Journal of Computational Physics 514 (2024) 113255

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

A block-spectral adaptive H-/p-refinement strategy for shock-dominated problems

Vaibhav Rajora [a,*], Tariq Aslam [b], Wesley W. Chapman [b], Chad Meyer [b], Steve F. Son [c], Carlo Scalo [a,c]

[a] Purdue University School of Aeronautics and Astronautics, United States of America
[b] Los Alamos National Laboratory, United States of America
[c] Purdue University School of Mechanical Engineering, United States of America

ARTICLE INFO

Keywords:
AMR
High-order numerics
Shock-capturing
Flux reconstruction

ABSTRACT

An adaptive H-/p-refinement strategy using a novel sensor is devised and tested in a block-spectral compressible Euler code equipped with adaptive-mesh refinement (AMR) and high-order flux-reconstruction numerics. At each Gauss quadrature point (or solution point) within each spectral block (or mesh element) the discrete velocity jump $\Delta U = \partial U/\partial y_1 \Delta y_1 + \partial V/\partial y_2 \Delta y_2 + \partial W/\partial y_3 \Delta y_3$ is calculated and normalized by the local speed of sound, a. The grid spacing, Δx_i, is calculated in each direction as the distance between auxiliary Gauss-Lobatto points, staggered relative to the solution points. The polynomial order is increased from $p = 0$ to $p = p_{\max}$ in regions of weak compression, $(\Delta U/a)_{\text{crit}} < \Delta U/a < 0$ and kept at $p = p_{\max}$ in regions of flow expansion $\Delta U/a \geq 0$, while staying at the $H = 0$ base mesh level. Regions experiencing strong compressions, i.e. $\Delta U/a < (\Delta U/a)_{\text{crit}}$, are H-refined up to $H = H_{\max}$ where $H_{\max}$ is applied at the location of maximum compression, $\Delta U/a = \min(\Delta U/a)$ in the domain, while keeping $p = 0$ to guarantee robustness and monotonicity of the solution in the H refined region. The critical value of $(\Delta U/a)_{\text{crit}} = -0.06$ is found to effectively separate smooth and non-smooth solution regions, supported by a 1D detonation initiation test case in ideal gas and a shock-to-detonation transition in high explosives. Using this value, the Sod shock tube, Shu-Osher problem, double Mach reflection and a 2D detonation in a high-explosive are simulated with the proposed adaptive H-/p-refinement. In the Sod shock tube case, p-refinement resolves the (weak) contact discontinuity while H-refinement enhances the grid resolution in the shock exploiting the monotonicity of the $p = 0$ reconstruction. For the Shu-Osher problem, p-refinement captures the small-scale oscillations trailing the shock that would be otherwise attenuated, while H-refinement triggered by the ΔU-sensor appropriately tracks the shock. In the double Mach reflection problem, H-refinement confines the numerical diffusion around the reflected shock while p-refinement recaptures many physical features trailing the shock. Finally, in the 2D high-explosive detonation case, H-refinement follows the leading shock and resolves the curvature of the detonation wave, while p-refinement adds resolution to the trailing reaction zone. Finally, the proposed methodology is tested in a detonation-wave propagation test case in high-explosives with numerical predictions comparing favorably against experiments.

* Corresponding author.
E-mail address: vrajora@purdue.edu (V. Rajora).

https://doi.org/10.1016/j.jcp.2024.113255
Received 2 October 2023; Received in revised form 24 May 2024; Accepted 30 June 2024

Available online 6 July 2024

1. Introduction

Shock-dominated problems are found in many engineering applications ranging from high-speed aerodynamics (transonic or supersonic internal or external flows) to detonation wave propagation in solid explosives, or reactive liquid/gaseous mixtures. Modeling any of these problems should ideally achieve high-order accuracy of spatial convergence, while also capturing flow discontinuities and guaranteeing monotonicity and positivity of the solution. These requirements are notoriously in conflict [1,2]. Adaptive $H-/p-$refinement strategies combining low-order flux exchange between mesh elements, i.e. across faces of the H mesh, and order p polynomial reconstruction within each element, may be one viable option to achieve the aforementioned desired numerical behavior for shock-dominated flows. The present manuscript explores such strategies relying on block-spectral flux reconstruction numerics (FR) and an unstructured code architecture.

1.1. High-order spectral differences / flux-reconstruction methods

FR numerics are an evolution of the spectral difference (SD) method, first introduced by Kopriva and Kolias [3,4], which relies on a staggered grid approach, combining Chebyshev-Gauss quadrature points for the solution and Gauss-Lobatto quadrature points for the flux point (see Section 4.2). At the cell faces, Riemann solvers [2,5–7] are used to ensure continuous flux values and the spatial derivative of the fluxes computed at the solution points are used to advance in time. Hunyh [8–10] later created the FR method by improving the stability and algorithmic simplicity of Kopriva's SD method, removing the need for flux points within the spectral element, relying on lifting functions to guarantee continuity of the flux across the faces and regulate the stability of the time advancement. Wang and Gao extended the FR methodology to triangular, prismatic, or tetrahedral meshes by using the lifting collocation penalty (LCP) approach. A correction field is used to correct the discontinuous flux divergence and is determined from a lifting operator. The FR formulation has been reported to be 27%-40% more computationally efficient than the staggered SD method [11,12].

While spectral methods offer high resolving power, like any high-order method, they are not robust in the presence of flow discontinuities. The inherently low degree of numerical dissipation leads to a pronounced Gibbs phenomena [13–15], which can quickly cause the solution to lose monotonicity and positivity. On the other hand, block-spectral methods are capable of sustaining a shock located exactly at the interface between cells, leading to the adoption of shock-fitting methods [16–20]. These methods retain a high-order rate of convergence but require shock-tracking methods and moving mesh capabilities. On the other hand, shock-capturing methods [21–25] do not require a moving mesh, but typically entail a number of tunable parameters that are often problem dependent.

1.2. Adaptive mesh refinement

Adaptive mesh refinement (AMR) dynamically increases the mesh resolution surrounding shocks to increase the robustness of the shock capturing and to ideally achieve a numerically resolved shock thickness that is smaller than the other relevant length scales of the problem. AMR has been a popular approach in computational fluid dynamics (CFD) for its ability to capture multi-scale features with a considerable saving of computational resources. It has been extensively employed in many applications: magnetohydrodynamics [26,27], multiphase flows [28,29], atomization of liquid impinging jets [30], bubble dynamics [31], and vortex dynamics [32–38].

The works by Berger and Oliger [39], and Berger and Colella [40] are among the earliest to be documented within the context of shock hydrodynamics. They employed a Richardson-type error estimation procedure to generate rectangular patches (subgrids) to track the discontinuity on a Cartesian mesh with a finite difference method. These subgrids were allowed to have an arbitrary orientation with respect to background mesh to better align with flow features. Multiple subgrids were allowed to be present in the computation, and evolve independently. Mavriplis [41,42] extended the AMR technique to spectral elements using an *a posteriori* error estimator based on the spectrum of the solution. The elements could be moved, reduced in size, and split into smaller elements. To avoid potential mesh degeneration, H-refinement was limited to elements with the greatest need for refinement with a voting strategy. Also de-refinement was implemented by moving mesh elements since mesh coarsening resulted in a loss of accuracy for the error estimator.

Lomtev et al. [43] adopted H-/p-refinement strategies using triangular 2D meshes for spectral methods based on a mixture of discontinuous Galerkin (DG) and continuous Galerkin formulations. A hierarchical spectral basis was used to ensure high-order accuracy, while high-order limiters reduced the accuracy to first-order in shocked elements to ensure monotonicity. Adaptive H-refinement was then used to add resolution at these locations. This approach is similar to the one proposed in the current manuscript. Bey [44] and Devine [45,46] used adaptive $H-/p-$refinement in a finite element formulation using a combination of *a priori* and *a posteriori* error estimates. The mesh was first H-refined (based on the *a priori* error) and then p-refined based on the *a posteriori* error estimate.

Our proposed H-/p-method reduces the order of accuracy around shocks, similar to Huerta et al. [47] and Persson et al. [48], who relied on a hybrid DG scheme. Sonntag and Munz [49] and Monzier et al. [50] extended their approach to an adaptive H-/p-method that subdivided a DG element into finite volume (FV) subcells to capture a shock contained within said element. A single *a priori* error estimator, based on the modal decay of the local polynomial solution representation, detects regions of insufficient grid resolution, dictating the degree of local p-refinement. Similar to Yang et al. [51] our methodology combines adaptive H-refinement with FR numerics. However, Yang employs uniform polynomial order (no adaptive p-refinement) and maintains monotonicity using localized artificial diffusion [24].

The current study proposed a combination of FR with adaptive H-/p-refinement. In regions of smooth flow, high-order numerics are used, while in regions of discontinuous or steep flow features, cells are split into FV subcells via H-refinement. The novelty lies in a dilatation-based velocity jump sensor normalized with respect to the local speed of sound. A dimensionless critical threshold of the thus-defined velocity jump is established relying on test cases ranging from ideal gases to high-explosives, also corroborated by previous scaling work on heat-release-induced shocks [52] in a variety of fluids under different thermodynamic conditions.

1.3. Paper outline

The rest of the article is organized as follows: Section 2 discusses the governing equations, Section 3 introduces the equations of state (EoS) used to model high-explosives, Section 4 addresses the flux reconstruction methodology as implemented in the current code, and Section 5 presents the results from the various test case where our adaptive H-/p-strategy is employed.

2. Governing equations

High-order flux reconstruction numerics are used to solve the three-dimensional compressible Euler equations that read

$$\frac{\partial Q}{\partial t}+\frac{\partial F}{\partial y_1}+\frac{\partial G}{\partial y_2}+\frac{\partial H}{\partial y_3}=0. \tag{1}$$

Here, $Q=[\rho,\rho U,\rho V,\rho W,E]$ is the vector of conserved variables and $\{y_1,y_2,y_3\}$ are Cartesian coordinates spanning the physical domain, and (U,V,W) are the velocity components in the same coordinate system. The flux vectors, F, G, and H are defined as:

$$F=\begin{bmatrix}\rho U\\ \rho U^2+P\\ \rho UV\\ \rho UW\\ (E+P)U\end{bmatrix},\qquad G=\begin{bmatrix}\rho V\\ \rho UV\\ \rho V^2+P\\ \rho VW\\ (E+P)V\end{bmatrix},\qquad H=\begin{bmatrix}\rho W\\ \rho UW\\ \rho VW\\ \rho W^2+P\\ (E+P)W\end{bmatrix}. \tag{2}$$

Two different equations of state (EoS) closures are used in this study and will be discussed later in section 3. The generic point in the physical space $\{y_1,y_2,y_3\}$ is mapped to a corresponding point in the computational space $\{x_1,x_2,x_3\}$ via a trilinear transformation between the 8 vertices of the hexahedral element in the physical and computational ([-1,1] x [-1,1] x [-1,1]) space.

The Jacobian matrix is given by $J_{ij}=\partial y_i/\partial x_j$ and the transformation from the computational to the physical space is retrieved by inverting this matrix. We can write the Euler equations in the computational space as:

$$\frac{\partial \overline{Q}}{\partial t}+\frac{\partial \overline{F}}{\partial x_1}+\frac{\partial \overline{G}}{\partial x_2}+\frac{\partial \overline{H}}{\partial x_3}=0. \tag{3}$$

Here,

$$\begin{aligned}\overline{Q}&=QJ,\\ \overline{F}&=J\left(\frac{\partial x_1}{\partial y_1}F+\frac{\partial x_1}{\partial y_2}G+\frac{\partial x_1}{\partial y_3}H\right),\\ \overline{G}&=J\left(\frac{\partial x_2}{\partial y_1}F+\frac{\partial x_2}{\partial y_2}G+\frac{\partial x_2}{\partial y_3}H\right),\\ \overline{H}&=J\left(\frac{\partial x_3}{\partial y_1}F+\frac{\partial x_3}{\partial y_2}G+\frac{\partial x_3}{\partial y_3}H\right),\end{aligned} \tag{4}$$

where J is the determinant of the Jacobian matrix.

3. Equations of state

For the purposes of this paper, the ideal gas EoS as well as a Mie-Grüneisen based EoS for high-explosives are used.

3.1. Ideal-gas

The ideal gas closure for the Euler equations, which do not require the explicit definition of temperature in their non-reactive form, is simply implemented via the extraction of pressure from the total energy and kinetic energy via

$$E=\frac{P}{\gamma-1}+\frac{1}{2}\rho(u^2+v^2+w^2), \tag{5}$$

where γ is the ratio of specific heats, which is a fluid property and E is the energy per unit volume.

3.2. High-explosive modeling

This section explains the modeling strategy for wave-propagation modeling in high-explosives.

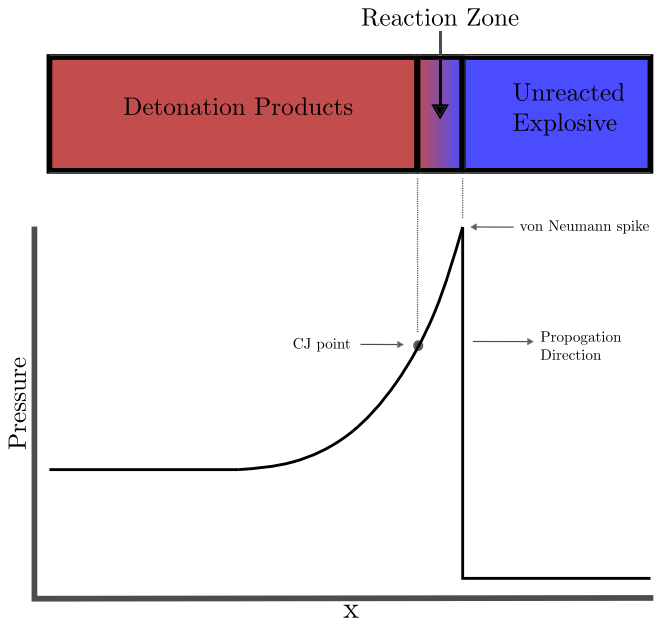

Fig. 1. The ZND structure and the resulting pressure plot for a 1D detonation wave in high-explosives.

3.2.1. The ZND theory

Unlike low explosives which deflagrate, combustion in high-explosives happens as a detonation, in which the front of the chemical reaction moves supersonically. Due to the large pressures reached inside these materials during the detonation processes, entailing fluid-like behavior, the Euler equations combined with an EoS can be used to model these detonations [53]. The Zel'dovitch, von Neumann, Döring (ZND) [54–56] detonation model is a 1D model for detonations in explosives and it consists of the following spatial regions defined with respect to a shock-attached frame of reference (see also Fig. 1):

1. First, an infinitesimally thin shock compresses the unreacted explosive achieving the so-called von Neumann spike.
2. Downstream of the spike, chemical reactions begin to occur. This region is called the reaction zone.
3. The reaction zone ends at the Chapman-Jouguet (CJ) state, which has a sonic velocity in the frame of reference of the shock. The energy released by the reaction zone sustains the propagation of the detonation wave.

3.2.2. Reactant EoS

To model these high-explosives, the Davis [57–59] EoS are used and the equation for the reactant pressure can be expressed as

$$p_r(\rho, T) = p_r^s(\rho) + \rho\Gamma_r[e - e_r^s(\rho)]. \tag{6}$$

Here $p_r^s(\rho)$ is the reference isentrope where $y = 1 - \rho_0/\rho$ and $\hat{p} = \rho_0 A^2/(4B)$.

$$p_r^s(\rho) = \begin{cases} \hat{p}[\exp(4By) - 1], & \rho < \rho_0 \quad \text{(a)} \\ \hat{p}[\sum_{j=2}^{4} \frac{(4B)^{j-1} y^j}{j!} + C\frac{(4By)^4}{4!} + \frac{y^2}{(1-y)^4}] & \text{otherwise.} \quad \text{(b)} \end{cases} \tag{7}$$

The energy on the reference isentrope, $e_r^s(\rho)$ can be found by integrating the reference pressure along the isentrope and results in the following:

$$e_r^s(\rho) = \begin{cases} \frac{\hat{p}}{\rho_0}[\frac{\exp(4By)}{4B} - y - \frac{1}{4B}] + E_0, & \rho < \rho_0 \quad \text{(a)} \\ \frac{\hat{p}}{\rho_0}[\sum_{j=2}^{4} \frac{(4B)^{j-1} y^j}{j!} + C\frac{(4B)^4 y^5}{5!} + \frac{y^3}{3(1-y)^3}] + E_0 & \text{otherwise.} \quad \text{(b)} \end{cases} \tag{8}$$

Γ_r^0 is the Grüneisen parameter defined by:

$$\Gamma_r(\rho) = \begin{cases} \Gamma_r^0 & \rho < \rho_0 \quad \text{(a)} \\ \Gamma_r^0 + Zy & \text{otherwise.} \quad \text{(b)} \end{cases} \tag{9}$$

Table 1
Reactant values for the 1,3,5,7-Tetranitro-1,3,5,7-tetrazocane (HMX) based explosive PBX 9501.

A (mm μs)	B	C	ρ_0 (g/cc)	Γ_r^0	C_{vr}^0 (kJ/gK)	α_{st}	T_0 (K)	E_0 (kJ/g)	Z
2.3	3.3	0.2	1.836	0.838	.001067	0.3662	297	5.60208	0

The equation for the reactant temperature is given by:

$$T_r(\rho, e) = T_r^S + [\frac{1+\alpha_{st}}{C_{vr}^0 T_r^s(\rho)}(e - e_r^s(\rho)) + 1]^{\frac{1}{1+\alpha_{st}}} \tag{10}$$

in which the temperature on the reference isentrope, $T_r^s(\rho)$, is defined by:

$$T_r^s(\rho) = \begin{cases} T^0(\frac{\rho}{\rho_0})^{\Gamma_r^0} & \rho < \rho_0 \quad \text{(a)} \\ T^0 \exp(-Zy)(\frac{\rho}{\rho_0})^{\Gamma_r^0 + Z} & \text{otherwise.} \quad \text{(b)} \end{cases} \tag{11}$$

C_{vr}^0 is the reactant isochoric specific heat. In these equations, $A, B, C, \rho_0, C_{vr}^0, \alpha_{st}, T_0, E_0$, and Z are all material dependent properties. The values of these constants [60] are provided in Table 1.

3.2.3. Product EoS

The Davis product equation of state is given by [61]:

$$p_p(v, e) = p_p^s(v) + \frac{\Gamma_p(v)}{v}[e - e_p^s(v)], \tag{12}$$

where $p_p^s(v)$ is:

$$p_p^s(v) = p_c \frac{[\frac{1}{2}(v/v_c)^n + \frac{1}{2}(v/v_c)^{-n}]^{a/n}}{(v/v_c)^{k+a}} \frac{k - 1 + F(v)}{k - 1 + a} \tag{13}$$

in which $F(v)$ is given by:

$$\frac{2a(v/v_c)^{-n}}{(v/v_c)^n + (v/v_c)^{-n}} \tag{14}$$

and the energy of the reference isentrope, $e_p^s(v)$, by:

$$e_p^s(v) = E_c \frac{[\frac{1}{2}(v/v_c)^n + \frac{1}{2}(v/v_c)^{-n}]^{a/n}}{(v/v_c)^{k-1+a}}. \tag{15}$$

In the energy of the reference isentrope, E_c is defined as:

$$E_c = \frac{p_c v_c}{k - 1 + a}. \tag{16}$$

The Grüneisen parameter is defined by:

$$\Gamma_p(v) = k - 1 + (1 - b)F(v). \tag{17}$$

The product temperature EoS is:

$$T_p(v, e) = T_p^s(v) + \frac{e - e_P^s(v)}{C_{vp}}, \tag{18}$$

in which the temperature along the reference isentrope is given by:

$$T_c = T_c \frac{[\frac{1}{2}(v/v_c)^n + \frac{1}{2}(v/v_c)^{-n}]^{(a/n)(1-b)}}{(v/v_c)^{k-1+a(1-b)}} \tag{19}$$

and T_c by:

$$T_c = \frac{2^{-ab/n}}{k - 1 + a} \frac{p_c v_c}{C_{vp}}. \tag{20}$$

As for the reactant EoS, a, k, v_c, p_c, n, b, and C_{vp} are all material dependent properties [60] and are given in Table 2.

To model multi-phase flow, an appropriate closure model must be chosen to determine the relative phase volume and energy distribution. For the purposes of this article, the pressure-temperature equilibration outlined in [62] is used.

Table 2
Product values for the HMX based explosive PBX 9501.

a	k	v_c (cc/g)	p_c (GPa)	n	b	C_{vp} (kJ/g K)
0.684311	1.62149	0.745829	5.38897	1.58683	1.25	7.3e-04

Table 3
Reaction rate values for the HMX based explosive PBX 9501.

P_{CJ} (GPa)	k (1/μs)	N	ν
36.3	110	3.5	0.93

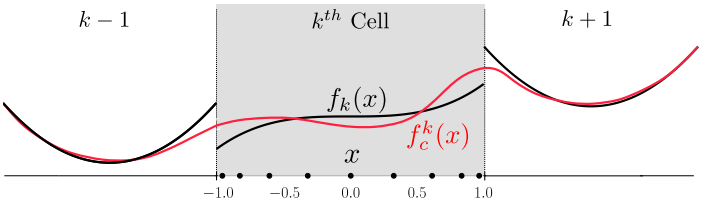

Fig. 2. Example of flux functions in 1D elements that are corrected from the shared information at the interface between elements. The figure shows three sample functions, $f_k(x)$ (black), and the continuous sample function, $f_k^c(x)$ (red). (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

3.2.4. Reaction rate model

A single-term fractional depletion pressure-dependent reaction rate [63] is used and given by:

$$r = k(1-\lambda)^{\nu}\left(\frac{P}{P_{CJ}}\right)^{N}. \tag{21}$$

Here, λ is the mass fraction of the detonation products.

The values for the constants in this reaction rate model are given by Table 3.

4. Numerical method

4.1. Flux reconstruction methodology

In the FR method, the fluxes are computed at the solution points (the roots of the Gauss quadrature rule) from the values of the conserved quantities in the computational space. Both conserved quantities and fluxes are then interpolated to the boundaries of the spectral element via Lagrange polynomials $L(x)$ associate with Gauss quadrature points $\{x_i\}$, which read

$$L_i(x) = \prod_{j=1, j\neq i}^{N} \frac{x - x_j}{x_i - x_j}. \tag{22}$$

This naturally yields discontinuities of the solution at the cell faces that are solved via the HLLC [5] (Harten-Lax-van Leer-Contact) or Rusanov [6] Riemann solvers. After the values at the interface are updated, the flux reconstruction method uses the left (g_L) and right (g_R) [8] correction functions to compute the values of the derivatives at the solution points. These functions are defined as symmetric N-th order polynomials ($g_R(x) = g_L(-x)$), and satisfy the boundary conditions $g_L(-1) = 1$ and $g_L(+1) = 0$. These properties allow for the reconstruction of a continuous function f_k^c:

$$f_k^c(x) = f_k(x) + (f^{un}_{k-\frac{1}{2}} - f_k(-1))g_L(x) + (f^{un}_{k+\frac{1}{2}} - f_k(+1))g_R(x), \tag{23}$$

where $f^{un}_{k-\frac{1}{2}}$ and $f^{un}_{k+\frac{1}{2}}$ are the update values at the left and right interfaces, where k refers to the cell index. Fig. 2 shows a 1D example of how this correction achieves a continuous flux across the interface.

The divergence of this function can be computed as:

$$\left.\frac{\partial f_k^c}{\partial x}\right|_{x=x_i} = D_{ij} f_k(x_j) + (f^{un}_{k-\frac{1}{2}} - f_k(-1))g_L'(x_i) + (f^{un}_{k+\frac{1}{2}} - f_k(+1))g_R'(x_i), \tag{24}$$

where D_{ij} is the matrix of weights for the first derivative collocated at the Gauss quadrature points given by

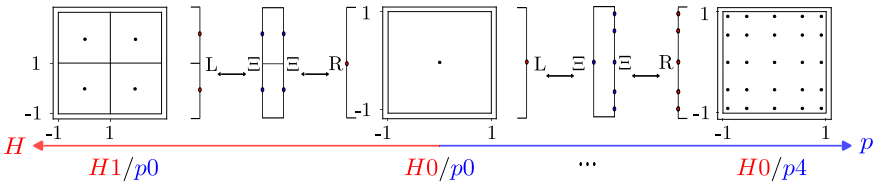

Fig. 3. An example of H-refinement and p-refinement alongside the resultant mortars (Ξ). Here, L and R reference the left and right elements of a face respectively. As H-refinement is increased, the parent cell splits into smaller children cell, each at polynomial order 0. As p-refinement is increased, more solution points are added to a cell.

$$L_i'(x) = L_i(x) \sum_{j=1, j\neq i}^{N} \frac{1}{x - x_j}, \tag{25}$$

evaluated at x_j, hence $D_{ij} = L_i'(x_j)$. The correction function used in the code is equal to:

$$g_{Ga,N} = \frac{N}{2N-1} R_{R,N} + \frac{N-1}{2N-1} R_{R,N-1}, \tag{26}$$

where $R_{R,N} = \frac{(-1)^N}{2}(L_N - L_{N-1})$ are the Radau polynomials and L_N the Lagrange polynomials. There are several correction functions in the literature [8], but this is one has the best compromise between accuracy and time step constraint.

Compared to the staggered SD method, the flux reconstruction (FR) technique requires less storage and fewer floating point operations per iteration. It is also more stable but dissipative. The latter problem can be remedied by running with a higher numerical order (p-refinement) or with a finer mesh (H-refinement).

4.2. Dimensionless velocity jump based adaptive H-/p-refinement strategy

There is a clear separation between the behavior of the linear and nonlinear flow regimes, which necessitates the need of two different strategies [64,52]. In the present section we explain the proposed H-/p-refinement strategy. H-refinement involves splitting a cell into smaller children cells, while p-refinement increases the polynomial order of a cell as seen in Fig. 3.

When adaptively refining elements, there are four conditions that are enforced. The first condition is a modified properly-nested assumption [39,65] which requires the following:

1. First, a level H(i+1) cell must be fully contained inside the level H(i) cell that created it (parent cell).
2. The boundaries of the level H(i+1) cell must coincide entirely with the boundaries of its parent cell unless the level H(i+1) boundary is an internal face. In this case, these boundaries must coincide entirely with those of their siblings (other H(i+1) cells sharing the same parent).
3. Third, the max allowable H-refinement difference between two adjacent cells will not exceed 1. While the code is able to handle p-refinement differences greater than 1 (e.g. a polynomial order 1 cell next to a polynomial order 3 cell), the p-refinement difference is also kept to be 1 at most.

The second condition requires that, if a steepened front exhibiting a compression rate above a certain threshold is detected in a high-order cell, the cell will first be restricted to $p = 0$. If further refinement is needed, the cell will be H-refined. Any cell that is H-refined will be kept at $p = 0$. This has two purposes: first, any region that is H-refined is detecting a numerical discontinuity and introducing any level of p-refinement may lead to the Gibbs phenomenon; the second purpose of this is to simplify the complexity of the surface interpolation operators on any mortars that may arise.

The third condition requires that any cell that is not H-refined but is adjacent to a cell that is H-refined be kept at order 0 to allow a smooth transition into the high-order regions. This also has the benefit of simplifying the surface interpolation operators acting on the mortars between different H levels, the solution being $p = 0$ on both sides.

The fourth condition concerns prolonging the solution from the parents to the children (H-refinement), vice-versa (H-derefinement via restriction operators), and p-refinement and p-derefinement. During H-refinement, since the parent and children cells are all order 0, the values for the conserved variables are copied from the parent cell to the children cell. During H-derefinement, a simple weighted average is taken to go from the children to the parent. For p-refinement and de-refinement, an analytical interpolant $\widetilde{Q}$ is created from the origin set of solution points $\{x_1, x_2, x_3\}_{(\mathrm{i,j,k})}$ using the following equation:

$$\widetilde{Q}(x_1, x_2, x_3) = \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{N} Q_{(\mathrm{i,j,k})} L_i(x_1) L_j(x_2) L_k(x_3), \tag{27}$$

where $Q_{(\mathrm{i,j,k})}$ is the value of a generic conserved variable at the (i, j, k) solution point in the computational element and $L_i(x_1)$, $L_j(x_2)$, $L_k(x_3)$ are the Lagrange polynomials in each direction given by equation (22).

To differentiate between regions of: 1) strong or 2) weak compressions, or 3) expansions in the flow, a sensor based on a discrete velocity jump normalized by the speed of sound, $\Delta U/a$, defined at each solution point is used. ΔU is given by:

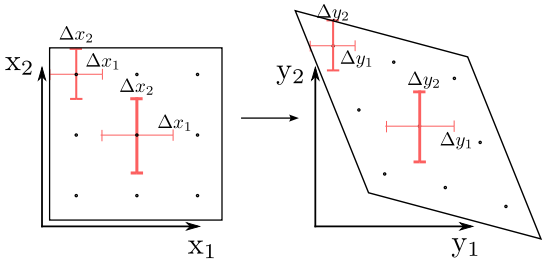

Fig. 4. Δx calculated in computational space (left) and the resultant Δy, used in (28), in the physical space (right).

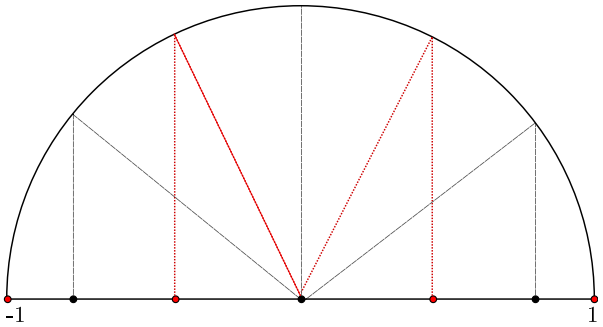

Fig. 5. The location of the Gauss quadrature points (black circles) and the Gauss-Lobatto quadrature points (red circles) used in calculating the sensor value. The Gauss-Lobatto points are only used for the sake of calculating the characteristic length scales $\Delta x, \Delta y, \Delta z$ in Eq. (28), and are not used by the flux-reconstruction numerics.

$$\Delta U = \frac{\partial U}{\partial y_1}\Delta y_1 + \frac{\partial V}{\partial y_2}\Delta y_2 + \frac{\partial W}{\partial y_3}\Delta y_3, \tag{28}$$

where $\partial U/\partial y_1$, $\partial V/\partial y_2$, and $\partial W/\partial y_3$ are calculated using the Lagrange derivative polynomial based on the solution point set. The spacings Δy_1, Δy_2, and Δy_3 are taken as the distance between the two Gauss-Lobatto quadrature points surrounding a given solution point, as seen in Figs. 4 and 5.

The value $(\Delta U/a)_{\text{crit}}$ is used as a threshold to discriminate between weak and strong compressions. Based on this value, the following strategy (Fig. 8) can be adopted for different regions of the flow:

1. Expansions ($\Delta U/a > 0$): flow expansions are amenable to high-order numerics. Here the p-order is set to the maximum prescribed by the user. H-refinement is turned off.
2. Weak Compressions ($(\Delta U/a)_{\text{crit}} < \Delta U/a < 0$): the prescribed p order is decreased linearly from its maximum value ($p = p_{\max}$ at $\Delta U/a = 0$) to $p = 0$ at $\Delta U/a = (\Delta U/a)_{\text{crit}}$ with H-refinement turned off.
3. Strong Compressions ($\Delta U/a < (\Delta U/a)_{\text{crit}}$): these regions are H-refined with $p = 0$ cells, based on the following mapping

$$\left\lceil \frac{H}{H_{\max}} \right\rceil = \frac{(\Delta U/a)^{\frac{1}{2H_{\max}+1}}}{(\Delta U/a)^{\frac{1}{2H_{\max}+1}}_{\min}}, \tag{29}$$

where $(\Delta U/a)_{\min}$ is the sensor value of the strongest compression in the domain. p-refinement is turned off.

In the following we share a few considerations regarding the mathematical and numerical nature of the definition of the ΔU sensor.

First, physical compressions (or expansion) always correspond to numerically negative (or positive) values of the sensor, regardless of Galilean transformations (trivial). The specific numerical value of the sensor, however, is weakly dependent on changes in mesh orientation, which dictates the coordinate transformation law between physical and computational space. The latter dependency is investigated numerically by initializing a left-traveling Mach 1.5 shock with an analytical profile on three different grids using a hyperbolic tangent function given by:

$$f(x) = 0.5(1.0 + \tanh(100.0 + (1 - x))). \tag{30}$$

The velocity field as a function of the x-coordinate was then:

$$u(x) = u_L + f(x)(u_R - u_L), \tag{31}$$

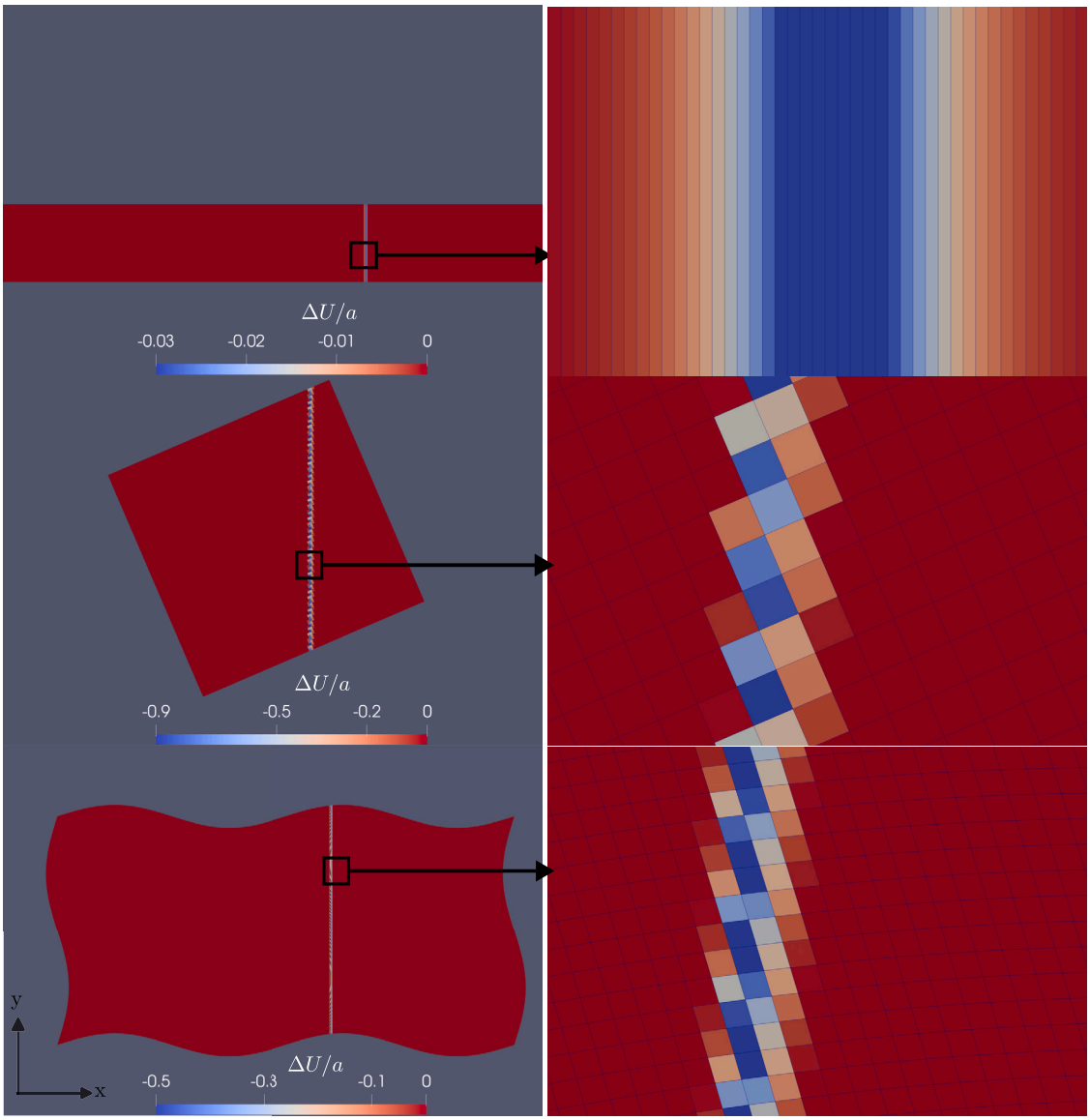

Fig. 6. A Mach 1.5 shock initialized with the same analytical relationship (30) on a one-dimensional uniform mesh (top), a uniform mesh rotated by 45° (middle) and a sinusoidal wavy mesh (bottom). The sensor $\Delta U/a$ values (28) calculated for each cell are shown in colored contours.

where L and R indicate the left and right state, respectively, with values of $\rho_R = 1.2$, $p_R = 101325$, and $u_R = -510$ and $\rho_L = 2.28$, $p_L = 249091$, and $u_L = -274$. The ΔU value for this case was calculated on three meshes (with no time advancement needed): a one-dimensional uniform mesh, a uniform mesh rotated by 45 degrees and a wavy mesh (see Figs. 6 and 7). The ΔU sensor remains strictly negative regardless of mesh orientation, with expected numerical variations among cases.

Another realization is that the thickness of captured shocks is grid dependent, and so is the sensor (by design). As the solution is H-refined, the shock thickness (proportional to $\Delta x, \Delta y, \Delta z$) decreases and so will the velocity jump, ΔU. Thus $\Delta U/a$ remains approximately constant for shocks regardless of refinement level. An expansion wave or other isentropic processes will be resolvable with enough resolution (p or H), at which point any increase of resolution will effectively decrease the value of $\Delta U/a$ as the solution converges. Normalizing ΔU by the speed of sound approximately removed the dependency from the specific equation of state as shown by the results for ideal gasses and high explosives in section 5.5.1. This universality is also supported by the work of Migliorino et al. [52] who tested 6 different fluids, each at 4 different thermodynamic conditions.

Unlike other refinement strategies [39,42,41,44,46,45,47–50], which aim to minimize an error-estimate, our strategy allows the user to prescribe a cost to the computation, by assigning a $p_{\max}$ and an $H_{\max}$, and strategically use this budget in the most critical compression-dominated regions. The deepest H level will be used on the strongest shock in the domain while ensuring that any other shock still has some level of H-refinement.

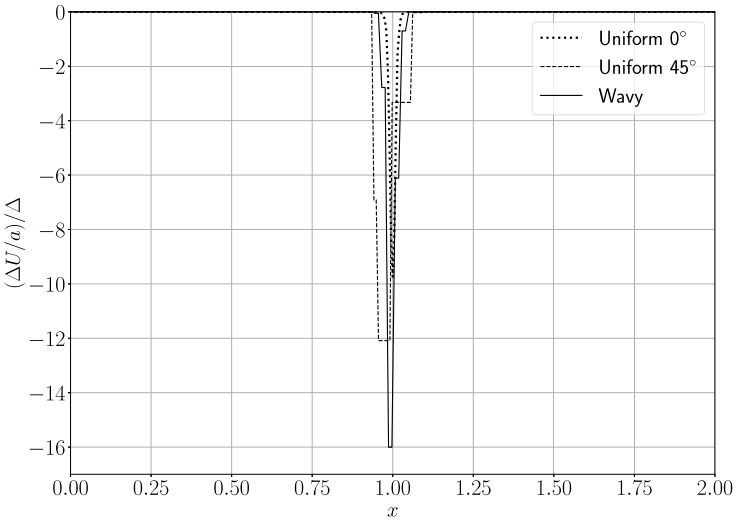

Fig. 7. Profiles of the ΔU sensor (28) normalized by the local mesh size $\Delta \sim \sqrt{A_{\text{cell}}}$ (i.e. the square root of the area of the cell) are shown for the three meshes in Fig. 6.

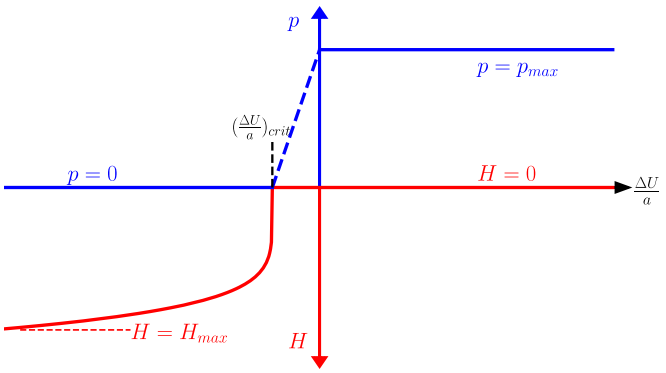

Fig. 8. The refinement strategy for adaptive H-/p-refinement relating H refinement and polynomial order p to the local degree of flow compression/dilatation ($\Delta U/a$) (see text).

The scaling in Eq. (29) is designed to rapidly refine strong compressive flow features ($\Delta U/a < (\Delta U/a)_{\text{crit}}$) when they appear. In various numerical trials it was found that the H-mesh tree should be updated approximately every 1/CFL number of iterations to balance computational cost and accuracy.

4.3. Mortars and implementation

4.3.1. Mortars

With adaptive H-/p-refinement, elements at various levels of refinement must be able to exchange fluxes. Mortars [3], or temporary faces, enable this. Because of the restrictions imposed in section 4.2, $H-$refined cells and $p-$refined cells are never adjacent to one another and a mortar is not needed between them. Thus, the two complex mortar scenarios that arise are those between elements of: 1) different $p-$refinement depths; and 2) of different $H-$refinement depths.

4.3.2. $p-$refinement mortars

When two adjacent elements are at different $p-$refinement levels, the face connecting the elements sees two different sets of flux points that do not match spatially. To allow flux exchanges to occur between the elements at the shared face, the following steps are taken:

1. First, the mortar is created and the mortar order is chosen to be the one of the higher order element. For example, if an order p and an order $p+1$ element are exchanging fluxes, the mortar order is chosen to be $p+1$;
2. The fluxes at the faces are sent to the mortars. For the $p+1$ face, this is just a trivial injection. The p face fluxes are interpolated to the mortar using eq. (22);
3. The numerical fluxes on the mortar are now generated by calling a Riemann solver;

4. The fluxes on the mortar are sent back to the faces. For the $p+1$ face, this is again just a trivial copy. For the p face, the fluxes are interpolated back from the mortar using eq. (22).

While we use a high-order interpolation, the error here is largely dependent on the order of the numerical flux scheme. In our case, both the HLLC and Rusanov schemes are order 1.

4.3.3. $H-$refinement mortars

When adjacent elements are at different depths (depth H and $H+1$), the following strategy is used:

1. First, the face at depth H is temporarily refined to $H+1$;
2. The fluxes at the (4) new children faces are sent to the mortars. Since all faces are at depth $H+1$, this is just a trivial copy for each mortar;
3. The fluxes on the mortars are exchanged using a Riemann solver;
4. The fluxes on the mortars are sent back to the faces. As before, this is just a trivial copy.
5. The face that was temporarily refined is de-refined. An average of the fluxes on the children faces is used as the value of the flux on the parent face.

Here, both our polynomial order and Riemann solver are order 1.

4.4. Code implementation

Algorithm 1 Pseudocode for implementing $H-/p-$refinement.

```
1:  Data: data_structure elements
2:
3:  for each element in elements do
4:      ΔU ← calculate_delta_u(element)
5:      element.setDeltaU(ΔU)
6:  end for
7:
8:  for each element in elements do
9:      ΔU ← element.getDeltaU()
10:     ΔU_scaled ← calculate_delta_u_scaled(ΔU)
11:     if element.needsRefinement(ΔU_scaled) then
12:         element.tagForRefinement()
13:     else if element.needsDerefinement(ΔU_scaled) then
14:         element.tagForDerefinement()
15:     end if
16: end for
17:
18: CheckHDepthDifference()
19:
20: for each element in elements do
21:     if element.isTaggedForRefinement() then
22:         element.refine()
23:     else if element.isTaggedForDerefinement() then
24:         element.derefine()
25:     end if
26: end for
```

Here, in line 20, CheckHDepthDifference is used to ensure that the max allowable H-refinement difference between two adjacent cells will not exceed 1. If an element wants to refine to depth $H+2$ and the adjacent element wishes to be at H, the H depth element is tagged to refine to depth $H+1$. This ensures that the maximum level of H-refinement is used to get the most accurate solution in regions of shocked flow.

5. Numerical tests

5.1. 1D detonation initiation in an ideal gas

To determine the point of transition from quasi-isentropic/weak to non-isentropic/strong compressions, a 1D detonation initiation test case is used [66]. The non-reactive 1D Euler equations are augmented with the transport equation

$$(\rho\lambda)_t + (\rho U\lambda)_x = \rho r_\lambda, \tag{32}$$

where λ is the mass fraction of the products. The EoS is based on the ideal gas law,

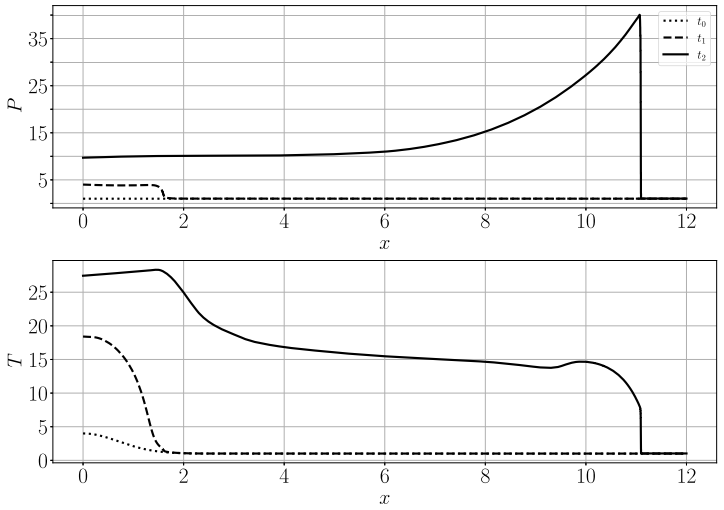

Fig. 9. Idealized detonation-wave formation in an ideal gas, with initial isobaric conditions ($t = t_0$), compression waves coalescing into a shock ($t = t_1$) and formation of a detonation wave ($t = t_2$).

$$e = \frac{P}{\rho(\gamma - 1)} - Q\lambda, \tag{33}$$

augmented by the mass fraction multiplying the heat of reaction Q. An Arrhenius-like reaction rate is assumed in Eq (32) and given by

$$r_\lambda = k(1 - \lambda)e^{-E/(P/\rho)}, \tag{34}$$

where $E = 10$ is the activation energy, and $k = 7$ is the rate constant. In a domain $x \in [0, 12]$, the initial density field is prescribed as

$$\rho = \frac{1}{1 + 3\, e^{-x^2}} \tag{35}$$

with $u = 0$ and $P = 1$ everywhere (at time $t = t_0$). All flow quantities are normalized with respect to the far field ($\rho_\infty = 1$, $p_\infty = 1$, $T_\infty = 1$).

As the flow is allowed to evolve ($t > t_0$), the reaction start to occur at the origin ($x = 0$), sending out right-traveling compression waves. These coalesce into a shock, at approximately $t_1 = 0.5$, which then transitions into a detonation wave at approximately $t_2 = 3.5$ as shown in Fig. 9. As the simulation advances, we calculate the maximum pressure gradient in the simulation at each time and the corresponding $\Delta U/a$. These results are shown in Fig. 10 for a base mesh of $N_x = 3200$ and p-orders [0,1,2].

From this figure, we qualitatively observe a steepening of the dP/dx around t = 0.4, corresponding to values of $\Delta U/a$ rapidly falling below -0.02. Various numerical trials have shown that adopting this value as a threshold to initiate H-refinement leads to over-refinement. A value of $(\Delta U/a)_{\text{crit}} = -0.06$ is rather chosen to maintain the desired balance between stability and accuracy.

5.2. Sod shock tube

The Sod shock tube [67] is a common ideal gas dimensionless test case meant to test a code's ability to handle shocks, contact discontinuities, and rarefaction waves at once. The solution is initialized as

$$\rho = \left\{ \begin{array}{ll} 1, & \text{if } x <= 0.5 \\ 0.125, & \text{if } x > 0.5 \end{array} \right\},\ p = \left\{ \begin{array}{ll} 1, & \text{if } x <= 0.5 \\ 0.1, & \text{if } x > 0.5 \end{array} \right\},\ u = \left\{ \begin{array}{ll} 0, & \text{if } x <= 0.5 \\ 0, & \text{if } x > 0.5 \end{array} \right\} \tag{36}$$

for $x \in [0, 1]$. As time progresses, a rightwards traveling shock and a leftward traveling rarefaction wave develop, separated by a contact discontinuity, resulting in 4 different uniform density regions.

In the results shown in Fig. 11, the simulation is run until $t = 0.2$. The base mesh ($H = 0$) comprises $K = 1000$ cells, with allowable maximum H-/p-levels of $H0/p0$, $H1/p1$ and $H2/p2$. These results are plotted against the analytical solution in Fig. 11 and 12. Note that the values plotted are the cell-averaged values and not the values on the Gauss-Lobatto points. We observe that as the H-refinement levels are increased, the shock sharpens and converges to the analytic solution while p-refinement adds resolution in the rarefaction wave and the (weak) contact discontinuity. Throughout the domain, there are no spurious oscillations associated with the Gibbs phenomenon despite the use of high-order numerics. As seen in Fig. 13, H-refinement is active along the rightward traveling shock while p-refinement is used in other regions, such as the rarefaction wave and contact discontinuity. While the contact discontinuity in this case is weak enough to be exposed to high-order numerics, extensions to this method might require the use of H-refinement for these features as opposed to p-refinement.

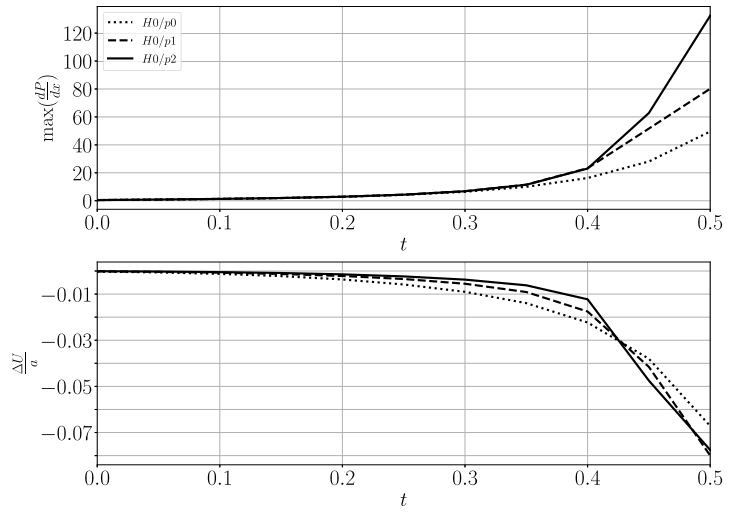

Fig. 10. The maximum dp/dx in the computation domain at any given time for the detonation initiation problem is shown on top for increasing uniform p-refinement of the mesh. The corresponding $\Delta U/a$ at the location of the maximum pressure gradient is shown in the bottom.

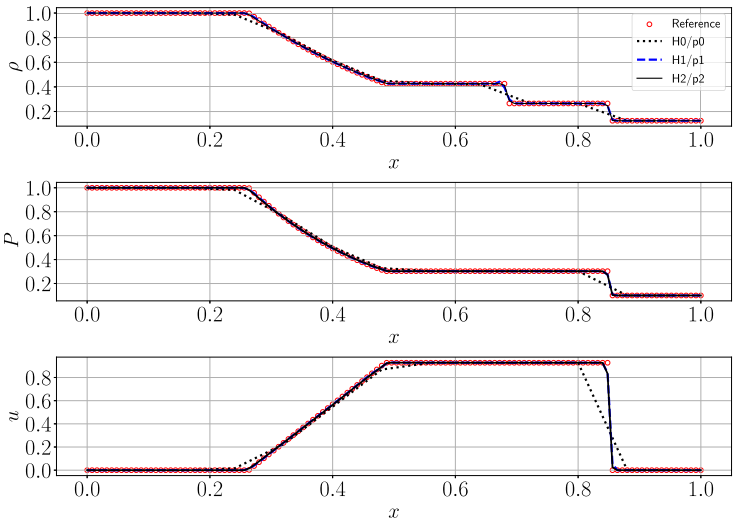

Fig. 11. The Sod shock tube run for increasing adaptive H-/p-refinement. The base mesh is $N_x = 1000$ and $N_y = 1$ and the results are compared against the analytical solution.

To study the efficiency of the refinement strategy, we compare the required wall time of adaptive H-refinement runs against their uniform-grid counterparts in the Sod shock tube case. For the purpose of this study, adaptive p-refinement is disabled. Three sets of H-refinement vs uniform refinement comparisons are made. Maximum H depths of 1, 2, and 3, are used, where $H = 0$ is the base mesh corresponding to K_x x K_y x K_z = [1000 x 1 x 1] cells. The uniform grid cases have a mesh resolution matching the deepest H level from the adaptive calculations. Thought this is inherently a 1D problem, as implemented, the adaptive H-refinement strategy refines in all directions. To accurately match the mesh resolution achieved during adaptive refinement, the uniform mesh was refined in all directions. For example, the refinement equivalent to the AMR 1 case is K_x x K_y x K_z = [2000 x 2 x 2]. The results of this can be seen in Table 4.

When calculating the wall time for the adaptive H-refinement cases, all associated overhead is included. This includes time spent calculating the sensor, ensuring the properly nested criteria is maintained, rebuilding the mesh and mesh connectives, and projecting from parents to children or vice versa. We see that, even with all of this overhead, the benefits of using AMR far out weigh the overhead imposed and the value only increases as more levels of refinement are introduced.

5.3. Shu-Osher problem

The Shu-Osher [68] shock-entropy wave interaction is a dimensionless numerical test case prescribed by the following initial conditions in the domain $x \in [-5, 5]$ for an ideal gas EoS:

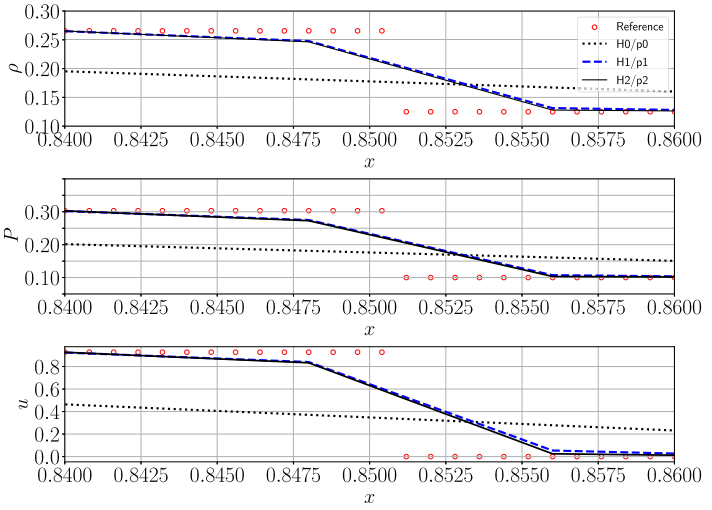

Fig. 12. The Sod shock tube run for increasing adaptive H-/p-refinement zoomed in at the shock location.

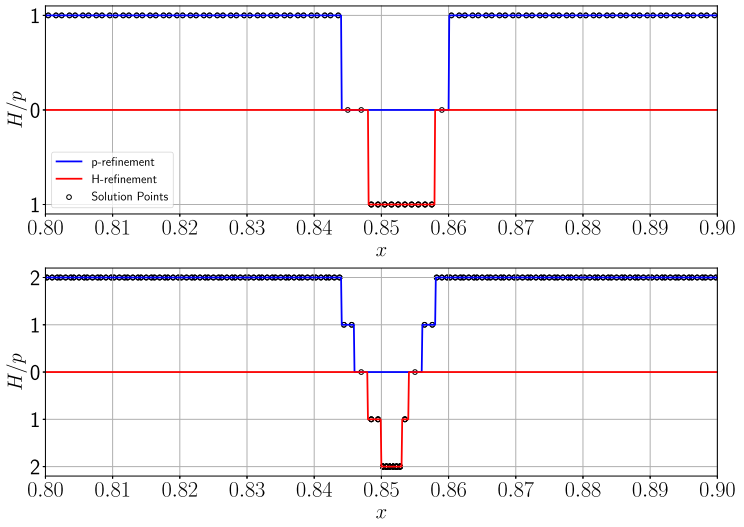

Fig. 13. Adaptive H-/p-refinement structure of the H1/p1 (top) and H2/p2 (bottom) Sod case.

Table 4
Table of wall times for the Sod case.

	Depth 1	Depth 2	Depth 3
Adaptive H (s)	49	86	368
Uniform (s)	68	258	8246
Speed-up	1.38	3	22.4

$$[\rho, P, u_x] = \left\{ \begin{array}{ll} 3.857143,\ 10.\bar{3},\ 2.629369 & \text{if } x <= -4 \\ 1.0 + 0.2\sin(5x)\,,1.0\,,0 & \text{if } x > -4 \end{array} \right\}. \tag{37}$$

A shock wave propagates into a resting fluid comprising a small amplitude harmonic density perturbation.

The results in Fig. 15 are advanced to $t = 1.8$. To plot the results, the value of density was shifted by 4 as the refinement levels were increased. Similar to the Sod case study, the baseline $H = 0$ mesh is kept at $K = 2000$ cells, while the maximum allowable H-refinement depth and polynomial order are varied from $H0/p0$ to $H2/p2$. While there is no analytical solution to this test case, the results generated on a mesh of 50,000 cells are used as a reference. The baseline $H0/p0$ case exhibits the expected shock smearing and damped oscillations trailing the shock. As the H-refinement and p-refinement levels are increased, the shock steepens and the correct amplitudes of the smooth oscillations are recovered. While there are some spurious oscillations in the trailing oscillations for $H = 2, P = 2$ case, the sensor quickly finds and fixes these oscillations as seen in Fig. 16, the same run continued on to t = 2.0.

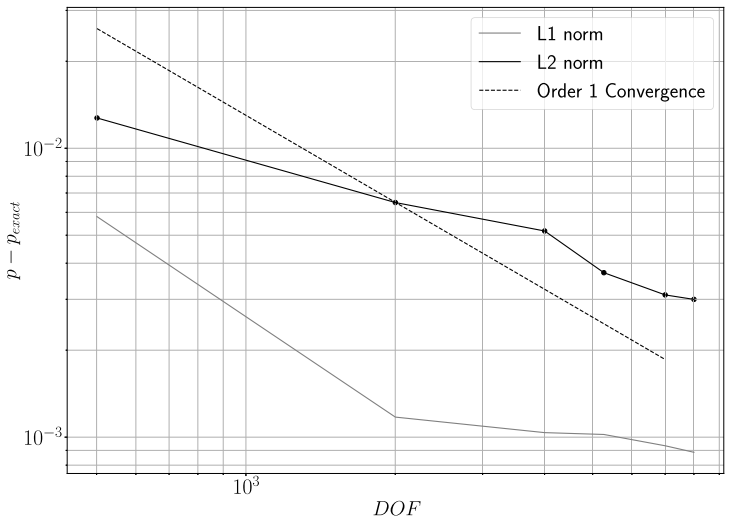

Fig. 14. The L1 and L2 norms of the error on pressure for the Sod shock tube problem compared against the order 1 convergence rate (dashed line).

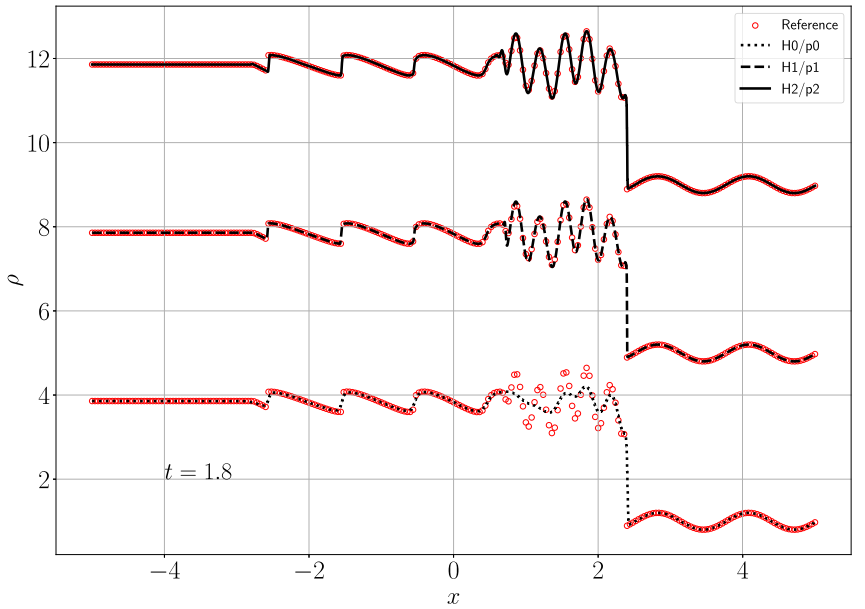

Fig. 15. The Shu-Osher test case for increasing H-/*p*-refinement levels compared against the reference solution. The base mesh for the computational results is $N_x = 2000$ and $N_y = 1$. The reference solution mesh is $N_x = 50{,}000$ and $N_y = 1$.

To test the robustness of this methodology, the computational domain of this problem is distorted with the following mapping function:

$$(x', y') = (x + 0.05L\sin(\frac{2\pi y}{L}), y + 0.05L\sin(\frac{2\pi x}{L})), \tag{38}$$

where (x', y') are the coordinates of the distorted grid and $L = 10$. The shock and the small scale oscillations are captured with minimal spurious oscillations and minimal diffusion of the shock, in spite of the high degree of grid skewness, especially in the center of the domain. See Fig. 17.

A time comparison of the adaptive vs non-adaptive strategy was done for both the Cartesian and wavy mesh cases. As before with the Sod shock tube, only H-refinement was considered to ensure a stable run. Again, we see that as the mesh size increases, the computational time saved increases drastically. Note that due to the increased mesh size in the wavy mesh cases, both the uniform and adaptive cases were run on twice as many processors as the Cartesian cases. See Tables 5 and 6.

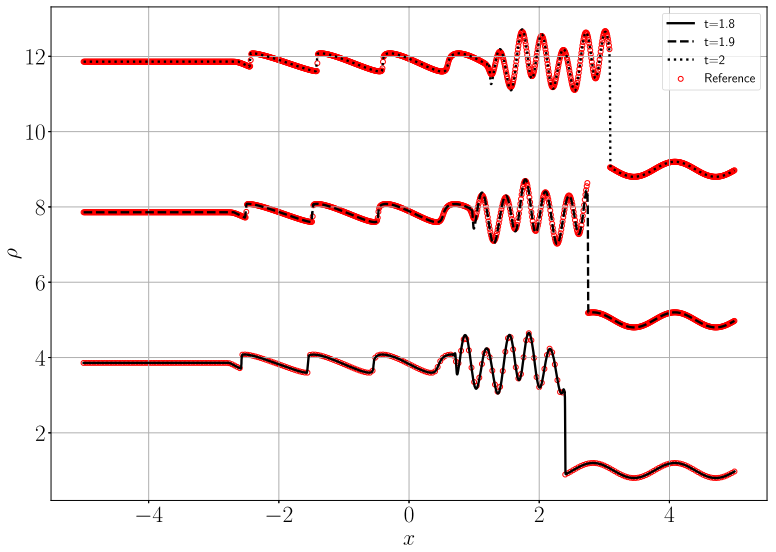

Fig. 16. The Shu-Osher test case for $H = 2$ and $p = 2$ refinement level run until t = 2.0. The small oscillations in the t = 1.8 case have been killed by the time the simulation reached t = 2.0.

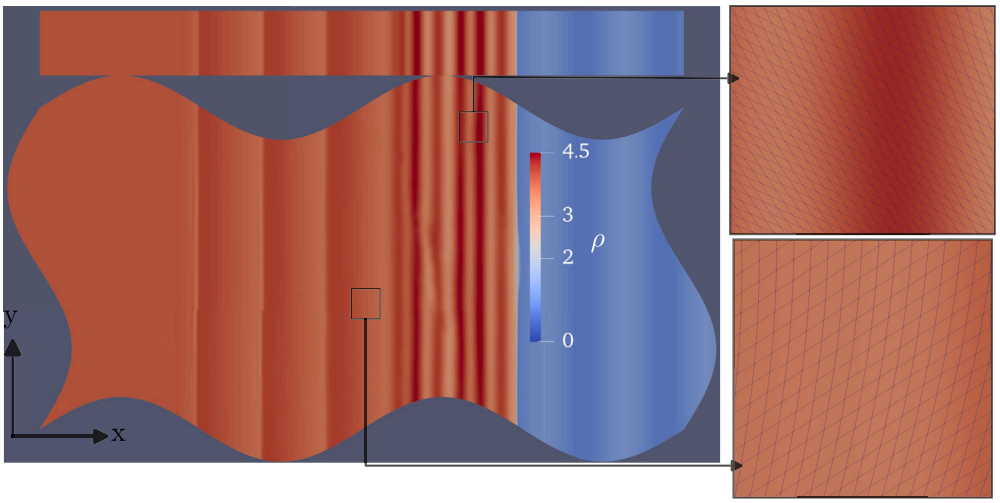

Fig. 17. Density contours of the Shu-Osher problem using a wavy mesh grid with $N_x = 500$ and $N_y = 250$ with H2/p1. The top Cartesian mesh has $N_x = 1000$ and $N_y = 1$ with H1/p1.

Table 5
Table of wall times for the Cartesian Shu-Osher case.

	Depth 1	Depth 2
Adaptive H (s)	176	1011
Uniform (s)	760	16365
Speed-up	4.3	13.4

Table 6
Table of wall times for the wavy mesh Shu-Osher case.

	Depth 1
Adaptive H (s)	3200
Uniform (s)	7873
Speed-up	2.5

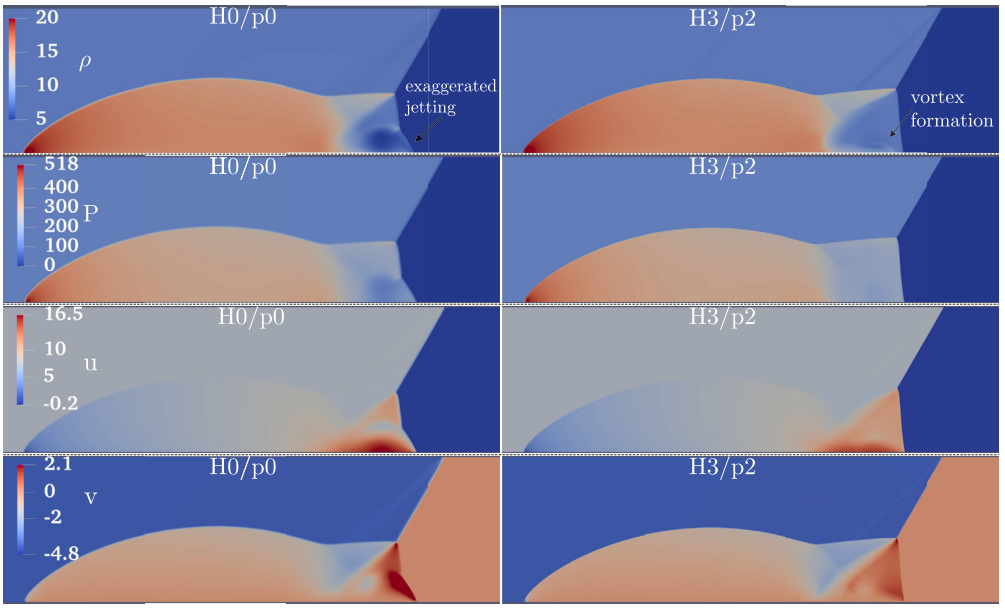

Fig. 18. The double Mach reflection problem run with and without adaptive mesh refinement. Both cases used the same base mesh of $N_x = 1000$ and $N_y = 250$. The H = 0 and p = 0 case (left) is compared to a $H = 3$, $p = 2$ case (right).

5.4. Double Mach reflection

The double Mach reflection problem is a dimensionless 2D unsteady problem in an ideal gas introduced by Woodward and Colella [69]. Using a computational domain of $\Omega = [0,4] \times [0,1]$, we first define the left and right states as

$$\begin{Bmatrix} \rho_L & P_L & u_L & v_L \\ \rho_R & P_R & u_R & v_R \end{Bmatrix} = \begin{Bmatrix} 8 & 116.5 & 4.125\sqrt{3} & -4.125 \\ 1.4 & 1 & 0 & 0 \end{Bmatrix}. \tag{39}$$

Here the top row, or the left-state, are the initial conditions for locations that are to the left of a line going from the point $(0, \frac{1}{6})$ to $(1, \frac{1}{6} + \frac{1}{\sqrt{3}})$ while the bottom row, or the right state, are the initial conditions for all other points. The boundary conditions are:

1. The left boundary of the domain is a constant inlet, supplying the left state.
2. For $x < \frac{1}{6}$, the bottom wall is assigned the left state value. This ensures the shock will remain attached at $x = \frac{1}{6}$. For $x >= \frac{1}{6}$, a reflecting wall boundary condition is applied.
3. At the right wall, an outlet condition is applied.
4. Finally, at the top wall, a time dependent inlet condition is applied. Defining $x_s = 1/6 + (1 + 2Mt)/\sqrt{3}$, we set any location along the top wall where $x < x_s$ as the left state and $x >= x_s$ as the right state.

This results in a semi-infinite Mach 10 shock which propagates at a 60° angle with respect to the x axis.

As seen in Fig. 18, the $H0/p0$ results are significantly more diffusive than the adaptive case, as expected. The jetting expected near the wall is exaggerated by the $H0/p0$ case and features such as the vortex formed near the shock front due to a high-speed jet have completely dissipated. The adaptive H-/p-calculations do not exhibit spurious oscillations in the domain. Fig. 19 shows the H-/p-structure for this problem. H-refinement is concentrated along the shock front while p-refinement is active elsewhere, such as the vortex region.

5.5. Detonation waves in high-explosives

5.5.1. Sensor validation

To test the validity of the sensor for various fluids and EoS, a shock to detonation transition in high-explosives is modeled. Using the equations and values outlined in section 3.2, a strong shock is introduced in PBX 9501 via impact. The material, initially at $\rho = 1836$ kg/m^3 and 0 Pa, impacts a wall at 500 m/s, which results in a shock wave. Chemical reactions behind the shock release energy which strengthens the shock and causes a detonation wave to form.

Much like the ideal gas case, the maximum dP/dx is tracked alongside with the corresponding $\Delta U/a$. As seen in Fig. 20, the value of $\Delta U/a = -0.06$ is again a suitable demarcator between the near-isentropic and nonlinearly steepening wave behavior. To ensure that this value is converged in space as well as time, the spatial values of the sensor are plotted for each case and the results are presented in Fig. 21. From this figure, we see that our sensor is grid independent and works well for different EoS.

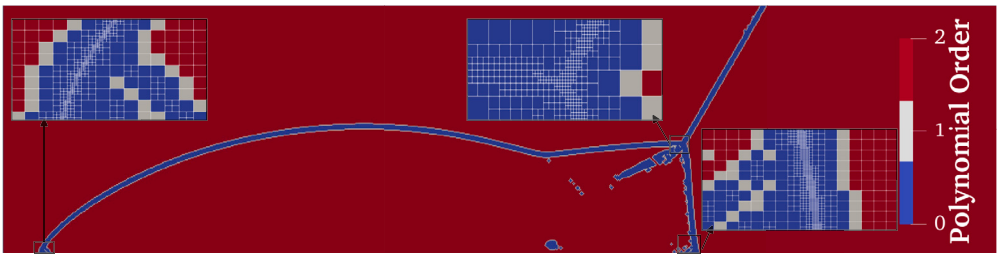

Fig. 19. The H-refinement and p-refinement structure of the double Mach reflection problem.

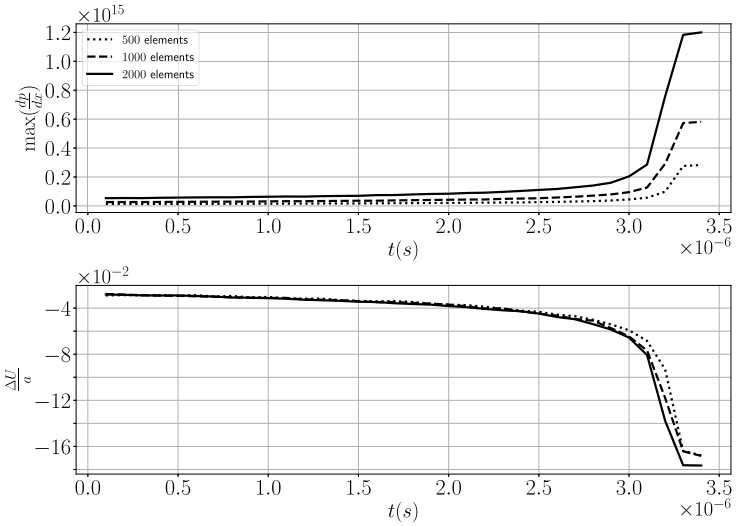

Fig. 20. The maximum dp/dx in the computation domain at any given time for the shock to detonation problem is shown on top for increasing uniform H-refinement of the mesh. The resultant $\Delta U/a$ at the location of the maximum is shown in the bottom.

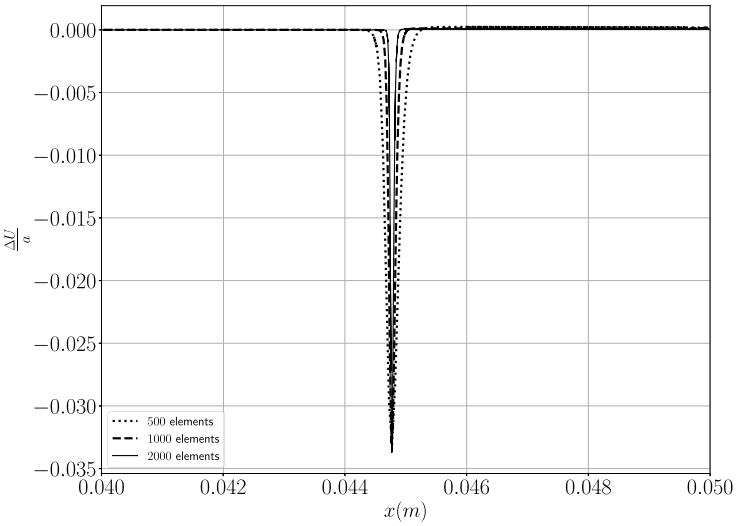

Fig. 21. The $\Delta U/a$ sensor in space for the shock to detonation transition case for uniform H-refinement.

5.5.2. 2D detonation wave

Having validated the sensor for a Mie Grüneisen based EoS, we model a two-dimensional detonation in a high-explosive. A computational domain of $\Omega = [0,.03]$ x [-0.001,0.01] is used. For $x < .01$ and $0 < y < .009$ the domain is initialized as:

$$[P, \rho, \lambda] = [55GPa, 2157.259\frac{kg}{m^3}, 1]. \tag{40}$$

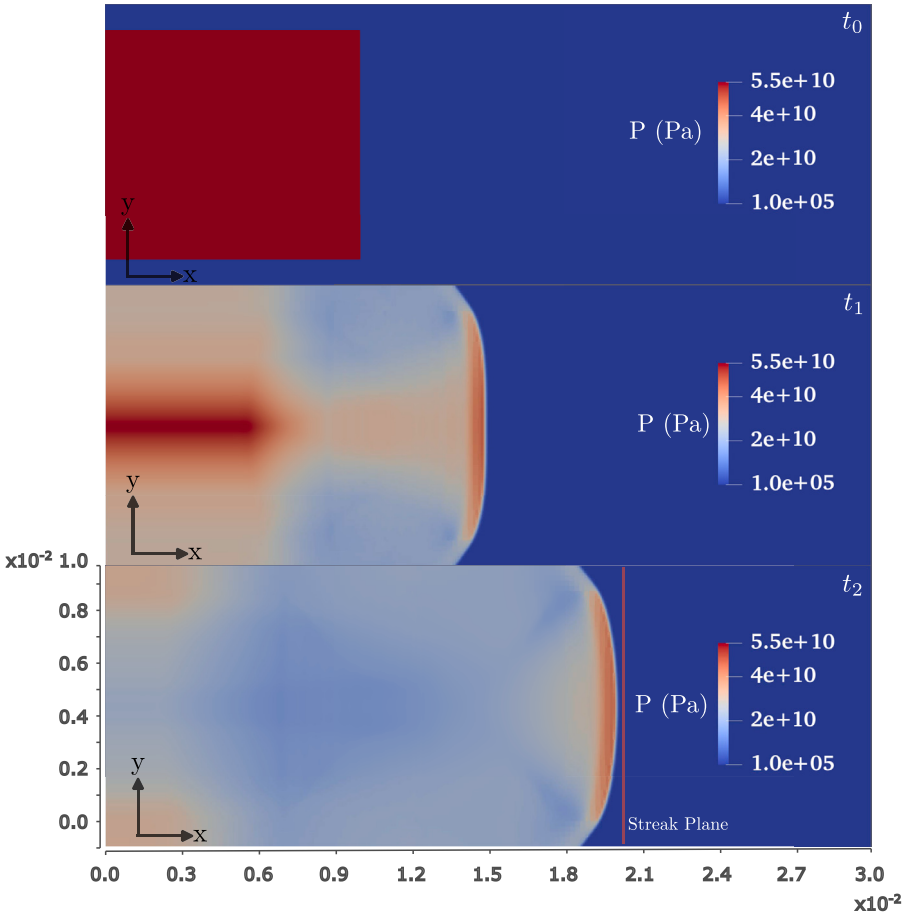

Fig. 22. A detonation inside the HMX-based explosive PBX-9501. The initial conditions are on top, followed by t = 1.5 μs in the middle and t = 3 μs on the bottom.

For the rest of the domain, the initial conditions are:

$$[P, \rho, \lambda] = [101325 Pa, 1836 \frac{kg}{m^3}, 0]. \tag{41}$$

At $y < 0$ or $y > 0.009$, the domain is filled with an inert material (the conditions given by equation (41) with reactions disabled). This resulted in a detonation wave forming at $x = 0.01$ that then propagates into the rest of the material (Fig. 22). In experiments, a high-explosive is surrounded by an inert material, such as air or metal. The presence of these inert materials causes the detonation wave to curve [70] and these curvatures have significant effect on the detonation process and must be accurately modeled [71,72].

We calculate the shock arrival time along the location $x \sim 0.02$ m by checking when the pressure at any given y-location exceeds 20 GPa. Once this pressure value is reached, the detonation has arrived at the given y-location and the results are plotted in Fig. 24. Due to the nature of adaptive refinement (which moves the locations of the solution points) the sampled x location was not exactly the same across various levels of grid refinement. The values chosen, listed by increasing refinement levels, are $x = (0.02\text{m}, 0.0205\text{m}, 0.02075\text{m}, \text{and } 0.020875\text{m})$.

The computational results obtained are compared against experimental results obtained by Chapman et al. [73]. The experimental explosive composite consisted of two particle size ranges of RDX, class 3 (Lot No. BAE13E011-101) and class 5 (Lot No. BAE15D097-002) mixed at a 1:1 ratio. The RDX crystals were then mixed at an 85 wt.% solids loading into a binder of 73.6 wt.% hydroxyl-terminated polybutadiene resin (HTPB), 15.0 wt.% isodecyl pelargonate plasticizer (IDP), and 11.4 wt.% modified MDI isocyanate curative. The mixture was mixed thoroughly using an acoustic mixer (Resodyn - Lab Ram) at 60% power for 6 min. The RDX/HTPB mixture was then hand-packed into a Teflon mold creating a charge with external dimensions of 6.8 mm x 9.1 mm x 70 mm and the sample was allowed to cure for 96 hr. To reduce inhomogeneities from packing, 5 mm was cut from the end faces of the cured sample using a precision wire saw (Princeton Scientific Model: WS25B) creating a sample that was 60 mm in length. This length to cross-sectional area ratio was chosen to allow for a fully developed steady detonation to form before the detonation reached the downstream breakout surface [74]. Before performing the detonation experiment the sample was scanned using micro-X-ray computed tomography (Skyscan 1272) to confirm that no internal voids greater than 50 μm were present in the cast composite material. The sample density was found to be 1.561 g/cc as measured using the Archimedes method with water as the working fluid. The detonation front breakout was observed using a Hamamatsu digital streak camera (C13410) with a sensor resolution of 1024x1344 pixels (Orca

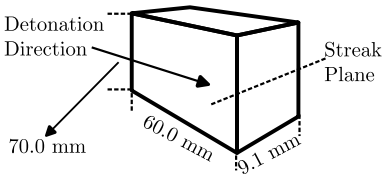

Fig. 23. Experimental setup showing streak plane placement.

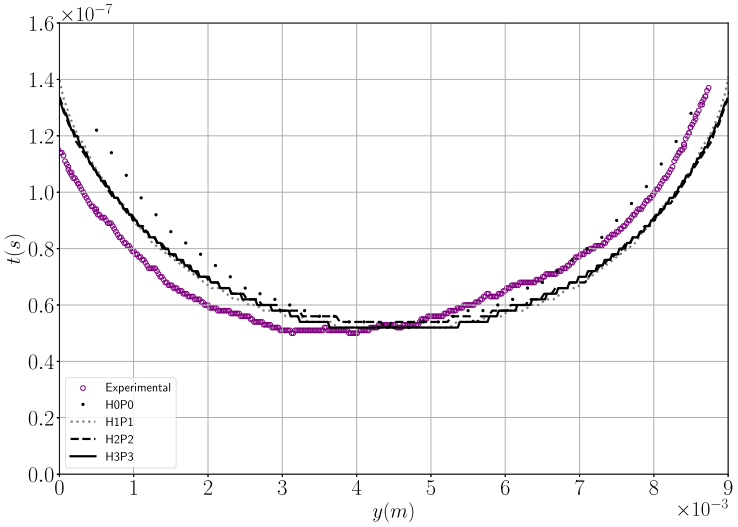

Fig. 24. Shock arrival times for various levels of refinement for the high-explosive PBX 9501 compared to experimental results of RDX-HTPB.

2a CMOS). A Nikon 70-300 telephoto lens was combined with optical bellows and a 45-deg corner turning mirror to provide sight access to the sample placed behind a half-inch thick polycarbonate window. A streak slit opening of 30 μm, and a total streak record time of 1 μs was used, similar to methods used by Montoya et al. [75]. The streak plane was oriented as shown in Fig. 23, with the plane centered vertically in the height direction of the sample. To improve the light emission of the shock during breakout, a 50 μm thick coating of alumina powder (Sigma Aldrich <10 μm) was applied to the downstream surface of the sample using an airbrush and a suspension of alumina in 90% ethanol alcohol [76]. The steak image captured for each sample was analyzed in MATLAB using the built-in canny method edge detection function [77]. Both the front and back edge of the breakout streak profile were found and an average of the two was calculated as the final data presented. The length scale of each streak image was calibrated using a captured still image of the corresponding sample with the camera and a fully open slit. A temporal correction was also applied using a provided formula from Hamamatsu. It is important to note that the experimental results used a different high-explosive which results in slightly different shape to the arrival times. The results were also shifted temporally so that the leading front of the detonations arrived at the same time. However, qualitatively, we see a similar structure in the detonation shape. While there is a lack of monotonicity in the computational results, this is a result of the adaptive nature of the problem. As discussed earlier, the sampled x locations varied slightly among the different grids. Particularly near the center line, where the detonation experiences minimal curvature effects, the x-plane is positioned closer to the advancing detonation wave with lower levels of refinement. Consequently, this leads to a slight delay in the arrival time with higher refinement levels. However, as we traverse away from the center line, heightened refinement mitigates the diffusion encountered in the curved regions of the detonation, resulting in an accelerated arrival time.

6. Conclusion

We have demonstrated a novel H-/p-refinement strategy that combines adaptive mesh refinement with high-order FR methods for the purpose of modeling shock dominated flow. The adaptive refinement sensor, $\Delta U/a$, guides the AMR algorithm in differentiating between regions that require adaptive H-refinement (to resolve shocks) and adaptive p-refinement (to resolve smooth flow features). On cell faces, the FR scheme ensures continuous fluxes between elements.

A detonation initiation in an ideal gas as well as a shock to detonation transition in a high-explosive are used to find a critical value of the sensor to switch between H- and p-refinement. Using this critical value, a variety of 1D and 2D problems are tested in very different fluids utilizing different equations of state. In each of these cases, the adaptive H-/p-strategy performs well at capturing the shock, resolving small scale features, and limiting any spurious oscillations. Furthermore, the methodology is also shown to work in deformed meshes utilizing grid transformations.

Limitations of the current approach include not addressing the presence of strong contact discontinuities or tackling interactions between shocks and inherently three-dimensional flow features such as turbulence, which were out of the scope of the funding supporting the current effort.

Stronger contact discontinuities may require an ad-hoc sensor for the local triggering of H-refinement, and a Ducros-like sensor may be required for shock-turbulence interaction problems. We anticipate that future developments of this H-/p-technique, in fact, will rely on multiple sensors targeting different phenomena, retaining the overarching approach of capturing discontinuities with lower order H-refined cells, and resolvable flow quantities with p-refinement.

Another limitation of the proposed method is the lack of a de-p-refinement strategy, which can be achieved by modulating the polynomial order along the $\Delta U > 0$ axis. We anticipate this being particularly useful for vortex-dominated problems characterized by localized strong vortical regions, for example.

CRediT authorship contribution statement

Vaibhav Rajora: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. **Tariq Aslam:** Conceptualization, Funding acquisition, Investigation, Methodology, Project administration, Supervision, Writing – original draft, Writing – review & editing. **Wesley W. Chapman:** Formal analysis, Investigation, Methodology, Writing – original draft. **Chad Meyer:** Conceptualization, Funding acquisition, Investigation, Project administration, Supervision. **Steve F. Son:** Conceptualization, Funding acquisition, Investigation, Project administration, Writing – original draft. **Carlo Scalo:** Conceptualization, Formal analysis, Funding acquisition, Investigation, Project administration, Supervision, Validation, Writing – original draft, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

Acknowledgements

This work was supported by the Advanced Simulation and Computing (ASC) Program through Los Alamos National Laboratory. Los Alamos National Laboratory is operated by Triad National Security, LLC, for the National Nuclear Security Administration of U.S. Department of Energy (Contract No. 89233218CNA000001).

References

[1] Sergei K. Godunov, I. Bohachevsky, Finite difference method for numerical computation of discontinuous solutions of the equations of fluid dynamics, Mat. Sb. 47 (3) (1959) 271–306.
[2] Eleuterio F. Toro, Riemann Solvers and Numerical Methods for Fluid Dynamics: a Practical Introduction, Springer Science & Business Media, 2013.
[3] David A. Kopriva, John H. Kolias, A conservative staggered-grid Chebyshev multidomain method for compressible flows, J. Comput. Phys. 125 (1) (1996) 244–261.
[4] David A. Kopriva, A staggered-grid multidomain spectral method for the compressible Navier-Stokes equations, J. Comput. Phys. 143 (1) (1998) 125–158.
[5] E.F. Toro, M. Spruce, W. Speares, Restoration of the contact surface in the HLL–Riemann Solver, Shock Waves 4 (1994) 25–34.
[6] V.V. Rusanov, Calculation of interaction of Non–Steady Shock Waves with obstacles, USSR Comput. Math. Math. Phys. 1 (1961) 267–279.
[7] Philip L. Roe, Approximate Riemann solvers, parameter vectors, and difference schemes, J. Comput. Phys. 43 (2) (1981) 357–372.
[8] Hung T. Huynh, A flux reconstruction approach to high-order schemes including discontinuous Galerkin methods, in: 18th AIAA Computational Fluid Dynamics Conference, 2007, p. 4079.
[9] Hung T. Huynh, A reconstruction approach to high-order schemes including discontinuous Galerkin for diffusion, in: 47th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition, 2009, p. 403.
[10] H.T. Huynh, Zhi J. Wang, Peter E. Vincent, High-order methods for computational fluid dynamics: a brief review of compact differential formulations on unstructured grids, Comput. Fluids 98 (2014) 209–220.
[11] Chunlei Liang, Christopher Cox, Michael Plesniak, A comparison of computational efficiencies of spectral difference method and correction procedure via reconstruction, J. Comput. Phys. 239 (2013) 138–146.
[12] Chunlei Liang, Koji Miyaji, Bin Zhang, An efficient correction procedure via reconstruction for simulation of viscous flow on moving and deforming domains, J. Comput. Phys. 256 (2014) 55–68.
[13] David Gottlieb, Chi-Wang Shu, Alex Solomonoff, Hervé Vandeven, On the Gibbs phenomenon I: recovering exponential accuracy from the Fourier partial sum of a nonperiodic analytic function, J. Comput. Appl. Math. 43 (1–2) (1992) 81–98.
[14] Cornelius Lanczos, John Boyd, Discourse on Fourier Series, SIAM, 2016.
[15] David Gottlieb, Steven A. Orszag, Numerical Analysis of Spectral Methods: Theory and Applications, SIAM, 1977.
[16] David A. Kopriva, Shock-fitted multidomain solution of supersonic flows, Comput. Methods Appl. Mech. Eng. 175 (3–4) (1999) 383–394.
[17] Wei Cai, High-order hybrid numerical simulations of two-dimensional detonation waves, AIAA J. 33 (7) (1995) 1248–1255.
[18] Gregory P. Brooks, Joseph M. Powers, Standardized pseudospectral formulation of the inviscid supersonic blunt body problem, J. Comput. Phys. 197 (1) (2004) 58–85.

[19] Pradeep Singh Rawat, Xiaolin Zhong, On high-order shock-fitting and front-tracking schemes for numerical simulation of shock–disturbance interactions, J. Comput. Phys. 229 (19) (2010) 6744–6780.
[20] Christopher M. Romick, Tariq D. Aslam, High-order shock-fitted detonation propagation in high explosives, J. Comput. Phys. 332 (2017) 210–235.
[21] John VonNeumann, Robert D. Richtmyer, A method for the numerical calculation of hydrodynamic shocks, J. Appl. Phys. 21 (3) (1950) 232–237.
[22] Antony Jameson, Wolfgang Schmidt, Eli Turkel, Numerical solution of the Euler equations by finite volume methods using Runge Kutta time stepping schemes, in: 14th Fluid and Plasma Dynamics Conference, 1981, p. 1259.
[23] Andrew W. Cook, William H. Cabot, A high-wavenumber viscosity for high-resolution numerical methods, J. Comput. Phys. 195 (2) (2004) 594–601.
[24] Takanori Haga, Soshi Kawai, Toward accurate simulation of shockwave-turbulence interaction on unstructured meshes: a coupling of high-order fr and lad schemes, in: 21st AIAA Computational Fluid Dynamics Conference, 2013, p. 3065.
[25] Soshi Kawai, Sanjiva K. Lele, Localized artificial diffusivity scheme for discontinuity capturing on curvilinear meshes, J. Comput. Phys. 227 (22) (2008) 9498–9526.
[26] Matthew Anderson, Eric W. Hirschmann, Steven L. Liebling, David Neilsen, Relativistic mhd with adaptive mesh refinement, Class. Quantum Gravity 23 (22) (2006) 6503.
[27] Michael Dumbser, Olindo Zanotti, Arturo Hidalgo, Dinshaw S. Balsara, Ader-weno finite volume schemes with space–time adaptive mesh refinement, J. Comput. Phys. 248 (2013) 257–286.
[28] George Shu Heng Pau, John B. Bell, Ann S. Almgren, Kirsten M. Fagnan, Michael J. Lijewski, An adaptive mesh refinement algorithm for compressible two-phase flow in porous media, Comput. Geosci. 16 (3) (2012) 577–592.
[29] Xiaodong Chen, Vigor Yang, Thickness-based adaptive mesh refinement methods for multi-phase flow simulations with thin regions, J. Comput. Phys. 269 (2014) 22–39.
[30] Stéphane Popinet, Graham Rickard, A tree-based solver for adaptive ocean modelling, Ocean Model. 16 (3–4) (2007) 224–249.
[31] Arpit Tiwari, Jonathan B. Freund, Carlos Pantano, A diffuse interface model with immiscibility preservation, J. Comput. Phys. 252 (2013) 290–309.
[32] Ann S. Almgren, Thomas Buttke, Phillip Colella, A fast adaptive vortex method in three dimensions, J. Comput. Phys. 113 (2) (1994) 177–200.
[33] Stéphane Popinet, Gerris: a tree-based adaptive solver for the incompressible Euler equations in complex geometries, J. Comput. Phys. 190 (2) (2003) 572–600.
[34] Andrew Wissink, Sean Kamkar, Thomas Pulliam, Jayanarayanan Sitaraman, Venkateswaran Sankaran, Cartesian adaptive mesh refinement for rotorcraft wake resolution, in: 28th AIAA Applied Aerodynamics Conference, 2010, p. 4554.
[35] Neal M. Chaderjian, Advances in rotor performance and turbulent wake simulation using des and adaptive mesh refinement, 2012.
[36] Neal M. Chaderjian, Jasim U. Ahmad, Detached eddy simulation of the uh-60 rotor wake using adaptive mesh refinement, 2012.
[37] Xinran Zhao, Carlo Scalo, Helicity dynamics in reconnection events of topologically complex vortex flows, J. Fluid Mech. 920 (2021) A30.
[38] Xinran Zhao, Zongxin Yu, Jean-Baptiste Chapelier, Carlo Scalo, Direct numerical and large-eddy simulation of trefoil knotted vortices, J. Fluid Mech. 910 (2021) A31.
[39] Marsha J. Berger, Jospeh Oliger, Adaptive mesh refinement for hyperbolic partial differential equations, J. Comput. Phys. 53 (1984).
[40] Marsha J. Berger, P. Colella, Local adaptive mesh refinement for shock hydrodynamics, J. Comput. Phys. 82 (1989) 64–84.
[41] Catherine Mavriplis, A posteriori error estimators for adaptive spectral element techniques, in: Proceedings of the Eighth GAMM-Conference on Numerical Methods in Fluid Mechanics, Springer, 1990, pp. 333–342.
[42] Catherine Mavriplis, Adaptive mesh strategies for the spectral element method, Comput. Methods Appl. Mech. Eng. 116 (1–4) (1994) 77–86.
[43] I. Lomtev, C.B. Quillen, G.E. Karniadakis, Spectral/hp methods for viscous compressible flows on unstructured 2d meshes, J. Comput. Phys. 144 (2) (1998) 325–357.
[44] Kim S. Bey, An hp adaptive discontinuous Galerkin method for hyperbolic conservation laws, Technical report, 1994.
[45] Karen D. Devine, Joseph E. Flaherty, Parallel adaptive hp-refinement techniques for conservation laws, Appl. Numer. Math. 20 (1996).
[46] Karen Marie Dragon Devine, An adaptive hp-finite element method with dynamic load balancing for the solution of hyperbolic conservation laws on massively parallel computers, 1995.
[47] Antonio Huerta, E. Casoni, Jaime Peraire, A simple shock-capturing technique for high-order discontinuous Galerkin methods, Int. J. Numer. Methods Fluids 69 (10) (2012) 1614–1632.
[48] Per-Olof Persson, Benjamin Stamm, A discontinuous Galerkin method for shock capturing using a mixed high-order and sub-grid low-order approximation space, J. Comput. Phys. 449 (2022) 110765.
[49] Matthias Sonntag, Claus-Dieter Munz, Efficient parallelization of a shock capturing for discontinuous Galerkin methods using finite volume sub-cells, J. Sci. Comput. 70 (2017) 1262–1289.
[50] Pascal Mossier, Andrea Beck, Claus-Dieter Munz, A p-adaptive discontinuous Galerkin method with hp-shock capturing, J. Sci. Comput. 91 (2022).
[51] Jingjing Yang, Bin Zhanga, Chunlei Lianga, Yongwu Rong, A high-order flux reconstruction method with adaptive mesh refinement and artificial diffusivity on unstructured moving/deforming mesh for shock capturing, Comput. Fluids 139 (2016).
[52] M.T. Migliorino, C. Scalo, Heat-induced planar shock waves in supercritical fluids, Shock Waves 30 (2) (2020) 153–167.
[53] Ralph Menikoff, Empirical equations of state for solids, in: ShockWave Science and Technology Reference Library, 2007, pp. 143–188.
[54] Yakov B. Zel'Dovich, On the theory of the propagation of detonation in gaseous systems, Zh. Èksp. Teor. Fiz. 10 (1940) 542–568.
[55] John von Neumann, Progress Report to the National Defense Research Committee Div. B, OSRD-549, 1942.
[56] Werner Döring, Über den detonationsvorgang in gasen, Ann. Phys. 435 (6–7) (1943) 421–436.
[57] William C. Davis, Equation of state for detonation products, Technical report, Los Alamos National Lab. (LANL), Los Alamos, NM (United States), 1998.
[58] William C. Davis, Complete equation of state for unreacted solid explosive, Combust. Flame 120 (3) (2000) 399–403.
[59] D. Scott Stewart, Sunhee Yoo, William C. Davis, Equation of state for modeling the detonation reaction zone, in: 12th Symp. (Intl) on Detonation, 2002, pp. 1–11.
[60] Tariq D. Aslam, Matthew A. Price, Christopher Ticknor, Joshua D. Coe, Jeffery A. Leiding, Marvin A. Zocher, Awsd calibration for the hmx based explosive pbx 9501, in: AIP Conference Proceedings, vol. 2272, AIP Publishing, 2020.
[61] B. Wescott, Donald Stewart, W. Davis, Equation of state and reaction rate for condensed-phase explosives, J. Appl. Phys. 98 (053514–053514) (2005) 10.
[62] Tariq Dennis Aslam, Nirmal Kumar Rai, Stephen Arthur Andrews, Matthew Anthony Price, David Benjamin Culp, Vaibhav Rajora, On two-phase pressure and temperature equilibration with Mie-Grüneisen equations of state, Technical report, Los Alamos National Lab. (LANL), Los Alamos, NM (United States), 2021.
[63] David E. Lambert, D. Scott Stewart, Sunhee Yoo, Bradley L. Wescott, Experimental validation of detonation shock dynamics in condensed explosives, J. Fluid Mech. 546 (2006) 227–253.
[64] Prateek Gupta, Guido Lodato, Carlo Scalo, Spectral energy cascade in thermoacoustic shock waves, J. Fluid Mech. 831 (2017) 358–393.
[65] Krzysztof J. Fidkowski, Philip L. Roe, An entropy adjoint approach to mesh refinement, SIAM J. Sci. Comput. 32 (3) (2010) 1261–1287.
[66] Shaojie Xu, Tariq Aslam, D. Scott Stewart, High resolution numerical simulation of ideal and non-ideal compressible reacting flows with embedded internal boundaries, Combust. Theory Model. 1 (1997).
[67] Gary A. Sod, A survey of several finite difference methods for systems of nonlinear hyperbolic conservation laws, J. Comput. Phys. 27 (1) (1978) 1–31.
[68] Chi-Wang Shu, Stanley Osher, Efficient implementation of essentially non-oscillatory shock-capturing schemes, II, J. Comput. Phys. 83 (1) (1989) 32–78.
[69] Paul Woodward, Phillip Colella, The numerical simulation of two-dimensional fluid flow with strong shocks, J. Comput. Phys. 54 (1) (1984) 115–173.
[70] P. Clark Souers, Raul Garza, Kinetic Information from Detonation Front Curvature, 1998.
[71] J.B. Bdzil, T.D. Aslam, D.S. Stewart, Curved detonation fronts in solid explosives: collisions and boundary interactions, Technical report, Los Alamos National Lab. (LANL), Los Alamos, NM (United States), 1995.

[72] Ralph Menikoff, Determining curvature effect on detonation velocity from rate stick experiment, Impact Comput. Sci. Eng. 1 (2) (1989) 168–179.
[73] Wesley Chapman, Gabriel Montoya, Terry Salyer, Jeffrey Rhoads, Steven Son, Effects of embedded energetic sensors within an RDX-based explosive, Bull. Am. Phys. Soc. 67 (2022).
[74] L.G. Hill, T.D. Aslam, The LANL detonation-confinement test: prototype development and sample results, in: AIP Conference Proceedings, vol. 706, American Institute of Physics, 2004, pp. 847–850.
[75] Gabriel A. Montoya, Wesley W. Chapman, Joseph R. Lawrence, Terry R. Salyer, Steven F. Son, Effects of sub-mm cylindrical voids on detonation performance in PBX 9501, Propellants Explos. Pyrotech. 48 (5) (2023) e202200213.
[76] Gabriel Montoya, Joshua Dean, Joseph Lawrence, Terry Salyer, Steven Son, Evaluation of explosively-driven flash coatings, Bull. Am. Phys. Soc. 67 (2022).
[77] John Canny, A computational approach to edge detection, IEEE Trans. Pattern Anal. Mach. Intell. 6 (1986) 679–698.