Seventh International Conference on Computational Fluid Dynamics (ICCFD7), Big Island, Hawaii, July 9-13, 2012

ICCFD7-1602

A Semi-Unstructured Multi-Block Fourth-Order Energy-Stable Weighted Essentially Non-Oscillatory Finite Difference Scheme*

C.C. Otis**,†
Corresponding author: collin.otis@gmail.com

** University of Pittsburgh, Pittsburgh, PA, USA.
† Spectral Energies, LLC, Dayton, OH, USA.

Abstract: A new methodology is devised and demonstrated for accurately capturing discontinuities in multi-block finite difference simulations of hyperbolic partial differential equations. The fourth-order energy-stable weighted essentially non-oscillatory (ESWENO) finite difference scheme on closed domains is combined with simultaneous approximation term (SAT) weak interface and boundary conditions. Smoothness of grid-spacing across subdomain interfaces is not required. WENO stencil-biasing is truncated near subdomain boundaries and only collocated interface points are communicated between neighboring subdomains. The methodology is demonstrated for significant jumps in grid-spacing across subdomain interfaces. Results are presented for the linear scalar hyperbolic wave equation in one and two dimensions and the Euler equations in one and two dimensions. It is demonstrated that this methodology allows strong discontinuities to be passed across subdomain interfaces without significant distortion. Moreover it is demonstrated that the methodology provides stable and accurate results even when large differences in the grid-spacing exist, whereas strong imposition of the interface conditions causes noticeable oscillations. Weak subdomain interdependence, low subdomain-to-subdomain message-passing overhead, and ease of local grid refinement make the new methodology promising for scalable massively-parallel simulations and complex geometries.

Keywords: Multi-Block Finite-Difference Schemes, Boundary Conditions, High-Order Numerical Methods, Shock-capturing, Computational Fluid Dynamics.

1 Introduction

High-order weighted essentially non-oscillatory (WENO) methods have become popular for simulations where delicate physical behavior exists in the presence of strong discontinuities.[1–4] The high-order nature of such schemes allow for accurate resolution of detailed physics while the stencil-biasing mechanics prevent numerical oscillations around discontinuities, thus maintaining numerical stability. The simulation of sound generation in a shock-vortex interaction[5,6] is an example computational fluid dynamics (CFD) application where WENO methods are well-suited. While WENO schemes are very robust in practice, they do not satisfy a summation by parts (SBP) convention;[7,8] therefore, a general stability proof cannot be derived.[9]

In contrast to classical WENO schemes, the recently-developed energy-stable WENO (ESWENO) scheme satisfies SBP and is time-stable in an L_2 sense. The ESWENO methodology was first introduced by Yamaleev and Carpenter in Ref. [9] where the theoretical foundation and a third-order implementation is developed. They later developed a basic framework for developing ESWENO schemes of arbitrary order.[10] Numerical examples were presented for systems of hyperbolic equations in one dimension that demonstrate the stability

*Work performed in part while visiting the Computational AeroSciences Branch, NASA Langley Research Center, Hampton, VA.

and convergence benefits of ESWENO schemes over fixed-stencil schemes and the classical WENO scheme of Jiang and Shu. These seminal ESWENO works developed in the context of a periodic domain were followed by boundary closures in the finite-domain fourth-order ESWENO work of Fisher et al., making ESWENO accessible for numerical simulations of a wider array of problems.[11,12] The fourth-order finite-domain scheme is denoted ESWENO 3-4-3 in reference to fourth-order interior accuracy and third-order accurate boundary closures. The global accuracy converges to fourth-order. Fisher et al. demonstrated the ESWENO 3-4-3 scheme in simulations of a two-dimensional chemically-reacting supersonic hydrogen-air spatially developing mixing layer. The purpose of the present work is to build upon previous ESWENO works by extending and demonstrating the ESWENO 3-4-3 scheme for multi-block domains.

A major challenge in the application of high-order numerical schemes such as ESWENO to problems of practical interest is in generating appropriate finite-difference grids. Grids must be smoothly-varying and match boundaries with high-order accuracy. In even moderately complex geometries, mapping such a smooth mesh to complicated boundaries can be extremely time consuming, if not impossible altogether. If such a mesh *can* be generated, the grid resolution is not necessarily tuned to to the problem physics. Excess grid points may exist in some regions while grid resolution may be lacking in key areas such as boundary and shear layers. Local grid refinement is possible; however, smoothness requirements dictate a smooth transition to a coarse mesh in the remaining domain, at the cost of excess grid resolution in some areas. In highly-complex geometries, mapping boundaries with a single curvilinear mesh is not feasible. Such geometries arise often in practical problems such as gas-turbine combustors and aircraft airframes; they are often handled with unstructured finite volume techniques at the sacrifice of high-order accuracy.

Multi-block meshes can alleviate the difficulties above. The spatial domain is decomposed into a number of smaller subdomains, each of which can be more easily mapped with a boundary-fitted curvilinear mesh. The subdomains are patched together at interfaces using interpolation functions. In cases with many subdomains, interpolation schemes can consume a significant portion of compute time. While this approach works when the simulation variables are smooth functions, the presence of discontinuities in the flow presents a unique challenge. A shock-capturing scheme such as WENO can be used on the subdomain interior in order to sufficiently resolve the shock; however, the lack of stencil-biasing and the imposition of interpolated interface conditions cause significant distortion of the shock at the boundary. Additionally, long-time energy-stability is difficult to ensure in such simulations.

The literature concerning WENO shock capturing in a multi-block setting is sparse. Sebastian and Shu demonstrated a multi-block WENO scheme that utilized Lagrange interpolation at subdomain interfaces.[13] Costa and Don developed a hybrid spectral-WENO scheme for multi-block domains.[14] Recently, Chao et al. demonstrated a multi-block hybrid compact-WENO scheme that utilized fixed stencil finite difference schemes at subdomain boundaries and WENO stencil-biasing within the interior.[15] To my knowledge, no literature exists for ESWENO schemes on multi-block domains.

In this work, I develop and demonstrate a multi-block ESWENO scheme and demonstrate its ability to capture strong shocks. The globally fourth-order finite-domain ESWENO 3-4-3 scheme is used. Each subdomain is mapped with an independent curvilinear mesh; the only requirement is that neighboring subdomains have collocated grid points along the interface surface. Stencil-biasing is preserved at all possible points; however, it is truncated near the subdomain boundaries where only collocated interface points are communicated between neighboring subdomains. Only points lying directly on an interface must be passed to the neighboring processor; no finite difference stencils overlap the interface. The simultaneous approximation term (SAT) interface closure is used to close interface and global domain boundaries.[16] Each subdomain is self-contained, in that at the beginning of each time-step, it contains all the information necessary to move the governing equations forward in time. Interface conditions are then applied a posteriori as an SAT penalty source term. This method allows for the subdomains to be loosely coupled. The independence afforded by the scheme makes it attractive for application in massively-parallel, distributed-memory simulations where the low communication overhead and loose coupling are expected to provide strong scalability. The scheme is shown to be stable for linear conservation laws with continuous solutions. One- and two-dimensional numerical examples are presented for systems of linear and nonlinear hyperbolic equations on multi-block domains.

2 Background

2.1 Energy Estimates

2.1.1 The Continuous Problem

Consider the linear scalar hyperbolic wave equation on a finite (non-periodic) domain

$$\begin{aligned} \frac{\partial v}{\partial t} + \frac{\partial f}{\partial x} &= 0, \quad f = av, \quad x \in \Omega, \quad \Omega = [A, B], \\ v(x,0) &= v_0(x), \\ v(A,t) &= g(t), \end{aligned} \tag{1}$$

where v is the continuous solution, a is a positive constant, and $v_0(x)$ and $g(t)$ are bounded continuous functions. To obtain an energy estimate, Eq. (1) is multiplied by v and integrated over the domain, which yields the continuous energy estimate

$$\frac{d}{dt}\|v\|_{L_2}^2 = -a\left[v^2(B,t) - g^2(t)\right] \tag{2}$$

where $\|\cdot\|_{L_2}$ is the L_2 norm. Equation (2) admits a solution that is stable in time ($\frac{d}{dt}\|v\|_{L_2}^2 \leq 0$). Note that Eq. (2) is obtained by assuming that $\frac{\partial f}{\partial x}$ satisfies integration by parts (IBP).

2.1.2 The Discrete Problem

Our goal is to obtain a pth-order accurate finite difference approximation for the continuous spatial derivative in Eq. (1),

$$\frac{\partial \boldsymbol{f}}{\partial x} = \boldsymbol{D} \cdot \boldsymbol{f} + O(\Delta x^p), \tag{3}$$

such that an energy estimate analogous to Eq. (2) can be obtained. In Eq. (3), $\boldsymbol{f}$ represents a projection of v onto a uniform N-point discretization of the domain Ω, $\boldsymbol{D}$ denotes a discrete derivative operator, and Δx denotes the constant grid-spacing of the discretization. Herein, bold variables (e.g. $\boldsymbol{D}$) represent discrete multicomponent arrays (vectors, matrices, etc.) and non-bold variables (e.g. x, x_i, t, etc.) represent scalars or continuous variables. It is well-established that in order to facilitate an energy estimate, $\boldsymbol{D}$ must satisfy summation by parts (SBP),[7,8] the discrete analog to IBP. SBP is embodied by the constraints

$$\boldsymbol{D} = \boldsymbol{P}^{-1}(\boldsymbol{Q} + \boldsymbol{R}) \quad ; \quad \boldsymbol{Q} + \boldsymbol{Q}^T = Diag[-1, 0, ..., 0, 1], \tag{4}$$

$$\boldsymbol{R} = \boldsymbol{R}^T \quad ; \quad \boldsymbol{v}^T \boldsymbol{R} \boldsymbol{v} \geq 0, \quad \boldsymbol{v} \neq 0, \tag{5}$$

$$\boldsymbol{P} = \boldsymbol{P}^T \quad ; \quad \boldsymbol{v}^T \boldsymbol{P} \boldsymbol{v} > 0, \quad \boldsymbol{v} \neq 0, \tag{6}$$

that is, $\boldsymbol{Q}$ almost skew-symmetric, $\boldsymbol{R}$ symmetric positive semidefinite (SPS), and $\boldsymbol{P}$ symmetric positive definite (SPD). To facilitate construction of an SPS matrix, define the dissipation matrix, $\boldsymbol{R}$, as

$$\boldsymbol{R} = \boldsymbol{\Lambda}_0 + \boldsymbol{\Delta}\boldsymbol{\Lambda}_1 \left[\boldsymbol{\Delta}\right]^T + \boldsymbol{\Delta}\left[\boldsymbol{\Delta}\right]^T \boldsymbol{\Lambda}_2 \boldsymbol{\Delta}\left[\boldsymbol{\Delta}\right]^T + \boldsymbol{\Delta}\left[\boldsymbol{\Delta}\right]^T \boldsymbol{\Delta}\boldsymbol{\Lambda}_3 \left[\boldsymbol{\Delta}\right]^T \boldsymbol{\Delta}\left[\boldsymbol{\Delta}\right]^T \tag{7}$$

where the $\boldsymbol{\Lambda}_i$ are diagonal SPS matrices of appropriate size

$$\boldsymbol{\Lambda}_e = Diag\left[\lambda_1, ..., \lambda_N\right] \quad ; \quad \lambda_j \geq 0, \quad j = 1, N \quad ; \quad e = 2p, \quad p = 0, 1 \tag{8}$$

$$\boldsymbol{\Lambda}_o = Diag\left[\lambda_1, ..., \lambda_N\right] \quad ; \quad \lambda_j \geq 0, \quad j = 0, N \quad ; \quad o = 2p+1, \quad p = 0, 1 \tag{9}$$

and $\mathbf{\Delta}$ is the $N+1 \times N$ differencing matrix

$$\mathbf{\Delta} = \begin{bmatrix} -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & \ddots & \ddots & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 \end{bmatrix}. \tag{10}$$

These forms result in a semi-discrete energy equation that yields stability.[12]

3 Non-Oscillatory Finite Difference Schemes

3.1 Discretization

Discretize the domain of Eq. (1) with two interdigitated grids:

$$\boldsymbol{x} = [x_1 = A, x_2, \ldots, x_N = B] \tag{11}$$

$$\bar{\boldsymbol{x}} = [\bar{x}_0 = A, \bar{x}_1, \ldots, \bar{x}_N = B]. \tag{12}$$

The finite difference grid-points $\boldsymbol{x}$ are denoted *solution-points* because they carry the solution data. Each solution-point, x_i, lies in a control volume bounded on the left and right by *flux-points* $\bar{x}_{i-1}$ and $\bar{x}_i$, respectively. The flux-points, $\bar{\boldsymbol{x}}$ are denoted as such because a numerical flux will be calculated at each of these points to facilitate conservative calculation of the derivative. Herein, data located at flux-points will carry the overbar notation (e.g. $\bar{\boldsymbol{f}}$) while data located at solution-points will be represented by variables without an overbar (e.g. $\boldsymbol{f}$). The flux- and solution-points are collocated at domain boundaries. The solution-points are equally-spaced (at least in computational space), while the flux-points may be unevenly-spaced between solution points.

3.2 WENO

The conventional fourth-order WENO finite difference scheme[17,18] for Eq. (1) is written in semi-discrete form as

$$\frac{\partial u_i}{\partial t} + \frac{\overline{f}_i - \overline{f}_{i-1}}{\Delta x} = 0 \tag{13}$$

where $\bar{f}_i$ is the WENO flux given by

$$\bar{\boldsymbol{f}} = \sum_r \bar{w}^{(r)} \bar{\boldsymbol{f}}^{(r)} \tag{14}$$

where $\bar{\boldsymbol{f}}^{(r)} = \boldsymbol{\mathcal{I}}^{(r)} \boldsymbol{f}$, $r \in \{L, C, R\}$ are second-order fluxes obtained by interpolating data from the solution-points to the flux-points for the three candidate stencils S^L, S^C, and S^R (Left, Center, Right). As described in Ref. [12], the interpolation operators $\boldsymbol{\mathcal{I}}^{(r)}$ are $N+1 \times N$ matrices that interpolate data from solution-points to flux-points.

The nonlinear weight functions, $\bar{w}^{(r)}$, embody the stencil-biasing mechanics of the WENO scheme. The classical weight functions of Jiang and Shu are defined

$$\bar{w}^{(r)} = \frac{\bar{\alpha}^{(r)}}{\sum_r \bar{\alpha}^{(r)}}, \quad \bar{\alpha}^{(r)} = \frac{\bar{d}^{(r)}}{(\epsilon + \bar{\beta}^{(r)})^2} \tag{15}$$

where $\bar{d}^{(r)}$ are the target weights that force the WENO scheme to asymptote to a target central difference scheme in smooth regions of the flow. The smoothness indicators, $\bar{\beta}^{(r)}$, are given by

$$\bar{\beta}_i^L = (f_i - f_{i-1})^2, \quad \bar{\beta}_i^C = (f_{i+1} - f_i)^2, \quad \bar{\beta}_i^R = (f_{i+2} - f_{i+1})^2 \tag{16}$$

and the parameter ϵ is usually set to 10^{-6} to prevent a zero denominator as recommended in Ref. [17]. The

WENO scheme using the nonlinear weights of Jiang and Shu does not follow SBP convention, thus an energy estimate is not straightforward.

3.3 ESWENO 3-4-3

As shown in Ref. [9], the traditional WENO scheme of Jiang and Shu does not satisfy SBP; therefore, a general energy estimate is not readily obtainable for the scheme. The ESWENO scheme involves a modification of the WENO weight functions such that the derivative operator satisfies SBP and an energy estimate is possible. Furthermore an artifical dissipation term is added to the derivative operator to maintain stability. In this work, we consider the finite-domain ESWENO 3-4-3 scheme, which uses a fourth-order approximation in the domain interior and inward-biased-third order approximation near boundaries. As discussed in Refs. [12, 19, 20], this scheme is globally fourth-order.

The finite-domain ESWENO 3-4-3 scheme is formed by constructing a derivative operator that permits the WENO stencil-biasing framework of Eq. (36) and satisfies the SBP requirements of Eqs. (4)-(6).

The grid-points are uniformly distributed with grid spacing $\delta x = x_i - x_{i-1}$, while the flux-points are nonuniformly distributed with grid-spacing $\bar{\delta}_x = \bar{x}_i - \bar{x}_{i-1}$. In this work the grid suggested in Ref. [12] is used:

$$\bar{x} = \left[A, \left(A + \frac{43\delta x}{144}\right), \left(A + \frac{61\delta x}{36}\right), \left(A + \frac{349\delta x}{144}\right), \left(A + \frac{7\delta x}{2}\right), \ldots, \left(B - \frac{7\delta x}{2}\right), \left(B - \frac{349\delta x}{144}\right), \left(B - \frac{61\delta x}{36}\right), \left(B - \frac{43\delta x}{144}\right), B\right] \tag{17}$$

The ESWENO differentiation operator has the form

$$\boldsymbol{D} = \boldsymbol{P}^{-1}\boldsymbol{\Delta}\sum_r \bar{w}^{(r)}\boldsymbol{\mathcal{I}}^{(r)} \tag{18}$$

where $r = L, C, R$ are the three candidate stencils. Enacting the differentiation operator on the flux, $\boldsymbol{f}$ yields

$$\boldsymbol{D}\boldsymbol{f} = \boldsymbol{P}^{-1}\boldsymbol{\Delta}\sum_r \bar{w}^{(r)}\bar{\boldsymbol{f}}^{(r)} = \boldsymbol{P}^{-1}\boldsymbol{\Delta}\sum_r \bar{w}^{(r)}\boldsymbol{\mathcal{I}}^{(r)}\boldsymbol{f}. \tag{19}$$

The weight functions are defined

$$\bar{w}^{(r)} = \frac{\bar{\alpha}^{(r)}}{\sum_r \bar{\alpha}^{(r)}}, \quad \bar{\alpha}^{(r)} = d^{(r)}\left(1 + \frac{\bar{\tau}}{(\epsilon + \bar{\beta}^{(r)})^2}\right) \tag{20}$$

with stencil biasing parameters

$$\bar{\tau}_i = (-f_{i-1} + 3f_i - 3f_{i+1} + f_{i+2})^2, \quad 2 \le i \le N-2 \tag{21}$$

$$\bar{\tau}_i = (-f_i + 3f_{i+1} - 3f_{i+2} + f_{i+3})^2, \quad i = 1 \tag{22}$$

$$\bar{\tau}_i = (-f_i + 3f_{i-1} - 3f_{i-2} + f_{i-3})^2, \quad i = N-1. \tag{23}$$

The smoothness indicators $\bar{\beta}^{(r)}$ remain the same as in WENO; however, extra stencils (S^{LL} and S^{RR}) and subsequent smoothness indicators

$$\bar{\beta}_i^{LL} = (f_{i-1} - f_{i-2})^2, \quad \bar{\beta}_i^{RR} = (f_{i+3} - f_{i+2})^2 \tag{24}$$

are needed to facilitate 3rd order stencil-biasing mechanics near the boundaries. We ensure that the downwind (DW) stencil weight does not overtake the central or upwind weights by applying the modification

$$\bar{\beta}_i^{\mathrm{DW}} = \left(\frac{1}{3}\sum_r \left[\bar{\beta}_i^{(r)}\right]^4\right)^{\frac{1}{4}} \tag{25}$$

to the downwind smoothness indicator. A block-norm $\boldsymbol{P}$ of the form

$$\boldsymbol{P} = \delta x \begin{pmatrix} \boldsymbol{P}_0 & 0 & 0 \\ 0 & \boldsymbol{I} & 0 \\ 0 & 0 & \boldsymbol{P}_0^{PT} \end{pmatrix}, \quad \boldsymbol{P}_0 = \begin{pmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{12} & p_{22} & p_{23} & p_{24} \\ p_{13} & p_{23} & p_{33} & p_{34} \\ p_{14} & p_{24} & p_{34} & p_{44} \end{pmatrix} \tag{26}$$

is used and the derivative operator takes the form

$$\boldsymbol{Q} = \begin{pmatrix} Q_0 & Q_d & 0 \\ -Q_d^T & Q_l & Q_d \\ 0 & -Q_d^T & -(Q_0)^{PT} \end{pmatrix}, \quad \boldsymbol{Q}_0 = \begin{pmatrix} -\frac{1}{2} & q_{12} & q_{13} & q_{14} \\ -q_{12} & 0 & q_{23} & q_{24} \\ -q_{13} & -q_{23} & 0 & q_{34} \\ -q_{14} & -q_{24} & -q_{34} & 0 \end{pmatrix}, \quad \boldsymbol{Q}_d = \begin{pmatrix} 0 & \dots & & \\ 0 & \dots & & \\ -\frac{1}{12} & 0 & \dots & \\ \frac{8}{12} & -\frac{1}{12} & 0 & \dots \end{pmatrix} \tag{27}$$

where PT denotes the per-symmetric transpose, $P_{ij} = P_{[N-(i-1)][N-(j-1)]}$. The matrix $\boldsymbol{P}$ and target operator $\boldsymbol{D}$ can be found in Ref. [12].

The distinction of the ESWENO scheme is its energy-stabilization term that ensures energy-stability in an L_2 sense. The baseline WENO scheme

$$\boldsymbol{D}_{weno} = \boldsymbol{P}^{-1} \left(\boldsymbol{Q} + \boldsymbol{R}_{weno} \right) \tag{28}$$

does not guarantee energy-stability: elements of R_{weno} may take negative values such that R_{weno} is not SPS. Stability is achieved by adding a dissipative operator, $\boldsymbol{R}_{es}$, to ensure that the total dissipation operator, $\boldsymbol{R} = \boldsymbol{R}_{weno} + \boldsymbol{R}_{es}$, is SPS. The ESWENO scheme is specified

$$\boldsymbol{D} = \boldsymbol{P}^{-1} \left(\boldsymbol{Q} + \boldsymbol{R}_{weno} + \boldsymbol{R}_{es} \right). \tag{29}$$

The stabilization operator $\boldsymbol{R}_{es}$ is determined by enforcing the condition that the total dissipation operator $\boldsymbol{R}$ is SPS and the entire scheme remains design-order. Expand $\boldsymbol{R}_{weno}$ as[9]

$$\boldsymbol{P}^{-1}\boldsymbol{R}_{weno} = \boldsymbol{\Lambda}_0 + \boldsymbol{\Delta}\boldsymbol{\Lambda}_1 \left[\boldsymbol{\Delta}\right]^T + \boldsymbol{\Delta} \left[\boldsymbol{\Delta}\right]^T \boldsymbol{\Lambda}_2 \boldsymbol{\Delta} \left[\boldsymbol{\Delta}\right]^T + \boldsymbol{\Delta} \left[\boldsymbol{\Delta}\right]^T \boldsymbol{\Delta}\boldsymbol{\Lambda}_3 \left[\boldsymbol{\Delta}\right]^T \boldsymbol{\Delta} \left[\boldsymbol{\Delta}\right]^T \tag{30}$$

and assume a similar form for $\boldsymbol{R}_{es}$

$$\boldsymbol{P}^{-1}\boldsymbol{R}_{es} = \hat{\boldsymbol{\Lambda}}_0 + \boldsymbol{\Delta}\hat{\boldsymbol{\Lambda}}_1 \left[\boldsymbol{\Delta}\right]^T + \boldsymbol{\Delta} \left[\boldsymbol{\Delta}\right]^T \hat{\boldsymbol{\Lambda}}_2 \boldsymbol{\Delta} \left[\boldsymbol{\Delta}\right]^T + \boldsymbol{\Delta} \left[\boldsymbol{\Delta}\right]^T \boldsymbol{\Delta}\hat{\boldsymbol{\Lambda}}_3 \left[\boldsymbol{\Delta}\right]^T \boldsymbol{\Delta} \left[\boldsymbol{\Delta}\right]^T, \tag{31}$$

Guarantee that $\boldsymbol{R}$ is SPS by smoothly enforcing positivity

$$[\lambda_j]_i + [\hat{\lambda}_j]_i \geq 0, \quad \forall\ i, j. \tag{32}$$

Achieve this by constructing the components of $\hat{\boldsymbol{\Lambda}}_j$ as

$$[\hat{\lambda}_j]_i = \frac{1}{2} \left(\sqrt{[\lambda_j]_i^2 + \delta_i^2} - [\lambda_j]_i \right). \tag{33}$$

Design-order accuracy is preserved if the constraint

$$\delta_1 \leq O\left(\delta x^3\right), \quad \delta_2 \leq O\left(\delta x^2\right) \tag{34}$$

is satisfied. The matrices $\boldsymbol{\Lambda}_i$ can be found in Ref. [12]. Specification of the dissipative energy-stabilization flux

$$\bar{\psi} = \left[\hat{\boldsymbol{\Lambda}}_1 \left[\boldsymbol{\Delta}\right]^T + \left[\boldsymbol{\Delta}\right]^T \hat{\boldsymbol{\Lambda}}_2 \boldsymbol{\Delta} \left[\boldsymbol{\Delta}\right]^T + \left[\boldsymbol{\Delta}\right]^T \boldsymbol{\Delta}\hat{\boldsymbol{\Lambda}}_3 \left[\boldsymbol{\Delta}\right]^T \boldsymbol{\Delta} \left[\boldsymbol{\Delta}\right]^T \right] \boldsymbol{f} \tag{35}$$

allows the energy-stabilization operator to be combined with the WENO operator, yielding the combined flux form:

$$\boldsymbol{f}_x = \boldsymbol{D}\boldsymbol{f} = \boldsymbol{P}^{-1}\boldsymbol{\Delta} \left(\bar{\boldsymbol{f}} + \bar{\psi} \right). \tag{36}$$

4 SAT Penalty Procedure

The SAT penalty procedure was first introduced by Carpenter et al. for boundary closures.[21] It was later extended to interface closures.[16] Since that work, it has been successfully used for boundary and interface closures in a myriad of high-order numerical schemes.[22–25] Both boundary and interface conditions are imposed using SAT in this work. Henceforth, the term *interface* shall generally refer to both subdomain interfaces *and* domain boundaries, as their SAT treatment is identical. A boundary is simply treated as a neighboring subdomain whose data on $\partial\Omega$ (domain boundary) is known and whose interior data is unknown.

The SAT methodology is a *weak* penalty procedure in that instead of strictly enforcing interface data by requiring the values of the solution at the interface x_i to be identical in the two neighboring subdomains, the values of the solution at these interfaces are penalized to pull the respective interface values together weakly. The advantage of such a weak imposition of the interface condition is that it allows the global finite difference scheme to maintain SBP and, consequently, stability. Design-order accuracy is also assured.

We begin by requiring that neighboring subdomains have collocated interface points. Denoting by $\boldsymbol{u}_L$ and $\boldsymbol{u}_R$ discrete solutions to Eq. (1) on the left and right subdomains, respectively, the semi-discrete form of Eq. (1) using the SAT interface penalty is

$$\frac{\partial \boldsymbol{u}_L}{\partial t} + a\boldsymbol{P}_L^{-1}\boldsymbol{Q}_L\boldsymbol{u}_L = \sigma_1 \boldsymbol{P}_L^{-1}\boldsymbol{e}_L\left[\boldsymbol{u}_L|_{x=x_i} - \boldsymbol{u}_R|_{x=x_i}\right] \tag{37a}$$

$$\frac{\partial \boldsymbol{u}_R}{\partial t} + a\boldsymbol{P}_R^{-1}\boldsymbol{Q}_R\boldsymbol{u}_R = \sigma_3 \boldsymbol{P}_R^{-1}\boldsymbol{e}_R\left[\boldsymbol{u}_R|_{x=x_i} - \boldsymbol{u}_L|_{x=x_i}\right] \tag{37b}$$

where x_i is the spatial location of the interface, $e_L = [0, ..., 0, 1]^T$, and $e_R = [1, 0, ..., 0]^T$. As described in Ref. [16], Eq. (37) is stable and preserves design-order accuracy if

$$\sigma_1 \leq \frac{a}{2} \tag{38}$$

$$\sigma_3 = \sigma_1 - a. \tag{39}$$

The SAT interface penalty provides subdomain connectivity for the multi-block formulation described in the next section.

5 Multi-Block Formulation

5.1 One-Dimensional Baseline Formulation

The formulation is first presented in the context of the continuous, one-dimensional linear scalar wave equation on the global domain $x \in [A, B]$:

$$\begin{aligned} \frac{\partial v}{\partial t} + \frac{\partial F}{\partial x} &= 0, \quad F = av, \quad x \in \Omega, \quad \Omega = [A, B] \\ v(x, 0) &= v_0(x), \\ v(A, t) &= g_L(t), \text{ if } a > 0, \\ v(B, t) &= g_R(t), \text{ if } a < 0 \end{aligned} \tag{40}$$

where a is the constant wave speed and g_L and g_R denote boundary conditions on the left and right sides of the global domain, respectively. The domain is partitioned into K subdomains. Subdomain $k \in \{1, 2, \ldots, K\}$ is discretized with N_k grid-points on the local domain $\Omega_k = [x_L, x_R] \subset \Omega$ and $x \in \Omega_k$,

$$\boldsymbol{x} = \left[x_1 = x_L, x_2, \ldots, x_{N_k} = x_R\right], \tag{41}$$

and we seek the discrete solution

$$\boldsymbol{u} = \left[u_1, u_2, \ldots, u_{N_k}\right]. \tag{42}$$

The first and last points in each subdomain lie on the subdomain boundary, such that there are two collocated points at each subdomain interface. Each subdomain utilizes an SBP-satisfying differentiation scheme $\boldsymbol{D} =$

$\boldsymbol{P}^{-1}\boldsymbol{Q}$, which may be different on each subdomain. This leads to the following semi-discrete form:

$$\boldsymbol{u}_t + a\boldsymbol{P}^{-1}\boldsymbol{Q}\boldsymbol{u} = (\sigma - \frac{1}{2})a\boldsymbol{P}^{-1}\left[\mathtt{I}_{a>0}\left(u_{L,k} - u_{L,n_L}\right)\boldsymbol{e}_1 - \left(1 - \mathtt{I}_{a>0}\right)\left(u_{R,k} - u_{R,n_R}\right)\boldsymbol{e}_{N_k}\right] \tag{43}$$

where the RHS represents the SAT interface penalty to account for interface conditions. In Eq. (43), the left and right interface values on subdomain k are denoted $u_{L,k}$ and $u_{R,k}$, respectively; corresponding interface values on its neighbors with partition numbers n_L and n_R are denoted u_{L,n_L} and u_{R,n_R}. The vectors

$$\begin{array}{rcl} \boldsymbol{e}_1 & = & [1, 0, \ldots, 0]^T \\ \boldsymbol{e}_{N_k} & = & [0, \ldots, 0, 1]^T \end{array} \tag{44}$$

have length N_k and are used to restrict the penalty term to only the collocated interface points, although a non-diagonal ESWENO 3-4-3 $\boldsymbol{P}$-matrix will effectively smear the effect of the boundary penalty across multiple points near the interface. The indicator function

$$\mathtt{I}_{a>0} = \begin{cases} 0 & : a \leq 0 \\ 1 & : a > 0 \end{cases} \tag{45}$$

is used to require that interface information always travels downwind. The SAT parameter σ controls the tightness of coupling between subdomains.[1] The discretization of Eq. (43) has been previously shown to be stable and accurate for $\sigma \leq 0$. Reducing the value of σ represents additional artificial dissipation at the interface. Herein, a value of $\sigma = -\frac{1}{2}$ is used for most results. Extra dissipation is added in same cases to maintain stability near subdomain interfaces because WENO stencil-biasing is not possible. In general, the second term on the LHS of Eq. (43) can be any SBP-satisfying differentiation scheme and need not be the same for each subdomain. In this work, the fourth-order accurate ESWENO 3-4-3 scheme is used for all subdomains.

Theorem 1. *Equation (43) is both stable and accurate to the design-order of $\boldsymbol{Q}$. Stability and accuracy of the multi-block scheme follows trivially by combining the proofs in Refs. [12, 16]. These hold due to the combination of stable and accurate derivative operators and boundary closures.*

5.2 Hyperbolic Systems

The multi-block formulation is developed in the context of a linear hyperbolic equation. Systems of hyperbolic equations are accommodated by via a characteristic decomposition that forms a set of uncoupled, frozen, linear hyperbolic equations that each fit the form of Eq. (40). The linear differentiation and penalty scheme is applied on each of the characteristic equations. The resultant characteristic quantities are transformed back into physical space prior to time integration. This methodology is described below.

5.2.1 One-Dimensional Systems

Consider the general one-dimensional hyperbolic system of M differential equations

$$\frac{\partial \boldsymbol{U}}{\partial t} + \frac{\partial \boldsymbol{F}}{\partial x} = 0, \quad \boldsymbol{F} = \boldsymbol{F}(\boldsymbol{U}) \tag{46}$$

where $\boldsymbol{U}(x,t) = [U_1(x,t), U_2(x,t), \ldots, U_M(x,t)]$ is a vector of M conserved variables and $\boldsymbol{F}(x,t) = [F_1(x,t), F_2(x,t), \ldots, F_M(x,t)]$ is the flux vector. Applying the chain rule to the second term on the LHS of Eq. (47) yields

$$\frac{\partial \boldsymbol{U}}{\partial t} + \boldsymbol{A}\frac{\partial \boldsymbol{U}}{\partial x} = 0 \tag{47}$$

where the tensor $\boldsymbol{A} = \frac{\partial \boldsymbol{F}}{\partial \boldsymbol{U}}$ is the Jacobian matrix. Search for a characteristic eigendecomposition of the form

$$\boldsymbol{A} = \boldsymbol{S}\boldsymbol{\Lambda}\boldsymbol{S}^{-1} \tag{48}$$

[1] The notation of the constant σ is presented in a slightly different notation than that of Eq. (37); however, the two forms are equivalent.

with eigenvalue matrix $\boldsymbol{\Lambda} = \text{Diag}\left[\lambda_1, \lambda_2, \ldots, \lambda_M\right]$ and left eigenvector matrix $\boldsymbol{S}$. Note that the eigendecomposition in Eq. (48) is not guaranteed, in general, for all hyperbolic systems. Freeze $\boldsymbol{S}$ and $\boldsymbol{\Lambda}$ at a single point in time and space, substitute Eq. (48) into Eq. (47), and premultiply by $\boldsymbol{S}^{-1}$ to find the characteristic form:

$$\frac{\partial \boldsymbol{U}_c}{\partial t} + \boldsymbol{\Lambda}\frac{\partial \boldsymbol{U}_c}{\partial x} = 0, \tag{49}$$

where $\boldsymbol{U}_c = \boldsymbol{S}^{-1}\boldsymbol{U}$ and $\boldsymbol{F}_c = \boldsymbol{\Lambda}\boldsymbol{U}_c$ are the characteristic variables formed by transforming the physical variables $\boldsymbol{U}$ into characteristic space via $\boldsymbol{S}^{-1}$. Equation (49) is a set of uncoupled hyperbolic equations in characteristic space. The SAT penalty is applied in the linearized characteristic space:

$$\frac{\partial \boldsymbol{U}_c}{\partial t} + \boldsymbol{\Lambda}\frac{\partial \boldsymbol{U}_c}{\partial x} = (\sigma - \frac{1}{2})\boldsymbol{P}^{-1}\boldsymbol{\Lambda}\left[\mathtt{I}_{\boldsymbol{\Lambda}>0}\left([\boldsymbol{U}_c]_{L,k} - [\boldsymbol{U}_c]_{L,n_L}\right)\boldsymbol{e}_1 \; - \; (1-\mathtt{I}_{\boldsymbol{\Lambda}>0})\left([\boldsymbol{U}_c]_{R,k} - [\boldsymbol{U}_c]_{R,n_R}\right)\boldsymbol{e}_{N_k}\right].$$

Characteristic implementation for the new ESWENO-SAT scheme is as follows. The multi-block discretization, Eq. (43) is applied in characteristic space and rotated back into physical space before time-integration. The semidiscrete form is

$$\begin{aligned}\frac{\partial \boldsymbol{U}}{\partial t} + \boldsymbol{D}\boldsymbol{F} = \quad & \boldsymbol{P}^{-1}\boldsymbol{S}(\sigma - \tfrac{1}{2})\boldsymbol{\Lambda}\left[\mathtt{I}_{\boldsymbol{\Lambda}>0}\left([\boldsymbol{U}_c]_{L,k} - [\boldsymbol{U}_c]_{L,n_L}\right)\boldsymbol{e}_1\right. && (50)\\ & \left. - (1-\mathtt{I}_{\boldsymbol{\Lambda}>0})\left([\boldsymbol{U}_c]_{R,k} - [\boldsymbol{U}_c]_{R,n_R}\right)\boldsymbol{e}_{N_k}\right], && (51)\end{aligned}$$

where $\boldsymbol{D}$ represents the ESWENO differentiation operator. Expand $\boldsymbol{D}$ by applying the flux-differencing form of Eq. (36),

$$\boldsymbol{D}\boldsymbol{F} = \boldsymbol{P}^{-1}\boldsymbol{\Delta}\left(\bar{\boldsymbol{F}} + \bar{\boldsymbol{\psi}}\right). \tag{52}$$

Expand $\bar{\boldsymbol{F}}$ using the interpolation matrices as in Eq. (19) as

$$\boldsymbol{D}\boldsymbol{F} = \boldsymbol{P}^{-1}\boldsymbol{\Delta}\left(\sum_r \bar{w}^{(r)}\boldsymbol{\mathcal{I}}^{(r)}\boldsymbol{F} + \bar{\boldsymbol{\psi}}\right). \tag{53}$$

Expand $\boldsymbol{F}$ using the characteristic decomposition in Eq. (48),

$$\begin{aligned}\boldsymbol{D}\boldsymbol{F} &= \boldsymbol{P}^{-1}\boldsymbol{\Delta}\left(\sum_r \bar{w}^{(r)}\boldsymbol{\mathcal{I}}^{(r)}A\boldsymbol{U} + \bar{\boldsymbol{\psi}}\right) && (54)\\ &= \boldsymbol{P}^{-1}\boldsymbol{\Delta}\left(\sum_r \bar{w}^{(r)}\boldsymbol{\mathcal{I}}^{(r)}\boldsymbol{S}\boldsymbol{\Lambda}\boldsymbol{S}^{-1}\boldsymbol{U} + \bar{\boldsymbol{\psi}}\right) && (55)\\ &= \boldsymbol{P}^{-1}\boldsymbol{\Delta}\boldsymbol{S}\left(\sum_r \bar{w}^{(r)}\boldsymbol{\mathcal{I}}^{(r)}\boldsymbol{\Lambda}\boldsymbol{U}_c + \boldsymbol{S}^{-1}\bar{\boldsymbol{\psi}}\right). && (56)\end{aligned}$$

To facilitate upwinding, introduce the Lax-Friedrichs flux splitting

$$\boldsymbol{F}_c^{\pm} = \frac{1}{2}\left(\boldsymbol{\Lambda}\boldsymbol{U}_c \pm \boldsymbol{\Lambda}_{max}\boldsymbol{U}_c\right) \tag{57}$$

where $\boldsymbol{\Lambda}_{max}$ is the length M vector of maximum eigenvalues on subdomain k. The final form of the derivative operator is

$$\boldsymbol{D}\boldsymbol{F} = \boldsymbol{P}^{-1}\boldsymbol{\Delta}\boldsymbol{S}\left(\sum_r \bar{w}^{(r)}\boldsymbol{\mathcal{I}}^{(r)}\boldsymbol{F}_c^{+} + \sum_r \bar{w}^{(r)}\boldsymbol{\mathcal{I}}^{(r)}\boldsymbol{F}_c^{-} + \boldsymbol{S}^{-1}\bar{\boldsymbol{\psi}}\right). \tag{58}$$

The final semidiscrete form is

$$\frac{\partial \boldsymbol{U}}{\partial t} + \boldsymbol{P}^{-1}\boldsymbol{\Delta}\boldsymbol{S}\left(\sum_r \bar{w}^{(r)}\boldsymbol{\mathcal{I}}^{(r)}\boldsymbol{F}_c^+ + \sum_r \bar{w}^{(r)}\boldsymbol{\mathcal{I}}^{(r)}\boldsymbol{F}_c^- + \boldsymbol{S}^{-1}\bar{\boldsymbol{\psi}}\right) = \tag{59}$$
$$\boldsymbol{P}^{-1}\boldsymbol{S}(\sigma - \frac{1}{2})\boldsymbol{\Lambda}\left[\mathtt{I}_{\boldsymbol{\Lambda}>0}\left([\boldsymbol{U}_c]_{L,k} - [\boldsymbol{U}_c]_{L,n_L}\right)\boldsymbol{e}_1 - (1-\mathtt{I}_{\boldsymbol{\Lambda}>0})\left([\boldsymbol{U}_c]_{R,k} - [\boldsymbol{U}_c]_{R,n_R}\right)\boldsymbol{e}_{N_k}\right].$$

This yields an *effective* multi-block ESWENO numerical approximation of the derivative,

$$\begin{aligned}\boldsymbol{D}_{eff}\boldsymbol{F} &= \boldsymbol{P}^{-1}\boldsymbol{\Delta}\boldsymbol{S}\left(\sum_r \bar{w}^{(r)}\boldsymbol{\mathcal{I}}^{(r)}\boldsymbol{F}_c^+ + \sum_r \bar{w}^{(r)}\boldsymbol{\mathcal{I}}^{(r)}\boldsymbol{F}_c^- + \boldsymbol{S}^{-1}\bar{\boldsymbol{\psi}}\right) \\ &- \boldsymbol{P}^{-1}\boldsymbol{S}(\sigma - \frac{1}{2})\boldsymbol{\Lambda}\left[\mathtt{I}_{\boldsymbol{\Lambda}>0}\left([\boldsymbol{U}_c]_{L,k} - [\boldsymbol{U}_c]_{L,n_L}\right)\boldsymbol{e}_1 - (1-\mathtt{I}_{\boldsymbol{\Lambda}>0})\left([\boldsymbol{U}_c]_{R,k} - [\boldsymbol{U}_c]_{R,n_R}\right)\boldsymbol{e}_{N_k}\right].\end{aligned} \tag{60}$$

5.2.2 Multi-Dimensional Systems

For illustration, consider the two-dimensional system of m coupled hyperbolic equations

$$\frac{\partial \boldsymbol{U}}{\partial t} + \frac{\partial \boldsymbol{F}}{\partial x} + \frac{\partial \boldsymbol{G}}{\partial y} = 0, \quad \boldsymbol{F} = \boldsymbol{F}(\boldsymbol{U}), \quad \boldsymbol{G} = \boldsymbol{G}(\boldsymbol{U}) \tag{61}$$

where $\boldsymbol{U} = [U_1, U_2, \ldots, U_M]$. Use chain rule to expand the last two terms on the RHS of Eq. (61)

$$\frac{\partial \boldsymbol{U}}{\partial t} + \boldsymbol{A}_x\frac{\partial \boldsymbol{U}}{\partial x} + \boldsymbol{A}_y\frac{\partial \boldsymbol{U}}{\partial y} = 0 \tag{62}$$

where the Jacobians $\boldsymbol{A}_x = \frac{\partial \boldsymbol{F}}{\partial \boldsymbol{U}}$ and $\boldsymbol{A}_y = \frac{\partial \boldsymbol{G}}{\partial \boldsymbol{U}}$ are assumed to have the eigen-decomposition

$$\boldsymbol{A}_x = \boldsymbol{S}_x\boldsymbol{\Lambda}_x\boldsymbol{S}_x^{-1} \tag{63a}$$
$$\boldsymbol{A}_y = \boldsymbol{S}_y\boldsymbol{\Lambda}_y\boldsymbol{S}_y^{-1}. \tag{63b}$$

Substitute Eq. (63) into Eq. (62):

$$\frac{\partial \boldsymbol{U}}{\partial t} + \boldsymbol{S}_x\frac{\partial \boldsymbol{F}_c}{\partial x} + \boldsymbol{S}_y\frac{\partial \boldsymbol{G}_c}{\partial y} = 0. \tag{64}$$

Solve Eq. (64) by independently applying Eq. (60) on $\frac{\partial \boldsymbol{F}}{\partial x}$ and $\frac{\partial \boldsymbol{G}}{\partial y}$, then integrating in time.

For example, the characteristic decomposition of the two-dimensional Euler equations,

$$\frac{\partial \boldsymbol{U}}{\partial t} + \frac{\partial \boldsymbol{F}}{\partial x} + \frac{\partial \boldsymbol{G}}{\partial y} = 0, \tag{65}$$

$$\boldsymbol{U} = \begin{pmatrix}\rho\\ \rho u\\ \rho v\\ \rho E\end{pmatrix} \quad \boldsymbol{F} = \begin{pmatrix}\rho u\\ \rho u^2 + p\\ \rho uv\\ (\rho E + p)u\end{pmatrix} \quad \boldsymbol{G} = \begin{pmatrix}\rho v\\ \rho uv\\ \rho v^2 + p\\ (\rho E + p)v\end{pmatrix} \tag{66}$$

is accomplished via the characteristic decomposition of Pulliam and Chaussee,[26]

$$\boldsymbol{S}_h = \begin{pmatrix}1 & 0 & \alpha & \alpha\\ \bar{u} & \bar{\rho}\,(1-\mathtt{I}_{h=\text{`x'}}) & \alpha(\bar{u}+\mathtt{I}_{h=\text{`x'}}c) & \alpha(\bar{u}-\mathtt{I}_{h=\text{`x'}}c)\\ \bar{v} & -\bar{\rho}\mathtt{I}_{h=\text{`x'}} & \alpha\left[\bar{v}+(1-\mathtt{I}_{h=\text{`x'}})\,c\right] & \alpha\left[\bar{v}-(1-\mathtt{I}_{h=\text{`x'}})\,c\right]\\ \frac{\phi^2}{\gamma-1} & \bar{\rho}\left[(1-\mathtt{I}_{h=\text{`x'}})\,\bar{u} - \mathtt{I}_{h=\text{`x'}}\bar{v}\right] & \alpha(\frac{\phi^2+c^2}{\gamma-1} + c\theta) & \alpha(\frac{\phi^2+c^2}{\gamma-1} - c\theta)\end{pmatrix} \tag{67}$$

$$\boldsymbol{S}_h^{-1} = \begin{pmatrix} 1-\frac{\phi^2}{c^2} & (\gamma-1)\frac{\bar{u}}{c^2} & (\gamma-1)\frac{\bar{v}}{c^2} & -\frac{\gamma-1}{c^2} \\ \frac{\mathtt{I}_{h=\text{`x'}}\bar{v}-(1-\mathtt{I}_{h=\text{`x'}})\bar{u}}{\bar{\rho}} & \frac{1-\mathtt{I}_{h=\text{`x'}}}{\bar{\rho}} & -\frac{\mathtt{I}_{h=\text{`x'}}}{\bar{\rho}} & 0 \\ \beta(\phi^2-c\theta) & \beta\left[\mathtt{I}_{h=\text{`x'}}c-(\gamma-1)\bar{u}\right] & \beta\left[(1-\mathtt{I}_{h=\text{`x'}})\,c-(\gamma-1)\bar{v}\right] & \beta(\gamma-1) \\ \beta(\phi^2+c\theta) & -\beta\left[\mathtt{I}_{h=\text{`x'}}c+(\gamma-1)\bar{u}\right] & -\beta\left[(1-\mathtt{I}_{h=\text{`x'}})\,c+(\gamma-1)\bar{v}\right] & \beta(\gamma-1) \end{pmatrix} \tag{68}$$

where

$$\alpha = \frac{\rho}{\sqrt{2}c} \tag{69}$$

$$\beta = \frac{1}{\sqrt{2}\rho c} \tag{70}$$

$$\theta = \mathtt{I}_{h=\text{`x'}}u + (1-\mathtt{I}_{h=\text{`x'}})\,v \tag{71}$$

$$\phi^2 = \frac{1}{2}(\gamma-1)\,u^2 + v^2 \tag{72}$$

and the matrices $\boldsymbol{S}_x$ and $\boldsymbol{S}_y$ are presented in combined the combined form $\boldsymbol{S}_h$, $h \in \{\text{`x'},\text{`y'}\}$. The operator $\mathtt{I}_{\mathtt{h=`x'}}$ is an indicator function for $h = \text{`x'}$.

5.3 Computational Implementation

The problem geometry is defined and the domain is mapped with a set of independent subdomains. If curvilinear meshes are used, the shape of the subdomains is flexible and the domain decomposition may be entirely unstructured with each block having an arbitrary shape.[2] Such flexibility allows blocks surrounding complex geometry to exactly match the body locally, instead of fitting the entire body with one mesh. The only requirement is that for each subdomain-to-subdomain interface, the sets of points lying on the interface owned by the left and right subdomains must be identical. That is, all points on the interface must be collocated.[3] No restriction is placed on similarity of grid topology or grid-spacing.

In our implementation, each subdomain is sent to a single processor on a distributed-memory computing architecture. The multi-block ESWENO scheme, Eq. (60), is carried out separately on each subdomain, in each dimension, with message-passing of collocated interface points between neighboring subdomains for calculation of the SAT penalty. Time integration is via an explicit fourth-order Runge-Kutta scheme.[29]

6 Numerical Results

Numerical tests are presented to show the stability and accuracy of the multi-block scheme. Design-order accuracy on smooth problems is shown and the ability of the scheme to accurately capture strong shocks without significant oscillation or diffusion is exemplified. A highlight of the results is the lack of interference of subdomain interfaces on the problem physics. Both delicate physical phenomena (such as a weak vortex) and strong shocks pass through interfaces without noticeable diffusion or dispersion, even in the presence of large jumps in grid-spacing across the interface.

Additionally, simulations using strongly-imposed interface conditions are contrasted with those of the new multi-block scheme, which utilizes weak SAT penalties. Strong interface condition imposition is the practice of overwriting collocated interface data with interface data calculated on the upwind subdomain. This strong interface condition upwinding is performed on the local Runge-Kutta substep variables before each Runge-Kutta substep and again on the global variable after each global time integration. Unless otherwise noted, simulations default to weak SAT interface conditions.

Results are presented for one- and two-dimensional test problems for both linear and nonlinear systems of hyperbolic equations. One- and two-dimensional calculations are presented for a linear scalar hyperbolic

[2] In this work, only Cartesian meshes are used; extension to curvilinear grids is reserved for a future work.

[3] It seems feasible that this requirement could be relaxed by repurposing ideas from adaptive mesh refinement (AMR).[27,28] Development for non-collocated interface points is reserved for a future work.

Table 1: L_2 error and convergence rates for one-dimensional sine wave advection problem. Results from single-domain ESWENO are compared with results from simulations on four equally-spaced subdomains using the new methodology. Error is calculated after ten flow-through times.

Number of points	Single domain		Four subdomains	
	Log L_2 error	Rate	Log L_2 error	Rate
21	-0.2998		0.1996	
49	-1.7806	3.89	-1.7819	5.21
101	-3.0587	4.01	-3.0589	4.01
201	-4.2638	4.00	-4.2640	4.00
401	-5.4682	4.00	-5.4683	4.00
801	-6.6723	4.00	-6.6724	4.00
1601	-7.8700	3.98	-7.8628	3.95

wave equation. For convergence results, the L_2 error,

$$\|E\|_{L_2} = \sqrt{\sum_{m=1}^{M} \int_{\Omega} (u_m - v_m)^2 \, dxdy}, \tag{73}$$

is integrated numerically using the midpoint rectangle rule.

6.1 One-dimensional Linear Scalar Wave Equation

The one-dimensional linear scalar hyperbolic wave equation

$$\frac{\partial \rho}{\partial t} + \frac{\partial \rho}{\partial x} = 0 \tag{74}$$

is solved for the conserved scalar $\rho(x,t)$. Numerical results are presented comparing strong versus weak boundary and interface conditions. Equation (74) is solved for both smooth and discontinuous initial conditions.

6.1.1 Sine-Wave Advection

Equation (74) is solved numerically using the smooth initial condition

$$\rho(x,0) = sin(x), \quad x \in [-\pi, \pi] \tag{75}$$

after ten flow-through times. Calculations are performed on a single subdomain and 4 equally-spaced subdomains. L_2 error and convergence rates are shown in Table 1. Both single- and multi-domain results converge to design-order. Moreover, the difference in L_2 error between the two configurations is negligible.

Calculations are also performed on 4 unequally-spaced subdomains using the initial condition

$$\rho(x,0) = sin(2\pi f x), \quad x \in [0, 1] \tag{76}$$

for frequency $f = 3.0$ on the domain $0 \leq x \leq 1$ with periodic boundary conditions for one flow-through time ($t = 1.0$). Both strong and weak interface conditions are shown in Fig. 1. The maximum grid compression ratio between neighboring subdomains is 4. No noticeable oscillations appear for either interface condition scheme.

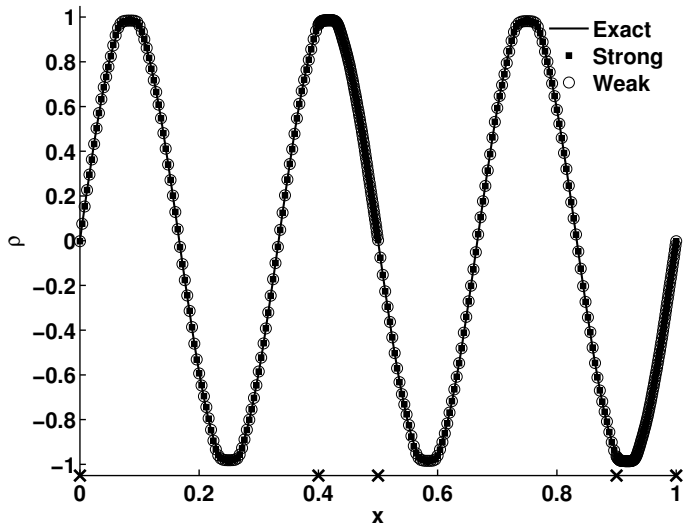


Figure 1: One-dimensional sine-wave advection problem after one flow-through time ($t = 1.0$) on 4 unequally-spaced subdomains for strong and weak (SAT) boundary and interface conditions. Each subdomain has 101 grid-points resulting in 404 total grid points and a maximum grid compression ratio of 4. X's on the horizontal axis denote interface locations.

6.1.2 Square-Wave Advection

Equation (74) is solved for discontinuous initial condition

$$\rho(x,0) = \begin{cases} 1 & : \quad x \in (0.3, 0.6) \\ 0 & : \quad x \notin (0.3, 0.6) \end{cases} \quad , \quad x \in [0,1] \tag{77}$$

with periodic boundary conditions for one flow-through time ($t = 1.0$). Calculations are performed on 4 unequally-spaced subdomains using strong and weak boundary and interface conditions shown in Fig. 2. The maximum grid compression jump between neighboring subdomains is 4. Strong imposition of the interface conditions results in significant oscillations, while weak conditions eliminate these oscillations.

6.2 Two-dimensional Linear Scalar Wave Equation

The two-dimensional linear scalar wave equation (hyperbolic)

$$\frac{\partial \rho}{\partial t} + a\frac{\partial \rho}{\partial x} + b\frac{\partial \rho}{\partial y} = 0 \tag{78}$$

is solved for the conserved scalar $\rho(x, y, t)$. The constants a and b are horizontal and vertical wave speeds, respectively.

6.2.1 Circle-Shaped Square-Wave Advection

Equation (78) is solved for the discontinuous initial condition shown in Fig. 3. The circle is advected diagonally towards the top-right of the domain at speed 1. Results for a complete flow through the domain are presented in Fig. 4. Calculations are performed on 4 equally-spaced subdomains using weak (SAT) boundary and interface conditions. Each subdomain has 101×101 grid-points and the entire grid is 202×202.

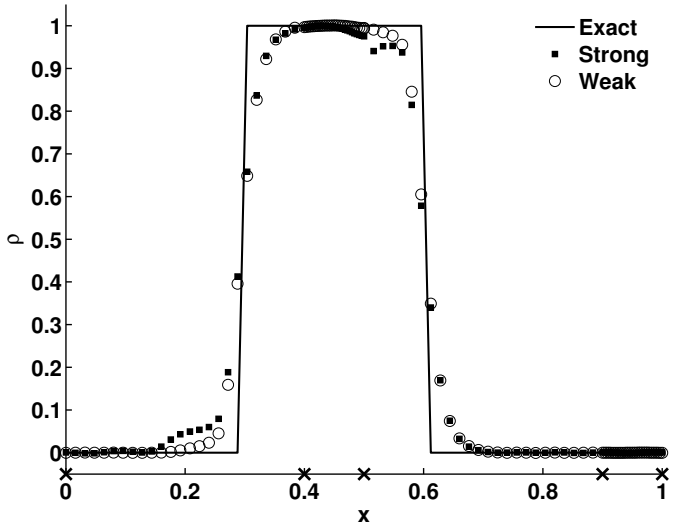


Figure 2: One-dimensional square-wave advection problem after one flow-through time ($t = 1.0$) on 4 unequally-spaced subdomains using strong and weak (SAT) boundary and interface conditions. Each subdomain has 26 grid-points resulting in 104 total grid-points and a maximum grid compression ratio of 4 between neighboring subdomains. X's on the horizontal axis denote interface locations.

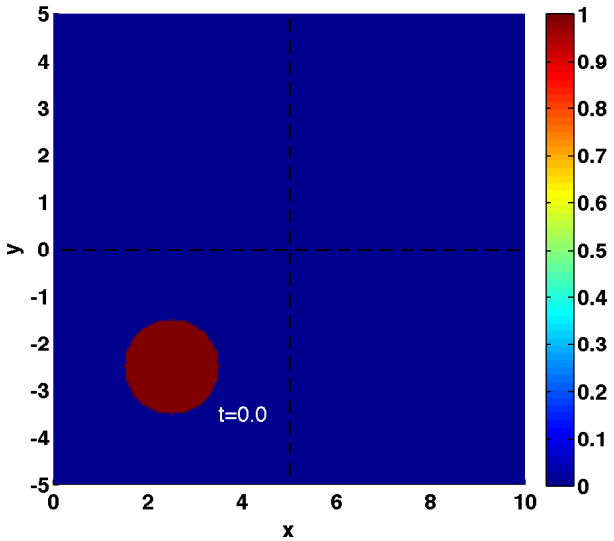
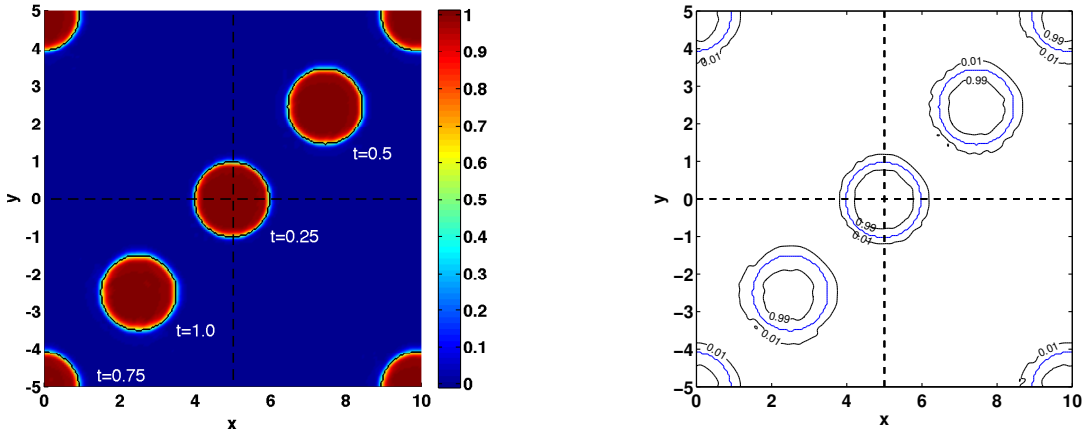


Figure 3: Initial condition (ρ) for two-dimensional square-wave advection problem.



(a) (b)

Figure 4: Two-dimensional square-wave advection problem for one flow-through time on 4 equally-spaced subdomains using weak (SAT) boundary and interface conditions. Each subdomain has 101×101 grid-points and the entire grid is 202×202. Dashed lines denote interface locations. In (a), solid lines denote exact solution and color contours are the numerical solution at various times. In (b), solid blue lines denote exact solution and solid black lines represent contours $\rho = 0.01$ and $\rho = 0.99$ of the numerical solution to magnify shock smearing.

6.3 Euler Equations

6.3.1 One-Dimensional Sod's Shock Tube Problem

The one-dimensional Euler equations,

$$\frac{\partial \boldsymbol{U}}{\partial t} + \frac{\partial \boldsymbol{F}}{\partial x} = 0,$$

$$\boldsymbol{U} = \begin{pmatrix} \rho \\ \rho u \\ \rho E \end{pmatrix}, \quad \boldsymbol{F} = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ (\rho E + p)u \end{pmatrix}, \tag{79}$$

are solved for Sod's shock tube problem.[30] Results for strong and weak boundary and interface conditions for 4 equally-spaced subdomains are shown in Fig. 5. Results for strong and weak boundary and interface conditions for 4 unequally-spaced subdomains with a maximum grid compression ratio of 4.0 are shown in Figs. 6. Noticeable oscillations develop for strongly imposed conditions. Weak conditions prevent oscillations in density and velocity; however, the some perturbation of the energy remains.

6.3.2 Two-Dimensional Inviscid Vortex

The two-dimensional Euler equations are solved for an inviscid vortex convection problem with exact solution

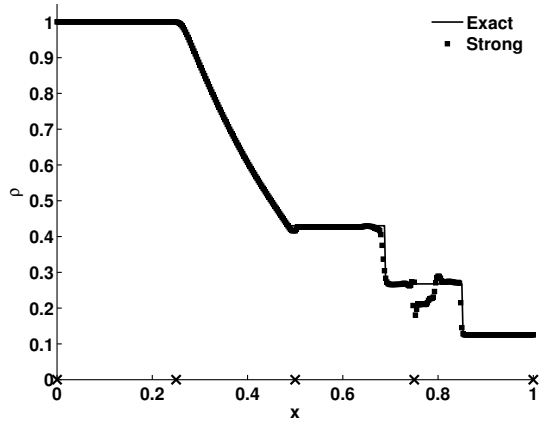
$$f(x,y,t) = 1 - ((x - x_0 - U_\infty t)^2 + (y - y_0)^2), \tag{80}$$

$$\rho(x,y,t) = \left(1 - \epsilon^2 \frac{\gamma - 1}{8\pi^2} exp(f(x,y,t))\right)^{\frac{1}{\gamma - 1}}, \quad p = \frac{\rho}{\gamma}, \tag{81}$$

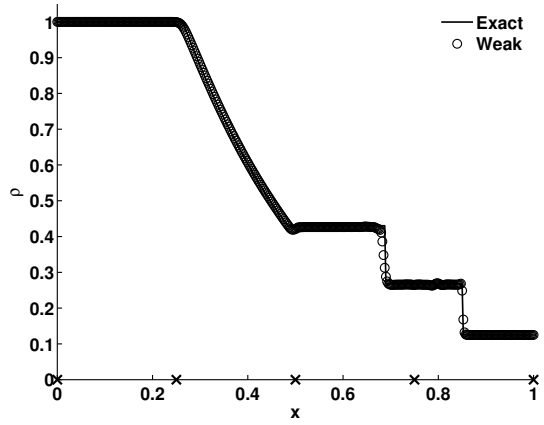
$$u(x,y,t) = U_\infty - \epsilon \frac{y - y_0}{2\pi} exp\left(\frac{f(x,y,t)}{2}\right), \tag{82}$$

$$v(x,y,t) = \epsilon \frac{x - x_0 - U_\infty t}{2\pi} exp\left(\frac{f(x,y,t)}{2}\right), \tag{83}$$

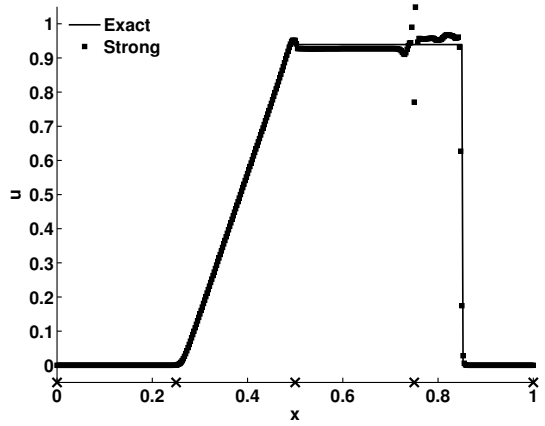
$$U_\infty = M_\infty c_\infty, \; M_\infty = 0.5, \quad x \in [0,10], \quad y \in [-5,5], \quad (x_0, y_0) = (5,0), \quad t \geq 0. \tag{84}$$



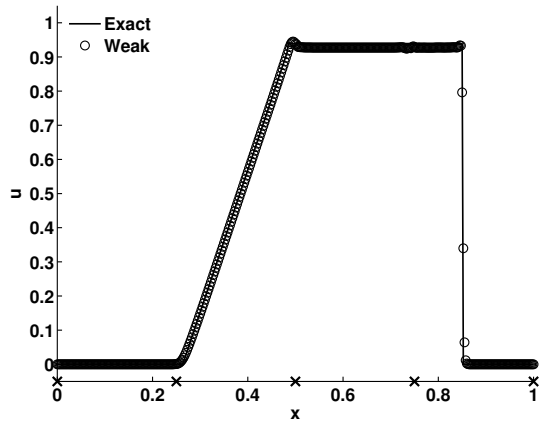
(a) Density



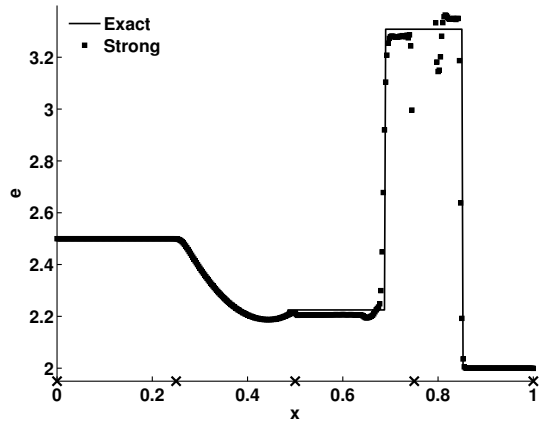
(b) Density



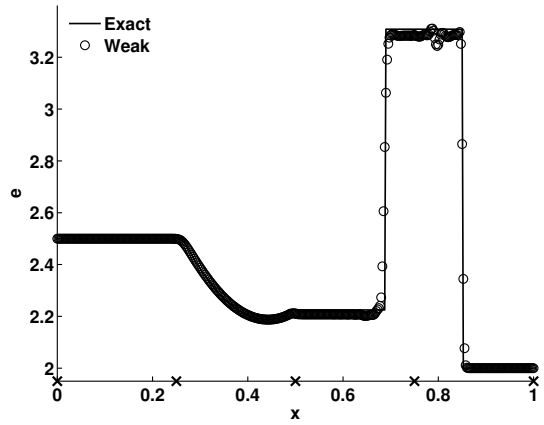
(c) Velocity



(d) Velocity

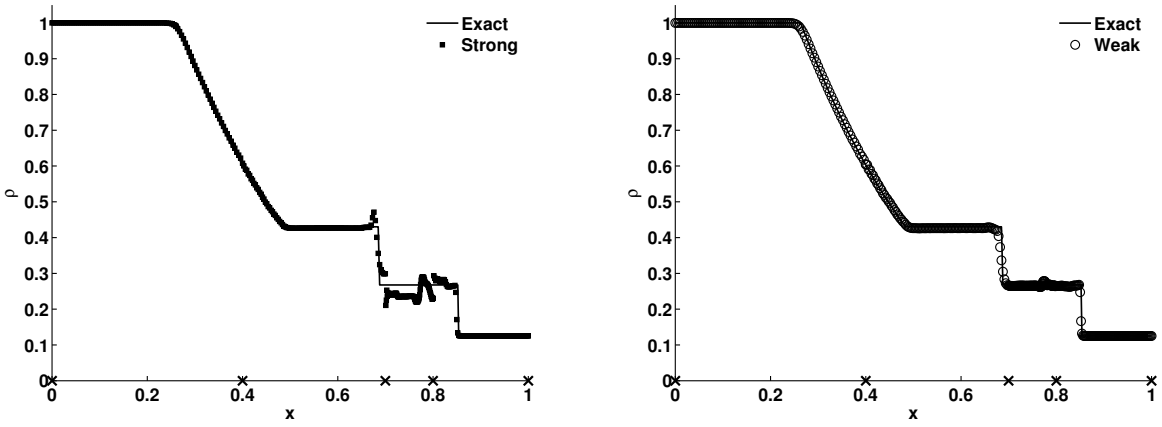


(e) Energy



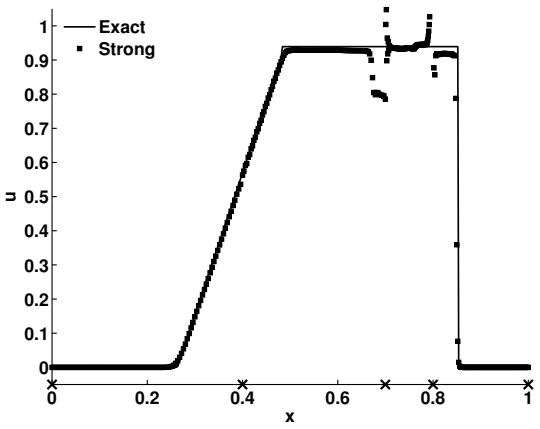
(f) Energy

Figure 5: Sods one-dimensional shock tube problem at $t = 0.2$ on 4 equally-spaced subdomains. Left column: strong interface conditions. Right column: weak SAT interface conditions. Each subdomain has 101 grid-points resulting in 404 total grid-points. X's on horizontal axis denote interface locations.

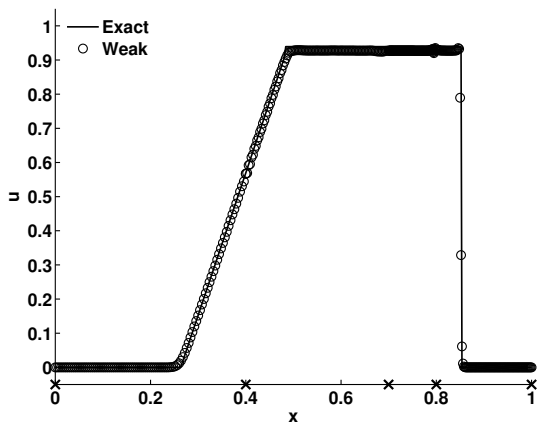


(a) Density

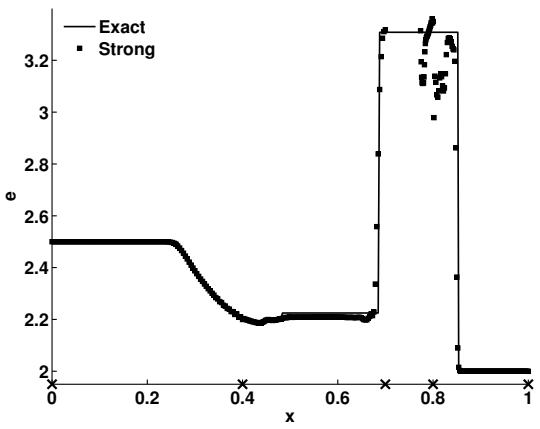
(b) Density



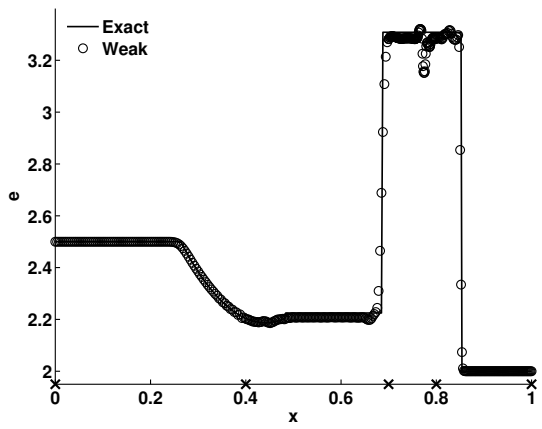
(c) Velocity



(d) Velocity



(e) Energy

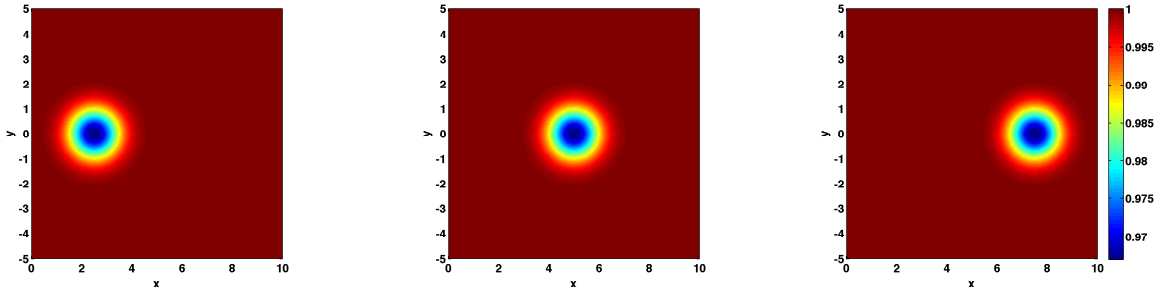


(f) Energy

Figure 6: Sods one-dimensional shock tube problem at $t = 0.2$ on 4 unequally-spaced subdomains using. Left column: strong interface conditions. Right column: weak SAT interface conditions. Each subdomain has 101 grid-points resulting in 404 total grid-points and a maximum grid compression ratio of 4 between neighboring subdomains. X's on horizontal axis denote interface locations.

Table 2: Total L_2 error and convergence rates for weak (SAT) boundary conditions for two-dimensional inviscid vortex convection problem on single domain and multiple domains. Error is calculated at $t = 10s$.

Grid	Single domain		Four subdomains	
	Log L_2 error	Rate	Log L_2 error	Rate
21×21	-0.73		-0.74	
41×41	-1.81	3.60	-1.82	3.58
81×81	-3.00	3.95	-2.98	3.86
161×161	-3.98	3.25	-3.91	3.08
321×321	-4.68	2.32	-4.61	2.33
641×641	-5.30	2.05	-5.24	2.09



(a) $t = 0$. (b) $t = 5$. (c) $t = 10$.

Figure 7: Exact density solution for the inviscid vortex advection problem at various times.

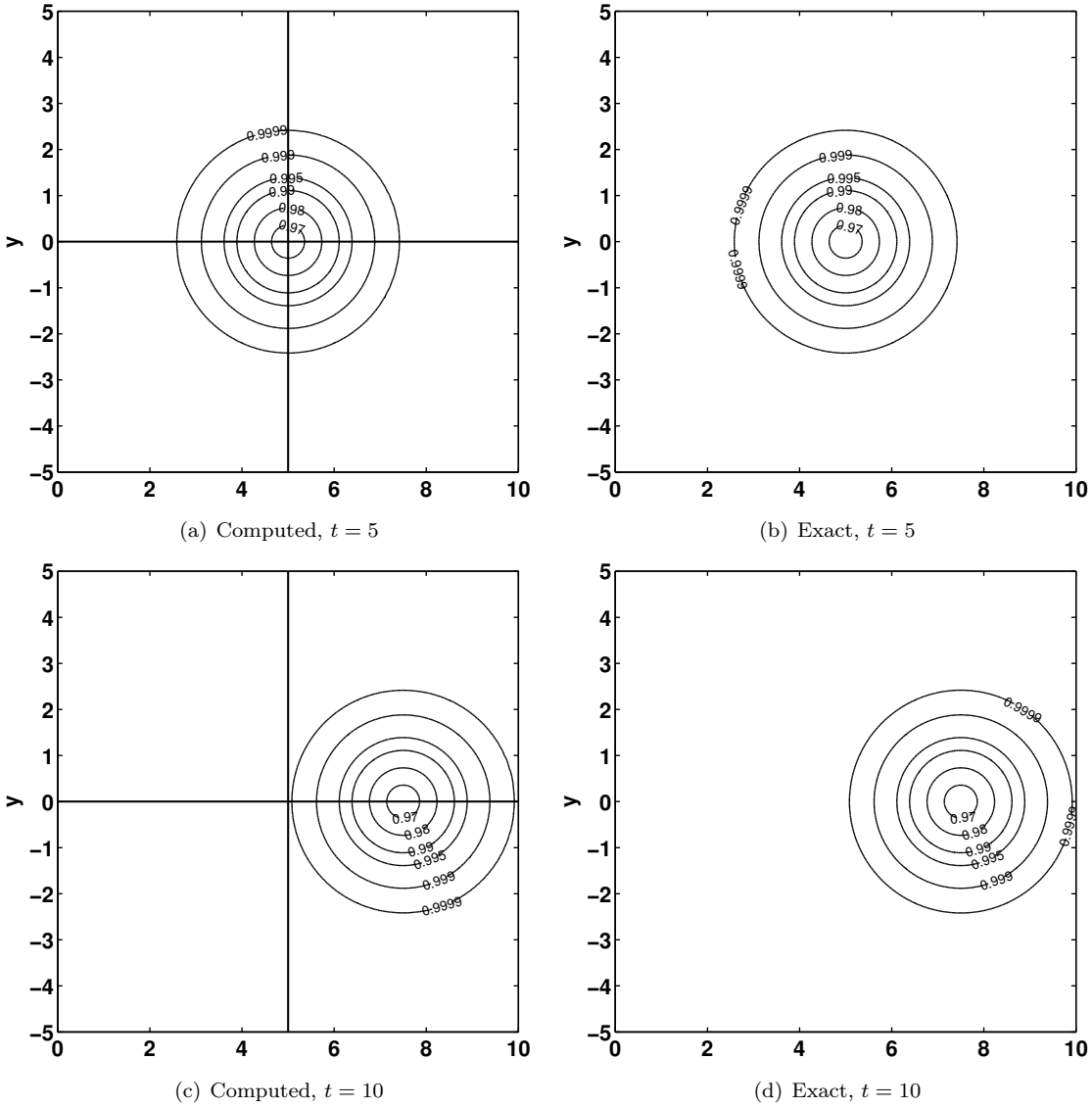
The exact density solution is shown in Fig. 7. Calculations are performed on 4 equally-spaced subdomains using weak SAT boundary and interface conditions. L_2 error and convergence rates are shown in Table 2. Both single- and multi-domain results converge to design-order. Moreover, the difference in L_2 error between the two configurations is small. Example results are shown in Fig. 8, left column, where each subdomain has 101×101 grid-points, thus the entire grid is 202×202. Calculations are also performed on 4 unequally-spaced subdomains using weak SAT boundary and interface conditions as shown in Fig. 10, left column. Subdomain grid-spacing is shown in Table 3. The entire grid is 292×292 and the maximum grid compression ratio across neighboring subdomains is 2.22.

Table 3: Number of grid-points in each subdomain for inviscid vortex convection problem with unequally-spaced subdomain grids, Fig. 10.

Subdomain	N_x	N_y
bottom left	201	91
bottom right	91	91
top left	201	201
top right	91	201

6.3.3 Two-dimensional Mach 3 Flow Over a Forward-Facing Step

Results are presented for compressible inviscid flow over a forward-facing step at $Ma = 3$.[18,31] The computed density at $t = 4.0$ seconds on 3 equally-spaced subdomains using weak SAT boundary and interface conditions is shown in Fig. 8. Subdomain partitions are shown. The entire grid is 241×81 and is compared to the fifth-order WENO result of Shu[18] on a 242×79 mesh. For demonstration, calculations were also performed on 3 subdomains with nonuniform grid-spacing. Results for $t = 3.0$ are shown in Fig. 10. The entire grid is



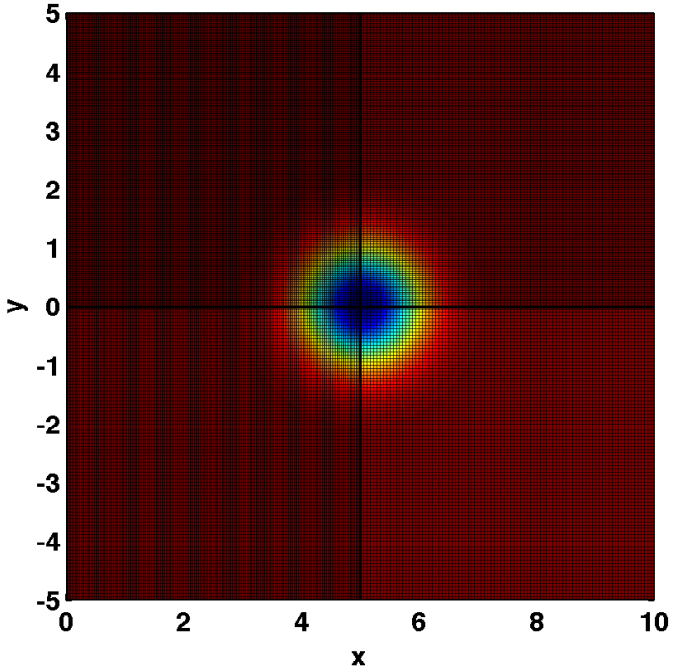
(a) Computed, $t = 5$

(b) Exact, $t = 5$

(c) Computed, $t = 10$

(d) Exact, $t = 10$

Figure 8: Left column: computed density contours for the inviscid vortex advection problem on 4 equally-spaced subdomains using weak (SAT) boundary and interface conditions. Each subdomain has 101×101 grid-points and the entire grid is 202×202. Solid lines denote interface locations. Right column: analytic solution for comparison.



(a) $t = 5$

Figure 9: Computed density for the inviscid vortex advection problem on 4 unequally-spaced subdomains using weak (SAT) boundary and interface conditions. Subdomain grid sizes are shown in Table 3. The entire grid is 292×292 and maximum grid compression ratio is 2.22. Solid lines denote interface locations.

Table 4: Grid spacing for Mach 3 flow over front-facing step, Fig. 12.

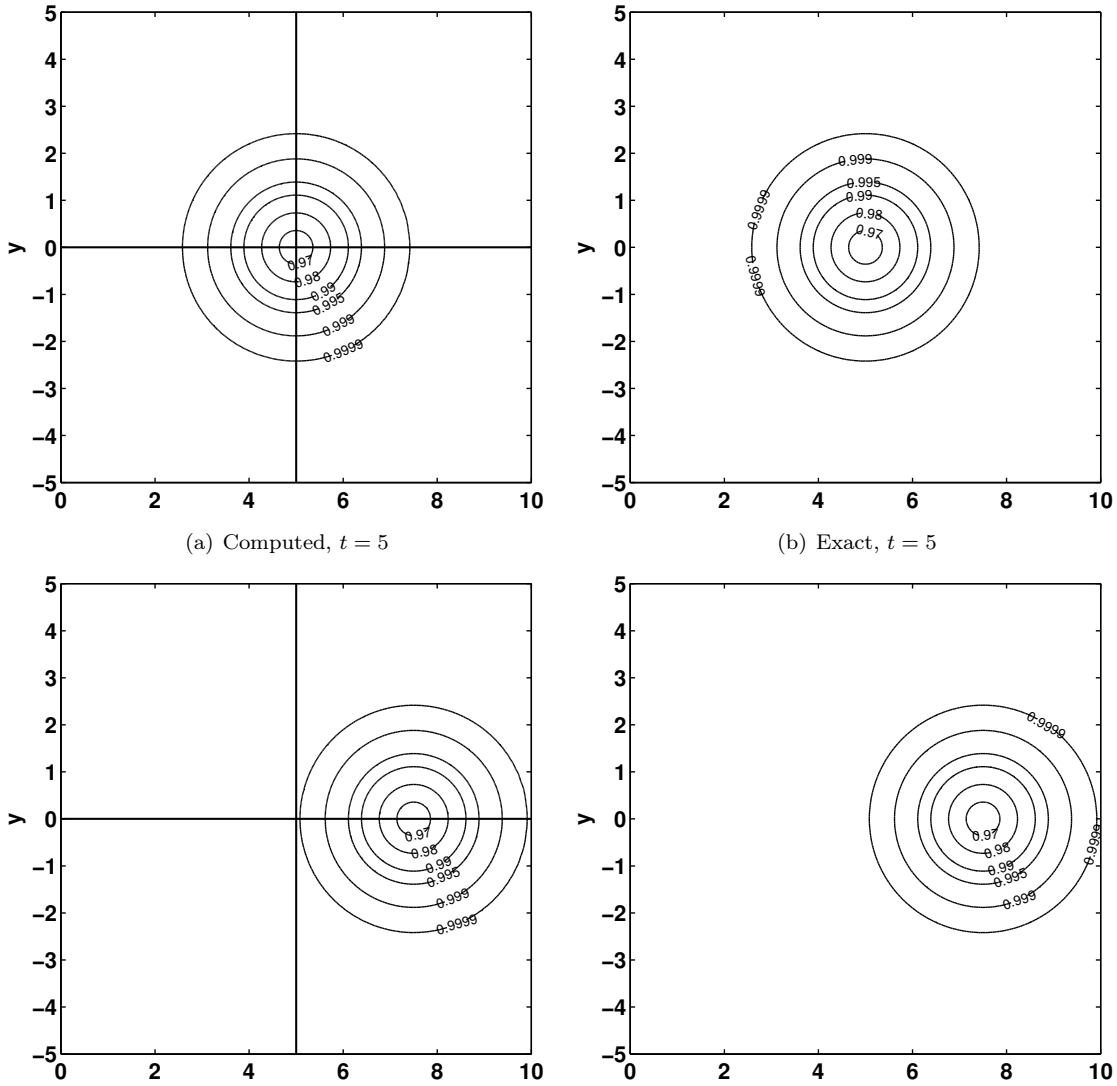
Subdomain	Δx	Δy
bottom left	0.0125	0.0125
top left	0.0125	0.025
top right	0.025	0.025

145×49 and subdomain grid-spacing is shown in Table 4. Maximum grid compression ratio across neighboring subdomains is 2. Interference from the interface conditions is slight. For these results, dissipation was added at interfaces via the SAT parameter, σ, to maintain stability.

7 Conclusions and Future Work

A methodology has been devised and demonstrated that allows smooth solutions and discontinuities to be captured and accurately passed across subdomain boundaries even when jumps in grid-spacing exist. The closed-domain ESWENO 3-4-3 differentiation scheme and SAT penalty method are married to produce a robust solver for hyperbolic-dominated problems. The presented numerical examples show the flexibility, stability, and accuracy of the scheme. Aside from producing a reliable multi-block scheme, there are three potential uses for the scheme that warrant its further development:

High-order accuracy on complex geometries. Use the scheme to facilitate high-order accurate simulations on complex geometries. Implementation of a curvilinear grid transformation on each subdomain would accept geometries of arbitrary complexity. Map the domain with a very coarse unstructured quadri-



(a) Computed, $t = 5$

(b) Exact, $t = 5$

(c) Computed, $t = 10$

(d) Exact, $t = 10$

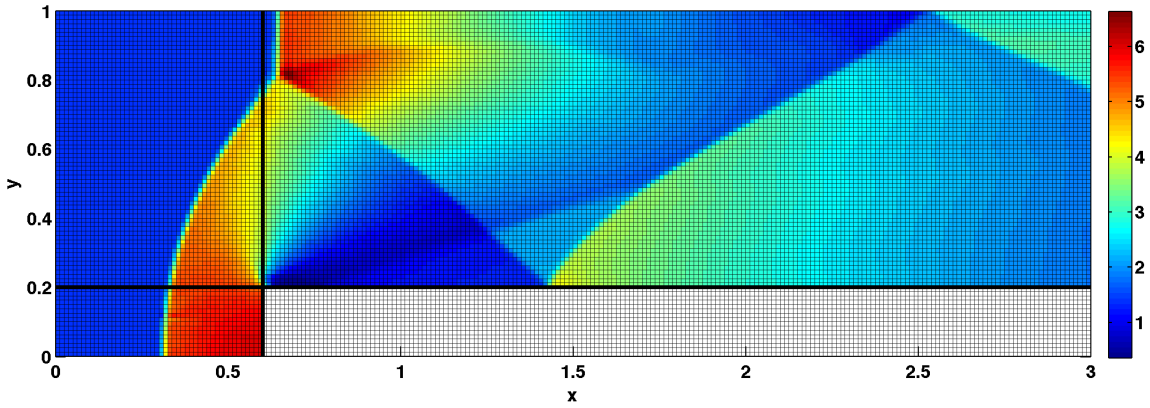
Figure 10: Left column: computed density for the inviscid vortex advection problem on 4 unequally-spaced subdomains using weak (SAT) boundary and interface conditions. Subdomain grid sizes are shown in Table 3. The entire grid is 292×292 and maximum grid compression ratio is 2.22. Solid lines denote interface locations. Right column: analytic solution for comparison.

lateral or hexahedral mesh. Treat each mesh volume as a subdomain and map each with a boundary-fitted curvilinear mesh.

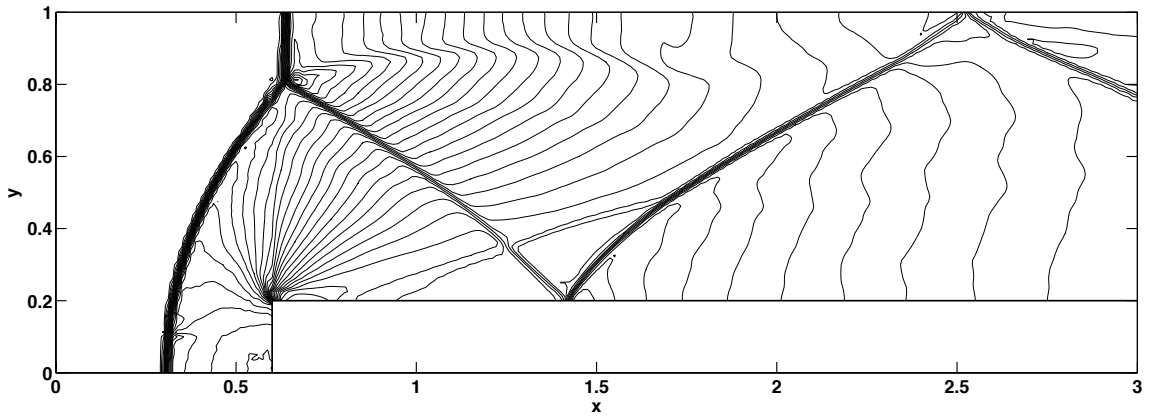
Low-overhead parallelization. Use the multi-block scheme solely for efficient parallelization. Map the domain with a traditional finite-difference mesh. Partition the domain to achieve load-balancing and use the new multi-block scheme to minimize message-passing cost as an alternative to message-passing all overlapping stencils. The new multi-block scheme seems to sacrifice neither stability nor accuracy; there would be little downside to such a parallelization scheme, if any.

Local grid refinement. Use the scheme to refine the mesh in areas of local interest without worrying about mesh smoothing.

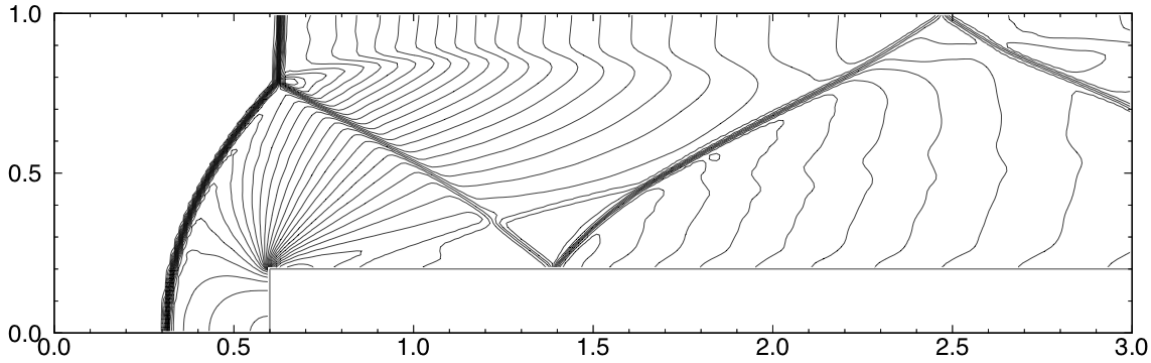
Each of these uses by itself represents a significant advantage; however, a combination of all three is an exciting proposition. The demonstrated methodology is extremely promising for such high-order accurate



(a) Density with mesh overlaid

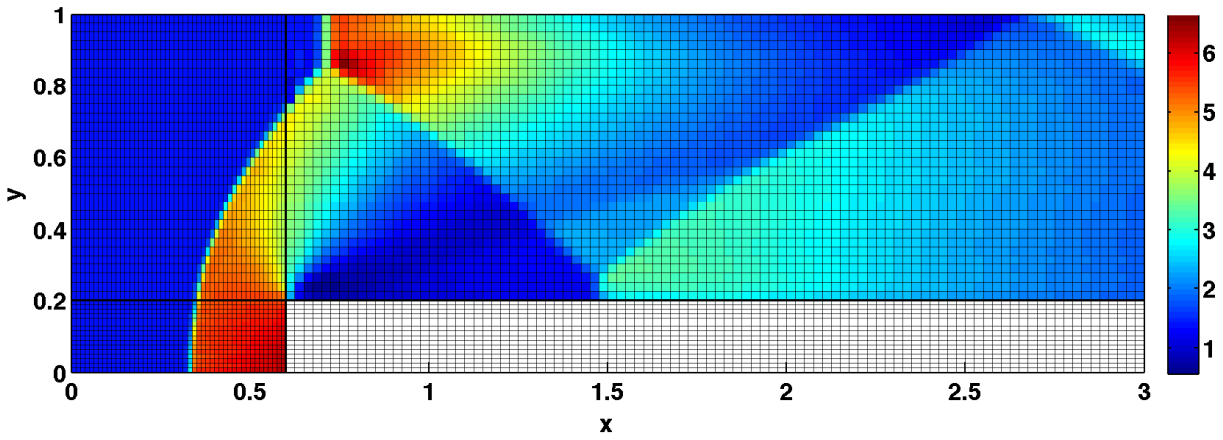


(b) Density

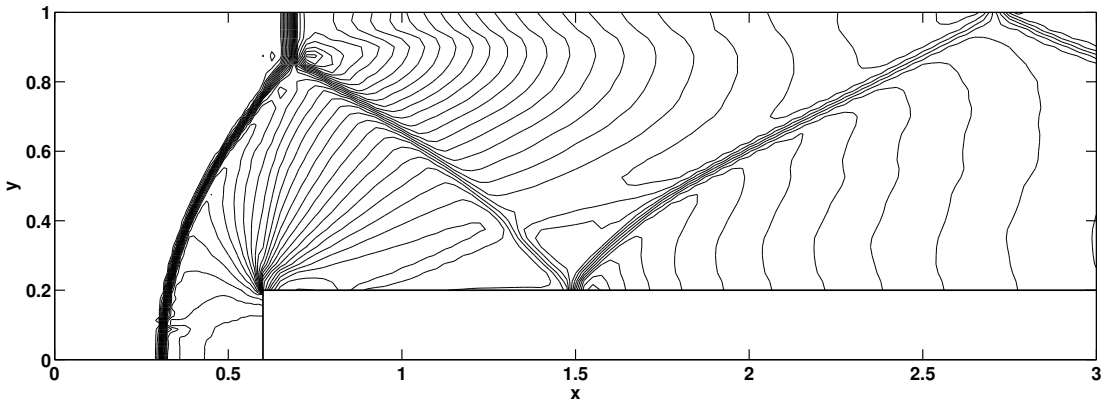


(c) Density from Ref. 18

Figure 11: (a,b) Computed density for two-dimensional Mach 3 flow over a front-facing step at $t = 4.0$ seconds on 3 equally-spaced subdomains using weak (SAT) boundary and interface conditions. In (a), bold solid lines denote interface locations. The entire grid is 241×81. (c) Result of Shu for 5th-order WENO on a 242×79 mesh.



(a) Density with mesh overlaid



(b) Density

Figure 12: (a,b) Computed density for two-dimensional Mach 3 flow over a front-facing step at $t = 3.0$ seconds on 3 unequally-spaced subdomains using weak (SAT) boundary and interface conditions. Subdomain grid-spacing is shown in Table 4 . Maximum grid compression ratio across neighboring subdomains is 2. Bold solid lines denote interface locations. The entire grid is 145×49.

simulations in complex configurations, including high-performance parallel computing environments and complicated geometries.

Acknowledgments

The original idea for this work was conceived by Dr. Mark H. Carpenter (NASA Langley Research Center). I sincerely thank him for many helpful comments, suggestions, and insights. I also thank Mr. Travis Fisher (NASA Langley Research Center) for many helpful conversations. *Errors and omissions in this work are my own.*

This work was supported by the NASA Aeronautics Scholarship Program and NASA/AFOSR Grant FA9550-09-1-0611 (National Center for Hypersonic Combined Cycle Propulsion). Additional support provided through AFOSR contract FA8650-10-C-2008 between Air Force Research Laboratory and Spectral Energies, LLC (Chiping Li, AFOSR Program Manager).

References

[1] Rafael Borges, Monique Carmona, Bruno Costa, and Wai Sun Don. An improved weighted essentially non-oscillatory scheme for hyperbolic conservation laws. *J. Comput. Phys.*, 227(6):3191–3211, 2008.

[2] Raimund Burger and Alice Kozakevicius. Adaptive multiresolution weno schemes for multi-species kinematic flow models. *J. Comput. Phys.*, 224(2):1190–1222, 2007.

[3] Chi-Wang Shu. High order weighted essentially non-oscillatory schemes for convection dominated problems. *SIAM Rev.*, 51(1):82–126, 2009.

[4] Ching-Shan Chou and Chi-Wang Shu. High order residual distribution conservative finite difference weno schemes for convection-diffusion steady state problems on non-smooth meshes. *J. Comput. Phys.*, 224(2):992–1020, 2007.

[5] F. Erlebacher, M. Hussaini, and C.-W. Shu. Interaction of a shock with a longitudinal vortex. *J. Fluid Mech.*, 337:129–153, 1997.

[6] S. Zhang, S. Jiang, Y. Zhang, and Shu C. The mechanism of sound generation in the interaction between a shock wave and two counter-rotating vortices. *Phys. Fluids*, 21(7), 2009.

[7] B. Strand. Summation by parts for finite difference approximations for d/dx. *J. Comput. Phys.*, 110(1):47–67, 1994.

[8] H.-O. Kreiss and G. Scherer. Finite element and finite difference methods for hyperbolic partial differential equations. In *Mathematical Aspects of Finite Elements in Partial Differential Equations*. Academic Press, Inc., New York, NY, 1974.

[9] Nail Yamaleev and Mark Carpenter. Third-order energy stable weno scheme. *J. Comput. Phys.*, 228(8):3025–3047, 2009.

[10] Nail Yamaleev and Mark Carpenter. A systematic methodology for constructing high-order energy stable weno schemes. *J. Comput. Phys.*, 228(11):4248–4272, 2009.

[11] Travis Fisher, Mark Carpenter, Nail Yamaleev, and Steven Frankel. Boundary closures for fourth-order energy stable weighted essentially non-oscillatory finite difference schemes. NASA TM 2009-216166, NASA Langley Research Center, Hampton, VA, 2009.

[12] Travis Fisher, Mark Carpenter, Nail Yamaleev, and Steven Frankel. Boundary closures for fourth-order energy stable weighted essentially non-oscillatory finite-difference schemes. *J. Comput. Phys.*, 230:3727–3752, 2011.

[13] Kurt Sebastian and Chi-Wang Shu. Multidomain weno finite difference method with interpolation at subdomain interfaces. *J. Sci. Comput.*, 19:405–438, 2003.

[14] Bruno Costa and Wai Sun Don. Multi-domain hybrid spectral-weno methods for hyperbolic conservation laws. *J. Comput. Phys.*, 224:970–991, 2007.

[15] J. Chao, A. Haselbacher, and S. Balachandar. A massively parallel multi-block hybrid compact-weno scheme for compressible flows. *J. Comput. Phys.*, 228:7473–7491, 2009.

[16] Mark H. Carpenter, Jan Nordstrom, and David Gottlieb. A stable and conservative interface treatment of arbitrary spatial accuracy. *J. Comput. Phys.*, 148:341–365, 1999.

[17] Guang-Shan Jiang and Chi-Wang Shu. Efficient implementation of weighted eno schemes. *J. Comput. Phys.*, 126(1):202–228, 1996.

[18] Chi-Wang Shu. Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws. In *Advanced Numerical Approximation of Nonlinear Hyperbolic Equations*, chapter 4, pages 325–432. Springer, New York, NY, 1998.

[19] B. Gustafsson. The convergence rate for difference approximations to mixed initial boundary value problems. *Math Comput.*, 29:396–406, 1975.

[20] M. Svard and J. Nordstrom. O the order of acurace for difference approximations of initial-boundary value problems. *J. Comput. Phys.*, 218:333–352, 2006.

[21] Mark H. Carpenter, David Gottlieb, and Saul Abarbanel. Time-stable boundary conditions for finite-difference schemes solving hyperbolic systems: methodology and application to high-order compact systems. *J. Comput. Phys.*, 111:220–236, 1994.

[22] Jan Nordstrom and Mark H. Carpenter. Boundary and interface conditions for high order finite difference methods applied to the euler and navier-stokes equations. *J. Comput. Phys.*, 148:621–645, 1999.

[23] Jan Nordstrom and Mark H. Carpenter. High-order finite difference methods, multidimensional linear problems, and curvilinear coordinates. *J. Comput. Phys.*, 173(1):149–174, 2001.

[24] Ken Mattsson, Magnus Svard, Mark Carpenter, and Jan Nordstrom. High-order accurate computations for unsteady aerodynamics. *Comp. Fluids*, 36:636–649, 2007.

[25] Magnus Svard, Mark H. Carpenter, and Jan Nordstrom. A stable high-order finite difference scheme for the compressible navier-stokes equations, far-field boundary conditions. *J. Comput. Phys.*, 225:1020–1038, 2007.

[26] T. Pulliam and D. Chaussee. A diagonal form of an implicit approximate-factorization algorithm. *J. Comput. Phys.*, 39(2):347–363, 1981.

[27] Marsha Berger and Joseph Oliger. Adaptive mesh refinement for hyperbolic partial differential equations. *J. Comput. Phys.*, 53(3):484–512, 1984.

[28] M.J. Berger and P. Colella. Local adaptive mesh refinement for shock hydrodynamics. *J. Comput. Phys.*, 82(1):64–84, 1989.

[29] J.D. Hoffman. *Numerical Methods for Engineers and Scientists*. Marcel Dekker, 2001.

[30] Gary A. Sod. A survey of several finite difference methods for systems of nonlinear hyperbolic conservation laws. *J. Comput. Phys.*, 27:1–31, 1978.

[31] Paul Woodward and Phillip Colella. The numerical simulation of two-dimensional fluid flow with strong shocks. *J. Comput. Phys.*, 54(1):115–173, 1984.