



HAL
open science

Mass conservative limiting and applications to the approximation of the steady-state radiation transport equations

Jean-Luc Guermond, Zuodong Wang

► **To cite this version:**

Jean-Luc Guermond, Zuodong Wang. Mass conservative limiting and applications to the approximation of the steady-state radiation transport equations. 2024. hal-04770516

HAL Id: hal-04770516

<https://hal.science/hal-04770516v1>

Preprint submitted on 6 Nov 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

1 **MASS CONSERVATIVE LIMITING AND APPLICATIONS TO THE**
2 **APPROXIMATION OF THE STEADY-STATE RADIATION TRANSPORT**
3 **EQUATIONS**

4 JEAN-LUC GUERMOND[‡] AND ZUODONG WANG[†]

5 **Abstract.** A limiting technique for scalar transport equations is presented. The originality of the method is
6 that it does not require solving nonlinear optimization problems nor does it rely on the construction of a low-
7 order approximation. The method has minimal complexity and is numerically demonstrated to maintain high-order
8 accuracy. The performance of the method is illustrated on the radiation transport equation.

9 **Key words.** limiting, advection equation, radiation transport equation, stiff sources, conservation equations,
10 asymptotic preserving, invariant domains.

11 **AMS subject classifications.** 35L65, 65M12, 65M60, 76V05

12 **1. Introduction.** The objective of the paper is to present two simple limiting techniques
13 for scalar-valued partial differential equations with a structure like the radiation transport equa-
14 tion. The first limiting method is iterative, locally mass conservative, and does not involve solv-
15 ing any nonlinear optimization problem. At convergence, this method enforces the local mini-
16 mum/maximum principle (assuming that local bounds are known). Two iterations of the method
17 are in general sufficient, but although the method is observed to converge quickly, there is no guar-
18 antee that the bounds are enforced everywhere after a number of iterations that is independent
19 of the meshsize. We then propose a second limiting method that is globally mass conservative
20 and that also does not involve solving any nonlinear optimization problem. The second limiting
21 is applied after the first one. The purpose of this second post-processing is simply to certify that
22 some global bound like positivity is exactly and unconditionally enforced while preserving the total
23 mass of the solution. The combination of these two limiting techniques is illustrated using con-
24 tinuous finite elements stabilized with the continuous interior penalty (CIP) technique (a.k.a. edge
25 stabilization) from Douglas and Dupont [7] and Burman and Hansbo [5]. The method presented in
26 the paper is not restricted to continuous elements and CIP though. It can be used with other spa-
27 tial discretization and other types of stabilization as well. For instance, one can use discontinuous
28 elements (of degree $p \geq 1$) stabilized with the upwind numerical flux. One can also use continuous
29 elements stabilized with other methods like Galerkin Least-Squares, Local Projection Stabilization,
30 Orthogonal Subscale Stabilization, and Subgrid Viscosity.

31 The original motivation for the work presented here is the solution to the radiation transport
32 equation. Based on our experience on nonlinear hyperbolic systems, we have tried for many years
33 to use upwinding, artificial viscosity, and nonlinear variations thereof to enforce positivity of the
34 angular intensity. But it is well established in the literature that one needs to be careful with this
35 type of method in the context of the radiation transport equations. For instance, it is shown in
36 Adams [1] (see also Larsen [14], Larsen et al. [15], and [9, Rmk. 5.2], [11, §3.2]), that upwinding

*Department of Mathematics, Texas A&M University 3368 TAMU, College Station, TX 77843, USA.

†CERMICS, Ecole des Ponts, 77455 Marne-la-Vallée Cedex 2, France and INRIA Paris, 75647 Paris, France

‡This material is based upon work supported in part by the National Science Foundation grant DMS2110868, the Air Force Office of Scientific Research, USAF, under grant/contract number FA9550-18-1-0397, the Army Research Office, under grant number W911NF-19-1-0431, and the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contracts B640889. The support of INRIA through the International Chair program is acknowledged.

37 and artificial viscosity may lock in the diffusion limit if the artificial viscosity is strong enough so
 38 as to guarantee that the local minimum/maximum principle holds. We have long tried to modify
 39 the artificial viscosity techniques to overcome this difficulty. For instance in [11] we have adopted a
 40 technique inspired by Gosse and Toscani [8] consisting of scaling the transport term, the scattering
 41 term, and the artificial diffusion by $\frac{1}{1+\sigma^s h}$ where h is the mesh size and σ^s is the scattering cross
 42 section. This gives a method that is indeed positive, convergent, and asymptotic preserving, but
 43 it is not locally conservative on nonuniform meshes. Moreover, it does not behave properly with
 44 grazing incidences in the diffusion regime unless the grazing boundary data is replaced by its angular
 45 average weighted with Chandrasekhar’s H-function, thereby seriously diminishing the usefulness of
 46 the method as estimating the H-function is nontrivial. In the paper we propose instead to rely on the
 47 Galerkin approximation augmented with a traditional linear stabilization technique and to enforce
 48 the local minimum/maximum principle by using a mass conservative post-processing technique.

49 An important aspect of the limiting technique we propose is the estimation of local bounds.
 50 The approach we use for this purpose consists of using the method of characteristics as in Yee
 51 et al. [28, Eq. (16)]. Note that here we do not propose to use the method of characteristics as a
 52 solution method but just as a means to estimate guaranteed local upper and lower bounds. We
 53 refer to Lathrop [16], Sanchez and McCormick [25, §II.C.2] for early reviews on solution methods
 54 based on the method of characteristics. Once local bounds are found, the next problem consists
 55 of enforcing these bounds. This can be done as in Maginot et al. [19] where the authors develop
 56 a nonlinear characteristics-based methods that is positivity-preserving. Another idea is to invoke
 57 strategies like the so-called flux corrected transport method of Zalesak combining in a nonlinear
 58 fashion a low-order and a high-order approximation [29]. The problem with this approach is that it
 59 relies on a low-order approximation which itself relies in one way or another on artificial viscosity,
 60 which as explained above leads to locking. Another possibility more aligned with what we propose
 61 in the paper is to enforce local bounds by using nonlinear optimization techniques. The problem is
 62 then to find a maximum-principle-satisfying object that minimizes some distance to the maximum-
 63 principle-violating solution while maintaining constant the total mass (either locally or globally).
 64 This type of method is developed in e.g., Bochev et al. [3, 4], Yee et al. [28] Peterson et al. [24], and
 65 references therein. A fast convergent algorithm for finding such a minimizer is proposed in Liu et al.
 66 [18], Ancellin et al. [2]. Here we propose a slightly different approach consisting of just computing a
 67 quasi-minimizer with a set of two methods whose algorithmic complexity is significantly lower than
 68 that of computing a genuine minimizer. We finally mention that if positivity is the only property
 69 one is interested in, one can use “fix-up” techniques like that presented in Lewis and Miller [17,
 70 Chap. 4]. This iterative method sets negative fluxes to zero and continues the iterations with the
 71 other degree of freedom to restore balance. This idea is generalized to the discontinuous Galerkin
 72 setting in Hamilton et al. [12]. Other improvements of the fix-up technique are presented in Maginot
 73 et al. [20].

74 The paper is organized as follows. We introduce in §2 the two limiting techniques mentioned
 75 above. We choose to present them without invoking the radiation transport equation as these two
 76 methods are quite general and can be used for other purposes. We recall in §3 how local bounds can
 77 be extracted for the scalar-valued linear transport equations using the method of characteristics.
 78 We also introduce in this section a second-order relaxation technique that is essential to achieve
 79 accuracy beyond second-order. We present in §4 the Galerkin approximation of the radiation
 80 transport equation stabilized with the continuous interior penalty technique. The full solution
 81 method is explained in Algorithms A.1 and A.2. The proposed method is illustrated on the scalar
 82 linear transport equation in §5 and on the radiation transport equation in §6.

83 **2. Iterative limiting algorithm.** We present in this section two simple conservative iterative
 84 limiting algorithms. The first one preserves mass locally. The second one only preserves mass
 85 globally, and is only used as theoretical safeguard. These two algorithms are quite general. They
 86 are not specific to the radiation transport equations and can be used in many different contexts.

87 **2.1. The setting.** Let $\{x_i\}_{i \in \mathcal{V}}$ be degrees of freedom that have to be limited, $x_i \in \mathbb{R}$ for
 88 all i in the index set \mathcal{V} . Let $\mathcal{M} := \sum_{i \in \mathcal{V}} m_i x_i$ be the total mass, where the coefficients m_i are
 89 nonnegative and are called mass at i . For all $i \in \mathcal{V}$, let u_i^{\min} and u_i^{\max} be the local minimum and
 90 maximum we want to enforce on the i -th degree of freedom. It is henceforth assumed that these
 91 local bounds u_i^{\min} and u_i^{\max} satisfy the following reasonable estimate:

$$92 \quad (2.1) \quad \sum_{i \in \mathcal{V}} m_i u_i^{\min} \leq \mathcal{M} \leq \sum_{i \in \mathcal{V}} m_i u_i^{\max}.$$

93 Our objective is to post-process the degrees of freedom $\{x_i\}_{i \in \mathcal{V}}$ to enforce the local bounds while
 94 maintaining mass conservation, either locally or globally.

95 We assume that the lumped mass is non-negative, i.e., $m_i \geq 0$ for all $i \in \mathcal{V}$. This assumption
 96 holds, for example, when using finite elements with Bernstein basis of any polynomial degree, and
 97 for Lagrange bases of degree 1, 2 and 3 with equally distributed interpolation nodes. We introduce
 98 a concept of locality by introducing a notion of stencil. For all $i \in \mathcal{V}$ we assume that there exists
 99 a collection of indices $\mathcal{I}(i) \subsetneq \mathcal{V}$ that can exchange mass with i . We call this index set stencil. For
 100 instance, for finite elements we have $j \in \mathcal{I}(i)$ if $\varphi_j \varphi_i \neq 0$, where φ_i and φ_j are the global shape
 101 function associated with the degrees of freedom i and j . We also define $\mathcal{I}(i)^* := \mathcal{I}(i) \setminus \{i\}$. For all
 102 the finite element tests reported in §5 and §6, the set $\mathcal{I}(i)$ is the standard stencil, i.e., $\mathcal{I}(i)$ collects
 103 the indices of all the global shape functions φ_j that are such that $\varphi_j \varphi_i \neq 0$.

104 **2.2. Local conservative limiting.** We describe here an iterative algorithm that is locally
 105 conservative. The algorithm consists of looping over all the indices i in \mathcal{V} and proceeds as follows.
 106 Let $i \in \mathcal{V}$. If $m_i = 0$, then the i -th degree of freedom does not contribute to the global mass; we
 107 simply set $y_i = \min(u_i^{\max}, \max(u_i^{\min}, x_i))$ and continue to the next degree of freedom in the list \mathcal{V} .
 108 This possible modification of the value at i does not change the mass, either locally or globally. Let
 109 us assume now that $m_i > 0$. Then either $x_i \leq u_i^{\max}$ (i.e., the maximum principle is satisfied) or
 110 $u_i^{\max} < x_i$ (i.e., the maximum principle is violated). If $x_i \leq u_i^{\max}$, we do nothing and continue to
 111 the next index in the loop. If $u_i^{\max} < x_i$, then we compute

$$112 \quad (2.2a) \quad a_i^+ := \sum_{j \in \mathcal{I}(i)^*} m_j \max(0, u_j^{\max} - x_j), \quad b_i^+ := \max\left(x_i - \frac{a_i^+}{m_i}, u_i^{\max}\right),$$

$$113 \quad (2.2b) \quad \ell_i^+ := \begin{cases} 0 & \text{if } 0 = a_i^+ \\ m_i \frac{x_i - b_i^+}{a_i^+} & \text{otherwise,} \end{cases}$$

114 and the actual limiting with respect to u_i^{\max} is done by setting

$$115 \quad (2.3a) \quad y_j := x_j + \ell_i^+ \max(0, u_j^{\max} - x_j), \quad \forall j \in \mathcal{I}(i)^*$$

$$116 \quad (2.3b) \quad y_i := b_i^+.$$

117 **LEMMA 2.1.** *Let $i \in \mathcal{V}$. If $u_i^{\max} < x_i$, then the following holds for all $\{y_j\}_{j \in \mathcal{I}(i)}$ given by (2.3):*
 118 (i) *There is local mass conservation, i.e., $m_i(y_i - x_i) + \sum_{j \in \mathcal{I}(i)^*} m_j(y_j - x_j) = 0$.*

119 (ii) $x_j \leq y_j \leq \max(x_j, u_j^{\max})$ for all $j \in \mathcal{I}(i)^*$.

120 (iii) $u_i^{\max} \leq y_i \leq x_i$. Moreover, $y_i < x_i$ if $0 < a_i^+$.

121 *Proof.* (i) Assume first that $a_i^+ \neq 0$. The equation (2.3a) implies that the mass transferred
 122 to the node $j \in \mathcal{I}(i)^*$ is $m_j \ell_i^+ \max(0, u_j^{\max} - x_j)$, while the equation (2.3b) implies that the mass
 123 transferred to the node i is $m_i(b_i^+ - x_i)$. The total mass exchange is then

$$\begin{aligned} 124 \quad \Delta_i &:= m_i(y_i - x_i) + \sum_{j \in \mathcal{I}(i)^*} m_j(y_j - x_j) \\ 125 \quad &= m_i(b_i^+ - x_i) + \ell_i^+ \sum_{j \in \mathcal{I}(i)^*} m_j \max(0, u_j^{\max} - x_j) = m_i(b_i^+ - x_i) + \ell_i^+ a_i^+. \end{aligned}$$

126 But since $a_i^+ \neq 0$, (2.2b) gives $\ell_i^+ = m_i \frac{x_i - b_i^+}{a_i^+}$; hence

$$127 \quad \Delta_i = m_i(b_i^+ - x_i) + m_i \frac{x_i - b_i^+}{a_i^+} a_i^+ = 0.$$

128 Otherwise, if $a_i^+ = 0$, then $b_i^+ = \max(x_i, u_i^{\max}) = x_i$ and $\ell_i^+ := 0$. Then again $\Delta_i = 0$.

129 (ii) The assertion is trivial when $a_i^+ = 0$. Let assume now that $a_i^+ > 0$. As $x_i - \frac{a_i^+}{m_i} \leq b_i^+$, we
 130 have $\ell_i^+ := m_i \frac{x_i - b_i^+}{a_i^+} \leq 1$. Moreover, the assumptions $u_i^{\max} < x_i$ and $b_i^+ := \max(x_i - \frac{a_i^+}{m_i}, u_i^{\max})$,
 131 imply that $b_i^+ \leq \max(x_i, u_i^{\max}) < x_i$; hence, $0 \leq \ell_i^+$. As $0 \leq \ell_i^+ \leq 1$, we infer that $x_j \leq$
 132 $x_j + \ell_i^+ \max(0, u_j^{\max} - x_j) := y_j \leq x_j + \max(0, u_j^{\max} - x_j) = \max(x_j, u_j^{\max})$.

133 (iii) By (2.2a), we have $u_i^{\max} \leq b_i^+$ and $b_i^+ = \max(x_i - \frac{a_i^+}{m_i}, u_i^{\max}) \leq \max(x_i, u_i^{\max}) = x_i$; hence,
 134 $u_i^{\max} \leq y_i := b_i^+ \leq x_i$. Let us now assume that $a_i^+ \neq 0$. Notice that if $b_i^+ = u_i^{\max}$ then $y_i = b_i^+ < x_i$
 135 (whether a_i^+ is equal to zero or not does not matter here). On the other hand, if $b_i^+ = x_i - \frac{a_i^+}{m_i}$,
 136 then the assumption $a_i^+ \neq 0$ implies that $y_i = b_i^+ x_i$. \square

137 The same idea as above can be used to enforce the minimum principle. Let $i \in \mathcal{V}$. Either
 138 $u_i^{\min} \leq x_i$ (i.e., the local minimum principle is satisfied) or $x_i < u_i^{\min}$ (i.e., the local minimum
 139 principle is violated). If $u_i^{\min} \leq x_i$, do nothing and continue to the next index in the list \mathcal{V} . If
 140 $x_i < u_i^{\min}$, then compute

$$141 \quad (2.4a) \quad a_i^- := \sum_{j \in \mathcal{I}(i)^*} m_j \max(0, x_j - u_j^{\min}), \quad b_i^- := \min(x_i + \frac{a_i^-}{m_i}, u_i^{\min}),$$

$$142 \quad (2.4b) \quad \ell_i^- := \begin{cases} 0 & \text{if } 0 = a_i^- \\ m_i \frac{x_i - b_i^-}{a_i^-} & \text{otherwise,} \end{cases}$$

143 and set

$$144 \quad (2.5a) \quad y_j := x_j + \ell_i^- \max(0, x_j - u_j^{\min}), \quad \forall j \in \mathcal{I}(i) \setminus \{i\}$$

$$145 \quad (2.5b) \quad y_i := b_i^-.$$

146 **LEMMA 2.2.** *Let $i \in \mathcal{V}$. If $x_i < u^{\min}$, then the following holds for all $\{y_j\}_{j \in \mathcal{I}(i)}$ given by (2.5):*

- 147 (i) $m_i(y_i - x_i) + \sum_{j \in \mathcal{I}(i)^*} m_j(y_j - x_j) = 0$, i.e., (2.5) is locally mass conservative
 148 (ii) $\min(x_j, u_j^{\min}) \leq y_j \leq x_j$ for all $j \in \mathcal{I}(i)^*$.
 149 (iii) $x_i \leq y_i \leq u_i^{\min}$. Moreover, $x_i < y_i$ if $0 < a_i^-$.

150 The algorithm (2.2)–(2.5) guarantees that the local maximum decreases and the local minimum
 151 increases until reaching the prescribed values (see Items (iii) and (iii) in Lemmas 2.1 and 2.2). The
 152 statements made in Items (ii) and (ii) in Lemmas 2.1 and 2.2 guarantee that by correcting $x_i \rightarrow y_i$,
 153 the neighboring values that are already in bounds stay in bounds after limiting. The algorithm is
 154 iterative, but there is no guarantee that the prescribed bounds are reached in a finite number of
 155 iterations. Extensive numerical tests show though that convergence is quick when one starts from
 156 a reasonable solution (e.g., a linearly stabilized Galerkin approximation). In all the tests reported
 157 in the paper the above algorithm is only applied two times in a row. The limiting algorithm is
 158 summarized in Algorithm 2.1. We note that the order of dofs does not influence the behavior of
 the algorithm.

Algorithm 2.1 Local conservative limiting (2.2)–(2.5)

Require: Bounds $\{u_i^{\max}, u_i^{\min}\}_{i \in \mathcal{V}}$, $\{x_i\}_{i \in \mathcal{V}}$.

```

1: for  $i \in \mathcal{V}$  do ▷ Loop over dofs
2:   if  $m_i = 0$  then
3:      $y_i = \min(u_i^{\max}, \max(u_i^{\min}, x_i))$ .
4:   else if  $u_i^{\max} < x_i$  then ▷ Local maximum principle violated
5:     Compute  $a_i^+$ ,  $b_i^+$ ,  $\ell_i^+$  using (2.2).
6:      $y_i = b_i^+$ 
7:     for  $j \in \mathcal{I}(i)^*$  do
8:        $y_j = x_j + \ell_i^+ \max(0, u_j^{\max} - x_j)$ 
9:     end for
10:  else if  $x_i < u_i^{\min}$  then ▷ Local minimum principle violated
11:    Compute  $a_i^-$ ,  $b_i^-$ ,  $\ell_i^-$  using (2.4).
12:     $y_i = b_i^-$ 
13:    for  $j \in \mathcal{I}(i)^*$  do
14:       $y_j = x_j + \ell_i^- \max(0, x_j - u_j^{\min})$ 
15:    end for
16:  else ▷ Do nothing
17:     $y_i = x_i$ 
18:  end if
19: end for
20: return  $\{y_i\}_{i \in \mathcal{V}}$ 

```

159

160 **2.3. Global conservative limiting.** As the iterative algorithm (2.2)–(2.5) does not guaran-
 161 tee that the prescribed bounds are achieved in a finite number of iterations, we now propose a final
 162 conservative post-processing that can be used to make sure that global bounds that are essential
 163 to the physics are strictly enforced.

164 Consider the set of degrees of freedom $\{x_i\}_{i \in \mathcal{V}}$, the associated mass $\mathcal{M} := \sum_{i \in \mathcal{V}} m_i x_i$, and the
 165 local bounds $\{u_i^{\min}, u_i^{\max}\}_{i \in \mathcal{V}}$, which we recall are assumed to satisfy (2.1). We compute a new set

166 of limited values $\{z_i\}_{i \in \mathcal{V}}$ as follows:

167 (2.6a) For all $i \in \mathcal{V}$, compute, $y_i := \min((\max(x_i, u_i^{\min}), u_i^{\max}))$.

168 (2.6b) Then compute $\alpha^+ := \max\left(0, \frac{\mathcal{M} - \sum_{j \in \mathcal{V}} m_j y_j}{\sum_{j \in \mathcal{V}} m_j (u_j^{\max} - y_j)}\right)$,

169 (2.6c) and $\alpha^- := \max\left(0, \frac{\mathcal{M} - \sum_{j \in \mathcal{V}} m_j y_j}{\sum_{j \in \mathcal{V}} m_j (u_j^{\min} - y_j)}\right)$.

170 (2.6d) For all $i \in \mathcal{V}$, set, $z_i := y_i + \alpha^+(u_i^{\max} - y_i) + \alpha^-(u_i^{\min} - y_i)$.

171

172 LEMMA 2.3. *The following holds for all $i \in \mathcal{V}$:*

173 (2.7) $u_i^{\min} \leq z_i \leq u_i^{\max}$,

174 (2.8) $\sum_{i \in \mathcal{V}} m_i z_i = \sum_{i \in \mathcal{V}} m_i x_i$.

175 *Proof.* (1) By definition $0 \leq \alpha^+$. If $\mathcal{M} - \sum_{i \in \mathcal{V}} m_i y_i \leq 0$ then $\alpha^+ = 0$. If $\mathcal{M} - \sum_{i \in \mathcal{V}} m_i y_i \geq 0$,
176 then using the assumption (2.1) together with $\sum_{i \in \mathcal{V}} m_i (u_i^{\max} - y_i) \geq 0$ (which holds owing to
177 (2.6a)), we infer that

178
$$\alpha^+ \leq \frac{\mathcal{M} - \sum_{i \in \mathcal{V}} m_i y_i}{\sum_{i \in \mathcal{V}} m_i (u_i^{\max} - y_i)} \leq \frac{\sum_{i \in \mathcal{V}} m_i u_i^{\max} - \sum_{i \in \mathcal{V}} m_i y_i}{\sum_{i \in \mathcal{V}} m_i (u_i^{\max} - y_i)} = 1.$$

179 Hence $0 \leq \alpha^+ \leq 1$. We proceed similarly to show that $0 \leq \alpha^- \leq 1$.

180 (2) Observing that $u_i^{\min} - y_i \leq 0$, $0 \leq u_i^{\max} - y_i$ (which holds owing to (2.6a)), $0 \leq \alpha^+ \leq 1$,
181 and $0 \leq \alpha^-$, we obtain

182
$$z_i := y_i + \alpha^+(u_i^{\max} - y_i) + \alpha^-(u_i^{\min} - y_i) \leq y_i + \alpha^+(u_i^{\max} - y_i)$$

183
$$\leq y_i + (u_i^{\max} - y_i) = u_i^{\max}.$$

184 We proceed similarly to prove that $u_i^{\min} \leq z_i$.

185 (3) Since $0 \leq \sum_{i \in \mathcal{V}} m_i (u_i^{\max} - y_i)$ and $\sum_{i \in \mathcal{V}} m_i (u_i^{\min} - y_i) \leq 0$, we have

186
$$\sum_{i \in \mathcal{V}} m_i z_i = \sum_{i \in \mathcal{V}} m_i y_i + \alpha^+ \sum_{i \in \mathcal{V}} m_i (u_i^{\max} - y_i) + \alpha^- \sum_{i \in \mathcal{V}} m_i (u_i^{\min} - y_i)$$

187
$$= \sum_{i \in \mathcal{V}} m_i y_i + \max(0, \mathcal{M} - \sum_{i \in \mathcal{V}} m_i y_i) + \min(0, \mathcal{M} - \sum_{i \in \mathcal{V}} m_i y_i) = \mathcal{M}.$$

188 This completes the proof. \square

189 *Remark 2.4* (Local vs. global bounds). Although the bounds u_i^{\min} and u_i^{\max} invoked in the
190 algorithm can in principle be local, we insist that the algorithm (2.6) should only be used to make
191 sure that global bounds are enforced because the mass conservation mechanism is not local. Notice
192 that the local and global limiting algorithms can be used to limit the solution of time-dependent
193 nonlinear conservation equations as well. Preliminary tests (not shown here for brevity) reveal
194 that these two algorithms perform very well when combined as described above. Preliminary tests,

195 (not shown here for brevity) also show that the global limiting algorithm should not be used alone.
 196 When used alone, the algorithm may make the approximation to converge to a solution that does not
 197 satisfy the Rankin-Hugoniot condition. The purpose of the global limiting is purely theoretical. Its
 198 purpose is simply to certify that some global bound (like positivity) is exactly and unconditionally
 199 enforced while preserving the total mass of the solution, but the bulk of the job is done by the local
 200 limiting algorithm \square

201 **3. Bounds for the transport equation.** A critical aspect of the algorithm (2.2)–(2.5) is
 202 the estimation of the local bounds $\{u_i^{\min}, u_i^{\max}\}_{i \in \mathcal{V}}$. We explain in this section how this can be
 203 done for scalar transport equations. To simplify the presentation, we assume that the degrees of
 204 freedoms enumerated with the list \mathcal{V} are Lagrange finite elements associated with the Lagrange
 205 nodes $\{\mathbf{x}_i\}_{i \in \mathcal{V}}$ from a mesh \mathcal{T}_h based on a domain $D \subset \mathbb{R}^d$, $d \in \{1, 2, 3\}$. In the rest of this section,
 206 the symbol x is a position vector in one dimension and \mathbf{x} is a position vector in two and higher
 207 dimensions.

208 **3.1. Computations of the bounds.** We start by explaining the method we use to compute
 209 bounds in one space dimension. We then generalize the method to high space dimensions later.
 210 No originality is claimed here as most of what is said in this section can be found in the radiation
 211 transport literature, see e.g., Lathrop [16].

212 **3.1.1. One-dimensional case.** Let us consider the one-dimensional transport equation

$$213 \quad (3.1) \quad \Omega(x) \partial_x u(x) + \sigma(x) u(x) = q(x).$$

214 We assume to simplify the presentation that both Ω and σ are piecewise constant over the mesh
 215 cells. We also assume that Ω is not equal to 0; otherwise, (3.1) is trivial and there is nothing to
 216 limit. For each $i \in \mathcal{V}$, let x_i be one of the Lagrangian node in the mesh where one wants to estimate
 217 an upper and a lower bound. Let K be the (unique) cell in the mesh \mathcal{T}_h such that the segment
 218 $\{x_i - s\Omega(s) \mid s \in \mathbb{R} > 0\} \cap K$ is not empty. This cell exists if x_i is not on the inflow boundary (if
 219 x_i is on the inflow boundary one can set $u_i^{\min} = u_i^{\max} = \alpha^\partial(x_i)$ where α^∂ is the inflow boundary
 220 data of the problem). Then one defines $x_i^{\text{up}} := \partial K \cap \{x_i - s\Omega(s) \mid s \in \mathbb{R} > 0\}$. The point x_i^{up}
 221 is the farthest away from x_i in K along the direction $-\Omega$. This point is at the upwind boundary of
 222 K (see the left panel in Figure 3.1 for example). Then, setting $\Omega := \Omega|_K$ and $\sigma := \sigma|_K$, the exact
 223 solution to (3.1) is such that

$$224 \quad (3.2) \quad u(x_i) = u(x_i^{\text{up}}) e^{\frac{\sigma}{\Omega}(x_i^{\text{up}} - x_i)} + \int_{x_i^{\text{up}}}^{x_i} \frac{q(x)}{\Omega} e^{\frac{\sigma}{\Omega} s} ds.$$

225 Setting $q^{\min} := \min_{x \in K} q(x)$, $q^{\max} := \max_{x \in K} q(x)$, this gives $u_i^{\min} \leq u(x_i) \leq u_i^{\max}$ where

$$226 \quad (3.3a) \quad u_i^{\min} := u(x_i^{\text{up}}) e^{\frac{\sigma}{\Omega}(x_i^{\text{up}} - x_i)} + \frac{q^{\min}}{\sigma} (1 - e^{\frac{\sigma}{\Omega}(x_i^{\text{up}} - x_i)}),$$

$$227 \quad (3.3b) \quad u_i^{\max} := u(x_i^{\text{up}}) e^{\frac{\sigma}{\Omega}(x_i^{\text{up}} - x_i)} + \frac{q^{\max}}{\sigma} (1 - e^{\frac{\sigma}{\Omega}(x_i^{\text{up}} - x_i)}).$$

228 As these expressions are not robust with respect to σ (in particular they do not make sense for
 229 $\sigma = 0$), we instead define $\Delta x := \left| \frac{x_i^{\text{up}} - x_i}{\Omega} \right|$, and use the following bounds motivated by Taylor

230 expansion, when $\frac{\sigma}{\Omega}(x_i - x_i^{\text{up}}) \leq 0.005$:

$$231 \quad (3.4a) \quad u_i^{\min} := u(x_i^{\text{up}})e^{\frac{\sigma}{\Omega}(x_i - x_i^{\text{up}})} + q^{\min} \left(\Delta x - \sigma \frac{\Delta x^2}{2} + \sigma^2 \frac{\Delta x^3}{6} - \sigma^3 \frac{\Delta x^4}{24} \right),$$

$$232 \quad (3.4b) \quad u_i^{\max} := u(x_i^{\text{up}})e^{\frac{\sigma}{\Omega}(x_i^{\text{up}} - x_i)} + q^{\max} \left(\Delta x - \sigma \frac{\Delta x^2}{2} + \sigma^2 \frac{\Delta x^3}{6} \right).$$

233 So assuming that $u(x_i^{\text{up}})$ is known, or one has access to some good approximation thereof by some
 234 iteration process, the bounds (3.3) (or (3.4) if $\sigma\Delta x$ is small) gives an estimate of the local bounds
 235 of the solution at x_i .

236 **3.1.2. Two-dimensional case.** The high-dimensional case is similar to the one-dimensional
 237 one. We consider $\mathcal{T}(i)$ the collection of the cells containing \mathbf{x}_i and we set $\mathbf{x}_i^{\text{up}} := \bigcap_{K \in \mathcal{T}(i)} \partial K \cap$
 238 $\{\mathbf{x}_i - s\boldsymbol{\Omega}(s) \mid s \in \mathbb{R} > 0\}$. The point \mathbf{x}_i^{up} (green one) is the farthest away from \mathbf{x}_i (red node) in
 239 $\mathcal{T}(i)$ along the direction $-\boldsymbol{\Omega}$ (see the right panel in Figure 3.1 for example). One key difference
 240 with the one-dimensional case is that \mathbf{x}_i^{up} is not necessarily a Lagrangian node. Then if one only
 241 knows u at the vertices of the triangulation, then one needs to reconstruct u at \mathbf{x}_i^{up} . For instance,
 242 in two space dimensions one can proceed as follows:

- 243 1. Compute $\mathbf{x}_i^{\text{up}} := \bigcap_{K \in \mathcal{T}(i)} \partial K \cap \{\mathbf{x}_i - s\boldsymbol{\Omega}(s) \mid s \in \mathbb{R} > 0\}$.
- 244 2. Define $u(\mathbf{x}_i^{\text{up}}) := u(\mathbf{x}_{i_1}) + \theta(u(\mathbf{x}_{i_2}) - u(\mathbf{x}_{i_1}))$, where $\theta = \frac{\|(\mathbf{x}_i - \mathbf{x}_{i_1}) \times \boldsymbol{\Omega}\|_{\ell^2}}{\|(\mathbf{x}_{i_2} - \mathbf{x}_{i_1}) \times \boldsymbol{\Omega}\|_{\ell^2}}$.

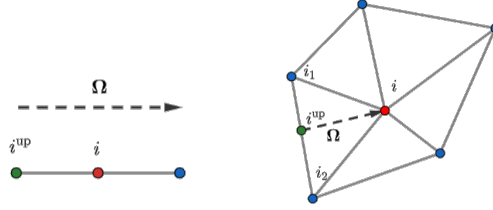


Figure 3.1: Examples for inflow nodes: left panel: 1D case; right panel: 2D case.

245 **3.2. Relaxation.** As for any local limiting technique that we know of, second-order relaxation
 246 of the bounds must be applied to avoid order reduction. We refer for instance to [10, §4.7] where
 247 this question is discussed at length. This issue is addressed by another way in the finite volume
 248 literature consists of relaxing the slope reconstructions; see Harten and Osher [13], Schmidtman
 249 et al. [26, §2.1]. Here we adopt the methodology proposed in [10, §4.7]. Let us assume that we
 250 have (approximate) knowledge of u at the Lagrange node $\{\mathbf{x}_i\}_{i \in \mathcal{V}}$, say $u(\mathbf{x}) \approx \sum_{i \in \mathcal{V}} u_i \varphi_i(\mathbf{x})$ where
 251 $\{\varphi_i\}_{i \in \mathcal{V}}$ are the global Lagrange shape functions. We estimate the local curvature of u by

$$252 \quad (3.5) \quad \Delta_i^2 := \frac{\sum_{j \in \mathcal{I}(i)^*} \beta_{ij}(u_i - u_j)}{\sum_{j \in \mathcal{I}(i)^*} \beta_{ij}}, \quad \text{and set} \quad \overline{\Delta}_i^2 := \min_{j \in \mathcal{I}(i)} \Delta_j^2,$$

253 where $\beta_{ij} = \int_D \nabla \varphi_j \cdot \nabla \varphi_i \, dx$ are the stiffness coefficients of the Laplace operator. Observe that
 254 $\sum_{j \in \mathcal{I}(i)^*} \beta_{ij} = -\beta_{ii} = -\int_D (\nabla \varphi_i)^2 \, dx \neq 0$. The relaxed local bounds are then defined as follows:

$$255 \quad (3.6) \quad \overline{u}_i^{\min} := \max(u_i^{\min} - \overline{\Delta}_i^2, u_i^{\min}), \quad \overline{u}_i^{\max} := \min(u_i^{\max} + \overline{\Delta}_i^2, u_i^{\max}),$$

256 where u^{\min} and u^{\max} are global bounds, if known (see Algorithm A.3). Relaxation is essential to
 257 recovers optimal convergence rates, see numerical tests in §5.

258 **4. Approximation of the radiation transport equations.** We are going to illustrate
 259 the local mass conserving limiting algorithm (2.2)–(2.5) and the global mass conserving limiting
 260 algorithm (2.6) on the radiation transport equation. We introduce here the radiation transport
 261 equation and the associated finite element approximation.

262 **4.1. The model problem.** The computational domain D is assumed to be an open, bounded,
 263 connected polyhedron in \mathbb{R}^3 . The boundary of D is denote by ∂D . The symbol \mathbf{n} denotes the outer
 264 unit normal on ∂D . The unit sphere in \mathbb{R}^3 is denoted by \mathcal{S} . The surface of the unit sphere is
 265 denoted $|\mathcal{S}|$; recall that $|\mathcal{S}| = 4\pi$. We set $\mathcal{O} := \partial D \times \mathcal{S}$, and to define the boundary conditions we
 266 introduce the inflow boundary $\mathcal{O}_- := \{(\mathbf{x}, \boldsymbol{\Omega}) \in \mathcal{O} \mid \boldsymbol{\Omega} \cdot \mathbf{n}(\mathbf{x}) < 0\}$.

267 Given a non-negative source term $q : D \times \mathcal{S} \rightarrow \mathbb{R}_+$, and a non-negative boundary data $\alpha^\partial : \mathcal{O}_- \rightarrow \mathbb{R}_+$, we look for $\Psi : D \times \mathcal{S} \rightarrow \mathbb{R}_+$ so that

269 (4.1a) $\boldsymbol{\Omega} \cdot \nabla \Psi(\mathbf{x}, \boldsymbol{\Omega}) + \sigma^t(\mathbf{x})\Psi(\mathbf{x}, \boldsymbol{\Omega}) = \sigma^s(\mathbf{x})\bar{\Psi}(\mathbf{x}) + q(\mathbf{x}, \boldsymbol{\Omega}),$ in $D \times \mathcal{S}$

270 (4.1b) $\Psi(\mathbf{x}, \boldsymbol{\Omega}) = \alpha^\partial(\mathbf{x}, \boldsymbol{\Omega}),$ in \mathcal{O}_-

271 (4.1c) $\bar{\Psi}(\mathbf{x}) := \frac{1}{|\mathcal{S}|} \int_{\mathcal{S}} \Psi(\mathbf{x}, \boldsymbol{\Omega}) \, d\boldsymbol{\Omega},$ in $D,$

272 The dependent variable $\Psi(\mathbf{x}, \boldsymbol{\Omega})$ is called angular intensity or angular flux, and $\bar{\Psi}(\mathbf{x})$ is called scalar
 273 intensity or scalar flux. The coefficient $\sigma^s : D \rightarrow \mathbb{R}_+$ is the scattering cross section and $\sigma^t : D \rightarrow \mathbb{R}_+$
 274 is the total cross section with $\sigma^t \geq \sigma^s$. At some occasions we are also going to use the absorption
 275 cross section $\sigma^a := \sigma^t - \sigma^s$.

276 Our goal is to construct an approximation of (4.1) that is positivity preserving and asymptotic
 277 preserving in the diffusion limit. We also want to make sure that the above properties hold with
 278 grazing incidences and inhomogeneous materials.

279 **4.2. Angular discretization.** To simplify the presentation of the method we use the dis-
 280 crete ordinate technique to do the discretization with respect to the angles. The resulting angular
 281 quadrature is denoted $\{\mu_l, \boldsymbol{\Omega}_l\}_{l \in \mathcal{L}}$ and is assumed to satisfy

282 (4.2) $\sum_{l \in \mathcal{L}} \mu_l = |\mathcal{S}|, \quad \sum_{l \in \mathcal{L}} \mu_l \boldsymbol{\Omega}_l = \mathbf{0}, \quad \sum_{l \in \mathcal{L}} \boldsymbol{\Omega}_l |c \cdot \boldsymbol{\Omega}_l| = \mathbf{0}, \quad \sum_{l \in \mathcal{L}} \mu_l \boldsymbol{\Omega}_l \otimes \boldsymbol{\Omega}_l = \frac{|\mathcal{S}|}{3} \mathbb{I},$

283 for all $c \in \mathbb{R}^3$, where \mathbb{I} is the 3×3 identity matrix. We define $L := \text{card}(\mathcal{L})$. All the simulations
 284 reported in the paper are done with the S_N technique (Gauss-Legendre quadrature along the polar
 285 axis and equi-distributed angles along the azimuth with $\frac{1}{8}N(N+2)$ angles per octant).

286 **4.3. Space discretization.** We are going to use continuous finite elements stabilized with the
 287 continuous interior penalty (CIP) technique (a.k.a. edge stabilization) from Douglas and Dupont
 288 [7] and Burman and Hansbo [5]. The method presented in the paper is not restricted to continuous
 289 elements and CIP though. It can be used with other types of stabilization. For instance, one
 290 can use discontinuous elements (of degree $p \geq 1$) stabilized with upwind numerical flux. One can
 291 also use continuous elements stabilized with methods like Galerkin Least-Squares (GaLS) (or its
 292 SUPG variation), Local Projection Stabilization (LPS), Orthogonal Subscale Stabilization (OSS),
 293 and Subgrid Viscosity (SGV).

294 Let $(\mathcal{T}_h)_{h \in \mathcal{H}}$ be a shape-regular family of unstructured matching meshes exactly covering D . For
 295 simplicity we assume that all the elements are generated from a reference element denoted \widehat{K} . The
 296 geometric transformation mapping \widehat{K} to an arbitrary element $K \in \mathcal{T}_h$ is denoted $T_K : \widehat{K} \rightarrow K$.
 297 We now introduce a reference finite element $(\widehat{K}, \widehat{P}, \widehat{\Sigma})$, which we assume, for simplicity, to be a
 298 Lagrange element. We define the following scalar-valued finite element space:

$$299 \quad (4.3) \quad V_h = \{v \in \mathcal{C}^0(D; \mathbb{R}) \mid v|_K \circ T_K \in \widehat{P}, \forall K \in \mathcal{T}_h\}.$$

300 The global shape functions are denoted by $\{\varphi_i\}_{i \in \mathcal{V}}$. Recall that $V_h = \text{span}\{\varphi_i\}_{i \in \mathcal{V}}$.

301 Given any mesh cell K in \mathcal{T}_h , we denote by h_K the diameter of K and \mathbf{n}_K the outward unit
 302 normal at the boundary of K . We set $h := \max_{K \in \mathcal{T}_h} h_K$. The collection of the mesh faces is denoted
 303 \mathcal{F}_h . The set of interfaces is denoted by \mathcal{F}_h° . The set of boundary faces is denoted by \mathcal{F}_h^∂ . Each
 304 interface F in \mathcal{F}_h° is oriented using a unit normal vector \mathbf{n}_F . Letting K_l, K_r be the two cells sharing
 305 an interface $F \in \mathcal{F}_h^\circ$, we adopt the convention that \mathbf{n}_F points from K_l to K_r . Every boundary face
 306 $F \in \mathcal{F}_h^\partial$ is oriented by the unit normal $\mathbf{n}_F := \mathbf{n}_D$. For all $F \in \mathcal{F}_h$ with $F = \partial K_l \cap \partial K_r$, and all
 307 function w smooth enough to have traces on F , we define the jump of \mathbf{w} across F as

$$308 \quad (4.4) \quad \llbracket w \rrbracket_F := \lim_{K_l \ni \mathbf{y} \rightarrow \mathbf{x}} w(\mathbf{x}) - \lim_{K_r \ni \mathbf{y} \rightarrow \mathbf{x}} w(\mathbf{x}).$$

309 **4.4. Finite element approximation.** First, we define the sesquilinear form associated with
 310 the continuous interior penalty. We start by defining $\sigma_\epsilon^t = \sigma^t + \epsilon \sigma_\epsilon$, $\sigma_\epsilon^s = \sigma^s + \epsilon \sigma_\epsilon$, where $\sigma_\epsilon =$
 311 $\epsilon \sup_{\mathbf{x} \in D} \sigma^t(\mathbf{x})$ and $\epsilon > 0$. The tests reported in the paper are done with $\epsilon = 10^{-10}$. Then, for all
 312 $\psi_h, \phi_h \in V_h$, we set

$$313 \quad (4.5a) \quad s_h(\psi_h, \phi_h) := \varpi \sum_{F \in \mathcal{F}_h^\circ} h_F^2 (\llbracket \nabla \psi_h \rrbracket_{\theta, F}, \llbracket \nabla \phi_h \rrbracket_{\theta, F})_{\mathbf{L}^2(F)},$$

$$314 \quad (4.5b) \quad \llbracket \nabla \phi_h \rrbracket_{\theta, F} := (\theta_r \nabla \phi_h|_{K_l} - \theta_l \nabla \phi_h|_{K_r}) \cdot \mathbf{n}_F,$$

$$315 \quad (4.5c) \quad \theta_r = \frac{\sigma_{\epsilon, l}^t}{\sigma_{\epsilon, l}^t + \sigma_{\epsilon, r}^t}, \quad \theta_l = \frac{\sigma_{\epsilon, r}^t}{\sigma_{\epsilon, l}^t + \sigma_{\epsilon, r}^t},$$

316 In all the simulations reported in the paper the parameters, ϖ , and h_F are defined by

$$317 \quad (4.6) \quad \varpi := \frac{d^2}{(1+p)^4}, \quad h_F = \frac{\frac{1}{2}(|K_l| + |K_r|)}{|F|},$$

318 where d is the space dimension and p is the polynomial degree of the approximation.

319 Next, for all $k \in \mathcal{L}$, we define the bilinear form associated with the operator $\psi \mapsto \boldsymbol{\Omega}_k \cdot \nabla \psi + \sigma^t \psi$
 320 and the bilinear form we use to weakly enforce boundary conditions. For all $k \in \mathcal{L}$, we set

$$321 \quad (4.7) \quad t_k(\psi, \phi) = \int_D (\boldsymbol{\Omega}_k \cdot \nabla \psi_{h, k}(\mathbf{x}) + \sigma^t(\mathbf{x}) \psi_{h, k}(\mathbf{x})) \varphi_i(\mathbf{x}) \, d\mathbf{x},$$

$$322 \quad (4.8) \quad b_k(\psi, \phi) := \sum_{F \in \mathcal{F}_h^\partial} \int_F \frac{1}{2} (|\boldsymbol{\Omega}_k \cdot \mathbf{n}| - \boldsymbol{\Omega}_k \cdot \mathbf{n}) \psi(\mathbf{x}) \phi(\mathbf{x}) \, ds.$$

323 The approximation of the angular flux $\boldsymbol{\psi}_h$ is done in $(V_h)^L$. We set $\boldsymbol{\psi}_h := (\psi_{h, 1}, \dots, \psi_{h, L}) \in$
 324 $V_h \times \dots \times V_h$, with $\psi_{h, k} := \sum_{j \in \mathcal{V}} \Psi_{ik} \varphi_j \in V_h$ for all the angular direction in the quadrature $k \in \mathcal{L}$.

325 Let α_k^∂ be the value of the boundary incidence along the quadrature angle Ω_k . The discrete ordinate
 326 Galerkin approximation of (4.1) consists of seeking $\Psi_h \in (V_h)^L$ so that the following holds for all
 327 $k \in \mathcal{L}$ and all $i \in \mathcal{V}$:

$$328 \quad (4.9) \quad t_k(\psi_{h,k}, \varphi_i) + s_h(\psi_{h,k}, \varphi_i) + b_k(\psi_{h,k}, \varphi_i) \\
 329 \quad \quad \quad = \int_D \sigma^s(\mathbf{x}) \bar{\psi}_h(\mathbf{x}) \varphi_i(\mathbf{x}) \, d\mathbf{x} + \int_D q(\mathbf{x}) \varphi_i(\mathbf{x}) \, d\mathbf{x} + b_k(\alpha_k^\partial, \varphi_i).$$

330 We are also going to make use of the diffusion approximation with weakly enforced Dirichlet
 331 boundary condition. For this purpose, for all $\phi_h, r_h \in V_h$, we set

$$332 \quad (4.10) \quad a(\phi_h, r_h) := \int_D \frac{1}{3\sigma_\epsilon^s(\mathbf{x})} \nabla \phi_h(\mathbf{x}) \cdot \nabla r_h(\mathbf{x}) \, d\mathbf{x} + \int_D \sigma^a(\mathbf{x}) \phi_h(\mathbf{x}) r_h(\mathbf{x}) \, d\mathbf{x} + \frac{1}{4} \int_{\partial D} \phi_h(\mathbf{x}) r_h(\mathbf{x}) \, ds,$$

333 where we recall that $\frac{1}{2|\mathcal{S}|} \int_{\Omega \in \mathcal{S}} (|\Omega \cdot \mathbf{n}| + \Omega \cdot \mathbf{n}) \, ds = \frac{1}{|\mathcal{S}|} \int_{\Omega \cdot \mathbf{n} < 0} |\Omega \cdot \mathbf{n}| \, ds = \frac{1}{4}$.

334 **4.5. Solution method.** There are many solution methods to solve (4.9). The method that
 335 is used does not really matter for the purpose of the paper which we recall is about conservative
 336 limiting not involving computing a low-order solution based on artificial viscosity. The key idea is
 337 that limiting is done after (4.9) is solved. We explain in Appendix A the method that we use for
 338 all the simulations reported in the paper. As the purpose of the paper is just to discuss limiting,
 339 we have adopted a simple source iteration technique preconditioned with a diffusion approximation
 340 and using a minimum residual technique.

341 **5. Numerical illustrations, scalar transport equation.** The objective of this section is
 342 to illustrate the limiting technique proposed in the paper. We start by testing the method on the
 343 scalar advection equation. Examples involving the radiation transport equation are reported in §6.

344 **5.1. Numerical details.** The tests are done with continuous finite elements in one and two
 345 space dimensions. Unless specified otherwise, the simulations realized in one dimension are done on
 346 uniform meshes and those realized in two dimensions are done on unstructured Delaunay meshes.
 347 In all the tests reported below the quadratures are exact for the mass matrix. The index I stands for
 348 the number of degrees of freedom (or gridpoints) of the approximation. Given a nonzero function
 349 $u \in L^1(D)$ and u_h its finite element approximation, we call relative error in the L^1 -norm the
 350 quantity $\|u - u_h\|_{L^1(D)} / \|u\|_{L^1(D)}$.

351 **5.2. 1D smooth solution.** We start by solving a transport problem in one space dimension
 352 with a smooth solution. We let $D = (0, 8)$ and solve

$$353 \quad (5.1) \quad \Omega \partial_x u + \sigma u = q, \quad \text{a.e. } x \in D, \quad u(0) = 0.$$

354 with $\Omega = 1$, $\sigma(x) = 1$, $q(x) = \Omega \pi \sin(\pi x) + \sigma(x)(1 - \cos(\pi x))$. The solution is $u(x) = 1 - \cos(\pi x)$.

355 We test the method with \mathbb{P}_1 , \mathbb{P}_2 , and \mathbb{P}_3 continuous finite elements on a series of meshes. We
 356 compute the relative error in the L^1 -norm. The results are shown in Table 5.1. We observe the
 357 expected convergence rates: 2, 3, and 4 for \mathbb{P}_1 , \mathbb{P}_2 , and \mathbb{P}_3 finite elements, respectively.

358 **5.3. 1D non-smooth solution.** We continue with one-dimensional transport problem with
 359 a non-smooth solution. We consider $D = (0, 1)$ and solve

$$360 \quad (5.2) \quad \Omega \partial_x u + \sigma u = q, \quad \text{a.e. } x \in D, \quad u(0) = 0.$$

Table 5.1: Problem (5.1). \mathbb{P}_1 , \mathbb{P}_2 , and \mathbb{P}_3 continuous finite elements. Relative error in the L^1 -norm.

| \mathbb{P}_1 | | | \mathbb{P}_2 | | | \mathbb{P}_3 | | |
|----------------|------------|------|----------------|------------|------|----------------|------------|------|
| I | L^1 -Err | rate | I | L^1 -Err | rate | I | L^1 -Err | rate |
| 101 | 2.28E-03 | – | 101 | 2.02E-04 | – | 100 | 1.07E-04 | – |
| 201 | 4.61E-04 | 2.32 | 201 | 2.22E-05 | 3.21 | 202 | 6.80E-06 | 3.91 |
| 401 | 1.02E-04 | 2.18 | 401 | 2.67E-06 | 3.07 | 400 | 4.46E-07 | 3.99 |
| 801 | 2.42E-05 | 2.08 | 801 | 3.30E-07 | 3.02 | 799 | 2.81E-08 | 4.00 |
| 1601 | 5.92E-06 | 2.04 | 1601 | 4.12E-08 | 3.01 | 1600 | 1.74E-09 | 4.00 |

361 with $\Omega = 1$, and the scalar fields $\sigma(x)$ and $q(x)$ are piecewise constants and given by

$$362 \quad (5.3) \quad \sigma(x) = \begin{cases} s_1 & x_0 \leq x \leq x_1 \\ s_2 & x_1 < x \leq x_2 \\ s_3 & x_2 < x \leq 1 \end{cases} \quad q(x) = \begin{cases} q_1 & x_0 \leq x \leq x_1 \\ q_2 & x_1 < x \leq x_2 \\ q_3 & x_2 < x \leq 1. \end{cases}$$

363 The exact solution is given by

$$364 \quad (5.4) \quad u(x) = \begin{cases} \frac{q_1}{s_1}(1 - \exp(s_1(x_0 - x))) & x_0 \leq x \leq x_1 \\ u_1 \exp(s_2(x_1 - x)) & x_1 < x \leq x_2 \\ u_2 \exp(s_3(x_2 - x)) + \frac{q_3}{s_3}(1 - \exp(s_3(x_2 - x))) & x_2 < x \leq 1, \end{cases}$$

365 where $u_1 = \frac{q_1}{s_1}(1 - \exp(s_1(x_0 - x_1)))$ and $u_2 = u_1 \exp(s_2(x_1 - x_2))$. In the simulations reported
366 below we use $x_0 = 0$, $x_1 = 0.3$, $x_2 = 0.6$, $s_1 = 1$, $s_2 = 10^3$, $s_3 = 2$, $q_1 = 1$, $q_2 = 0$, $q_3 = 1$.

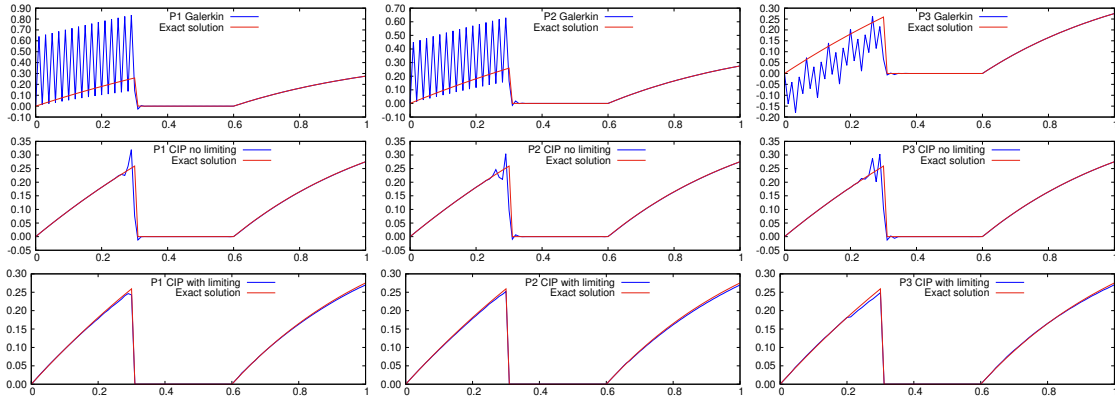


Figure 5.1: Nonsmooth solution (5.2). Left column: \mathbb{P}_1 , 101 dofs. Center column: \mathbb{P}_2 , 101 dofs. Right column: \mathbb{P}_3 , 91 dofs. Top row: Galerkin. Center row: CIP without limiting. Bottom row: CIP with limiting.

367 We test the method with \mathbb{P}_1 , \mathbb{P}_2 , and \mathbb{P}_3 continuous finite elements. We show in Figure 5.1 the
368 graph of the \mathbb{P}_1 , \mathbb{P}_2 , and \mathbb{P}_3 approximation on a mesh composed of 101, 101 and 91 grid points,

369 respectively. The panels in the top row of the figure show the unstabilized Galerkin solution.
 370 oscillations are clearly visible in the interval $(0, x_1)$. The panels in the second row of the figure
 371 show the solution stabilized with the CIP method. Most of the oscillations are gone at the exception
 372 of overshoots localized at the interface located at x_1 . We finally show in the panels of the third row
 373 the results obtained with CIP stabilization and the limiting technique proposed in the paper. The
 374 maximum principle is satisfied.

Table 5.2: Non-smooth solution (5.2). \mathbb{P}_1 , \mathbb{P}_2 , and \mathbb{P}_3 continuous finite elements. Relative error in the L^1 -norm.

| \mathbb{P}_1 | | | \mathbb{P}_2 | | | \mathbb{P}_3 | | |
|----------------|------------|------|----------------|------------|------|----------------|------------|------|
| I | L^1 -Err | rate | I | L^1 -Err | rate | I | L^1 -Err | rate |
| 101 | 1.49E-02 | – | 101 | 1.91E-02 | – | 91 | 1.68E-02 | – |
| 201 | 5.15E-03 | 1.54 | 201 | 2.82E-03 | 2.78 | 181 | 5.88E-03 | 1.53 |
| 401 | 1.56E-03 | 1.72 | 401 | 9.59E-04 | 1.56 | 361 | 1.49E-03 | 1.99 |
| 801 | 3.34E-04 | 2.23 | 801 | 4.12E-04 | 1.22 | 721 | 3.35E-04 | 2.16 |
| 1601 | 1.01E-04 | 1.73 | 1601 | 1.49E-04 | 1.47 | 1441 | 7.11E-05 | 2.24 |

375 The results of the convergence tests are shown in Table 5.2. We observe the expected conver-
 376 gence rate close to 2 for \mathbb{P}_3 elements and between 1.5 and 2 for \mathbb{P}_1 and \mathbb{P}_2 elements. Note that the
 377 convergence rate 2 is optimal since the gradient of the solution is only in BV.

378 **5.4. 2D slip line.** We now consider the two-dimensional unit square $D = (0, 1)^2$ and solve
 379 the problem

$$380 \quad (5.5) \quad \Omega \partial_x u = 0, \quad \text{a.e. } x \in D, \quad u_{\Gamma_1} = 0, \quad u_{\Gamma_2} = 1,$$

381 with $\Omega = (0, 1)^2$, and $\Gamma_1 = \{(0, y) \mid 0 < y \leq 1\}$, $\Gamma_2 = \{(x, 0) \mid 0 \leq x \leq 1\}$. The exact solution is
 382 discontinuous; it exhibits a slip line align the axis $\{x = y, x > 0\}$. The solution is given by

$$383 \quad (5.6) \quad u(x, y) = \begin{cases} 0 & x < y \\ 1 & y \leq x. \end{cases}$$

384 We show in the top three panels of Figure 5.2 the graph of the limited CIP solution obtained
 385 with \mathbb{P}_1 , \mathbb{P}_2 and \mathbb{P}_3 elements on meshes composed of 9535, 9529, and 9541 grid points, respectively.
 386 We show in the bottom three panels of the figure the meshes and 11 isolines $\{0.05, 0.1, \dots, 0.9, 0.95\}$
 387 (4 triangles are shown for each \mathbb{P}_2 cell, and 9 triangles are shown for each \mathbb{P}_3 cell)

388 We report in Table 5.3 the convergence rates for \mathbb{P}_1 , \mathbb{P}_2 , and \mathbb{P}_3 continuous finite elements. We
 389 observe a rate close to 0.73 for linear elements. The convergence rate for \mathbb{P}_2 and \mathbb{P}_3 elements is
 390 equal to 1 which is optimal as the solution is only in BV. This test confirms the near-optimality
 391 of the method for \mathbb{P}_2 and \mathbb{P}_3 elements.

392 **5.5. 2D non smooth problem.** Let $D = (0, 1)^2$. We solve the two-dimensional version of
 393 the problem (5.2),

$$394 \quad (5.7) \quad \Omega \cdot \nabla u + \sigma u = q, \quad \text{a.e. } x \in D, \quad u(0, y) = 0, \quad 0 \leq y \leq 1.$$

Figure 5.2: Slip line problem (5.5). Non-uniform Delaunay meshes. Left: \mathbb{P}_1 , 9535 grid points; Center: \mathbb{P}_2 , 9529 grid points; \mathbb{P}_3 , 9541 grid points.

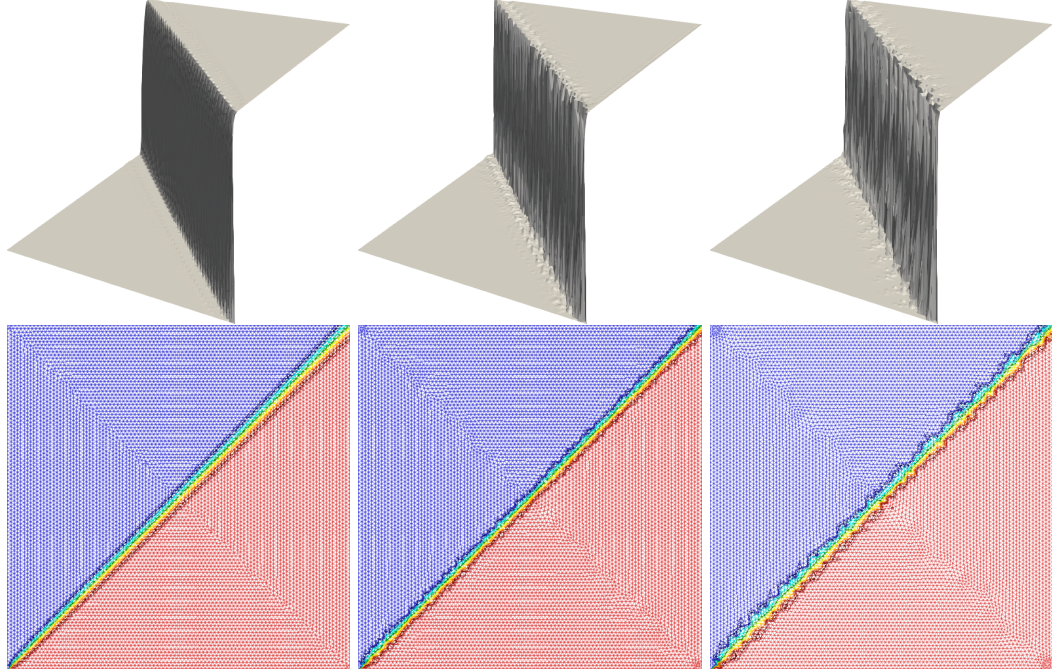


Table 5.3: Problem (5.5). \mathbb{P}_1 , \mathbb{P}_2 , and \mathbb{P}_3 continuous finite elements. Relative error in the L^1 -norm.

| \mathbb{P}_1 | | | \mathbb{P}_2 | | | \mathbb{P}_3 | | |
|----------------|------------|------|----------------|------------|------|----------------|------------|------|
| I | L^1 -Err | rate | I | L^1 -Err | rate | I | L^1 -Err | rate |
| 961 | 5.47E-02 | – | 1681 | 4.14E-02 | – | 961 | 7.09E-02 | – |
| 3721 | 3.45E-02 | 0.68 | 6561 | 2.16E-02 | 0.96 | 3721 | 3.71E-02 | 0.96 |
| 14641 | 2.12E-02 | 0.71 | 25921 | 1.10E-02 | 0.98 | 14641 | 1.89E-02 | 0.98 |
| 58081 | 1.28E-02 | 0.73 | 103041 | 5.57E-03 | 0.99 | 58081 | 9.57E-03 | 0.99 |
| 231361 | 7.75E-03 | 0.73 | 410881 | 2.80E-03 | 0.99 | 231361 | 4.81E-03 | 1.00 |

395 with $\boldsymbol{\Omega} = (1, 0)$, and, with the notation $\boldsymbol{x} := (x, y)$, the scalar fields $\sigma(\boldsymbol{x})$ and $q(\boldsymbol{x})$ given by

396 (5.8)
$$\sigma(x, y) = \begin{cases} s_1 & x_0 \leq x \leq x_1 \\ s_2 & x_1 < x \leq x_2 \\ s_3 & x_2 < x \leq 1 \end{cases} \quad q(x, y) = \begin{cases} q_1 & x_0 \leq x \leq x_1 \\ q_2 & x_1 < x \leq x_2 \\ q_3 & x_2 < x \leq 1. \end{cases}$$

397 The exact solution is given by

$$398 \quad (5.9) \quad u(x, y) = \begin{cases} \frac{q_1}{s_1}(1 - \exp(s_1(x_0 - x))) & x_0 \leq x \leq x_1 \\ u_1 \exp(s_2(x_1 - x)) & x_1 < x \leq x_2 \\ u_2 \exp(s_3(x_2 - x)) + \frac{q_3}{s_3}(1 - \exp(s_3(x_2 - x))) & x_2 < x \leq 1. \end{cases}$$

399 We use the same set of the coefficients $x_0, x_1, x_2, s_1, s_2, s_3$, and q_1, q_2, q_3 as in §5.3 (i.e., $x_0 = 0$,
400 $x_1 = 0.3, x_2 = 0.6, s_1 = 1, s_2 = 10^3, s_3 = 2, q_1 = 1, q_2 = 0, q_3 = 1$).

401 The results of the convergence tests done on uniform meshes are shown in Table 5.4. The
402 convergence rate varies between 1.1 and 2. Recall that the convergence rate 2 is the optimal since
403 the gradient of the solution is only in BV.

Table 5.4: Problem (5.7). $\mathbb{P}_1, \mathbb{P}_2$, and \mathbb{P}_3 continuous finite elements. Relative error in the L^1 -norm.

| \mathbb{P}_1 | | | \mathbb{P}_2 | | | \mathbb{P}_3 | | |
|----------------|------------|------|----------------|------------|------|----------------|------------|------|
| I | L^1 -Err | rate | I | L^1 -Err | rate | I | L^1 -Err | rate |
| 961 | 5.08E-02 | - | 1681 | 2.32E-02 | - | 961 | 4.12E-02 | - |
| 3721 | 2.34E-02 | 1.15 | 6561 | 1.04E-02 | 1.18 | 3721 | 1.96E-02 | 1.10 |
| 14641 | 9.62E-03 | 1.30 | 25921 | 3.74E-03 | 1.49 | 14641 | 7.08E-03 | 1.49 |
| 58081 | 3.18E-03 | 1.61 | 103041 | 1.09E-03 | 1.79 | 58081 | 1.71E-03 | 2.06 |
| 231361 | 8.28E-04 | 1.95 | 410881 | 4.64E-04 | 1.23 | 231361 | 4.18E-04 | 2.04 |

404 **6. Radiation transport.** In this section, we report the tests done on the radiation transport
405 equation using the algorithm described in the paper.

406 **6.1. Numerical details.** The positive- and asymptotic-preserving algorithm defined in Al-
407 gorithms A.1 and A.2 is implemented with continuous finite elements of degree $p \in \{1, 2, 3\}$ on
408 simplices. The meshes in one dimension are uniform. The meshes in two space dimension are non-
409 uniform and composed of triangles. The angular discretization is done with the Gauss-Chebyshev
410 S_N quadrature. In one dimension, the x_1 -component of the angles are the N quadrature points
411 of the Gaussian-Legendre quadrature over $[-1, 1]$ and the weights are the weights of the Gaussian-
412 Legendre quadrature. In two-dimensions we use the standard triangular S_N quadrature. As the
413 x_3 -direction is ignored, there are four quadrants and the total number of angular directions is
414 $\frac{1}{2}N(N + 2)$. Unless specified otherwise, the units are cm for lengths, nb.part./cm²·s·sr for ψ ,
415 nb.part./cm²·s for $\bar{\psi}$, nb.part./cm³·s for q , and cm⁻¹ for the cross sections.

416 **6.2. One-dimensional benchmark tests.** We start by illustrating the performance of the
417 method in one space dimension. We compare the results given by the proposed the method with
418 those given by the unlimited dG1 approximation stabilized by using the upwind flux. We use the
419 angular quadrature S_8 (8 discrete directions in 1D) for all the cases. The angles, characterized
420 by their x_1 -component, are enumerated in increasing order from 1 to 8. The data for the four
421 cases considered here are reported in Table 6.1. In each case we give the length of the domain and
422 the number of zones composing the domain. For each zone we give the values of σ^t, σ^s , and q
423 (constants), and we also give the number of cells composing the zone. The boundary condition for
424 cases 1, 3, and 4 are $\psi_{h|\partial D_-} = 0$ (this is the so-called vacuum boundary condition). We enforce a
425 grazing incidence for case 2; we set $\psi_{h,k|\partial D_-} = 0$ for $k \neq 5, 1 \leq k \leq 8$, and $\psi_{h,5}(0) = 1.0$.

Table 6.1: Data for the one-dimensional test cases.

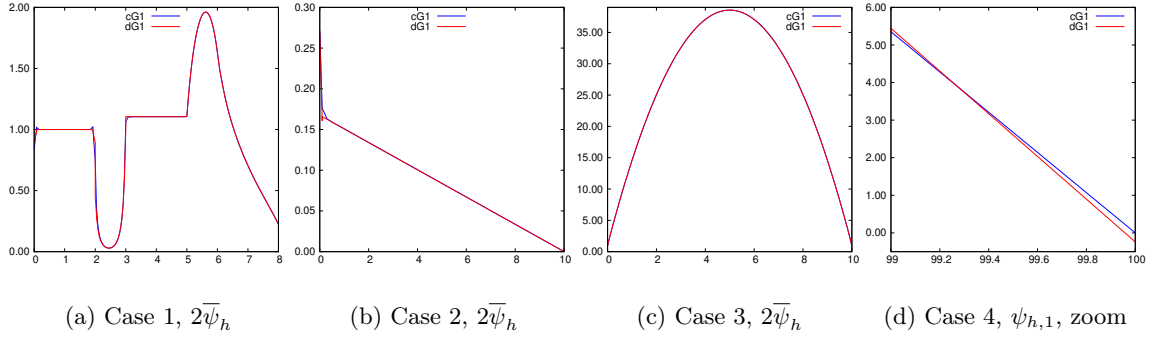
| Case 1 | | #zones | | | | |
|------------|--|--------|-----|-----|-----|-----|
| Length | | 2.0 | 1.0 | 2.0 | 1.0 | 2.0 |
| σ_s | | 0.0 | 0.0 | 0.0 | 0.9 | 0.9 |
| σ_t | | 50.0 | 5.0 | 0.0 | 1.0 | 1.0 |
| q | | 25. | 0.0 | 0.0 | .5 | 0.0 |
| #cells | | 25 | 25 | 25 | 25 | 25 |
| B.C. | | Vac. | | | | |

| Case 2 | | #zones |
|------------|--|-----------------|
| Length | | 10.0 |
| σ_s | | 100.0 |
| σ_t | | 100.0 |
| q | | 0.0 |
| #cells | | 100 |
| B.C. | | $\psi_5(0) = 1$ |

| Case 3 | | #zones |
|------------|--|--------|
| Length | | 10.0 |
| σ_s | | 10.0 |
| σ_t | | 10.0 |
| q | | 0.05 |
| #cells | | 100 |
| B.C. | | Vac. |

| Case 4 | | #zones |
|------------|--|----------|
| Length | | 100.0 |
| σ_s | | 0.099999 |
| σ_t | | 0.1 |
| q | | 0.5 |
| #cells | | 100 |
| B.C. | | Vac. |

Figure 6.1: Comparisons between the present method using cG1 and the upwind dG1 method.



426 The results of the simulations are reported in Figure 6.1. We show in Panels (6.1a)-(6.1c) the
427 total scalar flux, $2\bar{\psi}_h$, obtained with the dG1 approximation (labeled dG1) and with the proposed
428 method (labeled cG1). We observe a fair agreement between the two methods given the number
429 of grid points (recall that dG1 has two times as many grid points as cG1). Panel (6.1d) shows
430 the angular flux $\psi_{h,1}$ for case 4 in the last cell close to the right boundary. For this case the
431 dG1 approximation gives negative values at $x = 100$ on the angular fluxes 1, 2, and 3 (the values
432 are -0.24 , -0.22 , -0.066 , respectively (approximated to 2 digits)). In all the cases the proposed
433 method is always nonnegative.

434 **6.3. Grazing incidences.** We now focus on the second test case discussed in §6.2. A grazing
435 incidence is enforced on the left boundary of the domain (only $\psi_{h,5}(0)$ is nonzero). We also have
436 $\sigma^t h = \sigma^s h = 10$; that is, the diffusive regime is dominant. The conjunction of these two conditions
437 produces a boundary layer as shown in Chandrasekhar [6], Malvagi and Pomraning [21]. Moreover,
438 as established in Theorem 5.4 in [9], convergence of the dG approximation only occurs in a weak
439 norm; more precisely, convergence on $\bar{\psi}_h$ occurs in the Sobolev space $H^s(D)$ only for smoothness
440 indices s strictly less than $\frac{1}{2}$. That is to say, the values of $\bar{\psi}_h$ at the boundary does not converge
441 (recall that boundary traces do not exist when $s < \frac{1}{2}$). Let us for a moment denote by $\alpha^\partial(\Omega, \mathbf{x})$ the
442 boundary data of the problem. Then as observed in Adams [1, Eq. 66] (see also Prop. 3.6 in [9])
443 the leading term of the diffusion expansion of the dG approximation at the boundary is

$$444 \quad (6.1) \quad \bar{\psi}_h(\mathbf{x})|_{\partial D} \approx \frac{1}{2\pi} \int_{\Omega \cdot \mathbf{n} \leq 0} (|\Omega \cdot \mathbf{n}| + \frac{3}{2} |\Omega \cdot \mathbf{n}|^2) \alpha^\partial(\Omega, \mathbf{x}) \, d\Omega,$$

445 which is almost equal to $\frac{1}{2\pi} \int_{\Omega \cdot \mathbf{n} \leq 0} \frac{\sqrt{3}}{2} |\Omega \cdot \mathbf{n}| H(|\Omega \cdot \mathbf{n}|) \alpha^\partial(\Omega, x) d\Omega$, (where H is Chandrasekar’s func-
 446 tion), because $\frac{\sqrt{3}}{2} \mu H(\mu) \approx 0.91\mu + 1.635\mu^2 \approx \mu + \frac{3}{2}\mu^2$ up to a “a few percents” over the interval
 447 $\mu \in [0, 1]$. Hence, although, strictly speaking, the dG approximation is not asymptotic preserving
 448 when there are grazing incidences (unless the mesh or the shape functions are designed to resolve
 449 boundary layers), due to the above observation, it is a common practice in the radiation trans-
 450 port literature to say that the dG approximation with the upwind numerical flux is asymptotic
 451 preserving regardless of whether the boundary data is isotropic or not.

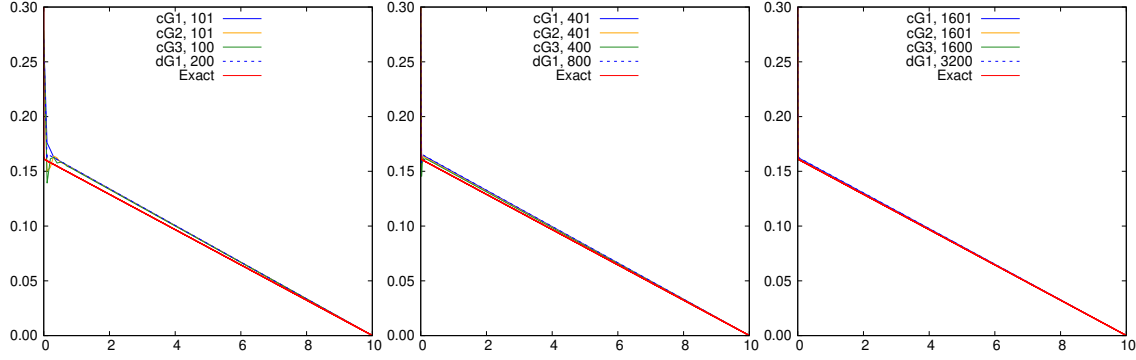


Figure 6.2: Total scalar flux, $2\bar{\psi}_h$, for the grazing problem with cG1, cG2, cG3, and dG1. Left: 100 grid points. Center: 400 grid points. Right: 1600 grid points

452 Note that the above comments regarding dG are independent of the polynomial degree, i.e.,
 453 (6.1) does not depend on p . Note also that the numerical results reported in the second panel of
 454 Figure 6.1 show that the result holds true as well for the stabilized cG1 approximation. The key
 455 here is that in both cases the boundary condition is enforced weakly using the bilinear form b_k
 456 defined in (4.8). We show in Figure 6.2 simulations done with the proposed method using cG p ,
 457 $p \in \{1, 2, 3\}$, on meshes composed on 100 (left panel), 400 (center panel), and 1600 (right panel)
 458 grid points. In each case, the number of cells is adjusted so as to maintain the same number of grid
 459 points for all $p \in \{1, 2, 3\}$. We also report in this figure the results from the dG1 approximation
 460 with 200, 800, and 3200 grid points. The red line labeled “exact” is obtained by using cG3 with
 461 300000 grid points. We observe that, for each number of degrees of freedom all the methods give
 462 almost exactly the same results (regardless of whether the method is dG or cG). We notice also that
 463 with 100 grid points the numerical solution is indeed “a few percents” away from the exact solution,
 464 as claimed in [1, Eq. 66]. The reader is invited to zoom on the right panel where we used 1600
 465 gridpoints: the dG1 solution and the three cG solutions all align on one line that is still slightly
 466 away from the exact solution. This confirms the theoretical result established in Theorem 5.4 in [9]
 467 where it is shown that when grazing incidences are enforced, convergence occurs in weak norms only
 468 and the convergence rate is weak (the convergence rate behaves like $\mathcal{O}(h^{\frac{1}{2}})$ in one space dimension).

469 *Remark 6.1* (Comparison with [11]). The positive and asymptotic preserving method presented
 470 in [11] behaves properly in the diffusion limit with grazing incidences only if the asymptotic bound-
 471 ary value (6.1) (or Chandrasekar’s exact value) is enforced weakly (just enforcing $\alpha^\partial(\Omega, \mathbf{x})$ is not
 472 asymptotic preserving). This is not the case here. The present method properly works in every
 473 regime by just enforcing the boundary conditions using the bilinear form b_k defined in (4.8). \square

474 We finally repeat the above test by adding a vacuum region on the left while still enforcing the
 475 same grazing incidence boundary condition as above. This test is meant to assess the behavior of
 476 the proposed method in the presence of interfaces with vacuum and grazing incidences. We show
 477 the results in Figure 6.3 using 200 grid points in the center panel and 800 grid points in the right
 478 panel. We observe that the proposed method properly behaves. The cG and dG results almost
 479 coincide in both cases. We observe again, that due to the grazing incidence all the methods are
 480 (almost) asymptotic preserving up to “a few percents”.

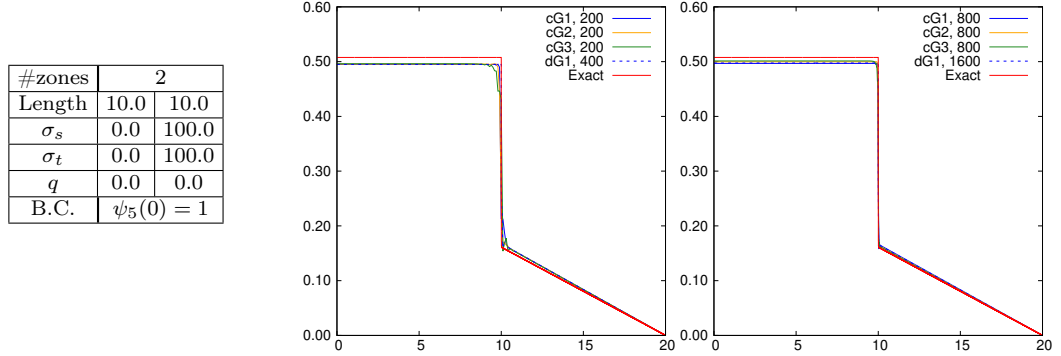


Figure 6.3: Total scalar flux, $2\bar{\psi}_h$, for the grazing problem with a vacuum interface using cG1, cG2, cG3, and dG1. Left: data. Center: 200 grid points. Right: 800 grid points

481 **6.4. Diffusion limit with constant cross sections.** To verify that the method does not
 482 lock in the diffusion regime, we consider the two-dimensional test reported in [11, §5.2.1]. The
 483 computational domain is $D = (0, 1)^2$. The cross sections are constant $\sigma^t = \sigma^s = \frac{1}{\epsilon}$ with $\epsilon > 0$. The
 484 source is isotropic and given by $q(\mathbf{x}) = \epsilon \frac{2}{3} \pi^2 \sin(\pi x_1) \sin(\pi x_2)$. When $\epsilon \rightarrow 0$, the regime becomes
 485 dominated by diffusion. The asymptotic limit in this case is $\psi^0(\mathbf{x}) = \sin(\pi x_1) \sin(\pi x_2)$.

486 We solve (4.1) with the method described in the paper using linear elements ($p = 1$). The tests
 487 are done using five meshes with 140, 507, 1927, 7545, and 29870 Lagrange nodes, respectively; the
 488 corresponding mesh-sizes are approximately $h \approx 0.1, 0.5, 0.025, 0.125$, and 0.00625 . We use the S_6
 489 quadrature (24 angular directions).

Table 6.2: Diffusion limit. Convergence test with respect to the mesh-size and ϵ .

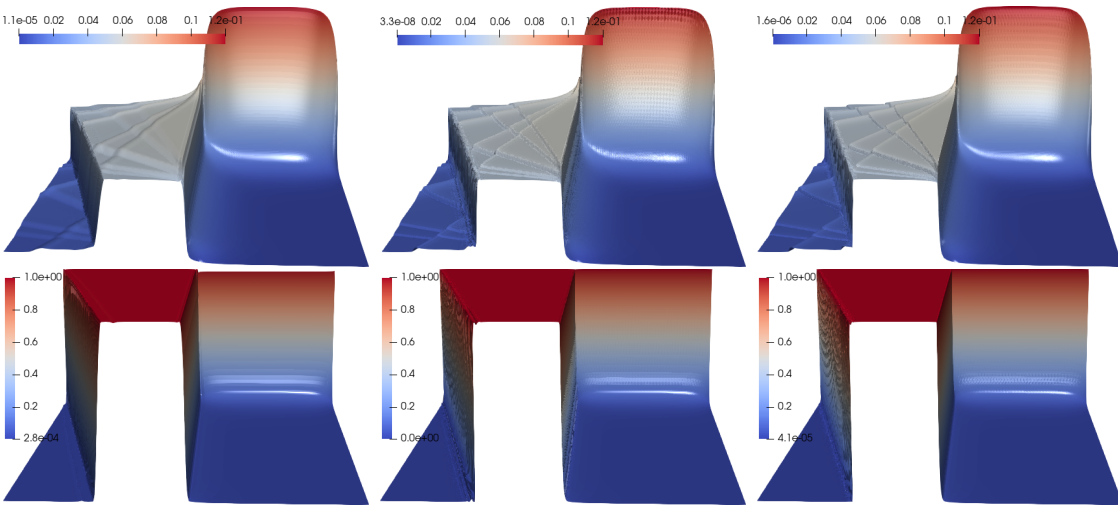
| ϵ | I | rel($\ e\ _{L^2}$) | rate | rel($\ \nabla e\ _{L^2}$) | rate |
|------------|-------|----------------------|-------|-----------------------------|------|
| 10^{-3} | 140 | 3.48E-02 | - | 7.57E-02 | - |
| | 507 | 5.19E-03 | 2.96 | 1.91E-02 | 2.14 |
| | 1927 | 2.35E-03 | 1.18 | 5.88E-03 | 1.77 |
| | 7545 | 2.91E-03 | -0.31 | 2.32E-03 | 1.36 |
| | 29870 | 3.05E-03 | -0.07 | 1.44E-03 | 0.69 |
| 10^{-4} | 140 | 1.52E-02 | - | 3.92E-02 | - |
| | 507 | 3.39E-03 | 2.33 | 1.20E-02 | 1.84 |
| | 1927 | 6.35E-04 | 2.51 | 4.23E-03 | 1.56 |
| | 7545 | 1.82E-04 | 1.83 | 1.19E-03 | 1.86 |
| | 29870 | 2.70E-04 | -0.57 | 3.14E-04 | 1.93 |
| 10^{-5} | 140 | 1.25E-02 | - | 2.24E-02 | - |
| | 507 | 3.16E-03 | 2.13 | 6.98E-03 | 1.81 |
| | 1927 | 7.66E-04 | 2.12 | 2.65E-03 | 1.45 |
| | 7545 | 1.70E-04 | 2.21 | 7.11E-04 | 1.93 |
| | 29870 | 2.81E-05 | 2.62 | 2.38E-04 | 1.59 |
| 10^{-6} | 140 | 1.23E-02 | - | 1.95E-02 | - |
| | 507 | 3.14E-03 | 2.12 | 5.75E-03 | 1.90 |
| | 1927 | 7.84E-04 | 2.08 | 1.98E-03 | 1.60 |
| | 7545 | 1.92E-04 | 2.06 | 7.06E-04 | 1.51 |
| | 29870 | 4.59E-05 | 2.08 | 2.35E-04 | 1.60 |

490 We show in Table 6.2 the results of the test for the following four values of the small parameter
 491 $\epsilon \in \{10^{-3}, 10^{-4}, 10^{-5}, 10^{-6}\}$. We report in this table the relative L^2 -norm and the relative H^1 -

492 semi-norm of the difference between $\bar{\psi}_h$ and the Lagrange interpolant of the asymptotic limit ψ^0 .
 493 We observe that, as proved in [9, Th. 5.3] for the upwind dG1 approximation, the scalar flux $\bar{\psi}_h$
 494 converges optimally to ψ^0 when ϵ is significantly smaller than the mesh-size. The convergence order
 495 is $\mathcal{O}(h^2)$ in the L^2 -norm. It seems that some super-closeness phenomenon occurs in the H^1 -semi-
 496 norm since the rate behaves like $\mathcal{O}(h^{1.5})$. Tests with quadratic and cubic elements demonstrate the
 497 same behavior. These tests are not reported for brevity.

498 **6.5. Reflection effects.** We now reproduce a test case proposed in [11] using exactly the
 499 same finite element grids and angular discretization in order to illustrate that the proposed method
 500 is indeed more accurate than that from the reference.

Figure 6.4: Grazing and reflection effects: Top: scalar intensity $\bar{\psi}_h$. Bottom: first angular intensity, $\psi_{h,1}$. Left, \mathbb{P}_1 , 76230 grid points. Center \mathbb{P}_2 , 303893 grid points. Right: \mathbb{P}_3 , 682990 grid points.



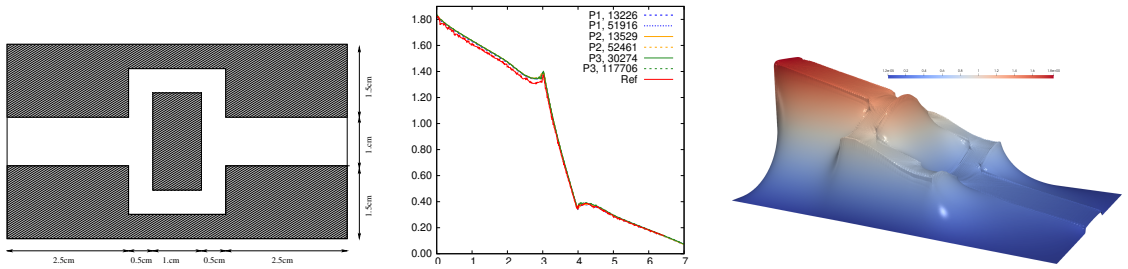
501 We consider the two-dimensional domain $D = (0, 1)^2$ composed of two regions: one that
 502 is optically thick and one without any scattering. The cross sections are distributed as fol-
 503 lows: $\sigma^t(\mathbf{x}) = 100$, $\sigma^s(\mathbf{x}) = 99$ if $x_2 \geq 0.5$ (optically thick and diffusive zone), and $\sigma_t(\mathbf{x}) =$
 504 $\sigma_s(\mathbf{x}) = 0$ if $x_2 \leq 0.5$ (void). The angular approximation is done with the S_6 quadrature (24
 505 angular directions). We enforce a grazing incidence boundary condition on the leftmost bound-
 506 ary: this boundary is illuminated with intensity 1 along the first direction of the quadrature
 507 $\boldsymbol{\Omega}_1 := (0.93802334, 0.25134260, 0.23861919)$ (eight digits truncation). All the other incoming angu-
 508 lar fluxes are set to zero on this boundary. All the incoming angular fluxes are set to zero on
 509 the other three boundaries. The approximation in space for the asymptotic-preserving method is
 510 done on a non-uniform grid composed of 151434 triangles. There are 76230 \mathbb{P}_1 grid points (i.e.,
 511 1 829 520 dofs in total), 303893 \mathbb{P}_2 grid points (i.e., 7 239 432 dofs in total), 682990 \mathbb{P}_3 grid points
 512 (i.e., 16 391 760 dofs in total).

513 The results are shown in Figure 6.4. Comparing these results with what is shown in the two
 514 leftmost panels in Fig. 3 in [11], we observe that the present method is significantly more accurate
 515 than that in [11] while being positivity-preserving and asymptotic preserving. We also notice the
 516 ray effect in the vacuum region $\{x_2 \leq 0.5\}$, which is an artifact of the S_N method (recall that we

517 are just using 24 angular directions on purpose). That the ray effect is so crisply captured clearly
 518 demonstrates that the space approximation is very accurate; one cannot discern any smoothing
 519 induced by numerical dissipation.

520 **6.6. Crooked pipe problem.** We now solve a problem known in the literature as the crooked
 521 pipe problem. We adopt here the setting used in Olivier et al. [22, §7.3]. The geometry of the
 522 problem is shown in the left panel of Figure 6.5, $D = (0, 7) \times (-2, 2)$. In the pipe we have $\sigma^s = 0.2$,
 523 $\sigma^a = 10^{-3}$, $q = 10^{-7}$. The characteristics of the material composing the walls are $\sigma^s = 200$,
 524 $\sigma^a = 10^{-3}$, and $q = 10^{-7}$. The boundary condition is $\Psi(x, y) = 2/|S| = 1/2\pi$ on $\{x = 0, |y| \leq 0.5\}$
 525 and $\Psi(x, y) = 0$ on the rest of the boundary.

Figure 6.5: Crooked pipe problem. Left: setting of the problem. Center: profile of the total scalar flux, $|\mathcal{S}|\bar{\psi}_h$ for various meshes and polynomial degrees. The symbol ‘‘Ref.’’ stands for the data from [22]. Right: total scalar flux distribution (\mathbb{P}_3)



526 We use unstructured triangular Delaunay meshes. The meshes are generated so that the
 527 pipe/wall interface is exactly represented. The computations are done on various meshes and
 528 with various polynomial degrees ranging from \mathbb{P}_1 to \mathbb{P}_3 : \mathbb{P}_1 with 13226 grid points, \mathbb{P}_1 with 51916
 529 grid points, \mathbb{P}_2 with 13529 grid points, \mathbb{P}_2 with 524621 gridpoints, \mathbb{P}_3 with 30274 grid points, and
 530 \mathbb{P}_3 with 117706 grid points.

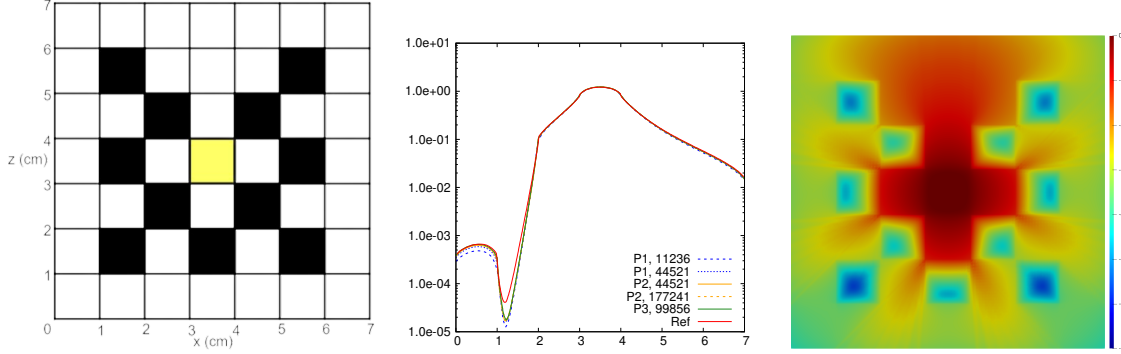
531 We show in the center panel of Figure 6.5, the profile of the total scalar intensity, $|\mathcal{S}|\bar{\psi}_h$, along
 532 the segment $\{0 \leq x \leq 7, y = 0\}$. There are six different simulations. The angular discretization is
 533 done as in [22, §7.3] using the S_{24} angular quadrature ($\frac{1}{2}N(N+2) = 312$ angles). We also report
 534 in this panel the results given in Olivier et al. [22, Fig. 10] (red line). We observe that the results
 535 from the six simulations with the proposed method collapse on a single curve, suggesting that all
 536 the spatial features are resolved. There are slight discrepancies of a few per cents with the results
 537 from [22], but overall the agreement is satisfactory.

538 **6.7. Lattice problem.** We continue with a benchmark test from Peng and McClarren [23,
 539 §5.2] called ‘‘Lattice problem’’ therein. The computational domain is $D = (0, 7)^2$. The material
 540 is organized in a checkerboard fashion. Each elementary region has size 1×1 . The details of the
 541 geometry are shown in the left panel of Figure 6.6. There are 11 purely absorbing regions ($\sigma^t = 10$,
 542 $\sigma^s = 0$, black boxes in Figure 6.6), there are 37 regions that are purely scattering ($\sigma^t = 1$, $\sigma^s = 1$,
 543 white boxes in Figure 6.6), and there is one region with a strong source and scattering material
 544 ($q = 1$, $\sigma^t = 1$, $\sigma^s = 1$, yellow box). The homogeneous Dirichlet boundary condition is enforced
 545 over the entire boundary of the domain.

546 The meshes are generated so that the material interfaces are exactly represented. The compu-
 547 tations are done on three meshes with polynomial degrees ranging from \mathbb{P}_1 to \mathbb{P}_3 . Five different

548 simulations done: \mathbb{P}_1 with 11236 grid points, \mathbb{P}_1 with 44521 grid points, \mathbb{P}_2 with 44521 grid points,
 549 \mathbb{P}_2 with 177241 grid points, and \mathbb{P}_3 with 99856 grid points. The angular quadrature is done with
 550 312 angular directions (this is the S_{24} quadrature).

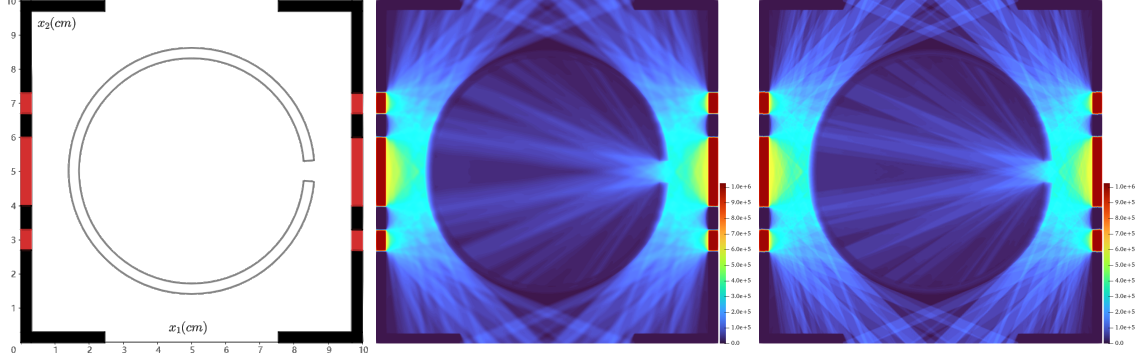
Figure 6.6: Lattice problem. Left: setting of the problem. Center: profile of the total scalar flux, $|\mathcal{S}|\bar{\psi}_h$ for various meshes and polynomial degrees. The symbol “Ref.” stands for the data collected from [23]. Right: total scalar flux distribution in logscale (\mathbb{P}_3 , 99856 grid points)



551 We show in the center panel of Figure 6.6 the profile of the total scalar intensity, $|\mathcal{S}|\bar{\psi}_h$, along the
 552 segment $\{x = 3.5, 0 \leq y \leq 7\}$. We also report in this figure results from Peng and McClarren [23,
 553 Fig. 8(e)]. These results are shown in red and identified with the symbol “Ref.” The representation
 554 is done in log scale along the y-axis. We observe that the agreement between the present method
 555 and the reference results is overall quite satisfactory.

556 **6.8. Hohraum.** We finish with a test that is purely qualitative and loosely inspired from the
 557 hohraum problem in Southworth et al. [27, IV.B]. The objective is to give some feeling on how
 558 the method behaves when solving a somewhat realistic problem. The problem under consideration
 559 is a very simplistic representation of a hohraum device used in the indirect-drive approach of
 560 inertial confinement fusion. The dimensions are not given in the reference, but we use a square
 561 domain $D = (0, 10)^2$ (i.e., 10cm×10cm). These values are significantly larger than those of an
 562 actual hohraum, but as the problem is linear, everything can be recalled by a length scale. The
 563 zero incidence boundary condition is imposed. The thickness of the walls of the cavity is 0.3. The
 564 width of the opening at the top and bottom of the cavity is 5.1. The “spherical” capsule inside the
 565 cavity is centered at (5., 5.). Its internal radius is 3.3 and its external radius is 3.6. The opening
 566 on the right side of the capsule (simulating a filling hole) is a cone with half angle equal to 5° and
 567 vertex located at (5., 5.). The sources are meant to simulate the heating of the wall by
 568 lasers, hence the sources are only located in the walls of the cavity in the region composed of the
 569 points $\mathbf{x} = (x_1, x_2)$ where $(0 \leq x_1 \leq 0.3$ or $9.7 \leq x_1 \leq 10)$ and $(2.7 \leq x_2 \leq 3.3$ or $4 \leq x_2 \leq 6$ or
 570 $6.7 \leq x_2 \leq 7.3)$. The constant value of the source is arbitrarily set to $q = 10^8$. The cross sections
 571 are distributed as follows. “Gold” wall of the cavity: $\sigma^t = 10^2$, $\sigma^s = 2.5$. “Helium filling” around the
 572 spherical capsule: $\sigma^t = 10^{-4}$, $\sigma^s = 10^{-4}$. “Plastic” capsule wall: $\sigma^t = 10$, $\sigma^s = 6$. “Hydrogen fuel”
 573 inside the capsule: $\sigma^t = 10^{-2}$, $\sigma^s = 10^{-2}$. Again, these values are inspired from [27, IV.B] and are
 574 not to be taken as actual values. Note that the problem is heterogeneous. It involves streaming
 575 and diffusive regions.

Figure 6.7: Hohraum problem. Left: geometry of the problem. Center, scalar flux, $\bar{\psi}_h$, \mathbb{P}_1 , 682261 grid points, S_{12} quadrature. Right, scalar flux, $\bar{\psi}_h$, \mathbb{P}_3 , 864754 grid points, S_{12} quadrature.



576 We show in the center and right panels of Figure 6.7 the results of two simulations using
 577 continuous \mathbb{P}_1 and \mathbb{P}_3 elements. The meshsize is approximately 0.0125 for the \mathbb{P}_1 approximation
 578 (682261 grid points) and 0.035 for the \mathbb{P}_3 approximation (864754 grid points). We use the S_{12}
 579 angular quadrature as in [27, IV.B]. This makes 84 directions. The figure shows the scalar flux, $\bar{\psi}_h$,
 580 in both cases. The results look visually similar to what is reported in Figure 3(b) in [27] although
 581 the color scheme is slightly different.

582 Appendix A. Preconditioned source iteration and limiting.

583 We explain here the preconditioned source iteration method that is used in the radiation trans-
 584 port tests reported in the paper. We start with a few definitions to help us make the algorithm
 585 more concise. We define the mapping $\Phi : V_h \rightarrow V_h$ so that for all $p \in V_h$ (not to be confused with
 586 the polynomial degree), $\Phi(p)$ solve the diffusion equation $a(\Phi(p), v) = \int_D pv \, dx$ for all $v \in V$ (notice
 587 that the boundary condition is homogeneous). For all angular quadrature index $k \in \mathcal{L}$, we define
 588 the mapping $\Psi_k^0 : V_h \rightarrow V_h$ so that for all scalar flux $\phi \in V_h$, $\Psi_k^0(\phi)$ solves (4.9) with homogenous
 589 boundary condition (i.e., zero incidence) and zero source term (i.e., $q \equiv 0$). Likewise we define
 590 $\Psi_k^* \in V_h$ so that Ψ_k^* solves (4.9) with the correct boundary condition and the correct source term
 591 q . In summary, $\Phi(p)$, $\Psi_k^0(\phi)$ and Ψ_k^* are defined so that the following holds for all $v \in V_h$:

$$592 \quad (\text{A.1}) \quad a(\Phi(p), v) = \int_D pv \, dx,$$

$$593 \quad (\text{A.2}) \quad t(\Psi_k^0(\phi), v) + s_h(\Psi_k^0(\phi), v) + b_h(\Psi_k^0(\phi), v) = \int_D \sigma^s \phi v \, dx,$$

$$594 \quad (\text{A.3}) \quad t(\Psi_k^*, v) + s_h(\Psi_k^*, v) + b_h(\Psi_k^*, v) = \int_D qv \, dx + b_k(\alpha_k^{\partial}, v).$$

595 We set $\Psi^0(\phi) := (\Psi_1^0(\phi), \dots, \Psi_L^0(\phi)) \in V_h^L$ and $\Psi^* := (\Psi_1^*, \dots, \Psi_L^*) \in V_h^L$. Then $\Psi(\phi) := \Psi^0(\phi) + \Psi^*$
 596 solves (4.9) iff $\overline{\Psi(\phi)} = \phi$. Hence, we have to solve the linear system: Find $\phi \in V_h$ so that

$$597 \quad (\text{A.4}) \quad \phi - \overline{\Psi^0(\phi)} = \overline{\Psi^*}.$$

598 This can be done in a multitude of ways. The simple algorithm we use in the numerical simulations
 599 reported in the paper proceeds as follows. We initialize the algorithm by setting $\Phi^0 = \Phi(q)$. Using

600 Krylov's method to construct the solution, it is natural to define the first search direction to be
 601 the residual of (A.4), i.e., $p^0 = \phi^0 - \overline{\Psi^0(\phi^0)} - \overline{\Psi^*}$. Let ϕ^n (here $n \in \mathbb{N}$ is the loop index) be some
 602 estimate of ϕ . We add the direction induced by the diffusion limit $r^n := \Phi(\sigma^s p^n)$. Then we search
 603 for $\alpha^n, \beta^n \in \mathbb{R}$ so that the new update

$$604 \quad (\text{A.5}) \quad \phi^{n+1} = \phi^n + \alpha^n p^n + \beta^n r^n,$$

605 minimizes $\|\phi^{n+1} - \overline{\Psi(\phi^{n+1})}\|$ where $\|\cdot\|$ is some norm induced by some inner product (\cdot, \cdot) that can
 606 be chosen by the user. As the next search direction is the residual $p^{n+1} = \phi^{n+1} - \overline{\Psi^0(\phi^{n+1})}$, after
 607 some algebraic manipulations we obtain

$$608 \quad (\text{A.6}) \quad p^{n+1} = p^n + \alpha^n (p^n - \overline{\Psi^0(p^n)}) + \beta^n (r^n - \overline{\Psi^0(r^n)}).$$

609 To avoid stalling, which is a standard for steepest descent methods, we enforce p^{n+1} to be orthogonal
 610 to p^n . Setting $dp := p^n - \overline{\Psi^0(p^n)}$ and $dr := r^n - \overline{\Psi^0(r^n)}$, the two constraints on α^n and β^n are then

$$611 \quad (\text{A.7}) \quad \|p^n\|^2 + \alpha^n (p^n, dp^n) + \beta^n (p^n, dr^n) = 0,$$

$$612 \quad (\text{A.8}) \quad \min [\|p^n\|^2 + 2\alpha^n (p^n, dp^n) + 2\beta^n (p^n, dr^n) + \|\alpha^n dp^n + \beta^n dr^n\|^2].$$

613 The solution to this quadratic system is

$$614 \quad (\text{A.9}) \quad \beta^n = \|p^n\|^2 \frac{\|dp^n\|^2 (p^n, dr^n) - (dp^n, dr^n) (p^n, dp^n)}{2(dp^n, dr^n) (p^n, dp^n) (p^n, dr^n) - \|dp^n\|^2 (p^n, dr^n)^2 - \|dr^n\|^2 (p^n, dp^n)^2},$$

$$615 \quad (\text{A.10}) \quad \alpha^n = \frac{-\|p^n\|^2 - \beta^n (p^n, dr^n)}{(p^n, dp^n)},$$

616 and (A.5) gives the next estimate of the solution. At this point one applies the global mass
 617 conserving limiting algorithm (2.6) to $\{\phi^{n+1}\}_{i \in \mathcal{V}}$ with the global mass $\mathcal{M} := \sum_{i \in \mathcal{V}} m_i \phi_i^{n+1}$ and
 618 $\phi_i^{\min} = 0$ (and one can also enforce ϕ_i^{\max} if it happens that the maximum is a priori known). The
 619 algorithm stops when $\|p^{n+1}\|/\|\phi^{n+1}\|$ is smaller than some tolerance.

620 The final and key part of the algorithm is the conservative limiting (local and global). For
 621 each angle $k \in \mathcal{L}$, we proceed as follows: (1) For every $i \in \mathcal{V}$, we use the method of characteristics
 622 explained in §3 to compute the lower and upper bounds on the angular intensity, $\{\Psi_{i,k}^{\min}, \Psi_{i,k}^{\max}\}_{i \in \mathcal{V}}$.
 623 (2) We then apply the local mass conserving algorithm (2.2)–(2.5) to $\{\Psi_{i,k}\}_{i \in \mathcal{V}}$ using the bounds
 624 $\{\Psi_{i,k}^{\min}, \Psi_{i,k}^{\max}\}_{i \in \mathcal{V}}$ computed above. (3) Finally we apply the global mass conserving algorithm (2.6)
 625 to $\{\overline{\Psi}_{i,k}\}_{i \in \mathcal{V}}$ with the global mass $\mathcal{M} := \sum_{i \in \mathcal{V}} m_i \overline{\Psi}_{i,k}$, $\overline{\Psi}_{i,k}^{\min} = 0$ (one can also enforce $\overline{\Psi}_{i,k}^{\max}$ if the
 626 maximum happens to be a priori known).

627 **Acknowledgments.** The first author heartily thanks Dr. Dominic Caron for many fruitful
 628 discussions they had at TAMU from 2020 to 2022 over possible extensions of [11]. These investi-
 629 gations unfortunately did not bear fruits but they eventually led the first author to the strategy
 630 presented in the paper (stabilized Galerkin + characteristics-based limiting). The first author also
 631 thanks the organizers of the 2024 ‘‘Mathematics for neutronics’’ one-day workshop, Feb 24, 2014 at
 632 Lab. J.-L. Lions, Paris, for the stimulating discussions that lead to the completion of this exhaust-
 633 ing project. Finally, the first author thanks M. Adams, J. Morel (TAMU) and T. Bailey (LLNL)
 634 for their indefectible support for this project over the years. The second author sincerely thanks
 635 Dr. Ari Rappaport for his kind help, his patience, and meaningful discussions about programming.

636 **References.**

Algorithm A.1 Preconditioned source iteration**Require:** Tolerance $\delta > 0$, upper bounds $\{\Phi_i^{\max}\}_{i \in \mathcal{V}}$ (optional).

- 1: $n = 0, e = \infty.$ ▷ initialize iteration count and error
- 2: $\phi^0 = \Phi(q), p^0 = \phi^0 - \overline{(\Psi^0(\phi^0) + \Psi^*)} \leftarrow$ solution of (A.3). ▷ Initialize scalar flux and residual
- 3: **while** $e > \delta$ **do** ▷ iteratively update $\Psi^0(\phi)$
 - $dp^n \leftarrow p^n - \overline{\Psi^0(p^n)}$ ▷ set dp^n
 - $r^n \leftarrow \Phi(\sigma^s p^n),$ solution of (A.1) with source $\sigma^s p^n.$ ▷ set r^n
 - $dr^n \leftarrow r^n - \overline{\Psi^0(r^n)}$ ▷ set dr^n
- 4: $\beta^n \leftarrow \frac{\|p^n\|^2(\|dp^n\|^2(p^n, dr^n) - (dp^n, dr^n)(p^n, dp^n))}{2(dp^n, dr^n)(p^n, dp^n)(p^n, dr^n) - \|dp^n\|^2(p^n, dr^n)^2 - \|dr^n\|^2(p^n, dp^n)^2}.$ ▷ optimize β^n
- $\alpha^n \leftarrow \frac{-\|p^n\|^2 - \beta^n(p^n, dr^n)}{(p^n, dp^n)}.$ ▷ optimize α^n
- 5: $p^{n+1} \leftarrow p^n + \alpha^n dp^n + \beta^n dr^n.$ ▷ update p^n
- 6: $\phi^{n+1} \leftarrow \phi^n + \alpha^n p^n + \beta^n r^n.$ ▷ update ϕ^n
- 7: $\mathcal{M} := \sum_{i \in \mathcal{V}} m_i \Phi^{n+1}$ ▷ compute the global mass
 - $\Phi_i^{n+1, \min} = 0$ for all $i \in \mathcal{V}.$ ▷ set lower bound
 - Optional: $\Phi_i^{n+1, \max} \leftarrow \Phi_i^{\max}$ for all $i \in \mathcal{V}.$ ▷ set upper bound
 - $\{\Phi_i^{n+1}\}_{i \in \mathcal{V}} \leftarrow$ (2.6) with above parameters. ▷ post-processing by global limiter
- 8: $e \leftarrow \frac{\|\phi^{n+1} - \phi^n\|}{\|\phi^{n+1}\|}.$ ▷ estimate error at current step
- 9: $n \leftarrow n + 1$ ▷ go to next iteration
- 10: **end while**
- 11: $\{\Psi_{i,k}\}_{i,k \in \mathcal{V} \times \mathcal{L}} \leftarrow \Psi^0(\phi^n) + \Psi^*.$ ▷ Compute final solution
- 12: **return** $\{\Psi_{i,k}\}_{i,k \in \mathcal{V} \times \mathcal{L}}$

Algorithm A.2 Conservative limiting**Require:** Numerical solution $\{\Psi_{i,k}\}_{i,k \in \mathcal{V} \times \mathcal{L}}$, Iteration number it^{\max} .

- 1: **for** $it \in \{1 : it^{\max}\}$ **do** ▷ iteratively apply local limiter
- 2: **for** $(i, k) \in \mathcal{V} \times \mathcal{L}$ **do** ▷ loop on each DoF
- 3: $\Psi_{i,k}^{\max}, \Psi_{i,k}^{\min} \leftarrow$ (3.3) with scattering source defined with $\overline{\Psi}_i.$ ▷ estimate local bounds
- $\Psi_{i,k}^{\max}, \Psi_{i,k}^{\min} \leftarrow$ Relaxation Algorithm A.3. ▷ relax local bounds
- $\Psi_{i,k} \leftarrow$ (2.2)–(2.5) with above bounds. ▷ Local conservative limiting
- 4: **end for**
- 5: **end for**
- 6: $\mathcal{M}_k := \sum_{i \in \mathcal{V}} m_i \Psi_{i,k}$ for all $k \in \mathcal{L}.$ ▷ set global mass for each angular
- $\Psi_{i,k}^{\min} = 0$ for all $(i, k) \in \mathcal{V} \times \mathcal{L}.$ ▷ set lower bounds for global limiter
- Optional: $\Psi_{i,k}^{\max} \leftarrow$ input data, for all $(i, k) \in \mathcal{V} \times \mathcal{L}.$ ▷ set upper bounds for global limiter
- $\{\Psi_{i,k}\}_{i,k \in \mathcal{V} \times \mathcal{L}} \leftarrow$ (2.6), with above bounds. ▷ apply global conservative limiter
- 7: **return** $\{\Psi_{i,k}\}_{i,k \in \mathcal{V} \times \mathcal{L}}$

- 637 [1] M. L. Adams. Discontinuous finite element transport solutions in thick diffusive problems.
638 *Nuclear Science and Engineering*, 137(3):298–333, 2001.
- 639 [2] M. Ancellin, B. Després, and S. Jaouen. Extension of generic two-component VOF interface
640 advection schemes to an arbitrary number of components. *J. Comput. Phys.*, 473:111721, 20,
641 2023.

Algorithm A.3 Relaxation

Require: Bounds $\{u_i^{\max}, u_i^{\min}\}_{i \in \mathcal{V}}$, $\{u_i\}_{i \in \mathcal{V}}$, FEM basis $\{\varphi_i\}_{i \in \mathcal{V}}$, u^{\min} , u^{\max} .

- 1: $\beta_{ij} := \int_D \nabla \varphi_i \cdot \nabla \varphi_j \, d\mathbf{x}$. ▷ set stiffness matrix
- 2: $\alpha_i \leftarrow \frac{\sum_{j \in \mathcal{I}(i)^*} d_{ij} \beta_{ij} (u_i - u_j)}{\sum_{j \in \mathcal{I}(i)^*} d_{ij}}$ for each $i \in \mathcal{V}$. ▷ estimate second-order increment
- 3: **for** $i \in \mathcal{V}$ **do** ▷ tighten bounds by minmod
- 4: $\beta_i \leftarrow \alpha_i$. ▷ initialize relaxation parameter
- 5: **for** $j \in \mathcal{I}(i)^*$ **do** ▷ minmod process
- 6: **if** $\beta_i \alpha_j \leq 0$ **then**
- 7: $\beta_i \leftarrow 0$.
- 8: **else if** $|\beta_i| > |\alpha_j|$ **then**
- 9: $\beta_i \leftarrow \alpha_j$.
- 10: **end if**
- 11: **end for**
- 12: **end for**
- 13: $u_i^{\max} \leftarrow \min(u_i^{\max} + |\beta_i|, u^{\max})$ for each $i \in \mathcal{V}$. ▷ relax upper bound
- 14: $u_i^{\min} \leftarrow \max(u_i^{\min} - |\beta_i|, u^{\min})$ for each $i \in \mathcal{V}$. ▷ relax lower bound
- 15: **return** $\{u_i^{\max}, u_i^{\min}\}_{i \in \mathcal{V}}$

- 642 [3] P. Bochev, D. Ridzal, G. Scovazzi, and M. Shashkov. Formulation, analysis and numerical
643 study of an optimization-based conservative interpolation (remap) of scalar fields for arbitrary
644 Lagrangian–Eulerian methods. *J. Comput. Phys.*, 230(13):5199–5225, 2011.
- 645 [4] P. Bochev, D. Ridzal, and K. Peterson. Optimization-based remap and transport: A divide
646 and conquer strategy for feature-preserving discretizations. *J. Comput. Phys.*, 257:1113–1139,
647 2014.
- 648 [5] E. Burman and P. Hansbo. Edge stabilization for Galerkin approximations of convection-
649 diffusion-reaction problems. *Comput. Methods Appl. Mech. Engrg.*, 193(15-16):1437–1453,
650 2004.
- 651 [6] S. Chandrasekhar. *Radiative Transfer*. Oxford University Press, 1950.
- 652 [7] J. Douglas and T. Dupont. Interior penalty procedures for elliptic and parabolic galerkin
653 methods. In R. Glowinski and J. L. Lions, editors, *Computing Methods in Applied Sciences*,
654 pages 207–216, Berlin, Heidelberg, 1976. Springer Berlin Heidelberg.
- 655 [8] L. Gosse and G. Toscani. An asymptotic-preserving well-balanced scheme for the hyperbolic
656 heat equations. *C. R. Math. Acad. Sci. Paris*, 334(4):337–342, 2002.
- 657 [9] J.-L. Guermond and G. Kanschat. Asymptotic analysis of upwind discontinuous Galerkin
658 approximation of the radiative transport equation in the diffusive limit. *SIAM J. Numer.*
659 *Anal.*, 48(1):53–78, 2010.
- 660 [10] J.-L. Guermond, M. Nazarov, B. Popov, and I. Tomas. Second-order invariant domain pre-
661 serving approximation of the Euler equations using convex limiting. *SIAM J. Sci. Comput.*,
662 40(5):A3211–A3239, 2018.
- 663 [11] J.-L. Guermond, B. Popov, and J. Ragusa. Positive and asymptotic preserving approximation
664 of the radiation transport equation. *SIAM J. Numer. Anal.*, 58(1):519–540, 2020.
- 665 [12] S. Hamilton, M. Benzi, and J. Warsa. Negative flux fixups in discontinuous finite element s_n
666 transport. Saratoga Springs, NY, 2009.
- 667 [13] A. Harten and S. Osher. Uniformly high-order accurate nonoscillatory schemes. I. *SIAM J.*

- 668 *Numer. Anal.*, 24(2):279–309, 1987.
- 669 [14] E. W. Larsen. On numerical solutions of transport problems in the diffusion limit. *Nuclear*
670 *Science and Engineering*, 83(1):90–99, 1983.
- 671 [15] E. W. Larsen, J. E. Morel, and W. F. Miller, Jr. Asymptotic solutions of numerical transport
672 problems in optically thick, diffusive regimes. *J. Comput. Phys.*, 69(2):283–324, 1987.
- 673 [16] K. Lathrop. Spatial differencing of the transport equation: Positivity vs. accuracy. *J. Comput.*
674 *Phys.*, 4(4):475–498, 1969.
- 675 [17] E. E. Lewis and W. F. Miller. *Computational methods of neutron transport*. American Nuclear
676 Society, 1993.
- 677 [18] C. Liu, B. Riviere, J. Shen, and X. Zhang. A simple and efficient convex optimization based
678 bound-preserving high order accurate limiter for cahn–hilliard–navier–stokes system. *SIAM*
679 *Journal on Scientific Computing*, 46(3):A1923–A1948, 2024.
- 680 [19] P. G. Maginot, J. E. Morel, and J. C. Ragusa. A non-negative moment-preserving spatial
681 discretization scheme for the linearized Boltzmann transport equation in 1-D and 2-D Cartesian
682 geometries. *J. Comput. Phys.*, 231(20):6801–6826, 2012.
- 683 [20] P. G. Maginot, J. C. Ragusa, and J. E. Morel. Nonnegative methods for bilinear discontinuous
684 differencing of the sn equations on quadrilaterals. *Nuclear Science and Engineering*, 185(1):
685 53–69, 2017.
- 686 [21] F. Malvagi and G. C. Pomraning. Initial and boundary conditions for diffusive linear transport
687 problems. *J. Math. Phys.*, 32(3):805–820, 1991.
- 688 [22] S. Olivier, W. Pazner, T. S. Haut, and B. C. Yee. A family of independent variable Eddington
689 factor methods with efficient preconditioned iterative solvers. *J. Comput. Phys.*, 473:111747,
690 39, 2023.
- 691 [23] Z. Peng and R. G. McClarren. A sweep-based low-rank method for the discrete ordinate
692 transport equation. *J. Comput. Phys.*, 473:111748, 2023.
- 693 [24] K. Peterson, P. Bochev, and D. Ridzal. Optimization-based, property-preserving algorithm for
694 passive tracer transport. *Computers & Mathematics with Applications*, 159:267–286, 2024.
- 695 [25] R. Sanchez and N. J. McCormick. A review of neutron transport approximations. *Nuclear*
696 *Science and Engineering*, 80(4):481–535, 1982.
- 697 [26] B. Schmidtman, R. Abgrall, and M. Torrilhon. On third-order limiter functions for finite
698 volume methods. *Bull. Braz. Math. Soc. (N.S.)*, 47(2):753–764, 2016.
- 699 [27] B. S. Southworth, M. Holec, and T. S. Haut. Diffusion synthetic acceleration for heterogeneous
700 domains, compatible with voids. *Nuclear Science and Engineering*, 195(2):119–136, 2021.
- 701 [28] B. C. Yee, S. S. Olivier, T. S. Haut, M. Holec, V. Z. Tomov, and P. G. Maginot. A quadratic
702 programming flux correction method for high-order dg discretizations of sn transport. *J.*
703 *Comput. Phys.*, 419:109696, 2020.
- 704 [29] S. T. Zalesak. Fully multidimensional flux-corrected transport algorithms for fluids. *J. Comput.*
705 *Phys.*, 31(3):335–362, 1979.