 4, and conclusions are given in Section 5.

2 MOMENT EQUATIONS

We begin by first considering the 2-D Cartesian discrete ordinates transport equation:

$$\vec{\Omega}_d \cdot \nabla \psi_d(x,y) + \sigma_t(x,y)\psi_d(x,y) = S_d(x,y)\,, \tag{1}$$

where $\vec{\Omega}_d$ is the neutron direction, $\psi_d(x,y)$ is the angular flux $[n/(cm^s\ sec\ ster)]$ in direction $\vec{\Omega}_d$, $\sigma_t(x,y)$ is the total interaction cross section $[cm^{-1}]$, and $S_d(x,y)$ is the total source (scattering + fixed sources) in direction $\vec{\Omega}_d$. Following the standard Galerkin procedure, we take the spatial moment of Eq. (1) with respect to basis function $B_i(x,y)$ by first multiplying by basis function $B_i(x,y)$ and integrating over spatial cell K. Assuming cell-wise constant σ_t, the $i-$th spatial moment is:

$$\int_K B_i(x,y)\left[\vec{\Omega}_d \cdot \nabla \psi_d(x,y) + \sigma_t \psi_d(x,y)\right]\,dxdy = \int_K B_i(x,y)S(x,y)\,dxdy\,. \tag{2}$$

Using integration by parts Eq. (2) becomes:

$$(\vec{\Omega}_d \cdot \vec{n}) \oint_K B_i(x,y)\psi_d(x,y)\,d\ell - \int_K \psi_d(x,y)\left[\vec{\Omega}_d \cdot (\nabla_{xy}B_i(x,y))\right]dxdy$$
$$+ \sigma_t \int_K B_i(x,y)\psi_d(x,y)\,dxdy\,. \tag{3}$$

Defining a reference element mapping as in Fig. 1: a reference point (s,t), $s \in [-1,1]$ $t \in [-1,1]$,
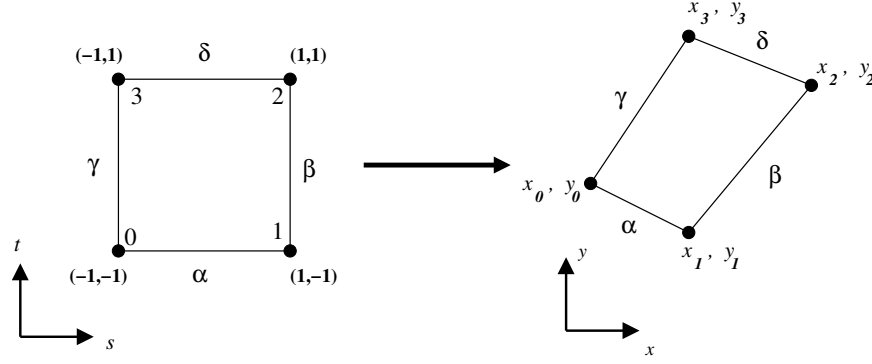
Figure 1. Reference element mapping.

is transformed to a physical point (x, y) such that:

$$x = x_0 B_0(s,t) + x_1 B_1(s,t) + x_2 B_2(s,t) + x_3 B_3(s,t) \tag{4}$$

$$y = y_0 B_0(s,t) + y_1 B_1(s,t) + y_2 B_2(s,t) + y_3 B_3(s,t) \, . \tag{5}$$

In this work, we use mapped, bilinear Lagrange interpolatory functions as basis functions:

$$B_0(s,t) = \frac{1-s}{2}\frac{1-t}{2} \tag{6a}$$

$$B_1(s,t) = \frac{s+1}{2}\frac{1-t}{2} \tag{6b}$$

$$B_2(s,t) = \frac{s+1}{2}\frac{t+1}{2} \tag{6c}$$

$$B_3(s,t) = \frac{1-s}{2}\frac{t+1}{2} \, . \tag{6d}$$

With this choice of basis functions, following the Galerkin procedure creates four equations with more unknowns than equations. To solve the moment equations, one must assume a solution representation, $\widetilde{\psi}_d(s,t)$, to approximate the true angular flux, $\psi_d(s,t)$.

The UBLD scheme assumes a solution trial space equal to the basis space,

$$\widetilde{\psi}_{UBLD} = \sum_{i=0}^{3} \psi_{i,UBLD} B_i(s,t) \, . \tag{7}$$

Under this assumption, the spatial moments of the transport equation become a 4×4 linear system of equations. Interested readers are directed to [1] or one of the many other papers that have derived and used UBLD on rectangles and quadrilaterals for a more complete derivation.

The FLBLD scheme [1,4], alternatively derived as the subcell balance method on quadrilaterals [5] begins with the UBLD equations then lumps (diagonalizes) the UBLD mass and surface matrices. The FLBLD equations are then further manipulated to result in a scheme that is yields strictly non-negative solutions for source-free pure absorber solutions. The FLBLD scheme is second order convergent in space, but can be less accurate than UBLD for cells of intermediate and thin optical thicknesses. Interested readers are directed to [1,4,5] for a more detailed derivation.

The BCSZ scheme is defined as being a bilinear function, $\widehat{\psi}_{BCSZ}(s,t)$,

$$\widehat{\psi}_{BCSZ}(s,t) = \sum_{i=0}^{3} \psi_{i,BCSZ} B_i(s,t), \tag{8}$$

everywhere $\widehat{\psi}_{BCSZ}$ is positive and zero otherwise:

$$\widetilde{\psi}_{BCSZ}(s,t) = \begin{cases} \widehat{\psi}_{BCSZ}(s,t) & \widehat{\psi}_{BCSZ}(s,t) > 0 \\ 0 & \text{otherwise} \end{cases}. \tag{9}$$

The initial iterate of $\widehat{\psi}_{BCSZ}$ is $\widetilde{\psi}_{UBLD}$. If $\widetilde{\psi}_{UBLD} \geq 0$ everywhere within a cell, $\widetilde{\psi}_{BCSZ} = \widetilde{\psi}_{UBLD}$. Using the definition of ψ_{BCSZ} given in Eq. (9) turns the four spatial moments of the transport equation into four non-linear equations with four fundamental unknowns, $\psi_{i,BCSZ}$ that describe the bilinear function $\widehat{\psi}_{BCSZ}$.

3   BCSZ SCHEME

For brevity, we omit a complete derivation of the four spatial moment equations. Rather, we choose to focus on defining the important characteristics of the BCSZ scheme, namely how the BCSZ scheme defines the following moments equation terms:

1. edge leakage, $\psi_{i,edge}$,

2. cell volume gradients, $\psi_{i,\mu}$ and $\psi_{i,\eta}$, and

3. cell volume moments, $\psi_{i,M}$.

The $\psi_{i,edge}$ terms come from the moment equation integration by parts:

$$(\vec{\Omega}_d \cdot \vec{n}) \oint_K B_i(x,y)\psi_d(x,y)\, d\ell = (\vec{\Omega}_d \cdot \vec{n}_\alpha)\frac{|J_\alpha|}{2}\int_{-1}^{1} B_i(s,-1)\psi_d(s,-1)\, ds$$

$$+ (\vec{\Omega}_d \cdot \vec{n}_\beta)\frac{|J_\beta|}{2}\int_{-1}^{1} B_i(1,t)\psi_d(1,t)\, dt + (\vec{\Omega}_d \cdot \vec{n}_\delta)\frac{|J_\delta|}{2}\int_{-1}^{1} B_i(s,1)\psi_d(s,1)\, ds$$

$$+ (\vec{\Omega}_d \cdot \vec{n}_\gamma)\frac{|J_\gamma|}{2}\int_{-1}^{1} B_i(-1,t)\psi_d(-1,t)\, dt. \tag{10a}$$

Defining an edge moment of the angular flux, $\psi_{i,edge}$, we have Eq. (10b)

$$(\vec{\Omega}_d\cdot\vec{n}) \oint_K B_i(x,y)\psi_d(x,y)\, d\ell = (\vec{\Omega}_d\cdot\vec{n}_\alpha)\frac{|J_\alpha|}{2}\psi_{i,\alpha}+(\vec{\Omega}_d\cdot\vec{n}_\beta)\frac{|J_\beta|}{2}\psi_{i,\beta}+(\vec{\Omega}_d\cdot\vec{n}_\delta)\frac{|J_\delta|}{2}\psi_{i,\delta}+(\vec{\Omega}_d\cdot\vec{n}_\gamma)\psi_{i,\gamma},$$

$$\tag{10b}$$

where $|J_z|$ is the length of side z, $\vec{n}_z$ is the outward directed unit normal of side z, and we have used edge short hands ($\alpha,\ \beta,\ \delta,\ \gamma$) as in Fig. 1. The cell volume gradients, $\psi_{i,\mu}$ and $\psi_{i,\eta}$ come from:

$$\int_K \psi_d(x,y) \left[\vec{\Omega}_d \cdot (\nabla_{xy} B_i(x,y)) \right] dxdy = \mu \int_{-1}^{1} \int_{-1}^{1} \psi_d(s,t) \left(\frac{\partial y}{\partial t} \frac{\partial B_i}{\partial s} - \frac{\partial y}{\partial s} \frac{\partial B_i}{\partial t} \right) dsdt$$

$$+ \eta \int_{-1}^{1} \int_{-1}^{1} \psi_d(s,t) \left(\frac{\partial x}{\partial s} \frac{\partial B_i}{\partial t} - \frac{\partial x}{\partial t} \frac{\partial B_i}{\partial s} \right) dsdt \quad \text{(11a)}$$

or in short,

$$\int_K \psi_d(x,y) \left[\vec{\Omega}_d \cdot (\nabla_{xy} B_i(x,y)) \right] dxdy = \mu \psi_{i,\mu} + \eta \psi_{i,\eta}, \quad \text{(11b)}$$

where $\vec{\Omega}_d = \langle \mu, \eta \rangle$. The $\psi_{i,M}$ term comes from the reaction term:

$$\sigma_t \int_K B_i(x,y)\psi_d(x,y) \; dxdy = \sigma_t \int_{-1}^{1} \int_{-1}^{1} B_i(s,t)\psi_d \left| \mathbf{J} \right| \; dsdt = \sigma_t \psi_{i,M}. \quad \text{(12)}$$

In Eq. (11), all $\frac{\partial}{\partial s}$ terms are linear functions in t, all $\frac{\partial}{\partial t}$ terms are linear functions in s, and $\mathbf{J}$ is the Jacobian matrix of the coordinate transformation:

$$\mathbf{J} = \begin{bmatrix} \frac{\partial x}{\partial s} & \frac{\partial y}{\partial s} \\ \frac{\partial x}{\partial t} & \frac{\partial y}{\partial t} \end{bmatrix}. \quad \text{(13)}$$

3.1 BCSZ Edge Integration

Along any edge, there are four possible cases. Assuming that we are looking at edge α, but with easy translation to any other edge:

1. $\widehat{\psi}_{BCSZ} > 0$ along whole edge,

2. $\widehat{\psi}_{BCSZ} < 0$ along whole edge,

3. $\psi_{0,BCSZ} < 0$, $\psi_{1,BCSZ} > 0$, and

4. $\psi_{0,BCSZ} > 0$, $\psi_{1,BCSZ} < 0$.

Case one is straightforward and identical to the integration used by the UBLD scheme. Case two is by definition, zero along the integral. Cases three and four are handled by first determining s_z, where $\widehat{\psi}_{BCSZ}(s_z, -1) = 0$:

$$s_z = \frac{\psi_{0,BCSZ} + \psi_{1,BCSZ}}{\psi_{0,BCSZ} - \psi_{1,BCSZ}} \quad \text{(14)}$$

Then a two point Gauss quadrature is mapped to the interval $[s_z, 1]$ for case 3 and $[-1, s_z]$ for case 4, and the integral of $\psi_{i,\alpha}$ is computed (exactly) using function evaluations at the mapped quadrature points.

3.2 BCSZ Cell Integration

Given the definition of $\widetilde{\psi}_{BCSZ}$ in Eq. (9), the integral contributions to $\psi_{i,\mu}$, $\psi_{i,\eta}$, and $\psi_{i,M}$ are non-trivial only over a portion of the cell where $\widehat{\psi}_{BCSZ}(s,t) > 0$. There are seven possible geometric integration cases to be considered for all values of $\widehat{\psi}_{BCSZ}$. Six are shown in Fig. 2. The other case is when $\widehat{\psi}_{BCSZ} > 0$ everywhere in the cell, in which case the integration of

(a) **One negative**

(b) **Two negative, same side**

(c) **Two negative, opposite sides,** $\psi_0\psi_2 > \psi_1\psi_3$

(d) **Three negative**

(e) **Two negative, block formation,** $\psi_1\psi_3 = \psi_2\psi_4$

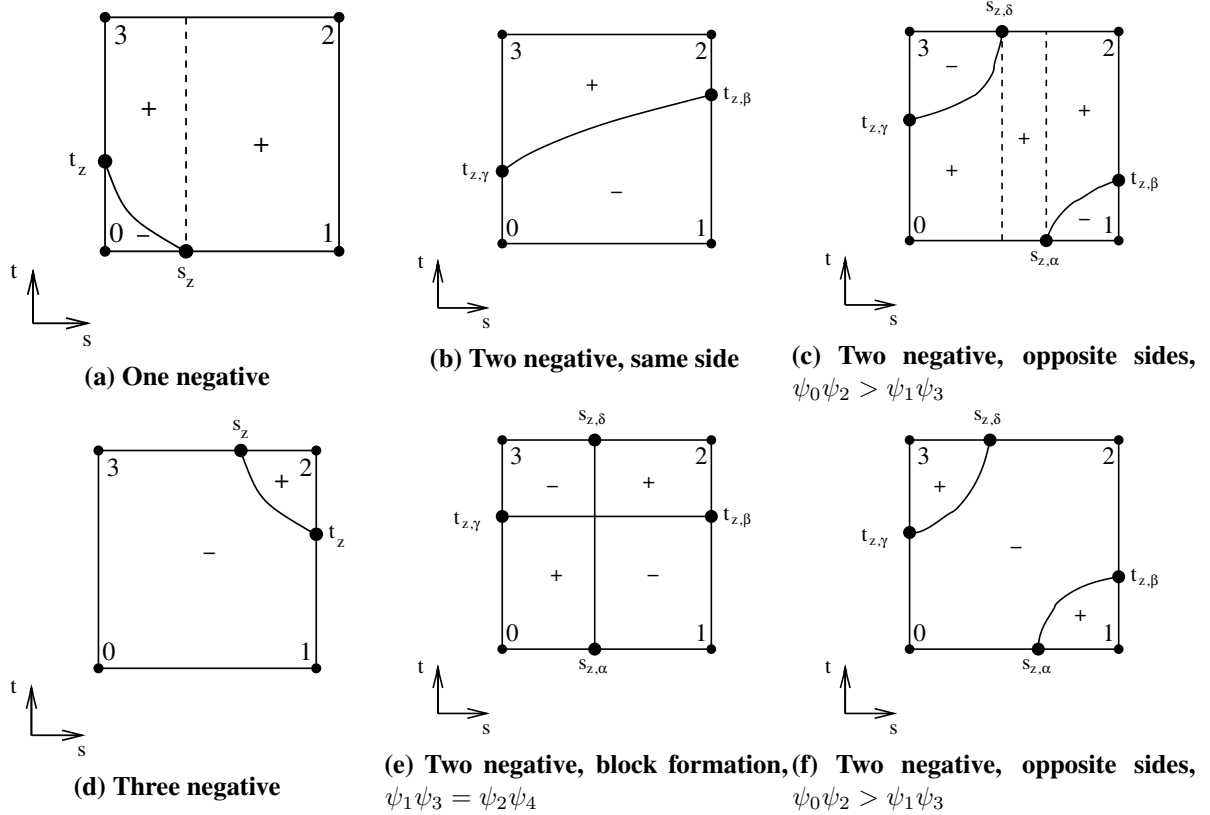(f) **Two negative, opposite sides,** $\psi_0\psi_2 > \psi_1\psi_3$

Figure 2. BCSZ cell integration cases requiring special treatment.

$\psi_{i,\mu}$, $\psi_{i,\eta}$, and $\psi_{i,M}$ is straight forward, and identical to that required by the definition of the UBLD scheme. Though other cases are possible, all can be transformed into one of the cases in Fig. 2 via rotation. $\psi_{i,\mu}$, $\psi_{i,\eta}$, and $\psi_{i,M}$ are evaluated by summing the integral contribution in each region of a particular case given in Fig. 2 To evaluate $\psi_{i,\mu}$, $\psi_{i,\eta}$, and $\psi_{i,M}$ contributions in a region, R, bounded by a curved boundary, we make use of variable limits of integration. Along the curved boundary of R, $\widehat{\psi}_{BCSZ} = 0$, this allows us to determine t as a function of s (or s as a function of t). To find a variable limit of integration with respect to t as a function of s, we first transform $\widehat{\psi}_{BCSZ}$ from an interpolatory polynomial to a moment based polynomial, $f(s,t)$:

$$f(s,t) = f_c + sf_s + tf_t + stf_{st}\,. \tag{15}$$

The coefficients of f are found formula using the interpolatory definition of the basis functions:

$$\begin{bmatrix} 1 & -1 & -1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix} \begin{bmatrix} f_c \\ f_s \\ f_t \\ f_{st} \end{bmatrix} = \vec{\psi}_{BCSZ}, \tag{16}$$

with

$$\vec{\psi}_{BCSZ} = [\psi_{0,BCSZ}, \psi_{1,BCSZ}, \psi_{2,BCSZ}, \psi_{3,BCSZ}]^T. \tag{17}$$

Eq. (15) is then manipulated to find a variable limit of integration with respect to t, $\hat{l}_t$, that is a function of s:

$$\hat{l}_t = -\frac{f_c + f_s s}{f_t + f_{st} s}. \tag{18}$$

Rather than evaluating each $\psi_{i,\mu}$, $\psi_{i,\eta}$, and $\psi_{i,M}$ integral, we evaluate a single, generic, bivariate polynomial integrand that is only a function of s and t. Using MATLAB [6], each integrand is further expanded, then terms of equal degree bivariate polynomials, $s^m t^n$, with $0 \le m \le 3$, $0 \le n \le 3$, are collected. This allows us to calculate the twelve separate integrations of $\psi_{i,\mu}$, $\psi_{i,\eta}$, and $\psi_{i,M}$, as the multiplication of twelve unique sets of constants, $\mathbf{C}_{i,\mu}$, $\mathbf{C}_{i,\eta}$, and $\mathbf{C}_{i,M}$, multiplied by the integrations of a single bivariate integration, over R.

3.2.1 Symbolic integration versus numerical integration

Initially, MATLAB symbolic algebra generated expressions for the integration of the generic bivariate polynomial over R were used to evaluate $\psi_{i,\mu}$, $\psi_{i,\eta}$, and $\psi_{i,M}$. This worked well at low cell counts, but caused the $\widetilde{\psi}_{BCSZ}$ non-linear iteration to fail at higher cell counts. To verify the MATLAB generated expressions, we compared the "exact" symbolic algebra generated results for calculating $\psi_{i,M}$ for $\vec{\psi}_{BCSZ} = [-2\ 0.1\ 200\ 10]^T$ to the value obtained using N_s Gauss quadrature points in s along the curved boundary. A two-point Gauss quadrature in t was used for each corresponding Gauss point in s (tensor product quadrature). An example of the quadrature layout for $N_s = 4$ is given in Fig. 3. In Fig. 4a, we plot $E_{i,quad}$, where:

$$E_i = \frac{|\psi_{i,sym} - \psi_{i,num}|}{|\psi_{i,sym}|}, \tag{19}$$

$\psi_{i,sym}$ is the evaluation of $\psi_{i,M}$ using the symbolic algebra generated expressions, and $\psi_{i,num}$ is the quadrature evaluation of $\psi_{i,M}$ using $2N_s + 4$ quadrature points. Compare the result of Fig. 4a to Fig. 4b, which plots $\widehat{E}_i$:

$$\widehat{E}_i = \frac{|\psi_{i,MAX} - \psi_{i,num}|}{|\psi_{i,MAX}|}, \tag{20}$$

where $\psi_{i,MAX}$ is the quadrature approximation of $\psi_{i,M}$ using Gauss quadrature and $N_S = 40$. Since increasing the number of Gauss quadrature points increases the accuracy of a given quadrature integration, from Fig. 4 it is clear that the "exact" symbolic evaluated expressions suffer from numerical round-off caused by taking the small difference of large numbers. As such, we prefer
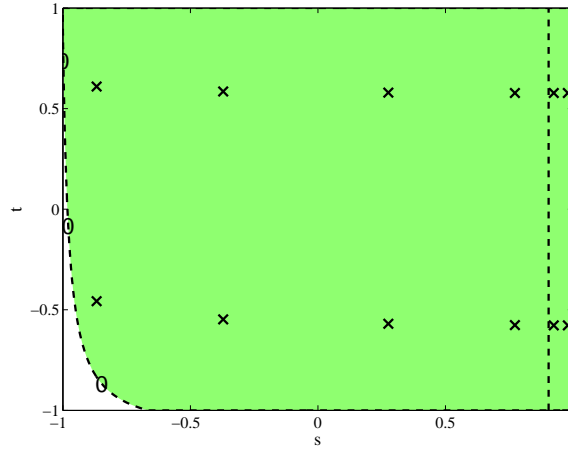
**Figure 3. Quadrature point locations for quadrature integration test.**

quadrature integration to evaluate cell integral quantities for the BCSZ scheme. However, for a fixed number of Gauss points in s, we cannot apriori estimate the error in our quadrature approximation of bivariate polynomial integrals over the region with a curved boundary. Gauss-Kronrod quadrature [7] allows for an estimate of the quadrature approximation error of an integration. In practice, we use the 7-point Gauss / 15-point Kronrod quadrature set in s with a 2-point Gauss quadrature in t to integrate all BCSZ quantities. Using sub-interval refinement, this quadrature integration strategy, and the decompositions shown in Fig. 2 allow for the computation of all BCSZ cell integral quantities to a tolerance less than our non-linear iteration tolerance.

3.3 Non-linear Iteration

To solve the four non-linear equations, we use Newton iteration with a finite difference formed Jacobian. Additionally, we solve for "well" scaled values, using the first $\widehat{\psi}_{BCSZ}$ iterate ($\widetilde{\psi}_{UBLD}$) as a scaling vector. As noted in [3, 8] solving for the non-linearity locally during each sweep precludes the use of Krylov methods in the scattering source iteration.

4   NUMERICAL RESULTS

We now present computational results for a simple test problem, a $10 \, [cm] \times 10 \, [cm]$ void, with S_4 level symmetric angular quadrature, vacuum boundary conditions on the left, top, and right sides, incident angular flux of $1 \, [n/(cm^2 - sec - ster)]$ in the direction of $\mu = 0.868890300722$, $\eta = 0.350021174582$ along the bottom edge, and vacuum conditions along the bottom edge for all other directions incident on the bottom face. In Fig. 5 we compare the results of the UBLD, FLBLD, and BCSZ schemes on an orthogonal mesh of 64×64 cells, plotting the cell average scalar fluxes on a logarithmic scale. Plotting on a logarithmic scale accents the negativities and oscillations of the UBLD scheme. Though undesirable, the negativities and oscillations of the UBLD dampen rapidly. The BCSZ and FLBLD solutions are both strictly non-negative. However, the FLBLD scheme has a
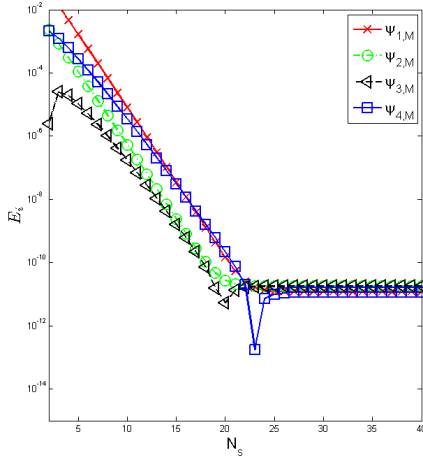
(a) E_i **for quadrature test.**
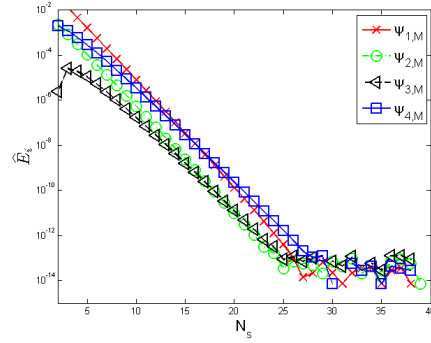
(b) $\widehat{E}_i$ **for quadrature test.**

Figure 4. Comparison of quadrature vs. exact integration errors.

significant amount of numerical diffusion, whereas the BCSZ scheme maintains a sharp boundary at the discontinuity.

In Fig. 6 we show the BCSZ and UBLD cell average scalar flux solutions on a 25 cell $\times$ 25 cell mesh where each interior vertex has been randomly perturbed to create an unstructured mesh. Figure 6 shows the BCSZ scheme is capable of being used for unstructured mesh problems. Though the BCSZ solution exhibits numerical diffusion in Fig. 6, it is a result of the mesh, as both the FLBLD and UBLD schemes exhibit similar increases in numerical diffusion. Again, the UBLD solution exhibits negativities and oscillations while the FLBLD and BCSZ solutions remain strictly non-negative. In Fig. 6, the visible increase in BCSZ numerical diffusion is more an artifact of the logarithmic color scale, the averages in the cells with increased numerical diffusion are roughly a factor of 10^{-6} of the maximum angular flux average in any cell.

Finally, in Fig. 7, we plot E_{ϕ_A},

$$E_{\phi_A} = \sqrt{\sum_{c=1}^{N_{cells}} \Delta x_c \Delta y_c (\widetilde{\phi}_A - \phi_{A,exact})^2} \,, \tag{21}$$

an L_2 like norm of the average scalar flux in each cell for the UBLD, FLBLD, and BCSZ schemes on orthogonal meshes as a function of mesh size. In Eq. (21), $\widetilde{\phi}_A$ is the computed cell average scalar flux for a particular scheme in cell c, Δx_c and Δy_c are the respective cell widths of cell c, and $\phi_{A,exact}$ is the analytic cell average scalar flux in cell c. From Fig. 7, we see that BCSZ is more accurate than UBLD, while both UBLD and BCSZ are significantly more accurate than FLBD. Also, we see that all methods reach the same asymptotic order of convergence, $h^{1/4}$. All of the methods considered converge the error in the cell average angular flux at this low rate due to the discontinuity in the analytic solution.
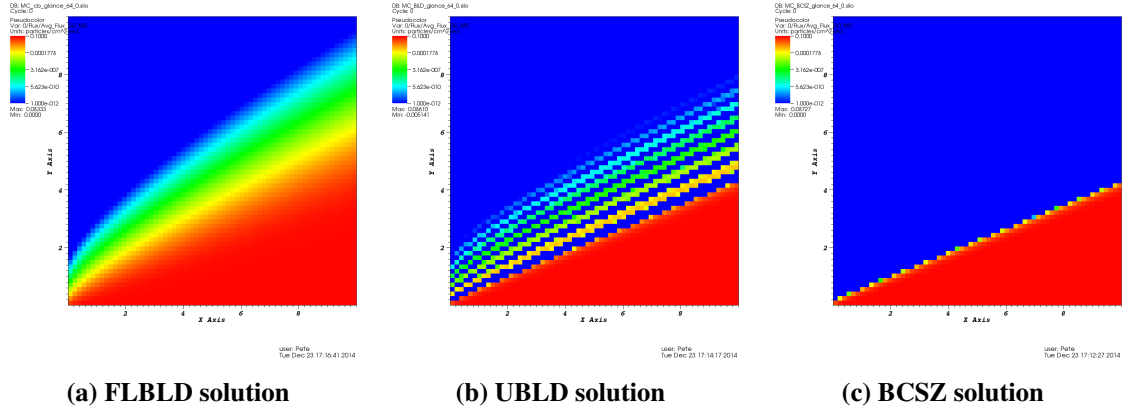
(a) FLBLD solution

(b) UBLD solution

(c) BCSZ solution

Figure 5. 64×64 **mesh cell cell average scalar flux solutions**

(a) FLBLD solution on unstructured grid.

(b) UBLD solution on unstructured grid.
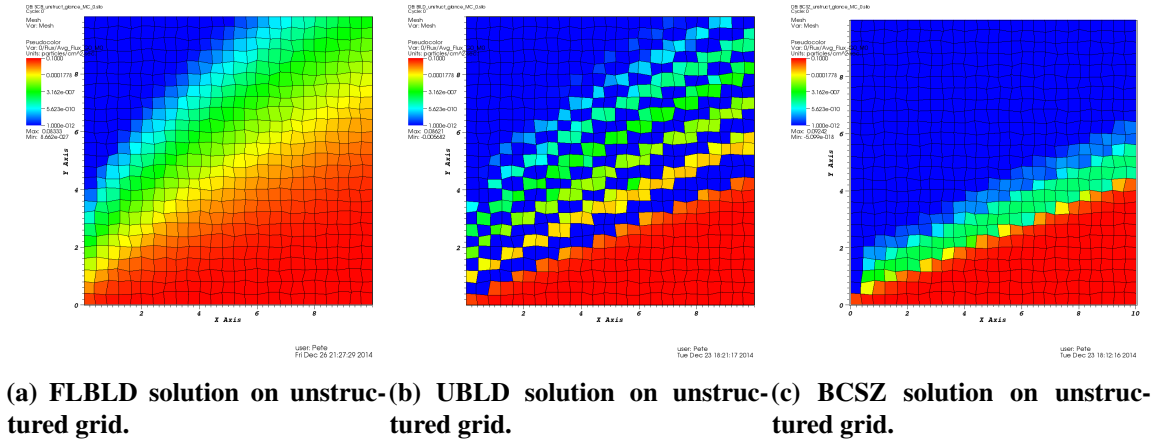
(c) BCSZ solution on unstructured grid.

Figure 6. Glancing void solutions with 25×25 mesh cells.

5 CONCLUSIONS

The BCSZ scheme produces strictly non-negative solutions that are at least as accurate as UBLD solutions. To date, the accuracy of the BCSZ scheme has only been measured against pure absorber/void test problems. We are currently awaiting the completion of a modified interior penalty diffusion synthetic acceleration (MIP DSA) [9] implementation in the TAMU developed S_N solver, PDT [10], to apply the BCSZ to more interesting neutron transport and radiative transfer problems. We see numerous opportunities for future research going forward including:

1. developing appropriate timing problems to assess the computational cost of BCSZ relative to UBLD,

2. comparing the efficiency of our current/near term solution methodology (fixed point iteration with MIP DSA) to the solution methodology proposed by Bruss, et al. [8], and
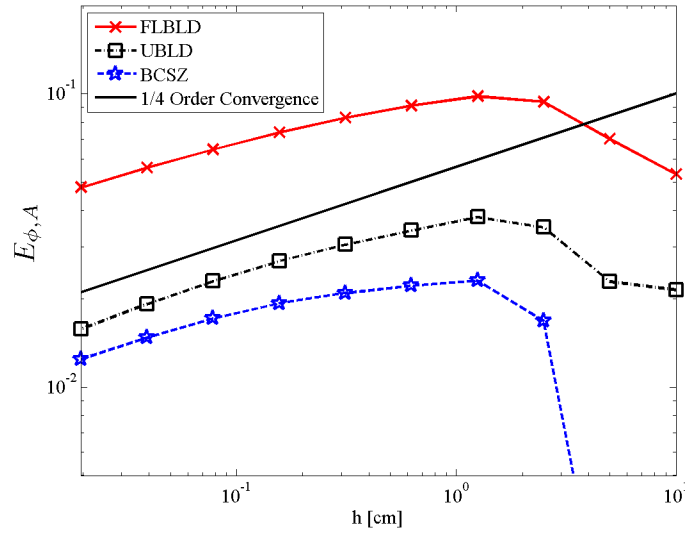
Figure 7. Convergence of cell average scalar flux on orthogonal grids.

3. generalizing BCSZ to higher order, Q_N, trial spaces through the extension of adaptive quadrature integration (adaptive in both s and t to avoid finding where a Q_N function is zero).

6 ACKNOWLEDGMENTS

7 REFERENCES

[1] M. L. Adams, "Discontinuous Finite Element Transport Solutions in Thick Diffusive Problems," *Nuclear Science and Engineering*, **137**, pp. 298–333 (2001).

[2] E. D. Fichtl, J. S. Warsa, and J. D. Densmore, "The Newton-Krylov Method Applied to Negative-Flux Fixup in S_N Transport Calculations," *Nuclear Science and Engineering*, **165**, pp. 331–341 (2010).

[3] P. G. Maginot, J. E. Morel, and J. C. Ragusa, "A Non-negative Moment Preserving Spatial Discretization Scheme for the Lienarized Boltzmann Transport Equation in 1-D and 2-D Cartesian Geometries," *Journal of Computational Physics*, **231**, pp. 6801–6826 (2012).

[4] T. A. Wareing, E. W. Larsen, and M. L. Adams, "Diffusion Accelerated Discontinuous Finite Element Schemes for the S_N Equations in Slab and X,Y Geometries," *Proc. International Topical Mtg. on Advances in Mathematics, Computations and Reactor Physics*, Pittsburgh, PA, April 28-May 2, 1991.

[5] M. L. Adams, "Subcell Balance Methods for Radiative Transfer of Arbitrary Grids," *Transport Theory and Statistical Physics*, **26**, pp. 385–431 (1997).

[6] Mathworks, "MATLAB R-2011a," 2011.

[7] T. N. L. Patterson, "The Optimum Addition of Points to Quadrature Formulae," *Mathematics of Computation*, **22**, pp. 847–856 (1968).

[8] D. E. Bruss, J. E. Morel, and J. C. Ragusa, "S_2SA Preconditioning for the S_N Equations with Strictly Non-negative Spatial Discretization," *Journal of Computational Physics*, **273**, pp. 706–719 (2014).

[9] M. W. Hackemack and J. C. Ragusa, "A DFEM Formulation of the Diffusion Equation on Arbitrary Polyhedral Grids," *Proceedings of the 2014 ANS Winter Meeting*, Anaheim, California, November 9-13, 2014.

[10] W. D. Hawkins, T. S. Bailey, and et al., "Validation of Full-Domain Massively Parallel Transport Sweep Algorithms," *Proceedings of the 2014 ANS Winter Meeting*, Anaheim, California, November 9-13, 2014.