An Efficient Sweep-based Solver for the S_N Equations on High-Order Meshes[1]

Terry S. Haut,[a] Peter G. Maginot,[b] Vladimir Z. Tomov,[a] Thomas A. Brunner,[b] Teresa S. Bailey[b]

[a]*Center for Applied Scientific Computing, Lawrence Livermore National Laboratory*
[b]*Design Physics Division, Lawrence Livermore National Laboratory*
P.O. Box 808, Livermore, CA
haut3@llnl.gov, maginot1@llnl.gov, tomov2@llnl.gov, brunner6@llnl.gov, bailey42@llnl.gov

INTRODUCTION

We discuss an efficient, graph-based method for solving the discrete ordinates equations discretized with high-order (HO) discontinuous Galerkin (DG) finite elements on HO (curved) meshes in Cartesian geometry such as those generated by the HO finite element Arbitrary Lagrangian-Euleraian hydrodynamics (ALE) code `BLAST` [1].

HO DG discretizations for discrete ordinates (S_N) thermal radiative transfer (TRT) calculations can effectively address three important multi-physics simulation needs:

1. robust coupling of TRT to high-order ALE,
2. reducing the number of spatial unknowns required to solve the TRT equations, and
3. effectively utilizing next-generation, communication-bound computing architectures.

We use "communication-bound" as a catch-all phrase to encompass the increased latency and reduced ratio of memory bandwidth to available floating point operations that arises from architectures with multiple non-uniform memory access levels within a single compute node. Our work builds on the research of Woods and Palmer [2, 3], which demonstrates that HO DG spatial discretizations of the linear Boltzmann equation on HO curved meshes have necessary mathematical properties for TRT discretizations.

Efficient solution of the S_N TRT equations requires the ability to efficiently solve the linear S_N equations. A key component of an efficient linear S_N calculation is the inversion of an advection-type operator for each radiation propagation direction and energy. For unstructured, but straight-edged meshes, it is usually possible to choose an appropriate mesh ordering that results in an advection-type operator matrix that is block lower-triangular [or nearly so]. This allows the inverse of this operator to be efficiently applied on an element-by-element basis in what is called a "transport sweep" (see e.g. [4, 5, 6, 7]). However, constructing efficient S_N transport solvers for HO curved meshes is an open question complicated by the facts that neighboring mesh elements can be both upwind of each other and that HO meshes can be significantly more distorted.

To handle the increased number of upwind dependencies, we extend previous graph-based methods (see e.g. [4, 5, 6, 7]) to HO curved meshes. A novel aspect of our approach is to additionally weight each edge of the graph with the norm of the "outgoing" part of the DG face matrix that is associated with the corresponding mesh element face. This has the effect of directly connecting the graph to not only the mesh topology, but also to the spatial discretization. To construct a quasi-optimal sweep-ordering, we remove edges in a way to make the graph acyclic while minimizing the total weight of edges removed. Our sweep-ordering is quasi-optimal in the sense that the underlying problem, the maximum acyclic subgraph problem, is known to be NP-hard (see e.g. [8]), and we use heuristic methods to compute a solution which is in general not provably optimal.

We explore this graph-theoretic sweep-based linear S_N solver on a HO (cubic) and low-order (LO) meshes generated via hydrodynamics simulations of the triple-point shock problem [9]. Despite the fact that the HO cubic mesh is extremely tangled, our new graph-theoretic based solver requires only 25% more iterations on this HO mesh as compared to the cycle free, LO mesh.

THEORY

Discontinuous Finite Element Equations

We begin with the mono-energetic, steady-state, discrete ordinates, linear Boltzmann equation,

$$\vec{\Omega}_d \cdot \nabla \psi_d + \sigma_t \psi_d = \frac{\sigma_s}{4\pi}\phi + S_d\,, \tag{1a}$$

subject to boundary conditions:

$$\psi_d = \psi_{in,d}(\vec{x}), \quad \vec{x} \in \partial\mathcal{D} \text{ and } \vec{n}(\vec{x}) \cdot \vec{\Omega}_d < 0\,, \tag{1b}$$

where we have assumed isotropic scattering and the validity of the discrete ordinates approximation, $\phi = \sum_{d'=1}^{N_d} \omega_{d'} \psi_{d'}$ In the above, $(\vec{\Omega}_d, \omega_d)$ are pairs of quadrature directions/weights of the angular quadrature set, $d \in [1, N_d]$, such that $\sum_d \omega_d = 4\pi$, $\psi_d(\vec{x})$ is the spatially dependent angular flux, $\sigma_t(\vec{x})$ is the spatially dependent total interaction cross section, $\sigma_s(\vec{x})$ is the spatially dependent total scattering cross section, $\phi(\vec{x})$ is the scalar flux, $S_d(\vec{x})$ is a spatially dependent source in the direction of $\vec{\Omega}_d$, the problem domain is $\mathcal{D}$, and $\vec{n}$ is the outward directed unit normal on the domain boundary $\partial\mathcal{D}$.

We discretize the angular and scalar fluxes within spatial element (zone or cell) z using N_{unk} DG interpolatory polynomial basis functions[2, 3], $b_i(\vec{x})$, per mesh element such that:

$$\psi_d^z(\vec{x}) \approx \widetilde{\psi}_d^z(\vec{x}) = \sum_{i=1}^{N_{unk}} \psi_{d,i}^z b_i(\vec{x})\,, \tag{2}$$

with the scalar flux being defined similarly. Following the Galerkin procedure and assuming upwinding of angular fluxes at mesh element boundaries, discretization of Eqs. (1) leads

[1] Reviewed for release as LLNL-PROC-744912

to a system of equations for all unknown angular fluxes in direction d, Ψ, that can be written as:

$$[\underline{\mathbf{F}}_d + \underline{\mathbf{G}}_d + \underline{\mathbf{R}}_t]\,\Psi_d = \frac{1}{4\pi}\underline{\mathbf{R}}_s\Phi + \vec{S}_d + \vec{B}_d\,. \tag{3}$$

In Eq. (3), we have defined the following vector quantities:

$$\Psi_d^T = \left[\psi^1_{1,d}\,,\ \ldots\ \psi^1_{N_{unk},d}\ldots\psi^z_{i,d}\ldots\psi^{N_z}_{N_{unk},d}\right] \tag{4a}$$

$$\Phi^T = \left[\phi^1_1\,,\ \ldots\ \phi^1_{N_{unk}}\ldots\phi^z_i\ldots\phi^{N_z}_{N_{unk}}\right] \tag{4b}$$

$$S_{\,d,\lceil i,z\rfloor} = \int_{\mathcal{D}} b_i(\vec{x})\,S_d(\vec{x})\,dV \tag{4c}$$

$$\vec{B}_{d,\lceil i,z\rfloor} = \oint_{\partial\mathcal{D}}\left[\vec{\Omega}_d\cdot\vec{n}(\vec{x})\right]\,b_i(\vec{x})\,\psi_{in,d}(\vec{x})\,dA,\quad \vec{\Omega}\cdot\vec{n}<0\,, \tag{4d}$$

assuming

$$\lceil i',z'\rfloor = i' + (z'-1)N_{unk}\,, \tag{5}$$

where N_z is the total number of mesh elements, and an arbitrary indexing of all mesh elements exists, such that $z \in [1, N_z]$. We note that matrices $\underline{\mathbf{G}}_d$ and $\underline{\mathbf{F}}_d$ are quadrature direction dependent. However, from this point forward we will drop the d subscript. Matrices $\underline{\mathbf{G}}$, $\underline{\mathbf{R}}_t$, and $\underline{\mathbf{R}}_s$ in Eq. (3) are strictly block diagonal matrices. For example, $\underline{\mathbf{R}}_t$ is defined as:

$$\underline{\mathbf{R}}_t = \begin{bmatrix} \mathbf{R}_t^1 & & 0 \\ & \ddots & \\ 0 & & \mathbf{R}_t^{N_z} \end{bmatrix}, \tag{6}$$

with $\underline{\mathbf{G}}$ and $\underline{\mathbf{R}}_s$ being defined similarly. Zone local matrices $\mathbf{R}_t^z$, $\mathbf{R}_s^z$, and $\mathbf{G}^z$ are defined as:

$$\mathbf{G}^z_{ij} = -\int_{V_z}\left[\vec{\Omega}\cdot\nabla b_i(\vec{x})\right]\,b_j(\vec{x})\,dV\,, \tag{7a}$$

$$\mathbf{R}^z_{t,ij} = \int_{V_z}\sigma_t(\vec{x})\,b_i(\vec{x})\,b_j(\vec{x})\,dV\,, \tag{7b}$$

$$\mathbf{R}^z_{s,ij} = \int_{V_z}\sigma_s(\vec{x})\,b_i(\vec{x})\,b_j(\vec{x})\,dV\,, \tag{7c}$$

Matrix $\underline{\mathbf{F}}$ in Eq. (3) embodies the upwinding of angular on mesh element faces. Regardless of element indexing and mesh shape $\underline{\mathbf{F}}$, has a block diagonal component. Depending on mesh element indexing and the physical shape of the mesh, $\underline{\mathbf{F}}$ may also have upper, lower, or both upper and lower block components.

Let e be a mesh element in the set, $\mathbb{N}^z$, of mesh elements adjacent to zone z of size N_e. Apriori, any of the N_e individual faces, Γ_{ez}, that exist between zones e and z may have, along some fraction of the face, an inflow condition to mesh element z. As such, the off-diagonal block components of $\underline{\mathbf{F}}$ are:

$$\underline{\mathbf{F}}_{\lceil i,z\rfloor,\lceil j,e\rfloor} = \int_{\Gamma_{ez}}\mathcal{A}(\vec{\Omega}_d,\vec{x})\,b_i^{(z)}(\vec{x})b_j^{(e)}(\vec{x})\,dA\,, \tag{8}$$

where

$$\mathcal{A}(\vec{\Omega}',\vec{x'}) = \begin{cases} \vec{\Omega}'\cdot\vec{n}_z(\vec{x'}) & \vec{\Omega}'\cdot\vec{n}_z(\vec{x'})<0 \\ 0 & \text{otherwise} \end{cases}, \tag{9}$$

and $\vec{n}_z(\vec{x'})$ is the outward directed, relative to mesh element z, unit normal of face Γ_{ez} at position $\vec{x'}$. Finally, the diagonal blocks of $\underline{\mathbf{F}}$ are a result of the outflow from mesh element z:

$$\underline{\mathbf{F}}_{\lceil i,z\rfloor,\lceil j,z\rfloor} = -\oint_{\partial z}\mathcal{A}(-\vec{\Omega}_d,\vec{x})\,b_i^{(z)}(\vec{x})\,b_j^{(z)}(\vec{x})\,dA\,, \tag{10}$$

where ∂z denotes the surface of mesh element z.

Iterative Solution Methodology

In this work, source iteration with transport sweeps [4, 5, 6, 7, 10],

$$[\underline{\mathbf{F}} + \underline{\mathbf{G}} + \underline{\mathbf{R}}_t]\,\Psi_d^{(n+1)} = \frac{1}{4\pi}\underline{\mathbf{R}}_s\Phi^{(n)} + \vec{S}_d + \vec{B}_d \tag{11a}$$

$$\Phi^{(n+1)} = \sum_d^{N_d}\omega_d\Psi_d^{(n+1)}\,, \tag{11b}$$

is used to iteratively solve Eq. (3), where n is the iteration index. Often there exists some ordering of mesh elements such that $\underline{\mathbf{F}}$ is lower block triangular; in this case, the left hand side operator of Eq. (11a), $\mathcal{L}$,

$$\mathcal{L} = [\underline{\mathbf{F}} + \underline{\mathbf{G}} + \underline{\mathbf{R}}_t]\,, \tag{12}$$

is strictly lower block triangular, permitting the calculation of Ψ_d mesh element by mesh element. However, on non-orthogonal meshes it is not always possible to order elements in such a manner that $\underline{\mathbf{F}}$ is lower block triangular, hence some information must be lagged [10]. In such cases, the mesh is said to contain "cycles" of mesh elements. In the presence of mesh cycles, any indexing of mesh elements results causes $\underline{\mathbf{F}}$ to have a non-zero block lower triangular component, $\underline{\mathbf{L}}$, and a strictly upper block triangular component, $\underline{\mathbf{U}}$, such that $\underline{\mathbf{F}} = \underline{\mathbf{L}} + \underline{\mathbf{U}}$.

In this work we choose to use a Gauss-Seidel type iteration to account for the action of $\underline{\mathbf{U}}$ in the source iteration process, such that we are solving:

$$[\underline{\mathbf{L}} + \underline{\mathbf{G}} + \underline{\mathbf{R}}_t]\,\Psi_d^{(n+1)} = \frac{1}{4\pi}\underline{\mathbf{R}}_s\Phi^{(n)} - \underline{\mathbf{U}}\Psi_d^{(n)} + \vec{S}_d + \vec{B}_d\,, \tag{13}$$

Graph Algorithm

We extend use of a graph-theoretic approach to finding a quasi-optimal transport sweep ordering for HO curved meshes. In particular, given Eq. (13), our goal is to choose a mesh ordering that minimizes the norm of $\underline{\mathbf{U}}$. To do so, consider the directed, weighted graph whose vertices z_1 and z_2 correspond to mesh elements. There is an edge (z_1, z_2) in the graph if mesh element z_1 is upwind of element z_2, and the weight w_{z_1,z_2} for this edge corresponds to the Frobenius norm of the "outgoing face matrix", $\mathbf{F}_{z_1,z_2}$ in the notation of Eq. (10). For high-order meshes, it is possible to have mesh elements z_1 and z_2 that are both upwind of each other, in which case the graph contains both edges (z_1, z_2) and (z_2, z_1).

Given this weighted graph, we would like to determine a set of edges to remove in order to make the resulting graph acyclic while simultaneously minimizing the total weight of the removed edges. This is known as determining a minimum feedback arc set (FAS). Determining a minimal FAS is known to be NP-hard in general (see e.g. [8]). However, there exist effective and efficient, scaling linearly with the number of mesh elements, heuristic methods for determining a "good" FAS. In this work, we use the heuristic algorithm of Eades [8] and other functions defined within the `igraph` library [11].

To determine our mesh element sweep ordering, we first determine all strongly connected components (SCCs) of the

graph [12]. Then, we use the fact that a globally minimum FAS can be determined by independently computing on each SCC a minimal FAS. Since most SCC involve only a few elements, this often allows the computation of a minimal FAS without approximation. In particular, if the number of vertices in the individual SCC is less than a fixed number, say 10, the FAS problem is solved exactly using integer programming. If the number of vertices in the individual SCC is larger than this constant, we determine an (approximate) minimum FAS using the heuristic algorithm of [8]. Once such a set of edges to remove is determined, a topological ordering of the resulting subgraph results in a mesh element ordering that lags degrees of freedom across the mesh element face associated with a removed edge.

RESULTS

To demonstrate our new method, we contrive a linear transport problem and impose this problem on hydrodynamics meshes generated from simulations of the "triple point" problem [9]. Materials and their respective macroscopic absorption cross sections (unrelated to the density in the resulting ALE simulation) are shown in their initial positions in Fig. 1. The

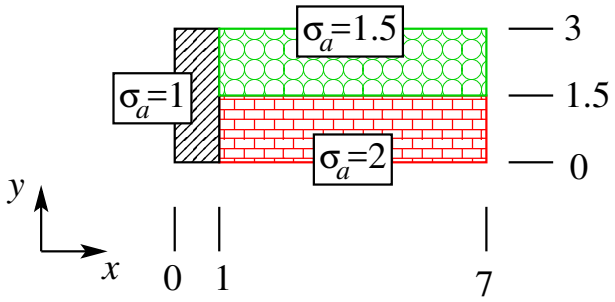

Fig. 1. Material cross sections for the triple point problem.

scattering cross section is a constant $\sigma_s = 1$ throughout the entire problem. The problem is driven by an isotropic source,

$$S_d(x,y) = 1 + \sin\left[(2x+y)^2\right], \tag{14}$$

with a constant inflow boundary condition of unity for all discrete ordinate directions. An S_2 level symmetric quadrature is used to discretize the problem in angle. We consider a total of three meshes, generated from various hydrodynamic simulations each using 336 mesh elements:

- Mesh #1: linear mesh elements from a purely Lagrange simulation [13], shown in Fig. 2,
- Mesh #2: cubic mesh elements from a purely Lagrange simulation [13], shown in Fig. 3, and
- Mesh #3: cubic mesh elements derived from an ALE simulation [1], shown in Fig. 4.

For all meshes the angular flux is spatially discretized using cubic DG basis functions.

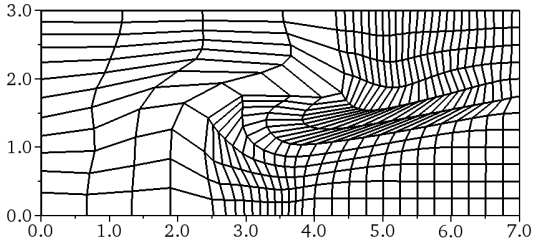

Fig. 2. Mesh #1: linear element Lagrange mesh.

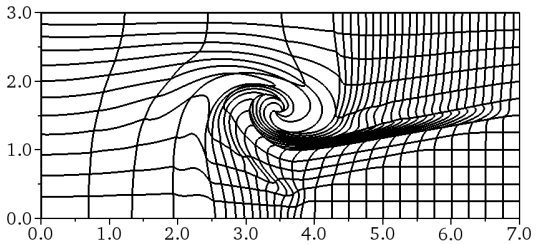

Fig. 3. Mesh #2: cubic element Lagrange mesh.

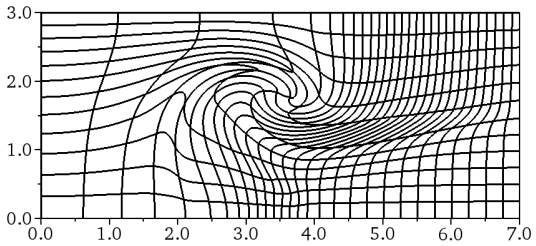

Fig. 4. Mesh #3: cubic element ALE mesh.

TABLE I. Number of strongly connected components and the size of each for Mesh #1-#3.

	Angle	Number of SCC	Elements in Each SCC	Total Edges Removed
Mesh #1	$\vec{\Omega}_1$	0	N/A	0
	$\vec{\Omega}_2$	0	N/A	0
	$\vec{\Omega}_3$	0	N/A	0
	$\vec{\Omega}_4$	0	N/A	0
Mesh #2	$\vec{\Omega}_1$	3	[2, 5, 88]	78
	$\vec{\Omega}_2$	3	[3, 7, 60]	53
	$\vec{\Omega}_3$	0	[3, 7, 60]	52
	$\vec{\Omega}_4$	0	[2, 5, 88]	73
Mesh #3	$\vec{\Omega}_1$	1	[83]	62
	$\vec{\Omega}_2$	2	[3, 35]	34
	$\vec{\Omega}_3$	0	[3, 35]	28
	$\vec{\Omega}_4$	0	[83]	69

Prevalence of Mesh Cycling

We now verify the assumption that HO hydrodynamics creates meshes with greater numbers of mesh cycles involving

a larger number of mesh elements in each mesh cycle. In Table I, we give the number of mesh elements in each SCC of Mesh #1-#3 for quadrature directions $\vec{\Omega}_1 = [\gamma, \gamma]$, $\vec{\Omega}_2 = [-\gamma, \gamma]$, $\vec{\Omega}_3 = [\gamma, -\gamma]$, and $\vec{\Omega}_4 = [-\gamma, -\gamma]$, with $\gamma = \sqrt{1/3}$.

Iterative Effectiveness of Graph Algorithm

In Figure Fig. 5, we plot the fixed point iteration residuals $\left\|\bar{\Psi}^{(n+1)} - \bar{\Psi}^{(n)}\right\|_\infty$, on a log scale as a function of the iteration index n, assuming $\Psi^{(0)} = 0$, where we have defined $\bar{\Psi}^T = [\Psi_1 \dots \Psi_{N_d}]$. As expected, the source iteration converges most rapidly for the cycle-free LO mesh, and most slowly for Mesh #2 which had the largest SCCs. The convergence rate for Mesh #3, the HO ALE mesh, lies between that of the cycle-free LO mesh and the highly distorted Lagrangian HO mesh. Since ALE is generally described as "relaxing" the Lagrangian mesh, we expect that an ALE mesh should have fewer and smaller magnitude SCC than a Lagrange mesh with the same number of elements and finite element order.

Significantly, despite the large increase in the number of upwind dependencies that need to be lagged for Mesh #2 (see Table I), the fixed-point iteration only needs about 25% more iterations than the cycle-free mesh.

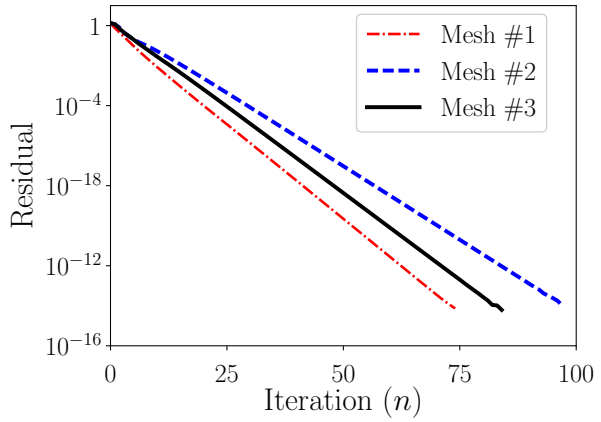

Fig. 5. Fixed point iteration residuals versus iteration count.

CONCLUSIONS

We have explored a sweep-based fixed-point iteration scheme for solving a high-order DG discretization of the linear Boltzmann discrete ordinates equations on high-order curved meshes. Our approach is based on a graph-theoretic formulation for obtaining a quasi-optimal transport sweep ordering. In future work, we plan to compare the sweep-graph generated by the given choice of graph node/edge weightings to methods which account for additional physics properties (mesh element optical thickness), methods which only account for the existence of upwind/downwind dependencies, and methods which do not use graph heuristics to break mesh cycles; explore the use of different graph-based heuristics for determining the sweep ordering; explore element aggregation on small strongly connected components; and examine our results when used within the context of a Krylov wrapped solve.

ACKNOWLEDGMENTS

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344 and was supported by the LLNL-LDRD Program under Project No. 18-ERD-002.

REFERENCES

1. R. W. ANDERSON, V. A. DOBREV, T. V. KOLEV, R. N. RIEBEN, and V. Z. TOMOV, "High-Order Multi-Material ALE Hydrodynamics," *SIAM Journal on Scientific Computing*, **40**, *1*, B32–B58 (2018).
2. D. WOODS, T. A. BRUNNER, and T. S. PALMER, "High-Order Finite Elements S_N transport in $X-Y$ Geometry on Meshes with Curved Surfaces," *Transactions of the American Nuclear Society*, **114**, 377–380 (2016).
3. D. WOODS, *High-Order Finite Elements S_N Transport in $X-Y$ Geometry on Meshes with Curved Surfaces in the Thick Diffusion Limit*, Master's thesis, Oregon State University (2016).
4. W. MCLENDON, B. HENDRICKSON, S. PLIMPTON, and L. RAUCHWERGER, "Finding Strongly Connected Components in Parallel Transport Sweeps," in "SPAA 2001," (2001), pp. 328–329.
5. S. J. PLIMPTON, B. HENDRICKSON, S. P. BURNS, W. MCLENDON, and L. RAUCHWERGER, "Parallel S_N Sweeps on Unstructured Grids: Algorithms for Prioritization, Partitioning, and Cycle Detection," *Nuclear Science and Engineering*, **150**, *3*, 267–283 (2005).
6. S. D. PAUTZ, "An Algorithm for Parallel Sn sweeps on Unstructured Meshes," *Nuclear Science and Engineering*, **140**, 111–136 (2002).
7. S. D. PAUTZ and T. S. BAILEY, "Parallel Deterministic Transport Sweeps of Structured and Unstructured Meshes with Overloaded Mesh Decompositions," *Nuclear Science and Engineering*, **185**, *1*, 70–77 (2017).
8. P. EADES, X. LIN, and W. F. SMYTH, "A Fast and Effective Heuristic for the Feedback Arc Set Problem," *Information Processing Letters*, **47**, *6*, 319–323 (1993).
9. M. KUCHARIK, R. V. GARIMELLA, S. P. SCHOFIELD, and M. J. SHASHKOV, "A Comparative Study of Interface Reconstruction Methods for Multi-Material ALE Simulations," *Journal of Computational Physics*, **229**, 2432–2452 (2010).
10. T. A. WAREING, J. M. MCGHEE, J. E. MOREL, and S. D. PAUTZ, "Discontinuous Finite Element S_N Methods on Three-Dimensional Unstructured Grids," *Nuclear Science and Engineering*, **138**, 256–268 (2001).
11. G. CSARDI and T. NEPUSZ, "The *igraph* Software Package for Complex Network Research," `http://igraph.org` (2006).
12. R. TARJAN, "Depth-First Search and Linear Graph Algorithms," *SIAM J. on Computing*, **1**, *2*, 146–160 (1972).
13. V. DOBREV, T. KOLEV, and R. RIEBEN, "High-Order Curvilinear Finite Element Methods for Lagrangian Hydrodynamics," *SIAM Journal on Scientific Computing*, **24**, B606–B641 (2012).