# High-Order Implicit Shock Tracking (HOIST)

Andrew Shi, Per-Olof Persson, and Matthew J. Zahr

**Abstract** We present a framework for resolving discontinuous solutions of conservation laws using *implicit tracking* and a high-order discontinuous Galerkin (DG) discretization. Central to the framework is an optimization problem and associated sequential quadratic programming solver which simultaneously solves for a discontinuity-aligned mesh and the corresponding high-order approximation to the flow that does not require explicit meshing of the *a priori* unknown discontinuity surface. We utilize an error-based objective function that penalizes violation of the DG residual in an enriched test space, which endows the method with $r$-adaptive behavior: mesh nodes move to track discontinuities with element faces and improve the conservation law approximation in smooth regions of the flow. This method is shown to deliver highly accurate solutions on coarse, high-order discretizations without nonlinear stabilization and recover optimal convergence rates $O(h^{p+1})$ for problems with discontinuous solutions. We demonstrate this framework on a series of inviscid steady and unsteady conservation laws, the latter of which using both a space-time and method of lines discretization.

Andrew Shi
Graduate Student, Department of Mathematics, University of California, Berkeley,
e-mail: andrewshi94@berkeley.edu

Per-Olof Persson
Professor, Department of Mathematics, University of California, Berkeley,
e-mail: persson@berkeley.edu

Matthew J. Zahr
Assistant Professor, Department of Aerospace and Mechanical Engineering, University of Notre Dame, e-mail: mzahr@nd.edu

## 1 Introduction and background

It is widely believed that higher fidelity is required for problems with propagating waves, turbulent fluid flow, nonlinear interactions, and multiple scales [33]. This has resulted in a significant interest in high-order accurate methods, such as discontinuous Galerkin (DG) methods [5, 17], which have the potential to produce accurate solutions on coarse meshes. Among the most significant challenges associated with high-order methods is their sensitivity to under-resolved features, in particular for nonlinear problems where the spurious oscillations often cause a breakdown of the numerical solvers. This is exacerbated for problems with shocks where the low dissipation associated with high-order methods is insufficient to stabilize the solution. Since shocks are present in many important problems in fields such as aerospace, astrophysics, and combustion, this poses a fundamental barrier to widespread adoption of these methods.

Most of the techniques for addressing shocks are based on *shock capturing*, that is, the numerical discretization somehow incorporates the discontinuities independently of the computational grid. One simple method is to use a sensor that identifies the mesh elements that contain shocks, and reduce their polynomial degrees [2, 4]. Related, more sophisticated approaches include limiting, such as the weighted essentially non-oscillatory (WENO) schemes [15, 21, 18]. For high-order methods, artificial viscosity has also proven to be competitive, since it can smoothly resolve the jumps in the solution without introducing additional discontinuities between the elements [25]. The main problem with all these approaches is they lead to globally first-order accurate schemes. This can be remedied by local mesh refinement around the shock ($h$-adaptivity) [9], although the anisotropic high-order mesh adaptation is challenging and requires highly refined elements near the shock.

An alternative approach is *shock tracking* or *shock fitting*, where the computational mesh is moved such that its faces are aligned with the discontinuities in the solution. This is natural in the setting of a DG method since the numerical scheme already incorporates jumps between the elements and the approximate Riemann solvers employed on the element faces handle the discontinuities correctly. However, it is a difficult meshing problem since it essentially requires generating a fitted mesh to the (unknown) shock surface. Many previous approaches employ specialized formulations and solvers which are dimension-dependent and do not easily generalize [16, 14, 3] and/or are limited to relatively simple problems [29, 30, 32]. In addition, early approaches to shock fitting have been applied to low-order schemes where the relative advantage over shock capturing is smaller than for high-order methods [31, 1]. Explicit shock tracking strategies [26, 23] have been proposed which are able to attain high-order accuracy in the presence of shocks, but require a specialized strategy to explicitly track the shock separately from the remainder of the flow. These methods are not easily applicable to discontinuities whose topologies not known *a priori*. For these reasons, shock tracking is largely not used in practical CFD today.

In [34, 36], we introduced a novel approach to shock tracking for steady conservation laws (including space-time formulations of time-dependent problems) that does not require explicitly generating a mesh of the unknown discontinuity surface.

Rather, the conservation law is discretized on a mesh without knowledge of the discontinuity surface and an optimization problem is formulated such that its solution is the pair $(\boldsymbol{u}, \boldsymbol{x})$, where $\boldsymbol{x}$ is the positions of the mesh nodes that cause element faces to align with discontinuities in the flow and $\boldsymbol{u}$ is the solution of discretized conservation law on the mesh defined by $\boldsymbol{x}$. That is, discontinuity tracking is implicitly achieved through the solution of an optimization problem and will be referred to as *implicit shock tracking*. While this approach works with any discretization that allows for inter-element discontinuities, we focus on high-order DG methods due to the high degree of accuracy attainable on coarse meshes, proper treatment of discontinuities with approximate Riemann solvers, and the ability to use curved elements to track discontinuities with curvature (Fig. 1). The implicit tracking optimization problem proposed in [36] minimizes the violation of the DG residual in an enriched test space while enforcing that the standard DG (same test and trial space) equations are satisfied. This objective function is a surrogate for violation of the infinite-dimensional weak formulation of the conservation law, which endows the method with $r$-adaptive behavior: it promotes alignment of the mesh with discontinuities and adjusts nodes in smooth region to improve approximation of the conservation law. The optimization problem is solved using a sequential quadratic programming method with a Levenberg-Marquardt Hessian approximation that simultaneously converges the mesh and solution to their optimal values, which never requires the fully converged DG solution on a non-aligned mesh and does not require nonlinear stabilization. The combination of implicit tracking with a DG discretization leads to a high-order accurate numerical method that has been shown to provide accurate approximations to high-speed inert [34, 36] and reacting flows [35].

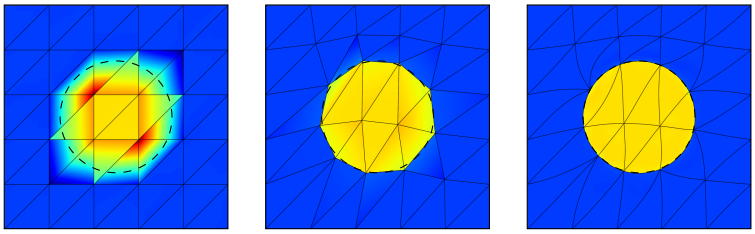


Fig. 1: $L^2$ projection of a piecewise constant function onto a non-aligned (*left*) vs. discontinuity-aligned mesh with linear (*middle*) and cubic (*right*) elements and corresponding solution. See Example 4.1 of [34] for details.

In [27], we further extend the framework developed in [34, 36] for steady conservation laws and space-time formulations of unsteady conservation laws to a method of lines discretization approach. While space-time methods are attractive for a number of reasons, the method of lines approach tends to be more practical for complex problems. The key ingredients of the method of lines approach are: 1) an Arbitrary Lagrangian-Eulerian formulation of the conservation law to handle the deforming mesh (which deforms to track the shock through the domain), 2) semi-discretization

with DG to obtain a system of ordinary differential equations, 3) high-order temporal discretization with a diagonally implicit Runge Kutta (DIRK) method, and 4) implicit shock tracking at each time step following the approach in [34, 36]. We utilize a Rankine-Hugoniot based procedure to construct an initial guess for each timestep which significantly accelerates the performance of the solver and allows to us to achieve Newton-like convergence.

To our knowledge, the only other approach to implicit shock tracking was proposed in [6, 8, 7], where the authors enforce a DG discretization with unconventional numerical fluxes and the Rankine-Hugoniot interface conditions in a minimum-residual sense. Interestingly, enforcement of the interface condition circumvented traditional stability requirements for the DG numerical fluxes, allowing them to solely rely on fluxes interior to an element. Their method was shown to successfully track even complex discontinuity surfaces and provide high-order approximations to the conservation law on traditionally coarse, high-order meshes. Their initial formulation was extended to viscous problems in [20] and recast as a least-square discontinuous Galerkin method in [19].

The remainder of this chapter is organized as follows. Section 2 introduces the governing system of inviscid conservation laws and its discretization using a DG method. Section 3 presents the error-based objective function and the constrained optimization framework in both the steady and unsteady case. Section 4 discusses practical details required for the proposed tracking framework such as initialization of the SQP solver and topological mesh operations to remove small elements. Finally, Section 5 presents a number of numerical experiments that demonstrate the method on a variety of steady and unsteady flows using coarse, high-order meshes and demonstrates high-order convergence of the method. For unsteady flows, we compare the space-time and method of lines approaches and discuss cases when one is more suitable.

## 2 Governing equations and high-order discretization

Consider a general system of $m$ inviscid conservation laws, defined on the fixed domain $\Omega \subset \mathbb{R}^d$ and subject to appropriate boundary conditions,

$$\frac{\partial U}{\partial t} + \nabla \cdot F(U) = 0 \quad \text{in } \Omega \times [0,T] \tag{1}$$

where $U : \Omega \times [0,T] \to \mathbb{R}^m$ is the solution of the system of conservation laws, $F : \mathbb{R}^m \to \mathbb{R}^{m \times d}$ is the flux function, $\nabla \coloneqq (\partial_{x_1}, \ldots, \partial_{x_d})$ is the gradient operator in the physical domain such that $\nabla W(x,t) = \left[\partial_{x_1} W(x,t) \cdots \partial_{x_d} W(x,t)\right] \in \mathbb{R}^{N \times d}$ for any $W : \Omega \times [0,T] \to \mathbb{R}^N$ and $x \in \Omega$, $t \in [0,T]$, and the boundary of the domain $\partial\Omega$ has outward unit normal $n : \partial\Omega \to \mathbb{R}^d$. The conservation law in (1) is supplemented with the initial condition $U(x,0) = \bar{U}(x)$ for all $x \in \Omega$, where $\bar{U} : \Omega \to \mathbb{R}^m$. In general, the solution $U(x)$ may contain discontinuities, in which

case, the conservation law (1) holds away from the discontinuities and the Rankine-Hugoniot conditions [22] hold at discontinuities.

We will construct a high-order numerical method that tracks discontinuities with the computational grid as they evolve through the domain, which places three requirements on the discretization: 1) a high-order, stable, and convergent discretization of the conservation law in (1), 2) employs a solution basis that supports discontinuities between computational cells or elements, and 3) allows for deformation of the computational domain. As such, our method is based on a standard high-order DG-DIRK discretization of an Arbitrary Lagrangian-Eulerian (ALE) formulation of the governing equations. We insist on a high-order discretization given their proven ability [34, 6] to deliver accurate solutions on coarse discretizations provided discontinuities are tracked.

## 2.1 Arbitrary Lagrangian-Eulerian formulation

We use an ALE formulation of the governing equations to account for the time-dependent domain deformations required to track discontinuities as they evolve. To this end, we introduce a time-dependent domain mapping (Fig. 2)

$$\mathcal{G} : \Omega_0 \times [0, T] \to \Omega; \qquad \mathcal{G} : (X, t) \mapsto \mathcal{G}(X, t), \tag{2}$$

where $\Omega_0 \subset \mathbb{R}^d$ is a fixed reference domain, $T$ is the final time, and at each time $t \in [0, T]$, $\mathcal{G}(\,\cdot\,, t) : \Omega_0 \to \Omega$ is a diffeomorphism. We note that the domain $\Omega$ is fixed, i.e., $\Omega$ occupies the same region of $\mathbb{R}^d$ at any time $t \in [0, T]$; the time-dependent diffeomorphism is introduced as an integral part of the proposed numerical method to track discontinuities as they evolve. Following the approach in [24], the conservation

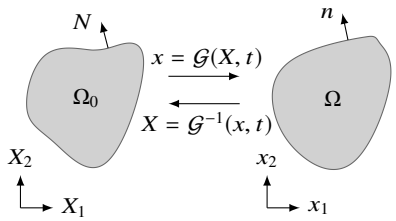



Fig. 2: Mapping between reference and physical domains.

law on the physical domain $\Omega$ is transformed to a conservation law on the reference domain $\Omega_0$

$$\frac{\partial U_X}{\partial t} + \nabla_X \cdot F_X(U_X; G, v) = 0 \quad \text{in } \Omega_0 \tag{3}$$

where $U_X : \Omega_0 \times [0, T] \to \mathbb{R}^m$ is the solution of the transformed conservation law, $F_X : \mathbb{R}^m \times \mathbb{R}^{d\times d} \times \mathbb{R}^d \to \mathbb{R}^{m\times d}$ is the transformed flux function, $\nabla_X \coloneqq (\partial_{X_1}, \ldots, \partial_{X_d})$

is the gradient operator on the reference domain, and the deformation gradient $G : \Omega_0 \times [0,T] \to \mathbb{R}^{d\times d}$, mapping Jacobian $g : \Omega_0 \times [0,T] \to \mathbb{R}$, and mapping velocity $v : \Omega_0 \times [0,T] \to \mathbb{R}^d$ are defined as

$$G = \nabla_X \mathcal{G}, \qquad g = \det G, \qquad v = \frac{\partial \mathcal{G}}{\partial t}. \tag{4}$$

The transformed and physical solutions are related, for any $X \in \Omega_0$ and $t \in [0,T]$, as

$$U_X(X,t) = g(X,t)U(\mathcal{G}(X,t),t) \tag{5}$$

and the transformed flux is defined as

$$F_X : (W_X; \Theta, \xi) \mapsto [(\det\Theta)F((\det\Theta)^{-1}W_X) - W_X \otimes \xi]\Theta^{-T}. \tag{6}$$

The transformed conservation law is supplemented with the initial condition $U_X(X,0) = \bar{U}_X(X)$ for all $X \in \Omega_0$, where $\bar{U}_X : \Omega_0 \to \mathbb{R}^m$ is $\bar{U}_X(X) = g(X,0)\bar{U}(\mathcal{G}(X,0))$.

## 2.2 Discontinuous Galerkin discretization

We use a nodal discontinuous Galerkin method [5, 17] to discretize the transformed conservation law (3). Let $\mathcal{E}_h$ represent a discretization of the reference domain $\Omega_0$ into non-overlapping, potentially curved, computational elements. Additionally, we introduce the DG approximation (trial) space of discontinuous piecewise polynomials associated with the mesh $\mathcal{E}_h$

$$\mathcal{V}_h^p = \left\{ v \in [L^2(\Omega_0 \times [0,T])]^m \;\middle|\; v(\cdot,t)|_K \in [\mathcal{P}_p(K)]^m,\ \forall K \in \mathcal{E}_h,\ t \in [0,T] \right\} \tag{7}$$

where $\mathcal{P}_p(K)$ is the space of polynomial functions of degree at most $p \geq 1$ on the element $K$, and we take the DG test space to be $\mathcal{V}_h^{p'}$, where $p' \geq p$. Finally, we define the space of admissible domain mappings as the space of continuous piecewise polynomials of degree $q$ associated with the mesh $\mathcal{E}_h$

$$\mathcal{W}_h = \left\{ v \in [C^0(\Omega_0 \times [0,T])]^d \;\middle|\; v(\cdot,t)|_K \in [\mathcal{P}_q(K)]^d,\ \forall K \in \mathcal{E}_h,\ t \in [0,T] \right\} \tag{8}$$

Then, the DG residual form $r : \mathcal{V}_h^p \times \mathcal{V}_h^{p'} \times \mathcal{W}_h \to \mathbb{R}$ is given by

$$r_h^{p,p'} : (W_X, \psi_X, \mathcal{Q}) \mapsto \sum_{K \in \mathcal{E}_{h,q}} r_K^{p,p'}(W_X, \psi_X, \mathcal{Q}), \tag{9}$$

where the elemental DG form $r_K^{p,p'} : \mathcal{V}_h^p \times \mathcal{V}_h^{p'} \times \mathcal{W}_h \to \mathbb{R}$ is given by

$$r_K^{p,p'} : (W_X, \psi_X, Q) \mapsto \int_K \psi_X \cdot \dot{W}_X \, dV + \int_{\partial K} \psi_X^+ \cdot \mathcal{H}_X(W_X^+, W_X^-, N; \nabla_X Q, \dot{Q}) \, dS - \int_K F_X(W_X; \nabla_X Q, \dot{Q}) : \nabla_X \psi_X \, dV, \tag{10}$$

where $N$ is the outward normal to the surface $\partial K$, $\mathcal{H}_X : \mathbb{R}^m \times \mathbb{R}^m \times \mathbb{R}^d \times \mathbb{R}^{d\times d} \times \mathbb{R}^d \to \mathbb{R}^m$ is the numerical flux function associated with the reference inviscid flux $F_X$, and $W_X^+$ ($W_X^-$) is the interior (exterior) trace of $W_X$ to element $K$; for points $X \in \partial K \cap \partial\Omega_0$, $W_X^-$ is a boundary state constructed to enforce the appropriate boundary condition. In this work, we take the numerical flux to be a smoothed version of the Roe flux (Section 2.3 of [36]).

Next, we introduce a (nodal) basis over each element for the trial space ($\mathcal{V}_h^p$), test space ($\mathcal{V}_h^{p'}$), and mapping space ($\mathcal{W}_h$) to reduce (10) to a system of ordinary differential equations (ODEs) in residual form. In the case where $p' = p$, we denote the residual $\boldsymbol{r}^{\text{uns}} : \mathbb{R}^{N_{\boldsymbol{u}}} \times \mathbb{R}^{N_{\boldsymbol{u}}} \times \mathbb{R}^{N_{\boldsymbol{x}}} \times \mathbb{R}^{N_{\boldsymbol{x}}} \to \mathbb{R}^{N_{\boldsymbol{u}}}$, which is defined as

$$\boldsymbol{r}^{\text{uns}} : (\mathring{\boldsymbol{w}}, \boldsymbol{w}, \mathring{\boldsymbol{y}}, \boldsymbol{y}) \mapsto \boldsymbol{m}\mathring{\boldsymbol{w}} + \boldsymbol{f}(\boldsymbol{w}, \boldsymbol{y}, \mathring{\boldsymbol{y}}), \tag{11}$$

where $N_{\boldsymbol{u}} = \dim \mathcal{V}_h^p$, $N_{\boldsymbol{x}} = \dim \mathcal{W}_h$, $\boldsymbol{m} \in \mathbb{R}^{N_{\boldsymbol{u}} \times N_{\boldsymbol{u}}}$ is the mass matrix associated with the test/trial space $\mathcal{V}_h^p$, and $\boldsymbol{f} : \mathbb{R}^{N_{\boldsymbol{u}}} \times \mathbb{R}^{N_{\boldsymbol{x}}} \times \mathbb{R}^{N_{\boldsymbol{x}}} \to \mathbb{R}^{N_{\boldsymbol{u}}}$ is the algebraic form of the second and third terms in (10). In this notation, the standard DG discretization reads: given $\boldsymbol{x} : [0, T] \to \mathbb{R}^{N_{\boldsymbol{x}}}$, find $\boldsymbol{u} : [0, T] \to \mathbb{R}^{N_{\boldsymbol{u}}}$ such that

$$\boldsymbol{r}^{\text{uns}}(\dot{\boldsymbol{u}}(t), \boldsymbol{u}(t), \dot{\boldsymbol{x}}(t), \boldsymbol{x}(t)) = \boldsymbol{0}, \quad \boldsymbol{u}(0) = \bar{\boldsymbol{u}}, \quad \boldsymbol{x}(0) = \bar{\boldsymbol{x}}, \tag{12}$$

for all $t \in [0, T]$, where $\boldsymbol{u}$ is the time-dependent coefficients of the DG solution, $\boldsymbol{x}$ is the time-dependent coefficients of the domain mapping (nodal coordinates of the mesh), $\bar{\boldsymbol{u}} \in \mathbb{R}^{N_{\boldsymbol{u}}}$ is the algebraic representation of the initial condition $\bar{U}_X$, and $\bar{\boldsymbol{x}} \in \mathbb{R}^{N_{\boldsymbol{x}}}$ is an initial condition for the nodal coordinates. Typically, the evolution of the mesh coordinates $\boldsymbol{x}(t)$ is known analytically or governed by a dynamical system (e.g., fluid-structure interaction); however, in this work, it will be determined as the solution of an optimization problem (after temporal discretization) such that discontinuities are tracked over time.

Finally, we use the expansions in the nodal bases to define the *enriched residual* $\boldsymbol{R}^{\text{uns}} : \mathbb{R}^{N_{\boldsymbol{u}}} \times \mathbb{R}^{N_{\boldsymbol{u}}} \times \mathbb{R}^{N_{\boldsymbol{x}}} \times \mathbb{R}^{N_{\boldsymbol{x}}} \to \mathbb{R}^{N'_{\boldsymbol{u}}}$ associated with a trial space of degree $p'$ as

$$\boldsymbol{R}^{\text{uns}} : (\mathring{\boldsymbol{w}}, \boldsymbol{w}, \mathring{\boldsymbol{y}}, \boldsymbol{y}) \mapsto \boldsymbol{M}\mathring{\boldsymbol{w}} + \boldsymbol{F}(\boldsymbol{w}, \boldsymbol{y}, \mathring{\boldsymbol{y}}), \tag{13}$$

where $N'_u = \dim \mathcal{V}_h^{p'}$, $\boldsymbol{M} \in \mathbb{R}^{N'_{\boldsymbol{u}} \times N_{\boldsymbol{u}}}$ is the mass matrix associated with the test space $\mathcal{V}_h^{p'}$ and trial space $\mathcal{V}_h^p$, and $\boldsymbol{F} : \mathbb{R}^{N_{\boldsymbol{u}}} \times \mathbb{R}^{N_{\boldsymbol{x}}} \times \mathbb{R}^{N_{\boldsymbol{x}}} \to \mathbb{R}^{N'_{\boldsymbol{u}}}$ is the algebraic form of the second and third terms in (10). In this work, we take $p' = p + 1$, but other choices are possible as well. The enriched residual will be used in the subsequent sections to define the implicit tracking objective function.

## 2.3 High-order temporal discretization

Proceeding with the method of lines, we discretize the dynamical system in (12) using a diagonally implicit Runge-Kutta method (DIRK) to yield a sequence of algebraic systems of equations. Unlike fully implicit Runge-Kutta methods, an $s$-stage DIRK scheme has a Butcher tableau $(A, b, c) \in \mathbb{R}^{s\times s} \times \mathbb{R}^s \times \mathbb{R}^s$ where $A$ is lower triangular (Table 1). As a result, the $i$th stage only depends on the solution for stages $1, \dots, i$,

Table 1: Butcher tableau for DIRK schemes

$$\begin{array}{c|c} \boldsymbol{c} & \boldsymbol{A} \\ \hline & \boldsymbol{b}^T \end{array} \quad = \quad \begin{array}{c|ccc} c_1 & a_{11} & & \\ \vdots & \vdots & \ddots & \\ c_s & a_{s1} & \dots & a_{ss} \\ \hline & b_1 & \dots & b_s \end{array}$$

allowing the stages to be solved sequentially. We partition the time interval $[0, T]$ into $N_T$ intervals of equal size $\Delta t = T/N_T$ with endpoints $\{t_n\}_{n=0}^{N_T}$, where $t_0 = 0$ and $t_n = t_{n-1} + \Delta t$ for $n = 1, \dots, N_T$. An $s$-stage DIRK discretization of (12) with Butcher tableau $(A, b, c)$ reads: for $n = 1, \dots, N_T$ and $i = 1, \dots, s$,

$$\begin{aligned} &\boldsymbol{u}_0 = \bar{\boldsymbol{u}}, \quad \boldsymbol{u}_{n+1} = \boldsymbol{u}_n + \sum_{j=1}^{s} b_i \boldsymbol{k}^{\boldsymbol{u}}_{n,j}, \quad \boldsymbol{u}_{n,i} = \boldsymbol{u}_n + \sum_{j=1}^{i} a_{ij} \boldsymbol{k}^{\boldsymbol{u}}_{n,j} \\ &\boldsymbol{x}_0 = \bar{\boldsymbol{x}}, \quad \boldsymbol{x}_{n+1} = \boldsymbol{x}_n + \sum_{j=1}^{s} b_i \boldsymbol{k}^{\boldsymbol{x}}_{n,j}, \quad \boldsymbol{x}_{n,i} = \boldsymbol{x}_n + \sum_{j=1}^{i} a_{ij} \boldsymbol{k}^{\boldsymbol{x}}_{n,j} \\ &\boldsymbol{m}\boldsymbol{k}^{\boldsymbol{u}}_{n,i} = -\Delta t \boldsymbol{f}\left(\boldsymbol{u}_{n,i}, \boldsymbol{x}_{n,i}, \boldsymbol{v}_{n,i}\right), \quad \boldsymbol{k}^{\boldsymbol{x}}_i = \Delta t \boldsymbol{v}_{n,i} \end{aligned} \tag{14}$$

where $\boldsymbol{u}_0, \boldsymbol{u}_n, \boldsymbol{u}_{n,i}, \boldsymbol{k}^{\boldsymbol{u}}_{n,i} \in \mathbb{R}^{N_{\boldsymbol{u}}}$ and $\boldsymbol{x}_0, \boldsymbol{x}_n, \boldsymbol{x}_{n,i}, \boldsymbol{k}^{\boldsymbol{x}}_{n,i} \in \mathbb{R}^{N_{\boldsymbol{x}}}$ are implicitly defined as the solution of (14); $\boldsymbol{u}_n \approx \boldsymbol{u}(t_n)$ and $\boldsymbol{x}_n \approx \boldsymbol{x}(t_n)$ are the state and mesh approximation at each time step $n = 0, \dots, T$; $\boldsymbol{u}_{n,i} \approx \boldsymbol{u}(t_n + c_i\Delta t)$ and $\boldsymbol{x}_{n,i} \approx \boldsymbol{x}(t_n + c_i\Delta t)$ are the state and mesh approximations at each stage $i = 1, \dots, s$ of each time interval $n = 1, \dots, T$; $\boldsymbol{k}^{\boldsymbol{u}}_{n,i}$ and $\boldsymbol{k}^{\boldsymbol{x}}_{n,i}$ are the solution and mesh stage updates, and $\boldsymbol{v}_{n,i} \in \mathbb{R}^{N_{\boldsymbol{x}}}$ is a stage-consistent mesh velocity [11, 27] defined as

$$\boldsymbol{v}_{n,i} = \sum_{j=1}^{i} (\boldsymbol{A}^{-1})_{ij} \frac{\boldsymbol{x}_{n,j} - \boldsymbol{x}_n}{\Delta t}, \tag{15}$$

i.e., the mesh velocity is consistent with the DIRK scheme and the mesh position at the stages.

We convert the modified DIRK system in (14) to residual form $\boldsymbol{r}_{n,i} : \mathbb{R}^{N_{\boldsymbol{u}}} \times \mathbb{R}^{N_{\boldsymbol{x}}} \to \mathbb{R}^{N_{\boldsymbol{u}}}$ at a fixed step $n \in \{1, \dots, N_T\}$ and stage $i \in \{1, \dots, s\}$ as

$$\boldsymbol{r}_{n,i}^{\text{uns}} : (\boldsymbol{w}, \boldsymbol{y}) \mapsto \boldsymbol{m}\boldsymbol{\xi}_{n,i}(\boldsymbol{w}) + \Delta t \boldsymbol{f}(\boldsymbol{w}, \boldsymbol{y}, \boldsymbol{\zeta}_{n,i}(\boldsymbol{y})), \tag{16}$$

where $\boldsymbol{\xi}_{n,i} : \mathbb{R}^{N_{\boldsymbol{u}}} \to \mathbb{R}^{N_{\boldsymbol{u}}}$ maps the state stage ($\boldsymbol{u}_{n,i}$) to the corresponding stage update ($\boldsymbol{k}^{\boldsymbol{u}}_{n,i}$)

$$\boldsymbol{\xi}_{n,i} : \boldsymbol{w} \mapsto (A^{-1})_{ii}(\boldsymbol{w} - \boldsymbol{u}_n) + \sum_{j=1}^{i-1} (A^{-1})_{ij}(\boldsymbol{u}_{n,j} - \boldsymbol{u}_n) \tag{17}$$

and $\boldsymbol{\zeta}_{n,i} : \mathbb{R}^{N_{\boldsymbol{x}}} \to \mathbb{R}^{N_{\boldsymbol{x}}}$ maps the mesh stage ($\boldsymbol{x}_{n,i}$) to the corresponding stage-consistent velocity ($\boldsymbol{v}_{n,i}$) as

$$\boldsymbol{\zeta}_{n,i} : \boldsymbol{y} \mapsto (A^{-1})_{ii} \frac{\boldsymbol{y} - \boldsymbol{x}_n}{\Delta t} + \sum_{j=1}^{i-1} (A^{-1})_{ij} \frac{\boldsymbol{x}_{n,j} - \boldsymbol{x}_n}{\Delta t}. \tag{18}$$

Similarly, we define the corresponding fully discrete enriched residual function $\boldsymbol{R}_{n,i} : \mathbb{R}^{N_{\boldsymbol{u}}} \times \mathbb{R}^{N_{\boldsymbol{x}}} \to \mathbb{R}^{N'_{\boldsymbol{u}}}$ as

$$\boldsymbol{R}_{n,i}^{\text{uns}} : (\boldsymbol{w}, \boldsymbol{y}) \mapsto \boldsymbol{M}\boldsymbol{\xi}_{n,i}(\boldsymbol{w}) + \Delta t \boldsymbol{F}(\boldsymbol{w}, \boldsymbol{y}, \boldsymbol{\zeta}_{n,i}(\boldsymbol{y})), \tag{19}$$

which we use in next section to define the objective function of the implicit tracking optimization problem. Notice that only the *spatial* test space is enriched in the definition of $\boldsymbol{R}_{n,i}^{\text{uns}}$ relative to $\boldsymbol{r}_{n,i}^{\text{uns}}$; the temporal discretization in $\boldsymbol{r}_{n,i}^{\text{uns}}$ and $\boldsymbol{R}_{n,i}^{\text{uns}}$ is identical.

In this work, we consider a third-order $L$-stable DIRK scheme with three stages, which we denote by DIRK3. The Butcher tableau is given in Table 2.

Table 2: Butcher tableau for DIRK3, where $\beta = 0.435866521508459$, $\gamma = -\frac{6\beta^2 - 16\beta + 1}{4}$, and $\omega = \frac{6\beta^2 - 20\beta + 5}{4}$.

$$\begin{array}{c|cc} \beta & \beta & \\ \frac{1+\beta}{2} & \frac{1+\beta}{2} - \beta & \beta \\ \gamma + \omega + \beta & \gamma & \omega \ \beta \\ \hline & \gamma & \omega \ \beta \end{array}$$

## 2.4 Special case: reduction to steady conservation law

We close this section by noting that this discretization procedure simplifies considerably for the special case of the steady conservation law, which is obtained by dropping the time derivative term in (1). In this setting, the mapping (2) is independent of time so all terms depending on the mapping velocity vanish. Furthermore, the transformed state $U_X$ does not need to account for the mapping Jacobian, which

further simplifies the definition of the transformed flux. In the steady setting, the DG discretization leads to an algebraic residual form $\boldsymbol{r}^{\text{ste}} : \mathbb{R}^{N_{\boldsymbol{u}}} \times \mathbb{R}^{N_{\boldsymbol{x}}} \to \mathbb{R}^{N_{\boldsymbol{u}}}$ and enriched residual form $\boldsymbol{R}^{\text{ste}} : \mathbb{R}^{N_{\boldsymbol{u}}} \times \mathbb{R}^{N_{\boldsymbol{x}}} \to \mathbb{R}^{N'_{\boldsymbol{u}}}$. A standard DG discretization reads: given $\boldsymbol{x} \in \mathbb{R}^{N_{\boldsymbol{x}}}$, find $\boldsymbol{u} \in \mathbb{R}^{N_{\boldsymbol{u}}}$ such that $\boldsymbol{r}(\boldsymbol{u}, \boldsymbol{x}) = \boldsymbol{0}$. See Section 2 of [36] for details.

## 3 Implicit shock tracking

In this section, we present the implicit tracking framework that poses an optimization problem over the discrete solution and mesh that aims to align element faces with discontinuities in the solution. First, we introduce the optimization formulation for steady conservation laws and extend it to the unsteady case (method of lines discretization) by "solving a steady problem at each timestep".

### 3.1 Steady case

We formulate the problem of tracking discontinuities as a constrained optimization problem over the PDE state and coordinates of the mesh nodes that minimizes an objective function, $f : \mathbb{R}^{N_{\boldsymbol{u}}} \times \mathbb{R}^{N_{\boldsymbol{x}}} \to \mathbb{R}$, while enforcing the DG discretization of the conservation law. That is, we define $\boldsymbol{u} \in \mathbb{R}^{N_{\boldsymbol{u}}}$ and $\boldsymbol{x} \in \mathbb{R}^{N_{\boldsymbol{x}}}$ as

$$(\boldsymbol{u}, \boldsymbol{x}) \coloneqq \underset{\boldsymbol{w} \in \mathbb{R}^{N_{\boldsymbol{u}}}, \boldsymbol{y} \in \mathbb{R}^{N_{\boldsymbol{x}}}}{\arg\min} f(\boldsymbol{w}, \boldsymbol{y}) \quad \text{subject to:} \quad \boldsymbol{r}^{\text{ste}}(\boldsymbol{w}, \boldsymbol{y}) = \boldsymbol{0}. \tag{20}$$

The objective function is constructed such that the solution of the optimization problem $(\boldsymbol{u}, \boldsymbol{x})$ is a mesh $(\boldsymbol{x})$ that aligns element faces with discontinuities in the DG solution $(\boldsymbol{u})$. The optimization-based tracking method directly inherits the benefits of standard DG methods, i.e., high-order accuracy and conservation, due to the constraint that exactly enforces the DG discretization. Finally, the proposed optimization-based tracking method is nonlinearly stable if all discontinuities are successfully tracked, i.e., additional stabilization (e.g., limiting, artificial viscosity) is not required, which we demonstrate in our numerical experiments in Section 5.

We propose an objective function that consists of two terms: one term penalizes a measure of the DG solution error $f_{\text{err}} : \mathbb{R}^{N_{\boldsymbol{u}}} \times \mathbb{R}^{N_{\boldsymbol{x}}} \to \mathbb{R}$ and the other term penalizes distortion of the mesh $f_{\text{msh}} : \mathbb{R}^{N_{\boldsymbol{x}}} \to \mathbb{R}$, i.e.,

$$f : (\boldsymbol{w}, \boldsymbol{y}) \mapsto f_{\text{err}}(\boldsymbol{w}, \boldsymbol{y}) + \kappa^2 f_{\text{msh}}(\boldsymbol{y}), \tag{21}$$

where $\kappa \in \mathbb{R}_+$ is a parameter that weights the contribution of the two terms. Since a piecewise polynomial solution on an aligned mesh will have much lower error than on a non-aligned mesh, $f_{\text{err}}$ promotes alignment of the mesh with discontinuities while $f_{\text{msh}}$ prevents the mesh from entangling or becoming unacceptably skewed.

For the error-like tracking term, we use the norm of the DG residual corresponding to an enriched *test space*, i.e.,

$$f_{\text{err}} : (\boldsymbol{w}, \boldsymbol{y}) \mapsto \frac{1}{2}\boldsymbol{R}^{\text{ste}}(\boldsymbol{w}, \boldsymbol{y})^T \boldsymbol{R}^{\text{ste}}(\boldsymbol{w}, \boldsymbol{y}), \tag{22}$$

where we enrich the test space using polynomials of one degree higher than the trial space. This follows on a large body of work that uses residual-based error indicators to drive $h$-, $p$-, and $r$-adaptivity [10]. Furthermore, it was shown to lead to a robust and reliable tracking framework [36].

To maintain a high-quality mesh, we define $f_{\text{msh}}$ as the distortion of the physical mesh relative to the distortion of the reference mesh

$$f_{\text{msh}} : \boldsymbol{y} \mapsto \frac{1}{2}(\boldsymbol{R}_{\text{msh}}(\boldsymbol{y}) - \boldsymbol{R}_{\text{msh}}(\boldsymbol{X}))^T(\boldsymbol{R}_{\text{msh}}(\boldsymbol{y}) - \boldsymbol{R}_{\text{msh}}(\boldsymbol{X})), \tag{23}$$

where $\boldsymbol{R}_{\text{msh}} : \mathbb{R}^{N_{\boldsymbol{x}}} \to \mathbb{R}^{|\mathcal{E}_{h,q}|}$ is the algebraic system corresponding to the elemental mesh distortion $r_K^{\text{msh}} : \mathcal{W}_h \to \mathbb{R}$, commonly used for high-order mesh generation [13], defined as

$$r_{\text{msh}}^K : Q \mapsto \int_K \left( \frac{\|\nabla_X Q\|_F^2}{d(\det \nabla_X Q)_+^{2/d}} \right)^2 dv \tag{24}$$

and $\boldsymbol{X} \in \mathbb{R}^{N_{\boldsymbol{x}}}$ is the nodal coordinates of the reference mesh.

The optimization problem in (20) is solved using the sequential quadratic programming (SQP) method proposed in [36] that defines the sequences $\{\boldsymbol{u}^{(k)}\}_{k=0}^{\infty} \subset \mathbb{R}^{N_{\boldsymbol{u}}}$ and $\{\boldsymbol{x}^{(k)}\}_{k=0}^{\infty} \subset \mathbb{R}^{N_{\boldsymbol{x}}}$ as

$$\boldsymbol{u}^{(k+1)} = \boldsymbol{u}^{(k)} + \alpha_{k+1}\Delta\boldsymbol{u}^{(k+1)}, \qquad \boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} + \alpha_{k+1}\Delta\boldsymbol{x}^{(k+1)}, \tag{25}$$

for $k = 0, 1, \ldots$, where $\alpha_{k+1} \in (0, 1]$ is a step length determined via a line search applied to the $\ell_1$ merit function, and $\Delta\boldsymbol{u}^{(k+1)} \in \mathbb{R}^{N_{\boldsymbol{u}}}$ and $\Delta\boldsymbol{x}^{(k+1)} \in \mathbb{R}^{N_{\boldsymbol{x}}}$ are search directions. At a given iteration $k$, the search directions $\Delta\boldsymbol{u}^{(k)}$ and $\Delta\boldsymbol{x}^{(k)}$ are computed simultaneously as the solution of a quadratic approximation to the optimization problem in (20) with a regularized Levenberg-Marquardt approximation of the Hessian; for details, see section 4 of [36].

### 3.2 Unsteady case (method of lines)

We extend the steady implicit shock tracking framework in Section 3.2 to time-dependent problems discretized using a method of lines approach (in contrast to the space-time approach in [6, 36]). To begin, we formulate the problem of tracking discontinuities at a given stage $i$ and timestep $n$ as a constrained optimization problem over the PDE state and mesh stage that minimizes an objective function, $f_{n,i} : \mathbb{R}^{N_{\boldsymbol{u}}} \times \mathbb{R}^{N_{\boldsymbol{x}}} \to \mathbb{R}$, while enforcing the DG-DIRK discretization. That is, we define

$\boldsymbol{u}_{n,i} \in \mathbb{R}^{N_{\boldsymbol{u}}}$ and $\boldsymbol{x}_{n,i} \in \mathbb{R}^{N_{\boldsymbol{x}}}$ as

$$(\boldsymbol{u}_{n,i}, \boldsymbol{x}_{n,i}) \coloneqq \underset{\boldsymbol{w} \in \mathbb{R}^{N_{\boldsymbol{u}}}, \boldsymbol{y} \in \mathbb{R}^{N_{\boldsymbol{x}}}}{\arg\min} f_{n,i}(\boldsymbol{w}, \boldsymbol{y}) \quad \text{subject to:} \quad \boldsymbol{r}^{\text{uns}}_{n,i}(\boldsymbol{w}, \boldsymbol{y}) = \boldsymbol{0}. \tag{26}$$

for $n = 1, \ldots, N_T$ and $i = 1, \ldots, s$. As in the steady case, the objective function is constructed such that the solution of the PDE-constrained optimization problem is a feature-aligned mesh by using the norm of the enriched DG-DIRK residual

$$f_{n,i} : (\boldsymbol{w}, \boldsymbol{y}) \mapsto \frac{1}{2} \left\| \boldsymbol{R}^{\text{uns}}_{n,i}(\boldsymbol{w}, \boldsymbol{y}) \right\|_2^2. \tag{27}$$

This choice of optimization formulation *exactly matches the form* of the optimization problem in (20), and therefore, we use the same SQP solver to compute the solution of (26). For a fixed time step $n$, once the stage states $\{\boldsymbol{u}_{n,i}\}_{i=1}^s$ and meshes $\{\boldsymbol{x}_{n,i}\}_{i=1}^s$ are computed, the state and mesh can be advanced to the next time step ($\boldsymbol{u}_{n+1}$ and $\boldsymbol{x}_{n+1}$) using the relationships in (14). The DIRK schemes considered in this work (Table 1) satisfy the property that $A_{si} = b_i$ for $i = 1, \ldots, s$, which implies the state and mesh at the final stage of time step $n$ are identical to the state and mesh at time step $n + 1$, i.e., $\boldsymbol{u}_{n+1} = \boldsymbol{u}_{n,s}$ and $\boldsymbol{x}_{n+1} = \boldsymbol{x}_{n,s}$.

Unlike the steady case (21), we do not include the mesh quality term in the objective function for the unsteady (method of lines) case. In the steady case, there is no information about the shock location *a priori*, which usually requires significant deformation of the initial mesh to align with shocks, necessitating the use of $f_{\text{msh}}$ (23). However, in the context of timestepping, we have useful information from the previous timestep to use as an initial guess for both the state and mesh. We can combine this information along with the Rankine-Hugoniot conditions and a high-order mesh smoothing procedure to obtain an excellent initial guess for the mesh at each Runge-Kutta stage (Section 4.1.2). In a sense, the additional time dimension allows the mesh regularization to be decoupled from the implicit shock tracking procedure, which is one advantage of the method of lines discretization over the space-time formulation.

# 4 Practical considerations

## 4.1 Solution and mesh initialization

The implicit shock tracking optimization problems in (20) and (26) are non-convex and therefore the initial guess for the SQP solver is critical to obtain a good solution. In the present context, this means we must provide a reasonable initial guess for the mesh coordinates $\boldsymbol{x}^{(0)}$ and DG solution $\boldsymbol{u}^{(0)}$.

#### 4.1.1 Steady case

The mesh is always initialized from the reference mesh which comes from mesh generation without knowledge of the discontinuity surface. To avoid local minima in the optimization problem (20) that arise when high-order meshes are used, we usually initialize the tracking problem for $p \geq 0$, $q > 1$ (solution space of degree $p$ and mesh of degree $q$) from the solution of the tracking problem for $p' \leq p$, $q' = 1$ because we have observed that $q = 1$ tracking is quite robust (Section 5.1.1 of [36]) and convergence of the $q > 1$ solution from a straight-sided tracking mesh is rapid.

#### 4.1.2 Unsteady case (method of lines)

In the method of lines setting where implicit shock tracking is performed at each time step, an obvious idea for the initial guess of the optimization problem (26) is the converged mesh and solution from the previous time $t_n$. In practice, this is too far off to attain good convergence properties except for prohibitively small choices of timestep $\Delta t$. Instead, we employ an initial guess where we advect each node on the discontinuity surface from the previous time by the instantaneous shock speed determined by the Rankine-Hugoniot conditions. The remaining nodes are updated using standard optimization-based mesh smoothing using the high-order mesh distortion metric (24). This provides a good initial guess for the shock-aligned mesh for each stage of the optimization problem (Fig. 3). In general, we will need a regularized distortion metric following a procedure described in [12] to handle tangled meshes.

For the initial guess for the solution, we use the converged physical solution ($U$) from the previous timestep. However, since we applied our DG discretization to the transformed conservation law (3), we are solving for the reference solution ($U_X$). Therefore, to use the physical solution at time $t_n$ as the initial guess for time $t_{n+1}$, we multiply it by the ratio of the Jacobian of the initial guess for the mapping at time $t_{n+1}$ (advection based on Rankine-Hugoniot conditions and smoothing) to the Jacobian of the converged mapping at time $t_n$. Future work to enhance the robustness of these initial guesses might consider more advanced methods to approximate the shock velocity [26] and extrapolation based estimates for the solution.

### 4.2 Edge collapses and solution transfer

As we will show in our numerical experiments in Section 5, for the steady case we typically start with an initial mesh that is far from alignment with the shock. In general, this requires large deformations of the initial mesh, which can result in severely ill-conditioned elements and drastically degrade the quality of our solution. To address this, we follow the approach in [6, 36] and collapse elements once they become problematic. In particular, after each SQP iteration, we tag elements for

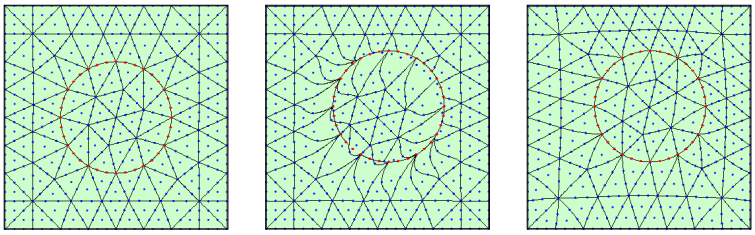


Fig. 3: Shock nodes (·····) (*left*), advected by instantaneous shock speed (*middle*) and smoothed mesh (*right*).

removal if their volume is smaller than some fraction (20% in this work) of their initial volume and remove them by collapsing their shortest edge and updating the connectivity of the mesh (Fig. 4). Further details of this edge collapse procedure and corresponding solution transfer are described in Section 5.2 of [36].

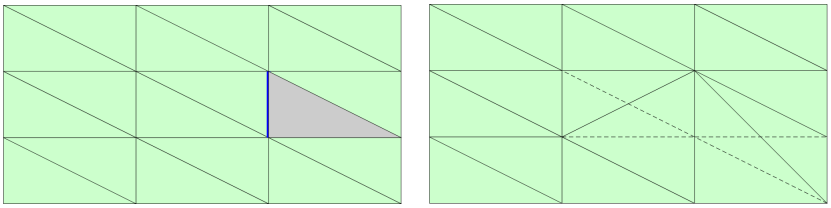


Fig. 4: Demonstration of edge collapse algorithm: the element identified in the original mesh (*left*) is collapsed along the highlighted edge to produce the new mesh (*right*) with the original elements shown in dashed lines for reference.

## 5 Numerical experiments

In this section, we demonstrate the implicit shock tracking method for a number of nonlinear conservation laws. For the unsteady problems, we apply implicit shock tracking in the space-time and method of lines settings, and compare the two approaches.

### 5.1 1D time-dependent, inviscid Burgers' equation

We consider the time-dependent, inviscid Burgers' equation in one spatial dimensions that governs nonlinear advection of a scalar quantity through the one-dimensional

spatial domain $\Omega = [-1, 1]$

$$\begin{aligned} \frac{\partial}{\partial t}U(x,t) + \frac{\partial}{\partial x}\left(\frac{1}{2}U(x,t)^2\right) &= 0 \qquad \text{for } x \in \Omega,\ t \in [0,T] \\ U(-1,t) &= 0 \qquad \text{for } t \in [0,T] \\ U(x,0) &= \bar{U}(x) \quad \text{for } x \in \Omega \end{aligned} \tag{28}$$

where $U : \Omega \times [0,T] \to \mathbb{R}$ is the conserved quantity implicitly defined as the solution of (28), $T = 1$ is the final time, $\bar{U} : \Omega \to \mathbb{R}$ is the initial condition, defined as

$$\bar{U} : x \mapsto 2(x+1)^2(1 - H(x)), \tag{29}$$

and $H : \mathbb{R} \to \{0, 1\}$ is the Heaviside function.

### 5.1.1 Space-time approach

First, we consider a space-time formulation of (28) that takes the form of a steady conservation law (Eqn. (1) without the time derivative term) over the space-time domain $\bar{\Omega} \coloneqq \Omega \times [0,T]$ with flux function $F : \mathbb{R} \to \mathbb{R}^{1\times 2}$, defined as

$$F : W \mapsto \left[\tfrac{1}{2}W^2 \ W\right]. \tag{30}$$

The implicit tracking procedure is initialized from a structured mesh of 64 space-time simplex elements; the polynomial degree of the solution ($p$) and mesh ($q$) are varied in $p = q \in \{1, 2, 3\}$. Even though the initial mesh is far from tracking the discontinuity (some faces are nearly orthogonal to the discontinuity, rather than parallel to it), our method tracks a faceted approximation ($p = q = 1$) to the discontinuity in only 40 iterations, requiring 7 element collapses (Figure 5). The mesh and solution for the high-order elements ($p = q > 1$) are initialized from the $p = q = 1$ tracking mesh and solution. These high-order approximations provide high-quality approximations of the discontinuous space-time solution on the coarse mesh (Figure 6).

### 5.1.2 Method of lines approach

Next, we consider (28) in a method of lines setting where we apply the DG discretization in space (Section 2.2), DIRK discretization in time (Section 2.3) and solve a steady implicit tracking problem at each time step (Section 3.2). That is, instead of solving a tracking problem over the two-dimensional space-time domain $\bar{\Omega}$ (all of space and time coupled), we solve a sequence of one-dimensional tracking problems, each one corresponding to an instant in time. This approach only requires a mesh of the reference domain $\Omega_0 = [-1, 1]$, which we construct such that an element interface lies at the initial shock location ($x = 0$), i.e., the shock in the initial condition is tracked. The shock tracking solution is computed using a DG discretization on this

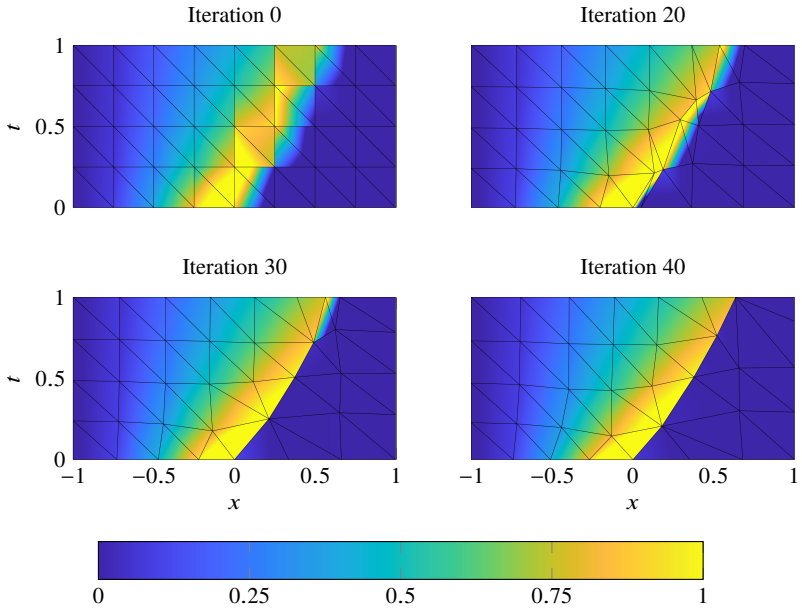



Fig. 5: Space-time solution of one-dimensional, inviscid Burgers' equation using the tracking method at various iterations throughout the solution procedure using a $p = q = 1$ basis for the solution and mesh.

mesh with 20 elements of degree $p = 3$, $q = 1$ and a DIRK3 temporal discretization with 20 time steps (Figure 7).

## 5.2 2D time-dependent, inviscid Burgers' equation

Next, we consider the time-dependent, inviscid Burgers' equation in two spatial dimensions that governs nonlinear advection of a scalar quantity through the two-dimensional spatial domain $\Omega = [-1, 1]^2$

$$\begin{aligned} \frac{\partial}{\partial t} U(x,t) + \frac{\partial}{\partial x_j}\left(\frac{1}{2} U(x,t)^2 \beta_j\right) &= 0 && \text{for } x \in \Omega,\ t \in [0,T] \\ U(x,t) &= 0 && \text{for } x \in \partial\Omega,\ t \in [0,T] \\ U(x,0) &= \bar{U}(x) && \text{for } x \in \Omega \end{aligned} \tag{31}$$

where $U : \Omega \times [0,T] \to \mathbb{R}$ is the conserved quantity implicitly defined as the solution of (31), $\beta = (1,0)$ is the flow direction, $T = 2$ is the final time, $\bar{U} : \Omega \to \mathbb{R}$ is the initial condition, defined as

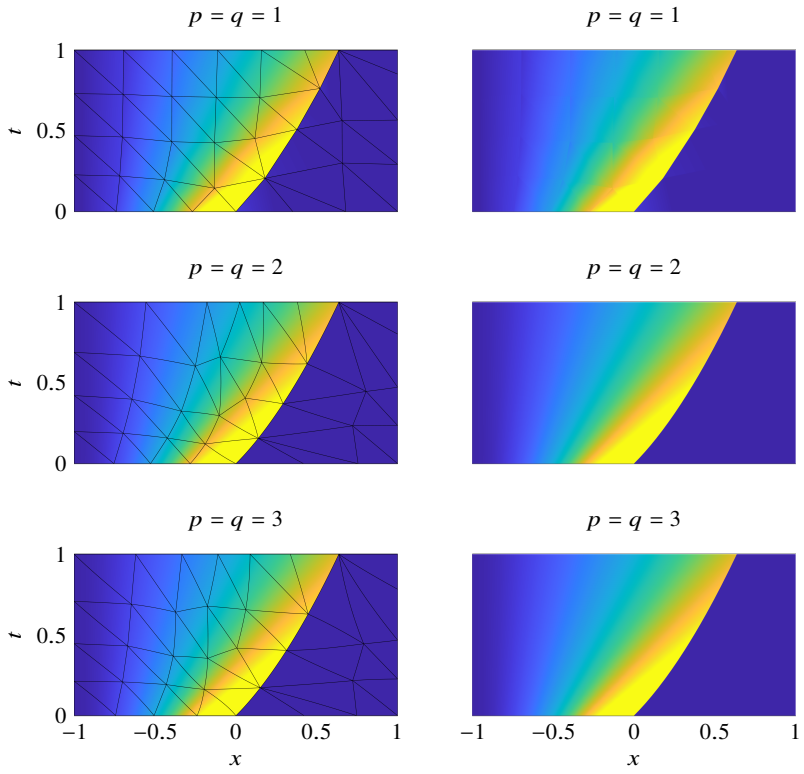



Fig. 6: Space-time solution of one-dimensional, inviscid Burgers' equation using the implicit tracking method with a $p = q = 1$ (*top*), $p = q = 2$ (*middle*), and $p = q = 3$ (*bottom*) basis for the solution and mesh with (*left*) and without (*right*) element boundaries.

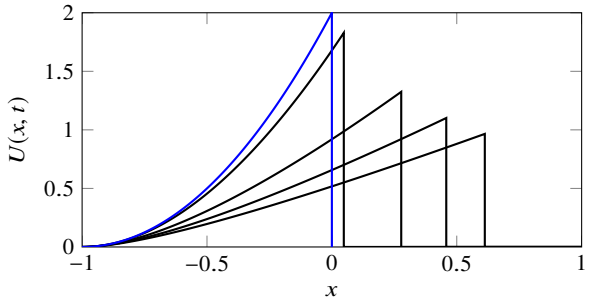



Fig. 7: Method of lines solution of the one-dimensional, inviscid Burgers' equation with $p = 4, q = 1$. Initial condition $\bar{U}(x)$ (——) and tracking solution at times $t = 0.05, 0.35, 0.65, 0.95$ (——).

$$\bar{U} : (x_1, x_2) \mapsto \begin{cases} (0.5 - 2(x_2^2 - 0.25)) \left(\frac{4}{3}(x_1 + 0.75)\right) & x \in \Omega_\square \\ 0 & \text{elsewhere,} \end{cases} \tag{32}$$

where $\Omega_\square := [-0.75, 0] \times [-0.5, 0.5]$. The initial condition is constructed such that the initially straight shock curves over time, which is tracked by the high-order mesh. The shock tracking solution is computed using a DG discretization on a mesh with 128 simplex elements of degree $p = q = 2$ and a DIRK3 temporal discretization with 40 time steps (Figure 8). The mesh smoothing procedure described in Section 4.1.2 is important to maintain high-quality elements as the shock moves across the domain.

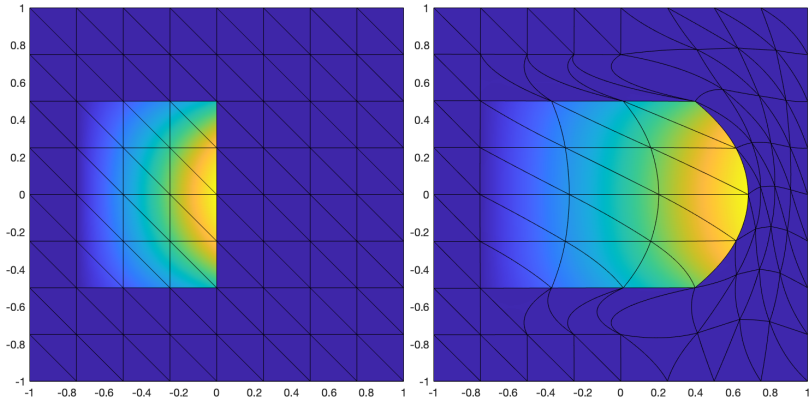
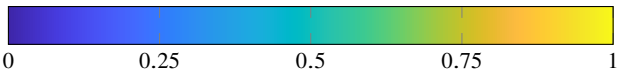



Fig. 8: Method of lines solution of two-dimensional, inviscid Burgers' equation with $p = q = 2$. Initial condition $\bar{U}(x)$ (*left*) and solution at $T = 2$ (*right*)




## 5.3 Euler equations

The Euler equations govern the flow of an inviscid, compressible fluid through a domain $\Omega \subset \mathbb{R}^d$

$$\begin{aligned} \frac{\partial}{\partial t}\rho(x,t) + \frac{\partial}{\partial x_j}\left(\rho(x,t)v_j(x,t)\right) &= 0 \\ \frac{\partial}{\partial t}\left(\rho(x,t)v_i(x,t)\right) + \frac{\partial}{\partial x_j}\left(\rho(x,t)v_i(x,t)v_j(x,t) + P(x,t)\delta_{ij}\right) &= 0 \\ \frac{\partial}{\partial t}\left(\rho(x,t)E(x,t)\right) + \frac{\partial}{\partial x_j}\left(\left[\rho(x,t)E(x,t) + P(x,t)\right]v_j(x,t)\right) &= 0 \end{aligned} \tag{33}$$

for all $x \in \Omega, t \in [0,T], i = 1,\dots,d$ and summation is implied over the repeated index $j = 1,\dots,d$, where $\rho : \Omega\times(0,T) \to \mathbb{R}_+$ is the density of the fluid, $v_i : \Omega\times(0,T) \to \mathbb{R}$ for $i = 1,\dots,d$ is the velocity of the fluid in the $x_i$ direction, and $E : \Omega\times(0,T) \to \mathbb{R}_{>0}$ is the total energy of the fluid, implicitly defined as the solution of (33). For a calorically ideal fluid, the pressure of the fluid, $P : \Omega \times (0,T) \to \mathbb{R}_{>0}$, is related to the energy via the ideal gas law

$$P = (\gamma - 1)\left(\rho E - \frac{\rho v_i v_i}{2}\right), \tag{34}$$

where $\gamma \in \mathbb{R}_{>0}$ is the ratio of specific heats. By combining the density, momentum, and energy into a vector of conservative variables $U : \Omega \times [0,T] \to \mathbb{R}^{d+2}$, defined as

$$U : (x,t) \mapsto \begin{bmatrix} \rho(x,t) \\ \rho(x,t)v(x,t) \\ \rho(x,t)E(x,t) \end{bmatrix} \tag{35}$$

the Euler equations are a conservation law of the form (1). Now, we investigate the shock tracking framework on three benchmark examples governed by these equations: Sod's shock tube (a Riemann problem), the Shu-Osher problem, and supersonic flow over a NACA0012 airfoil.

### 5.3.1 Sod's shock tube

Sod's shock tube is a Riemann problem for the Euler equations that models an idealized shock tube where the membrane separating a high pressure region from a low pressure one is instantaneously removed. This is a commonly used validation problem since it has an analytical solution and features a shock wave, a rarefaction wave, and contact discontinuity. The flow domain is $\Omega = [0,1]$, the final time is $T = 0.2$, the initial condition is given in terms of the density, velocity, and pressure as

$$\rho(x,0) = \begin{cases} 1 & x < 0.5 \\ 0.125 & x \geq 0.5 \end{cases}, \quad v(x,0) = 0, \quad P(x,0) = \begin{cases} 1 & x < 0.5 \\ 0.1 & x \geq 0.5, \end{cases} \tag{36}$$

and the density, velocity, and pressure are prescribed at $x = 0$ and the velocity is prescribed at $x = 1$ (values can be read from the initial condition). The solution of this problem contains three waves (shock, contact, rarefaction) that emanate from $x = 0.5$ and move at different speeds, which is a generalized triple point in space-time. The method of lines approach cannot handle this case because a single node lies at $(x,t) = (0.5, 0)$, which cannot track all three waves at time $t > 0$. Thus, we use the space-time implicit shock tracking approach to solve this problem. The method is initialized with an unstructured mesh of the space-time domain $\bar{\Omega} = [0,1] \times [0,0.2]$ consisting of 173 simplex elements ($p = 3$, $q = 1$) with additional refinement near $(0.5, 0)$ to resolve the geometric complexity of the triple point for a total of 5190

spatiotemporal degrees of freedom. The DG solution is initialized with the $p = 0$ solution on the reference mesh. The final space-time tracking solution is shown in Figure 9 where all features (head and tail of rarefaction, shock, and contact) are tracked. A total of 47 element collapses are required, mostly near the triple point to obtain elements that do not cross between the five distinct regions (left state, rarefaction, between rarefaction and contact, between contact and shock, and right state); the final mesh contains 126 elements (3780 degrees of freedom). Despite the reduction in the number of degrees of freedom, the solution at the final time $T$ agrees well with the exact solution (Figure 10) because the cubic basis functions do not cross discontinuities or kinks.

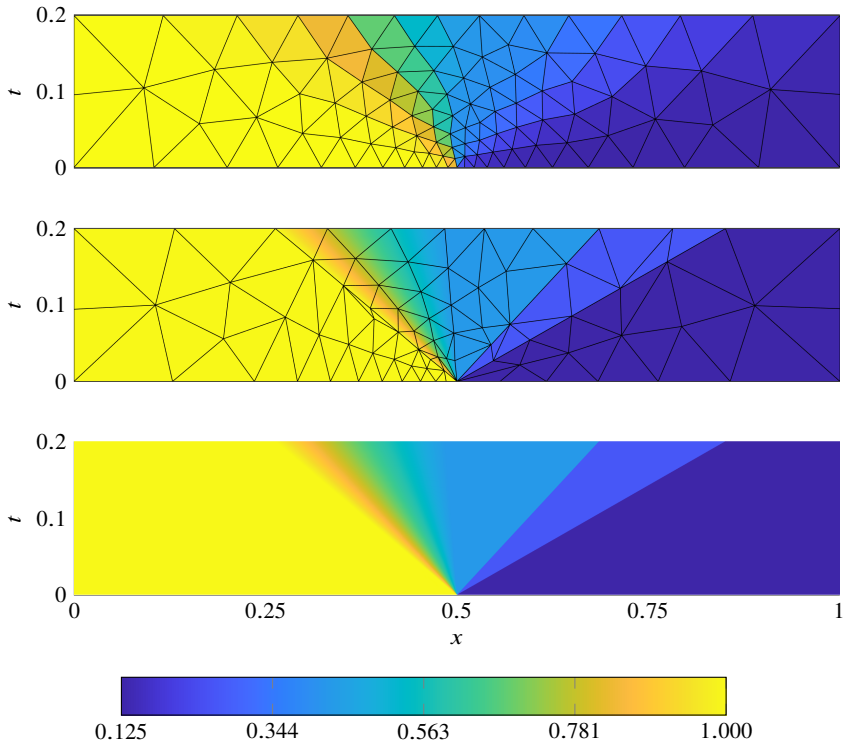



Fig. 9: Space-time solution of Sod's shock tube (density) using implicit shock tracking using a $p = 3$ DG discretization (*center* with element boundaries, *bottom* without element boundaries), initialized from an unstructured mesh without knowledge of the discontinuity surfaces and a $p = 0$ DG solution (*top*).

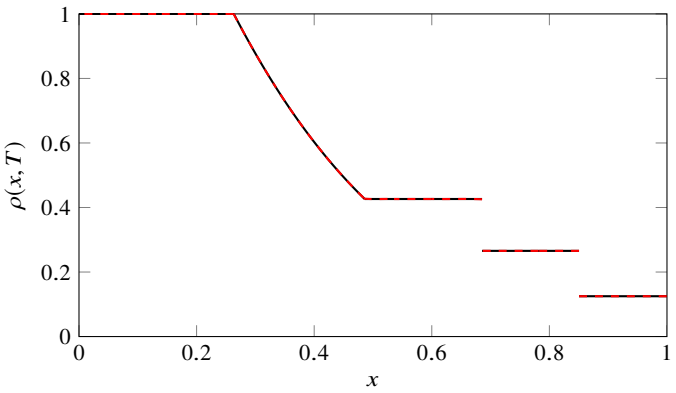



Fig. 10: Slice of space-time implicit shock tracking solution (——) at time $T = 0.2$ relative to the exact solution (- - -).

### 5.3.2 Shu-Osher problem

The Shu-Osher problem [28] is a one-dimensional idealization of shock-turbulence interaction where a Mach 3 shock moves into a field with a small sinusoidal density disturbance. The flow domain is $\Omega = [-4.5, 4.5]$, the final time is $T = 1.1$, the initial condition is given in terms of the density, velocity, and pressure as

$$\begin{aligned}
\rho(x,0) &= \begin{cases} 3.857143 & x < -4 \\ 1 + 0.2\sin(5x) & x \geq -4 \end{cases} \\
v_1(x,0) &= \begin{cases} 2.629369 & x < -4 \\ 0 & x \geq -4 \end{cases} \\
P(x,0) &= \begin{cases} 10.3333 & x < -4 \\ 1 & x \geq -4, \end{cases}
\end{aligned} \tag{37}$$

and the density, velocity, and pressure are prescribed at $x = -4.5$ and the velocity is prescribed at $x = 4.5$ (values can be read from the initial condition). The final time is chosen such that waves trailing behind the primary shock do not steepen into shock waves; shock formation will be the subject of future work. The shock tracking solution is computed using a DG discretization on a mesh with 288 elements of degree $p = 4$, $q = 1$, and a DIRK3 temporal discretization with 110 time steps (Figure 11) along with a reference solution computed using a fifth-order WENO method with 200 elements and temporal integration via RK4 with 110 timesteps [28]. The shock tracking solution actually overshoots the reference solution at the formation of the trailing waves, which suggests the reference solution is being overly dissipated by the WENO scheme (*left inset*). The shock is perfectly represented by

the aligned mesh in the shock tracking solution compared to the reference (*right inset*).

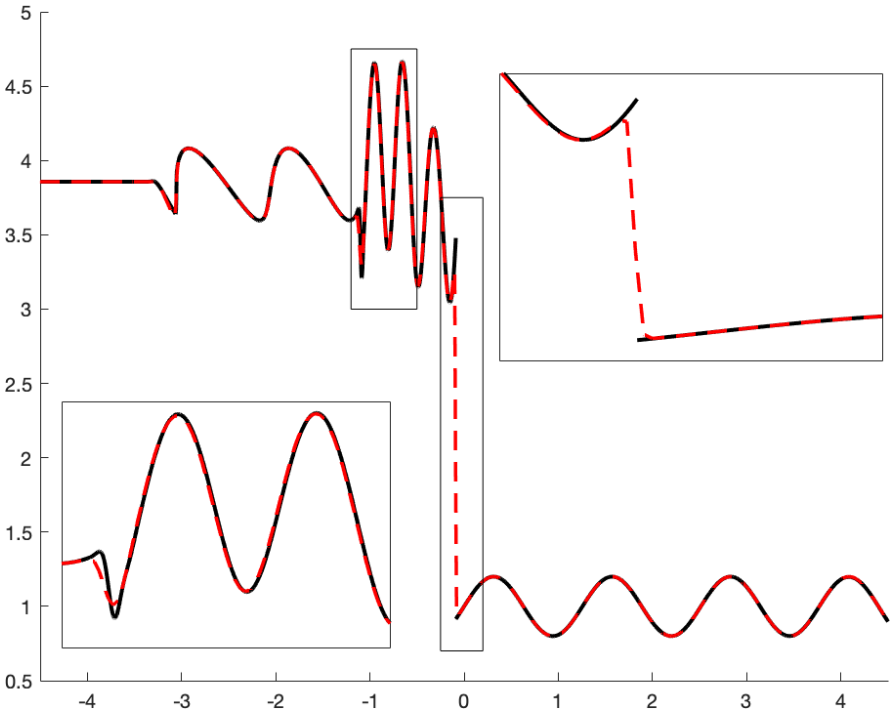



Fig. 11: Density at $T = 1.1$ of Shu-Osher problem for the the reference (- - -) and shock tracking (——) solutions.

Unlike Sod's shock tube, the Shu-Osher problem is well-suited for the method of lines approach because there is no space-time triple point, e.g., from multiple waves emanating from a point or intersecting discontinuities. In this case, the method of lines approach is preferred because computations are only required on a $d$-dimensional mesh as opposed to a $(d+1)$-dimensional space-time mesh (all time coupled), which is more practical for large problems.

### 5.3.3 Supersonic flow over airfoil

Finally, we apply the implicit tracking method to solve for supersonic flow over a NACA0012 airfoil (Figure 12), which is governed by the 2D steady, compressible Euler equations (see [36] for a complete description of the problem). This is a difficult problem because there are two distinct shocks that must be resolved: a bow shock ahead of the leading edge and an oblique shock off the tail. To demonstrate

the implicit shock tracking method, we use a coarse mesh consisting of 160 simplex elements with a second-order ($p = q = 1$) and third-order ($p = q = 2$) solution and mesh discretization; the initial mesh is generated without knowledge of the shock location (Figure 12, left). In both cases, the tracking procedure tracks the shocks given the resolution in the finite element space, despite the initial mesh and solution being far from aligned with the shock. The second-order approximation is somewhat underresolved as seen by the faceted shock approximation and solution near the airfoil; however, the third-order solution is well-resolved: the high-order elements curve to the shock and the flow solution is well-resolved throughout the domain.

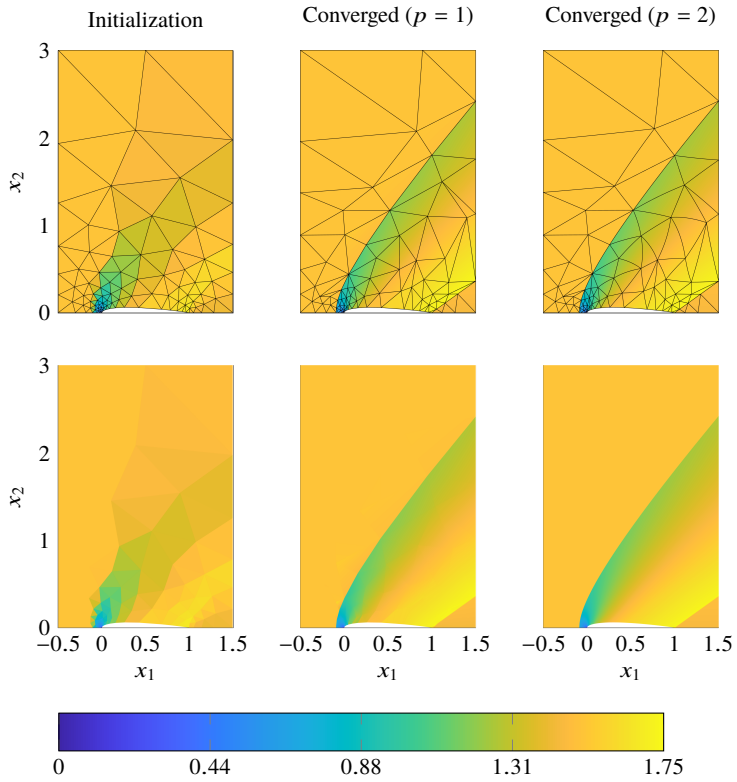



Fig. 12: Solution (Mach) of Euler equations over the NACA0012 airfoil ($M_\infty = 1.5$) using the implicit tracking method with a $p = q = 1$ (*center*) and $p = q = 2$ (*right*) basis for the solution and mesh with (*top*) and without (*bottom*) element boundaries. The implicit tracking procedure is initialized from a mesh generated without knowledge of the shock surface and a $p = 0$ DG solution (*left*).

## 6 Conclusion

This work provides an overview of the implicit shock tracking method developed in [34, 36, 27] for steady and unsteady conservation laws with discontinuous solutions. For unsteady problems, both space-time and method of lines discretization approaches are considered. The key ingredient of the method is an optimization formulation that imposes the standard DG discretization as a constraint and minimizes the magnitude of the DG residual corresponding to an enriched test space (and a mesh quality term). The optimization variables are taken to be the DG solution and nodal coordinates, which are computed simultaneously using a sequential quadratic programming method. In the method of lines setting, the tracking procedure is applied at each stage of the high-order DIRK temporal discretization. We demonstrated the implicit shock tracking procedure using a number of standard steady and unsteady flow problems; in all cases, the method is capable of tracking discontinuities and providing high-quality flow approximations using coarse, high-order meshes. For unsteady problems, the method of lines approach is more practical, particularly as the size and difficulty of the problem increases; however, it is limited in that it cannot handle colliding shocks (triple points in space-time) without complex mesh operations and solution reinitialization. In these cases, the space-time approach is preferred due to its generality of tracking discontinuities in space-time, which naturally handles triple points.

## Acknowledgments

This material is based upon work supported by the Air Force Office of Scientific Research (AFOSR) under award number FA9550-20-1-0236. The content of this publication does not necessarily reflect the position or policy of any of these supporters, and no official endorsement should be inferred.

## References


1. M. J. Baines, S. J. Leary, and M. E. Hubbard. Multidimensional least squares fluctuation distribution schemes with adaptive mesh movement for steady hyperbolic equations. *SIAM Journal on Scientific Computing*, 23(5):1485–1502, 2002.
2. C. E. Baumann and J. T. Oden. A discontinuous $hp$ finite element method for the Euler and Navier-Stokes equations. *International Journal for Numerical Methods in Fluids*, 31(1):79–95, 1999. Tenth International Conference on Finite Elements in Fluids (Tucson, AZ, 1998).
3. J. B. Bell, G. R. Shubin, and J. M. Solomon. Fully implicit shock tracking. *Journal of Computational Physics*, 48(2):223–245, 1982.
4. A. Burbeau, P. Sagaut, and C.-H. Bruneau. A problem-independent limiter for high-order Runge-Kutta discontinuous Galerkin methods. *Journal of Computational Physics*, 169(1):111–150, 2001.
5. B. Cockburn and C.-W. Shu. Runge-Kutta discontinuous Galerkin methods for convection-dominated problems. *Journal of Scientific Computing*, 16(3):173–261, 2001.
6. A. Corrigan, A. Kercher, and D. Kessler. A moving discontinuous Galerkin finite element method for flows with interfaces. *International Journal for Numerical Methods in Fluids*, 89(9):362–406, 2019.
7. A. Corrigan, A. Kercher, and D. Kessler. The moving discontinuous Galerkin method with interface condition enforcement for unsteady three-dimensional flows. In *AIAA Scitech 2019 Forum*, 2019.
8. A. Corrigan, A. Kercher, D. Kessler, and D. Wood-Thomas. Convergence of the moving discontinuous galerkin method with interface condition enforcement in the presence of an attached curved shock. In *AIAA Aviation 2019 Forum*, page 3207, 2019.
9. A. Dervieux, D. Leservoisier, P.-L. George, and Y. Coudière. About theoretical and practical impact of mesh adaptation on approximation of functions and PDE solutions. *International Journal for Numerical Methods in Fluids*, 43(5):507–516, 2003. ECCOMAS Computational Fluid Dynamics Conference, Part I (Swansea, 2001).
10. K. J. Fidkowski. Output error estimation strategies for discontinuous Galerkin discretizations of unsteady convection-dominated flows. *International Journal for Numerical Methods in Engineering*, 88(12):1297–1322, 2011.
11. B. Froehle and P.-O. Persson. Nonlinear elasticity for mesh deformation with high-order discontinuous Galerkin methods for the Navier-Stokes equations on deforming domains. In *Spectral and High Order Methods for Partial Differential Equations ICOSAHOM 2014*, pages 73–85. Springer, 2015.
12. A. Gargallo-Peiró, X Roca, J Peraire, and J. Sarrate. Optimization of a regularized distortion measure to generate curved high-order unstructured tetrahedral meshes. *International Journal for Numerical Methods in Engineering*, 103(5):342–363, 2015.
13. A. Gargallo-Peiró, X. Roca, J. Peraire, and J. Sarrate. A distortion measure to validate and generate curved high-order meshes on CAD surfaces with independence of parameterization. *International Journal for Numerical Methods in Engineering*, 106(13):1100–1130, 2016.
14. J. Glimm, X.-L. Li, Y.-J. Liu, Z.-L. Xu, and N. Zhao. Conservative front tracking with improved accuracy. *SIAM Journal on Numerical Analysis*, 41(5):1926–1947, 2003.
15. A. Harten, B. Engquist, S. J. Osher, and S. R. Chakravarthy. Uniformly high-order accurate essentially nonoscillatory schemes. III. *Journal of Computational Physics*, 71(2):231–303, 1987.
16. A. Harten and J. M. Hyman. Self adjusting grid methods for one-dimensional hyperbolic conservation laws. *Journal of Computational Physics*, 50(2):235–269, 1983.
17. J. Hesthaven and T. Warburton. *Nodal Discontinuous Galerkin Methods: Algorithms, Analysis, and Applications*. Springer Science & Business Media, 2007.
18. G.-S. Jiang and C.-W. Shu. Efficient implementation of weighted ENO schemes. *Journal of Computational Physics*, 126(1):202–228, 1996.
19. A. D. Kercher and A. Corrigan. A least-squares formulation of the Moving Discontinuous Galerkin Finite Element Method with Interface Condition Enforcement. *Computers & Mathematics with Applications*, November 2020.

20. A. D. Kercher, A. Corrigan, and D. A. Kessler. The Moving Discontinuous Galerkin Finite Element Method with Interface Condition Enforcement for Compressible Viscous Flows. *International Journal for Numerical Methods in Fluids*, November 2020.
21. X.-D. Liu, S. J. Osher, and T. Chan. Weighted essentially non-oscillatory schemes. *Journal of Computational Physics*, 115(1):200–212, 1994.
22. A. Majda. *Compressible fluid flow and systems of conservation laws in several space variables*, volume 53. Springer Science & Business Media, 2012.
23. G. Moretti. Thirty-six years of shock fitting. *Computers & Fluids*, 31(4-7):719–723, 2002.
24. P.-O. Persson, J. Bonet, and J. Peraire. Discontinuous Galerkin solution of the Navier–Stokes equations on deformable domains. *Computer Methods in Applied Mechanics and Engineering*, 198(17):1585–1595, 2009.
25. P.-O. Persson and J. Peraire. Sub-cell shock capturing for discontinuous Galerkin methods. In *44th AIAA Aerospace Sciences Meeting and Exhibit, Reno, Nevada*, 2006. AIAA-2006-0112.
26. P. S. Rawat and X. Zhong. On high-order shock-fitting and front-tracking schemes for numerical simulation of shock–disturbance interactions. *Journal of Computational Physics*, 229(19):6744–6780, 2010.
27. A. Shi, M. J. Zahr, and P.-O. Persson. Implicit shock tracking and the method of lines for shock-dominated, unsteady flows, In preparation.
28. C.-W. Shu and S. Osher. Efficient implementation of essentially non-oscillatory shock-capturing schemes, II. In *Upwind and High-Resolution Schemes*, pages 328–374. Springer, 1989.
29. G. R. Shubin, A. B. Stephens, and H. M. Glaz. Steady shock tracking and Newton's method applied to one-dimensional duct flow. *Journal of Computational Physics*, 39(2):364–374, 1981.
30. G. R. Shubin, A. B. Stephens, H. M. Glaz, A. B. Wardlaw, and L. B. Hackerman. Steady shock tracking, Newton's method, and the supersonic blunt body problem. *SIAM Journal on Scientific and Statistical Computing*, 3(2):127–144, 1982.
31. J.-Y. Trepanier, M. Paraschivoiu, M. Reggio, and R. Camarero. A conservative shock fitting method on unstructured grids. *Journal of Computational Physics*, 126(2):421 – 433, 1996.
32. J. Van Rosendale. Floating shock fitting via lagrangian adaptive meshes. Technical Report ICASE Report No. 94-89, Institute for Computer Applications in Science and Engineering, 1994.
33. Z. J. Wang, K. Fidkowski, R. Abgrall, F. Bassi, D. Caraeni, A. Cary, H. Deconinck, R. Hartmann, K. Hillewaert, H.T. Huynh, et al. High-order CFD methods: current status and perspective. *International Journal for Numerical Methods in Fluids*, 72(8):811–845, 2013.
34. M. J. Zahr and P.-O. Persson. An optimization-based approach for high-order accurate discretization of conservation laws with discontinuous solutions. *Journal of Computational Physics*, 365:105–134, 2018.
35. M. J. Zahr and J. M. Powers. Accurate, high-order resolution of multidimensional compressible reactive flow using implicit shock tracking. *AIAA Journal*, in press.
36. M. J. Zahr, A. Shi, and P.-O. Persson. Implicit shock tracking using an optimization-based high-order discontinuous Galerkin method. *Journal of Computational Physics*, 410:109385, June 2020.