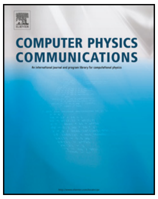
Computer Physics Communications 288 (2023) 108752

Contents lists available at ScienceDirect

Computer Physics Communications

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



A three-dimensional solver for simulating detonation on curvilinear adaptive meshes ☆

Han Peng *, Ralf Deiterding

School of Engineering, University of Southampton, Boldrewood Campus, Southampton, SO16 7QF, United Kingdom

ARTICLE INFO

Article history:
Received 30 October 2022
Received in revised form 5 March 2023
Accepted 11 April 2023
Available online 17 April 2023

Keywords:
Parallel adaptive mesh refinement
Mapped meshes
Reactive flow simulation
Detonation simulation

ABSTRACT

A generic solver in a parallel Cartesian adaptive mesh refinement framework is extended to simulate detonations on three-dimensional structured curvilinear meshes. A second-order accurate finite volume method is used with grid-aligned Riemann solvers for thermally perfect gas mixtures. Detailed, multi-step chemical kinetic mechanisms are employed and numerically incorporated with a splitting approach. The adaptive mesh refinement technique is applied to a mapped mesh using modified prolongation and restriction operators. The flux along the coarse-fine interface is considered in a correction procedure to ensure the conservation of the solver. The numerical accuracy, conservation and robustness of the simulations are verified and validated with suitable benchmark tests. The new solver is then used to simulate detonation problems in non-Cartesian geometries. A simulation is conducted of the three-dimensional detonation propagation in a 90-degree pipe bend. A detonation in a round tube is also simulated in a Galilean frame of reference. Both a rectangular mode and a spinning mode are observed in the simulations. In addition, the fundamental problem of detonation wave/boundary layer interaction is studied. The results show that the new solver can simulate high-speed reactive flows efficiently by the combined use of a curvilinear mapping with mesh adaptation.

1. Introduction

A detonation is a type of reactive phenomenon where a supersonic combustion wave drives a leading shock wave, resulting in the coupled propagation of both [1]. The thermodynamic states change sharply across a detonation, which is accompanied by fast energy release. A detonation is also considered as a pressure-gain and self-sustaining chemical reaction and is hence essentially different from a subsonic deflagration [2]. Theoretical calculations indicate that the Fickett-Jacobs cycle, that employs a pressure-gain detonation, can provide a higher thermal efficiency compared with the Brayton cycle, which is based on pressure-constant combustion [3]. Devices powered by detonations [4–7] have been proposed as promising replacements for the current propulsion and power generation systems that employ pressure-constant combustion.

A detonation travels at high temperature, high pressure and high speed. As a result, it is challenging to measure the detailed physical quantities in the flow field of a detonation by experimental methods [5,8]. CFD (Computational Fluid Dynamics) is a feasible way to predict results especially for reactive flows under extreme thermal conditions [8–10]. Numerical simulations therefore play an increasingly important role in detonation studies. Yet, an accurate and predictive detonation simulation is difficult in itself as a detonation is a supersonic combustion phenomenon that inherently involves a discontinuous shock wave [11]. Adequate mesh resolution is essential to resolve the chemical characteristic length [12,13], but large-scale numerical simulations of high-speed reacting flows are normally prohibitively expensive.

An effective approach for capturing supersonic combustion and discontinuous flow structures is adaptive mesh refinement (AMR) [14]. The AMR technique employs a strategy in which the grid is dynamically refined at specific places where the physical states change dramatically. It provides a balance between the numerical accuracy and computational costs. AMROC (Blockstructured Adaptive Mesh Refinement in object-orientated C++) [15] is a parallel open-source framework implemented on a Cartesian mesh. The Clawpack package [16] has been incorporated into AMROC with an extended detailed chemical-kinetic model for multi-species thermally perfect gas mixtures. The generic solvers have

---

☆ The review of this paper was arranged by Prof. David W. Walker.

* Corresponding author.

E-mail addresses: h.peng@soton.ac.uk (H. Peng), r.deiterding@soton.ac.uk (R. Deiterding).

https://doi.org/10.1016/j.cpc.2023.108752

been widely used to study supersonic combustion and detonation problems [17–20].

In AMROC, the ghost-fluid method [21] has been used previously to deal with non-Cartesian boundaries. States in ghost cells are set before the original numerical update to model the boundary conditions [22]. The actual boundary is considered by employing a scalar level set function [23,24]. Another strategy is the application of a quadrilateral but non-Cartesian mesh by adopting the mapping method with a capacity function as proposed in Ref. [16]. Using a body-fitted mesh can also avoid extremely small time steps, which may occur in some small cut-cells if no additional measures are taken. In this work, a previously developed two-dimensional AMROC inviscid solver [25] is extended to three spatial dimensions for viscous multi-species reactive problems based on curvilinear adaptive meshes.

A second-order finite volume scheme with shock-capturing capability and fluxes reconstruction is implemented for simulations of detonation with an arbitrary number of species and arbitrary chemical reaction systems. Benchmark test cases of 2-D and 3-D configurations are conducted to verify the performance of the proposed solver within the mesh adaptation framework. Then, the solver is applied to simulate various cases involving detonation waves. Thanks to the present solver's capabilities, such test cases show the robustness and efficiency of the implemented numerical methods in handling multi-scale reactive flows in a non-Cartesian geometry.

This paper is organised as follows: In Section 2, numerical methods and implementation details of the framework of AMROC–Clawpack are introduced. Section 3 first quantifies the numerical errors with the method of manufactured solutions and verifies the flux correction method by a conservation test. Then, results are presented for shock tube and shock bubble combustion problems by solving multi-component Navier-Stokes equations on mapped grids. In Section 4, the present solver is used to simulate the hydrogen-oxygen-argon detonation propagation in a pipe bend and in a round tube. The cellular structure of the respective 3-D detonation is analysed. The interaction between a detonation and boundary layers is also studied. Finally, concluding remarks and future work are discussed in Section 5.

2. Methodology

2.1. Governing equations

The multi-species Navier-Stokes equations with a detailed chemical model in three-dimensional form are solved as governing equations. They read

$$\begin{aligned}
&\partial_t(\rho \boldsymbol{Y}) + \nabla\cdot(\rho \boldsymbol{u}\boldsymbol{Y}) - \nabla\cdot\boldsymbol{j} = \dot{\boldsymbol{\omega}},\\
&\partial_t(\rho \boldsymbol{u}) + \nabla\cdot(\rho \boldsymbol{u}\boldsymbol{u} + p\boldsymbol{I}) - \nabla\cdot\boldsymbol{\tau} = 0,\\
&\partial_t(\rho E) + \nabla\cdot(\rho \boldsymbol{u}(E + p/\rho)) - \nabla\cdot(\boldsymbol{q} + \boldsymbol{u}\cdot\boldsymbol{\tau}) = 0,
\end{aligned} \tag{1}$$

where $\boldsymbol{Y}$ is the vector of species mass fractions. The total density is computed from the species conservation: $\rho = \sum_{i=1}^{N_{sp}} \rho Y_i$, and N_{sp} is the total number of species. $\boldsymbol{u} = [u, v, w]^{\mathrm{T}}$ is the velocity vector, and $\dot{\boldsymbol{\omega}}$ is the vector of species production rates. p is the pressure, $\boldsymbol{I}$ is the 3×3 identity matrix, and E is the total specific energy. The multi-species ideal gas state equation, $p = \sum_{i=1}^{N_{sp}} \rho_i T R_{\mathrm{u}}/W_i$, is used to close system (1).

As shown in (2), the species diffusion fluxes $\boldsymbol{j}$ are computed by Fick's law. $\boldsymbol{X}$ is the vector of species mole fractions, and $\boldsymbol{D}$ is the vector of the mixture-averaged diffusion coefficients. The viscous stress tensor $\boldsymbol{\tau}$ is calculated using Newton's law, and μ is the dynamic viscosity. The heat-flux vector is modelled using Fourier's law, and κ is the thermal conductivity.

$$\begin{aligned}
&\boldsymbol{j} = \rho \boldsymbol{Y}\boldsymbol{D}(\nabla\boldsymbol{X} + (\boldsymbol{X} - \boldsymbol{Y})\nabla p/p)/\boldsymbol{X},\\
&\boldsymbol{\tau} = -\frac{2}{3}\mu(\nabla\cdot\boldsymbol{u})\mathbf{I} + \mu\left[\nabla\boldsymbol{u} + (\nabla\boldsymbol{u})^T\right],\\
&\mathbf{q} = \kappa\nabla T + \sum_{i=1}^{N_{sp}} h_i j_i.
\end{aligned} \tag{2}$$

The total specific energy in (1) can be computed by

$$E = \sum_{i=1}^{N_{sp}} Y_i\Big(h^0_{\mathrm{ref}} + \int_{T_{\mathrm{ref}}}^{T} c_{\mathrm{p}_i}\, dT\Big) - \frac{p}{\rho} + \frac{1}{2}\boldsymbol{u}^2. \tag{3}$$

The constant pressure specific heat c_{p_i} is dependent on temperature and calculated by the polynomial functions in the CHEMKIN II library [26]. The species production rates are calculated by a chemical reaction mechanism of J steps as,

$$\begin{aligned}
\dot{\omega}_i = \sum_{j=1}^{J}\left(\nu^r_{ji} - \nu^f_{ji}\right)\left[k^f_j \prod_{n=1}^{N_{\mathrm{sp}}}\left(\frac{\rho_n}{W_n}\right)^{\nu^f_{jn}} - k^r_j \prod_{n=1}^{N_{\mathrm{sp}}}\left(\frac{\rho_n}{W_n}\right)^{\nu^r_{jn}}\right],\\
i = 1, \ldots N_{\mathrm{sp}}.
\end{aligned} \tag{4}$$

The rate constant of forward and reverse chemical reactions is given by the Arrhenius formula:

$$k^{f/r}_j(T) = A^{f/r}_j T^{\beta^{f/r}_j} \exp\left(-\frac{E^{f/r}_j}{RT}\right). \tag{5}$$

The chemical kinetics are integrated by a semi-implicit generalised Runge–Kutta method of fourth order (GRK4A) [15]. A second-order accurate Strang splitting method is adopted for the stiff source term treatment.

2.2. Mapping method

A mapping method [16] for 2-D geometry transformations has recently been implemented into AMROC [25]. This method enables the numerical algorithms based on a Cartesian mesh to be applied to a body-fitted mesh, which is beneficial to modelling the near-wall region of non-Cartesian geometries. The coordinates of a uniform Cartesian mesh in computational space (x, y, z) are firstly mapped onto the coordinates of a non-uniform, curvilinear structured mesh (ξ, η, ζ) in physical space by a mapping function. The method is working in the physical space and derives the cell-centred finite volume method on the non-uniform mesh with a capacity function $|C_{ijk}|$ [16],

$$\begin{aligned}
Q^{n+1}_{ijk} = Q^n_{ijk} &- \frac{\Delta t}{|C_{ijk}|\,\Delta\xi}\left[\left(\tilde{F} - \tilde{F}_{\mathrm{v}}\right)^n_{i+\frac{1}{2},j,k}\cdot\frac{\tilde{A}_{i+\frac{1}{2},j,k}}{\Delta\eta\Delta\zeta} - \left(\tilde{F} - \tilde{F}_{\mathrm{v}}\right)^n_{i-\frac{1}{2},j,k}\cdot\frac{\tilde{A}_{i-\frac{1}{2},j,k}}{\Delta\eta\Delta\zeta}\right]\\
&- \frac{\Delta t}{|C_{ijk}|\,\Delta\eta}\left[\left(\tilde{G} - \tilde{G}_{\mathrm{v}}\right)^n_{i,j+\frac{1}{2},k}\cdot\frac{\tilde{A}_{i,j+\frac{1}{2},k}}{\Delta\xi\Delta\zeta} - \left(\tilde{G} - \tilde{G}_{\mathrm{v}}\right)^n_{i,j-\frac{1}{2},k}\cdot\frac{\tilde{A}_{i,j-\frac{1}{2},k}}{\Delta\xi\Delta\zeta}\right]\\
&- \frac{\Delta t}{|C_{ijk}|\,\Delta\zeta}\left[\left(\tilde{H} - \tilde{H}_{\mathrm{v}}\right)^n_{i,j,k+\frac{1}{2}}\cdot\frac{\tilde{A}_{i,j,k+\frac{1}{2}}}{\Delta\xi\Delta\eta} - \left(\tilde{H} - \tilde{H}_{\mathrm{v}}\right)^n_{i,j,k-\frac{1}{2}}\cdot\frac{\tilde{A}_{i,j,k-\frac{1}{2}}}{\Delta\xi\Delta\eta}\right]\\
&+ \Delta t\cdot S.
\end{aligned} \tag{6}$$

For the 3-D solver, the capacity function $\left|C_{ijk}\right|$ represents the volume ratio of the physical cell to the computational cell. The inviscid fluxes $[\tilde{F}, \tilde{G}, \tilde{H}]^{\mathrm{T}}$ and the viscous fluxes $[\tilde{F}_{\mathrm{v}}, \tilde{G}_{\mathrm{v}}, \tilde{H}_{\mathrm{v}}]^{\mathrm{T}}$ are calculated separately and the details are introduced in the following subsections. The coefficient terms behind the numerical fluxes are the ratio of the area of the respective facet. The dimensional splitting method [27] is used to numerically handle multi-dimensional problems.

2.3. Fluxes evaluations

2.3.1. Inviscid fluxes

A hybrid Roe/HLL scheme [15] is used in this work to calculate the inviscid flux. This scheme incorporates the Roe method with multi-dimensional entropy correction, mass fraction positivity correction and the HLL (Harten-Lax-Van Leer) method. It is a robust and reliable scheme when simulating detonation problems. Further on, the standard HLLC method [27] has also been implemented with pressure–based wave speed estimates to solve the inviscid flux on the interface. The pressure in the star region is estimated by the primitive variable Riemann solver (PVRS).

The detailed steps of the multi-component Roe/HLL scheme and the HLLC scheme on the mapped structured mesh are given in Appendix A. Facet-dependent rotation matrices T_{s} are used to rotate the velocities aligning with the grid. The rotation matrices used for the 3-D multi-component equations are given in Appendix A. After computing the fluxes by the standard Riemann solver, the inverse matrix T_{s}^{-1} is employed to rotate the flux back to the Cartesian coordinate system in physical space. The overall mapped flux computation reads

$$\tilde{F}^{n}_{i+\frac{1}{2},j,k} = T_{\mathrm{s}}^{-1} F\left(T_{\mathrm{s}} Q^{n,\mathrm{l}}_{i+\frac{1}{2},j,k}, T_{\mathrm{s}} Q^{n,\mathrm{r}}_{i+\frac{1}{2},j,k}\right). \tag{7}$$

2.3.2. Viscous fluxes

The viscous fluxes on mapped structured meshes are calculated at each face in physical space through the coordinate transformation. For instance, the derivative of a variable in physical space is calculated through the chain rule:

$$\frac{\partial \varphi}{\partial x} = \left(\frac{\partial \varphi}{\partial \xi}\right)\left(\frac{\partial \xi}{\partial x}\right) + \left(\frac{\partial \varphi}{\partial \eta}\right)\left(\frac{\partial \eta}{\partial x}\right) + \left(\frac{\partial \varphi}{\partial \zeta}\right)\left(\frac{\partial \zeta}{\partial x}\right). \tag{8}$$

The derivatives of the variables φ with respect to the computational coordinates (ξ, η, ζ) are calculated by a second-order central difference. The derivatives of the computational coordinates with respect to the physical coordinates (x, y, z) are calculated either from the analytic mapping functions or using second-order accurate finite difference numerical derivatives.

2.3.3. CFL conditions

In addition, both the effects of inviscid fluxes and viscous fluxes should be considered when calculating the CFL condition given in Eq. (9). The global CFL number is defined by the maximum value in the three spatial dimensions and evaluated for system (1) as

$$\begin{aligned} CFL = \max\Bigg[& (|u| + a)\,\frac{\Delta t}{\Delta x} \\ & + \max\left(\frac{8\mu}{3\rho}, \frac{2k}{C_{\mathrm{v}}\rho}, D_1, \ldots, D_{N\mathrm{sp}}\right)\frac{\Delta t}{\Delta x^2}\Bigg]. \end{aligned} \tag{9}$$

The next global time step is calculated from the CFL number of the previous time step and a user-specified target CFL number. Since the method is strictly explicit, a target CFL number less than 1 is used in all the cases. Decreasing the CFL number can improve the stability of a computation, but it is more expensive.

2.4. Reconstruction

A second-order accurate MUSCL-Hancock scheme with limiter is used for the inviscid flux reconstruction. The MUSCL part reconstructs primitive variables or conservative variables in space and the Hancock part is used to update the reconstructed variables by half a time step. The variables Q in the physical space are transformed to computational space $\mathbf{Q} = JQ$ by using the Jacobian determinant of the grid $J = \det|\partial(x, y, z)/\partial(\xi, \eta, \zeta)|$. The standard MUSCL reconstruction is conducted in the computational space as

$$\begin{aligned} \mathbf{Q}^{l,n}_{i,j,k} &= \mathbf{Q}^{n}_{i,j,k} - \frac{1}{4}\varepsilon_i(1+\omega)\left(\mathbf{Q}_{i,j,k} - \mathbf{Q}_{i-1,j,k}\right) \\ &\quad - \frac{1}{4}\varepsilon_i(1-\omega)\left(\mathbf{Q}_{i+1,j,k} - \mathbf{Q}_{i,j,k}\right), \\ \mathbf{Q}^{r,n}_{i,j,k} &= \mathbf{Q}^{n}_{i,j,k} + \frac{1}{4}\varepsilon_i(1-\omega)\left(\mathbf{Q}_{i,j,k} - \mathbf{Q}_{i-1,j,k}\right) \\ &\quad + \frac{1}{4}\varepsilon_i(1+\omega)\left(\mathbf{Q}_{i+1,j,k} - \mathbf{Q}_{i,j,k}\right). \end{aligned} \tag{10}$$

In the latter, ε_i denotes the limiter and the Minmod limiter has been used for all the cases in the present work. The value of ω is set to 0 as a linear reconstruction. After the reconstruction, the variables are transformed back to the physical space and are integrated by half a time step before being used for the flux estimation in a Riemann solver.

2.5. Adaptive mesh refinement

AMROC adopts the block-structured adaptive mesh refinement technique (SAMR) after Berger and Colella [14]. This method employs a patch-wise refinement approach. The meshes are dynamically flagged by using specified refinement criteria. These flagged cells are then grouped into a region of various-sized rectangular blocks. By successively creating refined meshes from respectively coarser parents, a multi-level hierarchy of embedded grid patches is created.

Using the Berger-Colella AMR method, the main finite volume integrator is effectively separated from the adaptive strategy. The numerical schemes are employed simultaneously on grids at different levels. The curvilinear structured mesh can be applied to the entire AMR technique since the AMROC structured data layout is preserved in the present solver. For instance, to generate the coordinates of grids at a higher level $(l+1)$ or a lower level $(l-1)$, the coordinates of the flagged uniform Cartesian mesh in computational space at level l are refined or coarsened, respectively. By using the same mapping strategy (see Appendix B) on each level, the new coordinates in computational space $(\xi_{l\pm1}, \eta_{l\pm1}, \zeta_{l\pm1})$ are used to generate curvilinear structured grids in physical space $(x_{l\pm1}, y_{l\pm1}, z_{l\pm1})$ at the new level.

A number of general parameters are prescribed in the AMR algorithm. Unless otherwise mentioned, the following adaptive refinement settings are used in all of the computations in this work. The refinement flags are always set, and the grid hierarchy is recomposed at each coarser level time step. The buffer width is set to 2 in order to mark two more cells in all directions around each flagged cell. The clustering algorithm's threshold is set to 0.7, which indicates that the programme generates successively smaller grids until the ratio between flagged and all cells in each new grid exceeds this given threshold. Scaled gradients [25] are used as refinement criteria in the present solver. The differences of adjacent cells are computed in terms of scalar quantities derived from the vector of state. A cell is flagged for further refinement when one of its scaled gradients is beyond the given threshold in any directions.

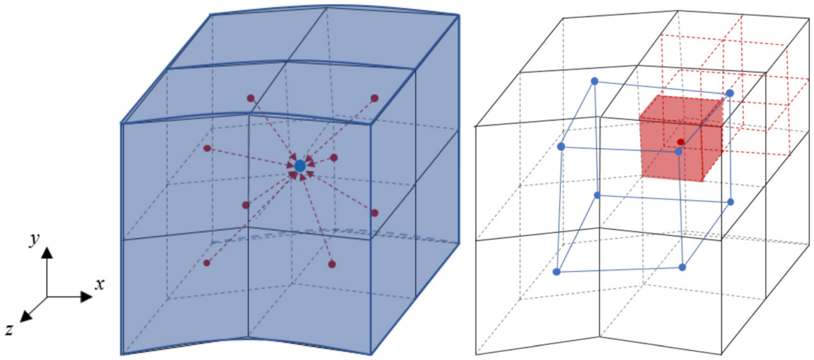


Fig. 1. Schematic of the restriction (left) and prolongation (right) operation.

2.5.1. Prolongation and restriction

Transferring values from fine to coarse cells is called restriction in AMR algorithms. Fig. 1 is an example of the restriction operation between grids on adjacent hierarchies with a refinement factor of 2. The restriction strategy is such that the value of the coarse element (displayed by the blue dot) is determined by its n^3 fine sub-elements (displayed by the red dots) through

$$Q_{\mathrm{c}} = \sum_{k=1}^{n^3} V_{\mathrm{f},k} Q_{\mathrm{f},k} / V_{\mathrm{c}}, \tag{11}$$

where subscript c denotes the coarse cell and subscript f denotes the fine cell. The n is the refinement factor between these two levels.

To illustrate the prolongation operation, the locations of the centroid of each cell are extracted in Fig. 1. A new element is generated by connecting the centroids of each coarse cell. The value in the refined cell can be computed by its closest n^3 coarse elements,

$$\begin{aligned} Q_{\mathrm{f}} = {} & (1-f_3)\{(1-f_2)[(1-f_1)Q_{\mathrm{c}1} + f_1 Q_{\mathrm{c}2}] \\ & + f_2[(1-f_1)Q_{\mathrm{c}3} + f_1 Q_{\mathrm{c}4}]\} \\ & + f_3\{(1-f_2)[(1-f_1)Q_{\mathrm{c}5} + f_1 Q_{\mathrm{c}6}] \\ & + f_2[(1-f_1)Q_{\mathrm{c}7} + f_1 Q_{\mathrm{c}8}]\}. \end{aligned} \tag{12}$$

A trilinear interpolation is applied for the coarse-to-fine grid prolongation, which involves a local nonlinear system solved by a Newton-Raphson method [28]. If this method is not convergent within a given number of iteration steps, the gradient descent algorithm would be used.

2.5.2. Fluxes correction

It is well known that the flux across coarse-fine interfaces in the Berger-Colella algorithm is not automatically strictly conservative. Hence, a flux correction or namely a flux fix-up is used before updating the states. In the updated formula of all cells, the coarse flux is approximated with all modified neighbouring cells by the sum of all overlying fine-level fluxes. For simplicity, only the 2-D fix-up process is described here, as depicted in Fig. 2, and the 3-D method can be deduced naturally by canonical extension. As an example on Cartesian grids [15], the correct update for Q in cell (j,k) on level i is given as

$$\begin{aligned} & \delta F^{d,i+1}_{j-\frac{1}{2},k} = -F^{d,i}_{j-\frac{1}{2},k}, \\ & \delta F^{d,i+1}_{j-\frac{1}{2},k} = \delta F^{d,i+1}_{j-\frac{1}{2},k} + \frac{1}{{r^{i+1}}^2} \sum_{l=0}^{r^{i+1}-1} \sum_{n=0}^{r^{i+1}-1} F^{d,i+1}_{v+\frac{1}{2},w+l}\left(t + n\Delta t^{i+1}\right), \\ & \check{Q}^{\,i}_{jk}\left(t+\Delta t^i\right) = Q^{\,i}_{jk}\left(t+\Delta t^i\right) + \frac{\Delta t^i}{\Delta x^i}\delta F^{d,i+1}_{j-\frac{1}{2},k}. \end{aligned} \tag{13}$$

As shown in Fig. 2, the shaded cells at the coarse level are modified with the correction term δF on the interface. An analogous correction method is used to update Q on a mapped mesh. The flux difference on a mapped grid is computed as

$$\begin{aligned} \delta F^{d,i+1}_{j-1/2,k} = {} & -F^{d,i}_{j-1/2,k}\frac{\Delta y^i}{\Delta \eta^i}, \\ \delta F^{d,i+1}_{j-1/2,k} = {} & \delta F^{d,i+1}_{j-1/2,k} \\ & + \frac{1}{{r^{i+1}}^2} \sum_{l=0}^{r^{i+1}-1} \sum_{n=0}^{r^{i+1}-1} F^{d,i+1}_{v+1/2,w+l}\left(t + n\Delta t^{i+1}\right)\frac{\Delta y^{i+1}}{\Delta \eta^{i+1}}. \end{aligned} \tag{14}$$

If the refinement factor r^{i+1} is set to 2 between level i and level $i+1$, this term $\delta F^{d,i+1}_{j-1/2,k}$ can be expanded as

$$\begin{aligned} \delta F^{d,i+1}_{j-1/2,k} = {} & -F^{d,i}_{j-1/2,k}\frac{\Delta y^i}{\Delta \eta^i} \\ & + \frac{1}{4}\Bigg[F^{d,i+1}_{v+1/2,w}(t)\frac{\Delta y^{i+1}_w}{\Delta \eta^{i+1}} + F^{d,i+1}_{v+1/2,w+1}(t)\frac{\Delta y^{i+1}_{w+1}}{\Delta \eta^{i+1}} \\ & + F^{d,i+1}_{v+1/2,w}\left(t+\Delta t^{i+1}\right)\frac{\Delta y^{i+1}_w}{\Delta \eta^{i+1}} \\ & + F^{d,i+1}_{v+1/2,w+1}\left(t+\Delta t^{i+1}\right)\frac{\Delta y^{i+1}_{w+1}}{\Delta \eta^{i+1}}\Bigg]. \end{aligned} \tag{15}$$

The difference on the coarse-fine interface is finally integrated into the state vectors by using the capacity function in the respective coarse grid cell by applying

$$\check{Q}^{\,i}_{jk}\left(t+\Delta t^i\right) = Q^{\,i}_{jk}\left(t+\Delta t^i\right) + \frac{\Delta t^i}{\left|C_{jk}\right|^i \Delta \xi^i}\delta F^{d,i+1}_{j-\frac{1}{2},k}. \tag{16}$$

The conservation of the entire scheme is hence maintained.

2.6. Boundary condition

In AMROC, the boundary conditions are specified by using ghost cells around a Cartesian mesh (see Ref. [15]). In the present solver, the state vectors are initialised in the physical space, where the velocities are orthonormal to the Cartesian coordinate system. As a result, the inflow and outflow boundary conditions are the same as those based on the Cartesian mesh. However, the symmetry and reflected wall boundary conditions need to be modified. For a symmetry or reflecting wall boundary condition, the normal velocity on the physical boundary should be zero and there is no

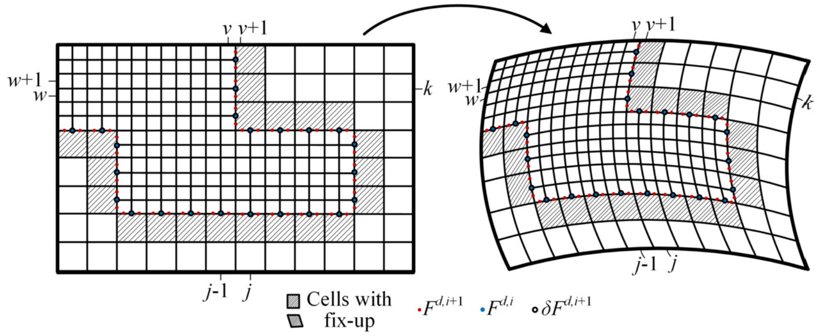


Fig. 2. Schematic of the flux correction on a Cartesian mesh (left) and on a mapped mesh (right).

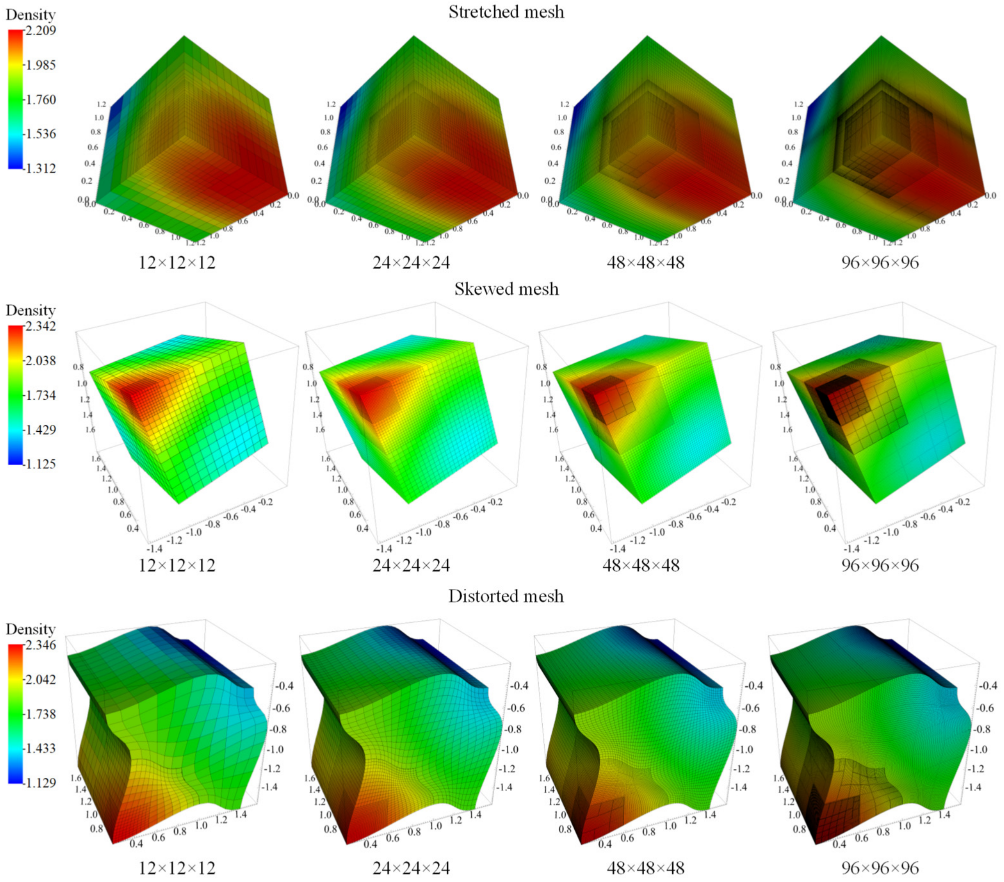


Fig. 3. The meshes used for 3-D MMS and corresponding density distribution at t = 0.01 s.

flow across the boundary. The transformation matrix is used to rotate the velocity of inner domain cells to align with the physical boundary. Then, the normal and tangential velocities on the boundary are eliminated by setting the value of the velocity in the ghost cell to be equal and opposite to the velocity in the domain cell. After that, the velocity in ghost cells is transformed back to the orthonormal direction of the Cartesian coordinate, respectively.

3. Numerical tests

3.1. Method of manufactured solutions

To verify the present solver, the method of manufactured solutions (MMS) [29,30] is first applied to different mapped meshes. The MMS is commonly used to verify CFD codes and test a solver's accuracy. In this case, the governing equations are constructed with manufactured solutions by replacing the chemical source terms with analytical source terms. Then, an error analysis is performed to compare the analytical solutions to the numerical results. The general form of the solutions is given as

$$\phi(x, y, z) = \phi_0 + \phi_x f_{s,x}\left(a_{\phi_x}\pi x\right) + \phi_y f_{s,y}\left(a_{\phi_y}\pi y\right) + \phi_z f_{s,z}\left(a_{\phi_z}\pi z\right). \tag{17}$$

ϕ indicates the initial states in terms of density, temperature, velocity and viscosity, etc. $f_{s,x}, f_{s,y}, f_{s,z}$ are trigonometric functions, and other parameters are constants (see Table C.1). A smoothly varying flow field is constructed by using the continuous functions.

Generally, in a supersonic case, the influence of the diffusion terms is relatively minor compared with the dominant convection terms. The MMS enables the use of a user-defined diffusive solution for the Navier-Stokes equations. In this case, the diffusive parameters are increased artificially $O(10^3) \sim O(10^5)$ times

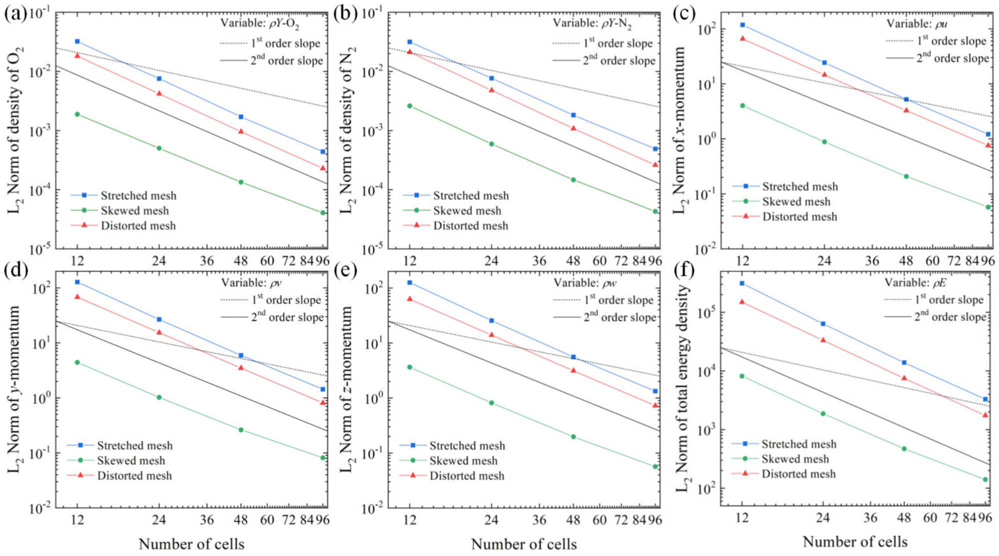


Fig. 4. L_2-norm of (a) density of oxygen, (b) density of nitrogen, (c) x-momentum, (d) y-momentum, (e) z-momentum, (f) total energy density when using the MMS on different mapped meshes.

to produce viscous fluxes of comparable magnitude to the inviscid fluxes. To simplify the derivation in terms of enthalpy in the analytical source terms, the specific heat ratio is assumed to be a constant.

The HLLC scheme is used, and the target CFL number in this case is set to 0.2. The whole domain is initialised with a mixture of oxygen and nitrogen. A Dirichlet boundary condition is used for the inflow boundary, and extrapolation is used for the supersonic outflow boundary. The computational domain is set as [0, 1] × [0, 1] × [0, 1] for the stretched mesh with a physical edge of 1.2 m. The domain is set as [-1.5 m, -0.3 m] × [0.3 m, 1.5 m] × [0.5 m, 1.7 m] for the skewed mesh and the distorted mesh in the computational space. A three-level refinement is used with a refinement factor of two for each level, as shown in Fig. 3.

Fig. 4 visualises the global L_2-norms in terms of partial density, momentum and energy density on various mapped meshes. This figure also depicts the order of accuracy, confirming that the current solver can achieve second-order accuracy when simulating a 3-D smoothly varying viscous flow field on a hierarchically refined mapped mesh.

3.2. Conservation test

A conservation test is carried out to verify the flux fix-up procedures. A uniform inviscid flow goes through a cube domain with an edge length of 0.2 m. The velocity is 200 m/s in each direction. A base grid of 100 × 100 × 100 cells is centrally distorted and supplemented with three levels of refinement. Periodic boundary conditions are used for all the boundaries. The target CFL number is 0.8 in this case, and it is run up to 0.1 ms, corresponding to 10 cycles. The total conservative error at time t is computed as

$$\mathrm{Err}(t) = \left| \frac{\sum_{i=1}^{N_{\max}} \left[Q_i(0) - Q_i(t) \right] V_i}{\sum_{i=1}^{N_{\max}} Q_i(0) V_i} \right|. \tag{18}$$

The volume V_i of the cell i is taken into account when computing the difference between the state $Q_i(t)$ at t and the initial state $Q_i(0)$. $N_{\max}$ is the total number of cells. As shown in Fig. 5, in the case without a fix-up on fluxes, the errors in terms of conservative variables start at 10^{-8} and reach 10^{-3} at the end of the calculation. An accumulated conservation error is observed when the fix-up is not used. In the case with a fix-up on fluxes, the conservation errors are negligible and are close to the machine error. This case tests the conservation of the inter-level operations, the flux fix-up method and confirms that the present solver is globally conservative on a multi-level mapped grid.

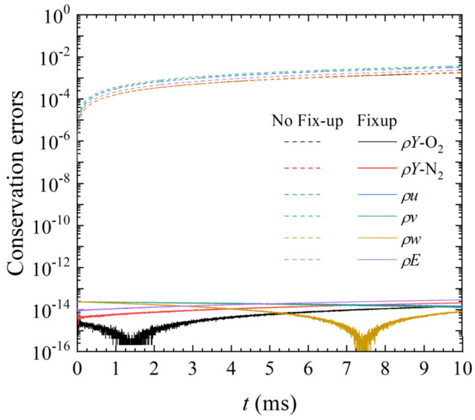


Fig. 5. Conservation errors in terms of conservative variables versus time.

3.3. Multi-component shocktube

This benchmark is a modified version of Sod's shock tube considering a multi-species mixture. It has been used to verify the robustness of numerical schemes in previous works [31,32]. In this case, a mixture of hydrogen–oxygen–argon at molar ratios 2:1:7 is initialised in a shock tube. The detailed Westbrook hydrogen mechanism [33] is used in this case, which contains 9 species and 34 elementary reactions. The size of the computation domain is 25 mm × 12.5 mm × 12.5 mm with slip wall boundary conditions on the side walls. For the adaptive cases, the maximal refinement level is set to three with a uniform refinement factor of 2 on each level. The respective minimum mesh size is 0.0625 mm in every direction. The refinement indicator threshold values for density and pressure are $\varepsilon_\rho = 0.002\,\mathrm{kg/m^3}$ and $\varepsilon_p = 4\,\mathrm{kPa}$, respectively. The CFL number in this case is set to 0.5. The adaptive computation uses approximately 2.6 M to 5.8 M cells in total instead of 16 M cells in the uniform case. The detailed initial conditions can be found in Ref. [32]. Fig. 6 shows how the grid is refined at the locations of the shock wave, contact discontinuity and rarefaction wave. The density profiles are extracted along the centre line in

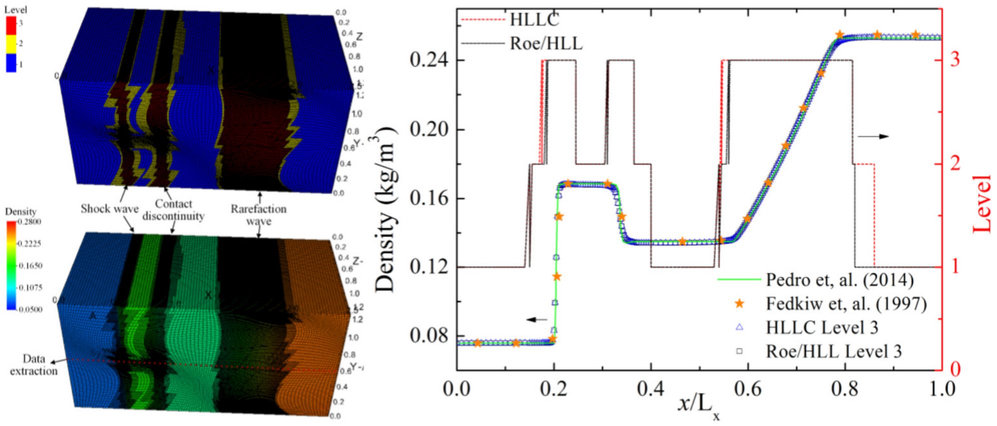


Fig. 6. Pseudo-colour image of the refinement levels and the density distribution.

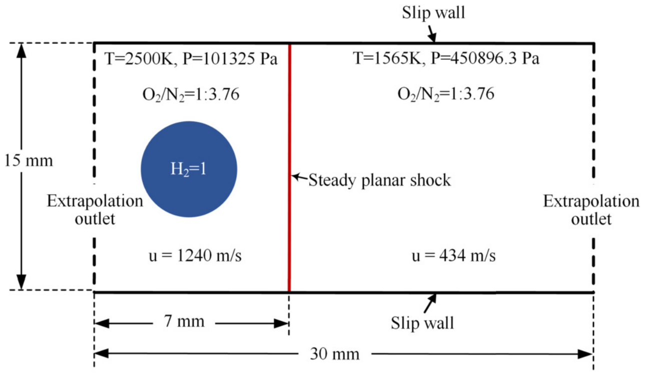


Fig. 7. Computational domain and numerical configuration of the shock bubble combustion problem.

the tube at 10 μs. The numerical results using AMROC are in good agreement with the results in Refs. [31,32].

3.4. Shock bubble combustion

The shock-driven combustion problem of a 2-D hydrogen bubble is tested to validate the viscous solver with chemical source terms. This validation case demonstrates the evolution of a hydrogen bubble and its interaction with a shock wave at Mach 2. The numerical configuration is shown in Fig. 7. A hydrogen sphere is initialised in the region on the left-hand side of the domain consisting of air. The temperature, pressure and velocity in different regions are also shown in the figure.

A distorted mesh is used in this case. The base mesh is set to 1024 × 512 cells with three refinement levels, and the refinement factor is 2 for each level. The refinement indicator thresholds are given in terms of the temperature, density and pressure $\varepsilon_T = 500\,\mathrm{K}$, $\varepsilon_\rho = 0.05\,\mathrm{kg/m^3}$ and $\varepsilon_p = 16\,\mathrm{kPa}$. The Jachimowski hydrogen/air mechanism [34] is used, which consists of 9 species and 19 reactions. The target CFL number in this case is set to 0.9.

Fig. 8 is the pseudo-colour image of the hydrogen mass fractions distribution (grey scale) superimposed on the pressure contours with black lines. At $t = 1.5$ μs, the hydrogen bubble collides with the stationary shock wave. The refracted, transmitted, reflected and incident pressure waves are all captured by the finest meshes as shown in the figure. At $t = 3.5$ μs, two reflected waves form on both sides of the bubble, and a secondary transmitted wave can be observed upstream of the bubble. Besides, a vortex is generated inside the hydrogen bubble. These features are in good agreement with the results in previous studies [35–37].

Fig. 9 shows a comparison of the pressure profile on the vertical central line between the present result and published numerical results. The present solver's numerical result, plotted with a purple dot line and an orange solid line respectively, agrees well with Ref [37] when using a similar distorted grid at the same mesh resolution. The strength of the right reflected wave is observed to decrease with a coarse mesh in our simulations, and there are differences between the results obtained by AMROC and literature solutions in terms of the strength and locations of the waves. The numerical methods used in the present solver and the literature references differ in terms of inviscid and viscous fluxes estimation. The discrepancies could also be caused by different approaches to modelling the fluid's transport properties.

4. Detonation simulations

4.1. Detonation in a smooth pipe bend

Simulations of detonation propagation in a smooth pipe bend are conducted to test the accuracy of the reactive solver on a mapped grid. For small radii and large bending angles, i.e., pipe bends of a 90-degree angle, the detonation wave structure is not maintained and triple point quenching can be observed at the outer side, while the detonation fails with originally regular cellular structure. The computational domain and initial conditions are set according to an experimental configuration [38]. In the present configuration, the channel width (4 cm) is at the length of 2.5 cells size and the channel height (1.6 cm) is at the length of 1 cell size. The computational domain and the numerical configuration are illustrated in Fig. 10.

The pipe is filled with a perfectly stirred stoichiometric hydrogen and oxygen mixture with 70% argon as dilution. The initial temperature is 298 K and the static pressure is 10 kPa. In order to speed up the calculation, the 2-D numerical results of detonation cells under the same conditions are initialised at the start of the pipe. Two unreacted sheets at 2086 K and 70 kPa are placed behind the detonation front as initial perturbations. The induction length L_{in} of the detonation is computed as 0.875 mm given the initial condition.

A four-level grid is used in this case and the refinement factor for each level is 2. The base mesh size is 0.25 mm (3.5 pts/L_{in}) in the η- and ζ-direction using 160 × 960 × 64 cells. In the ξ-direction, the base mesh size is around 0.242 mm along the central line, 0.363 mm on the outer wall, and 0.121 mm on the inner wall of the pipe bend. The mesh size at the finest level is 0.03025 mm (29 pts/L_{in}) on the central plane and changes from 0.015 mm (58 pts/L_{in}) to 0.045 mm (19.4 pts/L_{in}) in the bend. The refinement criteria for temperature, density and pressure are given as $\varepsilon_T = 500\,\mathrm{K}$,

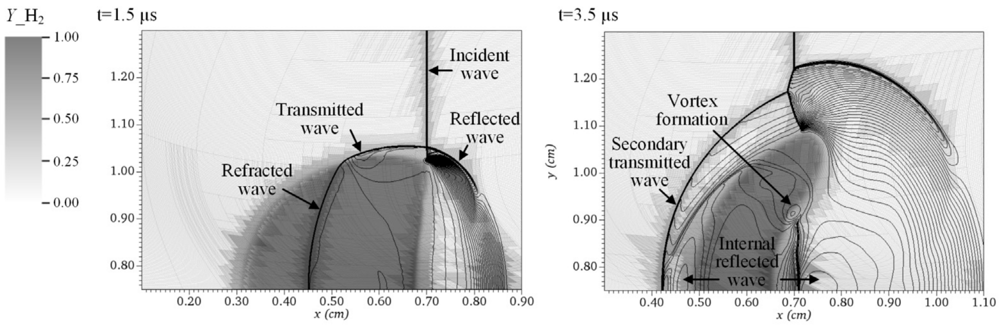


Fig. 8. Pseudo-colour image of the hydrogen mass fraction (grey scale) superposed on line contours of pressure (1–7.4 bar) at 1.5 µs and 3.5 µs.

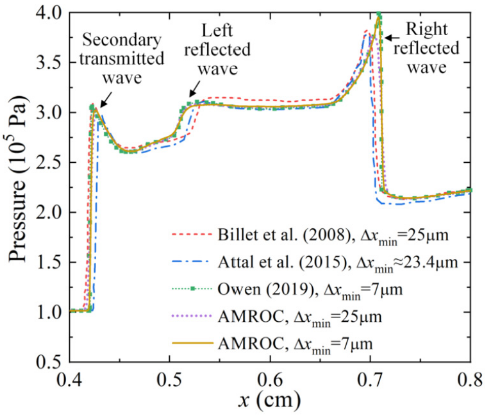


Fig. 9. Pressure profiles on the vertical central line compared with results in Ref. [35–37].

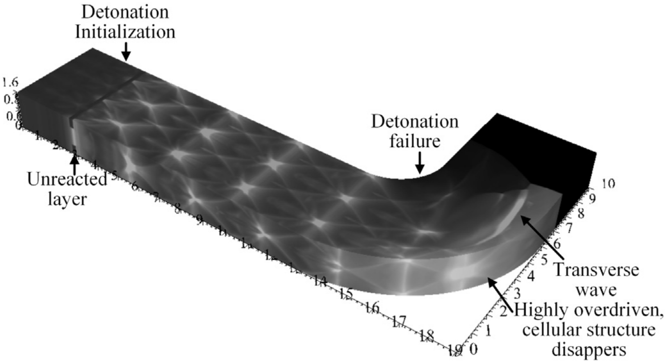


Fig. 11. Image of the numerical soot on the base grid.

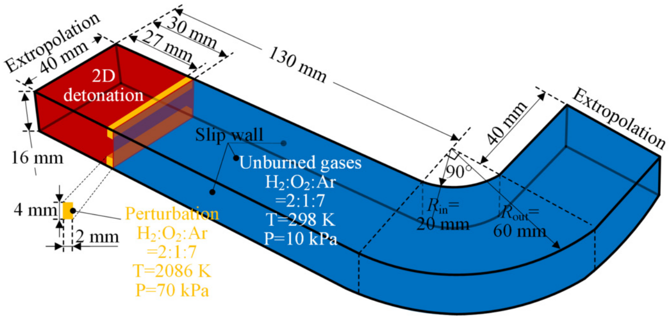


Fig. 10. Computational domain and numerical configuration of the detonation in a smooth pipe bend.

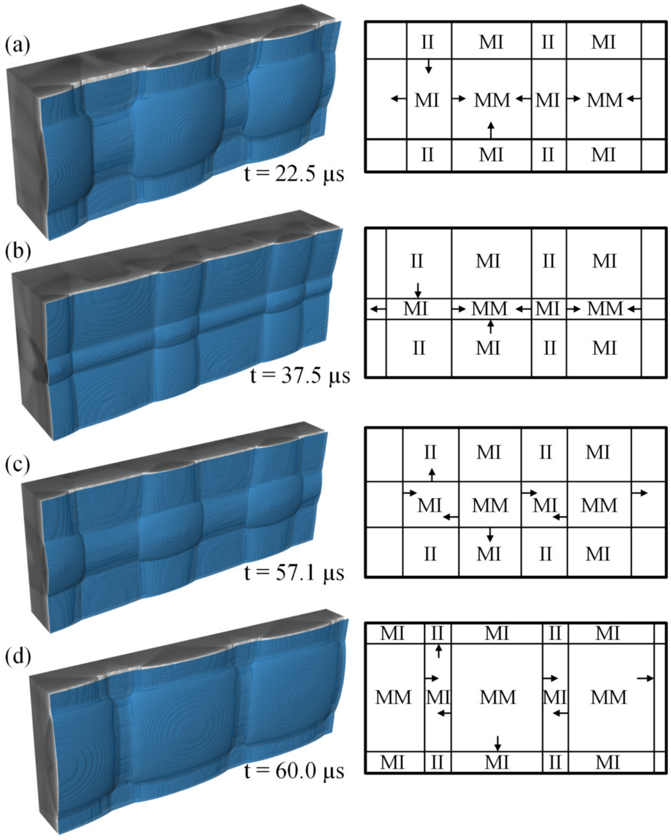


Fig. 12. Snapshots of the iso-volume of the OH mass fraction, overlapped by the pressure iso-surfaces (blue) at 20 kPa of 80% opacity (left) in the pipe and schematic front view of the periodic triple point line structure (right); Mach stem-Mach stem (MM), Mach stem–incident shock (MI), and incident shock–incident shock (II).

$\varepsilon_\rho = 0.05\,\text{kg/m}^3$ and $\varepsilon_p = 40\,\text{kPa}$, respectively. The adaptive computation uses approximately 86.1 M to 118.5 M instead of 5033.16 M cells in a uniform case. The calculations are performed on 480 cores (Intel Xeon E5-2670 2.0 GHz). Typical run times for a simulated time of 120 µs were approximately 6.5 days wall clock time.

Fig. 11 shows the accumulative results of the maximum pressure in the flow field on the base grid. The triple point is formed in the third direction after a short time and the 3-D cellular structure can be observed. When the detonation travels through the 90-degree bend, the cellular structure disappears near the inner wall, indicating that the detonation fails to be self-sustained. On the outer wall, the detonation is highly overdriven and transverse waves are generated after being reflected by the wall.

The detonation cellular front is visualised by the iso-volume of the OH mass fraction and the iso-surfaces of the pressure as shown in Fig. 12. The temporal development of the triple point line (TPL) structure is given in the schematic diagram. The front of the cellular structure can be divided into three different types of

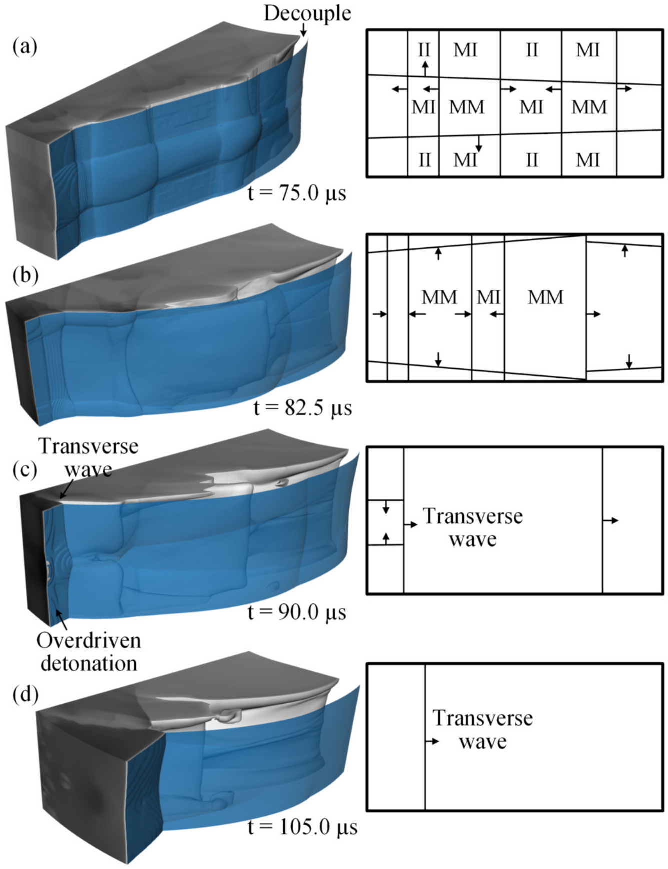


Fig. 13. Snapshots of the iso-volume of the OH mass fraction, overlapped by the pressure iso-surfaces (blue) at 20 kPa of 80% opacity (left) in the pipe bend and schematic front view of the periodic triple point line structure (right); Mach stem-Mach stem (MM), Mach stem–incident shock (MI), and incident shock–incident shock (II).

region [39]. The M-M region is constructed by both sets of Mach stems. The I-I region is constructed by both sets of incident shock waves and the M-I region is mixed, showing both types of waves. The cellular structure of the front was not recorded in the experiment [38], but the straight part of the pipe can be considered a rectangular channel. In Fig. 12, the TPLs on the detonation front in both dimensions do not collide with each other simultaneously, presenting a rectangular type out of phase [40–42].

Fig. 13 presents the decouple phenomena of the leading shock wave and the reaction front in the bend. The TPLs are not parallel to the wall because of the different local velocities at different radii, and the reaction front detaches from the shock near the inner wall. At 90 µs, the cellular structure finally disappears. Two formed transverse waves move along the radial direction. A large unburned and preheated region is observed at 105 µs. The detonation only is sustained near the outer wall in an over-driven state.

The slices on the bottom wall are extracted to make a comparison with the experimental results. As depicted in Fig. 14, the locations of the decoupled leading shock wave and the reaction front are in good agreement with the experimental schlieren images. The figures also show that the main features, including the leading shock waves and the flame front, are dynamically captured by the finest meshes.

4.2. Detonation in round tubes

In addition to the rectangular channel, the 3-D detonation structure is also simulated in a round tube to test the capability of the present solver. Experimental studies [43] and numerical simulations [44] both show that the cross-section shape and geometry confinement have effects on the detonation propagation mode and characteristics. Besides, the experimental observed spinning detonation [45–47] is a unique phenomenon in the round tube and it still has attracted researchers' attention in recent years [48,49].

However, there have been only a limited number of 3-D numerical studies on these problems over the past decades [44,50–53], because the combination of detailed chemistry models and high-resolution simulation is computationally quite expensive. The use of the adaptive mesh refinement technique can be particularly beneficial for this problem. In this section, the detonation is numerically studied in round tubes with two different diameters. These 3-D simulations are aimed at estimating the performance of the AMR technique for simulations of detonations in round tube problems.

To establish a cylindrical body-fitted grid without singularity in the centre, a convex combination mapping strategy [54] based on a single block hierarchy is used in the present case. A perfectly stirred stoichiometric hydrogen and oxygen mixture with 70% argon is filled into a round tube at 298 K and 10 kPa. The diameter of the tube D is 16 mm, corresponding to one detonation cell size λ. In order to decrease the computational cost, a nearly stationary reaction front under Galilean transformation is studied by setting the inflow velocity of the unburned gas as C-J velocity. The length of the tube is 30 mm and a 1-D ZND structure is initialised from 0 to 20 mm. An unreacted rectangular pocket is initially placed behind the reaction front at 2086 K and 70 kPa. The slip and adiabatic boundary condition is set for the wall. Extrapolation boundary conditions are employed for both the inlet and outlet.

A three-level grid is used in this case and the refinement factor for each level is 2. The base mesh uses 240 × 128 × 128 cells, resulting in a base mesh size of 0.125 mm (7 pts/L_{in}) in the ξ-direction, and a base mesh size ranging from 0.198 mm (4.4 pts/L_{in}) to 0.063 mm (13.9 pts/L_{in}) in both η-direction and ζ-direction. The corresponding mesh size at the finest level is 0.03125 mm (28 pts/L_{in}) in the ξ-direction, 0.0495 mm (17.6 pts/L_{in}) to 0.01575 mm (55.6 pts/L_{in}) in η- and ζ-direction. The refinement criteria for temperature, density and pressure are given as $\varepsilon_T = 500\,\text{K}$, $\varepsilon_\rho = 0.05\,\text{kg/m}^3$ and $\varepsilon_p = 40\,\text{kPa}$, respectively. The adaptive computation uses approximately 11.0 M to 38.2 M instead of 251.7 M cells in a uniform case. The calculations are performed on 240 cores (Intel Xeon E5-2670 2.0 GHz). Typical run times for a simulated time of 200 µs were approximately 4 days wall clock time.

Fig. 15 shows the slices of the temperature gradients in a detonation cell evolution period. At 123 µs, the triple points on the y- and z-planes are observed where the Mach stems, incident shock wave and transverse wave interact. The slip lines curl into vortices and embedded jets behind the incident shock [55]. The triple points on both planes all move towards the centre of the tube. At 127 µs, the triple points collide and new triple points are generated, which then move towards the wall. The jet behind the detonation front is detached from the leading shock. A new jet is formed after the collision in the centre. New triple points are reflected by the wall and finally travel towards the centre again at 137 µs, resulting in a similar structure compared with that at 123 µs. A whole period of the cellular structure evolution is completed and the duration is around 14 µs.

Fig. 16 displays the iso-surface in terms of the density at different times. The iso-surface is coloured by the temperature contours. Although the small structures in the wake of the shock are still not sharply resolved, the 3-D figure provides a good impression that the detonation propagates in the rectangular mode with a single cellular structure. The TPLs move to the tube centre and interact with the forward vortex ring. The vortex rings are detached and propagate downstream. A pair of new forward vortexes are formed

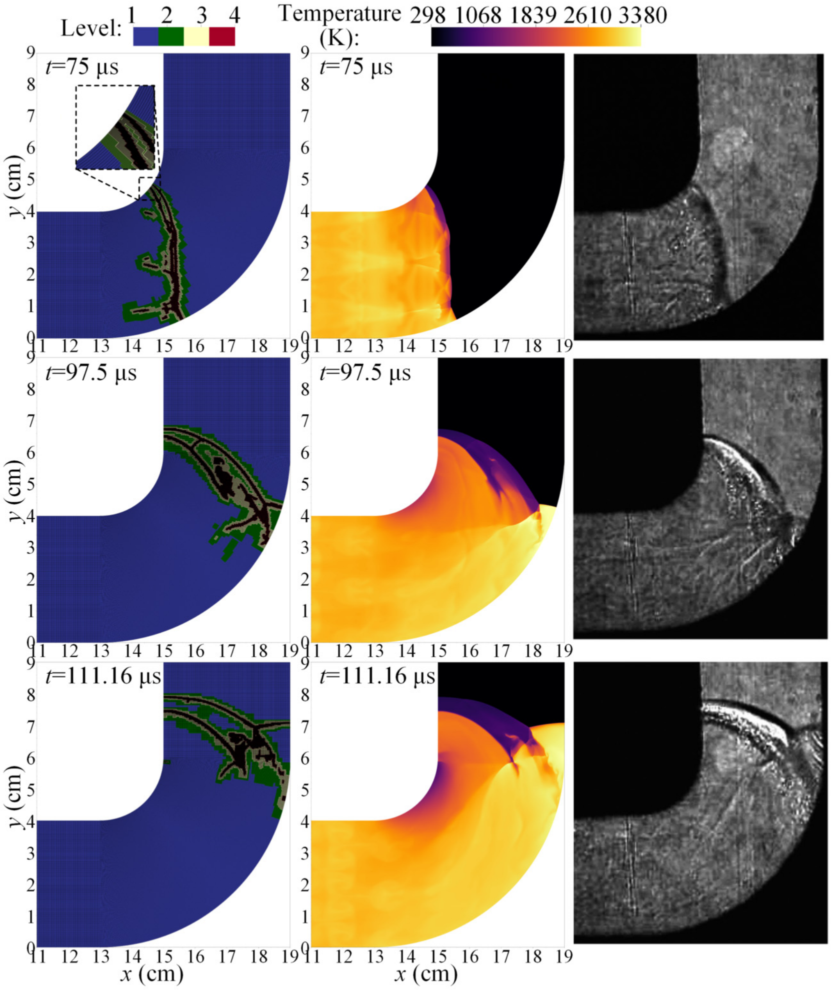


Fig. 14. Pseudo-colour image of refinement levels, temperature and experimental schlieren (from [38]).

after the collisions of TPLs, which causes the front to bulge. New TPLs are also generated and move to the wall, starting a new period of the cellular structure evolution.

Fig. 17 shows that the mesh around the detonation head and at the corners is adaptively refined to improve the accuracy of simulations. More processors are used in the refined regions with higher workloads. The hierarchical mesh is distributed to processors based on a space-filling curve [15] in computational space. Continuous redistribution, while the mesh is changing, is ensuring a balanced workload at run time.

In addition to the round tube with a diameter of one detonation cell, a half detonation cell size tube is also simulated with the same initial configurations. As displayed in Fig. 18, the detonation propagates in a rectangular mode at the early stage. Similar evolution of triple points and jets behind the incident shock is observed. At t = 120 μs, a single head detonation wave is generated at the front and the cellular pattern disappears. The detonation spins counter-clockwise direction from the right-hand side view. A typical spinning detonation is observed and propagates spirally to the outlet even in the Galilean frame. Compared with the case with D =0.5 λ, the transverse wave is constrained when the duct diameter is smaller than the detonation cell size. Only a single transverse wave remains and results in the spinning mode [1,13].

4.3. Detonation wave/boundary layer interaction

In the previously studied configurations, the unburned gases are considered stationary with respect to the reaction front, and slip boundary conditions are employed for the walls. However, with a high-speed inflow, the effects of bifurcated shock structures on shock-flame interactions become obvious [56]. In addition, the interaction of the boundary layer can impact the timing, location, and manner in which detonation occurs [57]. These factors are not negligible and are referred to as detonation wave/boundary layer interactions (DWBLI).

In order to test the capability of the present solver for simulating reactive boundary flow, a DWBLI case [58] is simulated using the solver on an adaptive stretched grid. The computational domain is set as a cuboid with a length of 85 mm, a height of 42 mm and a width of 16 mm. A high-speed hydrogen-oxygen-nitrogen mixture flows through a channel from the right-hand to the left-hand side. The interaction between the detonation and boundary layer is studied.

The initial temperature of the unburned $H_2/O_2/N_2$ mixture is 581 K and the pressure is set as 36.1 kPa with the molar ratio 0.56:1.0:2.9. The induction length L_{in} of the detonation is computed as 2.15 mm under this condition. The domain is initialised with a 1-D ZND solution from x = 0 mm to x = 50 mm. The right-hand boundary is given a Dirichlet boundary condition. The inflow velocity is set as 1532 m/s. The left-hand boundary is set as an extrapolated outflow boundary. The top and bottom boundaries are non-slip adiabatic walls from x = 0 mm to x = 80 mm, otherwise the boundaries are slip. The front and back boundaries are slip adiabatic walls.

A stretched grid is used as the base grid in this case, and the grid stretching ratio is set as 1.05 in the boundary layer. The ini-

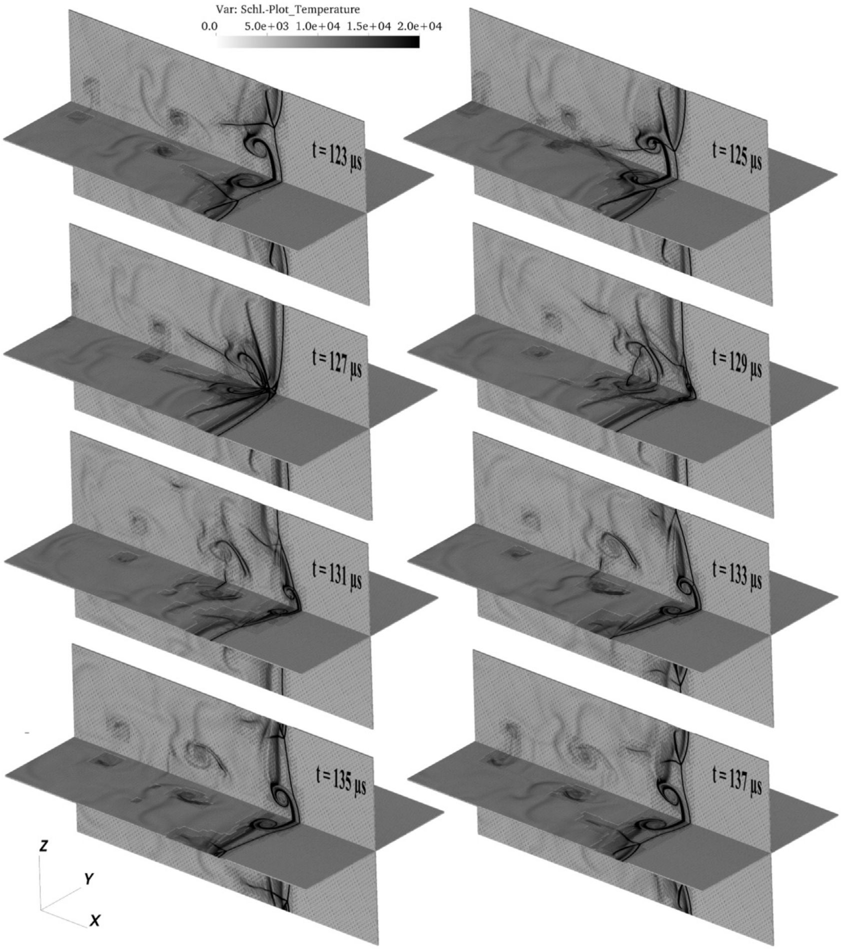


Fig. 15. Snapshots of the slices of the gradient of temperature, overlapped by the meshes at 80% opacity.

tial base mesh resolution is 340 × 400 × 64 cells. The height of the first layer mesh is 5.61 $\times 10^{-6}$ m and the viscous shear layer thickness is estimated as 5.59 $\times 10^{-5}$ m [58]. Hence, around 7 cells are distributed in the viscous scale of the boundary layer. The respective first 60 layers of cells from the top and bottom boundary are not adaptively refined in the computing process. The coarsest base grid in the channel centre is 0.28 mm. With a three-level refinement (2, 2), the coarsest grid in the domain is 0.07 mm (30.7 pts/L_{in}). The refinement criteria for temperature, density and pressure are given as $\varepsilon_T = 500\,\text{K}$, $\varepsilon_\rho = 0.03\,\text{kg/m}^3$ and $\varepsilon_p = 40\,\text{kPa}$, respectively. A finer resolution is used in the 2-D simulation based on a five-level Cartesian grid in Ref. [58], in which 5 cells are ensured in the viscous scale but a higher resolution is used for the detonation structure (137.8 pts/L_{in}).

It is noted that the viscous boundary layer and the viscous structure are still not fully resolved even under the present resolution. However, the grid resolution is chosen based on our available computing resources for the validation of the prototype solver. Using the stretched grid provides an approach to simulate the detonation problem involving viscous boundary layers efficiently compared to using a uniform grid. In this case, the adaptive computation uses approximately 10.8 M to 88.0 M instead of 557.1 M cells in a uniform case. The calculations are conducted on 240 cores (Intel Xeon E5-2670 2.0 GHz). Typical run times for a simulated time of 145 μs were approximately 12 days wall clock time.

Fig. 19 (a) shows the slices of the pseudo-colour image of the temperature gradient on the z =0 mm plane. The leading oblique shock, boundary layer flame and detonation front are observed in the numerical result. The vortex structures in the shear layer induced by the Kelvin–Helmholtz (KH) instability are also captured. Fig. 19 (b) is the mass fraction of OH superposed on pressure contours. The detonation front in the central flow is composed of a stand-off Mach stem and the subsequent reaction front. It is described as a Mach stem-induced detonation (MSID). The unburned pockets are also observed and further confirm the accuracy of the present solver when simulating this problem.

As shown in Fig. 19 (c) and (d), the base grid used in this case is stretched near the non-slip boundaries. The boundary flow is solved by the stretched grid itself, while the AMR is used in the domain interior except for the near-boundary region to improve the accuracy for solving the shock wave and detonation wave. These figures also show that the main features in the central flow are captured by the highest-level mesh.

Fig. 20 illustrates the slices of the pseudo-colour image of temperature at various times. At t = 0 μs, the 1-D ZND solution is initialised on a Galilean frame. At t = 50 μs, temperature recovery is observed near the non-slip part of the top and bottom wall. The oblique shock-induced combustion (OSIC) and boundary auto ignition result in a flame in the boundary layer. The Mach stem-induced detonation (MSID) front stands in the middle main stream. The slip lines behind the triple points develop into unstable shear layers downstream because of the KH instability. A pair of large-scale unburned pockets is formed between the MSID and the OSIC. Small unburned pockets are also observed in the boundary layer flame and are consumed in the following time sequences. The unstable combustion in the boundary layer has been considered as the main source for the oblique shock fluctuation.

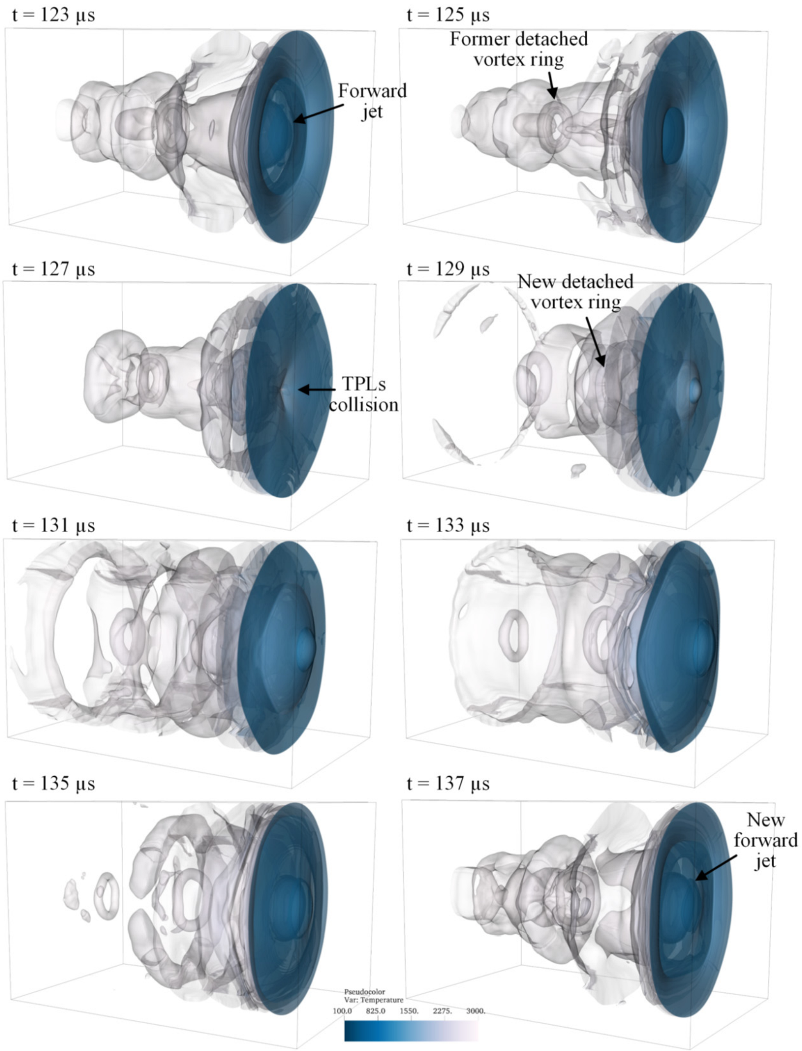


Fig. 16. Pseudo-colour image of the temperature based on the iso-surfaces of the density at 40% opacity, $D = \lambda$.

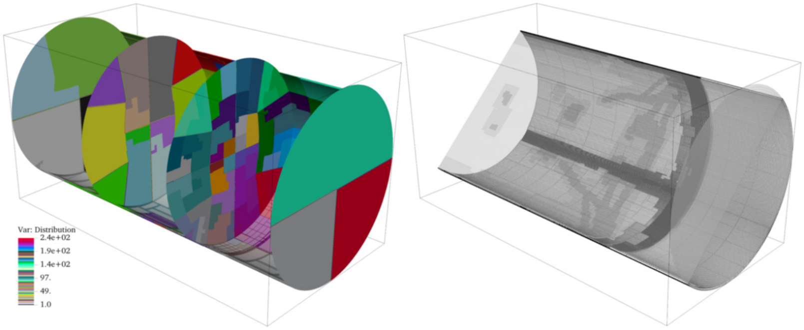


Fig. 17. Slices of processor distribution and slices of grids at 40% opacity.

The fluctuating oblique shock waves lead to an unsymmetrical structure and also enforce the triple points to move towards the channel centre. As a result, the height of the Mach stem eventually decreases and the Mach stem finally vanishes after the two oblique shock waves collide. Fig. 20 shows that the reaction front behind the Mach stem is detached from the intersection point and propagates downstream at t = 50 µs. The OSIC/MSID pattern turns into a pure OSIC pattern and only the OSIC is maintained in the channel.

Fig. 21 displays the typical OSIC/MSID pattern in a 3-D density iso-surface image coloured by the temperature. The small vortexes are resolved in the boundary layer. The present 3-D results reproduce some typical features, which are also observed in previous experiments and the 2-D simulations of Ref. [58]. The main difference between our present numerical result and the 2-D numerical result is that our 3-D simulation predicts a faster formation of the OSIC/MSID pattern. An OSIC/MSID mode is observed at t = 70 µs in the 3-D simulation, whereas a similar structure is formed only at t = 235 µs in the 2-D simulation. Besides, a mode change is observed in the 3-D results instead of dynamically maintaining it as in the 2-D results. A possible reason is that the 3-D simulation has more fluctuations in the z-direction. As shown in Fig. 20 and

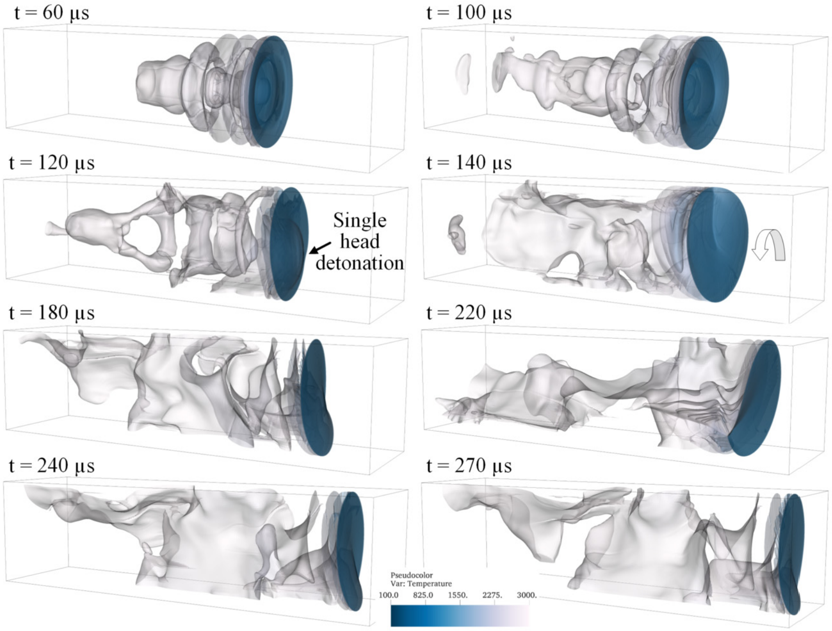


Fig. 18. Pseudo-colour image of the temperature based on the iso-surfaces of the density in 40% opacity, D =0.5 λ.

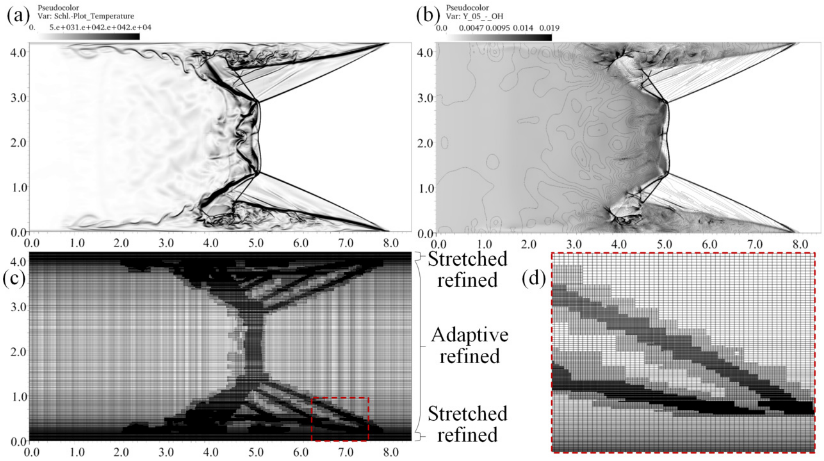


Fig. 19. Slices of pseudo-colour image of the temperature gradient, mass fraction of OH superposed on line contours of pressure and image of grids at t=60 μs.

Fig. 21, the non-uniform temperature distribution on the bottom wall also indicates the 3-D influences.

However, the real 3-D effects on this problem need to be studied by adopting the non-slip wall boundary conditions on both the front and back sides [56]. In addition, the temperature of the wall also plays an influential role in the OSIC. These factors are beyond the scope of this work and require further study.

5. Conclusion

In the present work, we extend our work on a parallel, adaptive high-speed combustion solver on mapped, structured meshes to three spatial dimensions, including the coordinate transformation, the grid-aligned flux scheme, MUSCL-Hancock reconstruction, fluxes corrections, restriction and prolongation operators for the adaptively refined mapped mesh. Some benchmark tests have been carried out to verify and validate the accuracy and robustness of the present solver. The numerical solutions obtained by the present solver are in good agreement with published numerical results.

The present solver is then used to simulate 3-D detonation propagation in a smooth pipe bend. This simulation reproduces typical features observed by published experimental results. The solver's capabilities are further demonstrated through typical cases, including detonation propagation in a smooth round tube and interaction with boundary layers in a channel. The dynamic adaption method reduces the total mesh scale from $O(10^8) \sim O(10^9)$ to $O(10^6) \sim O(10^7)$ in these 3-D cases without loss of main characteristics. The results confirm that the new 3-D solver is effective in adapting the parallel mesh at run time to rapidly propagating localised combustion fronts. In the future, a mapping strategy based on a multi-block hierarchy will be developed and the solver will be extended to higher order to improve the accuracy for further large-scale simulations of viscous multi-species reactive problems.

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

Data will be made available on request.

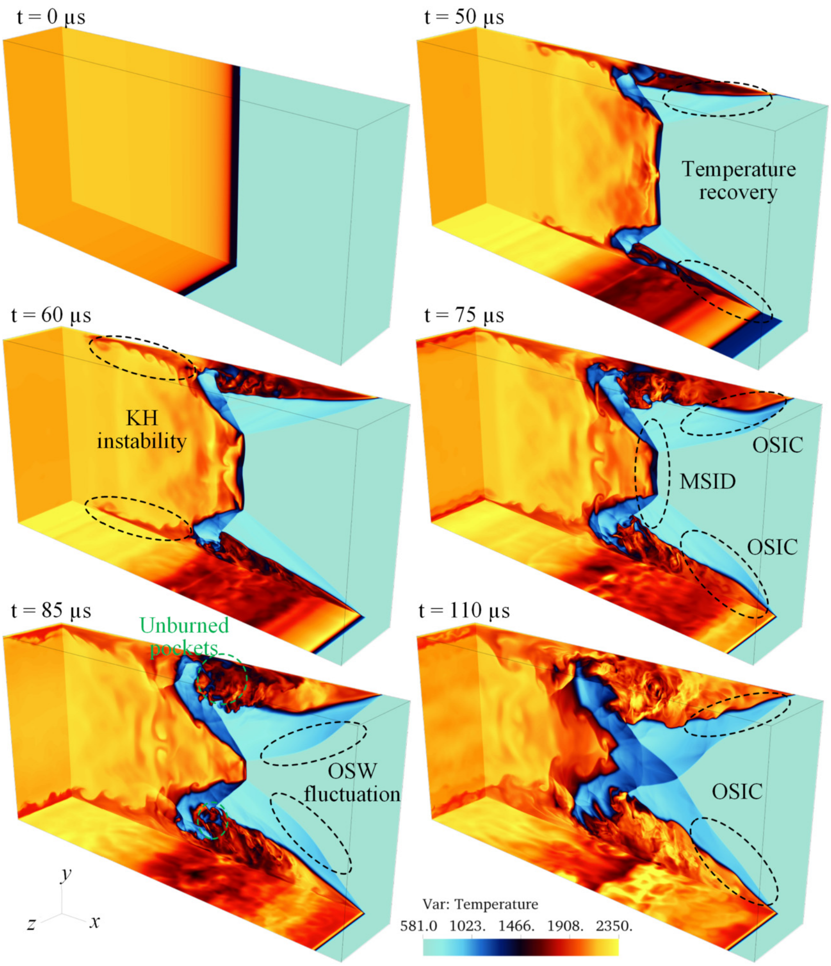


Fig. 20. Slices of pseudo-colour image of the temperature at different times.

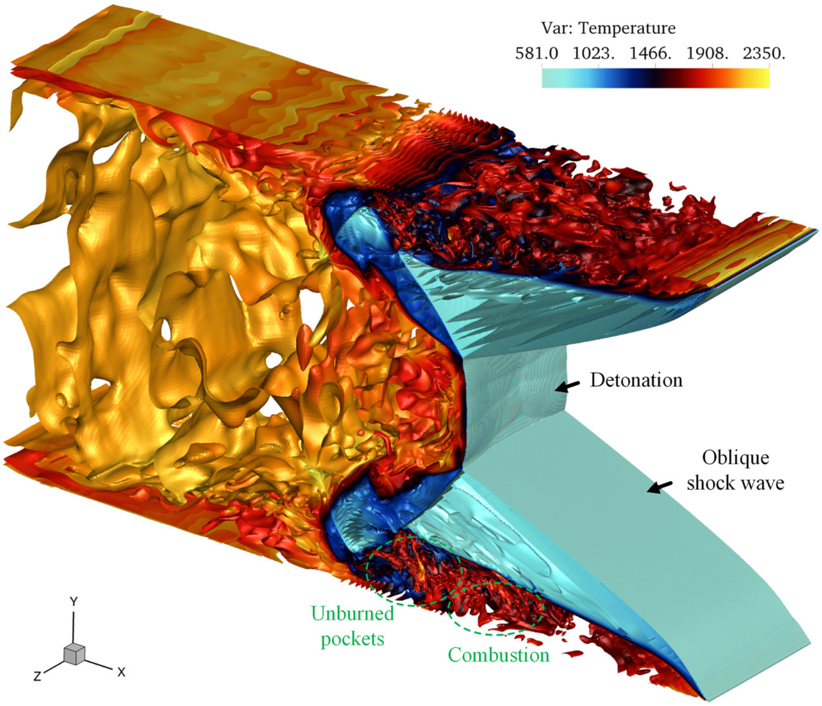


Fig. 21. Pseudo-colour image of the temperature based on the iso-surfaces of the density at t=70 μs for the OSIC/MSID structure.

Acknowledgement

The authors acknowledge the use of the IRIDIS High Performance Computing Facility, and associated support services at the University of Southampton. H. Peng also acknowledges financial support from the China Scholarship Council (CSC).

Appendix A. Riemann solver

The multi-component hybrid Roe/HLL scheme is used in this work to calculate the inviscid flux. The state vectors are rotated to the orthogonal direction of the interface. The flux on the interface can be computed by

$$F = \frac{1}{2}\Big[F_{\mathrm{L}} + F_{\mathrm{R}} - |\hat{A}|\,(Q_{\mathrm{R}} - Q_{\mathrm{L}})\Big]. \tag{A.1}$$

$\hat{A}$ is a constant matrix for the Riemann problem that approximates the original Jacobian. By projecting the data difference $(Q_{\mathrm{R}} - Q_{\mathrm{L}})$ onto the right eigenvectors of the diagonalizable matrix, the equation can be rewritten as

$$F = \frac{1}{2}\left(F_{\mathrm{L}} + F_{\mathrm{R}} - \sum_{m=1}^{N_{\mathrm{eq}}} \hat{\alpha}_m |\hat{\lambda}_m| r_m\right). \tag{A.2}$$

In order to compute the flux, one requires the solutions of the wave strengths $\hat{\alpha}_m$, the eigenvalues $\hat{\lambda}_m$ and the right eigenvectors r_m. N_{eq} is the number of equations. The wave strength $\hat{\alpha}_m$ for the multi-component equations is computed by

$$\begin{cases} \alpha_m = (\Delta p \mp \hat{\rho}\hat{a}\Delta u)/2\hat{a}^2, & m = 1 \text{ or } m = N_{\mathrm{sp}} + 4,\\ \alpha_m = \Delta\rho_i - \hat{Y}_i \Delta p/\hat{a}^2, & m = 2, \dots, N_{\mathrm{sp}} + 1,\\ \alpha_m = \hat{\rho}\Delta v, & m = N_{\mathrm{sp}} + 2,\\ \alpha_m = \hat{\rho}\Delta w, & m = N_{\mathrm{sp}} + 3. \end{cases} \tag{A.3}$$

The symbol Δ denotes the difference operator of the quantities and the superscript $\wedge$ denotes the Roe average of the quantities. The standard Roe averages are given as

$$\begin{aligned} &\hat{\rho} = \sqrt{\rho_L + \rho_{\mathrm{R}}},\\ &\hat{q} = \frac{\sqrt{\rho_{\mathrm{L}}} q_L + \sqrt{\rho_{\mathrm{R}}} q_R}{\sqrt{\rho_{\mathrm{L}}} + \sqrt{\rho_{\mathrm{R}}}}, \quad q = u, v, w, H, Y_i, T, \frac{1}{\bar{W}}. \end{aligned} \tag{A.4}$$

Use the average temperature to compute the specific heats for each species, then the averaged mixture specific heats at constant pressure and the averaged specific heat ratio can be evaluated as

$$\hat{c}_{\mathrm{p}} = \sum_{i=1}^{N_{\mathrm{sp}}} Y_i \hat{c}_{\mathrm{p}_i}, \quad \hat{\gamma} = \frac{\mathrm{R_u}}{\hat{\bar{W}}\hat{c}_{\mathrm{p}} - \mathrm{R_u}}. \tag{A.5}$$

The average speed of sound is computed by

$$\hat{a} = \sum_{i=1}^{N_{\mathrm{sp}}} \hat{Y}_i \hat{\phi}_i - (\hat{\gamma} - 1)\hat{U}^2 + (\hat{\gamma} - 1)\hat{H}, \tag{A.6}$$

where $\hat{\phi}_i$ is the average partial derivative of p with respect to Y_i, and $\hat{U}^2$ is the sum of the square of the average velocities,

$$\hat{\phi}_i = (\hat{\gamma} - 1)\left(\frac{\hat{U}^2}{2} - \hat{H}\right) + \hat{\gamma}\mathrm{R}_i\hat{T}, \quad \hat{U}^2 = \left(\hat{u}^2 + \hat{v}^2 + \hat{w}^2\right). \tag{A.7}$$

The matrix of right eigenvectors is given as

$$\begin{bmatrix} \hat{Y}_1 & 1 & 0 & \cdots & 0 & 0 & 0 & \hat{Y}_1\\ \vdots & 0 & \ddots & 0 & \vdots & \vdots & \vdots & \vdots\\ \vdots & \vdots & 0 & \ddots & 0 & \vdots & \vdots & \vdots\\ \hat{Y}_{N_{\mathrm{sp}}} & 0 & \cdots & 0 & 1 & 0 & 0 & \hat{Y}_{N_{\mathrm{sp}}}\\ \hat{u} - \hat{a} & \hat{u} & \cdots & \cdots & \hat{u} & 0 & 0 & \hat{u} + \hat{a}\\ \hat{v} & \hat{v} & \cdots & \cdots & \hat{v} & 1 & 0 & \hat{v}\\ \hat{w} & \hat{w} & \cdots & \cdots & \hat{w} & 0 & 1 & \hat{w}\\ \hat{H} - \hat{u}\hat{a} & \hat{U}^2 - \frac{\phi_1}{\hat{\gamma}} & \cdots & \cdots & \hat{U}^2 - \frac{\phi_{N_{\mathrm{sp}}}}{\hat{\gamma}} & \hat{v} & \hat{w} & \hat{H} + \hat{u}\hat{a} \end{bmatrix}. \tag{A.8}$$

The eigenvalues for the matrix $\hat{A}$ are given as

$$\begin{aligned} &\hat{\lambda}_1 = \hat{u} - \hat{a},\\ &\hat{\lambda}_m = \hat{u}, \quad m = 2, \dots, N_{\mathrm{sp}} + 3,\\ &\hat{\lambda}_m = \hat{u} + \hat{a}, \quad m = N_{\mathrm{sp}} + 4. \end{aligned} \tag{A.9}$$

Now, the quantities required for calculating the flux on the interface are all available. In addition, the density and pressure in the intermediate states of the linearized Riemann problem are evaluated to check their plausibility. If density and pressure are positive, the HLL method will not be used. A multi-dimensional entropy correction is employed to avoid the violation of the entropy condition. A mass fraction positivity correction is also used to avoid unphysical solutions.

If one of the quantities in terms of density and pressure is negative, the HLL method will be used to recompute the flux. The HLL method approximates the solution of the Riemann problem simply by two discontinuous waves. The flux on the interface is computed by

$$F = \begin{cases} F_{\mathrm{L}}, & s_{\mathrm{L}} > 0\\ \frac{s_{\mathrm{R}} F_{\mathrm{L}} - s_{\mathrm{L}} F_{\mathrm{R}} + s_{\mathrm{R}} s_{\mathrm{L}} (Q_{\mathrm{R}} - Q_{\mathrm{L}})}{s_{\mathrm{R}} - s_{\mathrm{L}}}, & s_{\mathrm{L}} \le 0 \le s_{\mathrm{R}}\\ F_{\mathrm{R}}, & s_{\mathrm{R}} < 0. \end{cases} \tag{A.10}$$

s_{L} and s_{R} denote approximations to the smallest and largest signal speed involved in the Riemann problem. The speeds are estimated by

$$\begin{aligned} &s_{\mathrm{L}} = \min\left(u_{\mathrm{L}} - a_{\mathrm{L}}, u_{\mathrm{R}} - a_{\mathrm{R}}\right),\\ &s_{\mathrm{R}} = \max\left(u_{\mathrm{L}} + a_{\mathrm{L}}, u_{\mathrm{R}} + a_{\mathrm{R}}\right). \end{aligned} \tag{A.11}$$

In addition to the hybrid Roe/HLL scheme, the HLLC scheme is also implemented in the present solver. Compared to the described HLL scheme, the HLLC scheme uses a three-wave model and resolves the contact discontinuity. Firstly, the pressure at the interface is evaluated as

$$p^* = \max\left(0, p_{\mathrm{pvrs}}\right), \quad p_{\mathrm{pvrs}} = \frac{1}{2}\left(p_{\mathrm{L}} + p_{\mathrm{R}}\right) - \frac{1}{2}\left(u_{\mathrm{R}} - u_{\mathrm{L}}\right)\bar{\rho}\bar{a}, \tag{A.12}$$

where

$$\bar{\rho} = \frac{1}{2}\left(\rho_{\mathrm{L}} + \rho_{\mathrm{R}}\right), \quad \bar{a} = \frac{1}{2}\left(a_{\mathrm{L}} + a_{\mathrm{R}}\right). \tag{A.13}$$

For the three-wave model, the left, right and intermediate wave speeds are estimated as

$$\begin{aligned} &s_{\mathrm{L}} = u_{\mathrm{L}} - a_{\mathrm{L}} q_{\mathrm{L}},\\ &s_{\mathrm{R}} = u_{\mathrm{R}} + a_{\mathrm{R}} q_{\mathrm{R}}, \end{aligned} \tag{A.14}$$

with

$$q_K = \begin{cases} 1 & \text{if } p^* \le p_K,\\ \left[1 + \frac{\gamma+1}{2\gamma}\left(\frac{p^*}{p_K} - 1\right)\right]^{1/2} & \text{if } p^* > p_K, \end{cases} \tag{A.15}$$

where the subscript K denotes the left or the right state, and

$$s^* = \frac{p_\mathrm{R} - p_\mathrm{L} + \rho_\mathrm{L} u_\mathrm{L}(s_\mathrm{L} - u_\mathrm{L}) - \rho_\mathrm{R} u_\mathrm{R}(s_\mathrm{R} - u_\mathrm{R})}{\rho_\mathrm{L}(s_\mathrm{L} - u_\mathrm{L}) - \rho_\mathrm{R}(s_\mathrm{R} - u_\mathrm{R})}. \tag{A.16}$$

The flux at the interface is then evaluated as

$$F = \begin{cases} F_\mathrm{L}, & s_\mathrm{L} > 0, \\ F_\mathrm{L} + s_\mathrm{L}(Q^*_\mathrm{L} - Q_\mathrm{L}), & s_\mathrm{L} \le 0 < s^*, \\ F_\mathrm{R} + s_\mathrm{R}(Q^*_\mathrm{R} - Q_\mathrm{R}), & s^* \le 0 < s_\mathrm{R}, \\ F_\mathrm{R}, & s_\mathrm{R} \le 0, \end{cases} \tag{A.17}$$

with the intermediate state vectors

$$Q^*_K = \left(\frac{s_K - u_K}{s_K - s^*}\right) \begin{bmatrix} \rho_K Y_{K,1} \\ \vdots \\ \rho_K Y_{K,N_\mathrm{sp}} \\ \rho_K s^* \\ \rho_K v_K \\ \rho_K w_K \\ \rho_K E_K + (s^* - u_K)\left[\rho_K s^* + \frac{p_K}{(s_K - u_K)}\right] \end{bmatrix}. \tag{A.18}$$

A 3-D matrix and its inverse matrix are used to rotate the flux before and after calling the Riemann solver. As introduced in Eq. (7), the transformation matrix T_s can be written as

$$T_\mathrm{s} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \ddots & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & a_x & a_y & a_z & 0 \\ 0 & 0 & 0 & b_x & b_y & b_z & 0 \\ 0 & 0 & 0 & c_x & c_y & c_z & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \tag{A.19}$$

where a_x, a_y and a_z are the components of the unit normal vector of the physical interface. b and c are the unit tangential vectors, which can be computed by an orthogonalization method.

Appendix B. Mapping functions used for the grid generation

The stretched mesh in Section 3.1 is mapped by a clustering function as

$$\begin{cases} y = r_d * L_\mathrm{y}\left[1 + \frac{\sinh[\beta(\eta - A)]}{\sinh(\beta A)}\right], A = \frac{1}{2\beta}\ln\left(\frac{1+(e^\beta - 1)r_d}{1+(e^{-\beta}-1)r_d}\right) \\ x = \xi * L_\mathrm{x} \\ z = r_d * L_\mathrm{z}\left[1 + \frac{\sinh[\beta(\zeta - A)]}{\sinh(\beta A)}\right], A = \frac{1}{2\beta}\ln\left(\frac{1+(e^\beta - 1)r_d}{1+(e^{-\beta}-1)r_d}\right) \end{cases}, \tag{B.1}$$

where the clustering factor β is 6 and the factor r_d is 0.5 for this case. L_x, L_y and L_z are the total length of the domain in x-, y- and z-direction, respectively. The following function is used to generate the skewed mesh used in Section 3.1:

$$\begin{cases} x = \xi + \eta\tan(\pi/12) \\ y = \eta + \zeta\tan(\pi/12) \\ z = \zeta + \xi\tan(\pi/12) \end{cases}. \tag{B.2}$$

For the distorted mesh in Section 3.1, the mapping function is given as

$$\begin{cases} x = \xi + L_\mathrm{x} S_\mathrm{x}\sin(2\pi\xi/L_\mathrm{x})\sin(2\pi\eta/L_\mathrm{y}) \\ z = \zeta \\ y = \eta + L_\mathrm{y} S_\mathrm{y}\sin(2\pi\xi L_\mathrm{x})\sin(2\pi\eta/L_\mathrm{y}) \end{cases}, \tag{B.3}$$

where the subscripts x and y denote the respective direction. L is the total length of the domain and S is the scaling factor in the respective direction. The scaling factor S is set to 0.075 for each direction. Both of these factors satisfy the constraint $0 \le 2\pi S \le 1$ to ensure that the mesh does not tangle. The same distorted mapping is used in Section 3.2, Section 3.3 and Section 3.4 (the z direction is not used). The different computational domains in these cases result in different physical domains. The use of negative computational coordinates leads to a distorted boundary as shown in Fig. 3, which is beneficial for verification. The non-negative computational coordinates hold the Cartesian boundary and distort the grid in the centre as shown in Fig. 8, which can be applied to a periodic boundary, i.e. in Section 3.2.

In Section 4.1, the pipe bend with a straight channel is constructed based on a piecewise function as

$$\begin{aligned} x &= \begin{cases} L_{\mathrm{x},1} + L_{\mathrm{x},1}(\eta - L_{\mathrm{y},1})/L_{\mathrm{y},1}, & \text{if } \eta \le L_{\mathrm{y},1} \\ L_{\mathrm{x},1} + (r_{in} + \xi W)\cos(0.5\pi\frac{\eta - L_{\mathrm{y},1}}{L_{\mathrm{y},2} - L_{\mathrm{y},1}} - 0.5\pi), & \\ & \text{if } L_{\mathrm{y},1} < \eta \le L_{\mathrm{y},2} \\ L_{\mathrm{x},1} + r_{in} + \xi W, & \text{otherwise} \end{cases} \\ y &= \begin{cases} (1 - \xi)W, & \text{if } \eta \le L_{\mathrm{y},1} \\ r_{out} + (r_{in} + \xi W)\sin(0.5\pi\frac{\eta - L_{\mathrm{y},1}}{L_{\mathrm{y},2} - L_{\mathrm{y},1}} - 0.5\pi), & \\ & \text{if } L_{\mathrm{y},1} < \eta \le L_{\mathrm{y},2} \\ r_{out} + L_{\mathrm{x},2}\frac{\eta - L_{\mathrm{y},2}}{1 - L_{\mathrm{y},2}}, & \text{otherwise} \end{cases} \\ z &= \zeta, \end{aligned} \tag{B.4}$$

where $L_{\mathrm{y},1} = 0.56$ and $L_{\mathrm{y},2} = 0.83$ are the bounds of the straight part of the pipe. The length of the horizontal straight pipe $L_{\mathrm{x},1}$ is 13 cm and that of the vertical straight pipe $L_{\mathrm{x},2}$ is 4 cm in this case. r_{in} and r_{out} are the inner (2 cm) and outer radius (6 cm) of the pipe bend. W is the pipe width (4 cm). The computational space is set to [0, 1] × [0, 1] × [0, 1.6].

In Section 4.2, a convex combination mapping strategy [54] is used that reads

$$\begin{aligned} & d = \max(|\eta|, |\zeta|), \\ & r = \max(\sqrt{\eta^2 + \zeta^2}, 1 \times 10^{-10}), \\ & \begin{cases} x = \xi \\ y = \left[d^3\eta/r + (1 - d^2)\eta\sqrt{2}\right] r_1 \\ z = \left[d^3\zeta/r + (1 - d^2)\zeta\sqrt{2}\right] r_1 \end{cases}. \end{aligned} \tag{B.5}$$

r_1 denotes the radius of the mapped cylinder. The computational space is set to $[0, 3] \times [-1, 1] \times [-1, 1]$.

For the 3-D detonation/boundary flow case in Section 4.3, the mesh is only stretched in the y-direction with β as 1.05,

$$\begin{cases} x = \xi \\ y = L_\mathrm{y}\frac{(\beta+1) - (\beta-1)\{[(\beta+1)/(\beta-1)]^{1-\eta}\}}{[(\beta+1)/(\beta-1)]^{1-\eta} + 1} \\ z = \zeta \end{cases}. \tag{B.6}$$

Appendix C. Analytical manufactured solution

The parameters for 3-D manufactured solutions are given in Table C.1.

Table C.1
Constants for 3-D Navier-Stokes supersonic manufactured solutions.

ϕ	ϕ_0	ϕ_x	$f_{s,x}$	a_{ϕ_x}	ϕ_y	$f_{s,y}$	a_{ϕ_y}	ϕ_z	$f_{s,z}$	a_{ϕ_z}
$\rho Y - O_2\ (kg/m^3)$	0.75	0.1	sin	1.0	0.2	cos	0.75	0.1	sin	0.5
$\rho Y - N_2\ (kg/m^3)$	1.0	−0.15	cos	0.5	0.1	sin	1.0	0.2	cos	0.75
$u (m/s)$	800	60	sin	1.5	30	sin	0.75	40	sin	0.5
$v (m/s)$	850	40	cos	1.0	50	sin	1.25	60	sin	0.5
$w (m/s)$	800	50	cos	1.25	60	sin	0.75	40	sin	1.0
$T (K)$	1000	100	cos	1.0	75	cos	0.5	50	cos	0.75
$\mu (kg/(m \cdot s))$	1.0	0.5	sin	0.5	0.3	cos	1.0	0.2	sin	0.75
$\kappa (W/(m \cdot K))$	1.0	0.3	cos	1.0	0.5	sin	1.0	0.4	cos	0.45
$D - O_2\ (m^2/s)$	1.0	0.25	sin	0.5	0.4	cos	0.75	0.3	cos	1.0
$D - N_2\ (m^2/s)$	1.0	-0.25	cos	0.75	0.5	sin	0.5	0.2	sin	0.75

References

[1] J.H. Lee, The Detonation Phenomenon, Cambridge University Press, 2008.
[2] W. Fickett, W.C. Davis, Detonation: Theory and Experiment, Courier Corporation, 2000.
[3] P. Wolański, Proc. Combust. Inst. 34 (1) (2013) 125–158.
[4] K. Kailasanath, AIAA J. 38 (9) (2000) 1698–1708.
[5] G.D. Roy, S.M. Frolov, A.A. Borisov, D.W. Netzer, Prog. Energy Combust. Sci. 30 (6) (2004) 545–672.
[6] S. Ashford, G. Emanuel, J. Propuls. Power 12 (2) (1996) 322–327.
[7] R. Zhou, D. Wu, J. Wang, Chin. J. Aeronaut. 29 (1) (2016) 15–29.
[8] F.K. Lu, E.M. Braun, J. Propuls. Power 30 (5) (2014) 1125–1142.
[9] D. Schwer, K. Kailasanath, Proc. Combust. Inst. 33 (2) (2011) 2195–2202.
[10] W. Rehm, C. Nae, W. Jahn, R. Vogelsang, B. Wang, Comput. Phys. Commun. 147 (1–2) (2002) 522–525.
[11] L.G. Marcantoni, J. Tamagno, S. Elaskar, Comput. Phys. Commun. 219 (2017) 209–222.
[12] E.S. Oran, J.P. Boris, J.P. Boris, Numerical Simulation of Reactive Flow, vol. 2, Cambridge University Press, Cambridge, 2001.
[13] C. Wang, C.-W. Shu, W. Han, J. Ning, Combust. Flame 160 (2) (2013) 447–462.
[14] M.J. Berger, P. Colella, J. Comput. Phys. 82 (1) (1989) 64–84.
[15] R. Deiterding, Parallel adaptive simulation of multi-dimensional detonation structures, Ph.D. thesis, Brandenburg University of Technology Cottbus–Senftenberg, 2003.
[16] R.J. LeVeque, et al., Finite Volume Methods for Hyperbolic Problems, vol. 31, Cambridge University Press, 2002.
[17] J.L. Ziegler, R. Deiterding, J.E. Shepherd, D.I. Pullin, J. Comput. Phys. 230 (20) (2011) 7598–7630.
[18] X. Cai, R. Deiterding, J. Liang, M. Sun, Y. Mahmoudi, J. Fluid Mech. 836 (2018) 324–351.
[19] H. Peng, Y. Huang, R. Deiterding, Z. Luan, F. Xing, Y. You, Combust. Flame 198 (2018) 69–80.
[20] H. Peng, Y. Huang, R. Deiterding, Y. You, Z. Luan, Combust. Sci. Technol. 193 (9) (2021) 1516–1537.
[21] R.P. Fedkiw, T. Aslam, B. Merriman, S. Osher, J. Comput. Phys. 152 (2) (1999) 457–492.
[22] R. Deiterding, Comput. Struct. 87 (11–12) (2009) 769–783.
[23] J.A. Sethian, Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry Fluid Mechanics, Computer Vision, and Materials Science, vol. 3, Cambridge University Press, 1999.
[24] S. Osher, R.P. Fedkiw, Level Set Methods and Dynamic Implicit Surfaces, vol. 1, Springer, New York, 2005.
[25] H. Peng, C.W. Atkins, R. Deiterding, Comput. Fluids 232 (2022) 105188.
[26] R.J. Kee, F.M. Rupley, J.A. Miller, Chemkin-II: a fortran chemical kinetics package for the analysis of gas-phase chemical kinetics, Tech. Rep, Sandia National Lab, Livermore, United States, 1989.
[27] E.F. Toro, Riemann Solvers and Numerical Methods for Fluid Dynamics: a Practical Introduction, Springer Science & Business Media, 2013.
[28] W.H. Press, S.A. Teukolsky, W.T. Vetterling, B.P. Flannery, in: The Art of Scientific Computing, 3rd edition, Cambridge University Press, 2007.
[29] K. Salari, P. Knupp, Code verification by the method of manufactured solutions, Tech. Rep., Sandia National Lab. (SNL-NM), Albuquerque, NM (United States), 2000.
[30] C.J. Roy, C. Nelson, T. Smith, C. Ober, Int. J. Numer. Methods Fluids 44 (6) (2004) 599–620.
[31] R.P. Fedkiw, A survey of chemically reacting, compressible flows, Ph.D. thesis, University of California, Los Angeles, 1996.
[32] P.J.M. Ferrer, R. Buttay, G. Lehnasch, A. Mura, Comput. Fluids 89 (2014) 88–110.
[33] C.K. Westbrook, Combust. Flame 46 (1982) 191–210.
[34] C.J. Jachimowski, An analytical study of the hydrogen-air reaction mechanism with application to scramjet combustion, Tech. Rep., NASA Langley Research Center, Hampton, Virginia, United States, 1988.
[35] G. Billet, V. Giovangigli, G. De Gassowski, Combust. Theory Model. 12 (2) (2008) 221–248.
[36] N. Attal, P. Ramaprabhu, J. Hossain, V. Karkhanis, M. Uddin, J. Gord, S. Roy, Comput. Fluids 107 (2015) 59–76.
[37] L. Owen, A fourth-order solution-adaptive finite-volume algorithm for compressible reacting flows on mapped domains, Ph.D. thesis, Colorado State University, 2019.
[38] X. Yuan, J. Zhou, Z. Lin, X. Cai, Int. J. Hydrog. Energy 41 (40) (2016) 18259–18272.
[39] D.N. Williams, L. Bauwens, E.S. Oran, Symp., Int., Combust. 26 (2) (1996) 2991–2998.
[40] M. Hanana, M. Lefebvre, Shock Waves 11 (2001) 77–88.
[41] N. Tsuboi, S. Katoh, A.K. Hayashi, Proc. Combust. Inst. 29 (2) (2002) 2783–2788.
[42] Y. Huang, H. Ji, F. Lien, H. Tang, Shock Waves 24 (4) (2014) 375–392.
[43] V. Monnier, V. Rodriguez, P. Vidal, R. Zitoun, Combust. Flame 245 (2022) 112310.
[44] J. Crane, J.T. Lipkowicz, X. Shi, I. Wlokas, A.M. Kempf, H. Wang, Proc. Combust. Inst. (2022), https://doi.org/10.1016/j.proci.2022.10.019.
[45] C. Campbell, J. Chem. Soc. 129 (1926) 3010–3021.
[46] G. Schott, Structure, chemistry, and instability of detonation in homogeneous low density fluids-gases, Tech. Rep., Los Alamos Scientific Lab., Univ. of California, N. Mex, 1965.
[47] G.L. Schott, Phys. Fluids 8 (5) (1965) 850–865.
[48] Y. Wu, J.H. Lee, Combust. Flame 162 (6) (2015) 2660–2669.
[49] B. Zhang, X. Shen, L. Pang, Y. Gao, Fuel 177 (2016) 1–7.
[50] N. Tsuboi, K. Eto, A. Hayashi, Combust. Flame 149 (1–2) (2007) 144–161.
[51] N. Tsuboi, Y. Daimon, A.K. Hayashi, Shock Waves 18 (5) (2008) 379–392.
[52] D.-R. Cho, S.-H. Won, J.-R. Shin, J.-Y. Choi, Proc. Combust. Inst. 34 (2) (2013) 1929–1937.
[53] W. Chen, J. Liang, X. Cai, Y. Mahmoudi, Phys. Fluids 32 (4) (2020) 046104.
[54] D.A. Calhoun, C. Helzel, R.J. LeVeque, SIAM Rev. 50 (4) (2008) 723–752.
[55] E.S. Oran, J.W. Weber Jr, E.I. Stefaniw, M.H. Lefebvre, J.D. Anderson Jr, Combust. Flame 113 (1–2) (1998) 147–163.
[56] E.S. Oran, V.N. Gamezo, Combust. Flame 148 (1–2) (2007) 4–47.
[57] V. Gamezo, E. Oran, A. Khokhlov, Proc. Combust. Inst. 30 (2) (2005) 1841–1847.
[58] X. Cai, J. Liang, R. Deiterding, Y. Mahmoudi, M. Sun, Combust. Flame 190 (2018) 201–215.