TYPE Methods
PUBLISHED 18 May 2023
DOI 10.3389/fenrg.2023.1203801

A total energy-based adaptive mesh refinement technique for the simulation of compressible flow

Xian Xu*, Yingchun Chen, Zhirong Han and Feng Zhou

Shanghai Aircraft Design and Research Institute, Shanghai, China

Check for updates

OPEN ACCESS

EDITED BY
Yang Yang,
Yangzhou University, China

REVIEWED BY
Wei Zhao,
Northwest University, China
Shuang-Xi Guo,
Chinese Academy of Sciences (CAS), China
Bo Chen,
Zhejiang Sci-Tech University, China

*CORRESPONDENCE
Xian Xu,
✉ xuxian@comac.cc

RECEIVED 11 April 2023
ACCEPTED 04 May 2023
PUBLISHED 18 May 2023

CITATION
Xu X, Chen Y, Han Z and Zhou F (2023), A total energy-based adaptive mesh refinement technique for the simulation of compressible flow.
Front. Energy Res. 11:1203801.
doi: 10.3389/fenrg.2023.1203801

In this paper, an adaptive mesh refinement technique is presented for simulation of compressible flows, which can effectively refine the mesh in the regions with shock waves and vortices. The present approach uses the total energy per unit volume as an indicator to capture the shock waves and vortical structures. In the approach, an h-refinement strategy is adopted. To save the computational effort, the flow variables on the new mesh are obtained from the previous step by interpolation, which ensures that the problem is always solved on the refined mesh. Both inviscid and viscous compressible flows are considered in this work. Their governing equations are, respectively, Euler equations and Navier–Stokes equations associated with the implementation of the Spalart–Allmaras turbulence model. The cell-centered finite volume method and Jameson scheme are chosen to carry out spatial discretization, and the five-stage Runge–Kutta scheme is applied to discretize the temporal derivative. The present approach is applied to simulate three test problems for its validation. Numerical results show that it can effectively capture the shock waves and vortices with improvement in solution accuracy.

KEYWORDS

adaptive technique, total energy per unit volume, shock wave, vortical structure, finite volume method

1 Introduction

To analyze the fluid characteristics, accurate simulation of the flow field is one of the main tasks in computational fluid dynamics (CFD). A high-quality mesh plays an important role in solution accuracy of the flow field. Generation of a good mesh usually requires some prior knowledge of the flow behavior in order to match the mesh point distribution to the essential features of the flow field. However, this prior knowledge may not always be available in advance. On the other hand, if the mesh is refined in the whole domain to guarantee the desired solution accuracy, the amount of computational time, effort, and resources may become excessive. How to balance the high-quality mesh and the computational effort is a critical issue. It seems that the solution-adaptive mesh refinement technique is an answer to this problem. It can effectively refine the mesh only in pivotal regions to improve the solution accuracy. In fact, research on this aspect is currently a hot topic in CFD (Harvey et al., 1992; Murayama et al., 2001; Yamazaki et al., 2007; Fossati et al., 2010).

One of the key issues in the solution-adaptive mesh refinement process is the identification of cells for refinement. Much effort has been devoted to this part (Murayama et al., 2001; Aftosmis and Berger, 2002; Jones et al., 2006; Yamazaki et al., 2007). The parameter used to identify the mesh refinement is generally called an adaptation function or indicator. Usually, the adaptation function is associated with some key physical variables such as density, entropy, kinetic energy, or a combination of them (Peraire et al., 1987). Some physical variables, such as helicity density and turbulent kinetic viscosity, are not always easy to derive, and the form of adaptation function may be complicated (Fossati et al., 2010). Under the premise of compressible flows, most adaptation functions are designed to resolve either shock waves or vortices (Pirzadeh, 1999; Ito et al., 2009). As the features of shock waves (high gradients for pressure and density) are quite different from those of vortices (high velocity gradients), a single adaptation function for compressible flows can usually only detect one feature of them. Wang et al. (2020) utilized adaptive mesh refinement (AMR) to capture vortices for improving the accuracy and efficiency of numerical simulation of the cavitation–vortex interaction. Steinthorsson et al. (1994) introduced a methodology based on the AMR algorithm of Berger and Colella (1989) for the accurate and efficient simulation of unsteady, compressible flows. Gou et al. (2018) introduced an accurate and robust AMR system suitable for turbomachinery applications and widely studied shock wave and tip leakage using the AMR method. Pantano et al. (2007) presented a methodology for the large-eddy simulation of compressible flows with a low-numerical dissipation scheme and structured adaptive mesh refinement (SAMR) used in turbulent flow regions while employing weighted essentially non-oscillatory order (WENO) to capture shocks. Papoutsakis et al. (2018) presented an adaptive mesh refinement (AMR) method suitable for hybrid unstructured meshes that allows for local refinement and de-refinement of the computational grid during the evolution of the flow to increase the order of accuracy in the region of shear layers and vortices. This paper takes this challenging issue and aims to present an indicator which can well detect both the shock waves and vortices in the compressible flow using h-type AMR.

This paper is organized as follows: in Section 2, the governing equations and methodologies for numerical discretization are briefly described. Section 3 depicts the solution-adaptive mesh refinement approach which involves the selection of the adaptation function and the h-refinement strategy (Pepper and Wang, 2007) based on body-fitted quadrilateral/hexahedral mesh. Section 4 presents three numerical examples to validate the present approach. We report conclusion in Section 5.

2 Governing equations and numerical discretization

The viscous, compressible flow of a perfect gas is governed by Navier–Stokes (N–S) equations (Blazek, 2001a). In a three-dimensional domain of volume Ω with boundary S, the integral form of the equations is expressed as

$$\frac{\partial}{\partial t}\int_{\Omega}\vec{W}d\Omega+\oint_{\partial\Omega}\left(\vec{F}_c-\vec{F}_v\right)dS=0. \tag{1}$$

Here, $\vec{W}$ is the vector of conservative variables which can be written in the three-dimensional form as

$$\vec{W}=\left[\rho\ \ \rho u\ \ \rho v\ \ \rho w\ \ \rho E\right]^T, \tag{2}$$

where ρ is the density of fluid; $u,\ v,\ w$ are the velocity components in x-, y-, and z-direction, respectively; and E is the energy. $\vec{F}_c$ is the vector of convective fluxes, which can be expressed as

$$\vec{F}_c=\begin{bmatrix}\rho V\\ \rho uV+n_x p\\ \rho vV+n_y p\\ \rho wV+n_z p\\ \rho HV\end{bmatrix}, \tag{3}$$

where p is the pressure and V is the velocity normal to the surface element dS, which is defined as the scalar product of the velocity vector and unit normal vector as follows:

$$V\equiv\vec{v}\cdot\vec{n}=n_x u+n_y v+n_z w. \tag{4}$$

The total enthalpy H is given by $E+p/\rho$. For the vector of viscous fluxes $\vec{F}_v$, we have

$$\vec{F}_v=\begin{bmatrix}0\\ n_x\tau_{xx}+n_y\tau_{xy}+n_z\tau_{xz}\\ n_x\tau_{yx}+n_y\tau_{yy}+n_z\tau_{yz}\\ n_x\tau_{zx}+n_y\tau_{zy}+n_z\tau_{zz}\\ n_x\Theta_x+n_y\Theta_y+n_z\Theta_z\end{bmatrix}, \tag{5}$$

where

$$\begin{aligned}\Theta_x&=u\tau_{xx}+v\tau_{xy}+w\tau_{xz}+k\frac{\partial T}{\partial x},\\ \Theta_y&=u\tau_{yx}+v\tau_{yy}+w\tau_{yz}+k\frac{\partial T}{\partial y},\\ \Theta_z&=u\tau_{zx}+v\tau_{zy}+w\tau_{zz}+k\frac{\partial T}{\partial z}\end{aligned} \tag{6}$$

are the terms describing the work of viscous stresses and the heat conduction in the fluid. T is the temperature. The thermal conductivity coefficient k is evaluated by

$$k=k_L+k_T=c_p\left(\frac{\mu_L}{Pr_L}+\frac{\mu_T}{Pr_T}\right). \tag{7}$$

Here, c_p is the specific heat coefficient at constant pressure, Pr_L and Pr_T are the laminar and turbulent Prandtl numbers, respectively; μ_L is the laminar viscosity; and μ_T is the turbulent eddy viscosity.

Generally, both Pr_L and Pr_T are assumed to be constant, i.e., $Pr_L=0.72$ and $Pr_T=0.9$. At the same time, the laminar viscosity can be calculated using the Sutherland formula: $\mu_L=10^{-6}\times1.45T^{1.5}/(T+110)$, where the temperature T is in Kelvin degree. For the turbulent eddy viscosity, it can be determined with the aid of the turbulence model (Blazek, 2001b).

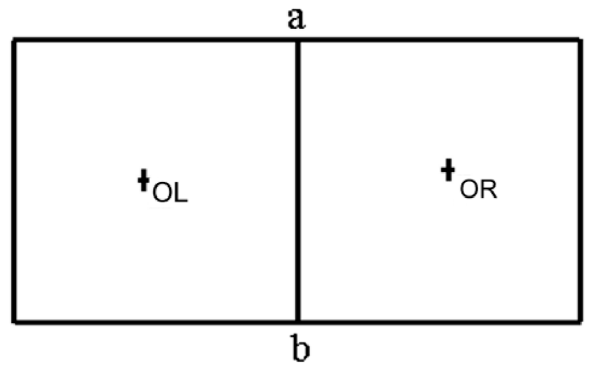

FIGURE 1
Cell-centered scheme of Jameson.

In this work, the Spalart–Allmaras (S–A) one-equation turbulence model (Spalart and Allmaras, 1992) is adopted. In this model, a transport equation is employed to calculate an eddy-viscosity variable $\tilde{\nu}$, and then the turbulent eddy viscosity μ_T is obtained from $\tilde{\nu}$.The S–A turbulence model allows for reasonably accurate predictions of turbulent flows with adverse pressure gradients. In this model, the solution at one point does not depend on the solution at other points. Therefore, it can be readily implemented on unstructured grids. It is also robust, converges fast to steady-state, and requires only moderate grid resolution in the near-wall region. The S–A model (Spalart and Allmaras, 1992) reads as follows:

$$\frac{\partial\tilde{\nu}}{\partial t}+\frac{\partial}{\partial x_i}(\tilde{\nu}u_i)=C_{b1}\tilde{S}\tilde{\nu}+\frac{1}{\sigma}\left\{\frac{\partial}{\partial x_j}\left[(\nu_L+\tilde{\nu})\frac{\partial\tilde{\nu}}{\partial x_j}\right]+C_{b2}\left(\frac{\partial\tilde{\nu}}{\partial x_j}\right)^2\right\}-C_{\omega 1}f_\omega\left(\frac{\tilde{\nu}}{d}\right)^2, \tag{8}$$

Here, $\nu_L=\mu_L/\rho$ is the laminar kinematic viscosity and $\sigma=2/3$.

The terms controlling the destruction of the eddy viscosity read

$$f_\omega=g\left(\frac{1+C_{\omega 3}^6}{g^6+C_{\omega 3}^6}\right)^{1/6},\quad g=r+C_{\omega 2}\left(r^6-r\right),\quad r=\frac{\tilde{\nu}}{\tilde{S}\kappa^2 d^2}, \tag{9}$$

and the production term $\tilde{S}$ is evaluated using the following formulae:

$$\tilde{S}=S_{SA}+\frac{\tilde{\nu}}{\kappa^2 d^2}f_{\nu 2},\quad f_{\nu 2}=1-\frac{\chi}{1+\chi f_{\nu 1}},\quad f_{\nu 1}=\frac{\chi^3}{\chi^3+C_{\nu 1}^3},\quad \chi=\frac{\tilde{\nu}}{\nu_L}. \tag{10}$$

Here, d is the distance to the closet wall and S_{SA} stands for the magnitude of the mean rotation rate, *i.e.*,

$$S_{SA}=\sqrt{2\Omega_{ij}\Omega_{ij}},\quad \Omega_{ij}=\frac{1}{2}\left(\frac{\partial u_i}{\partial x_i}-\frac{\partial u_j}{\partial x_i}\right). \tag{11}$$

Finally, the constants in Eqs 8–10 are defined as

$$C_{b1}=0.1355,\ C_{b2}=0.622,\ C_{\nu 1}=7.1,\ C_{\nu 2}=7.1,\\ \kappa=0.41,\ C_{\omega 1}=C_{b1}/\kappa^2+(1+C_{b2})/\sigma,\ C_{\omega 2}=0.3,\ C_{\omega 3}=2. \tag{12}$$

In current simulation, $\tilde{\nu}=0.1\nu_L$ is set as the initial value of $\tilde{\nu}$ as well as the inflow boundary conditions. At outflow boundaries, $\tilde{\nu}$ is simply extrapolated from the interior of the computational domain. At solid walls, it is appropriate to set $\tilde{\nu}=0$.

When the ideal gas is considered, we can simply neglect the viscous effect and get rid of $\vec{F}_v$; therefore, the N–S equations can be simplified to Euler equations.

To spatially discretize the governing equations, the finite volume method (FVM) is used in this study. FVM first divides the physical space into a number of control volumes and then integrates the governing equation (Eq. 1) over each control volume. In this work, the Jameson cell-centered scheme (Jameson et al., 1981) is used to define the location and shape of the control volume. The mean theorem is applied to approximate both the volume integral and the surface integral. The surface integral involves evaluation of fluxes at the interface between two neighboring control volumes. The flow quantities are stored at the centroids of each control volume which coincides with the grid cell. Note that with the mean theorem, the convective and the viscous fluxes are only evaluated at the center of each control surface using the flow information adjacent to two sides of the interface.

Two-dimensional grid is taken as an example. Using the cell-centered finite volume approach, in which the conserved variables $\vec{W}$ are located at the center of cells, the flux across the edge ab is calculated using the simple average of variables at the left cell center OL and right cell center OR (shown in Figure 1), e.g.,

$$\vec{W}_{ab}=\left(\vec{W}_{OL}+\vec{W}_{OR}\right)/2, \tag{13}$$

$$flux_{ab}=f\left(\vec{W}_{ab}\right). \tag{14}$$

Cell-centered schemes such as the one described previously would lead to odd–even decoupling of the solution, so that any errors are not damped and oscillations will be presented in the steady-state solution. Artificial dissipation terms D can eliminate these oscillations. Terms D are usually added to the convective fluxes in Eq. 8 and constructed as a blending of the second-order differences $d^{(2)}$ and the fourth-order differences $d^{(4)}$ of the conserved variables $\vec{W}$.

$$D=\sum_{i=1}^{kedges}d_i^{(2)}+\sum_{i=1}^{kedges}d_i^{(4)}, \tag{15}$$

$$d_i^{(2)}=\alpha_i\varepsilon_i^{(2)}\left(\vec{W}_{left}-\vec{W}_{right}\right)_i, \tag{16}$$

$$d_i^{(4)}=-\alpha_i\varepsilon_i^{(4)}\left(\nabla^2\vec{W}_{left}-\nabla^2\vec{W}_{right}\right)_i, \tag{17}$$

where the index i denotes the edges/faces delimiting the control volumes. $\vec{W}_{left}/\vec{W}_{right}$ denotes left/right cell conserved variables of the i^{th} edge/face. ∇^2 is defined for cell OL as

$$\nabla^2\vec{W}_{OL}=\sum_{OA=1}^{OLedges}\left(\vec{W}_{OA}-\vec{W}_{OL}\right), \tag{18}$$

where OA stands for the cells around the cell OL.

Adaptive coefficients $\varepsilon_i^{(2)}$ and $\varepsilon_i^{(4)}$ are defined as

$$\varepsilon_i^{(2)}=kk^{(2)}\nu_i,\varepsilon_i^{(4)}=\max\left(0,kk^{(4)}-\varepsilon_i^{(2)}\right), \tag{19}$$

$$\nu_i = \frac{\left|p_{left} - p_{right}\right|}{\left|p_{left} + p_{right}\right|}, \tag{20}$$

where $kk^{(2)}$ and $kk^{(4)}$ are two empirical constants, which typically have values ranging between $0.5 < kk^{(2)} < 1.0$ and $kk^{(4)} = kk^{(2)}/32.0$. p_{left} in the shock sensor ν_i denotes the left cell pressure of the i^{th} edge/face. The scaling factor α_i is defined as

$$\alpha_i = (|V| + c)\Delta S, \tag{21}$$

where c and ΔS are the local speed of sound and the length of the edge/face, respectively.

After spatial discretization, the resultant ordinary differential equations can be solved using the explicit five-stage Runge–Kutta scheme of Jameson et al. (1981).

$$\begin{aligned}
\vec{W}^{(0)} &= \vec{W}^{n},\\
\vec{W}^{(1)} &= \vec{W}^{(0)} - \frac{\alpha_1 \Delta t}{\Omega}\vec{R}^{n}\left(\vec{W}^{(0)}\right),\\
\vec{W}^{(2)} &= \vec{W}^{(0)} - \frac{\alpha_2 \Delta t}{\Omega}\vec{R}^{n}\left(\vec{W}^{(1)}\right),\\
\vec{W}^{(3)} &= \vec{W}^{(0)} - \frac{\alpha_3 \Delta t}{\Omega}\vec{R}^{n}\left(\vec{W}^{(2)}\right),\\
\vec{W}^{(4)} &= \vec{W}^{(0)} - \frac{\alpha_4 \Delta t}{\Omega}\vec{R}^{n}\left(\vec{W}^{(3)}\right),\\
\vec{W}^{n+1} &= \vec{W}^{(0)} - \frac{\alpha_5 \Delta t}{\Omega}\vec{R}^{n}\left(\vec{W}^{(4)}\right),
\end{aligned} \tag{22}$$

where $\vec{W}$ is the conservative variable in Eq. 1 and $\vec{R}$ is the corresponding residual. Δt is the time step, and Ω is the area of the cell. n is the current time level, $n+1$ is the new time level, and coefficients are taken as

$$\alpha_1 = 1/4, \alpha_2 = 1/6, \alpha_3 = 3/8, \alpha_4 = 1/2, \alpha_5 = 1. \tag{23}$$

3 Total energy-based adaptive mesh refinement approach

3.1 Selection of total energy per unit volume as the refinement indicator

After obtaining the basic solution of governing equations in the flow field, two important steps in any solution-adaptive mesh refinement technique must be followed. The first one requires an indicator to detect and locate the flow structures of interest, where the mesh refinement is needed. From a practical point of view, the adaptation function should indicate where the mesh must be refined to improve the accuracy. The adaptation function is usually selected from flow variables such as density, entropy, and turbulent kinetic energy. The flow considered in this paper is at high Reynolds number, and the Mach number ranges from low-compressible regime (Mach number < 0.3) to supersonic regime (Mach number > 1.0). In some cases, the viscous effect is very important. Thus, there is a practical demanding to select an adaptation function which can well capture both the features of shock waves and viscous effect.

It is known that any perturbation of flow must be followed by the perturbation of total energy per unit volume (ρE) in the flow field. For instance, when the air passes through the Mach wave, a perturbation of total energy per unit volume happens before and after the shock wave. ρE in the region with vortices is lower than that in other regions due to the effect of dissipation, and this is also a perturbation of total energy per unit volume. In this sense, ρE can be considered a universal adaptation variable. In this work, the adaptation function is constructed by the adaptation variable ρE as

$$\begin{aligned}
&\varphi_i = \frac{\Delta(\rho E)_i}{(\rho E)_\infty},\ \Delta(\rho E)_i = \max\left(\left|(\rho E)_i - (\rho E)_j\right|\right),\\
&\phi_i = \max\left(0, k_0 - \frac{(\rho E)_i}{(\rho E)_\infty}\right), \begin{cases} \text{Euler Eq.}\ \ k_0 = 0, \\ \text{N} - \text{S Eq.}\ \ k_0 = 1, \end{cases}
\end{aligned} \tag{24}$$

where subscripts ∞, i, and j stand for the free stream, the i^{th} grid cell, and the cells adjacent to cell i, respectively. $(\rho E)_\infty$ is used to normalize ϕ_i and φ_i. Since there is a perturbation near the shock wave and in the region with vortices, ϕ_i is supposed to capture the shock waves and vortical structures. Furthermore, φ_i mainly contributes to locate the vortical structure when k_0 is taken as 1 in the viscous flow. The standard deviations (Zhang et al., 2001) of ϕ and φ are taken as

$$\sigma_\phi = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\phi_i^{\,2}},\ \ \sigma_\varphi = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\varphi_i^{\,2}}. \tag{25}$$

The summation is only performed for cells which satisfy $\phi_i > \varepsilon\phi_{\max}$, $\varphi_i > \varepsilon\varphi_{\max}$, where ε is an empirical threshold and is equal to 0.05 in this paper.

In the refinement process, the value of $\phi_i + \varphi_i$ for each cell depends on the value of $\sigma_\varphi + \sigma_\phi$. When $\phi_i + \varphi_i$ is greater than or equal to $\sigma_\varphi + \sigma_\phi$, the corresponding i^{th} grid cell is flagged for refinement. ρE, the last component of vector $\vec{W}$ in Eq. 2, is easy to be obtained from either N–S equations or Euler equations. From Eq. 24, we know that $\phi_i + \varphi_i$ has no effect on mesh configuration and is a non-dimensional number. Therefore, it is very convenient to set the threshold for mesh refinement.

3.2 Mesh refinement process

After determining the indicator, the process of mesh refinement, as the second step in adaptive technique, can be executed. In general, adaptive methods can be roughly classified into three categories (Pirzadeh, 2000; Pepper and Wang, 2007): grid movement (r-refinement), grid enrichment (h-refinement), and local solution enhancement (p-refinement). Each method has its own merits and shortcomings (Pepper and Wang, 2007; Li et al., 2010).

In the r-refinement method, the value of the adaptation function directly determines the mesh spacing. As described previously, the values of adaptation function in this work do not represent the

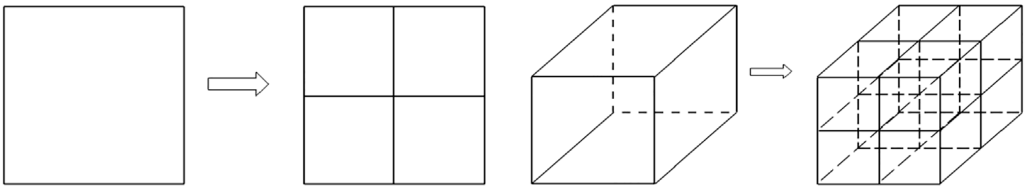

FIGURE 2
Refinement strategies.

magnitudes of estimated errors. Instead, they only indicate the location of dominant flow features. Therefore, r-refinement is not appropriate for this work. At the same time, although the p-refinement approach can obtain a solution with high order of accuracy in smooth flow regions, it is not effective in regions with discontinuity of flow variables. In addition, the coding of p-refinement is very complicated. In the methods based on h-refinement, however, an adaptation function only serves as a means to locate the regions which can be refined without considering the mesh spacing. Additionally, the h-refinement approach can be effectively applied in both the smooth flow and discontinuous flow regions. Due to these advantages, the h-refinement process is adopted in this work.

To reduce the number of cells, and in the meantime, to well capture the thin boundary layer, the structured mesh is taken as the background mesh. Grid cells are quadrilaterals for two-dimensional cases and hexahedra for three-dimensional cases. It should be indicated that as refinement process goes on, the overall unstructured mesh is formed.

During refinement, each quadrilateral/hexahedron is divided into four/eight sub-cells by joining the midpoints of opposite faces, as shown in Figure 2. For the two-dimensional case, to the initial coarse grid cell, a parent face is replaced by the current parent face and a new child face. This may form a hanging node (Spragle et al., 1995) between the initial coarse cell and the refined cells.

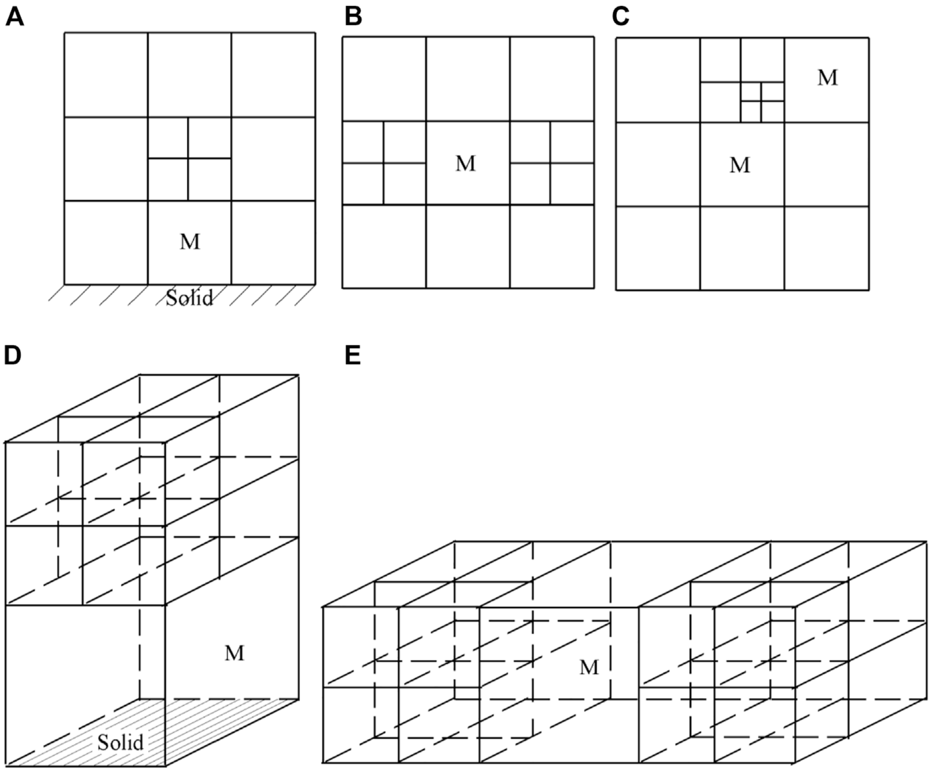

FIGURE 3
Unsmooth cells. **(A)** A two-dimensional cell has only one adaptive neighboring cell and its opposing neighboring cell is the wall. **(B)** A two-dimensional cell has two opposing adaptive neighboring cells. **(C)** A two-dimensional cell has a neighboring cell which has two or more adaptive levels. **(D)** A three-dimensional cell has only one adaptive neighboring cell and its opposing neighboring cell is the wall. **(E)** A three-dimensional cell has two opposing adaptive neighboring cells.

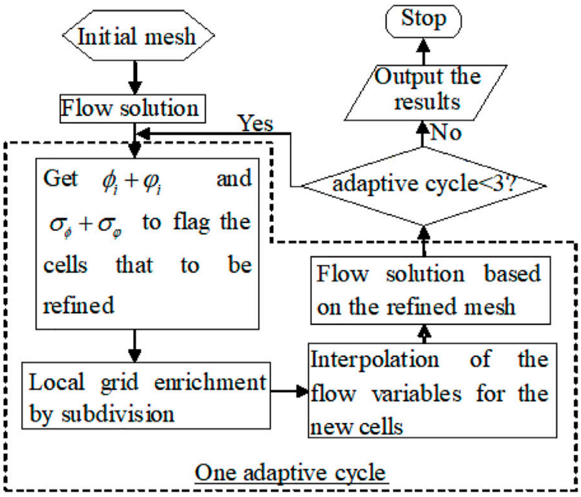
Initial mesh
Flow solution
Yes
Get $\phi_i + \varphi_i$ and $\sigma_\phi + \sigma_\varphi$ to flag the cells that to be refined
Local grid enrichment by subdivision
Stop
Output the results
No
adaptive cycle<3?
Flow solution based on the refined mesh
Interpolation of the flow variables for the new cells
One adaptive cycle

FIGURE 4
Flowchart of the process of the refinement strategy.

A node is a hanging node if it is not a vertex of all cells sharing one face. The hanging node grid adaptive scheme has the ability to efficiently operate on grids with a variety of cell shapes, including hybrid grids. However, the hanging node adaption scheme makes some solvers unusable, especially for the structured solvers. In contrast, the face-based unstructured solver presented by Li et al. (2010) provides an ideal environment for dealing with a hanging node adaption scheme. For the cell-centered scheme used in this work, as shown in Figure 1, the solver simply visits each face and then uses flow variables on its left and right grid cell to evaluate the face flux, and the contribution of the face flux is then sent to the two neighboring cells sharing the interface.

$$\begin{aligned} flux_{OL} &= flux_{OL} + flux_{ab}, \\ flux_{OR} &= flux_{OR} - flux_{ab}. \end{aligned} \tag{26}$$

Once the integration loop is performed along all face indexes, spatial discretization of the governing equations is completed.

To maintain a smooth variation of cell volume, additional cells are refined based on the relative position of neighboring cells or boundary conditions. There are three cases shown in Figure 3 in which cell M must be refined. One is that a cell has only one neighboring cell which is refined but its opposing neighboring cell is the solid wall (Figures 3A, D). The second is that a cell has two opposing neighboring cells (Figures 3B, E) or more than two neighboring cells are refined. If the initial grid needs more than one adaptation, then the third case (Figure 3C) arises: a cell has a neighboring cell which has two or more levels. For these cases, additional refinement is needed.

The flow field variables of new children cells can be interpolated from their parent cells. To carry out that, there are several different techniques. The simplest way, which is employed in this work, is to directly copy all physical variables of parent cells to their children cells. Such kind of implementation can improve the computational efficiency.

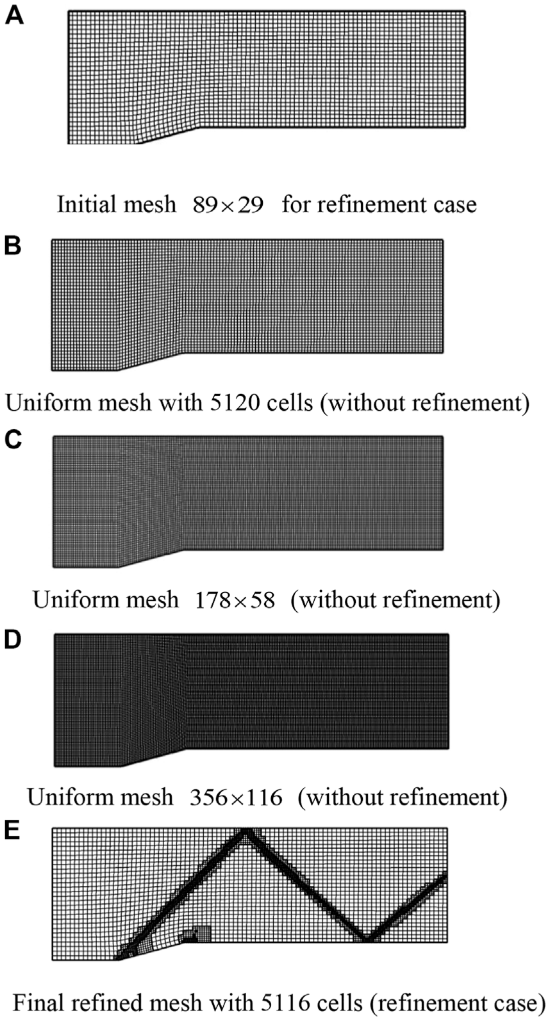
A
Initial mesh 89×29 for refinement case
B
Uniform mesh with 5120 cells (without refinement)
C
Uniform mesh 178×58 (without refinement)
D
Uniform mesh 356×116 (without refinement)
E
Final refined mesh with 5116 cells (refinement case)

FIGURE 5
Uniform meshes and final refined mesh after two levels of adaptation. **(A)** Initial mesh 89 × 29 for the refinement case. **(B)** Uniform mesh with 5,120 cells (without refinement). **(C)** Uniform mesh 178 × 58 (without refinement). **(D)** Uniform mesh 356 × 116 (without refinement). **(E)** Final refined mesh with 5,116 cells (refinement case).

The process of the refinement strategy is summarized in the flowchart as shown in Figure 4. On the other hand, it should be noted that the coarsening process, which is not required in this paper, can be easily implemented based on the obtained information during the refinement process.

4 Numerical examples

To validate the present approach and demonstrate its capability for effective simulation of inviscid/viscous compressible flows, three typical problems are selected. The first problem is a two-dimensional inviscid flow in a channel with a 15° ramp with the typical multi-channel shock wave and expansion wave, to demonstrate the capability to capture shock waves. The second problem is the three-dimensional inviscid flow around the ONERA M6 wing at a transonic speed resulting in a λ

TABLE 1 Comparison of computational cost among different meshes.

	Initial mesh cell number	Final mesh cell number	Time/s
Refinement (two adaption levels)	2,581	5,116	113
	5,120	5,120	37
Without refinement	10,324	10,324	104
	41,296	41,296	1,106

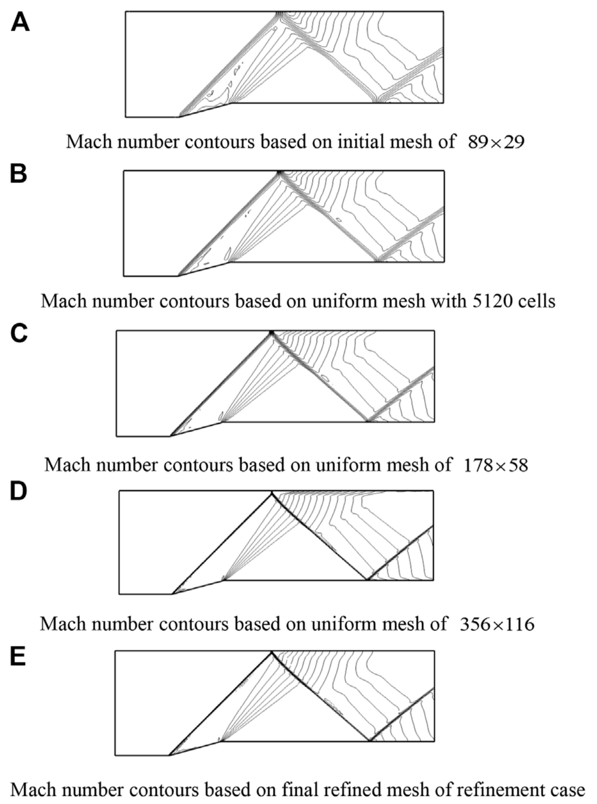

FIGURE 6
Mach number contours in the contracted channel based on uniform meshes and final refined mesh. **(A)** Mach number contours based on the initial mesh of 89 × 29. **(B)** Mach number contours based on the uniform mesh with 5,120 cells. **(C)** Mach number contours based on the uniform mesh of 178 × 58. **(D)** Mach number contours based on the uniform mesh of 356 × 116. **(E)** Mach number contours based on the final refined mesh of the refinement case.

shock wave on the upper surface of the wing, to show the capability of the algorithm to capture shock waves in a transonic flow. In the third problem, we hope to show the capability of the algorithm on simulating a detached vortex of a delta wing, especially the vortex core fragmentation, which is important in calculating the lift and drag coefficients.

4.1 Two-dimensional inviscid flow in a channel with a ramp

By solving Euler equations, the results for supersonic flow at a Mach number of 2 in a channel with a 15^0 ramp ranging from $x = 0.5$ to $x = 1.0$ can be obtained. The upper and bottom boundaries are solid walls, and the left and right boundaries are the inlet and outlet of the flow field, respectively.

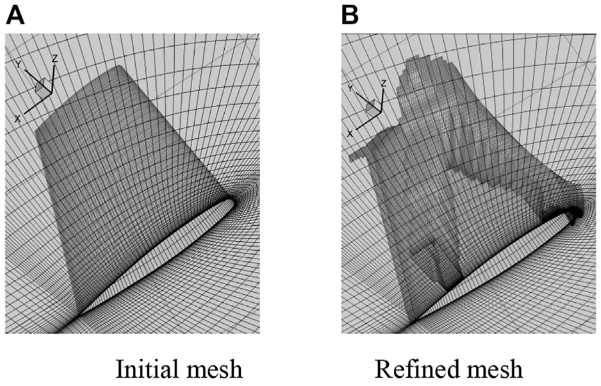

FIGURE 7
Initial and refined mesh for the ONERA M6 wing. **(A)** Initial mesh. **(B)** Refined mesh.

The initial mesh is a uniform mesh containing 89 × 29 quadrilateral cells as shown in Figure 5A. Figure 6A shows the Mach number contour lines for the entire flow field based on the initial mesh. The shock wave emanating from the leading ramp and its reflection as well as the expansion wave can be clearly captured. After two levels of adaptation are implemented, the final refined mesh, as depicted in Figure 5E, contains 5,116 quadrilateral cells. The Mach contours on the final refined mesh are displayed in Figure 6E.

To compare the computational cost (cells and seconds) between cases with refinement and without refinement, numerical simulation of the same problem on the three uniform meshes without refinement is carried out. The three uniform meshes without refinement are shown in Figures 5B–D. The mesh in Figure 5B has 5,120 cells. The number of cells for this mesh is the same as the final refined mesh of the refinement case. The mesh in Figure 5C has 10,324 cells which are approximately two times the cells in the final refined mesh of the refinement case. The mesh spacing for this case is the same as the minimum mesh spacing of the refinement case when the initial mesh shown in Figure 5A is refined by one adaption level. The mesh in Figure 5D has 41,296 cells which are approximately eight times the cells in the final refined mesh of the refinement case. The mesh spacing for this case is the same as the minimum mesh spacing of the refinement case when the initial mesh in Figure 5A is refined by two adaption levels. All the results (Mach number contour) are demonstrated in Figure 6. The minimum and maximum levels of contours are 1.0 and 1.8 respectively, and the number of levels is 14. Figures 6A–D show that the shock wave becomes thinner and thinner with the increase in cell numbers. It is seen clear from the figure that the results of the uniform mesh of 41,296 cells are very close to those of the final refined mesh where two adaption levels are used to refine the mesh. This can be well understood as both cases have the same mesh spacing in the vicinity of the

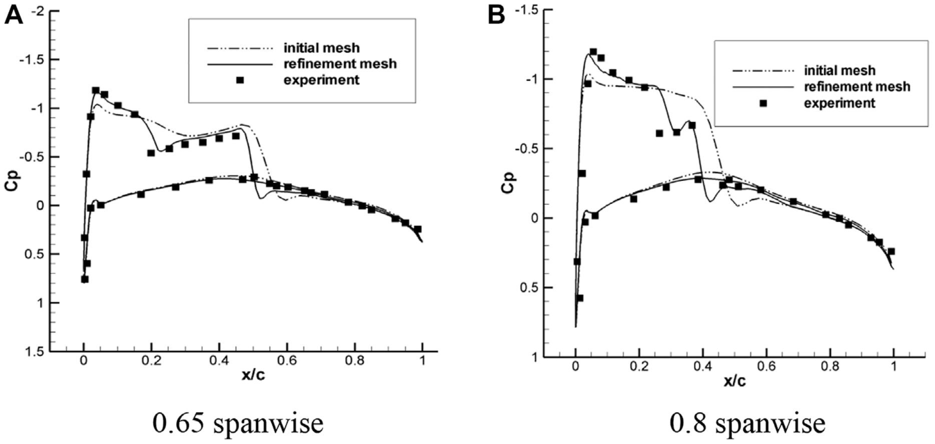

FIGURE 8
Pressure coefficient distributions at 0.65 and 0.8 spanwise locations for the ONERA M6 wing. **(A)** 0.65 spanwise. **(B)** 0.8 spanwise.

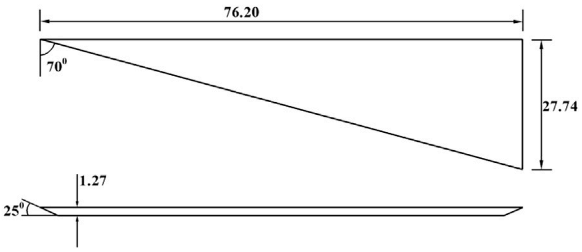

FIGURE 9
Flat-plate semispan delta wing model.

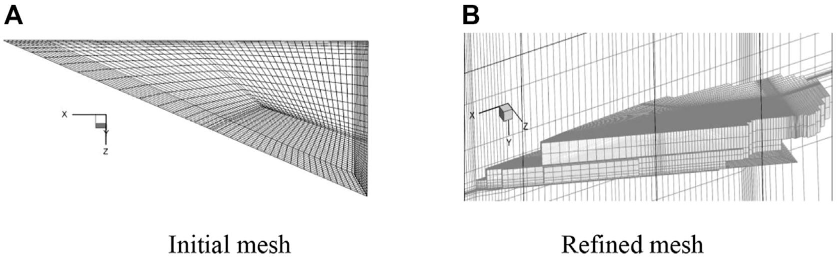

FIGURE 10
Initial mesh and refined mesh for the delta wing. **(A)** Initial mesh. **(B)** Refined mesh.

shock wave. The computational time required on the meshes shown in Figure 5 is listed in Table 1. It can be observed from Table 1 that the refinement case with two adaption levels only takes approximately 10% of the computational time of the uniform mesh case (without refinement) when the mesh spacing near the shock wave is kept the same. This well demonstrates high computational efficiency of the solution-adaptive approach.

4.2 Three-dimensional inviscid flow around the ONERA M6 wing

To further demonstrate the capability of the present solution-adaptive mesh refinement approach for capturing the shock wave, the inviscid flow around the ONERA M6 wing is considered. The Mach number is 0.84, and the incidence is 3.06°. Same as the case in Section 4.1, the governing equations are Euler equations.

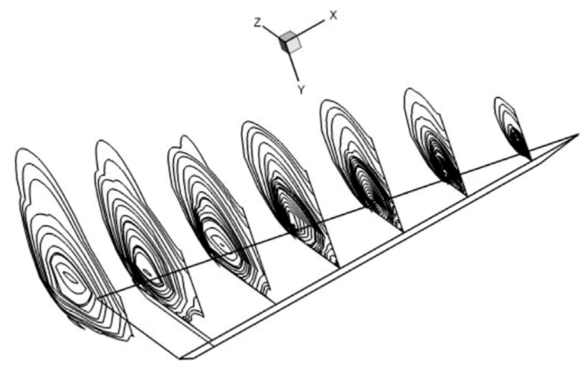

FIGURE 11
Contours of $\phi_i + \varphi_i$ at different cross sections.

A background mesh with a nearly uniform grid distribution is generated as shown in Figure 7A. The initial mesh contains 294,912 hexahedral cells. Figure 7B shows the final refined mesh which contains 440,365 hexahedral cells. Obviously, the majority of mesh refinement occurs in the shock wave regions, and the refined cells clearly outline the λ shock waves.

The surface pressure coefficient distributions at 0.65 and 0.8 spanwise locations obtained on the initial coarse mesh and the final refined mesh are compared in Figure 8 with the experimental data (Schmitt and Charpin, 1979). The comparison indicates that the shock wave is diffused due to the coarse mesh. The high resolution of shock waves by the refined mesh is evident as revealed by sharp discontinuity of the pressure distribution. There are some deviations between the numerical results and the experimental data. The reason may be due to the fact that the inviscid flow is assumed in the numerical computation, while the real flow always involves viscous effect.

Numerical simulations for two-dimensional inviscid flows in a channel with a ramp and three-dimensional inviscid flows around the ONERA M6 wing well demonstrate the capability of the present solution-adaptive mesh refinement approach for capturing shock waves and expansion waves. To illustrate the ability of the present approach for capturing vortices, numerical simulation of three-dimensional compressible viscous flows around a delta wing is considered in the following example. The prediction of leading-edge vortex breakdown on a delta wing at high angles of attack is made.

4.3 Three-dimensional viscous flow around a delta wing

The geometry chosen for this case is a flat-plate semispan delta wing with a leading-edge sweep of 70^0 and a 25^0 bevel on the lower surface along the leading and trailing edges (Agrawal et al., 1992) as shown in Figure 9. For this problem, N–S equations with the S–A turbulent model are taken as governing equations. Numerical simulation is carried out at a Mach number of 0.3 and an incidence of 30^0. The Reynolds number based on the root chord is taken as one million.

The initial mesh and the adaptive refined mesh are displayed in Figure 10. The corresponding adaptation function is shown in Figure 11. The initial mesh contains 320,624 hexahedral cells. After refinement, the final mesh contains 503,317 hexahedral cells.

Figure 12 shows some numerical results for this problem. The solution presented in Figure 12A shows the vortex core line without vortex breakdown, which is different from the solution based on the refined mesh presented in Figure 12B.

5 Conclusion

A solution-adaptive mesh refinement technique, which is based on the total energy per unit volume as the refinement indicator, is presented in this work. Different from previous adaptation indicator used, the present indicator can well detect both the shock waves and vortices. The technique is validated by applying it to simulate two-dimensional and three-dimensional steady compressible inviscid/viscous flows. The h-refinement approach by subdivision is adopted to perform the mesh refinement process.

The capability of the present solver is verified by applying it to handle three test problems. For inviscid/viscous flows, the Euler/

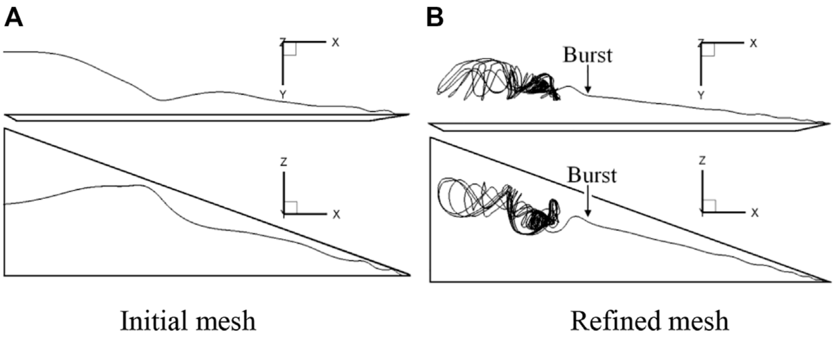

FIGURE 12
Vortex breakdown position indicated by the core line, $Ma_\infty = 0.3, \alpha = 30^0$, Re = 1.0E6. A vortex breakdown at approximately 0.6 root chord observed from the adaptive solution is supported by both experimental data and other numerical results (Agrawal et al., 1992; O'Neil et al., 1989). **(A)** Initial mesh. **(B)** Refined mesh.

Navier–Stokes (with the Spalart–Allmaras one-equation turbulence model) equations are employed. The cell-centered finite volume method is applied for spatial discretization, and an explicit five-stage Runge–Kutta scheme is used to implement time integration. Numerical results show that the proposed adaptation function can well capture the shock waves, expansion waves, and vortices in the flow field. As a consequence, a high-resolution solution of important flow features such as shock waves and vortices is obtained.

Data availability statement

The original contributions presented in the study are included in the article/Supplementary Material; further inquiries can be directed to the corresponding author.

Author contributions

Conceptualization and methodology, XX; methodology, YC; resources and data curation, ZH; writing—original draft preparation, XX; writing—review and editing, ZH; supervision, FZ; project administration, XX; funding acquisition, YC. All authors listed have made a substantial, direct, and intellectual contribution to the work and approved it for publication.

Funding

This research was funded by the National Key R&D Program of China (Project No. 2020YFA0712000).

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors, and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Aftosmis, M. J., and Berger, M. J. "Multilevel error estimation and adaptive h-refinement for Cartesian meshes with embedded boundaries," in Proceedings of the 40th AIAA Aerospace Sciences Meeting & Exhibit, Reno, NV, USA, January 2002. doi:10.2514/6.2002-863

Agrawal, S., Barnett, R. M., and Robinson, B. A. (1992). Numerical investigation of vortex breakdown on a delta wing. *AIAA J.* 30 (3), 584–591. doi:10.2514/3.10960

Berger, M. J., and Colella, P. (1989). Local adaptive mesh refinement for shock hydrodynamics. *J. Comput. Phys.* 82 (1), 64–84. doi:10.1016/0021-9991(89)90035-1

Blazek, J. (2001). *Computational fluid dynamics: Principles and applications.* Kidlington, England: Elsevier Science Ltd, 5–22.

Blazek, J. (2001). *Computational fluid dynamics: Principles and applications.* Kidlington, England: Elsevier Science Ltd, 225–265.

Fossati, M., Guardone, A., and Vigevano, L. (2010). A node-pair finite element/finite volume mesh adaptation technique for compressible flows. *Int. J. Numer. Methods Fluids* 70. doi:10.1002/fld.2728

Gou, J., Yuan, X., and Su, X. (2018). Adaptive mesh refinement method based investigation of the interaction between shock wave, boundary layer, and tip vortex in a transonic compressor. *Proc. Institution Mech. Eng. Part G J. Aerosp. Eng.* 232 (4), 694–715. doi:10.1177/0954410016687142

Harvey, A. D., Acharya, S., and Lawrence, S. L. (1992). Solution-adaptive grid procedure for the parabolized Navier-Stokes equations. *AIAA J.* 30 (4), 953–962. doi:10.2514/3.11014

Ito, Y., Shih, A., Koomullil, R., Kasmai, N., Jankun-Kelly, M., and Thompson, D. (2009). Solution adaptive mesh generation using feature-aligned embedded surface meshes. *AIAA J.* 47 (8), 1879–1888. doi:10.2514/1.39378

Jameson, A., Schmidt, W., and Turkel, E. "Numerical solutions of the Euler equations by finite volume methods using Runge-Kutta time-stepping schemes," in Proceedings of the 14th Fluid and Plasma Dynamics Conference, Palo Alto, CA, USA, June 1981. doi:10.2514/6.1981-1259

Jones, W. T., Nielsen, E. J., and Park, M. A. "Validation of 3D adjoint based error estimation and mesh adaptation for sonic boom prediction," in Proceedings of the 44th AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, USA, January 2006. doi:10.2514/6.2006-1150

Li, Y., Premasuthan, S., and Jameson, A. "Comparison of h- and p-adaptations for spectral difference methods," in Proceedings of the 40th Fluid Dynamics Conference and Exhibit, Chicago, IL, USA, June 2010.

Murayama, M., Nakahashi, K., and Sawada, K. (2001). Simulation of vortex breakdown using adaptive grid refinement with vortex-center identification. *AIAA J.* 39 (7), 1305–1312. doi:10.2514/2.1448

O'Neil, P. J., Roos, F. W., Kegelman, J. T., Barnett, R. M., and Hawk, J. D. (1989) NADC-89114-60. Canada: NRC.Investigation of flow characteristics of a developed vortex

Pantano, C., Deiterding, R., Hill, D. J., and Pullin, D. (2007). A low numerical dissipation patch-based adaptive mesh refinement method for large-eddy simulation of compressible flows. *J. Comput. Phys.* 221 (1), 63–87. doi:10.1016/j.jcp.2006.06.011

Papoutsakis, A., Sazhin, S. S., Begg, S., Danaila, I., and Luddens, F. (2018). An efficient Adaptive Mesh Refinement (AMR) algorithm for the Discontinuous Galerkin method: Applications for the computation of compressible two-phase flows. *J. Comput. Phys.* 363, 399–427. doi:10.1016/j.jcp.2018.02.048

Pepper, D. W., and Wang, X. (2007). Application of an h-adaptive finite element model for wind energy assessment in Nevada. *Renew. Energy* 32, 1705–1722. doi:10.1016/j.renene.2006.08.011

Peraire, J., Vadhati, M., Morgan, K., and Zienkiewicz, O. C. (1987). Adaptive remeshing for compressible flow computations. *J. Comput. Phys.* 72, 449–466. doi:10.1016/0021-9991(87)90093-3

Pirzadeh, S. Z. (2000). A solution-adaptive unstructured grid method by grid subdivision and local remeshing. *J. Aircr.* 37 (5), 818–824. doi:10.2514/2.2675

Pirzadeh, S. Z. "An adaptive unstructured grid method by grid subdivision, local remeshing, and grid movement," in Proceedings of the 14th Computational Fluid Dynamics Conference, Norfolk, VA, USA, November 1999. doi:10.2514/6.1999-3255

Schmitt, V., and Charpin, F. (1979). Pressure distributions on the ONERA M6-wing at transonic mach numbers. *AGARD Ar.* 138.

Spalart, S. R., and Allmaras, S. A. "A one-equation turbulence model for aerodynamic flows," in Proceedings of the 30th Aerospace Sciences Meeting and Exhibit, Reno, NV, USA, January 1992. doi:10.2514/6.1992-439

Spragle, G. S., Smith, W. A., and Weiss, J. M. "Hanging node solution adaption on unstructured grids," in Proceedings of the 33rd Aerospace Sciences Meeting and Exhibit, Reno, NV, USA, January 1995. doi:10.2514/6.1995-216

Steinthorsson, E., Modiano, D., and Colella, P. "Computations of unsteady viscous compressible flows using adaptive mesh refinement in curvilinear body-fitted grid systems," in Proceedings of the Fluid Dynamics Conference, Colorado Springs, CO, USA, June 1994. doi:10.2514/6.1994-2330

Wang, Z., Li, L., Cheng, H., and Ji, B. (2020). Numerical investigation of unsteady cloud cavitating flow around the Clark-Y hydrofoil with adaptive mesh refinement using OpenFOAM. *Ocean. Eng.* 206, 107349. doi:10.1016/j.oceaneng.2020.107349

Yamazaki, W., Matsushima, K., and Nakahashi, K. (2007). Drag decomposition-based adaptive mesh refinement. *J. Aircr.* 44 (6), 1896–1905. doi:10.2514/1.31064

Zhang, S. J., Liu, J., and Chen, Y. S. "Adaptation for hybrid unstructured grid with hanging node method," in Proceedings of the 15th AIAA Computational Fluid Dynamics Conference, Anaheim,CA,U.S.A, June 2001. doi:10.2514/6.2001-2657