Chih-Yung Wen
Yazhong Jiang
Lisong Shi

Space–Time Conservation Element and Solution Element Method

Advances and Applications in Engineering Sciences

Springer

Engineering Applications of Computational Methods

Volume 13

Series Editors

Liang Gao, State Key Laboratory of Digital Manufacturing Equipment
and Technology, Huazhong University of Science and Technology,
Wuhan, Hubei, China

Akhil Garg, School of Mechanical Science and Engineering, Huazhong University
of Science and Technology, Wuhan, Hubei, China

The book series Engineering Applications of Computational Methods addresses the numerous applications of mathematical theory and latest computational or numerical methods in various fields of engineering. It emphasizes the practical application of these methods, with possible aspects in programming. New and developing computational methods using big data, machine learning and AI are discussed in this book series, and could be applied to engineering fields, such as manufacturing, industrial engineering, control engineering, civil engineering, energy engineering and material engineering.

The book series Engineering Applications of Computational Methods aims to introduce important computational methods adopted in different engineering projects to researchers and engineers. The individual book volumes in the series are thematic. The goal of each volume is to give readers a comprehensive overview of how the computational methods in a certain engineering area can be used. As a collection, the series provides valuable resources to a wide audience in academia, the engineering research community, industry and anyone else who are looking to expand their knowledge of computational methods.

This book series is indexed in both the **Scopus** and **Compendex** databases.

Chih-Yung Wen · Yazhong Jiang · Lisong Shi

Space–Time Conservation Element and Solution Element Method

Advances and Applications in Engineering Sciences

Chih-Yung Wen [ID]
Department of Aeronautical and Aviation
Engineering, Faculty of Engineering
The Hong Kong Polytechnic University
Hong Kong, China

Lisong Shi [ID]
Department of Aeronautical and Aviation
Engineering, Faculty of Engineering
The Hong Kong Polytechnic University
Hong Kong, China

Yazhong Jiang [ID]
Hubei Key Laboratory of Theory
and Application of Advanced Materials
Mechanics, School of Science
Wuhan University of Technology
Wuhan, Hubei, China

Preface

Methods for numerical simulations in engineering sciences keep updating every day, and naturally the state-of-the-art advancement may not be incorporated into the well-known textbooks. This short book introduces the Conservation Element and Solution Element (CESE) method, which has a relatively short history. The CESE method provides time-accurate simulations of physical systems and overcomes the difficulties in capturing discontinuities, by combining many innovative considerations in a nontraditional manner. This book elaborates on how the CESE method works and presents a comprehensive survey of the recent CESE developments and applications.

Hong Kong, China   Chih-Yung Wen
Wuhan, China   Yazhong Jiang
Hong Kong, China   Lisong Shi

Contents

Chapter 1
Introduction

With the rapid development of electronic computers, numerical computation has become an important paradigm of scientific discovery as well as a powerful tool for engineering research. Solving complex problems in a computational fashion is more than applying theories, equations, and formulas. Computational methods, also called algorithms or schemes, have strong influences on the outcomes of computations.

The space–time conservation element and solution element (CESE) method is aimed to numerically solve equations of conservation laws in various physical systems. The major concern herein will be the computational fluid dynamics (CFD), which is a hot area in scientific and engineering researches. Nevertheless, it may help at the outset to recognize that the conservation laws in CFD problems have a lot in common with conservation laws in other fields, such as acoustics, solid dynamics, electromagnetics, and magnetohydrodynamics. The physics may be different, but the mathematics are similar. For instance, physics involving dynamical evolution of waves and discontinuities are usually modelled by time-dependent nonlinear hyperbolic partial differential equations (PDEs). Some CFD problems happen to be representative of such physical problems in a wider context.

1.1 Background

Most of the key achievements in conventional CFD have been incorporated into the computational procedure of the finite volume method (FVM). Indeed, the FVM is well established and widely used, and therefore it is sometimes regarded as a mature technique. There are two important steps in the FVM, namely the reconstruction and the evolution [1, 2]. In the reconstruction step, one needs to locally approximate the flow field with simple functions such as polynomials, and then interpolate the values of flow quantities at the cell interfaces by utilizing the cell-average values. In the evolution step, the numerical flux at each cell interface is determined by some kind

of flux technique, and then the time integration of the conservation equations can be performed.

A major category of CFD problems face the challenges brought by (1) the coexistence of smooth regions and discontinuities in the flow (e.g., shock waves and material interfaces) and (2) the wide spectrum of wave numbers/frequencies in the flow (e.g., acoustic waves and turbulence). Due to these physical phenomena, the CFD solver is required to well handle the numerical dissipation, which leads to an endless debate in the CFD community. First, for the robustness in capturing shock waves, there must be a certain amount of numerical dissipation to suppress the spurious oscillations. However, a low numerical dissipation is very favourable for the accuracy and resolution of the viscous layers, contact surfaces, and small-scale structures in the flow, which would be smeared out by excessive dissipation.

To achieve the low but controllable numerical dissipation, the research works on FVM are mostly dedicated to two approaches: (1) developing high-order reconstruction techniques in conjunction with nonlinear limiters and (2) improving the flux solvers (e.g., approximate Riemann solvers). In both ways, numerous important advances have been obtained.

In spite of these successes in the FVM development, several shortcomings may arise. (1) A conventional FVM implementation highly relies on the selections of special techniques, such as the limiter and the Riemann solver. Actually, each special technique has its own pros and cons. There are many alternatives for each option, but none of them can be proven to be optimal and general. (2) The coupling of space and time is usually overlooked and may not be guaranteed in the numerical solution. (3) The multi-dimensionality can be questionable because most flux solvers are only based on one-dimensional physics. (4) The popular high-order FVM schemes are usually not compact, i.e., large stencils are used. Under this background, the CESE method was proposed to overcome the difficulties in the conventional FVM to some extent, and it became a new member in the family of FVM.

1.2   History of the CESE Method

The space–time conservation element and solution element method was initiated by Sin-Chung Chang of NASA Glenn Research Centre and his collaborators for aerodynamic computation in early 1990s [3, 4]. There were several motivations to propose the CESE method. First, they wanted to construct multidimensional and space–time unified CFD schemes, without using dimensional splitting or separated treatments of space discretization and time discretization. Second, they believed a CESE solver should be built from a non-dissipative core scheme so that numerical dissipation can be controlled effectively, dynamically, and even actively. Third, they attempted to avoid using the Riemann solver in the CESE method.

Around 1994, the CESE project was approved and supported by NASA Glenn Research Centre. The work of the research group showed that this time-accurate scheme possesses low numerical dissipation, which is valuable in CFD simulation.

For the first time, the transonic resonance in a convergent-divergent nozzle with frequency-staging effect was successfully simulated by the CESE scheme [5, 6]. In this case, the blind prediction by the CESE scheme matched the experimental data. Therefore, the CESE method is proved to be capable of simultaneously simulating multidimensional unsteady shock waves and acoustic waves with high accuracy.

Owing to its accuracy and robustness, the CESE method is employed in the CFD module of the simulation software LS-DYNA [7]. The Jacobs Technology Inc. (designer of the NASA's hypersonic flow test facilities) developed its own in-house CESE code called JUSTUS (Jacobs Unified Space–Time Unstructured Solver) for hypersonic flow simulations. Another in-house CESE code, called "ez4d" software, was developed by the NASA Langley Research Centre. This is a time-accurate 3D Navier–Stokes flow solver on unstructured meshes [8, 8].

1.3   Main Features of the CESE Method

The CESE method is a special finite-volume-type numerical method, with the aim of solving the governing equations of fluid dynamics and other conservation laws in various physical systems. The CESE method possesses many features such as the unified treatment for space and time, the fully discrete one-step explicit scheme, and the highly compact stencil. Without enlarging the stencil or adding stages of time integration, the CESE method can achieve arbitrary high-order accuracy for both space and time.

The essential ingredients in the CESE method include: (1) the spatial derivatives of physical variables are stored as independent unknowns, in addition to the physical variables themselves. In every time step, these derivatives are updated by a specially designed procedure. (2) a staggered mesh and a staggered time-marching strategy are employed. (3) the interior structure within each solution element is built with the Taylor expansion. (4) the time-marching approach is based on the Cauchy–Kowalewski procedure.

Through a combination of the above techniques, the CESE method has low dissipation and high compactness. When applied to the simulations of complex physical processes, the CESE method can catch shock waves, contact discontinuities, fine structures, and small disturbances, with high resolution and strong robustness. Therefore, the CESE method demonstrates good performances in the numerical simulations of wave-propagation problems (e.g., shock waves, acoustic waves, detonation waves, stress waves in solid, and the electromagnetic waves), interfacial instabilities, as well as the interactions of gaseous and liquid phases. In many research areas including high-speed aerodynamics, shock dynamics, detonation, aeroacoustics, and solid dynamics, the CESE method proves to be suitable and shows a good development prospect.

1.4 Outline of the Book

The remainder of this book is organized as follows. First of all, the non-dissipative core scheme of the CESE method is introduced in Chap. 2, with detailed descriptions of the basic concepts. Then, the practical shock-capturing CESE schemes with numerical dissipation including the classical a–α scheme, the Courant number insensitive scheme, and the recently proposed upwind CESE schemes are presented in Chap. 3. Furthermore, Chaps. 4 and 5 extend the CESE method to multidimensional and high-order versions. In Chap. 6, numerical properties of various CESE schemes are analysed, along with comparisons to other numerical schemes. Chapters 7–9 provide an overview of the applications of the CESE method, most of which are quite relevant to the aerospace engineering. Finally, a summary and an outlook of the CESE method are given in Chap. 10.

References

1. Toro, E. F. (2009). *Riemann solvers and numerical methods for fluid dynamics: A practical introduction.* Springer.
2. Kitamura, K. (2020). *Advancement of shock capturing computational fluid dynamics methods: numerical flux functions in finite volume method.* Springer Nature.
3. Chang, S. C. & To, W. M. (1991). *A new numerical framework for solving conservation laws: the method of space-time conservation element and solution element.* National Aeronautics and Space Administration.
4. Chang, S. C. (1995). The method of space-time conservation element and solution element—A new approach for solving the Navier-Stokes and Euler equations. *Journal of Computational Physics, 119*(2), 295–324.
5. Loh, C. Y., & Zaman, K. (2002). Numerical investigation of transonic resonance with a convergent-divergent nozzle. *AIAA Journal, 40*(12), 2393–2401.
6. Zaman, K., Dahl, M., Bencic, T., & Loh, C. (2002). Investigation of a 'transonic resonance' with convergent–divergent nozzles. *Journal of Fluid Mechanics, 463*, 313–343.
7. Cook, G., Zhang, Z. C. & Im, K. S. (2013). Applications of the CESE method in LS-DYNA. In *21st AIAA Computational Fluid Dynamics Conference.*
8. Chang, C. L. (2006). Time-accurate, unstructured mesh Navier-Stokes computations with the CESE method. In *42nd AIAA/ASME/SAE/ASEE Joint Propulsion Conference & Exhibit.*
9. Chang, C. L. (2007). Three-dimensional Navier-Stokes calculations using the modified space-time CESE method. In: *43rd AIAA/ASME/SAE/ASEE Joint Propulsion Conference & Exhibit.*

Chapter 2
Non-dissipative Core Scheme of CESE Method

This chapter is devoted to demonstrating the basic ideas in the CESE method. These ideas include the adoption of a space–time integral form of governing equations as the starting point of scheme construction, as well as the introduction of conservation element (CE) and solution element (SE) in the discretization of space–time domain. Then, the non-dissipative core scheme of the CESE method will be presented in detail.

2.1 Space–Time Integral Form of Governing Equations

Governing equations for a specific physical problem can be expressed in both differential and integral forms. Usually, a differential equation or a set of differential equations can clearly describe the evolution of a physical system. The differential forms of governing equations are prevalent in textbooks and theoretical research works, due to their compactness in writing and the maturity in the mathematical analysis on differential equations.

In the area of numerical simulation, the finite difference method is directly applied to the differential equations, while the finite volume method and finite element method require the governing equations to be cast into integral forms before numerical treatments. In fact, the performance of a numerical method will be influenced by the form of equations that is used in the numerical method, even though different forms of governing equations can be mathematically equivalent. As will be shown in this book, a major feature of the CESE method is the adoption of a space–time integral form of the governing equations, in which time and space are treated on the same footing.

For illustrative purposes, we consider the compressible Euler equations for 2D planar flows, which can be written in a differential form as

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x} + \frac{\partial \mathbf{G}}{\partial y} = \mathbf{0}, \tag{2.1}$$

where U denotes the vector of conserved variables such that

$$\mathbf{U} = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ E \end{pmatrix}, \tag{2.2}$$

where F and G are the inviscid fluxes of the form

$$\mathbf{F} = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho u v \\ (E + p)u \end{pmatrix}, \quad \mathbf{G} = \begin{pmatrix} \rho v \\ \rho u v \\ \rho v^2 + p \\ (E + p)v \end{pmatrix} \tag{2.3}$$

In Eqs. (2.2) and (2.3), ρ is the density of the fluid, u and v are the x-component and y-component of the flow velocity, respectively, p is the static pressure, and E is the total energy per unit volume of the fluid. Note that once the equation of state (EOS) is provided, Eq. (2.1) becomes a closed set of equations. Here, flux vectors F and G are regarded as functions of conserved vector U. A set of equations in the form of Eq. (2.1) is called a system of conservation laws.

By using the gradient operator in 2D physical space

$$\nabla \equiv \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y} \right), \tag{2.4}$$

Equation (2.1) can also be written in a divergence form

$$\frac{\partial \mathbf{U}}{\partial t} + \nabla \cdot \mathbf{H} = \mathbf{0}, \tag{2.5}$$

where H is a matrix composed of spatial flux vectors F and G:

$$\mathbf{H} = (\mathbf{F}, \mathbf{G}) \tag{2.6}$$

Therefore, the divergence form of Euler equation states that at time t and point (x, y), the temporal rate of change of conserved quantities plus the divergence of the spatial flux of these quantities must be zero.

Consider a control volume denoted by V in 2D physical space, as shown in Fig. 2.1, with the surface $S(V)$ and the unit outward normal vector $\boldsymbol{n}$ on the surface. Integrating Eq. (2.5) over the control volume V and applying the Gauss's theorem, one obtains

Fig. 2.1 Schematic of a
control volume in
two-dimensional physical
space
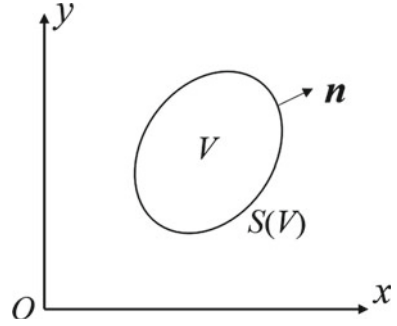
$$\frac{\partial}{\partial t}\left(\int_V \mathbf{U}\, dV\right) + \oint_{S(V)} \mathbf{H}\cdot\mathbf{n}\, dS = \mathbf{0},\tag{2.7}$$

which can be interpreted as follows: the temporal rate of change of conserved quantities within the volume V must equal to the net inward flow rate of these quantities through the surface $S(V)$. This integral form of Euler Eq. (2.7) is the starting point of the well-known finite volume method (FVM) in computational fluid dynamics.

With a unified treatment of time and physical space, we can define x, y, and t as coordinates in a three-dimensional Euclidean space, called the space–time domain. On this basis, the definition of gradient operator can be extended as

$$\nabla \equiv \left(\frac{\partial}{\partial x},\ \frac{\partial}{\partial y},\ \frac{\partial}{\partial t}\right)\tag{2.8}$$

and Eq. (2.1) can be written in a divergence-free form as

$$\nabla\cdot\mathbf{h} = \mathbf{0},\tag{2.9}$$

where the matrix $\mathbf{h}$ is composed of both flux vectors and the conserved vector $\mathbf{U}$:

$$\mathbf{h} = (\mathbf{F},\mathbf{G},\mathbf{U})\tag{2.10}$$

By applying Gauss's theorem to the integration of Eq. (2.9) over an arbitrary control volume V in the 3D space–time domain as shown in Fig. 2.2, the Euler equation is eventually formulated as

$$\oint_{S(V)} \mathbf{h}\cdot\mathbf{n}\, dS = \mathbf{0}\tag{2.11}$$

which means the balancing of flux is enforced for a space–time control volume. Clearly, this equation gives a direct description of the space–time conservation of mass, momentum, and energy in fluid flows, which faithfully preserves the original physical laws.

Fig. 2.2 Schematic of a control volume in three-dimensional space–time domain
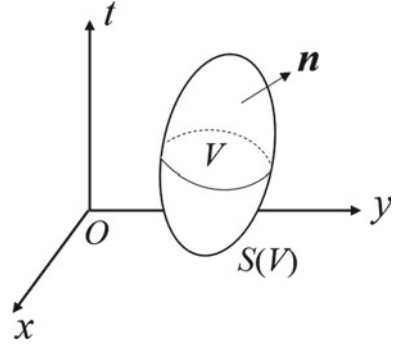
Equation (2.11) provides an example of the space–time integral form of governing equations for physical problems. It is worth noting that the space–time integral form and the traditional integral form like Eq. (2.7) state the same physics and must be equivalent to each other in mathematics, but they can lead to different numerical schemes. In contrast to the finite volume method, the CESE method numerically implements the space–time integral form of governing equations, emphasizing on the conservative property in the unity of time and physical space.

2.2 Definitions of Conservation Elements and Solution Elements

The CESE method starts with discretizing the space–time domain which is relevant to the computation. The discretization procedure generates the mesh for the physical space and determines the time-step size for the time-marching algorithm, similar to most CFD methods. The features of the CESE discretization include the construction of control volumes for the space–time integral form of the governing equation, the arrangement of the solution points, and the selection of the unknown variables to be calculated and stored at each solution point. All these features can be demonstrated by introducing two special concepts: the conservation element (CE) and the solution element (SE), which coin the algorithm accordingly.

To explain the CESE method in a simple way, we begin with the application of the CESE method to a 1D scalar conservation law of the form, without loss of generality

$$\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = 0. \tag{2.12}$$

With a uniform division of the 1D physical space and a constant time step, the space–time mesh is constructed as a 2D x–t plane as shown in Fig. 2.3 (solid lines). The spatial coordinate of the j-th mesh node is denoted by x_j, with the corresponding cell centre position at $x_{j+1/2} = (x_j + x_{j+1})/2$, and the cell size of $\Delta x = x_{j+1} - x_j$. In this
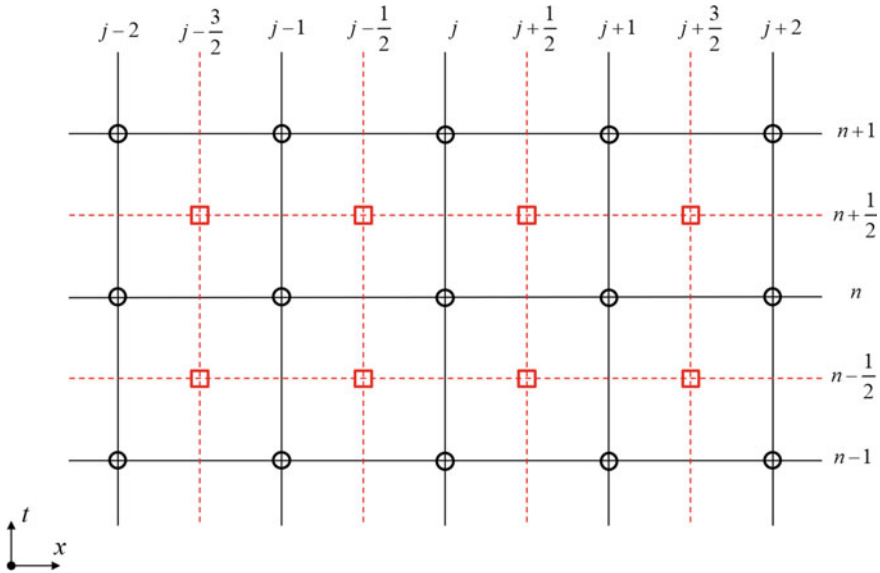
Fig. 2.3 Mesh and arrangement of solution points for 1D CESE scheme

time-marching algorithm, each step between t_{n-1} and t_n consists of two half steps: $[t_{n-1}, t_{n-1/2}]$ and $[t_{n-1/2}, t_n]$, with the time-step size of $\Delta t = t_n - t_{n-1}$.

The unknown function $u(x, t)$ is represented by the discrete values of u at a set of specific space–time points, called solution points which are shown as mesh nodes in Fig. 2.3—black circles for integer time levels $\{t_0, t_1, ..., t_n, ...\}$ and red squares (cell centres) for half-integer time levels $\{t_{1/2}, t_{3/2}, ..., t_{n+1/2},...\}$. In other words, a staggered mesh is used at each intermediate time level. In the CESE scheme, both the unknown variable u and its spatial derivative u_x will be calculated and stored at each solution point (e.g. point (j, n)):

$$u_j^n \equiv u(x_j, t_n), \ (u_x)_j^n \equiv \frac{\partial u}{\partial x}(x_j, t_n) \tag{2.13}$$

The paths of information flow in a single CESE time step is illustrated in the schematic diagram in Fig. 2.4. As seen, a highly compact stencil in space and time is used in the CESE scheme. If a half time step it is treated as the basic iteration, the halfwidth of the symmetric stencil is then $\Delta x/2$ and unknowns at point (j, n) only depend on the data stored at $(j - 1/2, n - 1/2)$ and $(j + 1/2, n - 1/2)$.

Based on the space–time mesh shown in Fig. 2.3, a set of small space–time elements, named conservation elements (CE), can be constructed, as demonstrated in Fig. 2.5. A CE is assigned to each solution point. For example, $(CE)_j^n$ is denoted as the CE for point (j, n), which is the rectangle with four vertices at the points $(j - 1/2, n - 1/2)$, $(j + 1/2, n - 1/2)$, $(j + 1/2, n)$, and $(j - 1/2, n)$. Apparently, the CEs cover the whole space–time domain without overlap. It is noteworthy that
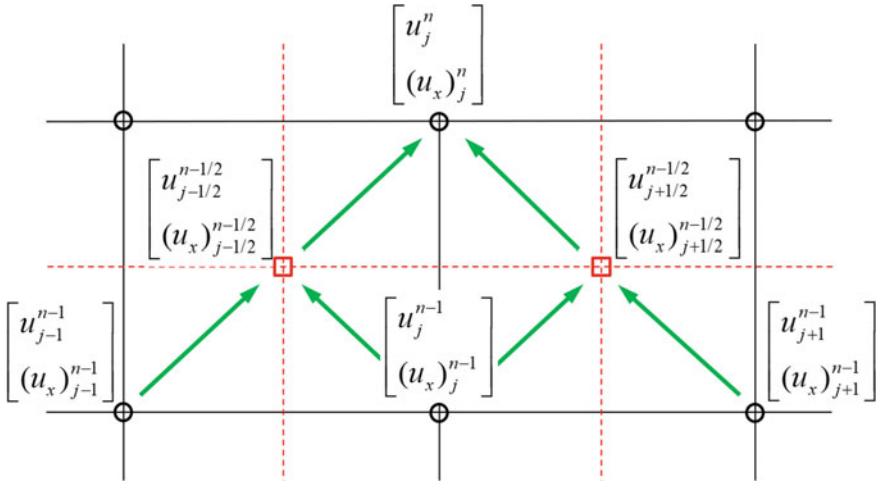
Fig. 2.4 The information flow in one CESE time step with time-marching variables

the arrangement of CEs is staggered in two successive half time steps. Accordingly, the space–time integral form of Eq. (2.12) is numerically implemented within each conservation element and discrete equations for unknowns are established.

When performing the space–time integration of Eq. (2.12) over a CE, an important question arises: how to evaluate u and f along the boundary of the CE. This leads to the
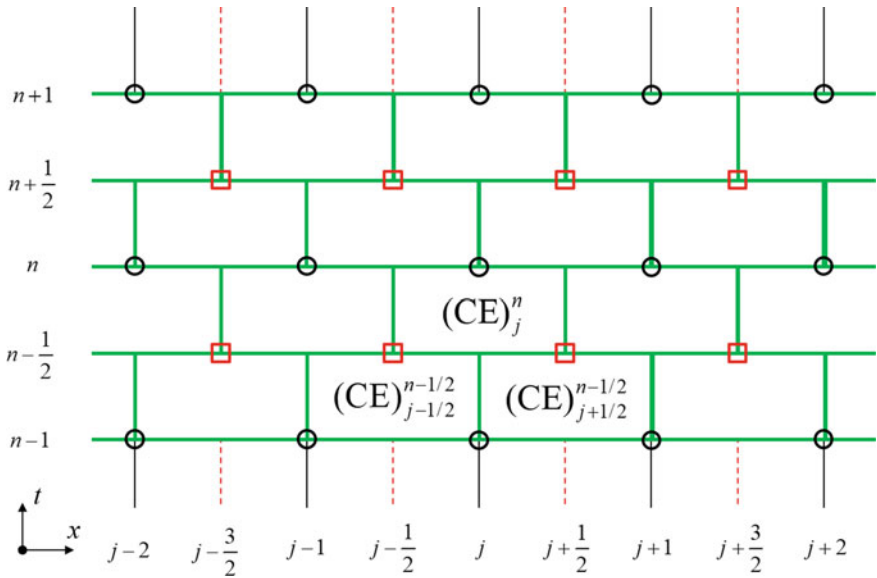
Fig. 2.5 Schematic of conservation elements (CEs) arrangement

Fig. 2.6 Schematic of the
solution element (SE)
associated with solution
point (j, n)

$(SE)_j^n$

$\Delta t/2$

$\Delta x/2 \qquad \Delta x/2$

(j, n)

t

$\Delta t/2$

x
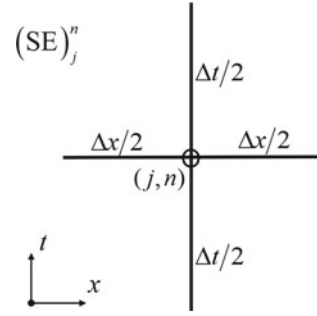
introduction of a solution element (SE) for each solution point, as shown in Fig. 2.6.
$(SE)_j^n$ composed of two line segments bisecting each other at point (j, n), forming
a cross with four endpoints $(j + 1/2, n)$, $(j, n + 1/2)$, $(j - 1/2, n)$, and $(j, n - 1/2)$.
For half-integer points, SEs can be defined in the same way. Note that conservation
element $(CE)_j^n$ is bounded by line segments belonging to three solution elements:
$(SE)_j^n$, $(SE)_{j-1/2}^{n-1/2}$, and $(SE)_{j+1/2}^{n-1/2}$. In the CESE scheme, the functions $u(x, t)$ and $f(x, t)$ are assumed to be linear along each line segment of a SE and are approximated by
first-order Taylor expansions about the centre of the SE. Specifically, in the solution
element $(SE)_j^n$, u and f are constructed as

$$u(x, t) = u_j^n + (u_x)_j^n(x - x_j) + (u_t)_j^n(t - t_n), \ (x, t) \in (SE)_j^n \qquad (2.14)$$

$$f(x, t) = f_j^n + (f_x)_j^n(x - x_j) + (f_t)_j^n(t - t_n), \ (x, t) \in (SE)_j^n \qquad (2.15)$$

where u_t and f_t are the temporal derivatives of u and f, respectively.

2.3   Non-dissipative Core Scheme: a Scheme

In this section, we present a non-dissipative CESE scheme, named the *a* scheme, to
solve the 1D scalar conservation law, Eq. (2.12). A space–time flux vector is defined
as

$$\mathbf{h} = (f, u) \qquad (2.16)$$

where f and u can be regarded as the components of flux vector $\boldsymbol{h}$ in x-direction and
t-direction, respectively. According to this definition, Eq. (2.12) can be converted into
the space–time integral form of Eq. (2.11) by following the procedure in Sect. 2.1.

Consider a half step marching from time level $n-1/2$ to time level n. At the solution
point (j, n), two independent unknowns u_j^n and $(u_x)_j^n$, need to be calculated simul-
taneously. Therefore, two algebraic equations need to be formulated by discretizing

Fig. 2.7 Definition of
sub-CEs: CE$^-$ and CE$^+$, and
the associated SEs
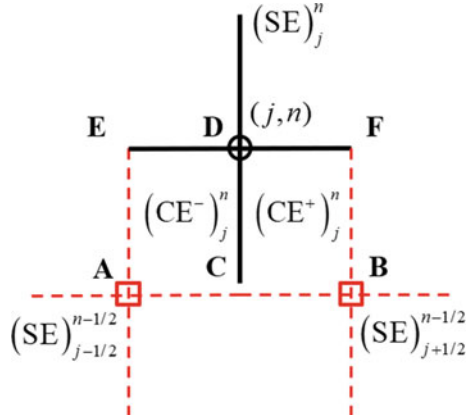
the integral conservation law. For this purpose, the conservation element $(CE)_j^n$ is
split into two sub-elements: $(CE^-)_j^n$ and $(CE^+)_j^n$. As shown in Fig. 2.7, $(CE^-)_j^n$ and
$(CE^+)_j^n$ are the rectangle ACDE and the rectangle CBFD, respectively. Each edge of
these rectangles belongs to one of the three SEs associated with $(CE)_j^n$.

Next, the space–time integral conservation law is implemented over each of the
sub-CEs. Let the control volume V in Eq. (2.11) be $(CE^-)_j^n$ and $(CE^+)_j^n$ in turn, one
can obtain

$$\oint_{S(CE^-);} \mathbf{h} \cdot \mathbf{n} \mathrm{dS} = 0 \tag{2.17}$$

and

$$\oint_{S(CE^+)_j^n} \mathbf{h} \cdot \mathbf{n} \, \mathrm{d}S = 0, \tag{2.18}$$

where $\mathbf{h}$ is the space–time flux vector expressed in Eq. (2.16) and $\mathbf{n}$ is the unit outward
normal vector on the boundary of $(CE^-)_j^n$ or $(CE^+)_j^n$.

As depicted in Fig. 2.8, the average values of $u(x, t)$ on line segments DE, DF,
AC, and BC are denoted by U_L^*, U_R^*, U_L, and U_R, respectively, while the average
values of $f(x, t)$ on line segments AE, BF, and CD are presented by F_L, F_R, and F_C,
respectively. Notably, CD is the interface between two sub-CEs. With this notation,
integral Eqs. (2.17) and (2.18) can be expressed as

$$U_L^* \frac{\Delta x}{2} = U_L \frac{\Delta x}{2} + (F_L - F_C) \frac{\Delta t}{2}, \tag{2.19}$$

$$U_R^* \frac{\Delta x}{2} = U_R \frac{\Delta x}{2} + (F_C - F_R) \frac{\Delta t}{2}, \tag{2.20}$$

which explicitly state the balance of space–time flux in each sub-CE.
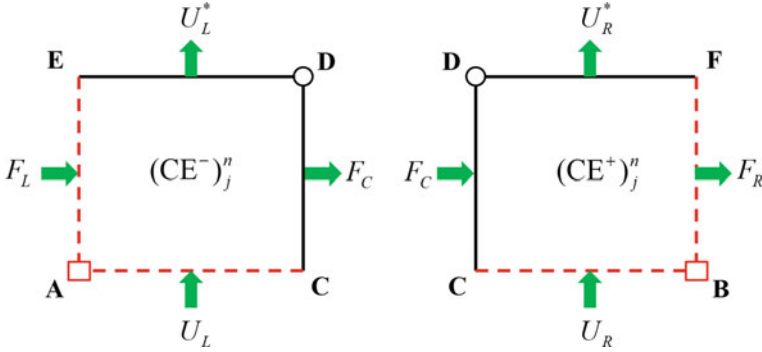
Fig. 2.8 Space–time flux through the each boundary of sub-CEs

To proceed with the scheme construction, the time-marching variables (i.e. u and u_x) at the solution points will be linked with U_L^*, U_R^*, U_L, U_R, F_L, F_R, and F_C in Eqs. (2.19) and (2.20), using the concept of the solution element. Recall that u and f can be calculated by first-order Taylor expansions about the solution point, by assuming them to be linear functions of x and t inside each individual SE. Since line segments AC and AE belong to $(SE)_{j-1/2}^{n-1/2}$, U_L and F_L can be obtained in terms of the known information stored at $(j - 1/2, n - 1/2)$, i.e., point A in Fig. 2.8. Performing Taylor expansions in $(SE)_{j-1/2}^{n-1/2}$ yields

$$U_L = u_{j-1/2}^{n-1/2} + \frac{\Delta x}{4}(u_x)_{j-1/2}^{n-1/2}, \qquad (2.21)$$

$$F_L = f_{j-1/2}^{n-1/2} + \frac{\Delta t}{4}(f_t)_{j-1/2}^{n-1/2} \qquad (2.22)$$

Similarly, because BC and BF belong to $(SE)_{j+1/2}^{n-1/2}$, U_R and F_R can be obtained by the Taylor expansions about solution point $(j + 1/2, n - 1/2)$:

$$U_R = u_{j+1/2}^{n-1/2} - \frac{\Delta x}{4}(u_x)_{j+1/2}^{n-1/2} \qquad (2.23)$$

$$F_R = f_{j+1/2}^{n-1/2} + \frac{\Delta t}{4}(f_t)_{j+1/2}^{n-1/2} \qquad (2.24)$$

In Eqs. (2.21)–(2.24), u and u_x at time level $n - 1/2$ are the known values. In addition, since $f = f(u)$ is a prescribed function in the conservation law,

$$f_{j\pm1/2}^{n-1/2} = f(u_{j\pm1/2}^{n-1/2}) \qquad (2.25)$$

By applying the chain rule, the spatial derivative of f is described as

$$(f_x)_{j\pm1/2}^{n-1/2} = \left[(\partial f / \partial u)u_x\right]_{j\pm1/2}^{n-1/2} \tag{2.26}$$

The temporal derivative of u can be obtained using Eq. (2.12),

$$(u_t)_{j\pm1/2}^{n-1/2} = -(f_x)_{j\pm1/2}^{n-1/2} \tag{2.27}$$

By applying the chain rule again, the temporal derivative of f is derived as

$$(f_t)_{j\pm1/2}^{n-1/2} = \left[(\partial f / \partial u)u_t\right]_{j\pm1/2}^{n-1/2} \tag{2.28}$$

Accordingly, U_L, F_L, U_R, and F_R can be explicitly calculated using the above formulas.

The first-order Taylor expansion in $(SE)_j^n$ is used again to relate U_L^* and U_R^* to unknowns u_j^n and $(u_x)_j^n$ at time level n:

$$U_L^* = u_j^n - \frac{\Delta x}{4}(u_x)_j^n \tag{2.29}$$

$$U_R^* = u_j^n + \frac{\Delta x}{4}(u_x)_j^n \tag{2.30}$$

Substituting Eqs. (2.29) and (2.30) into (2.19) and (2.20) yields

$$u_j^n - \frac{\Delta x}{4}(u_x)_j^n = U_L + (F_L - F_C)\frac{\Delta t}{\Delta x}, \tag{2.31}$$

$$u_j^n + \frac{\Delta x}{4}(u_x)_j^n = U_R + (F_C - F_R)\frac{\Delta t}{\Delta x} \tag{2.32}$$

Adding Eqs. (2.31)–(2.32), one can derive

$$u_j^n = \frac{1}{2}(U_L + U_R) + \frac{\Delta t}{2\Delta x}(F_L - F_R) \tag{2.33}$$

and subtracting Eqs. (2.31) from (2.32), one has

$$\frac{\Delta x}{4}(u_x)_j^n = \frac{1}{2}(U_R - U_L) + \frac{\Delta t}{2\Delta x}(2F_C - F_L - F_R) \tag{2.34}$$

Up to now, Eq. (2.33) is presented as an explicit time-marching formula for u_j^n, but $(u_x)_j^n$ still cannot be directly calculated by Eq. (2.34). The reason is because F_C, which presents the average value of flux f through the interface CD, has not been addressed. In the original CESE scheme, F_C is provided by the Taylor expansion in $(SE)_j^n$, i.e.,

$$F_C = f_j^n - \frac{\Delta t}{4}(f_t)_j^n \tag{2.35}$$

With a similar procedure to Eqs. (2.25)–(2.28), f_j^n and $(f_t)_j^n$ can be expressed in terms of u_j^n and $(u_x)_j^n$. Since u_j^n is determined by Eq. (2.33), a time-marching formula for the only unknown in Eq. (2.34), $(u_x)_j^n$, can eventually be derived.

Note that a complete CESE time step consists of two half steps: a half step marching from nodes to centres and another half step marching from centres to nodes. The marching schemes for both half steps are identical, except for the indexes of the solution points. Usually, the initial conditions of u and u_x are specified at the initial time, $t_0 = 0$ and the corresponding boundary conditions for u and u_x need to be implemented at the boundaries of x.

The above CESE scheme is referred to as the a scheme [1] in the literature. Define a solution vector

$$\mathbf{q}_j^n = \begin{bmatrix} u_j^n \\ \dfrac{\Delta x}{4}(u_x)_j^n \end{bmatrix}, \tag{2.36}$$

and let function $f(u)$ be a linear one:

$$f = au, \ a \text{ is a constant} \tag{2.37}$$

The a scheme, which is mainly expressed by Eqs. (2.33) and (2.34), can then be rewritten in a matrix form:

$$\mathbf{q}_j^n = \mathbf{Q}_L \mathbf{q}_{j-1/2}^{n-1/2} + \mathbf{Q}_R \mathbf{q}_{j+1/2}^{n-1/2} \tag{2.38}$$

where the coefficient matrices are

$$\mathbf{Q}_L = \frac{1}{2}\begin{bmatrix} 1+v & 1-v^2 \\ -1 & -1+v \end{bmatrix}, \ \mathbf{Q}_R = \frac{1}{2}\begin{bmatrix} 1-v & -1+v^2 \\ 1 & -1-v \end{bmatrix} \tag{2.39}$$

and v is a constant, which is defined as

$$v \equiv a\frac{\Delta t}{\Delta x} \tag{2.40}$$

Analogously, we apply Eq. (2.38) to the half-step marching from time level $n-1$ to time level $n-1/2$, and obtain

$$\mathbf{q}_{j-1/2}^{n-1/2} = \mathbf{Q}_L \mathbf{q}_{j-1}^{n-1} + \mathbf{Q}_R \mathbf{q}_j^{n-1}, \tag{2.41}$$

$$\mathbf{q}_{j+1/2}^{n-1/2} = \mathbf{Q}_L \mathbf{q}_j^{n-1} + \mathbf{Q}_R \mathbf{q}_{j+1}^{n-1}, \tag{2.42}$$

After substituting Eqs. (2.41) and (2.42) into (2.38), a complete time step in the a scheme can be formulated as.

$$\mathbf{q}_j^n = (\mathbf{Q}_L)^2 \mathbf{q}_{j-1}^{n-1} + (\mathbf{Q}_L \mathbf{Q}_R + \mathbf{Q}_R \mathbf{Q}_L) \mathbf{q}_j^{n-1} + (\mathbf{Q}_R)^2 \mathbf{q}_{j+1}^{n-1}. \qquad (2.43)$$

In fact, the numerical dissipation, dispersion, and stability of different CESE schemes can be conveniently analysed using their matrix forms like Eq. (2.43).

An unique feature of this a scheme is its use of Taylor expansion in the inverse time direction, to relate the interface flux F_C to the marching variables (see Eq. (2.35)). Such a treatment makes the a scheme the space–time inversion invariant and renders this scheme non-dissipative [2]. From a historical perspective, the a scheme forms the non-dissipative core of subsequent CESE-family schemes.

The a scheme described in this chapter is restricted to 1D scalar problems and uniform meshes. First-order Taylor expansions are used in each solution element in the present scheme. Since the a scheme is reversible in time, it cannot be used for practical applications, considering that real physical processes are irreversible in time to respect the second law of thermodynamics. Accordingly, substantial improvements have been made to develop this a scheme into a viable numerical method. In Chap. 3, we will describe different approaches to introducing necessary numerical dissipation into the CESE scheme, which result in two categories of CESE schemes: the central CESE schemes and the upwind CESE schemes. In Chap. 4, we will describe the extensions of CESE schemes to multi-dimensional Cartesian and unstructured computational meshes. In Chap. 5, high-order Taylor expansions in solution elements will be used to achieve high-order accuracy in both space and time, without enlarging the spatial stencil or adding time-integration stages.

References

1. Chang, S. C. (1995). The method of space-time conservation element and solution element—A new approach for solving the Navier-Stokes and Euler equations. *Journal of computational Physics, 119*(2), 295–324.
2. Chang, S. C. (2006). On space-time inversion invariance and its relation to non-dissipatedness of a CESE core scheme. In *42nd AIAA/ASME/SAE/ASEE Joint Propulsion Conference & Exhibit*.

Chapter 3
CESE Schemes with Numerical Dissipation

As depicted in Chap. 2, the interface between the two sub-CEs (CD in Fig. 2.7), belongs to the SE of (j, n). The flux F_C needs to be calculated through the Taylor expansion at point (j, n) toward the inverse time direction. As a result, the a scheme is reversible. This violates the second law of thermodynamics. Thus, the non-dissipative core suffers from the unphysical oscillations for practical applications. This chapter will present the improvements based on the non-dissipative core by introducing necessary numerical dissipation. The resulting dissipative CESE schemes can be categorized into the central CESE schemes and the upwind CESE schemes. It is convenient to begin with the most widely spread CESE scheme, the a–α scheme.

3.1 a–α Scheme

Similar to the a scheme described in Chap. 2, the a–α scheme is also a central scheme because of the fact that techniques to calculate upwind numerical fluxes are not used. Again, we consider the 1D scalar conservation law of the form

$$\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = 0 \tag{3.1}$$

To solve Eq. (3.1) with the a–α scheme, the discretization and the definitions of CE and SE keep the same as those for the a scheme, which can be found in Sect. 2.2. Basically, the solution algorithm follows that of the a scheme (see Sect. 2.3), except for Eq. (2.35) for the flux F_C. It is worth noting that abandoning the calculation of F_C does not affect the correctness of Eq. (2.33) for updating u_j^n. Actually, Eq. (2.33) is purely a result of conservation in the entire conservation element $(CE)_j^n$, irrespective of the flux F_C. Therefore, the formula for updating the node value of the unknown u remains unchanged, which is written as

$$u_j^n = \frac{1}{2}(U_L + U_R) + \frac{\Delta t}{2\Delta x}(F_L - F_R) \tag{3.2}$$

The calculation of U_L, F_L, U_R, and F_R are explicit and has been presented in Sect. 2.3, and then u_j^n can be evaluated using Eq. (3.2).

The remaining part of the a–α scheme is to update the spatial derivative $(u_x)_j^n$ in a different way from the procedure of the a scheme. It is in this step where the necessary numerical dissipation is added into the a–α scheme. First, two different estimations for $(u_x)_j^n$ can be obtained:

$$\left(u_x^-\right)_j^n = \frac{u_j^n - \left[u_{j-1/2}^{n-1/2} + (\Delta t/2)(u_t)_{j-1/2}^{n-1/2}\right]}{\Delta x/2}, \tag{3.3}$$

$$\left(u_x^+\right)_j^n = \frac{\left[u_{j+1/2}^{n-1/2} + (\Delta t/2)(u_t)_{j+1/2}^{n-1/2}\right] - u_j^n}{\Delta x/2} \tag{3.4}$$

In these equations, the temporal derivatives u_t at time level $n-1/2$ can be readily obtained using Eqs. (2.27) and (2.26). Next, $(u_x)_j^n$ is taken as a weighted average of $(u_x^-)_j^n$ and $(u_x^+)_j^n$:

$$(u_x)_j^n = W\left((u_x^-)_j^n, \ (u_x^+)_j^n, \ \alpha\right) \tag{3.5}$$

where W is a weighted average function with an adjustable parameter α ($\alpha \geq 0$, the commonly used values are $\alpha = 0$, 1 and 2), expressed as

$$W\left(x^-, \ x^+, \ \alpha\right) = \frac{\left|x^+\right|^\alpha x^- + \left|x^-\right|^\alpha x^+}{\left|x^+\right|^\alpha + \left|x^-\right|^\alpha}, \tag{3.6}$$

The role of this weighted average function is similar to the slope limiter functions in upwind schemes, and it proves to be effective in suppressing the spurious oscillations near a discontinuity.

The above a–α scheme has a clear and simple logic. However, the dissipation of the a–α scheme increases dramatically as the Courant–Friedrichs–Lewy (CFL) number (the parameter defined by Eq. (2.40), $\nu = a\Delta t/\Delta x$) approaches zero. In practice, for small values of ν, the discontinuities in the solution can be smeared out. Therefore, the accuracy of the a–α scheme is considered to be CFL-number sensitive.

To demonstrate this shortcoming, we consider $f = au$ (a is a constant) in Eq. (3.1), and take $\alpha = 0$ in Eq. (3.6). In this case, the weighted average function is reduced to a simple arithmetic averaging, and Eq. (3.5) can be written as

$$(u_x)_j^n = \frac{\left[u_{j+1/2}^{n-1/2} + (\Delta t/2)(u_t)_{j+1/2}^{n-1/2}\right] - \left[u_{j-1/2}^{n-1/2} + (\Delta t/2)(u_t)_{j-1/2}^{n-1/2}\right]}{\Delta x}, \tag{3.7}$$

which means a central difference approximation at time level n is used to update the spatial derivative $(u_x)_j^n$. Recall that in Sect. 2.3 we have shown that a CESE scheme applied to the linear scalar convection equation can be written in a matrix form

$$\mathbf{q}_j^n = \mathbf{Q}_L \mathbf{q}_{j-1/2}^{n-1/2} + \mathbf{Q}_R \mathbf{q}_{j+1/2}^{n-1/2} \tag{3.8}$$

for a half step or

$$\mathbf{q}_j^n = (\mathbf{Q}_L)^2 \mathbf{q}_{j-1}^{n-1} + (\mathbf{Q}_L\mathbf{Q}_R + \mathbf{Q}_R\mathbf{Q}_L)\mathbf{q}_j^{n-1} + (\mathbf{Q}_R)^2 \mathbf{q}_{j+1}^{n-1} \tag{3.9}$$

for a complete step. Here, the solution vector $\boldsymbol{q}$ is defined by Eq. (2.36) and the matrices $\boldsymbol{Q}_L$ and $\boldsymbol{Q}_R$ are functions of the CFL number $\nu = a\Delta t/\Delta x$. By combining Eqs. (3.2) and (3.7), the a–α scheme with $\alpha = 0$ can also be cast into the form of Eq. (3.9) with $\boldsymbol{Q}_L$ and $\boldsymbol{Q}_R$ as follows:

$$\mathbf{Q}_L = \frac{1}{2}\begin{bmatrix} 1+\nu & 1-\nu^2 \\ -1/2 & \nu \end{bmatrix}, \ \mathbf{Q}_R = \frac{1}{2}\begin{bmatrix} 1-\nu & -1+\nu^2 \\ 1/2 & -\nu \end{bmatrix} \tag{3.10}$$

Now, consider a limiting case of $\Delta t = 0$ (but $\Delta x > 0$), which leads to $\nu = 0$. It can be shown that Eq. (3.9) leads to

$$\mathbf{q}_j^n = \frac{1}{4}\begin{bmatrix} 1/2 & 1 \\ -1/2 & -1/2 \end{bmatrix}\mathbf{q}_{j-1}^{n-1} + \frac{1}{4}\begin{bmatrix} 3 & 0 \\ 0 & 1 \end{bmatrix}\mathbf{q}_j^{n-1} + \frac{1}{4}\begin{bmatrix} 1/2 & -1 \\ 1/2 & -1/2 \end{bmatrix}\mathbf{q}_{j+1}^{n-1} \tag{3.11}$$

However, a reasonable time-marching scheme should guarantee that $\mathbf{q}_j^n = \mathbf{q}_j^{n-1}$ when Δt vanishes, which indicates that the a–α scheme suffers from numerical error when the CFL number is very small. Further numerical experiments show that, in practice, when the CFL number ν becomes smaller than 0.1, the excessive numerical dissipation of the a–α scheme can be remarkable.

3.2   Courant–Number–Insensitive Scheme

To overcome the shortcoming of the a–α scheme, a Courant–Number–Insensitive (CNI) scheme was constructed [1]. This improvement is based on two requirements: (1) the CNI scheme should reduce to the non-dissipative a scheme when the CFL number $\nu = 0$ and (2) the CNI scheme should resemble the a–α scheme when $|\nu| = 1$. In this section, the construction of the CNI scheme will be present, and then the relationship between the CNI scheme, the a scheme, and the a–α scheme will be shown.

First, the formula for updating the node value of the unknown u remains the same as Eq. (3.2), which has been used in a scheme and the a–α scheme. By substituting formulas for U_L, F_L, U_R, and F_R (Eqs. (2.21)–(2.28)) into Eq. (3.2), the explicit time

marching formula for u_j^n can be written as

$$u_j^n = \frac{1+v}{2} u_{j-1/2}^{n-1/2} + \frac{1-v^2}{8} \Delta x (u_x)_{j-1/2}^{n-1/2} + \frac{1-v}{2} u_{j+1/2}^{n-1/2} + \frac{v^2-1}{8} \Delta x (u_x)_{j+1/2}^{n-1/2}$$

(3.12)

Then, an algorithm for updating $(u_x)_j^n$ is devised. As shown in Fig. 3.1, the points $(j-1/2, n)$ and $(j+1/2, n)$ are denoted by V_- and V_+, respectively. The point M_- is the midpoint between $(j-1/2, n)$ and (j, n), and M_+ is the midpoint between $(j+1/2, n)$ and (j, n). We define two important points P_- and P_+, which are on the line segments V_-M_- and V_+M_+, respectively. The distances between $P_\pm$ and node (j, n) are marked in Fig. 3.1. It is clear that $P_\pm$ coincides with $V_\pm$ when $v = 1$, and they approach $M_\pm$ as $v \to 0$. The values of $u(x, t)$ at $P_\pm$ are estimated by Taylor expansion at points $(j \pm 1/2, n-1/2)$:

$$u(P_\pm) = u_{j\pm1/2}^{n-1/2} + \frac{\Delta t}{2} (u_t)_{j\pm1/2}^{n-1/2} \mp \frac{(1-|v|)\Delta x}{4} (u_x)_{j\pm1/2}^{n-1/2}$$

(3.13)

where $u_{j\pm1/2}^{n-1/2}$ and $(u_x)_{j\pm1/2}^{n-1/2}$ are known, while $(u_t)_{j\pm1/2}^{n-1/2}$ can be obtained using Eqs. (2.27) and (2.26).

Next, $u(P_-)$ and $u(P_+)$ are used to construct two different approximations of $(u_x)_j^n$

$$(\hat{u}_x^-)_j^n = \frac{u_j^n - u(P_-)}{(1+|v|)\Delta x/4}$$

(3.14)

and

$$(\hat{u}_x^+)_j^n = \frac{u(P_+) - u_j^n}{(1+|v|)\Delta x/4}$$

(3.15)
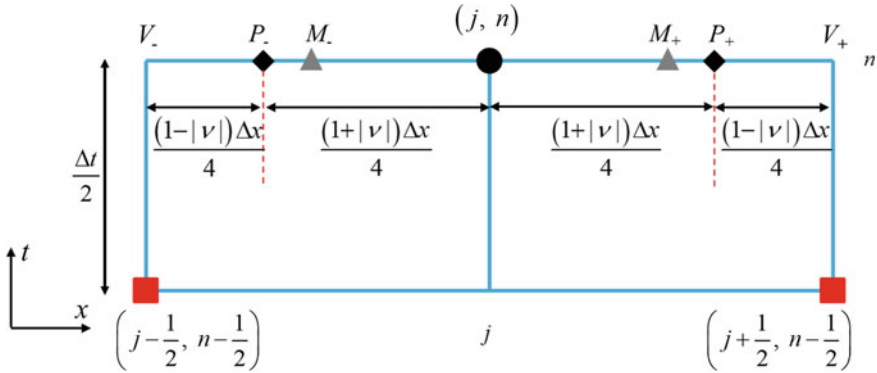
Fig. 3.1 Definition of points in CNI scheme

Finally, the CNI scheme use a weighted average of $(\hat{u}_x^-)_j^n$ and $(\hat{u}_x^+)_j^n$ as the updated value of $(u_x)_j^n$, in a way similar to the a–α scheme. However, the weighted average function in the a–α scheme, i.e. Eq. (3.6), is replaced with a more sophisticated one as

$$(u_x)_j^n = \frac{\left[1 + f(|\nu|)(s_-)_j^n\right](\hat{u}_x^+)_j^n + \left[1 + f(|\nu|)(s_+)_j^n\right](\hat{u}_x^-)_j^n}{2 + f(|\nu|)\left[(s_-)_j^n + (s_+)_j^n\right]},$$

(3.16)

where

$$(s_\pm)_j^n = \frac{\left|(\hat{u}_x^\pm)_j^n\right|}{\min\left(\left|(\hat{u}_x^-)_j^n\right|, \left|(\hat{u}_x^+)_j^n\right|\right)} - 1$$

(3.17)

and

$$f(|\nu|) = 0.5 \big/ |\nu|.$$

(3.18)

Hence, the overshoot phenomenon near discontinuities can be suppressed by the artificial dissipation, just like the a–α scheme. Moreover, the dissipation brought by Eq. (3.16) can be adjusted dynamically according to the CFL number ν.

When $\nu = 0$, we can show that the CNI scheme will reduce to the non-dissipative a scheme. Substituting $\nu = 0$ (which also means $\Delta t = 0$) into Eqs. (3.12) and (3.13) yields

$$u_j^n = \frac{1}{2}u_{j-1/2}^{n-1/2} + \frac{1}{8}\Delta x (u_x)_{j-1/2}^{n-1/2} + \frac{1}{2}u_{j+1/2}^{n-1/2} - \frac{1}{8}\Delta x (u_x)_{j+1/2}^{n-1/2}$$

(3.19)

and

$$u(P_\pm) = u_{j\pm1/2}^{n-1/2} \mp \frac{\Delta x}{4}(u_x)_{j\pm1/2}^{n-1/2}$$

(3.20)

From Eqs. (3.19) and (3.20), we can get

$$u_j^n = \frac{u(P_-) + u(P_+)}{2}$$

(3.21)

By using $\nu = 0$ and Eq. (3.21), it is readily shown that

$$(\hat{u}_x^+)_j^n = \frac{u(P_+) - u(P_-)}{\Delta x / 2} = (\hat{u}_x^-)_j^n$$

(3.22)

Thus, the weighted average of $(\hat{u}_x^-)_j^n$ and $(\hat{u}_x^+)_j^n$ is equal to $[u(P_+) - u(P_-)]/(\Delta x/2)$, which gives

$$(u_x)_j^n = \frac{1}{\Delta x/2}\left\{\left[u_{j+1/2}^{n-1/2} - \frac{\Delta x}{4}(u_x)_{j+1/2}^{n-1/2}\right] - \left[u_{j-1/2}^{n-1/2} + \frac{\Delta x}{4}(u_x)_{j-1/2}^{n-1/2}\right]\right\} \quad (3.23)$$

Recall the expression of the *a* scheme (see Eqs. (2.38) and (2.39) in Chap. 2) and let $\nu = 0$, we can reproduce Eqs. (3.19) and (3.23). Therefore, in the limiting case of $\nu = 0$, the CNI scheme and the *a* scheme are identical.

As for the other limiting case $|\nu| = 1$, Eqs. (3.13)–(3.15) lead to

$$\left(\hat{u}_x^-\right)_j^n = \frac{u_j^n - \left[u_{j-1/2}^{n-1/2} + (\Delta t/2)(u_t)_{j-1/2}^{n-1/2}\right]}{\Delta x/2}, \quad (3.24)$$

and

$$\left(\hat{u}_x^+\right)_j^n = \frac{\left[u_{j+1/2}^{n-1/2} + (\Delta t/2)(u_t)_{j+1/2}^{n-1/2}\right] - u_j^n}{\Delta x/2} \quad (3.25)$$

There is no difference between $(\hat{u}_x^{\pm})_j^n$ in the CNI scheme and $(u_x^{\pm})_j^n$ in the *a*–α scheme. Hence, the algorithms for updating $(u_x)_j^n$ in the CNI and the *a*–α schemes are basically the same when $|\nu| = 1$, except for the specific forms of the weighted average function.

Both the *a*–α scheme in Sect. 3.1 and the CNI scheme in this section belong to the central CESE schemes. Extensions of these schemes to 2-D and 3-D cases for various systems of conservation equations (e.g. Euler equations for compressible gas dynamics) are straightforward and have been well implemented.

3.3   Upwind CESE Scheme

The aforementioned central CESE schemes are used to solve nonlinear hyperbolic systems of conservation laws. However, they did not explicitly resort to the knowledge of characteristics or eigenvalues of the systems. As an alternative approach, the characteristic-based upwind CESE scheme proposed by Shen et al. [2], elegantly combines the basic ideas of the CESE method and the upwind numerical flux technique in the Godunov-type FVM method. The upwind CESE scheme is naturally CFL-number-insensitive, which means it does not suffer from the drawback of the *a*–α scheme presented in Sect. 3.1. For some challenging CFD problems such as the simulations of detonations and multiphase flows, the upwind CESE scheme captures discontinuities in flow fields with improved accuracy and robustness, especially for contact discontinuities (e.g., material interfaces).

3.3.1 Construction of Upwind CESE Scheme

For illustrative purposes, the 1D scalar conservation law (Eq. (3.1)) is considered again. Before introducing the upwind CESE algorithm, some elementary concepts about the space–time discretization need to be clarified. The same computational mesh and solution points as shown in Figs. 2.3 and 2.4 are adopted. The definition of CEs is also retained (see Fig. 2.5). Nevertheless, some modification is made to define the SEs, as sketched in Fig. 3.2. In comparison to the $(SE)_j^n$ in Fig. 2.6, the newly designed $(SE)_j^n$ no longer contains any part that allows for $t < t_n$. Apparently, the SEs pave the whole space–time domain without overlap. It is assuemed that $u(x, t)$ and $f(x, t)$ are piecewise linear. For instance, inside $(SE)_j^n$, they can be approximated by the first-order Taylor expansion at point (j, n). The boundaries of conservation element $(CE)_j^n$ belong to three solution elements: DE and DF belong to $(SE)_j^n$, AC and AE belong to $(SE)_{j-1/2}^{n-1/2}$, and BC and BF belong to $(SE)_{j+1/2}^{n-1/2}$. The interface between $(SE)_{j-1/2}^{n-1/2}$ and $(SE)_{j+1/2}^{n-1/2}$, i.e. line segment CD, splits the conservation element $(CE)_j^n$ into two sub-CEs: $(CE^-)_j^n$ and $(CE^+)_j^n$.

The construction of the upwind CESE scheme closely follows the framework of the non-dissipative core scheme (i.e., the *a* scheme in Sect. 2.3). Indeed, Eqs. (2.16)–(2.34) still hold for the upwind CESE scheme, and they can be derived by the same procedures as presented in Sect. 2.3. Therefore, just like the *a* scheme, the *a*–*α* scheme, and the CNI scheme, the formula to update u_j^n in the upwind CESE scheme remains the same as Eq. (3.2). However, for the purpose of updating $(u_x)_j^n$, the Eq. (2.34) will be utilized, which is the direct result of the space–time integral form of conservation law. For convenience, here we recall this equation:

$$\frac{\Delta x}{4}(u_x)_j^n = \frac{1}{2}(U_R - U_L) + \frac{\Delta t}{2\Delta x}(2F_C - F_L - F_R) \tag{3.26}$$

Note that this equation is actually discarded in the *a*–*α* and the CNI schemes, but both the *a* scheme and the upwind CESE scheme make full use of it. Furthermore,
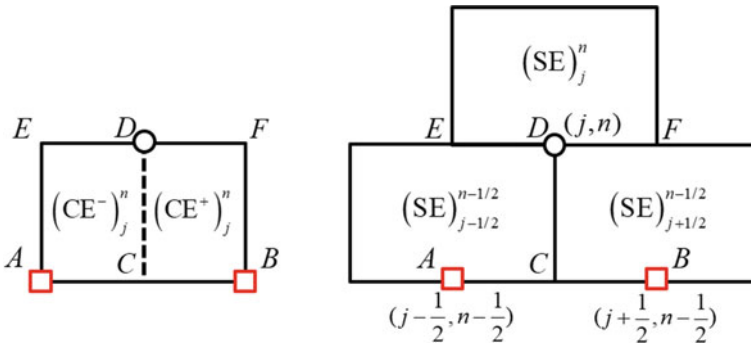
Fig. 3.2 Definitions of CE and SE for upwind CESE schemes

the key difference between the a scheme and the upwind CESE scheme lies in the treatment of F_C in Eq. (3.26), which denotes the average flux through the interface CD between $(CE^-)^n_j$ and $(CE^+)^n_j$ (see Fig. 3.2).

As can be inferred from the new definition of solution elements in Fig. 3.2, the operation used in the a scheme to link F_C with $(u_x)^n_j$ is forbidden in the upwind CESE scheme. Instead, we intend to obtain F_C from the known data at time level $n - 1/2$, and then insert F_C into Eq. (3.26) to get $(u_x)^n_j$. Toward this end, a local Riemann problem can be built at the midpoint of CD (denoted by $(j, n - 1/4)$), because CD is the interface between $(SE)^{n-1/2}_{j-1/2}$ and $(SE)^{n-1/2}_{j+1/2}$. To be specific, by Taylor expansion in $(SE)^{n-1/2}_{j-1/2}$, the left state at point $(j, n - 1/4)$ is evaluated as

$$(u_L)^{n-1/4}_j = U_L + \frac{\Delta x}{4}(u_x)^{n-1/2}_{j-1/2} + \frac{\Delta t}{4}(u_t)^{n-1/2}_{j-1/2} \qquad (3.27)$$

Meanwhile, Taylor expansion in $(SE)^{n-1/2}_{j+1/2}$ provides the right state at point $(j, n - 1/4)$:

$$(u_R)^{n-1/4}_j = U_R - \frac{\Delta x}{4}(u_x)^{n-1/2}_{j+1/2} + \frac{\Delta t}{4}(u_t)^{n-1/2}_{j+1/2} \qquad (3.28)$$

Recall that U_L and U_R are given by Eqs. (2.21) and (2.23). According to the definition of solution elements, the values of $(u_L)^{n-1/4}_j$ and $(u_R)^{n-1/4}_j$ are not necessarily equal to each other, and in general $(u_L)^{n-1/4}_j$ and $(u_R)^{n-1/4}_j$ form the discontinuous initial data on the left and right sides of the "diaphragm" CD.

Next, the average flux F_C through CD can be evaluated based on the local Riemann problem with initial data (3.27) and (3.28) as

$$F_C = \hat{F}\left((u_L)^{n-1/4}_j, (u_R)^{n-1/4}_j\right) \qquad (3.29)$$

where $\hat{F}$ stands for any upwind numerical flux solver, or loosely referred to as Riemann solver. Consequently, the special derivative $(u_x)^n_j$ can be updated explicitly by Eq. (3.26). It is worth noting that, in the upwind CESE scheme, the time marching formula Eq. (3.2) for u^n_j has no concern with the upwind procedure above, since flux F_C never appears in Eq. (3.2).

In case of strong discontinuities, using proper limiters for the derivatives in Eqs. (3.27) and (3.28) bocomes crucial to suppress spurious oscilations. When limiters are used, the reconstruction of $(u_L)^{n-1/4}_j$ and $(u_R)^{n-1/4}_j$ is written as

$$(u_L)^{n-1/4}_j = U_L + \frac{\Delta x}{4}u^L_x + \frac{\Delta t}{4}u^L_t, \qquad (3.30)$$

$$(u_R)^{n-1/4}_j = U_R - \frac{\Delta x}{4}u^R_x + \frac{\Delta t}{4}u^R_t, \qquad (3.31)$$

where u_x^L, u_x^R, u_t^L, and u_t^R are the limited derivatives. In this book, the weighted biased averaging procedure limiter (WBAP-L2) [3] is adopted. The limited slopes (spatial derivatives) are

$$u_x^L = (u_x)_{j-1/2}^{n-1/2} W(1, \theta_1^L, \theta_2^L), \ \theta_1^L = \frac{(U_R - U_L)/(\Delta x/2)}{(u_x)_{j-1/2}^{n-1/2}}, \theta_2^L = \frac{(u_x)_{j+1/2}^{n-1/2}}{(u_x)_{j-1/2}^{n-1/2}},$$
(3.32)

$$u_x^R = (u_x)_{j+1/2}^{n-1/2} W(1, \theta_1^R, \theta_2^R), \ \theta_1^R = \frac{(U_R - U_L)/(\Delta x/2)}{(u_x)_{j+1/2}^{n-1/2}}, \theta_2^R = \frac{(u_x)_{j-1/2}^{n-1/2}}{(u_x)_{j+1/2}^{n-1/2}},$$
(3.33)

where the limiter function is

$$W(1, \theta_1, \theta_2) = \begin{cases} \frac{5+1/\theta_1+1/\theta_2}{5+1/\theta_1^2+1/\theta_2^2}, & \text{if } \theta_1 \text{ and } \theta_2 > 0 \\ 0, & \text{else} \end{cases}$$
(3.34)

Once the limited slopes are obtained, the limited temporal derivatives can be calculated by the chain rule and the conservation equation itself, i.e.,

$$u_t^L = -u_x^L \left. \frac{\partial f}{\partial u} \right|_{u=U_L}, u_t^R = -u_x^R \left. \frac{\partial f}{\partial u} \right|_{u=U_R}$$
(3.35)

3.3.2 Scheme for Linear Scalar Convection Equation

Let $f = au$, then Eq. (3.1) becomes the linear scalar convection equation. Without loss of generality, a is assumed to be a positive constant. For this simple situation, the exact solution of the local Riemann problem with initial data $(u_L)_j^{n-1/4}$ and $(u_R)_j^{n-1/4}$ can be readily obtained. If the limiter is not used, the interface flux F_C (Eq. (3.29)) is

$$F_C = a \left[U_L + \frac{\Delta x}{4} (u_x)_{j-1/2}^{n-1/2} + \frac{\Delta t}{4} (u_t)_{j-1/2}^{n-1/2} \right], \ (a > 0)$$
(3.36)

With this F_C, the upwind CESE scheme, which is a combination of Eqs. (3.2) and (3.26), can be written in a matrix form as Eqs. (3.8) and (3.9), in which $q = [u, (\Delta x/4)u_x]^T$ and the matrices Q_L and Q_R are functions of the CFL number $\nu = a\Delta t/\Delta x$:

$$Q_L = \frac{1}{2} \begin{bmatrix} 1+\nu & 1-\nu^2 \\ -1+\nu & -1+4\nu-\nu^2 \end{bmatrix}, \ Q_R = \frac{1}{2} \begin{bmatrix} 1-\nu & -1+\nu^2 \\ 1-\nu & -1+\nu^2 \end{bmatrix}$$
(3.37)

If the initial data $(u_L)_j^{n-1/4}$ and $(u_R)_j^{n-1/4}$ are reconstructed using the limiter, the matrix form shown as Eqs. (3.8) and (3.9) still holds, but the matrices $\mathbf{Q}_L$ and $\mathbf{Q}_R$ should be

$$\mathbf{Q}_L = \frac{1}{2}\begin{bmatrix} 1+\nu & 1-\nu^2 \\ -1+\nu & -1+2(1+\phi_L)\nu+(1-2\phi_L)\nu^2 \end{bmatrix}, \ \mathbf{Q}_R = \frac{1}{2}\begin{bmatrix} 1-\nu & -1+\nu^2 \\ 1-\nu & -1+\nu^2 \end{bmatrix}$$

(3.38)

where

$$\phi_L = W(1,\theta_1^L,\theta_2^L), \theta_1^L = \frac{(U_R-U_L)/(\Delta x/2)}{(u_x)_{j-1/2}^{n-1/2}}, \theta_2^L = \frac{(u_x)_{j+1/2}^{n-1/2}}{(u_x)_{j-1/2}^{n-1/2}}$$

(3.39)

The scheme without limiter Eq. (3.37) can be regarded as the special case of Eq. (3.38) with the limiter function $\phi_L = 1$.

Based on matrices $\mathbf{Q}_L$ and $\mathbf{Q}_R$ in Eq. (3.38), we can examine the property of the upwind CESE scheme at the limiting case of $\nu = 0$. Recall that the a–α scheme becomes very diffusive when $\nu \to 0$ and cannot guarantee that $\mathbf{q}_j^n \to \mathbf{q}_j^{n-1}$ as $\Delta t \to 0$ (but Δx is a finite constant). However, the upwind CESE scheme does not suffer from such a deficiency. A direct evaluation of matrices $\mathbf{Q}_L$ and $\mathbf{Q}_R$ in Eq. (3.38) as $\nu \to 0$ shows

$$\mathbf{Q}_L^2 \to \frac{1}{4}\begin{bmatrix} 1 & 1 \\ -1 & -1 \end{bmatrix}^2 = \mathbf{0}, \ \mathbf{Q}_R^2 \to \frac{1}{4}\begin{bmatrix} 1 & -1 \\ 1 & -1 \end{bmatrix}^2 = \mathbf{0},$$

$$\mathbf{Q}_L\mathbf{Q}_R + \mathbf{Q}_R\mathbf{Q}_L \to \frac{1}{4}\begin{bmatrix} 1 & 1 \\ -1 & -1 \end{bmatrix}\begin{bmatrix} 1 & -1 \\ 1 & -1 \end{bmatrix} + \frac{1}{4}\begin{bmatrix} 1 & -1 \\ 1 & -1 \end{bmatrix}\begin{bmatrix} 1 & 1 \\ -1 & -1 \end{bmatrix} = \mathbf{I}$$

(3.40)

Consequently, the time marching scheme for a complete time step (Eq. (3.9)) can always ensure that $\mathbf{q}_j^n \to \mathbf{q}_j^{n-1}$ as $\Delta t \to 0$, regardless of the form of the limiter function. Unlike the a–α scheme, the upwind CESE scheme is inherently CFL-number insensitive.

3.3.3   Scheme for Euler Equations

The 1D unsteady Euler equations for a perfect gas are written as

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

(3.41)

where $\boldsymbol{U} = [\rho, \ \rho u, \ E]^T$, $\boldsymbol{F} = \left[\rho u, \ \rho u^2 + p, \ (E+p)u\right]^T$ are the vectors of the conserved variables and the inviscid flux. Here, ρ, u, and p represent the density,

velocity, and pressure, respectively. The total energy per unit volume is denoted by E, which is $E = p/(\gamma - 1) + \rho u^2/2$ with constant ratio of specific heat $\gamma = 1.4$.

The upwind CESE scheme for a scalar equation can be directly applied to solve each component of $\mathbf{U}$. However, the upwind procedure to calculate the flux through the interface between $(\text{CE}^-)_j^n$ and $(\text{CE}^+)_j^n$ needs a substantial extension, because of the complexity of the eigen-structure of Eq. (3.41). Still, the flux vector $\mathbf{F}_C$ can be expressed as

$$\mathbf{F}_C = \hat{\mathbf{F}}\left((\mathbf{U}_L)_j^{n-1/4}, (\mathbf{U}_R)_j^{n-1/4}\right), \tag{3.42}$$

where $\hat{\mathbf{F}}$ stands for any appropriate upwind numerical flux solver, e.g., approximate Rimann solvers and other upwind flux functions. Moreover, $(\mathbf{U}_L)_j^{n-1/4}$ and $(\mathbf{U}_R)_j^{n-1/4}$ are the data on the two sides of line segment CD in Fig. 3.2, which are reconstructed with an appropriate limiter and serve as the discontinuous initial data of the local Riemann problem. Fortunately, any efficient and robust upwind flux solver [4, 5] exsiting in the FVM literature can be employed to calculate $\mathbf{F}_C$.

3.3.4 Remarks on Upwind CESE Method

For the purpose of capturing discontinuities, the upwind CESE method introduces necessary numerical dissipation by the upwind procedure to calculate F_C (see Eqs. (3.29)–(3.35)). This fact might lead to confusion between the classical upwind FVM [4, 5] and the present upwind CESE method. In the former, the upwind flux technique to tackle local Riemann problems is viewed as the building block of the whole method. In the latter, the upwind flux technique plays a different role.

To shed light on this issue, the interval $[j − 1/2, j + 1/2]$ is taken as a representative control volume to analyse, which is called "cell" in the FVM, and marked as line segment AB in Fig. 3.2. In this section, three fluxes are related with $[j − 1/2, j + 1/2]$, denoted by F_L, F_R, and F_C. Apparently, these fluxes can be classified as the flux through the boundary of the control volume (F_L and F_R) and the flux through the diaphragm inside the control volume (F_C). For the upwind CESE scheme, the upwind procedure is performed only to calculate F_C, while fluxes through the boundaries are not linked to any local Riemann problems. Owing to the time-marching strategy on the staggered mesh, information at points $x_{j-1/2}$ and $x_{j+1/2}$ are ready before evaluating F_L and F_R. Thus, the procedure to calculate F_L and F_R only involves Taylor expansion inside the SE, the definition of physical flux function $f(u)$, and the Cauchy-Kowalewski procedure (steps to derive u_t and f_t from u_x as shown in Eqs. (2.26)–(2.28)). On the contrary, the non-staggered FVM employs the upwind flux solver to get all fluxes through boundaries of control volumes, because each F_L or F_R should be treated as the result of a local Riemann problem. Usually, F_C is not considered in the FVM.

Recall that F_C is absent in Eq. (3.2). Therefore, the upwind procedure in the CESE scheme never affects the time-marching algorithm for the average value of $u(x, t)$ on $[j − 1/2, j + 1/2]$, but solely redistributes $u(x, t)$ on $[j − 1/2, j + 1/2]$ through the calculation of u_x using Eq. (3.26). However, the upwind FVM relies on two upwind numerical fluxes $\hat{f}_{j−1/2}$ and $\hat{f}_{j+1/2}$ to update the average value of $u(x, t)$ on $[j − 1/2, j + 1/2]$. This is the reason why we claim that the upwind CESE method utilizes the upwind technique in a different way from the traditional upwind FVM.

3.4   Comparison of Different CESE Schemes

So far, four CESE schemes, namely the a scheme, the a–α scheme, the CNI scheme, and the upwind CESE scheme, have been introduced. To make a comparison, their main formulas and properties are listed in the following tables. Table 3.1 indicates that the CNI and upwind CESE schemes are more favourable for the practical simulations, since they can capture discontinuities with reasonable numerical dissipation and they are free of the sensitivity issue which is encountered by the a–α scheme. The key formulas in each scheme are shown in Table 3.2. It is notable that all CESE schemes share a common approach to updating u_j^n. Only the a scheme and the upwind CESE scheme update $(u_x)_j^n$ are based on the conservation law for sub-CEs. In contrast, the a–α and the CNI schemes construct $(u_x)_j^n$ using the finite-difference-like approximation followed by some kind of weighted averaging technique.

Table 3.1 Properties of four different CESE schemes

	Central or upwind	Dissipative or non-dissipative	CFL-number sensitive or insensitive
a scheme	Central	Non-dissipative	Insensitive
a–α scheme	Central	Dissipative	Sensitive
CNI scheme	Central	Dissipative	Insensitive
Upwind CESE	Upwind	Dissipative	Insensitive

Table 3.2 The formulas to update u_j^n and $(u_x)_j^n$ in each CESE scheme

	Formula for u_j^n	Formula for $(u_x)_j^n$
a scheme		$\frac{\Delta x}{4}(u_x)_j^n = \frac{1}{2}(U_R − U_L) + \frac{\Delta t}{2\Delta x}(2F_C − F_L − F_R)$
a–α scheme	$u_j^n = \frac{1}{2}(U_L + U_R)$	$(u_x)_j^n = W\left((u_x^−)_j^n, (u_x^+)_j^n, \alpha\right)$
CNI scheme	$+ \frac{\Delta t}{2\Delta x}(F_L − F_R)$	$(u_x)_j^n = \hat{W}\left((\hat{u}_x^−)_j^n, (\hat{u}_x^+)_j^n, \nu\right)$
Upwind CESE		$\frac{\Delta x}{4}(u_x)_j^n = \frac{1}{2}(U_R − U_L) + \frac{\Delta t}{2\Delta x}(2F_C − F_L − F_R)$

Table 3.3 The matrices Q_L and Q_R in Eq. (3.8), which is the matrix form of the CESE method applied to the linear scalar convection equation (Eq. (3.1)) with $f = au$, $a > 0$)

	Q_L	Q_R
a scheme	$\frac{1}{2}\begin{bmatrix} 1+v & 1-v^2 \\ -1 & -1+v \end{bmatrix}$	$\frac{1}{2}\begin{bmatrix} 1-v & -1+v^2 \\ 1 & -1-v \end{bmatrix}$
a–α scheme ($\alpha = 0$)	$\frac{1}{2}\begin{bmatrix} 1+v & 1-v^2 \\ -1/2 & v \end{bmatrix}$	$\frac{1}{2}\begin{bmatrix} 1-v & -1+v^2 \\ 1/2 & -v \end{bmatrix}$
Upwind (without limiter)	$\frac{1}{2}\begin{bmatrix} 1+v & 1-v^2 \\ -1+v & -1+4v-v^2 \end{bmatrix}$	$\frac{1}{2}\begin{bmatrix} 1-v & -1+v^2 \\ 1-v & -1+v^2 \end{bmatrix}$

When applied to the linear scalar convection equation (Eq. (3.1) with $f = au$ and a is a positive constant), it is possible to rewrite the CESE scheme in a very compact matrix form. In Table 3.3, the coefficient matrices for different CESE schemes are listed. The matrices for the CNI scheme are not tabulated due to their lengthiness. Such Q_L and Q_R, together with Eqs. (3.8) and (3.9) are useful for revealing the instrinct properties of different CESE schemes, e.g., the stability of each scheme.

3.5 Numerical Examples

Here, the 1D scalar convection problem and Sod's shock-tube problem with uniform grid size of 0.01 are computed using different CESE schemes. The corresponding C + + source codes implementing the $a-\alpha$ CESE scheme are presented in the Appendix. For Eq. (3.1) with $f = u$, a square wave propagates in a computational domain $[-1, 1]$ till $t = 2.0$ with the initial condition described as

$$u(x, 0) = \begin{cases} 1, & if \ -0.5 \le x \le 0.5 \\ 0. & else \end{cases} \qquad (3.43)$$

The periodic boundary condition is imposed on both ends. The results are plotted in Fig. 3.3. For CFL $= 1$, all three schemes provide a virtually exact result. When CFL $= 0.8$, both the a-α and the CNI schemes suffer from strong oscillations on the upwind sides of the discontinuities, while the upwind scheme provides a result with satisfactory accuracy. If CFL is further decreased to an extremely small value, the CNI and upwind scheme can capture the discontinuities very well, but the large dissipation in the $a-\alpha$ scheme abnormally smears out the discontinuities. This case proves the CFL insensitivity of the CNI and upwind CESE schemes.

In the Sod's shock-tube problem [6], the gas is initially separated at $x = 1$ with left and right states $(\rho, u, p)_L = (1.0, 0.0, 1.0)$ and $(\rho, u, p)_R = (0.125, 0.0, 0.1)$. In Fig. 3.4, the computed density profiles at $t = 0.4$ are shown. For CFL $= 0.8$, all

(a) *a-α* CESE
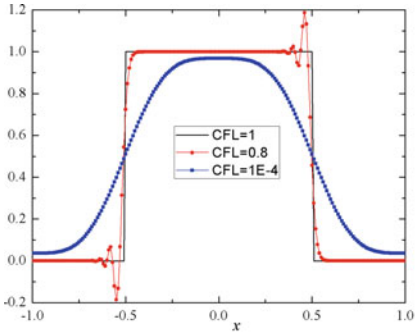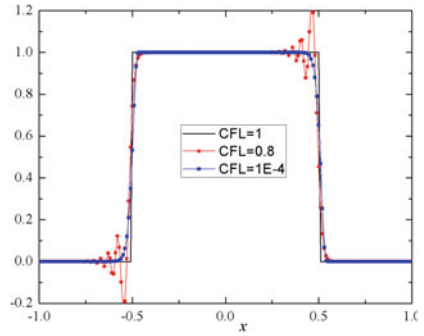
(b) CNI CESE
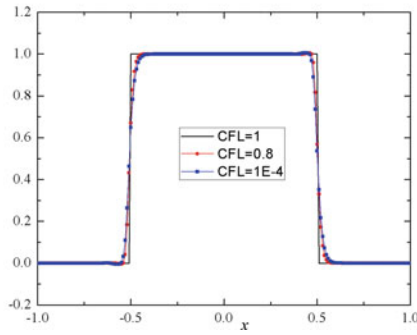
(c) Upwind CESE

Fig. 3.3 Solutions for scalar convection equation: square wave problem at $t = 2$

the schemes can capture the wave structures precisely. For an extremely small CFL number, the CNI and upwind schemes maintain great accuracy, but the *a-α* scheme seriously smeared out the rarefaction, contact, and shock waves.

(a) *a-α* CESE
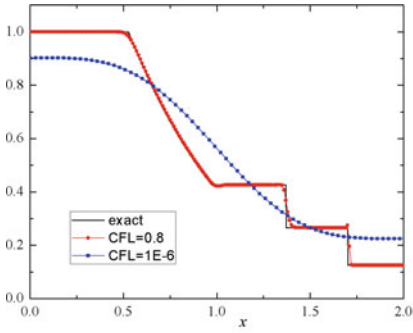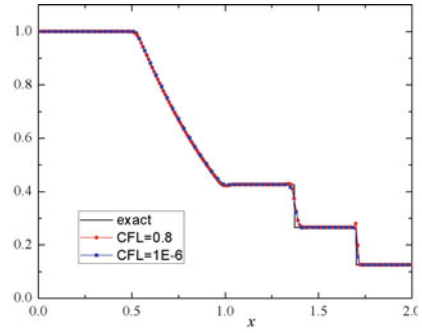
(b) CNI CESE
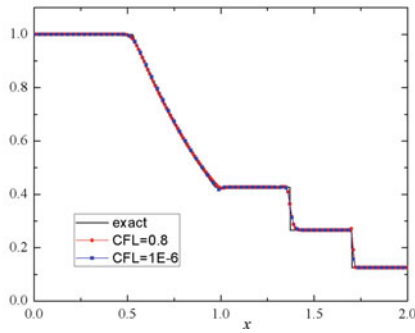
(c) upwind CESE

Fig. 3.4 Density profile of Sod's shock-tube problem at $t = 0.4$

References

1. Chang, S.C, & Wang, X.Y. (2002). Courant number insensitive CE/SE Euler scheme. In *38th AIAA/ASME/SAE/ASEE Joint Propulsion Conference & Exhibit*.
2. Shen, H., Wen, C. Y., & Zhang, D. L. (2015). A characteristic space–time conservation element and solution element method for conservation laws. *Journal of Computational Physics, 288*, 101–118.
3. Li, W., & Ren, Y. X. (2012). The multi-dimensional limiters for solving hyperbolic conservation laws on unstructured grids. *Journal of Computational Physics, 230*, 7775–7795.
4. Toro, E. F. (2009). *Riemann solvers and numerical methods for fluid dynamics: A practical introduction*. Springer.
5. Kitamura, K. (2020). *Advancement of shock capturing computational fluid dynamics methods: Numerical flux functions in finite volume method*. Springer Nature.
6. Sod, G.A. (1978). A survey of several finite difference methods for systems of nonlinear hyperbolic conservation laws.

Chapter 4
Multi-dimensional CESE Schemes

The previous chapter has shown that the necessary numerical dissipation can be introduced in 1D CESE schemes through either a central or upwind approach. In this chapter, we present the extensions of 2D CESE schemes based on cartesian meshes, followed by a detailed description of implementation on unstructured meshes. Both the central schemes and upwind schemes will be presented. Then, numerical examples will be provided to demonstrate the capabilities of the present schemes.

4.1 CESE Schemes on Cartesian Meshes

4.1.1 The Improved a-α CESE Scheme

The marching scheme requires physical variables and their spatial derivatives at each mesh point. In the pioneer development of the 2D CESE solver [1], the domain is discretized by congruent triangles. Based on a similar technique, Zhang, Yu, and Chang [2] reported a further extension of the 2D CESE scheme on quadrilateral meshes. In the above versions of CESE schemes, the solution points are solely updated at the cell centers with a staggered stencil. As a result, a generalized flux technique was also proposed as a post-marching procedure to handle the "flux decoupling" problem when two mesh points cohost one sub-CE. Alternatively, an improved 2D CESE scheme [3, 4] was proposed with a new definition of SE and CE. In this updated scheme, the entire space–time region is divided into non-overlapping CEs, making it convenient and simpler for calculation and straightforward for extension to 3D scheme. Figure 4.1 shows the space–time geometrical configuration of the improved 2D CESE scheme with a uniform rectangular mesh. The spatial projections of the solution points at $(n-1/2)$th and nth time levels are denoted as ■ and

●, respectively. The solution is updated alternatively between the cell centers ● and cell vertices ■. Here $\Delta t/2 = t_n - t_{n-1/2}$ and P, P', and P'' subsequently represent the points at three successive timesteps. The conservation element of point P', denoted by CE(P'), is defined by the space–time hexahedron $ABCDA'B'C'D'$, i.e., the union of four sub-CEs: $CE^{LD}(P')$, $CE^{LU}(P')$, $CE^{RD}(P')$, and $CE^{RU}(P')$. In particular, $CE^{LD}(P')$ is defined as $AFPEA'F'P'E'$. Other sub-CEs are defined similarly. Next, consider the solution element of P', SE(P'), which consists of three orthogonal planes, $A'B'C'D'$, $EGG''E''$, and $FHH''F''$. The physical flux vector is assumed to be smooth within each SE and can be approximated by Taylor expansions about the mesh point associated with the SE. In the following, we present the construction of the 2D CESE solver using similar techniques as described in the 1D scheme.

(a) Mesh points on the x-y plane   (b) CE(P')

(c) SE(P')   (d) Fluxes in the sub-CEs
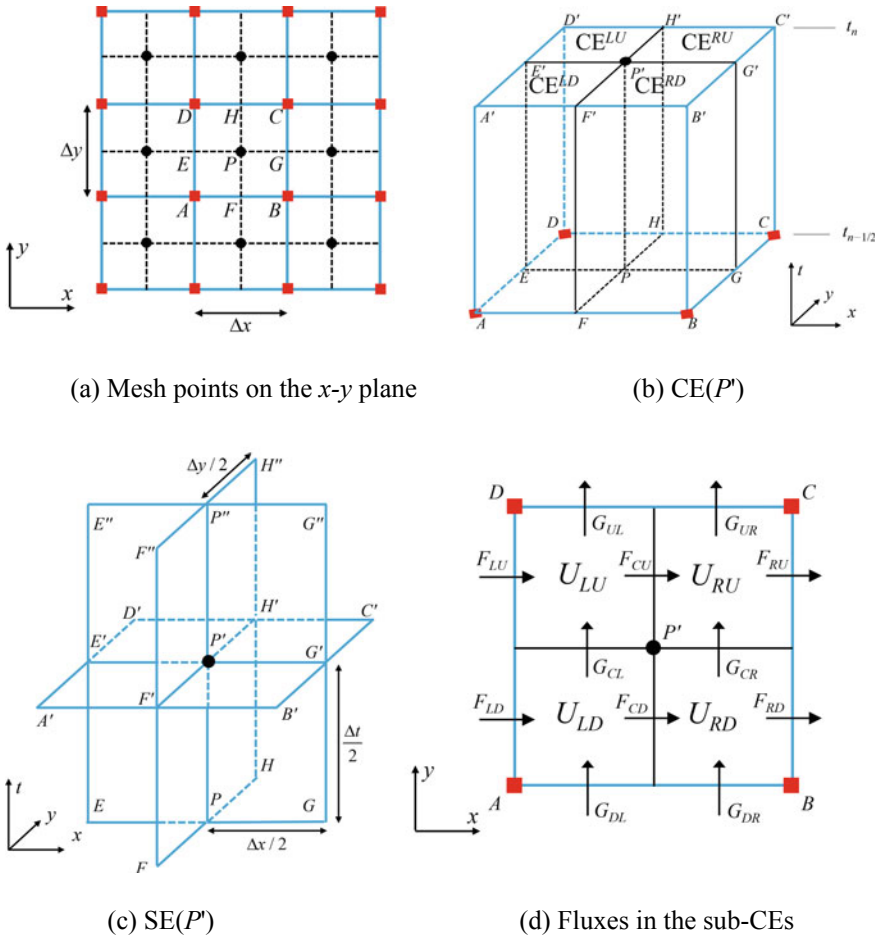
Fig. 4.1 Grid points in the spatial domain, definitions of CE and SE, and corresponding fluxes in a CE for the 2D CESE scheme on rectangular meshes

In this section, we consider the 2D scalar hyperbolic conservation law

$$\frac{\partial u}{\partial t} + \frac{\partial f}{\partial x} + \frac{\partial g}{\partial y} = 0. \tag{4.1}$$

By using the Gauss' divergence theorem, it is shown that Eq. (4.1) can be written in the form of

$$\oint_{S(CE(P'))} \boldsymbol{h} \cdot \boldsymbol{n} \, \mathrm{d}S = 0, \tag{4.2}$$

where $\boldsymbol{h} = (f, g, u)$ is the space–time flux vector, $S\big(\mathrm{CE}(P')\big)$ is the boundary of $\mathrm{CE}(P')$, $\boldsymbol{n}$ is the unit outward normal vector on the surface of the control volume. Note that the u, f, and g are approximated by first-order Taylor expansion about point P'. For any $(x, y, t) \in \mathrm{SE}(P')$, let

$$u(x, y, t) = u(\delta x, \delta y, \delta t)_{P'}, \tag{4.3}$$

$$f(x, y, t) = f(\delta x, \delta y, \delta t)_{P'}, \tag{4.4}$$

$$g(x, y, t) = g(\delta x, \delta y, \delta t)_{P'}, \tag{4.5}$$

where $X(\delta x, \delta y, \delta t)_N$ denotes the first-order Taylor expansion about point N as

$$X(\delta x, \delta y, \delta t)_N = X_N + (X_x)_N \delta x + \big(X_y\big)_N \delta y + (X_t)_N \delta t, \tag{4.6}$$

and

$$\delta x = x - x_N, \quad \delta y = y - y_N, \quad \delta t = t - t_N. \tag{4.7}$$

In addition, the spatial and temporal derivatives of $f(u)$ and $g(u)$ can be derived by the chain rule (Eqs. (2.26) and (2.28)) and $u_t = -f_x - g_y$.

Meanwhile, one obtains local flux conservation of the four sub-CEs by integrating Eq. (4.2) over their surfaces, given by

$$U'_{LD} \frac{\Delta x \Delta y}{4} = U_{LD} \frac{\Delta x \Delta y}{4} + (F_{LD} - F_{CD}) \frac{\Delta y \Delta t}{4} + (G_{DL} - G_{CL}) \frac{\Delta x \Delta t}{4}, \tag{4.8}$$

$$U'_{RD} \frac{\Delta x \Delta y}{4} = U_{RD} \frac{\Delta x \Delta y}{4} + (F_{CD} - F_{RD}) \frac{\Delta y \Delta t}{4} + (G_{DR} - G_{CR}) \frac{\Delta x \Delta t}{4}, \tag{4.9}$$

$$U'_{RU}\frac{\Delta x \Delta y}{4} = U_{RU}\frac{\Delta x \Delta y}{4} + (F_{CU} - F_{RU})\frac{\Delta y \Delta t}{4} + (G_{CR} - G_{UR})\frac{\Delta x \Delta t}{4},$$
$$(4.10)$$

$$U'_{LU}\frac{\Delta x \Delta y}{4} = U_{LU}\frac{\Delta x \Delta y}{4} + (F_{LU} - F_{CU})\frac{\Delta y \Delta t}{4} + (G_{CL} - G_{UL})\frac{\Delta x \Delta t}{4},$$
$$(4.11)$$

where U_{LD}, U'_{LD}, F_{LD}, F_{CD}, G_{DL}, and G_{CL} are the average fluxes on the surfaces of CE^{LD}(P'), AFPE, A'F'P'E', AEE'A', FPP'F', AFF'A', and EPP'E', respectively. Similar definitions are applied to the other sub-CEs. By definition, the fluxes U_{LD}, F_{LD}, and G_{DL} can be determined via the Taylor expansion in SE(A). It is also important to emphasize that the fluxes F_{CD} and G_{CL} that denote the fluxes across the "inner" surface of two neighboring sub-CEs, however, are not trivially available due to the presence of discontinuities. In fact, it will be shown later that all the fluxes across the interfaces among sub-CEs vanish in the formulation of the time marching scheme of u. Apart from this, based on the assumption that the distribution within each SE is linear, the average fluxes on the exterior surfaces of CE^{LD}(P') read

$$U_{LD} = u\left(\frac{\Delta x}{4}, \frac{\Delta y}{4}, 0\right)_A$$
$$(4.12)$$

$$U'_{LD} = u\left(-\frac{\Delta x}{4}, -\frac{\Delta y}{4}, 0\right)_{P'}$$
$$(4.13)$$

$$G_{DL} = g\left(\frac{\Delta x}{4}, 0, \frac{\Delta t}{4}\right)_A$$
$$(4.14)$$

$$F_{LD} = g\left(0, \frac{\Delta y}{4}, \frac{\Delta t}{4}\right)_A$$
$$(4.15)$$

Fluxes are expressed similarly for the other three sub-CEs.

To proceed, by adding Eqs. (4.8)–(4.11), the fluxes through the interfaces between sub-CEs are well balanced. Consequently, the value of the solution point P' can be derived as

$$u(P') = \frac{1}{4}(U_{LD} + U_{RD} + U_{RU} + U_{LU}) + \frac{\Delta t}{4\Delta x}(F_{LD} - F_{RD} - F_{RU} + F_{LU})$$
$$+ \frac{\Delta t}{4\Delta y}(G_{LD} + G_{RD} - G_{RU} - G_{LU})$$
$$(4.16)$$

Moreover, by replacing the average fluxes in Eq. (4.16) with the Taylor expansions about their corresponding SEs centers, we can obtain the explicit time marching scheme for $u(P')$ as

$$u(P') = \frac{1}{4}\overline{u} + \frac{\Delta t}{4\Delta x}\overline{f} + \frac{\Delta t}{4\Delta y}\overline{g}, \tag{4.17}$$

where

$$\overline{u} = u\left(\frac{\Delta x}{4}, \frac{\Delta y}{4}, 0\right)_A + u\left(-\frac{\Delta x}{4}, \frac{\Delta y}{4}, 0\right)_B + u\left(-\frac{\Delta x}{4}, -\frac{\Delta y}{4}, 0\right)_C + u\left(\frac{\Delta x}{4}, -\frac{\Delta y}{4}, 0\right)_D,$$

$$\overline{f} = f\left(0, \frac{\Delta y}{4}, \frac{\Delta t}{4}\right)_A + f\left(0, -\frac{\Delta y}{4}, \frac{\Delta t}{4}\right)_D - f\left(0, \frac{\Delta y}{4}, \frac{\Delta t}{4}\right)_B - f\left(0, -\frac{\Delta y}{4}, \frac{\Delta t}{4}\right)_C,$$

$$\overline{g} = g\left(\frac{\Delta x}{4}, 0, \frac{\Delta t}{4}\right)_A + g\left(-\frac{\Delta x}{4}, 0, \frac{\Delta t}{4}\right)_B - g\left(-\frac{\Delta x}{4}, 0, \frac{\Delta t}{4}\right)_C - g\left(\frac{\Delta x}{4}, 0, \frac{\Delta t}{4}\right)_D.$$

With respect to the spatial derivatives, by using the continuous assumptions at points A', B', C', and D', the following relations can be formulated

$$u\left(-\frac{\Delta x}{2}, -\frac{\Delta y}{2}, 0\right)_{P'} = u\left(0, 0, \frac{\Delta t}{2}\right)_A, \tag{4.18}$$

$$u\left(\frac{\Delta x}{2}, -\frac{\Delta y}{2}, 0\right)_{P'} = u\left(0, 0, \frac{\Delta t}{2}\right)_B, \tag{4.19}$$

$$u\left(\frac{\Delta x}{2}, \frac{\Delta y}{2}, 0\right)_{P'} = u\left(0, 0, \frac{\Delta t}{2}\right)_C, \tag{4.20}$$

$$u\left(-\frac{\Delta x}{2}, \frac{\Delta y}{2}, 0\right)_{P'} = u\left(0, 0, \frac{\Delta t}{2}\right)_D. \tag{4.21}$$

After mathematical manipulation of Eqs. (4.19) and (4.20), (4.18) and (4.21), (4.20) and (4.21), (4.18) and (4.19), respectively, the spatial derivatives at solution point P' can be explicitly calculated as

$$u_x^+(P') = \frac{1}{\Delta x}\left[u\left(0, 0, \frac{\Delta t}{2}\right)_B + u\left(0, 0, \frac{\Delta t}{2}\right)_C - 2u(P')\right], \tag{4.22}$$

$$u_x^-(P') = -\frac{1}{\Delta x}\left[u\left(0, 0, \frac{\Delta t}{2}\right)_A + u\left(0, 0, \frac{\Delta t}{2}\right)_D - 2u(P')\right], \tag{4.23}$$

$$u_y^+(P') = \frac{1}{\Delta y}\left[u\left(0, 0, \frac{\Delta t}{2}\right)_C + u\left(0, 0, \frac{\Delta t}{2}\right)_D - 2u(P')\right], \tag{4.24}$$

$$u_y^-(P') = -\frac{1}{\Delta y}\left[u\left(0, 0, \frac{\Delta t}{2}\right)_A + u\left(0, 0, \frac{\Delta t}{2}\right)_B - 2u(P')\right]. \tag{4.25}$$

One note that, two derivatives were estimated along both spatial directions (denoted by superscripts $+$ and $-$). Recall that to a weighted averaged function Eq. (3.6) is again adopted to suppress numerical wiggles, such that

$$u_x(P') = W\big((u_x^-)_{P'}, (u_x^+)_{P'}, \alpha\big), \tag{4.26}$$

$$u_y(P') = W\Big((u_y^-)_{P'}, (u_y^+)_{P'}, \alpha\Big). \tag{4.27}$$

The Eqs. (4.17) and (4.22)–(4.27) constitute the first half-step marching scheme from cell vertices to cell centers. The second half-step scheme that marches from centers to vertices follows a similar approach. As described above, constructing the improved 2D a–α CESE scheme based on rectangular grids is genuinely simple and could be straightforwardly extended to the 3D scheme.

4.1.2 CNI CESE Scheme

When applying the above 2D a–α schemes [1–4] for solving problems with highly non-uniform meshes, the local CFL number may vary significantly from its stability limit of 1 to approaching 0. Namely, the solutions tend to smear out by decreasing the CFL number. The Courant number insensitive (CNI) scheme was proposed to address this issue to alleviate the sensitivity problem by improving the strategy in updating the derivatives [5]. A new point, I_m, is designated dynamically moving along the line segment connecting the vertices and the centers M'_m of each sub-CEs (Fig. 4.2). Analogous to the 1D CNI scheme (Sect. 3.2), if we define the local and global Courant numbers as v and v_0, the interpolated points are designated to approach the centers when $v/v_0 \to 0$, and approach the cell vertices when $v/v_0 \to 1$.

For example, the coordinates of I'_1 are

$$x(I'_1) = \frac{v}{v_0}x(A') + \left(1 - \frac{v}{v_0}\right)x(M'_1), \tag{4.28}$$

$$y(I'_1) = \frac{v}{v_0}y(A') + \left(1 - \frac{v}{v_0}\right)y(M'_1). \tag{4.29}$$

If we approximate $u(I'_1)$ at $t = t_n$ from the expansions about A at $t = t_{n-1/2}$ and P' at $t = t_n$ as

$$u(\delta x_+, \delta y_+, 0)_{P'} = u(I'_1) = u\left(\delta x_-, \delta y_-, \frac{\Delta t}{2}\right)_A \tag{4.30}$$

where $\delta x_+ = x(I'_1) - x(P')$, $\delta y_+ = y(I'_1) - y(P')$, $\delta x_- = x(I'_1) - x(A)$, and $\delta y_- = y(I'_1) - y(A)$. Similar relations can be derived for each I'_m. Combining Eq. (4.30)

Fig. 4.2 Definition of points
in 2D CNI CESE scheme
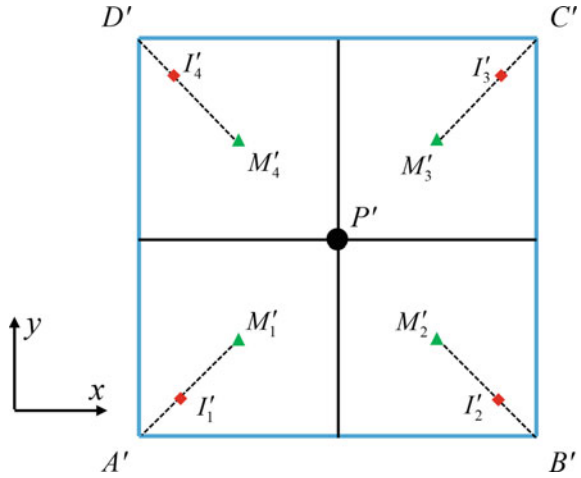
for $u(I'_1)$ and that for $u(I'_4)$, one can explicitly formulate two independent conditions
for calculating the spatial derivatives of $u(P')$. Repeating the same procedure for
combinations of $u(I'_2)$ and $u(I'_3)$, $u(I'_1)$ and $u(I'_2)$, $u(I'_3)$ and $u(I'_4)$, we arrive at four
sets of spatial derivatives. The weighted average function is then used to calculate
the optimal derivatives for capturing the discontinuities.

4.1.3 Upwind CESE Scheme

In the 2D upwind CESE scheme that Shen et al. [6] proposed, the CE retains the
same form as the central CESE scheme. The time marching scheme of u is the
same as in the a-α scheme. The upwind procedure only affects the calculation of
the spatial derivatives, and global space–time conservation is ensured. In analogue
to the definition in the 1D upwind scheme, for example, SE(P') is defined as cuboid
$A'B'C'D'A''B''C''D''$ (Fig. 4.3), where the physical flux vector is approximated by
the Taylor expansion about point P'. The average values at the top surfaces of the
four sub-CEs of CE(P') read

$$U'_{LD} = u\left(-\frac{\Delta x}{4}, -\frac{\Delta y}{4}, 0\right)_{P'}, \tag{4.31}$$

$$U'_{RD} = u\left(\frac{\Delta x}{4}, -\frac{\Delta y}{4}, 0\right)_{P'}, \tag{4.32}$$

$$U'_{RU} = u\left(\frac{\Delta x}{4}, \frac{\Delta y}{4}, 0\right)_{P'}, \tag{4.33}$$
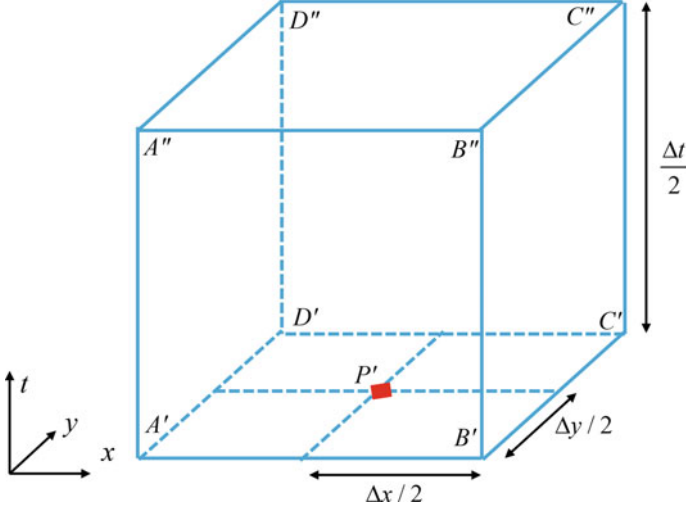
Fig. 4.3 Definition of SE of the upwind CESE scheme

$$U'_{LU} = u\left(-\frac{\Delta x}{4}, \frac{\Delta y}{4}, 0\right)_{P'}.$$ (4.34)

Referring to Fig. 4.1d, by substituting Eqs. (4.31) in (4.8), (4.32) in (4.9) and subtracting the two, one obtains

$$u_x^D(P') = \frac{2}{\Delta x}\left[(U_{RD} - U_{LD}) + \frac{\Delta t}{\Delta x}(2F_{CD} - F_{LD} - F_{RD}) + \frac{\Delta t}{\Delta y}(G_{DR} - G_{CR} - G_{DL} + G_{CL})\right].$$ (4.35)

Equation (4.35) represents the spatial derivative in the x direction based on the information of the lower pair of CEs. Note that in contrast to central schemes, the upwind scheme requires additional estimations of inner fluxes (e.g., F_{CD}, G_{CR}, and G_{CL} in Eq. (4.35)) to compute spatial derivatives. Analogously, with the aid of Eqs. (4.31) and (4.34) and by combining with Eqs. (4.10) and (4.11), Eqs. (4.8) and (4.11), Eqs. (4.9) and (4.10), respectively. The following equations for spatial derivatives can be formulated

$$u_x^U(P') = \frac{2}{\Delta x}\left[(U_{RU} - U_{LU}) + \frac{\Delta t}{\Delta x}(2F_{CU} - F_{LU} - F_{RU}) + \frac{\Delta t}{\Delta y}(G_{CR} - G_{UR} - G_{CL} + G_{UL})\right],$$ (4.36)

$$u_y^L(P') = \frac{2}{\Delta y}\left[(U_{LU} - U_{LD}) + \frac{\Delta t}{\Delta x}(F_{LU} - F_{CU} - F_{LD} + F_{CD}) + \frac{\Delta t}{\Delta y}(2G_{CL} - G_{DL} - G_{UL})\right],$$ (4.37)

$$u_y^R(P') = \frac{2}{\Delta y}\left[(U_{RU} - U_{RD}) + \frac{\Delta t}{\Delta x}(F_{CU} - F_{RU} - F_{CD} + F_{RD}) + \frac{\Delta t}{\Delta y}(2G_{CR} - G_{DR} - G_{UR})\right].$$ (4.38)

The final spatial derivatives are calculated by using WBAP limiter to remove small noises near the strong discontinuities, e.g.,

$$u_x(P') = u_x^C(P')WBAP^{L2}(1, \theta_1, \theta_2) = \frac{1}{2}\Big[u_x^D(P') + u_x^U(P')\Big]WBAP^{L2}(1, \theta_1, \theta_2), \tag{4.39}$$

where

$$\theta_1 = u_x^D(P')/u_x^C(P'),$$

$$\theta_2 = u_x^U(P')/u_x^C(P').$$

In the expressions for spatial derivatives in Eqs. (4.35)–(4.38), the fluxes on CE(P') surfaces can be easily approximated by the Taylor expansions of the values from $t = t_{n-1/2}$. To complete Eqs. (4.35)–(4.38), the fluxes through the inner boundaries of CE(P'), F_{CD}, F_{CU}, G_{CL}, and G_{CR}, need to be solved. However, the points at the two sides of the interface between two sub-CEs belong to different SEs. In that case, the fluxes across these interfaces may be discontinuous. For this reason, they can be calculated by an upwind procedure. For example,

$$F_{CD} = \widehat{F}\big(u_{CD}^-, u_{CD}^+\big), \tag{4.40}$$

where $\widehat{F}$ refers to any efficient Riemann solver, u_{CD}^- and u_{CD}^+ denote the values at the centroid of the interface *FPP'F'*, and they are respectively approximated by

$$u_{CD}^- = U_{LD} + \big(u_x^*\big)_{LD}\frac{\Delta x}{4} + \big(u_t^*\big)_{LD}\frac{\Delta t}{4} \tag{4.41}$$

$$u_{CD}^+ = U_{RD} - \big(u_x^*\big)_{RD}\frac{\Delta x}{4} + \big(u_t^*\big)_{RD}\frac{\Delta t}{4} \tag{4.42}$$

As an analogue to the 1D upwind CESE scheme, reconstructed slopes are employed in the calculation of the upwind fluxes to eliminate the spurious oscillations:

$$\big(u_x^*\big)_{LD} = u_x(A)WBAP^{L2}\big(1, \theta_{LD}^1, \theta_{LD}^2\big) \tag{4.43}$$

$$\big(u_x^*\big)_{RD} = u_x(B)WBAP^{L2}\big(1, \theta_{RD}^1, \theta_{RD}^2\big) \tag{4.44}$$

where the parameters in WBAP limiter are

$$\theta_{LD}^1 = \frac{(U_{RD} - U_{LD})/(\Delta x/2)}{u_x(A)}, \quad \theta_{LD}^2 = \frac{u_x(B)}{u_x(A)},$$

$$\theta_{RD}^1 = \frac{(U_{RD} - U_{LD})/(\Delta x/2)}{u_x(B)}, \quad \theta_{RD}^2 = \frac{u_x(A)}{u_x(B)}. \tag{4.45}$$

The temporal derivatives are hereafter calculated by the chain rule and Eq. (4.1),

$$\left(u_t^*\right)_{LD} = -f_x\left(U_{LD}, \left(u_x^*\right)_{LD}\right) - g_y\left(U_{LD}, u_y(A)\right) \tag{4.46}$$

$$\left(u_t^*\right)_{RD} = -f_x\left(U_{RD}, \left(u_x^*\right)_{RD}\right) - g_y\left(U_{RD}, u_y(B)\right) \tag{4.47}$$

4.2 CESE Schemes on Unstructured Meshes

In order to extend the capability of the CESE scheme to solve problems with complex spatial geometries, CESE schemes on unstructured meshes have been proposed and developed. Notably, both central and upwind versions have been developed for hybrid meshes of triangular and quadrilateral elements. Hereafter, the second order CESE scheme on hybrid meshes [7] is presented here.

4.2.1 a-α CESE Scheme

Consider a 2D mesh consisting of quadrilateral and triangular elements as sketched in Fig. 4.4. The V_i denotes the vertices of the cells. Initially, the value of u, and its spatial derivatives, u_x and u_y, are provided at these vertices. For each cell, C_m represents its centroid, and O_m is the mid-point of the cell edge. Meanwhile, V_i, V_i', and V_i'' denote points at time step $t_{n-1/2}$, t_n, $t_{n+1/2}$, respectively. Other space–time points are defined similarly.

The staggered marching strategy is again adopted here. The scheme marches from vertices V_i' to cell centroids C_m'' from $t = t_n$ to $t = t_{n+1/2}$. The corresponding CE for C_m'' is either a triangular prism or a hexahedron (Fig. 4.4c and d). SE$\left(C_m''\right)$) is defined as four (e.g., SE$\left(C_1''\right)$) or five (e.g., SE$\left(C_2''\right)$) planes intersecting at the centers. Similarly, in the other half marching step, from $t = t_{n-1/2}$ to $t = t_n$, C_m and O_m form a polygonal element of which the centroid is denoted as G_i, and the scheme marches from C_m to the G_i. Note that G_i does not necessarily coincide with V_i. Thus, interpolation is required at each marching step. SE(V_i') is defined as M vertical planes and one horizontal plane intersecting at V_i', where M denotes the number of C_m linked to CE(V_i). Each CE is composed of a group of hexahedral sub-CEs. As shown in Fig. 4.5, $O_{m-1}C_m O_m V_i O_{m-1}'C_m' O_m' V_i'$ is the mth sub-CE belonging to CE(V_i'), $\tilde{G}_m$ and $\tilde{G}_m'$ are defined as the centroids of the bottom surface $O_{m-1}C_m O_m V_i$ and top surface $O_{m-1}'C_m' O_m' V_i'$, respectively.

(a) A hybrid mesh

(b) CE(V_1) and SE(V_1)

(c) CE(C_1)

(d) CE(C_2)
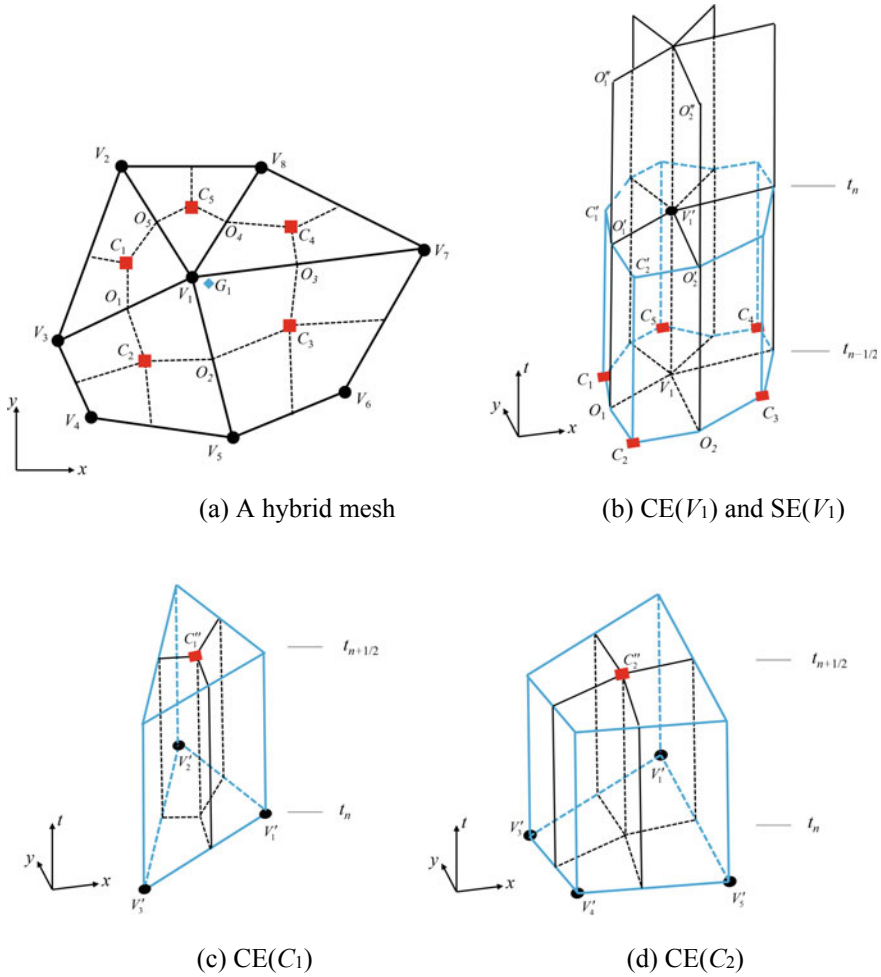
Fig. 4.4 Definition of conservation elements in CESE schemes based on a hybrid mesh

It should be noted that the marching scheme from $t = t_n$ to $t = t_{n+1/2}$ is a particular case of that from $t = t_{n-1/2}$ to $t = t_n$. From $t = t_n$ to $t = t_{n+1/2}$, the projection of a CE on the spatial domain is either a quadrilateral or a triangle, while the projection in the other step is a polygon with an arbitrary shape. Without loss of generality, we present the marching scheme from $t = t_{n-1/2}$ to $t = t_n$. By applying the integral conservation law Eq. (4.1) to the mth sub-CE, the flux-balancing relation is obtained as

$$\iint_{O'_{m-1}C'_m O'_m V'_i} u \, d\sigma = \iint_{O_{m-1}C_m O_m V_i} u \, d\sigma - \iint_{\substack{O_{m-1}C_m C'_m O'_{m-1}+C_m O_m O'_m C'_m \\ +O_{m-1}V_i V'_i O'_{m-1}+O_m V_i V'_i O'_m}} \mathbf{V} \cdot \mathbf{n} \, d\sigma, \quad (4.48)$$
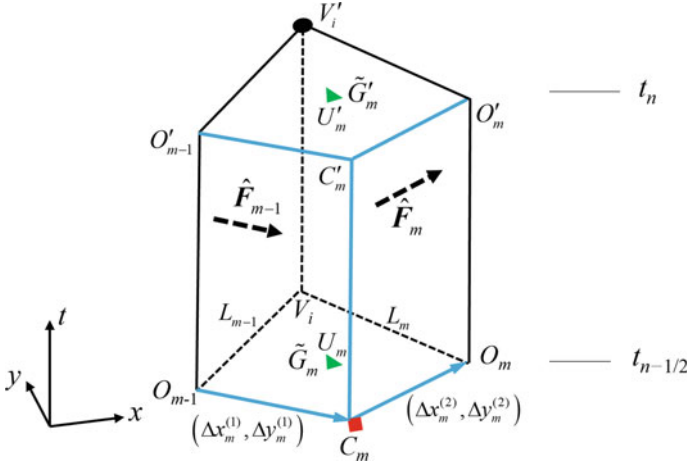
Fig. 4.5 The mth sub-CE of CE(V_i)

where $\mathbf{V} = (f, g)$ is the vector of the fluxes, and $\mathbf{n}$ is the unit outward normal vector of the surface. In second-order schemes, the integration over a surface is equivalent to the average value at the centroid multiplied by the surface area. Therefore, the above equation can be rewritten as

$$U'_m = U_m - \frac{\Delta t}{2S_m} \sum_{l=1}^{2} \left(\Delta y_m^{(l)} F_m^{(l)} - \Delta x_m^{(l)} G_m^{(l)} \right) + \frac{\Delta t}{2S_m} \left(L_{m-1} \widehat{F}_{m-1} - L_m \widehat{F}_m \right),$$
(4.49)

where S_m denotes the area of the top surface, and $\Delta t/2 = t_n - t_{n-1/2}$. U'_m, U_m, $(F_m^{(1)}, G_m^{(1)})$, and $(F_m^{(2)}, G_m^{(2)})$ represent the average values over $O'_{m-1} C'_m O'_m V'_i$, $O_{m-1} C_m O_m V_i$, $O_{m-1} C_m C'_m O'_{m-1}$, and $C_m O_m O'_m C'_m$, respectively. They can be approximated by first-order Taylor expansion in the corresponding SEs. $\widehat{F}_{m-1}$ and $\widehat{F}_m$ are the fluxes in the normal direction across the interfaces $O_{m-1} V_i V'_i O'_{m-1}$ and $O_m V_i V'_i O'_m$ with the neighbouring sub-CEs. L_{m-1} and L_m are the lengths of the line segments $O_{m-1} V_i$ and $O_m V_i$, respectively. The summation of the flux-balancing equations of every sub-CEs is expressed as

$$\iint\limits_{C'_1 O'_2 C'_2 ... C'_M} u \, d\sigma = \sum_{m=1}^{M} \iint\limits_{O'_{m-1} C'_m O'_m V'_i} u \, d\sigma$$
(4.50)

where $O_0 \equiv O_M$. By substituting Eqs. (4.48) and (4.49) into (4.50), the interface fluxes between sub-CEs are cancelled, the conservation law on the entire CE(V'_i) yields

$$u(G_i') = \left[\sum_{m=1}^{M} U_m S_m - \frac{\Delta t}{2} \sum_{m=1}^{M} \sum_{l=1}^{2} \left(\Delta y_m^{(l)} F_m^{(l)} - \Delta x_m^{(l)} G_m^{(l)} \right) \right] \bigg/ \sum_{m=1}^{M} S_m \quad (4.51)$$

Equation (4.51) is the explicit marching scheme for $u(G_i')$. In addition, the treatment of spatial derivatives is similar to that in cartesian meshes. By using the continuous assumption at C_m', the following relation can be established

$$u(C_m) + u_t(C_m) \frac{\Delta t}{2} = u(C_m') = u(G_i') + u_x(G_i')\delta x_m + u_y(G_i')\delta y_m \quad (4.52)$$

for each m, where $\delta x_m = x(C_m) - x(G_i)$, and $\delta y_m = y(C_m) - y(G_i)$. Apply the same relation at C_{m+1}', together with Eq. (4.52), the two spatial derivatives are ready to be solved using Cramer's rule,

$$u_x(G_i') = \frac{\begin{vmatrix} \delta u_m & \delta y_m \\ \delta u_{m+1} & \delta y_{m+1} \end{vmatrix}}{\begin{vmatrix} \delta x_m & \delta y_m \\ \delta x_{m+1} & \delta y_{m+1} \end{vmatrix}}, \quad u_y(G_i') = \frac{\begin{vmatrix} \delta x_m & \delta u_m \\ \delta x_{m+1} & \delta u_{m+1} \end{vmatrix}}{\begin{vmatrix} \delta x_m & \delta y_m \\ \delta x_{m+1} & \delta y_{m+1} \end{vmatrix}}, \quad (4.53)$$

where

$$\delta u_m = u(C_m) + u_t(C_m) \frac{\Delta t}{2} - u(G_i'). \quad (4.54)$$

Totally, M pairs of derivatives can be calculated in this approach. Shen et al. [7] suggested using the weighted average function to obtain the final derivatives as

$$u_x(G_i') = \frac{\sum_{m=1}^{M} W^{(m)} u_x^{(m)}(G')}{\sum_{m=1}^{M} W^{(m)}}, \quad u_y(G_i') = \frac{\sum_{m=1}^{M} W^{(m)} u_y^{(m)}(G')}{\sum_{m=1}^{M} W^{(m)}}, \quad (4.55)$$

where

$$W^{(m)} = \left(\prod_{i=1,\, i\neq m}^{M} \sqrt{\left[u_x^{(i)}(G')_i \right]^2 + \left[u_y^{(i)}(G')_i \right]^2} \right)^{\alpha} \quad (4.56)$$

Here, the adjustable variable α ($\alpha = 0 \sim 2$) controls the dissipation near the discontinuities. Smaller α results in lower dissipation, and the function will reduce to an arithmetic average when $\alpha = 0$. Finally, the value at V_i' is interpolated from G_i' as

$$u(V_i') = u(G_i') + u_x(G_i')\left[x(V_i') - x(G_i') \right] + u_y(G_i')\left[y(V_i') - y(G_i') \right]. \quad (4.57)$$

The a–α CESE scheme on the hybrid mesh formulated by Eqs. (4.51) and (4.57), while robust, is nevertheless sensitive to the Courant number. The following section addresses the further improvements to make the scheme Courant number insensitive, including the CNI scheme and upwind scheme. These two updated schemes, being closely analogous to those presented in Sects. 4.1.2 and 4.1.3, possess the identical time-marching scheme for u as provided in the a-α CESE scheme on a hybrid mesh. Only the marching scheme for spatial derivatives is modified accordingly.

4.2.2 CNI CESE Scheme

As a straightforward extension of the 1D CNI scheme, Shen and Parsani [8] extended the CNI scheme to a hybrid mesh. Recall that the central idea of the CNI scheme is that when the local Courant number (ν) approaches the global Courant number (ν_0), the scheme becomes the a-α scheme. When ν approaches 0, the scheme reduces to the non-dissipative counterpart. As depicted in Fig. 4.6, a new point I'_m is defined such that

$$x(I'_m) = \frac{\nu}{\nu_0} x(C'_m) + \left(1 - \frac{\nu}{\nu_0}\right) x(\tilde{G}'_m), \qquad (4.58)$$

$$y(I'_m) = \frac{\nu}{\nu_0} y(C'_m) + \left(1 - \frac{\nu}{\nu_0}\right) y(\tilde{G}'_m). \qquad (4.59)$$

The value of I'_m can be obtained using the expansion from G'_i as

$$u(I'_m) = u\big(x(I'_m) - x(G'_i), y(I'_m) - y(G'_i), 0\big)_{G'_i}, \qquad (4.60)$$

Fig. 4.6 Definition of points in instructed mesh for CNI scheme
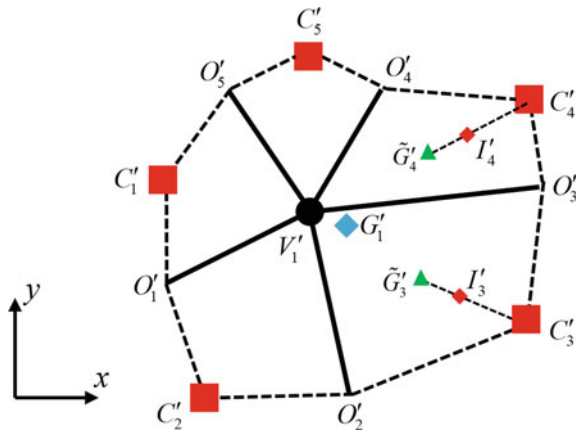
where $u(I_m')$ can be calculated as

$$u(I_m') = u\left(x(I_m') - x(C_m), y(I_m') - y(C_m), \frac{\Delta t}{2}\right)_{C_m}.$$ (4.61)

With the aid of Eqs. (4.60) and (4.61) to each I_m'. Every two neighboring I_m' explicitly provide one set of $u_x(G_i')$ and $u_y(G_i')$. Then, the optimal derivatives can be weightedly averaged from these M sets of derivatives.

4.2.3   Upwind CESE Scheme

In the upwind CESE scheme, the definition of SE is modified and SE(V_i) is the polyhedron $C_1' O_1' C_2'...C_M' O_M' C_1'' P'' O_1'' C_2''...C_M'' O_M''$ (Fig. 4.4b). The values of u, f, and g are continuous within each SE. However, the interface flux $\widehat{F}_m$ between two neighboring sub-CEs may be discontinuous. The flux $\widehat{F}_m$ can be evaluated in an upwind manner in the normal direction as

$$\hat{F}_m = \hat{F}(u_L, u_R)$$ (4.62)

where $\hat{F}$ denotes the Riemann solver, and the values at the center of the interface, u_L and u_R, are reconstructed from SE(C_m) and SE(C_{m+1}) with a similar procedure as the 2D upwind CESE scheme on a uniform mesh. The WBAP limiter (Eq. (3.42)) is used to reconstruct the derivatives. Once the fluxes across inner interfaces are determined, the averaged value on the top surface of mth sub-CE U_m' in Eq. (4.49) can be explicitly computed. We can therefore relate U_m' to $u(G_i')$ using the Taylor expansion as

$$U_m' = u(G_i') + u_x(G_i')\left[x(G_m') - x(G_i')\right] + u_y(G_i')\left[y(G_m') - y(G_i')\right]$$ (4.63)

Two linear equations can be derived for each set of m and $m + 1$. Accordingly, a similar procedure for averaging the derivatives in the central scheme can be directly applied here.

4.3   Numerical Examples

Two canonical problems are selected to demonstrate the performance of the a–α, CNI, and upwind CESE schemes on either structured or unstructured meshes. For Euler equations, the above schemes can be directly applied to solve each component of the conserved variables. The interface fluxes in the upwind CESE scheme can be solved as a Riemann problem, where Riemann solvers including Harten, Lax and van Leer (HLL) Riemann solver, the contact discontinuity restoring HLLC Riemann solver, and the Roe Riemann solver, can be applied to solve the local Riemann problem. Since the upwind direction may not be perpendicular to the grid-aligned direction, here we use the rotated HLLC Riemann solver. The first example is the Mach 3 wind tunnel problem with step. The computational domain is [0, 3] × [0, 1] with a step of height 0.2 placed at 0.6 from the entrance. The inflow boundary condition is imposed at the left boundary, and supersonic outflow is applied at the right boundary. The upper and lower boundaries are treated as reflective boundaries. Initially, the primitive variables (ρ, u, v, p) are set as (1.4, 3, 0, 1). Figure 4.7 depicted the simulations using different CESE schemes with 1800×600 structured cartesian meshes. Both the CNI scheme and the upwind scheme can capture the shear layer instability that is missed in the result from a–α scheme.

The second example is the double Mach reflection problem. A Mach 10 shock propagates to the right and reflects over a solid wedge with an incline angle of 30 °. A supersonic inflow boundary condition was implemented on the left boundary. Reflective and non-reflective boundary conditions were applied on the lower and upper boundary, respectively. The problem was solved using unstructured quadrilateral meshes with a mesh size of 1/400. Figure 4.8 compares the results by applying different schemes at computation time $t = 0.2$. All the three schemes can accurately describe the structures while the upwind scheme surpasses with finer resolved feature near the wall-adjacent jet.

(a) a-αCESE

(b) Rotated HLLC CESE

(c) CNI CESE
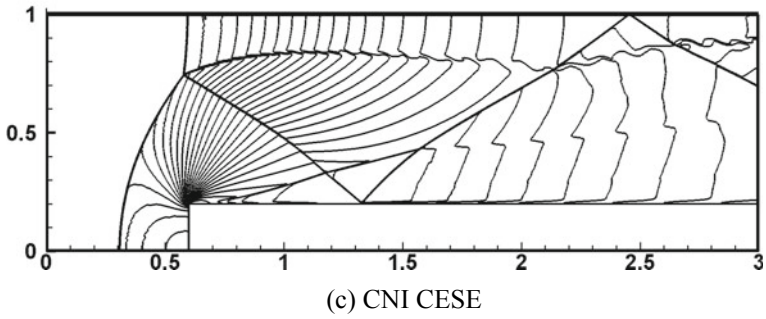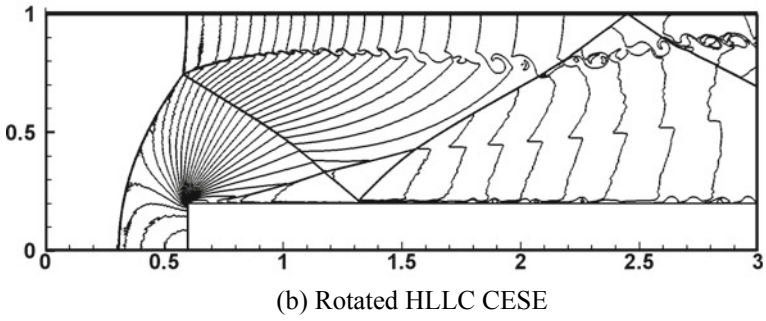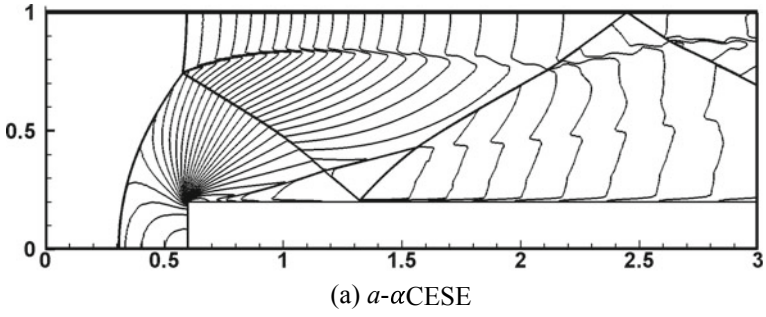
Fig. 4.7 Mach 3 wind tunnel with a step simulated by different CESE schemes with 1800×600 mesh grids: density contours at $t = 4.0$

Fig. 4.8 Double Mach
reflection problem computed
by different CESE schemes
with unstructured
quadrilateral meshes: density
contours at $t = 0.2$

(a) a-αCESE

(b) Rotated HLLC CESE

(c) CNI CESE
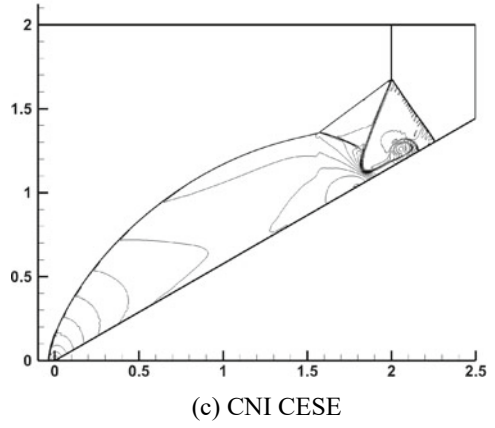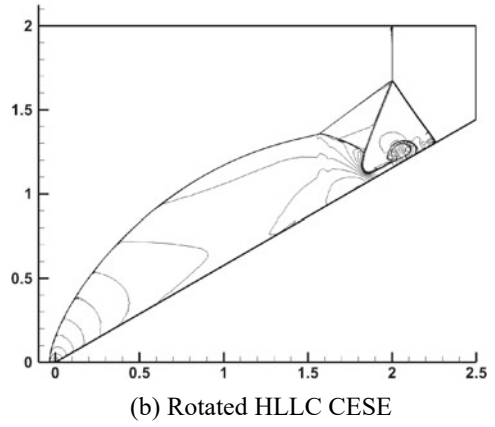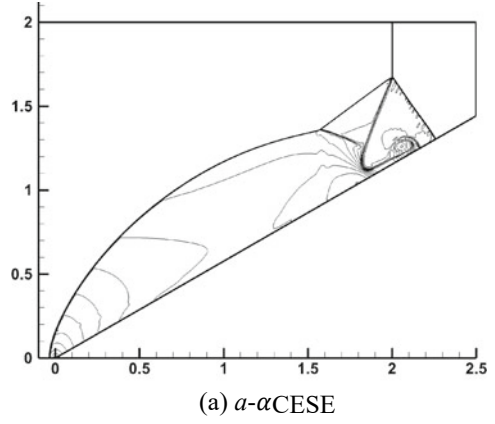
References

1. Chang, S. C., Wang, X. Y., & Chow, C. Y. (1999). The space-time conservation element and solution element method: A new high-resolution and genuinely multidimensional paradigm for solving conservation laws. *Journal of Computational Physics, 156*(1), 89–136.
2. Zhang, Z. C., Yu, S. J., & Chang, S. C. (2002). A space-time conservation element and solution element method for solving the two-and three-dimensional unsteady Euler equations using quadrilateral and hexahedral meshes. *Journal of Computational Physics, 175*(1), 168–199.
3. Wang, J. T., Liu, K. X., & Zhang, D. L. (2009). An improved CE/SE scheme for multi-material elastic–plastic flows and its applications. *Computers & fluids, 38*(3), 544–551.
4. Wang, G., Zhang, D. L., Liu, K. X., & Wang, J. T. (2010). An improved CE/SE scheme for numerical simulation of gaseous and two-phase detonations. *Computers & fluids, 39*(1), 168–177.
5. Chang, S.C., & Wang, X.Y. (2003). Multi-dimensional Courant number insensitive Euler solvers for applications involving highly nonuniform meshes. In *39th AIAA/ASME/SAE/ASEE Joint Propulsion Conference and Exhibit*.
6. Shen, H., & Wen, C. Y. (2016). A characteristic space–time conservation element and solution element method for conservation laws II. Multidimensional extension. *Journal of Computational Physics, 305*, 775–792.
7. Shen, H., Wen, C. Y., Liu, K. X., & Zhang, D. L. (2015). Robust high-order space–time conservative schemes for solving conservation laws on hybrid meshes. *Journal of Computational Physics, 281*, 375–402.
8. Shen, H., & Parsani, M. (2018). Positivity-preserving CE/SE schemes for solving the compressible Euler and Navier-Stokes equations on hybrid unstructured meshes. *Computer Physics Communications, 232*, 165–176.

Chapter 5
High-Order CESE Schemes

This chapter is dedicated to the description of high-order CESE schemes. In the second-order CESE schemes, the first-order Taylor expansion was employed to approximate the unknowns and fluxes within the solution elements. The accuracy of the scheme mainly depends on the approximations on the surfaces of the conservation elements. Analogously, the high-order CESE schemes with Mth-order in space and time are generally derived from $(M-1)$th-order Taylor expansions in the solution elements. The high-order CESE schemes use a highly compact stencil. Furthermore, spatial and temporal high-order accuracy can be achieved simultaneously. We shall start with constructing a 1D high-order scheme and then extend it to multi-dimensional schemes.

5.1   Construction of High-Order CESE Schemes

Several attempts have been made to obtain higher-order accuracy for CESE schemes. One of the primary advantages of the high-order CESE schemes is the usage of the most compact stencil. High-order accuracy is achieved by the approximation with high-order Taylor expansions. For example, $a(3)$ scheme [1] has 4th order of accuracy by applying second-order Taylor expansion, and it is stable for $v < 0.5$. Furthermore, $a(4)$ scheme [2] with 4th to 5th order of accuracy was developed by defining more CEs at each grid point. The CFL number needs to be constrained below 1/3. The constructions of high-order schemes from the above two approaches were limited to only 1D scenarios. Alternatively, Chang [3] proposed a high-order scheme that can be extended to arbitrary order with CFL limited below 1. In this scheme, the even derivatives are advanced in the same manner as the conserved variables, whereas the odd derivatives are advanced through finite difference. This method has been extended to solve Euler equations [4]. In this section, we follow the approaches of Liu and Wang [5] in deriving the high-order 1D scheme, Wang et al. [6] for 2D high-order

schemes on uniform meshes, and Shen et al. [7] for the 2D high-order scheme for hybrid meshes. In these schemes, only the conserved variables are computed through the integration on the CE surfaces, while all the physical derivatives are computed through the finite difference of lower-order derivatives. In the following text, we will present the constructions of the third-order CESE schemes. The construction of arbitrary order schemes could be achieved by implementing the same strategy, and details can be accessed in Shen et al. [7] and Yang et al. [8].

5.1.1 Construction of a Third-Order 1D CESE Scheme

Recall the 1D scalar problem

$$\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = 0 \tag{5.1}$$

with $f = au$. To formulate a third-order scheme, $u(x, t)$ and $f(x, t)$ in any $(x, t) \in \text{SE}(j, n)$ are approximated by the second-order Taylor expansion as

$$u(x, t) = u_j^n + (u_x)_j^n(x - x_j) + (u_t)_j^n(t - t_n) + (u_{xt})_j^n(x - x_j)(t - t_n)$$
$$+ \frac{1}{2}(u_{xx})_j^n(x - x_j)^2 + \frac{1}{2}(u_{tt})_j^n(t - t_n)^2, \quad (x, t) \in (\text{SE})_j^n \tag{5.2}$$

$$f(x, t) = f_j^n + (f_x)_j^n(x - x_j) + (f_t)_j^n(t - t_n) + (f_{xt})_j^n(x - x_j)(t - t_n)$$
$$+ \frac{1}{2}(f_{xx})_j^n(x - x_j)^2 + \frac{1}{2}(f_{tt})_j^n(t - t_n)^2. \quad (x, t) \in (\text{SE})_j^n \tag{5.3}$$

where $u_{xt}, u_{xx}, u_{tt}, f_{xt}, f_{xx},$ and f_{tt} are the second-order derivatives of u and f, respectively. These derivatives are assumed constant within the SE. With the aid of the Eqs. (5.1) and (5.2), then one has

$$(u_t)_j^n = -a(u_x)_j^n, \quad (u_{xt})_j^n = -a(u_{xx})_j^n, \quad (u_{tt})_j^n = a^2(u_{xx})_j^n. \tag{5.4}$$

It implies that u, u_x, u_{xx} are the only independent variables associated with each grid point. Substituting Eqs. (5.2) and (5.3) into the integral form of Eq. (5.1). The following algebraic relation can be derived as

$$u_j^n \Delta x + (u_{xx})_j^n \frac{\Delta x^3}{24} = U_L + U_R + F_L - F_R, \tag{5.5}$$

where

$$
\begin{aligned}
U_L &= \tfrac{\Delta x}{2} u_{j-1/2}^{n-1/2} + \tfrac{\Delta x^2}{8}(u_x)_{j-1/2}^{n-1/2} + \tfrac{\Delta x^3}{48}(u_{xx})_{j-1/2}^{n-1/2} \\
U_R &= \tfrac{\Delta x}{2} u_{j+1/2}^{n-1/2} - \tfrac{\Delta x^2}{8}(u_x)_{j+1/2}^{n-1/2} + \tfrac{\Delta x^3}{48}(u_{xx})_{j+1/2}^{n-1/2} \\
F_L &= \tfrac{\Delta t}{2} f_{j-1/2}^{n-1/2} + \tfrac{\Delta t^2}{8}(f_t)_{j-1/2}^{n-1/2} + \tfrac{\Delta t^3}{48}(f_{tt})_{j-1/2}^{n-1/2} \\
F_R &= \tfrac{\Delta t}{2} f_{j+1/2}^{n-1/2} + \tfrac{\Delta t^2}{8}(f_t)_{j+1/2}^{n-1/2} + \tfrac{\Delta t^3}{48}(f_{tt})_{j+1/2}^{n-1/2}
\end{aligned}
\tag{5.6}
$$

In Eqs. (5.5) and (5.6), the values at $t = t_{n-1/2}$ are already known. It is necessary to calculate $(u_{xx})_j^n$ at $t = t_n$ before explicitly computing u_j^n. The second-order derivative, $(u_{xx})_j^n$, is approximated by a central difference as

$$
(u_{xx})_j^n = \frac{(u_x)_{j+1/2}^n - (u_x)_{j-1/2}^n}{\Delta x}.
\tag{5.7}
$$

Then u_j^n can be directly computed from Eq. (5.5), and $(u_x)_j^n$ can be calculated in the same manner as the second-order a-α schemes using the weighted average function.

5.1.2   Construction of a Third-Order 2D CESE Scheme on Uniform Mesh

Recall the 2D scalar hyperbolic conservation law

$$
\frac{\partial u}{\partial t} + \frac{\partial f}{\partial x} + \frac{\partial g}{\partial y} = 0.
\tag{5.8}
$$

which can be cast into the integral form

$$
\oint_{S(CE(P'))} \boldsymbol{h} \cdot \boldsymbol{n}\, \mathrm{d}S = 0
\tag{5.9}
$$

where $\boldsymbol{h} = (f, g, u)$ is the space–time flux vector, $\boldsymbol{n}$ is the unit outward normal vector on the surface of the control volume. Inspired by the construction of the high-order 1D scheme, second-order Taylor expansion is utilized inside the SE to estimate u, f, and g. The definitions of SE and CE are kept the same as the second-order schemes (Fig. 4.1). For a point (x, y, t) that belongs to SE(P'),

$$
u(x, y, t) = u(\delta x, \delta y, \delta t)_{P'}, \quad (x, y, t) \in SE(P')
\tag{5.10}
$$

$$
f(x, y, t) = f(\delta x, \delta y, \delta t)_{P'}, \quad (x, y, t) \in SE(P')
\tag{5.11}
$$

$$
g(x, y, t) = g(\delta x, \delta y, \delta t)_{P'}, \quad (x, y, t) \in SE(P')
\tag{5.12}
$$

where $X(\delta x, \delta y, \delta t)_N$ denotes the second-order Taylor expansion about point N as

$$
\begin{aligned}
X(\delta x, \delta y, \delta t)_N &= X_N + (X_x)_N \delta x + (X_y)_N \delta y + (X_t)_N \delta t \\
&\quad + \frac{1}{2}(X_{xx})_N (\delta x)^2 + \frac{1}{2}(X_{yy})_N (\delta y)^2 + \frac{1}{2}(X_{tt})_N (\delta t)^2 \\
&\quad + (X_{xy})_N \delta x \delta y + (X_{yt})_N \delta y \delta t + (X_{xt})_N \delta x \delta t.
\end{aligned} \tag{5.13}
$$

and

$$
\delta x = x - x_N, \quad \delta y = y - y_N, \quad \delta t = t - t_N. \tag{5.14}
$$

Substituting Eqs. (5.10)–(5.12) into (5.8), one obtains

$$
\begin{aligned}
u_t &= -f_x - g_y \\
u_{tt} &= -f_{xt} - g_{yt} \\
u_{xt} &= -f_{xx} - g_{xy} \\
u_{yt} &= -f_{xy} - g_{yy}
\end{aligned}. \tag{5.15}
$$

Together with the chain rule, to compute the derivatives in Eq. (5.13), the only unknowns are u, u_x, u_y, u_{xx}, u_{yy}, and u_{xy}. By integrating over the surface of the CE(P'), Eq. (5.9) leads to

$$
\begin{aligned}
\oint_{S(V)} \mathbf{h} \cdot \mathbf{n} \, \mathrm{d}S &= \int_{A'B'C'D'} \mathbf{h} \cdot \mathbf{n} \, \mathrm{d}S + \int_{ABCD} \mathbf{h} \cdot \mathbf{n} \, \mathrm{d}S + \int_{ABB'A'} \mathbf{h} \cdot \mathbf{n} \, \mathrm{d}S \\
&\quad + \int_{BCC'B'} \mathbf{h} \cdot \mathbf{n} \, \mathrm{d}S + \int_{CDD'C'} \mathbf{h} \cdot \mathbf{n} \, \mathrm{d}S + \int_{DAA'D'} \mathbf{h} \cdot \mathbf{n} \, \mathrm{d}S = 0.
\end{aligned} \tag{5.16}
$$

Thereafter, with the aid of Eqs. (5.10) –(5.12) and (5.16) can be rearranged as

$$
(u)_{P'} + \frac{(\Delta x)^2}{24}(u_{xx})_{P'} + \frac{(\Delta y)^2}{24}(u_{yy})_{P'} = \frac{1}{4}\left(\bar{u} + \frac{\Delta t}{\Delta x}\bar{f} + \frac{\Delta t}{\Delta y}\bar{g}\right), \tag{5.17}
$$

where

$$
\begin{aligned}
\bar{u} =& \hat{u}\left(\frac{\Delta x}{4}, \frac{\Delta y}{4}, 0\right)_A + \hat{u}\left(-\frac{\Delta x}{4}, \frac{\Delta y}{4}, 0\right)_B \\
&+ \hat{u}\left(-\frac{\Delta x}{4}, -\frac{\Delta y}{4}, 0\right)_C + \hat{u}\left(\frac{\Delta x}{4}, -\frac{\Delta y}{4}, 0\right)_D, \\
\bar{f} =& \hat{f}\left(0, \frac{\Delta y}{4}, \frac{\Delta t}{4}\right)_A - \hat{f}\left(0, \frac{\Delta y}{4}, \frac{\Delta t}{4}\right)_B
\end{aligned} \tag{5.18}
$$

$$- \hat{f}\left(0, -\frac{\Delta y}{4}, \frac{\Delta t}{4}\right)_C + \hat{f}\left(0, -\frac{\Delta y}{4}, \frac{\Delta t}{4}\right)_D, \tag{5.19}$$

$$\bar{g} = \hat{g}\left(\frac{\Delta x}{4}, 0, \frac{\Delta t}{4}\right)_A + \hat{g}\left(-\frac{\Delta x}{4}, 0, \frac{\Delta t}{4}\right)_B$$
$$- \hat{g}\left(-\frac{\Delta x}{4}, 0, \frac{\Delta t}{4}\right)_C - \hat{g}\left(\frac{\Delta x}{4}, 0, \frac{\Delta t}{4}\right)_D, \tag{5.20}$$

and

$$\hat{u}(\delta x, \delta y, \delta t)_N = (u)_N + (u_x)_N \delta x + \left(u_y\right)_N \delta y + (u_t)_N \delta t$$
$$+ \frac{1}{6}(u_{xx})_N (\delta x)^2 + \frac{1}{6}\left(u_{yy}\right)_N (\delta y)^2 + \left(u_{xy}\right)_N \delta x \delta y, \tag{5.21}$$

$$\hat{f}(\delta x, \delta y, \delta t)_N = (f)_N + (f_x)_N \delta x + \left(f_y\right)_N \delta y + (f_t)_N \delta t$$
$$+ \frac{1}{6}\left(f_{yy}\right)_N (\delta y)^2 + \frac{1}{6}(f_{tt})_N (\delta t)^2 + \left(f_{yt}\right)_N \delta y \delta t, \tag{5.22}$$

$$\hat{g}(\delta x, \delta y, \delta t)_N = (g)_N + (g_x)_N \delta x + \left(g_y\right)_N \delta y + (g_t)_N \delta t$$
$$+ \frac{1}{6}(g_{xx})_N (\delta x)^2 + \frac{1}{6}(g_{tt})_N (\delta t)^2 + (g_{xt})_N \delta x \delta t. \tag{5.23}$$

Prior to explicitly computing $(u)_{p'}$ from Eq. (5.17), $(u_{xx})_{p'}$ and $\left(u_{yy}\right)_{p'}$ are required to be obtained from the finite difference of the interpolations from the previous time step. The second-order derivative $(u_{xx})_{p'}$ can be estimated as

$$(u_{xx})_{p'}^+ = \frac{(u_x)_B + \frac{\Delta t}{2}(u_{xt})_B - (u_x)_A - \frac{\Delta t}{2}(u_{xt})_A}{\Delta x}, \tag{5.24}$$

$$(u_{xx})_{p'}^- = \frac{(u_x)_C + \frac{\Delta t}{2}(u_{xt})_C - (u_x)_D - \frac{\Delta t}{2}(u_{xt})_D}{\Delta x}. \tag{5.25}$$

$(u_{xx})_{p'}$ is then computed from simple average as

$$(u_{xx})_{p'} = \frac{(u_{xx})_{p'}^+ + (u_{xx})_{p'}^-}{2}. \tag{5.26}$$

Similarly, the derivative $\left(u_{yy}\right)_{p'}$ is computed as

$$\left(u_{yy}\right)_{p'}^+ = \frac{\left(u_y\right)_D + \frac{\Delta t}{2}\left(u_{yt}\right)_D - \left(u_y\right)_A - \frac{\Delta t}{2}\left(u_{yt}\right)_A}{\Delta y}, \tag{5.27}$$

$$\left(u_{yy}\right)^-_{p'} = \frac{\left(u_y\right)_C + \frac{\Delta t}{2}\left(u_{yt}\right)_C - \left(u_y\right)_B - \frac{\Delta t}{2}\left(u_{yt}\right)_B}{\Delta y}, \tag{5.28}$$

$$\left(u_{yy}\right)_{p'} = \frac{\left(u_{yy}\right)^+_{p'} + \left(u_{yy}\right)^-_{p'}}{2}. \tag{5.29}$$

Equations (5.17), (5.26), and (5.29) form the procedure to get the values of $(u)_{p'}$, $(u_{xx})_{p'}$ and $\left(u_{yy}\right)_{p'}$ at the new time level. To complete the time marching, we need to further compute u_x, u_y, and u_{xy}. The weighted average of the sided-estimations computes the first-order derivatives as

$$(u_x)^+_{p'} = \frac{1}{\Delta x}\left[u\left(0, 0, \frac{\Delta t}{2}\right)_B + u\left(0, 0, \frac{\Delta t}{2}\right)_C - 2(u_x)_{p'}\right], \tag{5.30}$$

$$(u_x)^-_{p'} = \frac{1}{\Delta x}\left[2(u_x)_{p'} - u\left(0, 0, \frac{\Delta t}{2}\right)_A - u\left(0, 0, \frac{\Delta t}{2}\right)_D\right], \tag{5.31}$$

$$(u_y)^+_{p'} = \frac{1}{\Delta y}\left[u\left(0, 0, \frac{\Delta t}{2}\right)_C + u\left(0, 0, \frac{\Delta t}{2}\right)_D - 2(u_x)_{p'}\right], \tag{5.32}$$

$$(u_y)^-_{p'} = \frac{1}{\Delta y}\left[2(u_y)_{p'} - u\left(0, 0, \frac{\Delta t}{2}\right)_A - u\left(0, 0, \frac{\Delta t}{2}\right)_B\right], \tag{5.33}$$

$$(u_x)_{p'} = W\left((u^-_x)_{P'}, (u^+_x)_{P'}, \alpha\right), \tag{5.34}$$

$$(u_y)_{p'} = W\left((u^-_y)_{P'}, (u^+_y)_{P'}, \alpha\right). \tag{5.35}$$

Meanwhile, the mixed derivative u_{xy} is computed from

$$\left(u_{xy}\right)^+_{p'} = \frac{(u_x)_D + \frac{\Delta t}{2}(u_{xt})_D - (u_x)_A - \frac{\Delta t}{2}(u_{xt})_A}{\Delta y}, \tag{5.36}$$

$$\left(u_{xy}\right)^-_{p'} = \frac{(u_x)_C + \frac{\Delta t}{2}(u_{xt})_C - (u_x)_B - \frac{\Delta t}{2}(u_{xt})_B}{\Delta y}, \tag{5.37}$$

$$\left(u_{yx}\right)^+_{p'} = \frac{(u_x)_B + \frac{\Delta t}{2}(u_{xt})_B - (u_x)_A - \frac{\Delta t}{2}(u_{xt})_A}{\Delta x}, \tag{5.38}$$

$$\left(u_{yx}\right)^-_{p'} = \frac{(u_x)_C + \frac{\Delta t}{2}(u_{xt})_C - (u_x)_D - \frac{\Delta t}{2}(u_{xt})_D}{\Delta x}, \tag{5.39}$$

$$\left(u_{xy}\right)_{p'} = \frac{1}{4}\left[\left(u_{xy}\right)^+_{p'} + \left(u_{xy}\right)^-_{p'} + \left(u_{yx}\right)^+_{p'} + \left(u_{yx}\right)^-_{p'}\right]. \tag{5.40}$$

5.1.3 Construction of a Third-Order 2D CESE Scheme on Unstructured Mesh

Similar to constructing the third-order scheme on a a uniform mesh, second-order Taylor expansion is utilized here, and the variables stored at each grid point are still $u, u_x, u_y, u_{xx}, u_{yy}$, and u_{xy}. Now we recollect the definitions of SE and CE in the 2D CESE scheme on the unstructured mesh as depicted in Fig. 4.4, and the flux balancing relation was derived in Eq. (4.48). From the summation of the conservation law for all the sub-CEs, we arrive at the flux balancing relation for CE(V_i'):

$$
\iint\limits_{C_1'O_1'\cdots C_M'O_M'} u\, d\sigma = \sum_{m=1}^{M} \iint\limits_{C_m O_m V_i O_{m-1}} u\, d\sigma - \sum_{m=1}^{M} \iint\limits_{C_m O_m O_m' C_m' + O_{m-1} C_m C_m' O_{m-1}'} \mathbf{V} \cdot \mathbf{n} d\sigma.
$$

(5.41)

The LHS of Eq. (5.41) represents the integration of u on the top surface of CE(V_i'), i.e.,

$$
\iint\limits_{C_1'O_1'\cdots C_M'O_M'} u\, d\sigma = u(G_i') \cdot S + \text{TERM1} = u(G_i') \cdot S + \sum_{m=1}^{M} \iint\limits_{C_m'O_m'G_i'O_{m-1}'} \frac{1}{2}\frac{\partial^2 u(G_i')}{\partial x^2}(\delta x)^2
$$
$$
+ \frac{1}{2}\frac{\partial^2 u(G_i')}{\partial y^2}(\delta y)^2 + \frac{\partial^2 u(G_i')}{\partial x \partial y}(\delta x)(\delta y)dxdy,
$$

(5.42)

where $\delta x = x - x(G_i')$, $\delta y = y - y(G_i')$, and S represents the area of the polygon $C_1'O_1'\ldots C_M'O_M'$. The second term in RHS (TERM1) of Eq. (5.42) denotes the integration of the high-order terms of u. The two terms in RHS of Eq. (5.41) can be respectively expressed as

$$
\sum_{m=1}^{M} \iint\limits_{C_m O_m V_i O_{m-1}} u\, d\sigma = \text{TERM2} = \sum_{m=1}^{M} \iint\limits_{C_m O_m V_i O_{m-1}} u(C_m) + \frac{\partial u(C_m)}{\partial x}(\delta x) + \frac{\partial u(C_m)}{\partial y}(\delta y)
$$
$$
+ \frac{1}{2}\frac{\partial^2 u(C_m)}{\partial x^2}(\delta x)^2 + \frac{1}{2}\frac{\partial^2 u(C_m)}{\partial y^2}(\delta y)^2 + \frac{\partial^2 u(C_m)}{\partial x \partial y}(\delta x)(\delta y)dxdy,
$$

(5.43)

where $\delta x = x - x(C_m)$, $\delta y = y - y(C_m)$.

$$
\sum_{m=1}^{M} \iint\limits_{C_m O_m O_m' C_m' + O_{m-1} C_m C_m' O_{m-1}'} \mathbf{V} \cdot \mathbf{n} d\sigma = \text{TERM3} = \sum_{m=1}^{M} \left[FLUX_1^{(m)} + FLUX_2^{(m)} \right],
$$

(5.44)

with

$$FLUX_1^{(m)} = \frac{\Delta t}{2} \Delta y_1^{(m)} \sum_{a=0}^{2} \sum_{b=0}^{2-a} \sum_{c=0}^{2-a-b} \frac{1}{(a+b+1)a!b!(c+1)!} \frac{\partial^{a+b+c} f(C_m)}{\partial x^a \partial y^b \partial t^c} \left[\Delta x_1^{(m)} \right]^a \left[\Delta y_1^{(m)} \right]^b \left[\frac{\Delta t}{2} \right]^c$$

$$- \frac{\Delta t}{2} \Delta x_1^{(m)} \sum_{a=0}^{2} \sum_{b=0}^{2-a} \sum_{c=0}^{2-a-b} \frac{1}{(a+b+1)a!b!(c+1)!} \frac{\partial^{a+b+c} g(C_m)}{\partial x^a \partial y^b \partial t^c} \left[\Delta x_1^{(m)} \right]^a \left[\Delta y_1^{(m)} \right]^b \left[\frac{\Delta t}{2} \right]^c,$$

$$\tag{5.45}$$

$$FLUX_2^{(m)} = \frac{\Delta t}{2} \Delta y_2^{(m)} \sum_{a=0}^{2} \sum_{b=0}^{2-a} \sum_{c=0}^{2-a-b} \frac{1}{(a+b+1)a!b!(c+1)!} \frac{\partial^{a+b+c} f(C_m)}{\partial x^a \partial y^b \partial t^c} \left[-\Delta x_2^{(m)} \right]^a \left[-\Delta y_2^{(m)} \right]^b \left[\frac{\Delta t}{2} \right]^c$$

$$- \frac{\Delta t}{2} \Delta x_2^{(m)} \sum_{a=0}^{2} \sum_{b=0}^{2-a} \sum_{c=0}^{2-a-b} \frac{1}{(a+b+1)a!b!(c+1)!} \frac{\partial^{a+b+c} g(C_m)}{\partial x^a \partial y^b \partial t^c} \left[-\Delta x_2^{(m)} \right]^a \left[-\Delta y_2^{(m)} \right]^b \left[\frac{\Delta t}{2} \right]^c,$$

$$\tag{5.46}$$

in which $\Delta x_1^{(m)} = x(O_m) - x(C_m)$, $\Delta y_1^{(m)} = y(O_m) - y(C_m)$, $\Delta x_2^{(m)} = x(C_m) - x(O_{m-1})$, $\Delta y_2^{(m)} = y(C_m) - y(O_{m-1})$. $FLUX^{(m)}_1$ and $FLUX^{(m)}_2$ represent the fluxes across the adjacent CE surfaces on which the points are defined within $SE(C_m)$.

By substituting Eqs. (5.42)–(5.44) into (5.41) and moving the high-order terms to RHS. Consequently, the value of $u(G'_i)$ can be expressed by

$$u(G'_i) = \frac{1}{S}(\text{TERM2} - \text{TERM3} - \text{TERM1}). \tag{5.47}$$

Two remarks should be noted regarding Eq. (5.47). First of all, since u, f, and g are nonlinearly distributed on the surfaces of CE, the integral terms appeared in Eqs. (5.42) and (5.43) are not simple multiplications of the value at the centroid and the surface area as what has been implemented in the second-order scheme. In contrast, they are now calculated using the coordinate transformation method:

$$x = \sum_{i=1}^{4} x_i N_i(\xi, \eta), \tag{5.48}$$

$$y = \sum_{i=1}^{4} y_i N_i(\xi, \eta), \tag{5.49}$$

where

$$N_1 = \frac{1}{4}(1 - \xi)(1 - \eta), \qquad N_2 = \frac{1}{4}(1 + \xi)(1 - \eta),$$

$$N_3 = \frac{1}{4}(1 + \xi)(1 + \eta), \qquad N_4 = \frac{1}{4}(1 - \xi)(1 + \eta). \tag{5.50}$$

Thereafter, the integration of an arbitrary function $\phi(x, y)$ over the irregular quadrangle A is transformed into the integration over a rectangle (Fig. 5.1) as
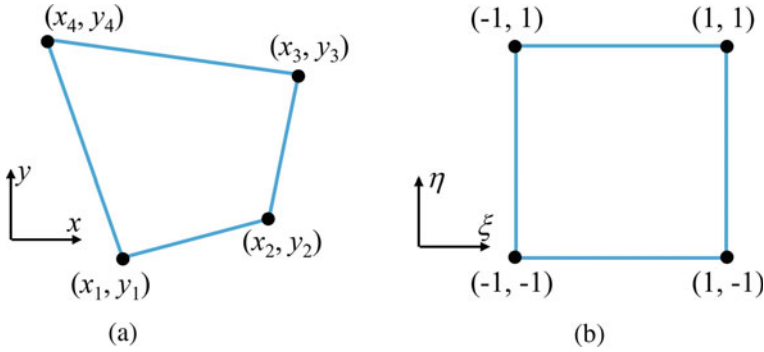
Fig. 5.1 Sketch of the coordinate transformation

$$\iint_A \phi(x, y)ds = \int_{-1}^{1} d\eta \int_{-1}^{1} \phi[x(\xi, \eta), y(\xi, \eta)]\mathbf{J}d\xi, \tag{5.51}$$

where $\mathbf{J}$ is the Jacobian matrix of the coordinate transformation,

$$\mathbf{J} = \frac{\partial(x, y)}{\partial(\xi, \eta)} = \begin{vmatrix} \sum_{i=1}^{4} x_i \frac{\partial N_i}{\partial \xi} & \sum_{i=1}^{4} y_i \frac{\partial N_i}{\partial \xi} \\ \sum_{i=1}^{4} x_i \frac{\partial N_i}{\partial \eta} & \sum_{i=1}^{4} y_i \frac{\partial N_i}{\partial \eta} \end{vmatrix}. \tag{5.52}$$

In addition, TERM2 and TERM3 in Eq. (5.47) can be computed from the information at $t = t_{n-1/2}$. TERM1 consists of high-order terms (u_{xx}, u_{yy}, and u_{xy}) of u at $t = t_n$. As a result, the high-order derivatives must be solved prior to computing u. The idea to calculate the high-order derivatives are analogue to that in Sects. 5.1.1 and 5.1.2. For example, from the Taylor expansion C'_m we have

$$u_x\left(C'_{m-1}\right) - u_x\left(C'_m\right) = u_{xx}\left(C'_m\right)\left[x(C'_{m-1}) - x(C'_m)\right] + u_{xy}\left(C'_m\right)\left[y(C'_{m-1}) - y(C'_m)\right], \tag{5.53}$$

$$u_x\left(C'_{m+1}\right) - u_x\left(C'_m\right) = u_{xx}\left(C'_m\right)\left[x(C'_{m+1}) - x(C'_m)\right] + u_{xy}\left(C'_m\right)\left[y(C'_{m+1}) - y(C'_m)\right]. \tag{5.54}$$

On the other hand, $u_x(C'_{m-1}), u_x(C'_m), u_x(C'_{m+1})$ can be interpolated from the previous time level,

$$u_x\left(C'_{m-1}\right) = u_x(C_{m-1}) + u_{xx}(C_{m-1})\frac{\Delta t}{2}$$

$$u_x\left(C'_m\right) = u_x(C_m) + u_{xx}(C_m)\frac{\Delta t}{2}$$

$$u_x\left(C'_{m+1}\right) = u_x(C_{m+1}) + u_{xx}(C_{m+1})\frac{\Delta t}{2} \tag{5.55}$$

By substituting Eq. (5.55) into Eqs. (5.53) and (5.54), a pair of equations with two unknowns ($u_{xx}(C'_m), u_{xy}(C'_m)$) are ready to be solved by Cramer's rule. A simple

average (or weighted average to suppress oscillations near discontinuities) can be applied for M pairs of derivatives. Once all the second-order derivatives are obtained under the same routine, $u(G'_i)$ can be immediately calculated from Eq. (5.47). The last unknowns remain to be computed are u_x and u_y, which can be calculated similarly in Sect. 4.2, e.g.,

$$
\begin{aligned}
&u(G'_i) + u_x(G'_i)\delta x + u_y(G'_i)\delta y + \tfrac{1}{2}u_{xx}(G'_i)(\delta x)^2 + \tfrac{1}{2}u_{yy}(G'_i)(\delta y)^2 \\
&+u_{xy}(G'_i)(\delta x)(\delta y) = u(C_m) + u_t(C_m)\tfrac{\Delta t}{2} + \tfrac{1}{2}u_{tt}(C_m)\left(\tfrac{\Delta t}{2}\right)^2
\end{aligned}
, \qquad (5.56)
$$

where $\delta x = x(C_m) - x(G'_i)$, $\delta y = y(C_m) - y(G'_i)$. Similarly, on applying Eq. (5.56) to C_{m+1}, a pair of u_x and u_y can be easily determined. Then, we apply the weighted average function to obtain the optimal derivatives. Finally, $u(V'_i)$ and its derivatives are interpolated by Taylor expansion from the information at $u(G'_i)$.

5.2 Numerical Examples

The Shu-Osher problem consists of a Mach 3 shock interacting with an entropy wave, which requires the capability of capturing fine structures in the flow. The computational domain [0, 10] is discretized by a uniform grid size of 0.05, the non-reflection boundary condition is imposed on both sides, and the initial condition is described as

$$
(\rho,\ u,\ p) = \begin{cases} (3.857143,\ 2.629369,\ 10.33333) & x \le 1 \\ (1 + 0.2\sin(5x),\ 0,\ 1) & x > 1 \end{cases} \qquad (5.57)
$$

The density distributions at $t = 1.8$ are depicted in (Fig. 5.2). Compared to the benchmark solution, which is calculated with a-α scheme using very high grid resolution, higher-order CESE schemes are less dissipative and can reveal better flow structures.

The second example is the Mach 3 wind tunnel problem with step. This problem has been described in Sect. 4.3. Here, we use the identical numerical setups except with a mesh size of 1/250. As depicted in Fig. 5.3, vortex-like structures generated in the results using high-order schemes, however, disappeared in the low-order schemes due to the increasing dissipation. It is not surprising that the triangular mesh result surpasses the quadrilateral ones because the mesh area is only half of the latter when the mesh sizes are the same.
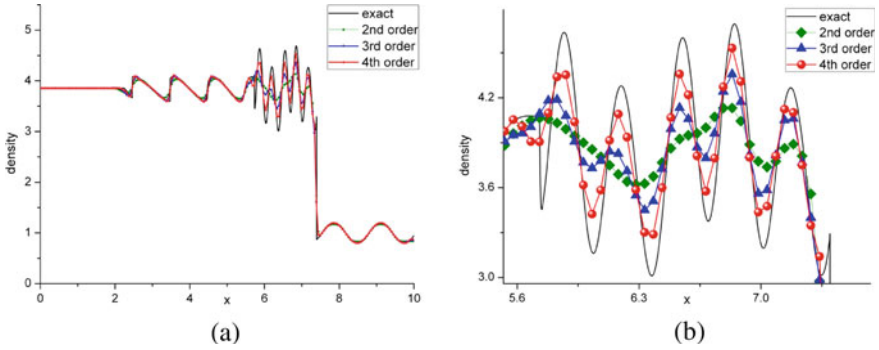
Fig. 5.2 Density distributions for Shu-Osher problem at $t = 1.8$. **a** Entire view, **b** Enlargement. Courtesy of Shen [7]
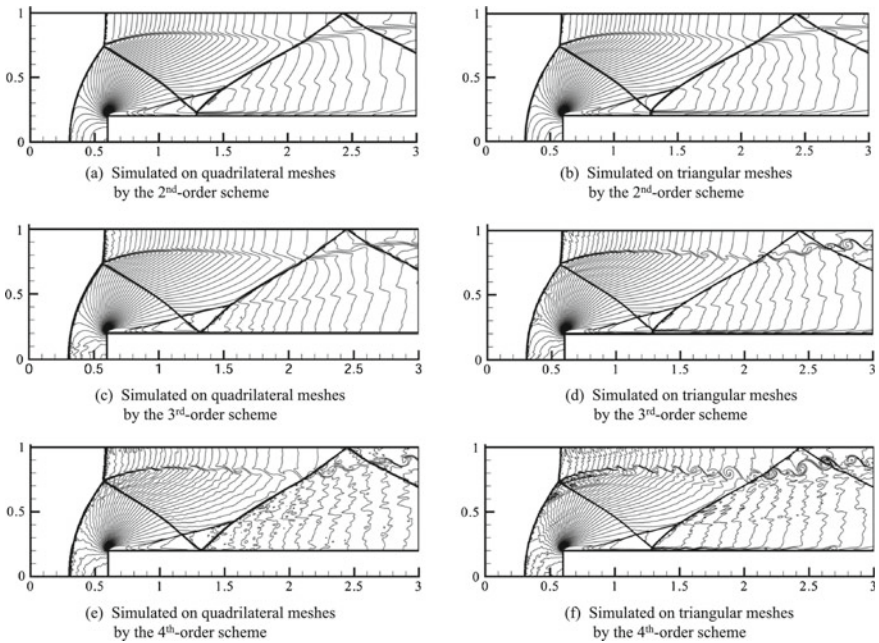
Fig. 5.3 Density contours for Mach 3 wind tunnel with a step at $t = 4.0$. Courtesy of Shen [7]

References

1. Chang, S.C. (2007). The $a(3)$ Scheme: A fourth-order neutrally stable CESE solver. In *18th AIAA Computational Fluid Dynamics Conference*.
2. Chang, S.C. (2009). The $a(4)$ Scheme: A high order neutrally stable CESE solver. In *43rd AIAA/ASME/SAE/ASEE Joint Propulsion Conference & Exhibit*.
3. Chang, S.C. (2010). A new approach for constructing highly stable high order CESE schemes.

In *48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition.*

4. Bilyeu, D., Chen, Y.Y. & Yu, S.T.J. (2011). High-order CESE methods for the Euler equations. In *49th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition.*

5. Liu, K. X., & Wang, J. T. (2004). Analysis of high accuracy conservation-element and solution-element schemes. *Chinese Physics Letters, 21*(11), 2085.

6. Wang, G., Zhang, D. L., & Liu, K. X. (2007). An improved CE/SE scheme and its application to detonation propagation. *Chinese Physics Letters, 24*(12), 3563.

7. Shen, H., Wen, C. Y., Liu, K. X., & Zhang, D. L. (2015). Robust high-order space–time conservative schemes for solving conservation laws on hybrid meshes. *Journal of Computational Physics, 281*, 375–402.

8. Yang, Y., Feng, X. S., & Jiang, C. W. (2017). A high-order CESE scheme with a new divergence-free method for MHD numerical simulation. *Journal of Computational Physics, 349*, 561–581.

Chapter 6
Numerical Features of CESE Schemes

In this chapter, some remarks are made on the numerical characteristics of the CESE schemes described in foregoing chapters. Due to the special formulation, rigorous analysis of the CESE schemes will take more efforts than that of traditional finite difference schemes. However, it is possible to extend the widely used modified equation analysis, modified wavenumber analysis, and von Neumann stability analysis to the CESE schemes. Here, a stability analysis for the second-order upwind CESE scheme applied to the linear scalar convection equation is presented. Additionally, a rough estimation of the algorithm complexity is presented to show the computational efficiency of the CESE method. In order to demonstrate the accuracy of the CESE method in comparison with established numerical approaches, simulation results of some canonical problems are shown.

6.1 Stability

All CESE schemes in this book are explicit time-stepping schemes. Similar to other explicit schemes, the numerical stability of CESE schemes is closely related to the Courant–Friedrichs–Lewy (CFL) condition. Only when the CFL condition is satisfied, the system can remain numerically stable. Consequently, the time-step size used by a CESE scheme has to be restricted according to the upper bound of the stable CFL number ν for that scheme.

It has been proven in [1, 2] that the stability conditions for the second-order a scheme (Sect. 2.3) and the second-order a–α scheme (Sect. 3.1) are both $\nu \leq 1$. For the second-order upwind CESE scheme (Sect. 3.3), the stable upper bound of the CFL number is also unity. This stability analysis is first given by Jiang et al. [3] and introduced as follows.

For illustrative purpose, we consider the simplest upwind CESE scheme applied to the 1D linear scalar convection equation

$$\frac{\partial u}{\partial t} + a\frac{\partial u}{\partial x} = 0 \tag{6.1}$$

where a is a positive constant. Without using any slope limiter, the upwind CESE scheme for Eq. (6.1) can be written in a matrix form as

$$\boldsymbol{q}_j^n = \boldsymbol{Q}_L \boldsymbol{q}_{j-1/2}^{n-1/2} + \boldsymbol{Q}_R \boldsymbol{q}_{j+1/2}^{n-1/2} \tag{6.2}$$

with the vector of time-marching variables

$$\boldsymbol{q}_j^n = \begin{bmatrix} u_j^n \\ \dfrac{\Delta x}{4}(u_x)_j^n \end{bmatrix}, \tag{6.3}$$

and the coefficient matrices (recall Chap. 3)

$$\boldsymbol{Q}_L = \frac{1}{2}\begin{bmatrix} 1+\nu & 1-\nu^2 \\ -1+\nu & -1+4\nu-\nu^2 \end{bmatrix}, \qquad \boldsymbol{Q}_R = \frac{1}{2}\begin{bmatrix} 1-\nu & -1+\nu^2 \\ 1-\nu & -1+\nu^2 \end{bmatrix}, \tag{6.4}$$

where ν is the CFL number, which is defined as

$$\nu \equiv a\frac{\Delta t}{\Delta x}. \tag{6.5}$$

Following the routine of von Neumann stability analysis, we assume the vector $\boldsymbol{q}$ to be a Fourier component with arbitrary wavenumber k such that

$$\boldsymbol{q}_j^n = \boldsymbol{A}_n(\theta)e^{ij\theta}, \quad (-\pi < \theta \leq \pi), \tag{6.6}$$

where $\theta = k\Delta x$ is the reduced wavenumber, and i is the imaginary unit. Substituting Eqs. (6.6) into (6.2) yields

$$\boldsymbol{A}_n(\theta) = \left[\boldsymbol{Q}_L e^{-i\theta/2} + \boldsymbol{Q}_R e^{i\theta/2} \right] \boldsymbol{A}_{n-1/2}(\theta). \tag{6.7}$$

Hence, the amplification matrix for each half step is

$$\begin{aligned}
\boldsymbol{M} &= \boldsymbol{Q}_L e^{-i\theta/2} + \boldsymbol{Q}_R e^{i\theta/2} \\
&= \begin{bmatrix} \cos\frac{\theta}{2} - i\nu\sin\frac{\theta}{2} & i(\nu^2-1)\sin\frac{\theta}{2} \\ i(1-\nu)\sin\frac{\theta}{2} & (2\nu-1)\cos\frac{\theta}{2} + i(\nu^2-2\nu)\sin\frac{\theta}{2} \end{bmatrix}.
\end{aligned} \tag{6.8}$$

The spectral radius of matrix $\boldsymbol{M}$, denoted as $\rho(\nu, \theta)$, is calculated numerically as a function of the CFL number ($0 \leq \nu \leq 1$) and reduced wavenumber ($-\pi \leq \theta \leq \pi$). In Fig. 6.1, it can be seen that $\rho(\nu, \theta) \leq 1$ holds for all wavenumbers as long as $0 \leq \nu \leq 1$. Thus, this second-order upwind CESE scheme is numerically stable when ν
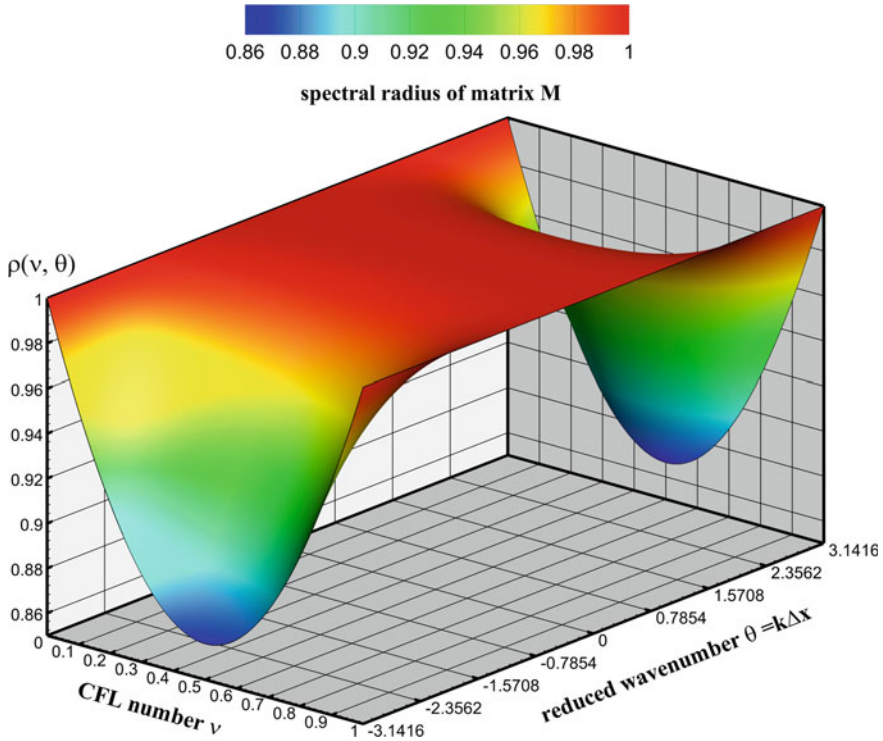
Fig. 6.1 Spectral radius $\rho(v, \theta)$ of matrix M in the von Neumann analysis of the upwind CESE scheme. Courtesy of Jiang [3]

is less than or equal to 1. Also shown in Fig. 6.1, as the CFL number approaches 0 or 1, the numerical dissipation in the second-order upwind CESE scheme vanishes.

For high-order CESE schemes, the stability analysis is more complicate and the stable upper bound of the CFL number depends on the scheme's construction. A novel approach to construct highly stable high-order CESE schemes was proposed by Chang [4], using the same space–time stencil as that in the second-order CESE scheme. The same CFL-number constraint for numerical stability (i.e., $v \leq 1$) is retained, which is favourable for explicit time-stepping computations.

6.2 Efficiency

The CESE method has a highly compact stencil in space and time, regardless of the order of the scheme. This compactness feature provides the CESE method a great advantage when parallel computation is performed.

For a CESE scheme, the nominal order of accuracy and spatial dimensionality will determine the number of independent variables (denoted by K) to be stored and updated at each solution point. For a M-th-order accurate ($M \geq 2$) and spatially D-dimensional ($D = 1, 2,$ or 3) CESE scheme, K can be determined by a precise counting and

$$K(D, M) = \begin{cases} M, & D = 1 \\ M(M + 1)/2, & D = 2 \\ M(M + 1)(M + 2)/6 & D = 3 \end{cases}. \tag{6.9}$$

Notably, the number K in the CESE scheme is the same as the number of degrees of freedom for each cell in the discontinuous Galerkin (DG) scheme. Therefore, if the CESE and DG schemes are of the same order and used to solve the same problem on the same mesh, the memory requirements for a CESE scheme and a DG scheme are comparable.

The CESE method requires neither a reconstruction procedure based on a wide stencil nor a multi-stage time-integration technique. Besides, no upwind operation related to flux calculation is included in central CESE schemes. Nevertheless, the CESE method adopts the Cauchy–Kowalewski procedure to express the temporal derivatives in terms of spatial derivatives. It also takes time to update the spatial derivatives using the algorithms described in Chaps. 3, 4 and 5. The combined effect of all these factors determines the computational speed of the CESE method.

6.3 Accuracy

Generally, the accuracy of a CESE scheme depends on the order of the Taylor expansions, i.e., the degree of the approximation polynomials that are used to approximate unknowns and fluxes within each solution element. To achieve the theoretical M-th order of accuracy of the scheme in space and time, the $(M-1)$-th-order Taylor expansions are needed. According to practical applications, the second-order CESE schemes, including the central and upwind schemes, can usually provide a satisfactory trade-off between accuracy and efficiency.

The ability to capture shock waves and contact discontinuities with high resolution is one of the excellent features of the CESE scheme. Here, two canonical flow problems are used to demonstrate the accuracy of CESE schemes by comparing the numerical simulation results by CESE and other schemes. As the first example, Sod's shock-tube problem [5] is solved by both the CESE method and the finite volume method (FVM). The CESE code is an implementation of the second order a–α scheme described in Chap. 3, while the FVM counterpart incorporates the second-order MUSCL (Monotonic Upstream-centered Scheme for Conservation Laws) reconstruction, the van Leer limiter, and the HLLC Riemann solver. A uniform mesh with 200 cells and a CFL number of 6.9 are employed in both the CESE and the FVM

computations. Density and internal energy distributions from $x = -1$ to $x = 1$ at time $t = 6.5$ computed by CESE and FVM schemes are plotted in Fig. 6.2a and b, respectively, in comparison with the exact solution. As shown in Fig. 6.2, a narrower contact discontinuity is resolved by the CESE scheme than the FVM counterpart. Meanwhile the CESE results match the exact solution better than the FVM results at the tail of the rarefaction wave.

Fig. 6.2 Comparison of numerical results of Sod's shock-tube problem by CESE and FVM schemes at dimensionless time $t = 0.5$. Courtesy of Jiang [3]
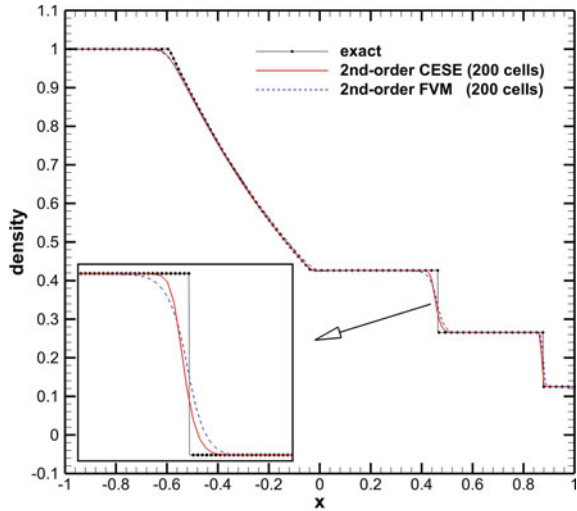
(a) CESE result vs. FVM result: density
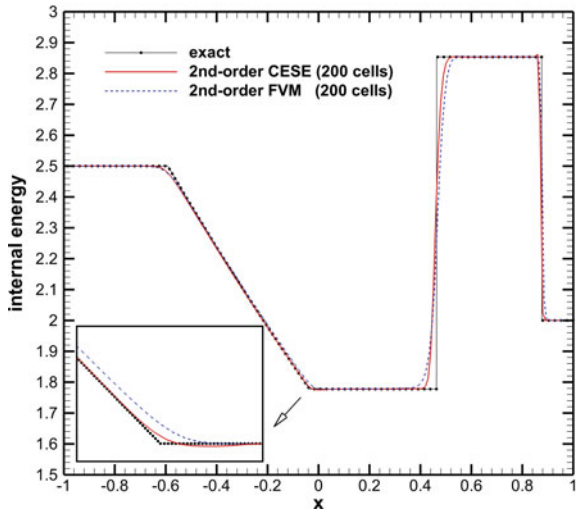
(b) CESE result vs. FVM result: internal energy
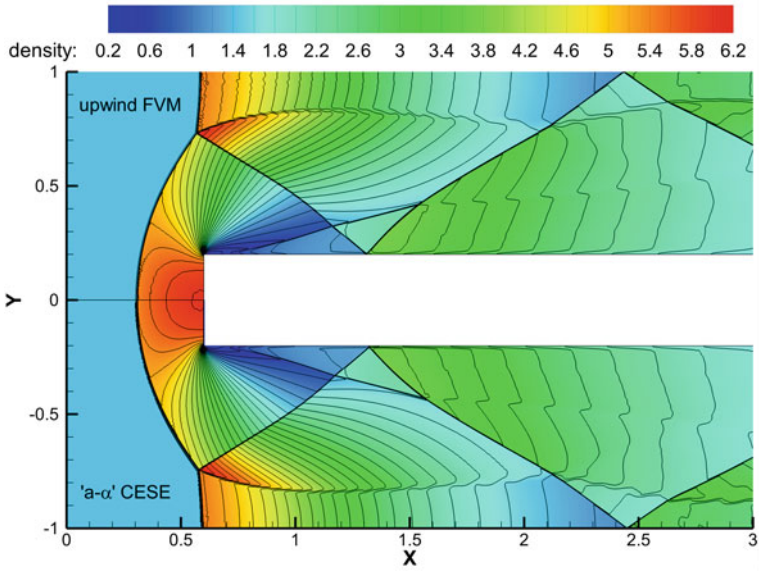
(a) *a–α* CESE result vs. FVM result
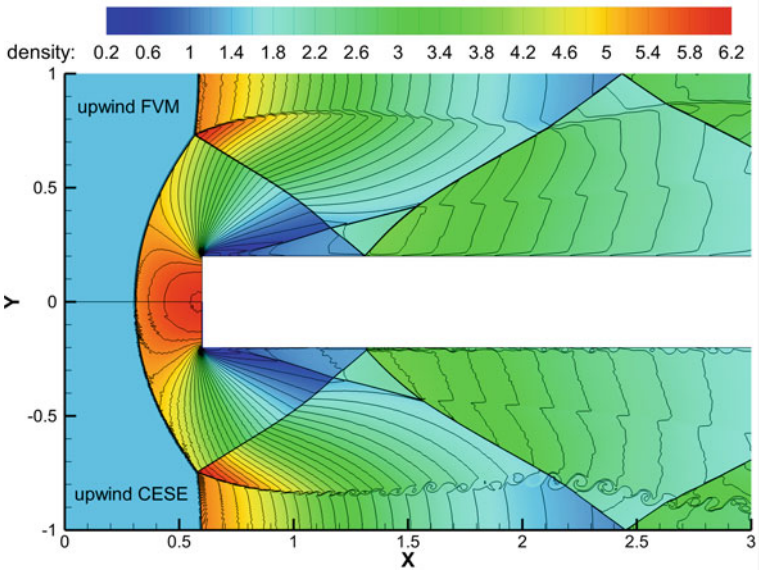
(b) Upwind CESE result vs. FVM result

Fig. 6.3 Comparison of FVM and CESE results for Mach 3 step problem: density contours at time $t = 4.6$. Courtesy of Jiang [3]

In the second example, the problem of a Mach-3 inviscid flow through a 2D tunnel with a forward-facing step inside is solved by three schemes (a–α CESE, upwind CESE, and upwind FVM) of second-order accuracy in space and time. The initial conditions, boundary conditions, and geometrical parameters are the same as that in [6]. The upwind FVM flow solver is provided by Hao et al. [7]. The same computational mesh and the same Courant–Friedrichs–Lewy (CFL) number of 6.95 are employed for all the three schemes. Density fields at time $t = 4.0$ are shown in Fig. 6.3a and b. Compared with the upwind FVM scheme, both the a–α and upwind CESE schemes can accurately capture the structure of shock waves, but only the upwind CESE scheme can resolve the phenomenon of shear-layer instability (see the lower part of Fig. 6.3b), which is not observed in the present a-α CESE and FVM results.

To assess the accuracy of the high-order CESE schemes described in Chap. 5, the Shu–Osher problem and the double Mach-reflection problem were studied intensively in Ref. [8], where the fourth-order CESE scheme provided a good resolution of fine structures in the flows.

References

1. Chang, S. C. (1995). The method of space-time conservation element and solution element— A new approach for solving the Navier-Stokes and Euler equations. *Journal of computational Physics, 119*(2), 295–324.
2. Yang, D., Yu, S., & Zhao, J. (2001). Convergence and error bound analysis for the space-time CESE method. *Numerical Methods for Partial Differential Equations: An International Journal, 17*(1), 64–78.
3. Jiang, Y. Z., Wen, C. Y., & Zhang, D. L. (2020). Space-time conservation element and solution element method and its applications. *AIAA Journal, 58*(12), 5408–5436.
4. Chang, S.C. (2010). A new approach for constructing highly stable high order CESE schemes. In *48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition.*
5. Sod, G. A. (1978). A survey of several finite difference methods for systems of nonlinear hyperbolic conservation laws. *Journal of computational physics, 27*(1), 1–31.
6. Woodward, P., & Colella, P. (1984). The numerical simulation of two-dimensional fluid flow with strong shocks. *Journal of computational physics, 54*(1), 115–173.
7. Hao, J., Wang, J., & Lee, C. (2017). Numerical simulation of high-enthalpy double-cone flows. *AIAA Journal, 55*(7), 2471–2475.
8. Shen, H., Wen, C. Y., Liu, K. X., & Zhang, D. L. (2015). Robust high-order space–time conservative schemes for solving conservation laws on hybrid meshes. *Journal of Computational Physics, 281*, 375–402.

Chapter 7
Application: Compressible Multi-fluid Flows

Multi-fluid flows involving shock-accelerated inhomogeneities and shock-induced instability play essential roles in a wide variety of problems including, but not limited to, supersonic combustion [1], inertial confinement fusion [2], and supernova explosion [3]. Numerical simulations of these complex flows prove to be challenging in the presence of moving and deformable material interfaces, especially for fluids with large differences in their densities or thermodynamic properties. Therefore, a discontinuity-capturing, mass-conserving, and positivity-preserving scheme is desirable for compressible multi-fluid simulations.

Qamar et al. [4] implemented the CESE method of Chang [5] and Zhang et al. [6] for solving the one- and two-dimensional compressible two-fluid models of Kreeft and Koren [7]. Numerical simulations were performed for gas–liquid Riemann problems and interactions of air shock waves with inhomogeneities containing lighter and heavier gases. In comparison with the non-oscillatory central scheme [5] and the kinetic flux-vector splitting scheme [8], the CESE scheme gives better resolution of discontinuities. Because of its outstanding ability for discontinuity capturing, the studies on Richtmyer–Meshkov instability (RMI) [9, 10] and Rayleigh–Taylor instability (RTI) [11, 12] have greatly benefited from the development of this method. Those numerical results not only eliminated disturbing factors in experiments (e.g., dimensional effect, boundary effect, liquid mist scattering), but also provided more ideal working conditions (that cannot be easily realized in experiment) and more flow details (that cannot be timely recorded by camera).

Recently, the upwind time–space CESE method was used extensively in studies of shock-accelerated inhomogeneous flows [13–15] and RMI [16–18]. The compressible two-fluid flows are described by a volume-fraction-based five-equation model [19] coupled with the stiffened gas equation of state [20]. Extensive numerical simulations were carried out using the maximum-principle-satisfying upwind CESE scheme [13], which is an improved version of the upwind CESE scheme presented in Chap. 4. The maximum-principle satisfying property is achieved by adopting a very simple limiter proposed by [21]. Furthermore, the ability to capture contact discontinuities

(material interfaces) can be enhanced by employing the HLLC Riemann solver in the upwind procedure. As a result, challenging numerical simulations of the gas–gas and gas–liquid Riemann problems were successfully performed, in which the mass of each fluid component is conserved and the positivity of volume fractions is preserved [13].

In this chapter, we use extensive numerical examples to indicate that the CESE method captures shocks and contact discontinuities sharply without spurious oscillations and proves to be a robust and accurate numerical tool for studies of compressible multi-fluid flows.

7.1   Richtmyer–Meshkov Instability

The Richtmyer–Meshkov instability occurs when an initially perturbed interface separating two different fluids is impulsively accelerated. This impulsive acceleration, in RMI studies especially, is provided by shock waves mostly. Two basic mechanisms dominate the RMI development, are the baroclinic vorticity and the pressure disturbance. For the baroclinic vorticity, the extent of the misalignment of the pressure gradient across the shock with the density gradient across the material interface makes contribution to the perturbation growth. While for the pressure disturbance, the wave system plays an important role.

As the inertial confinement fusion (ICF) cares more about the interaction of a converging shock with a disturbed interface, the converging RMI has become an imperative [22, 23]. The nature of geometrical convergence in converging RMI, however, makes the perturbation development more complicated because of the coupling of the Bell–Plesset (BP) effect [24, 25], Rayleigh–Taylor (RT) effect [26, 27], and the multiple shock impacts therein.

In perfect agreement with their experimental images [28], Zhai et al. [17] used the upwind CESE method to make further discussion about the converging RMI issues. Figure 7.1 shows the shock behaviors and interface morphologies after a converging shock interacting with perturbed interface. The inner test gas is SF6 and the interface initial amplitude is 1 mm. The comparison clearly approved the applicability of the method used in simulating converging shock waves and converging RMI issues.

The Rayleigh–Taylor effect on the phase inversion (RTPI) was the major concern in Zhai's work [17], which is supposed to be helpful in finding a freezing state interface in ICF physics. In this work, the influence of the initial amplitude on the RTPI was firstly investigated. Figure 7.2 depicts three converging RMI cases with different initial amplitudes (Cases I1, I2 and I3 with $a_0 = 1.0$, 1.65, and 2.0 mm, respectively), according to which, Fig. 7.3 records the time-variation of the interface amplitudes. It was found that the amplitude histories can be roughly segmented into three stages after the initial shock compression. At the first stage, the amplitude of the perturbed interface increases because of the RMI, while during the second stage before re-shock, the amplitude reduces owing to the RT stabilization effect which is caused by the stronger adverse pressure gradient near the geometry origin. Based
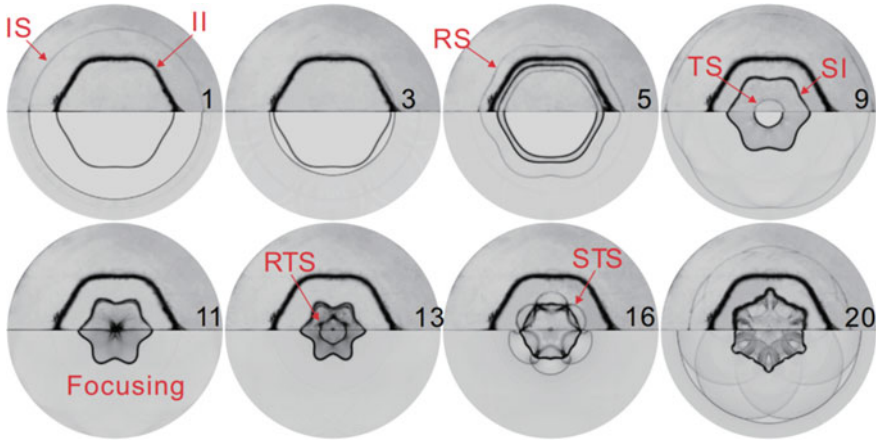
Fig. 7.1 The evolution of a single-mode interface with initial amplitude 1 mm. upper: experimental schlieren image; lower: numerical schlieren image at the same instants. (IS = incident shock, II = initial interface, RS = reflected shock, TS = transmitted shock, SI = shocked interface, RTS = reflected transmitted shock, STS = secondary transmitted shock). Courtesy of Z. G. Zhai [17]

on the amplitude histories, they concluded that whether a normal phase inversion occurs or not depends on the competition between the RT stabilization effect and RM instability caused by the re-shock. Especially, for a special initial amplitude, there is a critical state with a zero-amplitude of the interface at the re-shock.

Furthermore, a parametric study was performed to evaluate the influences of Atwood number, shock Mach number, and initial radius of the interface on the variation of critical amplitude with the azimuthal mode number. Their thorough investigation well explained the reason why this sensitive RTPI phenomenon was not observed in previous converging RMI studies.

The even more complicated converging RMI at a dual-mode interface was numerically studied by Zhou et al. [18] using the same upwind CESE methodology. It is not surprising that the dual-mode or multi-mode interface is far more complicated than the single-mode converging counterpart due to the existence of mode interferences such as harmonic generation and bubble merger. By comparing the detailed processes of the interface deformation and wave propagation for different dual-mode cases together with the development of a referencing single-mode interface, significant influence of the phase difference between two basic waves on the instability development can be clearly distinguished. While after the re-shock, the discrepancy becomes much smaller due to the weak dependence of the re-shocked RMI on the pre-re-shock state.

Fig. 7.2 Schlieren images showing the evolution of the single-mode interface. (PI = phase inversion, RTPI = Rayleigh–Taylor phase inversion). Courtesy of Z. G. Zhai [17]
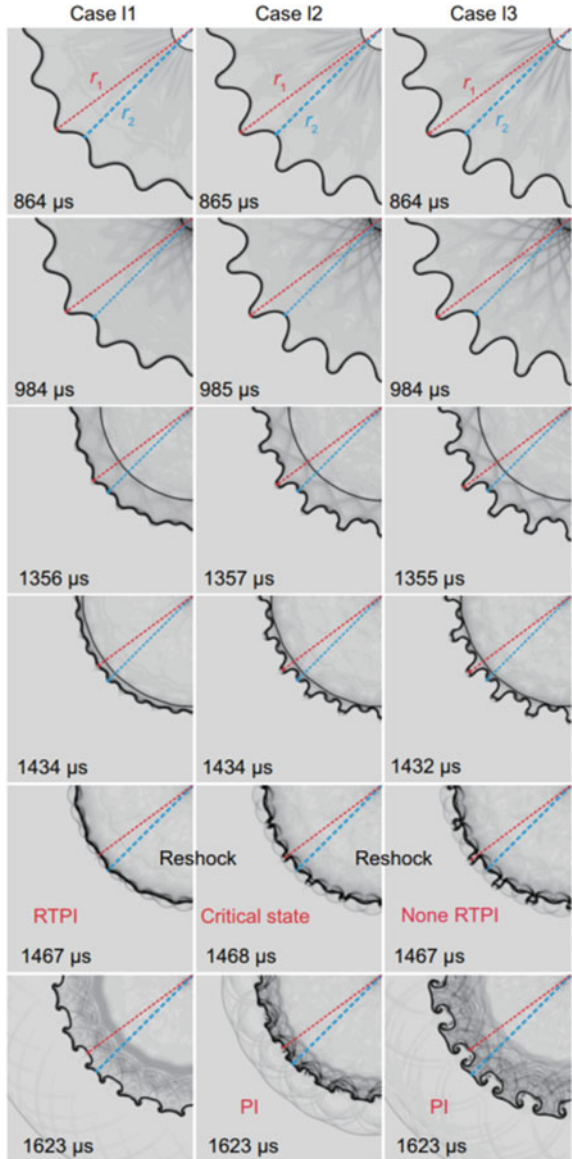
The study on dynamics of the dual-mode converging RMI clearly reveals the mode coupling effect, in which the growth of the first mode was found to be inhibited, promoted, or not influenced, depending upon the first mode amplitude as well as the phase difference between the two basic waves. These findings, including all the other parametric studies in their work [18], were considered to be of great help for designing an optimal structure of the ICF capsule.
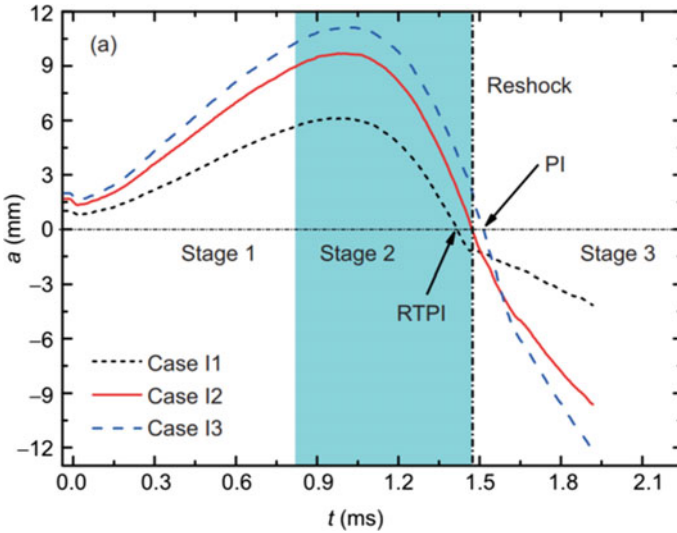
Fig. 7.3 Time-variation of interface amplitude in three perturbed cases. (PI = phase inversion, RTPI = Rayleigh–Taylor phase inversion). Courtesy of Z. G. Zhai [17]

7.2 Rayleigh–Taylor Instability

Other than the shock-induced interface instability, the rarefaction-induced instability was performed by [29] using the upwind space–time CESE method. Some of their numerical schlieren images of the SF_6/air interface instabilities induced by the rarefaction waves are presented in Fig. 7.4, in which the rarefaction waves were generated by simulating a diaphragm burst in a SF_6 shock tube. The influence of rarefaction wave acting time was demonstrated by varying the distance between the initial diaphragm and the gas interface (shown in cases SA1, SA3, and SA5); while the influence of the rarefaction wave strength was demonstrated by setting different pressure ratios between the two sides of the initial diaphragm (shown in SA3, SA7, and SA9).

Generally, after the rarefaction wave sweeps the SF6/air interface from the downside, RTI is triggered and the perturbation on the interface gradually grows. Different from the classical RMI where an impulsive action is exerted, and also different from the classical RTI where a continuous body force exists, the rarefaction wave imposes acceleration to the interface within a limited time. When the rarefaction wave leaves the interface, the baroclinic vorticity dominates the later interface evolution and KHI occurs on the interface as the further stretching of spikes and bubbles.

Based on the numerical results provided by the upwind CESE method, theoretical models were modified in Liang's work [29] considering the time-varying acceleration and the growth rate transition from RTI to RMI. It was also found that the interface perturbation can be more unstable under the rarefaction wave condition than under
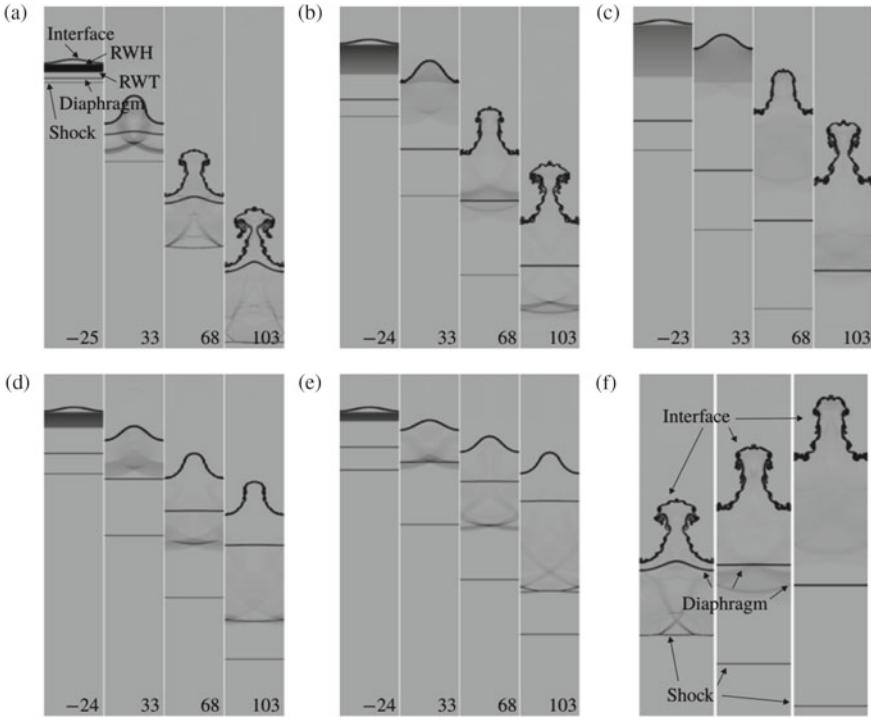
Fig. 7.4 Schlieren images of the SF_6/air interface evolution induced by rarefaction waves in cases **a** SA1, **b** SA3, **c** SA5, **d** SA7, and **e** SA9. Numbers represent the time in μs. **f** schlieren images of the interface evolution in cases SA1 (left), SA3 (middle) and SA5 (right) at dimensionless time 10.3. (RWH = rarefaction wave head, RWT = rarefaction wave tail). Courtesy of Y. Liang [29]

the shock wave condition due to the larger amount of vorticity deposited by the continuous pressure gradient.

7.3   Shock Refraction

The upwind space–time CESE method has also been used in simulating the shock refraction phenomenon at an inclined air/helium interface in the cylindrical converging shock scenario [16]. Figure 7.5 presents the experimental and numerical schlieren images of the shock refraction, in which good agreement between the two was achieved. Moreover, the numerical results apparently provide much cleaner images than their experimental counterparts (especially the region behind the deformed gas interface).

It is observed that, during the incident shock wave converging, when the shock velocity increases and the incident angle (with respect to the gas interface) decreases,
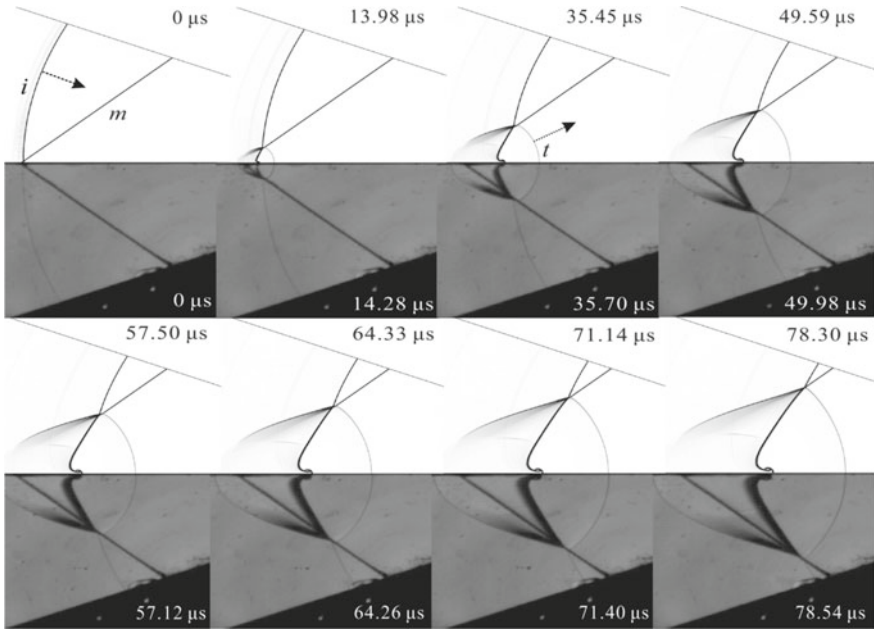
Fig. 7.5 Sequences of experimental (lower) and numerical (upper) schlieren frames showing the evolution of a converging shock wave refracting at a 45° tilted interface. (i = incident shock, m = material interface, t = transmitted shock). Courtesy of Z. G. Zhai [16]

the wave pattern is found to transit from one to another. In the work of Zhai et al. [16], two interfaces with 45° and 60° initial incident angles were examined. For the 45° case, the shock pattern was found to transit from free precursor refraction (FPR) to bound precursor refraction (BPR), and then to regular refraction with reflected shock (RRR); while for the 60° case, it was found to be from twin von Neumann refraction (TNR), to twin regular refraction (TRR), to free precursor von Neumann refraction (FNR), and finally to FPR. These clearly depicted transition sequences that never occur in the planar shock scenarios, greatly enriched the shock refraction classification.

7.4 Shock–Gas-Bubble Interaction

Here, we review the numerical studies of shock-accelerated inhomogeneous flows conducted by Shen et al. [13], Fan et al. [15], and Guan et al. [14] first. Figure 7.6 shows a general schematic of the computational setting of a shock-accelerated inhomogeneity. The homogeneity can be gas bubble [13], water column [13] or droplet [14], and the shape of the initial homogeneity can be circular, square, rectangular, and even triangular [15].
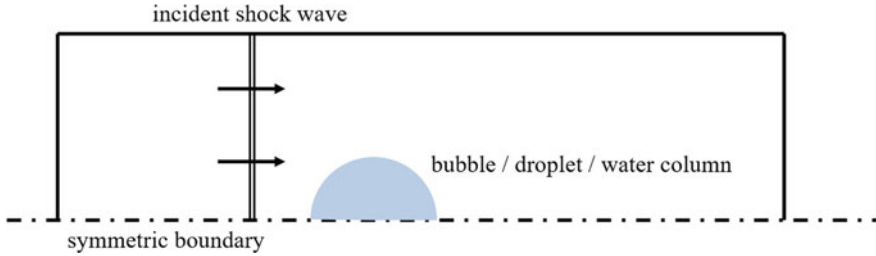
Fig. 7.6 Schematic of the initial setup for the interaction of an incident shock wave and a gas/water inhomogeneity

In the shock–helium-cylinder case [13], a planar shock propagates from left to right at Mach 1.22 and impacts the helium cylinder with a radius of 25 mm. The cylinder morphology after shock impact is shown in Fig. 7.7, in which the gas interface is sharply captured. The vortices, deposited due to baroclinicity, are clearly seen developing on the interface. As is shown, a salient feature of the helium cylinder morphology is the piercing of a middle air jet that finally separates the whole cylinder into two symmetric lobes.

A quantitative validation of the CESE method in simulating the shock–helium-cylinder interaction was performed by comparing the trajectories of the upstream point, the jet point, and the downstream point of the helium cylinder with the ones obtained by the level set method [30]. Good agreement of the results was achieved between the two methods, as depicted in Fig. 7.8.

The maximum-principle-satisfying upwind CESE scheme was applied by Fan et al. [15] to present a comprehensive study of jet-formation phenomenon in the interaction of a planar shock with a variety of heavy gas inhomogeneities. Comparing with the above light helium cylinder scenario, the heavy R22 cylinder experiences an obvious different deformation pattern. Figure 7.9 shows the cylinder morphology of the R22 cylinder with a radius of 25 mm impacted by a planar incident shock with Mach number 1.22 (to numerically reproduce the experiment conducted by Haas and Sturtevant [31]). The shocks that transmitted into this heavy cylinder were found converging at the downstream pole. The ensuing shock pattern and pressure gradient leads to a jet at the downstream cylinder surface.

Fan et al. [15] summarized that the heavy gas jet forms not only in the scenario of circular cylinder, but also in a series of differently shaped heavy gas cylinders, including square, rectangle, and triangle. Fan's square case reproduced the experiment performed by Luo et al. [32] where the shock Mach was 1.17 in air and the side length of the SF_6 square was 56.6 mm. To reveal more details of the shock converging patterns in a small region, a slightly over-fined mesh was used in their simulation, which greatly reduced the numerical viscosity, making the secondary vortices at the late stage more pronounced than its corresponding experimental images. Other than that, numerical results shown in the lower halves of Fig. 7.10 agree quite well with the upper halves' experimental images, including the gas interface morphology, the shock patterns, and the consequent jet formation at the middle leeward surface of

(a) time = 199.5 μs

(b) time = 399.7 μs

(c) time = 600.6 μs

(d) time = 800.2 μs

(e) time = 999.5 μs

(f) time = 1200.0 μs
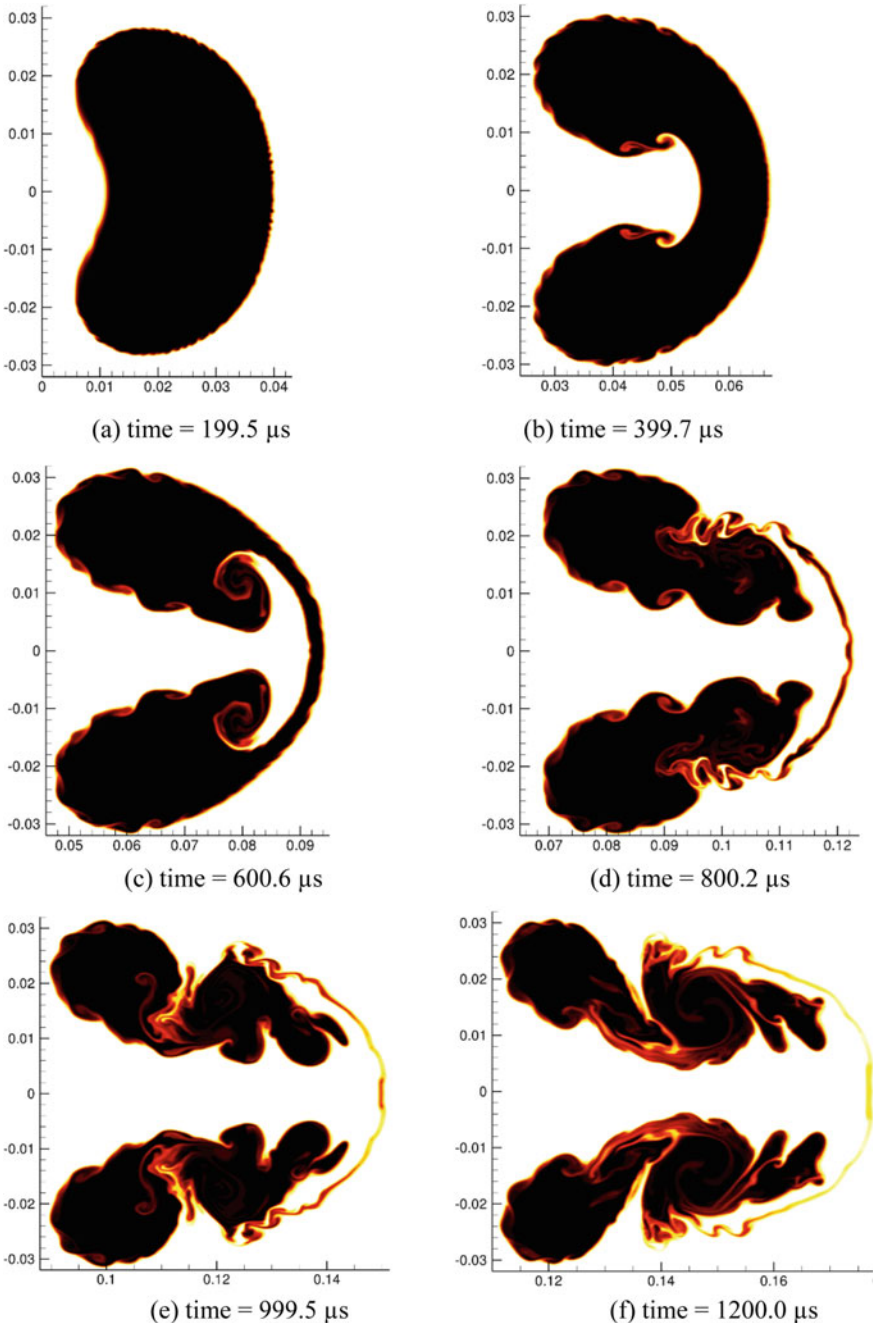
Fig. 7.7 Deformation history of a helium bubble impacted by a Mach 1.22 shock. Courtesy of H. Shen [13]
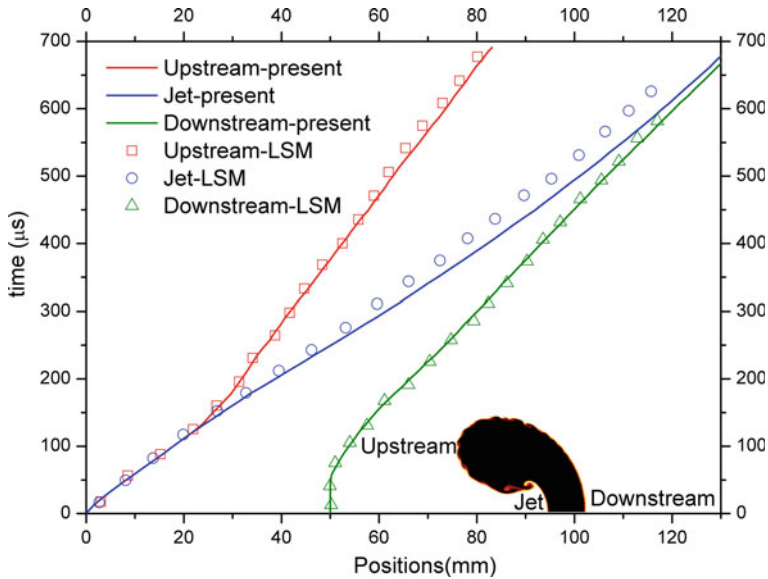
Fig. 7.8 Position history of the upstream point, the jet point, and the downstream point of the helium bubble. Courtesy of H. Shen [13]
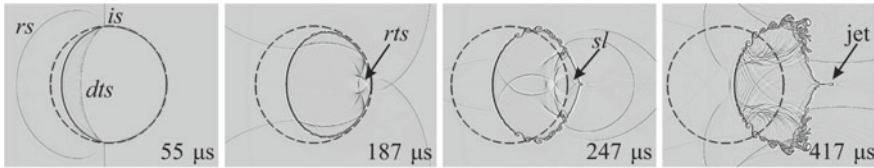
Fig. 7.9 Shadowgraphs of the evolution of an R22 circular cylinder under the impact of an incident shock with Mach number 1.22. (*is* = incident shock; *dts* = direct transmitted shock; *rs* = reflected shock; *rts* = reversed transmitted shock; *sl* = slip line.) Courtesy of E. Fan [15]

the square. Based on the detailed information provided by the CESE simulation, the mechanism of jet formation was revealed. It was found that for all the cases they simulated, the key factor in forming a downstream jet is the formation of type II shock-shock interaction [33]. According to this, a geometrical criterion was proposed to determine whether a jet will be formed [15].

7.5　Shock–Water-Droplet Interaction

The ability of the upwind CESE method to handle gas–liquid interfaces was initially demonstrated in Shen's work [13]. The setup of this problem resembles that of the shock–bubble interaction (see Fig. 7.6). Due to the high-density ratio and large difference between the thermodynamic properties of air and water, more challenges
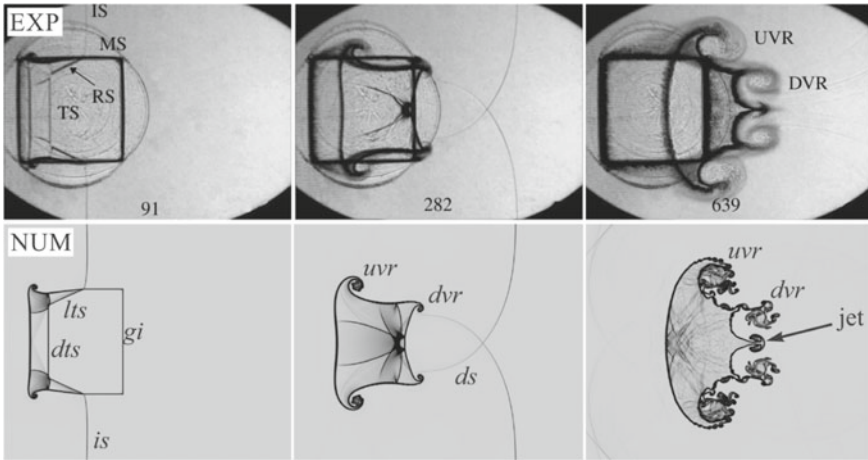
Fig. 7.10 Comparison of experimental images (upper) by Luo et al. [32] and the numerical results (lower) of the planar incident shock impact on a square cylinder using CESE method. (*is* = incident shock, *rs* = refracted shock, *ts* = transmitted shock, *ms* = Mach stem, *gi* = gas interface, *dts* = direct transmitted shock, *lts* = lateral transmitted shock, *ds* = diffracted shock, *uvr* = upstream vortex pair, *dvr* = downstream vortex pair). Courtesy of E. Fan [15]

are there in simulating the shock–water-column interaction. In this case, the radius of the water column was 2.4 mm and a shock wave with Mach number 1.47 was launched to sweep over the column from left to right. Upon the interaction of a shock with water column (or water droplet), two essential issues are extensively discussed. One issue concerns the immediate interaction of the shock wave (or rarefaction wave) with the water column (or droplet). It covers an extremely short period of time compared to the whole water column breakup, in spite of its vital role in the following interface deformation. The other issue concerns the long-period water column deformation, which covers from the initial protrusion growth on the water column surface to the later large interface distortion and mass loss of the bulk of the column. At the early stage, as shown in Fig. 7.11, the shock propagates through the water column as though passing over a rigid cylinder, nearly no interface deformation is detected. However, the shock waves as well as rarefaction waves bounce back and forth inside the water column, making the internal pressure change dramatically and becomes highly heterogeneous. After the incident shock has passed, as shown in Fig. 7.12, atomized water is gradually stripped away by the high-speed post-shocked flow. The water column develops into a crescent shape and the downstream was covered up by the transversely spreading atomized water. Note that the instabilities were captured in detail in this CESE simulation. The history of drag coefficient of the water column was also derived from the numerical results, as shown in Fig. 7.13. Again, good agreement was obtained in comparison with numerical simulations of Chen [34] and experimental data of Igra and Takayama [35].

(a) time = 1.85 µs

(b) time = 3.69 µs

(c) time = 6.15 µs

(d) time = 8.61 µs

(e) time = 11.1 µs

(f) time = 16.0 µs
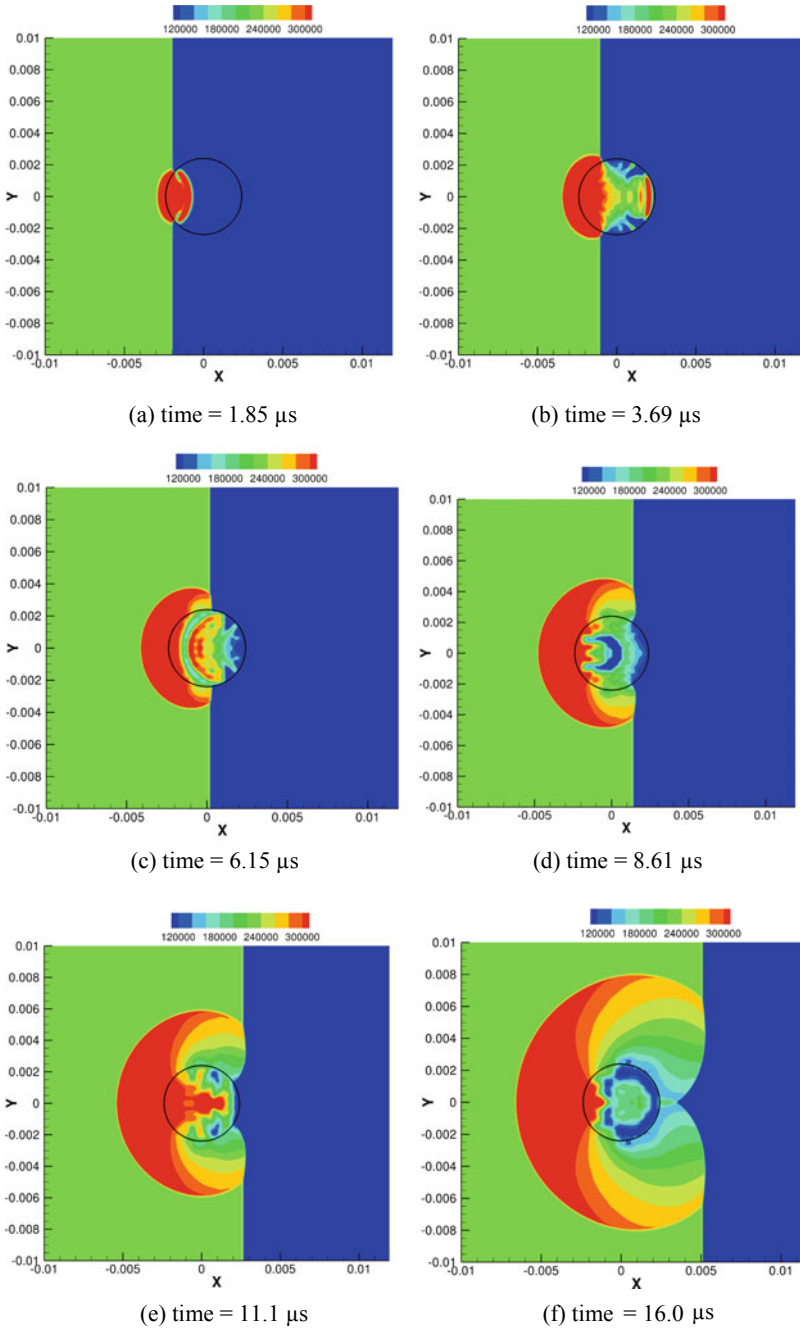
Fig. 7.11 Sequences of pressure contours (unit: Pa) in the early stage of a shock–water-column interaction. Courtesy of H. Shen [13]
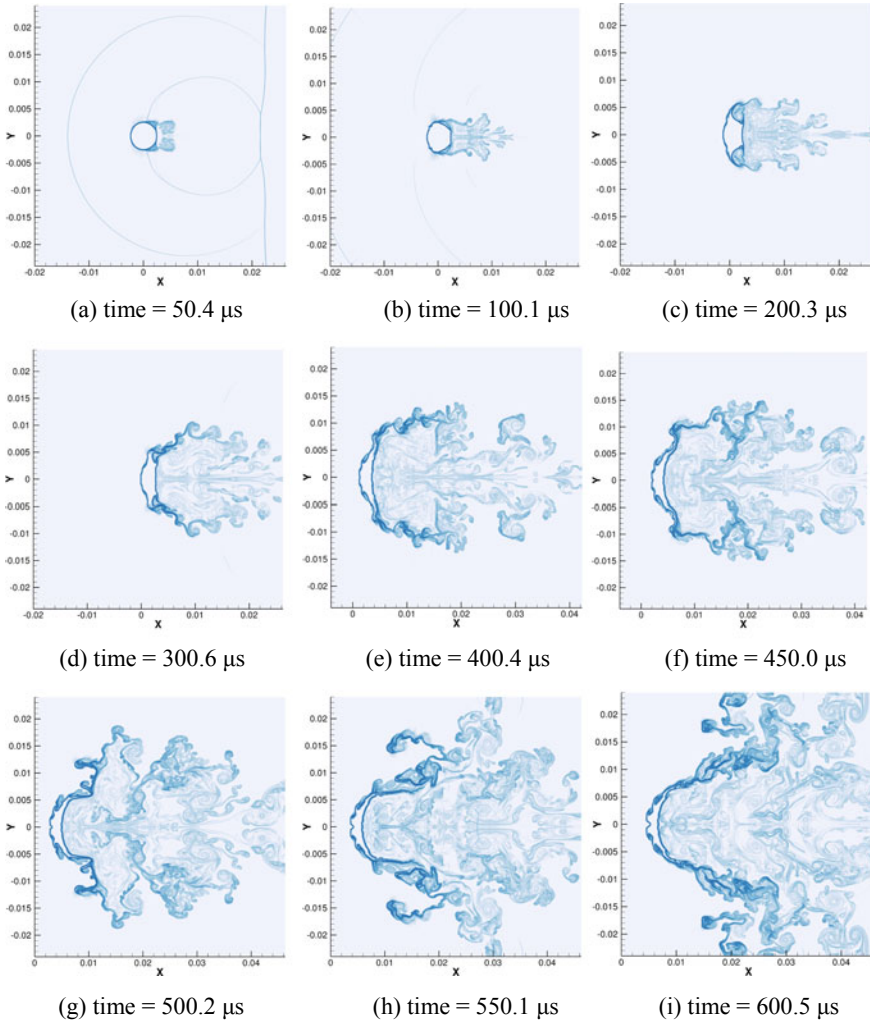
Fig. 7.12 Evolution of the numerical schlieren plot during the stripping breakup of the water column. Courtesy of H. Shen [13]

The case of a spherical water droplet deformation under planar shock impact was studied by Guan et al. [14] using axisymmetric simulation. As shown in Fig. 7.14, the initial protrusions emerging on the droplet surface were clearly captured by the CESE methodology. The internal flow pattern of the droplet was visualized numerically in this study. It is seen in Fig. 7.15 that the internal flow pattern of the droplet is established soon after the impact by the incident shock and is held steady for a long
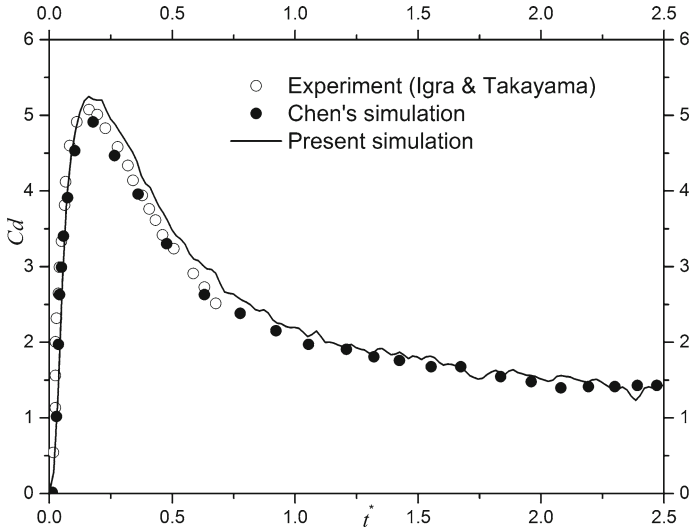
Fig. 7.13 Drag coefficient C_d history of a water column impacted by a shock versus dimensionless time t^*. Courtesy of H. Shen [13]
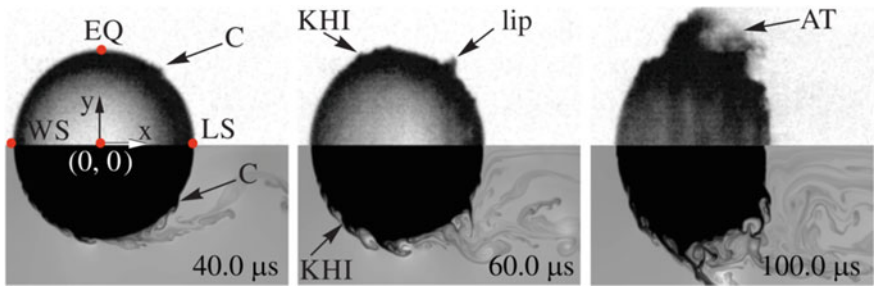
Fig. 7.14 Comparison of the numerical (lower part) and experimental (upper part) results at different instants. (EQ = equator; WS = windward stagnation; LS = leeward stagnation; C = corrugation; KHI = Kelvin–Helmholtz instability, AT = atomization). Courtesy of B. Guan [14]

time. For the first time, a saddle point was observed in this internal flow pattern. Further, a simple theory was proposed to correlate the stationary position of the saddle point with the Mach number of the incident shock.
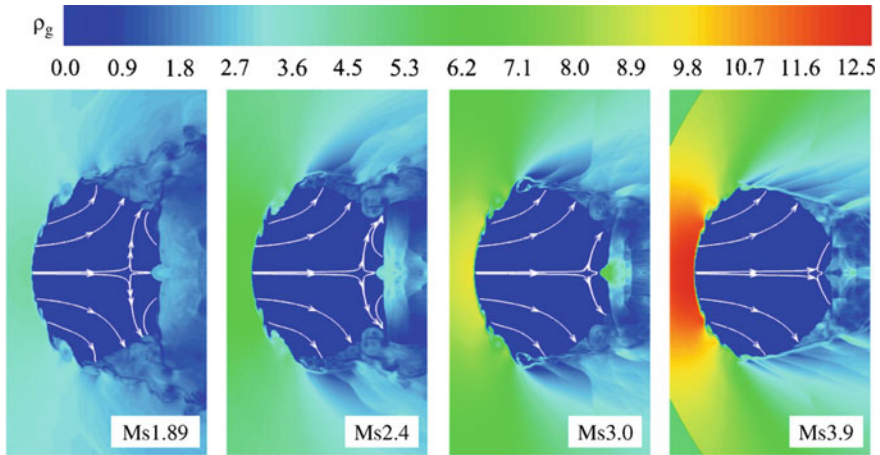
Fig. 7.15 Morphologies and internal flow patterns of a water droplet under shock impact with different shock Mach numbers *Ms*. Contours show the density of air with units kg/m^3. Courtesy of B. Guan [14]

References

1. Park, S. J., Yu, S. T., Lai, M. C., Chang, S. C., & Jorgenson, C. (1999). Numerical calculation of unstable detonations by the method of space-time conservation element and solution element. In *37th Aerospace Sciences Meeting and Exhibit*.
2. Lindl, J., Landen, O., Edwards, J., Moses, E., & Team, N. (2014). Review of the national ignition campaign 2009–2012. *Physics of Plasmas, 21*(2), 020501.
3. Shimoda, J., Inoue, T., Ohira, Y., Yamazaki, R., Bamba, A., & Vink, J. (2015). On cosmic-ray production efficiency at supernova remnant shocks propagating into realistic diffuse interstellar medium. *The Astrophysical Journal, 803*(2), 98.
4. Qamar, S., Ahmed, M., & Ali, I. (2012). The space-time CE/SE method for solving reduced two-fluid flow model. *Communications in Computational Physics, 12*(4), 1070–1095.
5. Chang, S. C. (1995). The method of space-time conservation element and solution element—A new approach for solving the Navier-Stokes and Euler equations. *Journal of computational Physics, 119*(2), 295–324.
6. Zhang, Z. C., Yu, S. J., & Chang, S. C. (2002). A space-time conservation element and solution element method for solving the two-and three-dimensional unsteady Euler equations using quadrilateral and hexahedral meshes. *Journal of Computational Physics, 175*(1), 168–199.
7. Kreeft, J. J., & Koren, B. (2010). A new formulation of Kapila's five-equation model for compressible two-fluid flow, and its numerical treatment. *Journal of Computational Physics, 229*(18), 6220–6242.
8. Qamar, S., & Ahmed, M. (2009). A high order kinetic flux-vector splitting method for the reduced five-equation model of compressible two-fluid flows. *Journal of Computational Physics, 228*(24), 9059–9078.
9. Richtmyer, R. D. (1954). *Taylor instability in shock acceleration of compressible fluids.* Los Alamos Scientific Lab., N. Mex.
10. Meshkov, E. (1969). Instability of the interface of two gases accelerated by a shock wave. *Fluid Dynamics, 4*(5), 101–104.
11. Rayleigh. (1900). Investigation of the character of the equilibrium of an incompressible heavy fluid of variable density. *Scientific Papers*, 200–207.

12. Taylor, G. I. (1950). The instability of liquid surfaces when accelerated in a direction perpendicular to their planes. I. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences, 201*(1065), 192–196.
13. Shen, H., Wen, C. Y., Parsani, M., & Shu, C. W. (2017). Maximum-principle-satisfying spacetime conservation element and solution element scheme applied to compressible multifluids. *Journal of Computational Physics, 330*, 668–692.
14. Guan, B., Liu, Y., Wen, C. Y., & Shen, H. (2018). Numerical study on liquid droplet internal flow under shock impact. *AIAA Journal, 56*(9), 3382–3387.
15. Fan, E., Guan, B., Wen, C. Y., & Shen, H. (2019). Numerical study on the jet formation of simple-geometry heavy gas inhomogeneities. *Physics of Fluids, 31*(2), 026103.
16. Zhai, Z., Li, W., Si, T., Luo, X., Yang, J., & Lu, X. (2017). Refraction of cylindrical converging shock wave at an air/helium gaseous interface. *Physics of Fluids, 29*(1), 016102.
17. Zhai, Z., Zhang, F., Zhou, Z., Ding, J., & Wen, C. Y. (2019). Numerical study on Rayleigh–Taylor effect on cylindrically converging Richtmyer–Meshkov instability. *Science China Physics, Mechanics & Astronomy, 62*(12), 1–10.
18. Zhou, Z., Ding, J., Zhai, Z., Cheng, W., & Luo, X. (2020). Mode coupling in converging Richtmyer–Meshkov instability of dual-mode interface. *Acta Mechanica Sinica, 36*(2), 356–366.
19. Shyue, K. M. (1998). An efficient shock-capturing algorithm for compressible multicomponent problems. *Journal of Computational Physics, 142*(1), 208–242.
20. Saurel, R., & Abgrall, R. (1999). A simple method for compressible multifluid flows. *SIAM Journal on Scientific Computing, 21*(3), 1115–1145.
21. Zhang, X., & Shu, C. W. (2010). On maximum-principle-satisfying high order schemes for scalar conservation laws. *Journal of Computational Physics, 229*(9), 3091–3120.
22. Liang, Y., Zhai, Z., & Luo, X. (2018). Interaction of strong converging shock wave with SF6 gas bubble. *Science China Physics, Mechanics & Astronomy, 61*(6), 1–9.
23. Wu, Z., Huang, S., Ding, J., Wang, W., & Luo, X. (2018). Molecular dynamics simulation of cylindrical Richtmyer–Meshkov instability. *Science China Physics, Mechanics & Astronomy, 61*(11), 1–11.
24. Bell, G. I. (1951). Taylor instability on cylinders and spheres in the small amplitude approximation. Report No. LA-1321, LANL 1321.
25. Plesset, M. (1954). On the stability of fluid flows with spherical symmetry. *Journal of Applied Physics, 25*(1), 96–98.
26. Fincke, J., Lanier, N., Batha, S., Hueckstaedt, R., Magelssen, G., Rothman, S., Parker, K., & Horsfield, C. (2004). Postponement of saturation of the Richtmyer–Meshkov instability in a convergent geometry. *Physical Review Letters, 93*(11), 115003.
27. Lombardini, M., Pullin, D., & Meiron, D. (2014). Turbulent mixing driven by spherical implosions. Part 1. Flow description and mixing-layer growth. *Journal of Fluid Mechanics, 748*, 85–112.
28. Luo, X., Zhang, F., Ding, J., Si, T., Yang, J., Zhai, Z., & Wen, C. Y. (2018). Long-term effect of Rayleigh–Taylor stabilization on converging Richtmyer–Meshkov instability. *Journal of Fluid Mechanics, 849*, 231–244.
29. Liang, Y., Zhai, Z., Luo, X., & Wen, C. Y. (2020). Interfacial instability at a heavy/light interface induced by rarefaction waves. *Journal of Fluid Mechanics, 885*, A42.
30. Terashima, H., & Tryggvason, G. (2009). A front-tracking/ghost-fluid method for fluid interfaces in compressible flows. *Journal of Computational Physics, 228*(11), 4012–4037.
31. Haas, J. F., & Sturtevant, B. (1987). Interaction of weak shock waves with cylindrical and spherical gas inhomogeneities. *Journal of Fluid Mechanics, 181*, 41–76.
32. Luo, X., Wang, M., Si, T., & Zhai, Z. (2015). On the interaction of a planar shock with an polygon. *Journal of Fluid Mechanics, 773*, 366–394.
33. Edney, B. (1968). Anomalous heat transfer and pressure distributions on blunt bodies at hypersonic speeds in the presence of an impinging shock, Flygtekniska Forsoksanstalten, Stockholm (Sweden).

34. Chen, H. (2008). Two-dimensional simulation of stripping breakup of a water droplet. *AIAA Journal, 46*(5), 1135–1143.
35. Igra, D., & Takayama, K. (2001). A study of shock wave loading on a cylindrical water column. *Report of the Institute of Fluid Science, Tohoku University, 13*, 19–36.

Chapter 8
Application: Detonations

Detonation is a shock-induced combustion in which chemical reactions are closely coupled with shock waves. The shock wave compresses the reactant with an abrupt increase in temperature and pressure, initiating the reactants to be burnt into products. The intense heat release permits the high propagating speed of the shock wave to be sustained. It is fundamental research related to both the safety industry and propulsion systems. For most explosive mixtures, detonation wave speeds are formulated by Chapman–Jouguet (CJ) theory. Typical detonation velocities for gaseous mixtures generally range from 1400 to 3000 m/s. Behind the shock, the time scale for reactions is commonly on the order of microseconds or even less. Furthermore, the detonation front is intrinsically unstable, forming transient multi-dimensional structures. Many studies revealed that high resolution is necessary to resolve the essential detonation structures. Due to its complex nature and multiple time scales, detonation is thus a challenging problem for solvers on shock-capturing capability, robustness, and computational efficiency. This chapter will present several essential aspects of detonation research by applying the CESE schemes.

8.1 Gaseous Detonations

In detonation simulations, the requirement for the computational resources is usually very high because the satisfactory resolution is necessary to compute the coupling between flow and chemical reactions and to resolve the detonation structures. As a result, applying detailed chemistry is usually limited to problems with simple and small domains or particular situations. Otherwise, simulations with simplified models are preferred to gain insight into the physics without loss of credibility. For illustrative purposes, we consider a system for ideal gas and a simplified chemical model. The 2D inviscid reactive compressible flow equations can be expressed as

$$\frac{\partial U}{\partial t} + \frac{\partial F}{\partial x} + \frac{\partial G}{\partial y} = S \tag{8.1}$$

where the conserved variable vector U, the flux vectors F, G and source term vector S are defined as $U = [\rho, \rho u, \rho v, E, \rho \lambda]^T$, $F = [\rho u, \rho u^2 + p, \rho u v, (E + p)u, \rho \lambda u]^T$, $G = [\rho v, \rho u v, \rho v^2 + p, (E + p)v, \rho \lambda v]^T$, $S = [0, 0, 0, 0, \dot{\omega}]^T$. The symbols ρ, u, v, p, E, and λ denote density, velocities in x and y directions, pressure, the total energy per unit volume, and mass fraction of the reactant, respectively. The perfect gas law was adopted here as $p = \rho R T$, where R is the gas constant and T is the gas temperature. The total energy is

$$E = \frac{p}{(\gamma - 1)} + \frac{1}{2}\rho(u^2 + v^2) + \rho \lambda Q, \tag{8.2}$$

where γ and Q are the specific heat ratio and the specific heat release. The chemical reaction rate is formulated by the one-step Arrhenius model as

$$\dot{\omega} = -k\rho \lambda e^{-E_a/RT}, \tag{8.3}$$

where k is the pre-exponential factor, and E_a denotes the activation energy.

8.1.1   Ignition of Detonations

Ignition of detonation refers to the formation of a detonation wave through either an instantaneous onset or the transition from deflagration. Shen and Parsani [1] studied the direct ignition of spherical detonation by using the upwind CESE scheme to solve Eq. (8.1). The distance between the shock wave and the position where half of the reactant is consumed can be defined as the half-reaction length. A uniform resolution with 20 meshes per half-reaction length was adopted, and the number of mesh points is 1.6 billion. It is well-known that the dynamics of detonation are sensitive to the activation energy. Thus, three activation energies representing stable ($E_a = 15$), mildly unstable ($E_a = 27$), and highly unstable ($E_a = 50$) detonations were studied. For different E_a, k was adjusted to fix the half-reaction length in unit length. A hot spot with $p_s = (\gamma - 1)E_s$, $T_s = 20.0$, and a radius of 1 was used to form a blast wave to ignite the mixture. All variables above are nondimensionalized with respect to the state of the unburnt reactant.

The peak pressure history behind the shock for 1D detonations are depicted in Fig. 8.1, and typical 2D contours are plotted in Fig. 8.2. For stable case (Fig. 8.1a), the subcritical, critical, and supercritical regimes are observed successively when E_s increases from 1.8×10^4 to 1.0×10^5. In the critical regime, $E_s = 2.0 \times 10^4$ (Fig. 8.1b), P_{sh} first decays and then rapidly increases due to the formation and amplification of the pressure pulse behind the leading shock. The 2D result shows a
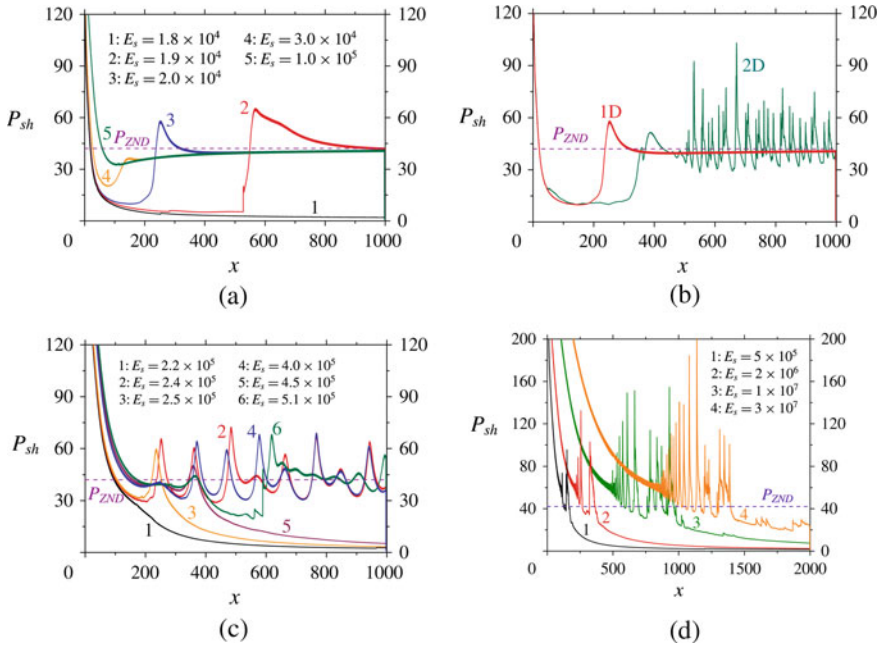
Fig. 8.1 Peak pressure history at the shock front (P_{sh}) as a function of position and source energy (E_s), for **a** $E_a = 15$, **b** $E_a = 15$, $E_s = 2 \times 10^4$, **c** $E_a = 27$, **d** $E_a = 50$. Courtesy of H. Shen [1]

highly oscillatory pattern for P_{sh} due to the development of multidimensional instabilities, and the run-up distance deviates from 1D solution. Three critical energies were detected for the mildly unstable case (Fig. 8.1c), The authors further observed the fourth critical energy when refining the search using a minor incremental E_s. However, there might be a unique critical energy in the 2D case. The inconsistency between 1 and 2D becomes more apparent for the highly unstable cases. For all the E_s tested, 1D detonation eventually failed (Fig. 8.1d). The results suggest that, without large overdriven, the 1D detonation propagation through auto-ignition is impossible for the highly unstable cases. Contrarily, the critical energy is approximately 1.5×10^5 for 2D cases (Fig. 8.2d). The key factors dominating the direct ignition of 2D detonation are the unsteadiness arising from the decay of the leading shock, heat release from the chemical reaction, and the inherent multidimensional instabilities. In 1D detonations, only the first two aforementioned factors exist. Detonation fails when the excessive unsteadiness overtakes the heat release. The transverse waves induced by the multidimensional instabilities create local over-driven detonations. On the one hand, these local over-driven detonations facilitate the propagation of the global detonation; on the other hand, induce stronger expansion waves to increase the risk of failure. The competition among these three factors should be considered to provide a comprehensive model of direct initiations.
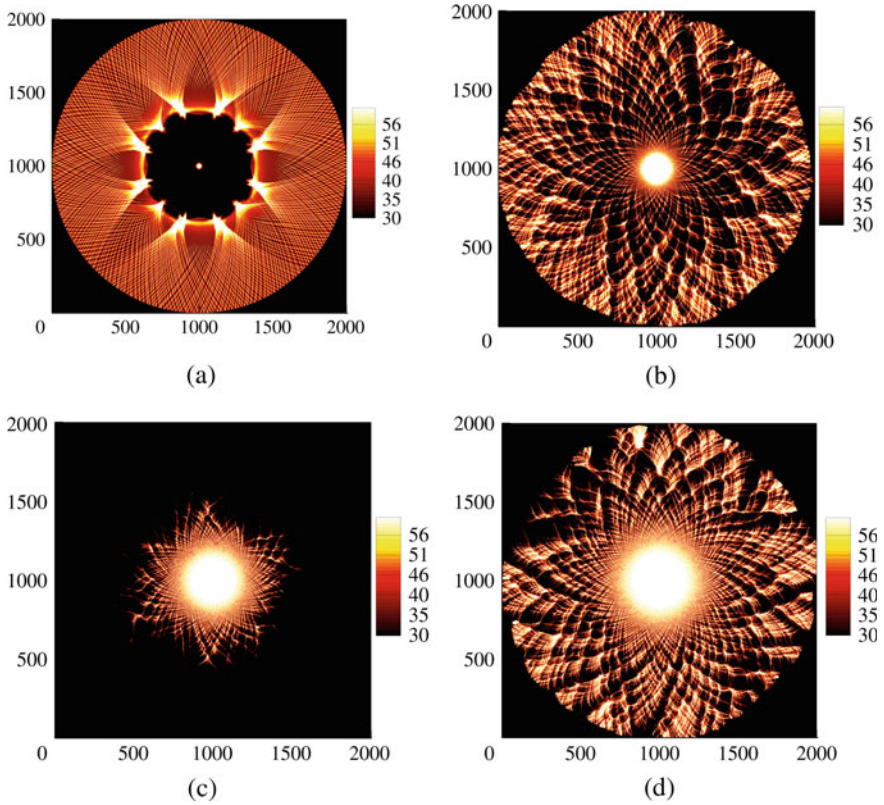
Fig. 8.2 Maximum pressure history in two dimensions. **a** $E_a = 15$, $E_s = 2.5 \times 10^4$, **b** $E_a = 27$, $E_s = 4.5 \times 10^4$, **c** $E_a = 50$, $E_s = 1.0 \times 10^4$, **d** $E_a = 50$, $E_s = 1.5 \times 10^4$. Courtesy of H. Shen [1]

Another situation of detonation ignition occurs when the gaseous reactant in the unconfined space is ignited through the detonation wave emerging from a small tube. When the detonation wave diffracts from a tube into the unconfined space, the substantial expansion will perturb the detonation front. If tube size is small enough, the detonation wave will quench (subcritical), otherwise, the detonation will continue to propagate (supercritical). Shi et al. [2] investigated the 2D detonation diffraction using the a–α CESE scheme. Two scenarios were considered. If the detonation wave inside the tube does not contain transverse waves, it is classified as the planar case. Otherwise, it is classified as the cellular case. In these simulations, Eqs. (8.1)–(8.3) were employed with $E_a = 24$ to assure that 1D detonation wave is stable (For E_a higher than 25.27, 1D detonation is intrinsically unstable). A resolution with 24 meshes per half-reaction length was tested converged for current settings, and the total mesh points are around 410 million. A first glance at the results revealed that the critical channel width for cellular scenarios is smaller than that for the planar case (Table 8.1). Four cases, including supercritical and subcritical for planar and cellular

Table 8.1 Critical channel
width for $E_a = 24$

	Lower limit	Upper limit
Planar	100	110
Cellular	75	85

scenarios, were carefully discussed by examining wave structures and Lagrangian particles that are initially scattered in the flow.

For planar scenarios, the disturbance originates from the corner propagates towards the symmetric line. The diffracted shock wave decelerates and decouples with the flame front. In the shocked reactant, a compression wave is formed and amplified through the temperature gradient towards the leading shock. The compression wave is indicated by the successive abrupt change in temperature for particles E7–E9 (Fig. 8.3). When this compression wave merges with the leading shock, detonation may be facilitated if the strengthened shock wave exceeds a critical value. On the other hand, the aforementioned disturbance reflects from the symmetric line, and the reflected rarefaction (Fig. 8.4) wave may simultaneously affect the shocked reactant and hinder the formation and amplification of the compression wave. For the sub-critical case, i.e., $w = 100$, re-initiation is reproduced if the boundary condition is modified such that the rarefaction wave does not reflect towards the leading shock. Therefore, for the planar cases, re-initiation is attributed to the competition between the coalescence of the amplified compression wave with the leading shock and the strength of the reflected rarefaction wave.

For cellular scenarios, the re-initiation patterns are different from the planar cases. Despite the number of transverse waves decreases when the detonation wave expands into the unconfined space. It is interesting to note that from Fig. 8.5, a complete decoupling is not observed during the propagation of the supercritical case ($w = 85$) (a complete decouple is observed before re-initiation in the planar $w = 110$
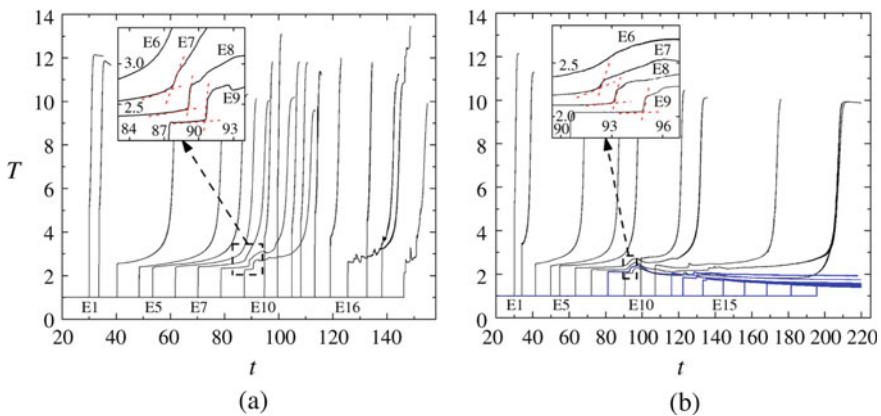
Fig. 8.3 Temperature traces of particles along the re-initiation direction for planar detonation diffraction: **a** $w = 110$ and **b** $w = 100$. Courtesy of L. S. Shi [2]
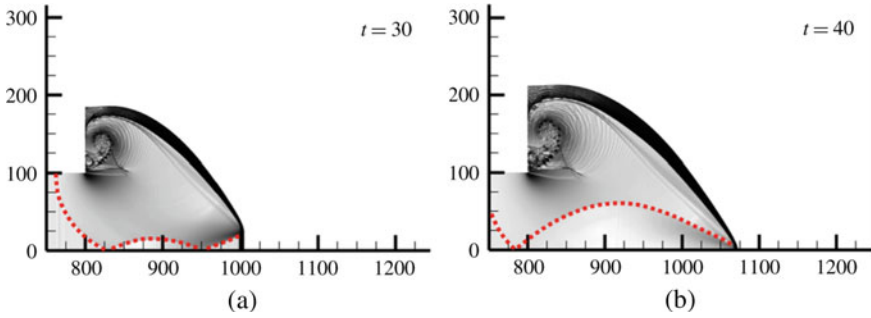
Fig. 8.4 Numerical schlieren images for planar case with $w = 100$. Red dash lines indicated the head of rarefaction. Courtesy of L. S. Shi [2]
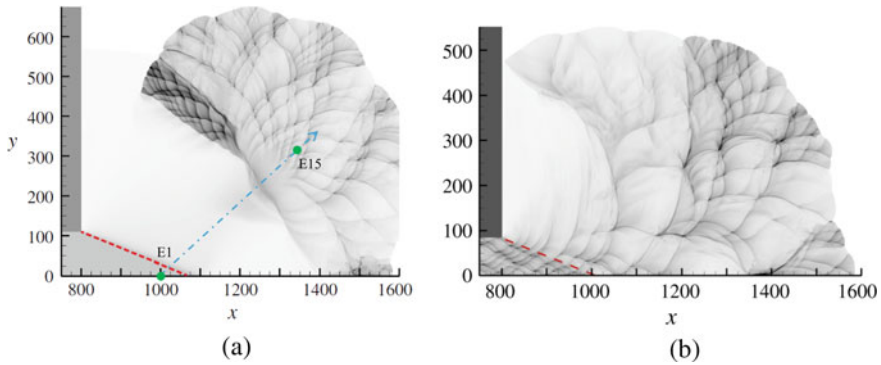
Fig. 8.5 Numerical soot foils for **a** planar detonation diffractions, $w = 110$, **b** cellular detonation diffractions, $w = 85$. The red dashed lines indicate the disturbance line predicted by Skews' model. The blue dash-dot line is the re-initiation path along which particles E_i are initially located. Courtesy of L. S. Shi [2]

case). Detonation wave sustains through the continuous formation of local overdriven detonation when transverse waves collide. Previous experimental research revealed that the critical tube diameter for unstable mixtures is $d_c \approx 13\lambda$ (λ: detonation cell size), but $d_c \approx 30\lambda$ or even more were measured for stable mixtures highly diluted with argon. The distinct difference between the re-initiation mechanisms planar and cellular detonation diffractions elaborated in current simulation provides a plausible explanation of the discrepancy of the correlations on critical tube sizes. In unstable detonations, where the formation of local hot spots is essential to sustain the detonation, the correlation between the d_c and λ is established. It is well-known that the cellular patterns in stable detonation are regular, with transverse wave velocity approximately being the sound velocity in the burned products. This corresponds to the planar scenarios. Detonation re-initiates if the influence of the reflected rarefaction wave is negligible. Thus the d_c / λ correlation may not be necessary.

8.1.2 Dynamics of Detonations

As discussed in Sect. 8.1, multi-dimensional instability is a fundamental phenomenon in detonation. In this section, we will elaborate more on the studies on detonation propagation in one, two, and three dimensions using CESE schemes.

Wang et al. [3] have implemented 2D CESE on detonation reflection on a wedge using a detailed reaction model. Later, in the study on the propagation modes of stoichiometric H_2/O_2 3D cellular detonation in a square duct [4], two types of propagation modes were verified, which depended on the configuration of the initial conditions. Unreacted pockets with complex structures were also captured by using the improved CESE scheme.

Numerous numerical studies based on detailed chemical mechanisms have quantitatively reproduced the key structures of multi-dimensional detonations. They experienced barriers to predict the cell size correctly. For example, for H_2/O_2 mixture highly diluted by argon, the regular cell sizes predicted from simulation using detailed chemical mechanisms were approximately half of those measured in experiments. One of the important physical phenomena in high-speed flow, the vibrational non-equilibrium of molecules, was usually ignored in early numerical studies. Behind the detonation wave, the time scales for vibrational relaxation and chemical reaction are comparable. Shi et al. [5] re-examined this problem with the consideration of vibrational relaxation and its coupling with reactions. The convection of vibrational energy was integrated into governing equations, and the translational-rotational energy and the vibrational energy were separated in the formulation of the total energy, which is composed of enthalpy of formation, translational-rotational energy, kinetic energy, and vibrational energy. The energy exchange rate was calculated by the Landau-Teller model.

Four scenarios were compared in 1D and 2D simulations. (1) The mixture is assumed to be thermodynamically equilibrium. (2) Vibrational relaxation is incorporated, and the translational temperature is kept as the dominant temperature of the chemical reactions. (3) Vibrational relaxation is considered, and the geometric average temperature is used to compute the reaction rates. (4) The physically consistent vibration-chemistry-vibration coupling model is adopted to account for the effect of vibrational non-equilibrium on chemical reaction rates. Examples of 1D detonation structures are depicted in Fig. 8.6. Model (2) exhibits a temperature overshot behind the shock compared to the equilibrium case because the mixture remains vibrational cold. The vibrational temperature approaches the equilibrium state gradually, and a noticeable disparity still exists when exothermic reactions are triggered. Since translational-rotational temperature controls reactions in model (2), the overshot in temperature results in a marginal decrease in the half-reaction length (δ). In model (3), the geometric averaged temperature is used to calculate the reaction rate. The onset of severe chemical reactions occurs only when the vibrational relaxation approaches equilibrium. The half-reaction length increases by 1.64 times that in the model (1). A similar trend is observed in the case using model (4). The detonation
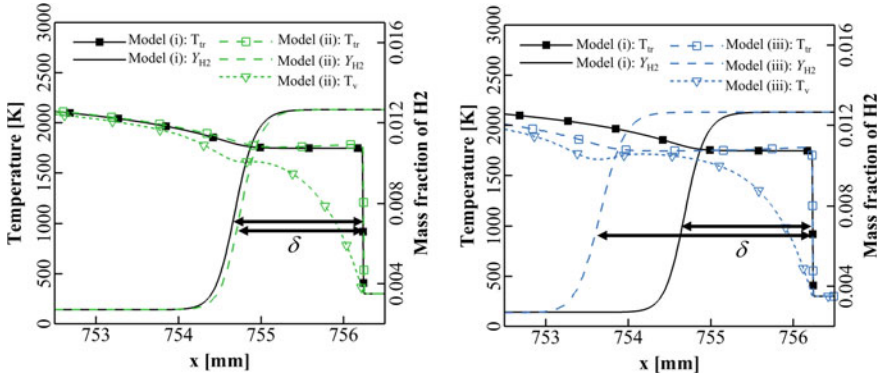
Fig. 8.6 Temperature and H$_2$ distribution with initial condition of 0.1 atm and 300 K. Left: scenarios 1 and 2. Right: scenarios 1 and 3. Courtesy of L. S. Shi [5]

cell widths are predicted from 2D simulation using models (1) and (2) are approximately the same. The cell widths using models (3) and (4) are approximately 1.32 and 1.34 times the value of that in the model (1). It reveals that the vibrational-chemical coupling effect may be one of the factors responsible for the failure to correctly predict the cell size.

Later, Uy et al. [6] further examine the effect of vibrational non-equilibrium on the one-dimensional instabilities using the CESE scheme. In this study, 1D piston-supported detonation case with fixed non-dimensional specific heat release $Q = 50$, the ratio of specific heats $\gamma = 1.2$, and characteristic vibrational temperature $\theta = 20$. The ratio between the chemical time scale τ_c and the vibrational time scale τ_V was selected as a control parameter. Park's two-temperature model was used to evaluate the effect of vibrational non-equilibrium on the chemical reaction rate. The neutral stability limit under the vibrational equilibrium assumption is approximately $E_a = 26.47$, above which the detonation is unstable with amplified oscillation. Numerical tests reveal $E_a = 27$ with the thermal equilibrium assumption being longitudinally unstable. Nevertheless, for finite vibrational relaxation rate (i.e., the $\tau^\alpha \equiv \tau_c/\tau_V \neq \infty$), the amplitudes of pressure oscillations decay for smaller $\tau^\alpha = 3, 5, 7$, and the oscillation decays much faster for as τ^α decreases. The results imply that the propagation of detonation is stabilized because of the vibrational non-equilibrium. Theoretical analysis using linear stability analysis (LSA) under different overdriven factors $f = (D/D_{CJ})^2$ was also provided in Uy et al. [7]. The neutral stability limit predicted using the CESE scheme agrees very well with LSA (Table 8.2). The accuracy of CESE in solving detonation problems is further confirmed.

Table 8.2 Comparison of the neutral stability limit (NSL) and the period of oscillation (PO) computed by LSA and numerical simulation. Case I: $f = 1$, thermal equilibrium (eq). Case II: $f = 1$, thermal nonequilibrium (neq), $\tau^\alpha = 5$. Case III: $f = 1$, neq, $\tau^\alpha = 7$. Case IV: $f = 1$, neq, $\tau^\alpha = 9$. Case V: $E_a = 50$, eq. Case VI: $E_a = 50$, neq, $\tau^\alpha = 5$. Case VII: $E_a = 50$, neq, $\tau^\alpha = 10$. Case VIII: Ea = 50, neq, $\tau^\alpha = 20$. Courtesy of C. K. Uy [7]

Case	Linear stability analysis		Numerical simulation	
	NSL	PO	NSL	PO
I	$E_a = 26.46$	10.63	$E_a = 26.47$	10.64
II	$E_a = 27.13$	12.14	$E_a = 27.14$	12.15
III	$E_a = 26.99$	11.77	$E_a = 27.02$	11.74
IV	$E_a = 26.90$	11.54	$E_a = 26.92$	11.51
V	$f = 1.62$	8.02	$f = 1.62$	8.03
VI	$f = 1.555$	9.51	$f = 1.554$	9.52
VII	$f = 1.582$	8.80	$f = 1.581$	8.83
VIII	$f = 1.60$	8.46	$f = 1.598$	8.43

8.1.3 Rotating Detonation Waves

Because of the rapid compression of the mixture and the quasi-constant volume combustion, the high thermodynamic efficiency of detonation makes it favourable to be employed in developing potential detonation engines. The rotating detonation engine (RDE) is one of the promising candidates in which the detonation wave propagates circumferentially, and the reactant is continuously injected into the combustion chamber.

Figure 8.7 depicts the flow features of RDE simulated with the upwind CESE schemes. The 2D simulation is essentially the unwrapped RDE channel from a 3D geometry. The top is the RDE outlet described by a non-reflective boundary. The reactant is injected into the combustor from the bottom through micro-Laval nozzles. The ideal injection was assumed that all the bottom areas could inject fresh reactant. Typical flow features of RDE can be observed, characterized by a rotating detonation wave, triangular reactant layer, oblique shock, and slip line. Furthermore, Wang et al. [8] used an improved CNI 2D CESE scheme to study the kerosene/air RDE. The results suggested that decreasing the inlet/wall ratio would reduce the strength of the detonation wave, and the reaction zone would be elongated. With further decreasing the air ratio, the burned gas emerges in the triangular zone, and the detonation wave fails to be self-sustained.

(a) 2D simulation   (b) 3D simulation
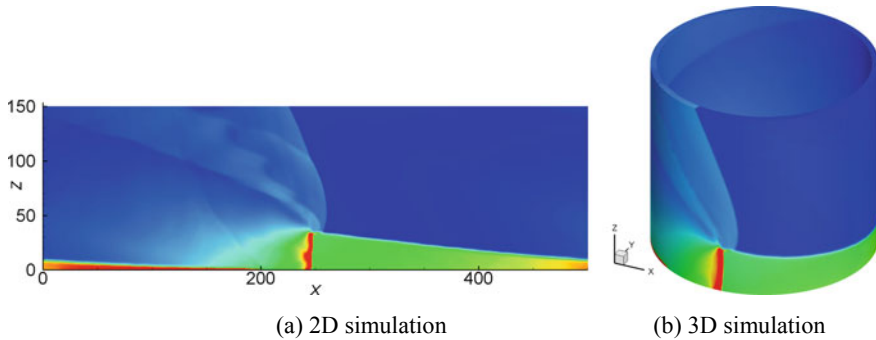
Fig. 8.7 Density contours of 2D and 3D RDE simulations using the upwind CESE scheme

8.2   Two-Phase Detonations

Liquid fuels and energetic powders are of advantages of high energy density and easy storage, which makes them common in practical propulsion systems. Detonation-based propulsion systems using these non-gaseous fuels involve reactive multi-phase high-speed flows. Although gaseous, liquid and/or solid phases are involved in these detonation phenomena, it is always referred to as two-phase detonation from the viewpoint of reactants, namely gas–liquid two-phase detonation and gas–solid two-phase detonation. Compared to gaseous detonations, two-phase detonations are characterized not only by shock waves and fast combustion, but also by multi-phase interaction and multi-scales, leading to great difficulties in numerical simulations. With the successful applications of the CESE method in gaseous detonation simulations, it is imperative to extend the CESE method to two-phase detonation problems and test its capabilities of accuracy and robustness in two-phase detonation simulations.

There are mainly two kinds of frameworks that are used to address the discrete phase in two-phase detonation simulations, namely the Eulerian–Eulerian framework and the Eulerian–Lagrangian framework. Between these two frameworks, the Eulerian–Eulerian framework, which is also referred to as the two-fluid model/method, is more common and easier in implementation and extension in two-phase detonation simulations. The discrete phase (the liquid phase in gas–liquid two-phase detonations or the solid phase in gas–solid two-phase detonations) is considered as a special continuum, such that continuum mechanics can be employed to describe the bulk motion of the discrete phase. Then, the discrete phase becomes another "fluid", and the gaseous-phase flow and the discrete-phase flow can be solved by similar approaches. As for the Eulerian–Lagrangian framework, the gaseous phase is solved as in the Eulerian–Eulerian framework, but every discrete liquid particle (droplet) or solid particle is tracked individually by Newton's laws of motion.

In two-phase detonation modelling, it is a general way to assume that the particle (liquid or solid) of the discrete phase is small, spherical in shape and

uniformly suspended in the gas atmosphere. Additionally, the two-phase suspension is assumed to be diluted enough to neglect the volume fraction of discrete particles and particle–particle collisions. Two-phase detonation is a highly transient two-phase flow problem involving strong shock waves; hence, thermal and mechanical non-equilibrium between the gas and particles should be considered. Further, uniform distribution of temperature is considered within particles due to their small particle sizes, and ideally, all reaction heat is absorbed by gas only. Based on the above assumptions, the two-dimensional governing equation of the gaseous phase in an Eulerian–Eulerian framework can be expressed as follows [9]:

$$\frac{\partial U}{\partial t} + \frac{\partial F}{\partial x} + \frac{\partial G}{\partial y} = S + W, \tag{8.4}$$

where U is the vector of conserved variables, F and G the conservation flux vectors in the x- and y-directions, S the vector of two-phase interaction source terms, and W the vector of chemical reaction source terms, respectively. They can be expressed as

$$U = \begin{bmatrix} \rho_1 \\ \cdots \\ \rho_{ns} \\ \rho u \\ \rho v \\ E \end{bmatrix}, \quad F = \begin{bmatrix} \rho_1 u \\ \cdots \\ \rho_{ns} u \\ \rho u^2 + p \\ \rho uv \\ (E+p)u \end{bmatrix}, \quad G = \begin{bmatrix} \rho_1 v \\ \cdots \\ \rho_{ns} v \\ \rho uv \\ \rho v^2 + p \\ (E+p)v \end{bmatrix}, \quad W = \begin{bmatrix} \dot{\omega}_1 \\ \cdots \\ \dot{\omega}_{ns} \\ 0 \\ 0 \\ 0 \end{bmatrix}, \tag{8.5}$$

and

$$S = \begin{bmatrix} 0 \\ \vdots \\ J_p \\ \vdots \\ 0 \\ -f_x + u_p J_p \\ -f_y + v_p J_p \\ -q_p - (u_p f_x + v_p f_y) + (E_p/\rho_p) J_p \end{bmatrix}. \tag{8.6}$$

In the above equations, ρ_i is the mass density of species i and $i = 1, \ldots, ns$; ns is the number of species contained in the gas mixture; p, u, and v are the gas pressure and x- and y-components of gas velocity, respectively; $\dot{\omega}_i$ is the mass production rate of gaseous species i by chemical reactions; and ρ and E are the mass density and total energy per unit volume of the gas mixture, expressed as

$$\rho = \sum_{i=1}^{ns} \rho_i, \quad E = \rho h - p + \frac{1}{2}\rho(u^2 + v^2), \tag{8.7}$$

where h is the specific enthalpy of the gas mixture, calculated by

$$h = \sum_{i=1}^{ns} \frac{\rho_i}{\rho} h_i, \tag{8.8}$$

with the specific enthalpy of each individual species h_i as a function of gas temperature T, obtained from a species thermodynamic data base, such as the NASA Glenn data base [10]. By further assuming each species performs as a perfect gas, the equation of state of the gas mixture is then given by

$$p = \sum_{i=1}^{ns} \rho_i R_i T, \quad R_i = \frac{R_0}{W_i} \tag{8.9}$$

where $R_0 = 8.314$ J/(mol K) is the universal gas constant, and W_i is the molar mass of species i. Variables in Eq. (8.6) are related to the properties of the discrete phase and the interaction between two phases. Their definitions will be given in the following paragraphs.

As for the discrete phase, the governing equation has a similar form as Eq. (8.4):

$$\frac{\partial U_p}{\partial t} + \frac{\partial F_p}{\partial x} + \frac{\partial G_p}{\partial y} = S_p, \tag{8.10}$$

where U_p, F_p and G_p are the vectors of conserved variables, conservation fluxes in the x- and y-directions of the discrete phase, which can be given by

$$U_p = \begin{bmatrix} \rho_p \\ \rho_p u_p \\ \rho_p v_p \\ E_p \\ N_p \end{bmatrix}, \quad F_p = \begin{bmatrix} \rho_p u_p \\ \rho_p u_p^2 \\ \rho_p u_p v_p \\ E_p u_p \\ N_p u_p \end{bmatrix}, \quad G_p = \begin{bmatrix} \rho_p v_p \\ \rho_p u_p v_p \\ \rho_p v_p^2 \\ E_p v_p \\ N_p v_p \end{bmatrix},$$

$$S_p = \begin{bmatrix} -J_p \\ f_x - u_p J_p \\ f_y - v_p J_p \\ q_p + (u_p f_x + v_p f_y) - (E_p/\rho_p) J_p \\ 0 \end{bmatrix} \tag{8.11}$$

In Eqs. (8.6) and (8.11), J_p is the mass regression rate (the combustion rate) of the discrete phase and it is determined by the combustion model of the discrete phase. ρ_p is the density of the discrete phase in the suspension. u_p and v_p are the x- and

y-components of the velocity of the discrete phase, respectively. E_p and N_p are the total energy per unit volume and the particle number density of the discrete phase, respectively. They can be calculated by

$$E_p = \rho_p e_p + \frac{1}{2}\rho_p(u_p^2 + v_p^2), \quad N_p = \frac{6\rho_p}{\pi\rho_m d_p^3}, \tag{8.12}$$

where e_p is the specific internal energy of the discrete phase and can also be obtained from a species thermodynamic data base as a function of discrete phase temperature T_p; d_p is the diameter of the particle; and ρ_m is the material density of the discrete phase.

Additionally, the x- and y-components of drag force acting on the discrete phase, f_x and f_y, can be modelled as follows:

$$\begin{cases} f_x = N_p \dfrac{\pi}{8} C_D d_p^2 \rho |V - V_p|(u - u_p) \\ f_y = N_p \dfrac{\pi}{8} C_D d_p^2 \rho |V - V_p|(v - v_p) \end{cases}, \tag{8.13}$$

where C_D is the drag coefficient,

$$C_D = \begin{cases} \frac{24}{\mathrm{Re}_p}\left(1 + \frac{1}{6}\mathrm{Re}_p^{2/3}\right), & \text{for } \mathrm{Re}_p < 1000 \\ 0.424, & \text{for } \mathrm{Re}_p \geq 1000 \end{cases}. \tag{8.14}$$

In Eqs. (8.13) and (8.14), the relative velocity and the relative Reynolds number Re_p between the gas and discrete phases can be calculated by

$$|V - V_p| = \left[(u - u_p)^2 + (v - v_p)^2\right]^{1/2}, \tag{8.15}$$

and

$$\mathrm{Re}_p = \frac{\rho d_p |V - V_p|}{\mu}, \tag{8.16}$$

where μ is the viscosity coefficient of gas. Accordingly, the convection heat transfer between the two phases is expressed as follows:

$$q_p = N_p \pi d_p \lambda \mathrm{Nu}_p(T - T_p), \tag{8.17}$$

with the two-phase Nusselt number expressed as functions of Re_p and Prandtl number Pr:

$$\mathrm{Nu}_p = 2 + 0.459\,\mathrm{Re}_p^{0.55}\,\mathrm{Pr}^{0.33}. \tag{8.18}$$

The CESE method has been proven to be of high accuracy and good stability in solving gaseous detonation problems in Sect. 8.1, and it is chosen to solve the two-phase detonation problems as well. That is, Eqs. (8.4) and (8.10) in the Eulerian–Eulerian framework can be solved using a CESE method, such as the a-α scheme. Theoretically, the source terms in the governing equations can be addressed together with the space–time integration in the CESE method [11–15]. However, a separated treatment of source terms is always employed in solving two-phase detonations, because the Jacobian matrixes of the interphase interaction and chemical reaction source terms in coupling treatment are rather complicated to calculate. Notably, with the source terms treated separately, it has been proven that the good accuracy and stability of the CESE method are preserved in high-speed reactive flow simulations [5, 16, 17]. On the other hand, the characteristic time scales of the interphase interaction and chemical reaction source terms are much smaller than that of flow dynamics, always leading to stiffness problems in two-phase detonation simulations. To overcome this problem, the operator-splitting technique with multiple sub-time steps [18] is always employed and then the source terms of interphase interactions and chemical reactions are explicitly integrated as ordinary differential equations. The detailed implementation process under the Eulerian–Eulerian framework can be illustrated as follows:

$$
\begin{cases}
\mathbf{U}_n \xrightarrow[\mathbf{S}=\mathbf{W}=0]{\text{CESE}} \widetilde{\mathbf{U}}_{n+1} \\
\mathbf{U}_p \xrightarrow[\mathbf{S}_p=0]{\text{CESE}} \widetilde{\mathbf{U}}_{p,n+1}
\end{cases}
\Rightarrow
\begin{cases}
\Delta t' = \Delta t/N \\
\mathbf{U}_{n+1}^{(0)} = \widehat{\mathbf{U}}_{n+1} \\
\mathbf{U}_{p,n+1}^{(0)} = \widetilde{\mathbf{U}}_{p,n+1}
\end{cases}
$$

$$
\Rightarrow
\begin{cases}
\mathbf{U}_{n+1}^{(m)}, \mathbf{U}_{p,n+1}^{(m)} \to \mathbf{S}^{(m)}, \mathbf{W}^{(m)}, \mathbf{S}_p^{(m)} \\
\mathbf{U}_{n+1}^{(m+1)} = \mathbf{U}_{n+1}^{(m)} + \Delta t'[\mathbf{S}^{(m)} + \mathbf{W}^{(m)}] \\
\mathbf{U}_{p,n+1}^{(m+1)} = \mathbf{U}_{p,n+1}^{(m)} + \Delta t' \cdot \mathbf{S}_p^{(m)}
\end{cases}
$$

$$
\Rightarrow
\begin{cases}
\mathbf{U}_{n+1} = \mathbf{U}_{n+1}^{(N)} \\
\mathbf{U}_{p,n+1} = \mathbf{U}_{p,n+1}^{(N)}
\end{cases}, \tag{8.19}
$$

where the subscripts n and m refer to the global time step (Δt) and the sub-time step ($\Delta t'$), respectively, and N is the total number of sub-time steps within one global convection time step of the CESE method. Depending on the degree of stiffness in the problem, N can be chosen to be $10-20$.

To test the accuracy and robustness of the CESE method in two-phase detonation simulations under the Eulerian–Eulerian framework, Wang et al. [15] applied the CESE method to simulate the detonation synthesis of titania (TiO_2) nanoparticles. The corresponding chemical reaction can be described as follows:

$$
TiCl_4(g) + O_2(g) + 2H_2(g) \to TiO_2(s) + 4HCl(g) \tag{8.20}
$$

As seen, it involves TiO_2 solid particles as well as $TiCl_4$, O_2, H_2 and HCl gases, implying that it is a gas–solid two-phase detonation problem. Simulation results showed that the simulated detonation profiles of gas density, pressure and temperature
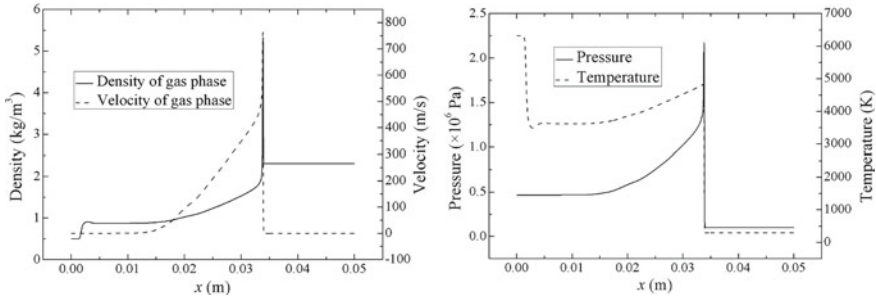
Fig. 8.8 Profiles of gas density, velocity, pressure and temperature in the detonation synthesis of nanosized TiO$_2$ particles. Courtesy of G. Wang [15]

are similar to those in the gaseous detonation wave, but a sudden drop of gas density can be captured after the detonation front because of phase transition from gas to solid particles, as shown in Fig. 8.8. Additionally, it is also found that the simulated particle size of the produced TiO$_2$ and the simulated peak pressure agreed well with other calculation and experiment data [19], respectively.

As for gas–liquid two-phase detonation, Wang et al. [3] solved the two-phase planar detonations of liquid hydrocarbon fuels with the same CESE method under the Eulerian–Eulerian framework. It is also found that the detonation profiles are similar to those of gaseous detonations, but with higher gas density, pressure and temperature behind the detonation front due to higher energy density of liquid fuels. Additionally, the length of the reaction zone is found to be longer as well, which is relative to the slower combustion rate of the liquid particle. Table 8.3 shows the comparisons of simulated detonation speeds to the experimental values. It can be revealed that the relative errors are less than 10% for particle sizes small than about 145 µm. As for two cases with larger particle sizes, the cases with relative errors exceeding 10%, it may be caused by the neglect of deformation of large liquid particles in modelling, since the combustion rate of liquid particle is influenced by particle deformation. All the above numerical results showed that that the CESE method is of high accuracy and good robustness in two-phase detonation simulations under an Eulerian–Eulerian framework.

The Eulerian–Eulerian framework is a simple and effective way to deal with the discrete phase in two-phase detonations, but it has some inherent limitations in modelling realistic two-phase suspension in industries or experiments with a specific particle size distribution, where a relatively wide range of particle diameters are involved [20–22]. Under the Eulerian–Eulerian framework, all particles within one numerical mesh are assumed to be in the same states, such as the same particle size, temperature, velocity and so on. However, the number of particles within one mesh may be large, and their states may differ depending on their initial states and interaction histories with gas phase. Moreover, the forces and the heat transfers between the gas and particles also differ by the particle size, resulting in different temperatures and

Table 8.3 Comparisons of detonation speeds of liquid hydrocarbon fuels between simulations and experiments. Courtesy of G. Wang [3]

Fuel	Particle size (μm)	Equivalence ratio	Experiment (m/s)	Simulation (m/s)	Relative error (%)
C_6H_{14}	20–30	0.41	1670	1544.83	−7.50
C_6H_{14}	20–30	0.49	1720	1671.30	−2.83
C_6H_{14}	20–30	0.56	1700	1773.92	4.35
C_6H_{14}	20–30	0.68	1780	1935.89	8.76
$C_{10}H_{20}$	145	1.0	2130	1946.93	−8.59
$C_{10}H_{20}$	375	0.914	1810–1850	1996.72	7.93–10.32
$C_{10}H_{20}$	1300	0.23	970–1250	1169.97	−6.40–20.62

velocities of particles within one computational mesh. Further, it had been demonstrated that many two-phase detonation characteristics are significantly influenced by the particle size. Consequently, the Eulerian–Eulerian framework is insufficient to reflect the true physics of realistic two-phase detonations with particle size distributions and to simulate them accurately, which raises the demand of modelling realistic two-phase detonation under the Eulerian–Lagrangian framework where the discrete particles are tracked individually.

Under the Eulerian–Lagrangian framework, Eq. (8.4) is still applied as the governing equation of the gas phase, and **U**, **F**, **G** and **W** have the same forms as Eq. (8.5). However, the two-phase interaction source term **S** is expressed in the following form instead,

$$
\mathbf{S} = \frac{1}{dV} \sum_{k=1}^{np}
\begin{bmatrix}
0 \\
\vdots \\
J_{pk} \\
\vdots \\
0 \\
-f_{xk} + u_{pk} J_{pk} \\
-f_{yk} + v_{pk} J_{pk} \\
-q_{pk} - \left(u_{pk} f_{xk} + v_{pk} f_{yk} \right) + \frac{1}{2} \left(u_{pk}^2 + v_{pk}^2 \right) \cdot J_{pk} + e_{pk} \cdot J_{pk}
\end{bmatrix},
\tag{8.21}
$$

where subscript k represents all the quantities related to the kth particle (solid or liquid). To include all effects of particles into the source term **S** of the gaseous equation, the summation is done within the gaseous mesh element dV, and np is the number of particles in dV.

The motion of every particle is then descripted using Newton's laws of motion instead of the Eulerian form Eq. (8.10). For the kth particle, the corresponding governing equation can be written as

$$\frac{d\mathbf{L}_{pk}}{dt} = \mathbf{S}_{pk},\tag{8.22}$$

where $\mathbf{L}_{pk}$ and $\mathbf{S}_{pk}$ are the vectors of the Lagrangian variables of the kth particle and the corresponding source terms, respectively, and they are expressed as

$$\mathbf{L}_{pk} = \begin{bmatrix} m_{pk}, x_{pk}, y_{pk}, m_{pk}u_{pk}, m_{pk}v_{pk}, E_{pk} \end{bmatrix}^{\mathrm{T}},\tag{8.23}$$

and

$$\mathbf{S}_{pk} = \begin{bmatrix} -J_{pk}, u_{pk}, v_{pk}, f_{xk}, f_{yk}, -e_{pk}J_{pk} + q_{pk} \end{bmatrix}^{\mathrm{T}}.\tag{8.24}$$

In Eq. (8.23), m_{pk} and E_{pk} are the mass and total internal energy of the kth particle, respectively, and $E_{pk} = m_{pk}{\cdot}e_{pk}$.

Further, the Lagrangian forms of the drag force and the convection heat of the kth particle can be easily derived from the according Eulerian forms Eqs. (8.13) and (8.17), as follows:

$$\begin{cases} f_{xk} = \dfrac{\pi}{8}C_{Dk}d_{pk}^2\rho\left|V - V_{pk}\right|\left(u - u_{pk}\right) \\[2mm] f_{yk} = \dfrac{\pi}{8}C_{Dk}d_{pk}^2\rho\left|V - V_{pk}\right|\left(v - v_{pk}\right) \end{cases},\tag{8.25}$$

and

$$q_{pk} = \pi d_{pk}\lambda \mathrm{Nu}_{pk}\left(T - T_{pk}\right).\tag{8.26}$$

Under the Eulerian–Lagrangian framework, the CESE method is again applied to solve the gas phase equation, while the source terms of interphase interactions and chemical reactions are treated separately and integrated explicitly as ordinary differential equations, along with the integration of particle Lagrangian equations, by using the operator-splitting technique as well, as depicted below,

$$\mathbf{U}_n \xrightarrow[\mathbf{S}=\mathbf{W}=0]{\text{CESE}} \tilde{\mathbf{U}}_{n+1} \Rightarrow \begin{cases} \Delta t' = \Delta t/N \\ \mathbf{U}_{n+1}^{(0)} = \tilde{\mathbf{U}}_{n+1} \\ \mathbf{L}_{pk,n+1}^{(0)} = \mathbf{L}_{pk,n} \end{cases}$$

$$\Rightarrow \begin{cases} \mathbf{U}_{n+1}^{(m)}, \mathbf{L}_{pk,n+1}^{(m)} \to \mathbf{S}^{(m)}, \mathbf{W}^{(m)}, \mathbf{S}_{pk}^{(m)} \\ \mathbf{U}_{n+1}^{(m+1)} = \mathbf{U}_{n+1}^{(m)} + \Delta t'[\mathbf{S}^{(m)} + \mathbf{W}^{(m)}] \\ \mathbf{L}_{pk,n+1}^{(m+1)} = \mathbf{L}_{pk,n+1}^{(m)} + \Delta t' \cdot \mathbf{S}_{pk}^{(m)} \end{cases}$$

$$\Rightarrow \begin{cases} \mathbf{U}_{n+1} = \mathbf{U}_{n+1}^{(N)} \\ \mathbf{L}_{pk,n+1} = \mathbf{L}_{pk,n+1}^{(N)} \end{cases}.\tag{8.27}$$

Notably, the Eulerian–Lagrangian framework had rarely been developed to solve high-speed reactive two-phase flows, mainly because the number of fine particles has proven too large and simulations too expensive to achieve in the past. Nowadays, with the rapid development of computer technologies, parallel computation techniques such as the Message Passing Interface (MPI) technique may help to make simulations of two-phase detonations with fine particles under an Eulerian–Lagrangian framework possible. However, the application of MPI parallel computation technique to an Eulerian–Lagrangian framework is not as straightforward as that under a purely Eulerian framework. A large amount of information exchange from one computational core to another is needed for calculations of the interaction source terms between two phases, leading to formidable communication cost even exceeding the computational cost, especially when the Eulerian framework is staggered with the Lagrangian framework due to the relative movement between these two phases.

To solve the communication problem of parallel computing technique of high-speed two-phase flows under an Eulerian–Lagrangian framework, the traditional static data structure, such as multi-dimensional array, should be avoided using to store the information of the discrete phase described under the Lagrangian framework. Notably, the order of particles presented under the Lagrangian framework is not important and does not need to be preserved as its initial order. The only operation required for coding is to traverse every particle one by one. Inspired by the above facts, dynamic data structures can be introduced to store particle information and solve the communication problem under the Eulerian–Lagrangian framework. For example, the structures and the corresponding operations of linked lists used to store particle information are schematically shown in Fig. 8.9. As seen, each CPU owns one linked list to store information of the particles that are at the corresponding locations to the gas phase. In each linked list, one node represents one particle and consists of two parts: the data part that stores particle information, and the pointer that points to the next node of the particle for traversing. The pointer of the last node always points to the "NULL", implying that the linked list has ended.

Additionally, there are four basic operations for the linked list to adjust the particle's storage location: allocate, free, delete and insert, as depicted in Fig. 8.9. The "allocate" operation is used to allocate new memory to store information about a "new" particle, the "free" operation to free the memory that stores information about an "old" particle, the "delete" operation to disconnect one particle from the linked list, and the "insert" operation to connect one particle at the end of the linked list. When the particle phase is staggered with the gas phase because of relative motion, the information of the particles, whose locations exceed the corresponding location ranges assigned to the present CPUs, will be transferred to and stored in the CPUs with the correct particle location ranges. For example, for CPU A in Fig. 8.9, where one particle is removed, the following sequence of operations is needed:

(1) send the information of the specific particle to CPU B;
(2) delete the particle node from the linked list;
(3) free the memory of the separated node.
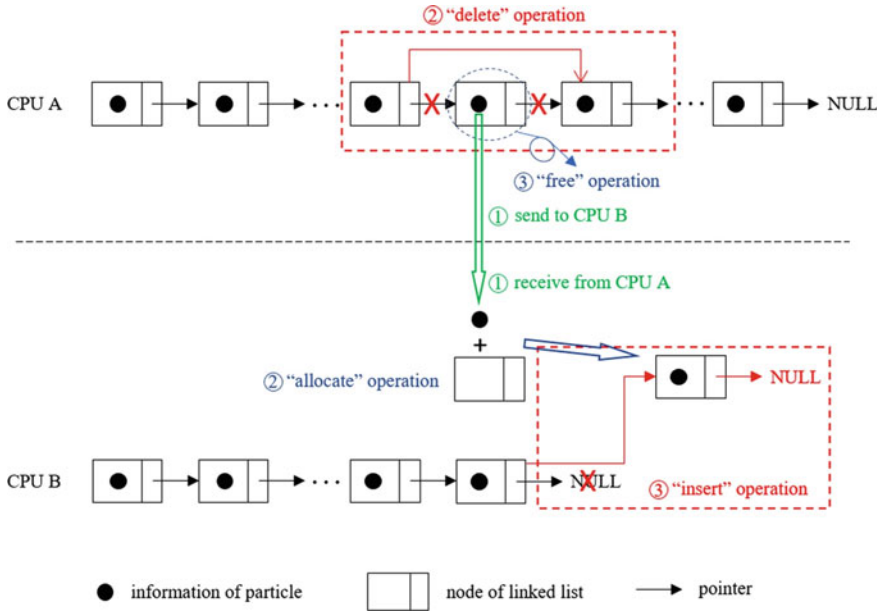
Fig. 8.9 Linked lists and operation sequences. Courtesy of Z. J. Zhang [9]

Meanwhile, for CPU B in Fig. 8.9, where a new particle is received, the corresponding operation sequence is:

(1) receive the information of the new particle from CPU A;
(2) allocate a new node and store the information into the data field of the node;
(3) insert the new particle node at the end of the linked list.

With this dynamic data structure and the corresponding operation sequences, the information about Lagrangian particles is always stored in the CPUs of the correct Eulerian coordinates, and therefore excessive communication between CPUs when calculating gas-particle interactions is avoided, as shown in Fig. 8.10. Moreover, with the limitation of the global time step by the CFL condition, only "one" particle at most will cross the CPU boundary at every iteration; that is, the information of "one" particle at most will be transferred to the other CPU at one iteration step by the above operation sequence. As a result, the communication cost of the MPI parallel for the gas-particle interaction calculation will be reduced from $O(N)$ to $O(1)$ when using this data structure. Here, N is the number of particles stored in each CPU.

To demonstrate the MPI parallelization performance with the use of the above linked lists and the corresponding operation sequences, Fig. 8.11 depicts the speedup parameters of a 2D Al-air detonation propagation problem using the CESE method under an Eulerian–Lagrangian framework, which involves approximately 24 million Eulerian meshes and about 75 million Lagrangian particles in the computational domain. This test case is simulated on the Tianhe-2 supercomputer from China with core numbers of 1, 2, 3, 4, 6, 12, 24, 48, 96, 192 and 384. The use of one core means
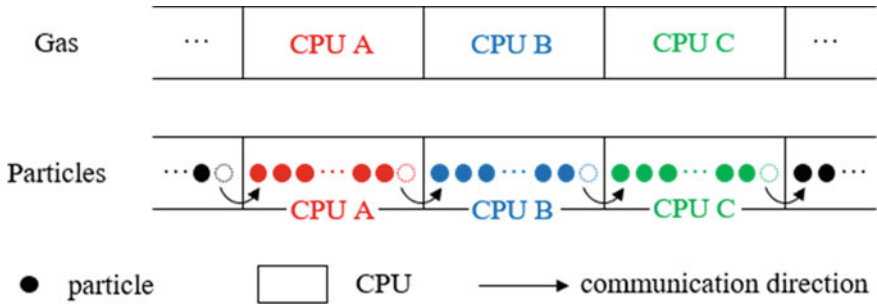
Fig. 8.10 Dynamic data structure using the linked list in the Eulerian–Lagrangian framework. Courtesy of Z. J. Zhang [9]

that the simulation is done serially. As depicted in Fig. 8.11, when 384 cores are used, the code using linked lists still has a reasonable parallel efficiency of about 50% for the tested problem, while MPI parallelization under the Eulerian–Lagrangian framework using multi-dimensional static arrays is impossible, even with only 2 cores, as the communication cost is shown to be unacceptably large. This means that the linked list dynamic data structure works well in the MPI implement when solving two-phase detonations under the Eulerian–Lagrangian framework.

To test the performances of CESE method in simulations of two-phase detonation under the Eulerian–Lagrangian framework, Shen et al. [23] simulated the gas–liquid two-phase detonations in the $C_{10}H_{22}$–O_2/air systems with different fuel droplet sizes and equivalence ratios. A deficit in the detonation speed compared to the corresponding purely gaseous one was observed in the gas–liquid suspensions with lean fuel and larger droplet sizes, while an increase in the detonation speed was observed

Fig. 8.11 Speedups of the tested problem using linked lists. Courtesy of Z. J. Zhang [9]
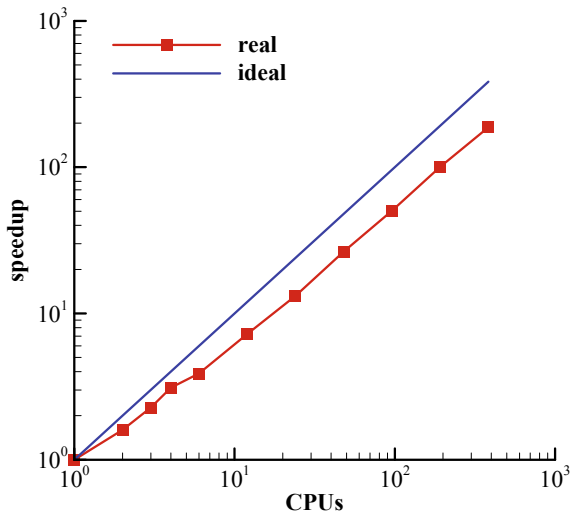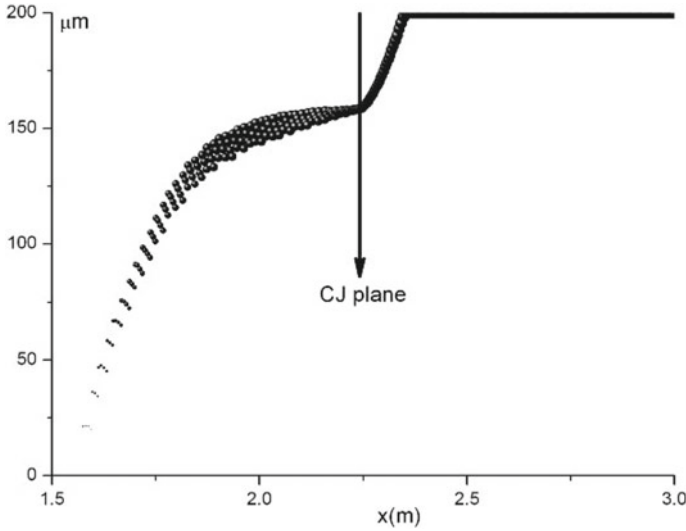
Fig. 8.12 Distribution of particle size near the detonation front in gas–liquid two-phase detonations. Courtesy of H. Shen [23]

with very rich fuel. This special two-phase detonation feature in rich fuel is due to the slow combustion rate of the droplet, leading to a large amount of fuel burnt behind the C–J point, as shown in Fig. 8.12. As a result, the efficient equivalence ratio is reduced and detonation speed shifts to the equivalence ratio direction. The accuracy and robustness of the CESE method applied to two-phase detonation simulations under an Eulerian–Lagrangian framework had been demonstrated.

Notably, simulations of the monodisperse two-phase detonation problems (with single particle size) under the Eulerian–Lagrangian framework always present the same results as those performed under the Eulerian–Eulerian framework. The raising of modelling two-phase detonations under the Eulerian–Lagrangian framework is in fact to simulate the realistic polydisperse two-phase detonations, where a particle size distribution is involved and the particle diameters are in a relatively wide range. One of the frequently used particle size distribution models to describe the polydisperse suspension is the log-normal distribution function, expressed by the number frequency distribution function as follows [24]:

$$f_n(d_p) = \frac{1}{\sqrt{2\pi}\sigma_0} \exp\left[-\frac{1}{2}\left(\frac{\ln d_p - \ln d_{nM}}{\sigma_0}\right)^2\right] \frac{1}{d_p}, \tag{8.28}$$

where d_{nM} and σ_0 are the number median diameter and standard deviation of the distribution, respectively. In practice, the specific particle size distribution of a polydisperse suspension is always given by other two particle size parameters that can be

easily measured, namely the volume-average diameter ($\overline{d}$) and the mass-weighted-average particle diameter ($\overline{d}_m$). For example, the measured polydisperse parameters of the Al powder tested in the experiment of Zhang et al. [20] are $\overline{d} = 2\,\mu\text{m}$ and $\overline{d}_m = 3.3\,\mu\text{m}$. The relationships between (d_{nM}, σ_0) and ($\overline{d}, \overline{d}_m$) can be obtained through the integrations of Eq. (8.28), as follows

$$\begin{cases} \overline{d} = \left[\int_0^\infty s^3 f_n(s)ds\right]^{1/3} = d_{nM} e^{\frac{3}{2}\sigma_0^2} \\ \overline{d}_m = \left[\int_0^\infty s^4 f_n(s)ds\right] / \left[\int_0^\infty s^3 f_n(s)ds\right] = d_{nM} e^{\frac{7}{2}\sigma_0^2} \end{cases}. \tag{8.29}$$

Consequently, the corresponding parameters of the above-mentioned Al powder are $d_{nM} = 1.37\,\mu\text{m}$ and $\sigma_0 = 0.5$. However, it is a good way to use the parameter set of ($\overline{d}, \sigma_0$) to discuss the polydisperse suspensions in numerical simulations, because the effects of particle size distribution on polydisperse detonations can be easily discussed by varying σ_0. Some specific particle size distributions with fixed $\overline{d} = 2\,\mu\text{m}$ and different σ_0 values are depicted in Fig. 8.13.

In the followings, the typical Al-Air detonation problems with different particle size distributions, taking as an example, are simulated by the above numerical algorithm under the Eulerian–Lagrangian framework, to show the capacities of the CESE method in solving polydisperse two-phase detonation and show the differences between monodisperse and polydisperse two-phase detonations. The single-step global chemical reaction occurs in the Al-air suspension is

$$2\text{Al(s)} + 3/2\text{O}_2\text{(g)} \rightarrow \text{Al}_2\text{O}_3\text{(s)} \tag{8.30}$$
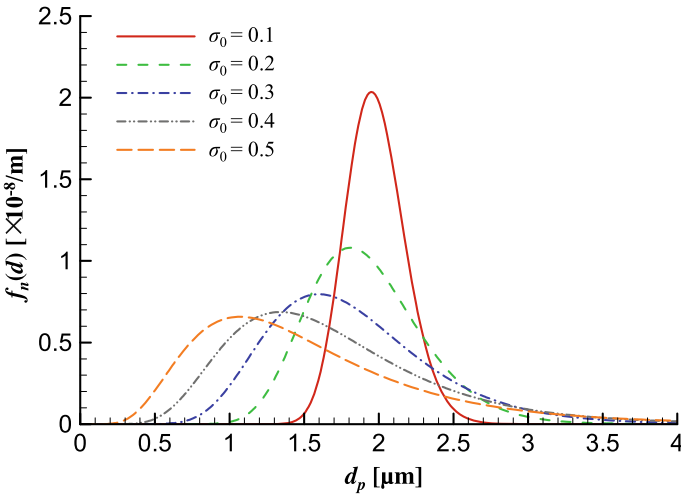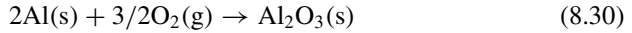
Fig. 8.13 Log-normal particle size distributions with $\overline{d} = 2\,\mu\text{m}$. Courtesy of Z. J. Zhang [25]

To model the combustion rate of every single Al particle, the surface-kinetic-oxidation and diffusion hybrid combustion model (originally proposed by Zhang et al. [26]) can be employed. The Lagrangian form of the combustion rate of the kth Al particle is given as

$$J_{pk,\text{Al}} = \pi d_{pk,\text{Al}}^2 C_{O_2} \frac{\nu_{\text{Al}} W_{\text{Al}}}{\nu_{O_2} W_{O_2}} \cdot \frac{k_{dk} k_{sk}}{k_{dk} + k_{sk}}. \tag{8.31}$$

where the reaction rates for the diffusion-controlled and kinetic-controlled combustion regime are expressed by

$$\begin{cases} k_{dk} = \dfrac{\nu_{O_2} W_{O_2}}{\nu_{\text{Al}} W_{\text{Al}}} \dfrac{\rho_{\text{Al}} d_{pk,\text{Al}}}{2 C_{\text{total}} K d_{pk0,\text{Al}}^2} \left(1 + 0.276 \text{Re}_{pk}^{1/2} \text{Pr}^{1/3} \right) . \\ k_{sk} = k_0 e^{-E_a / R_0 T_{sk}} \end{cases} \tag{8.32}$$

In Eqs. (8.31) and (8.32), ν_{Al} and ν_{O_2} are the stoichiometric coefficients for Al and O_2, respectively; C_{O_2} and C_{total} are the mole concentrations of O_2 and gas mixture, respectively; T_{sk} is the particle surface temperature; and K, k_0 and E_a are model constants.

As depicted in Fig. 8.14. Some special two-phase detonation features corresponding to multi-phase interaction can be clearly identified in the detonation front structures of monodisperse Al-air detonation. The first feature is known as "double peaks" in the gas pressure, density and velocity profiles (Fig. 8.14a, c), which is distinctly different from the single peak feature observed in gaseous detonations. It is demonstrated, from the one-dimensional flow theory in gas dynamics that the second peak in the detonation front structures of monodisperse Al suspension is caused by the dominant stage of heat transfer due to intense phase transition (Al evaporation) at a specific location after the shock front. The second feature, shown in Fig. 8.14b, is the plateau of particle temperature due to Al evaporation, which is equal to 2750 K and results in the observed "kink" in the gas temperature profile due to the intense heat transfer between the gas and particles. The third feature, shown in Fig. 8.14c, is the particle velocity lag in the velocity relaxation process, resulting in the alternative forces acting on particles and momentum transfers between the gas and particles. All these two-phase features for monodisperse Al-air detonation are similar with those obtained by Zhang et al. [26] and Teng and Jiang [27, 28].

However, as for the polydisperse detonation with a log-normal particle size distribution of $\sigma_0 = 0.5$, most features of monodisperse Al detonation, including the double peaks in gas pressure, density, velocity profiles and the kink in gas temperature profile, disappear, and only single peaks of gas quantities exist in the detonation front, which are quite similar to the wave front structures in gaseous detonations. The double peaks in the detonation front is demonstrated theoretically to be attributed to the space-dispersed phase transition processes of particles of different sizes result in an overall moderate heat transfer intensity, which hinders the formation of the heat-transfer-dominant stage. These differences between monodisperse and polydisperse
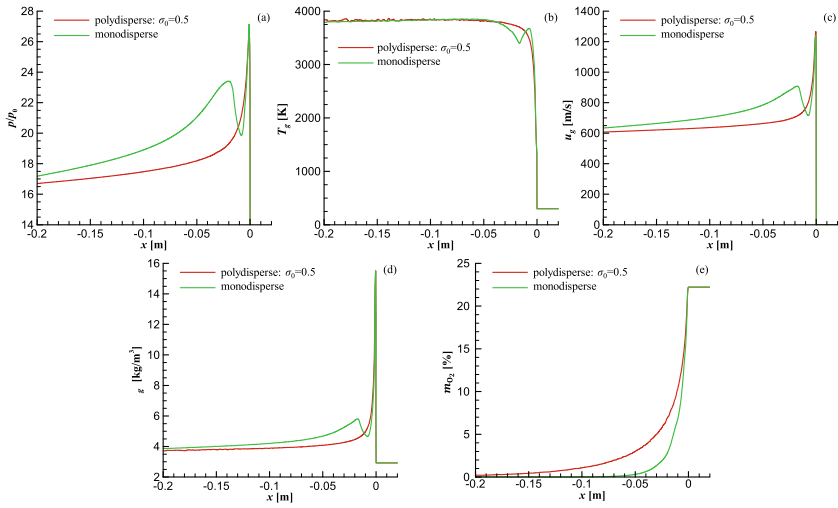
Fig. 8.14 Comparison of front structures in gas phase of polydisperse and monodisperse Al-air detonations: **a** pressure, **b** temperature, **c** velocity, **d** density, **e** mass fraction of O_2. Courtesy of Z. J. Zhang [25]

two-phase detonations are relative to the multiple timescales and length scales in polydisperse suspension with a continuous particle size distribution.

As for the multi-dimensional two-phase detonation features, Fig. 8.15 shows the comparisons of cellular detonation flow fields between monodisperse and polydisperse detonations. As indicated in Fig. 8.15a, typical cellular detonation structures, including pairs of triple points, Mach stems, incident shock waves and pairs of transverse waves, can be observed in both the monodisperse and polydisperse detonation fronts. Moreover, the gas temperature and the O_2 species mass fraction distributions for both monodisperse and polydisperse cases are non-uniform behind the detonation fronts, which are characterized by irregular local (high and low) temperature and O_2 concentration zones, respectively, as shown in Fig. 8.15b, c. These irregular distributions of the flow field parameters in monodisperse and polydisperse detonations are both caused by the periodical motions of triple points along the detonation fronts. All these features in two-phase detonations are similar to those observed in gaseous detonations, except for the transverse waves. The transverse waves in both the monodisperse and polydisperse detonation fronts are weak and degenerate fairly fast in the rear flows, which are different from the strong transverse waves observed in purely gaseous detonations. According to Zhang et al. [26], these weak transverse waves can be attributed to the slow diffusion-controlled combustion of the majority of Al particles after their kinetic-inductions and a considerable amount of condensed Al oxide formed in the detonation products without contributing to gas pressure.
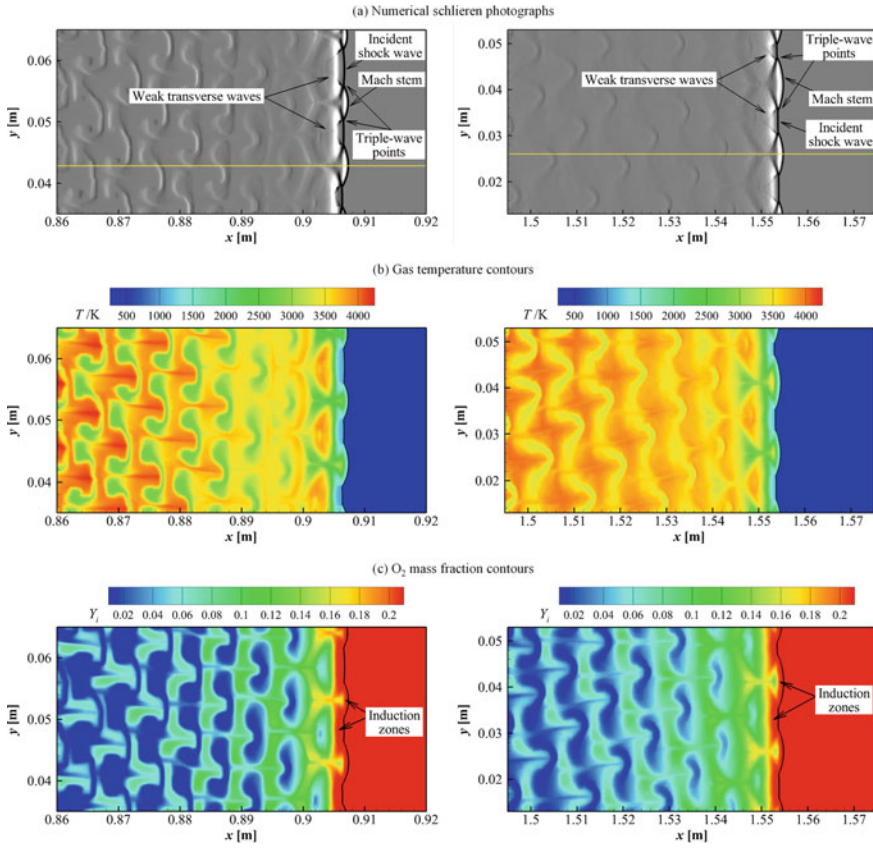
Fig. 8.15 Comparison of flow fields in Al-air detonation fronts. Left: monodisperse with $d_p = 2\,\mu m$ at $t = 0.6$ ms. Right: polydisperse with $\bar{d} = 2\,\mu m$ and $\sigma_0 = 0.5$ at $t = 1$ ms. Courtesy of Z. J. Zhang [25]

Further, larger detonation cell sizes are expected in polydisperse two-phase detonation compared to monodisperse detonation, since the reaction zone is larger, as shown in Fig. 8.16 by peak pressure contours. The estimated cell sizes of the monodisperse detonation and the polydisperse detonations ($\sigma_0 = 0.5$ and 0.8) are $\lambda_{\mathrm{monodisperse}} = 10.5 \pm 0.5$ mm, $\lambda_{0.5} = 13.3 \pm 0.8$ mm and $\lambda_{0.8} = 20.0 \pm 1.8$ mm, respectively. $\lambda_{0.5}$ is 27% larger than $\lambda_{\mathrm{monodisperse}}$, and $\lambda_{0.8}$ is even 190% larger. Figure 8.17 plots these detonation cell sizes as a function of the square of standard deviation σ_0^2 by a logarithmic scatter diagram, together with those of other three polydisperse detonations with different σ_0. An approximately linear relationship between $\ln \lambda$ and σ_0^2 can be captured. Then, linear fitting is employed, and the fitting function appears to be $\ln \lambda = 0.9367\sigma_0^2 + 0.0655$, implying that the detonation cell size λ of polydisperse suspensions is an exponent function of the square of standard deviation of the distribution σ_0^2. This is an important quantitative relation of polydisperse two-phase
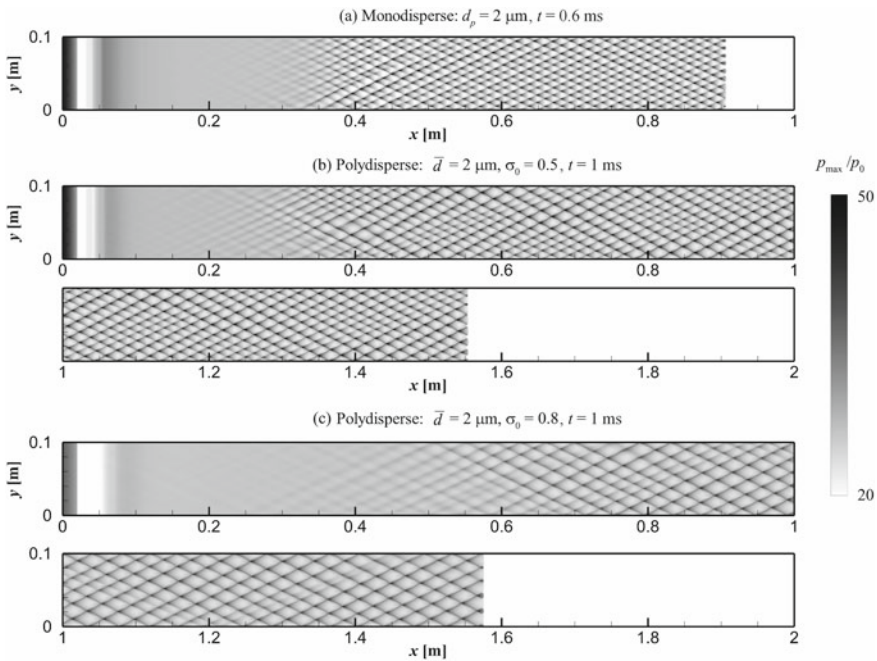
Fig. 8.16 Comparison of cellular detonations (peak pressure contours) of Al-air mixtures: **a** monodisperse with $d_p = 2\,\mu m$, **b** polydisperse with $\bar{d} = 2\,\mu m$ and $\sigma_0 = 0.5$, and **c** polydisperse with $\bar{d} = 2\,\mu m$ and $\sigma_0 = 0.8$. Courtesy of Z. J. Zhang [25]

Fig. 8.17 Detonation cell size λ of Al-air suspension as a function of square of standard deviation σ_0^2
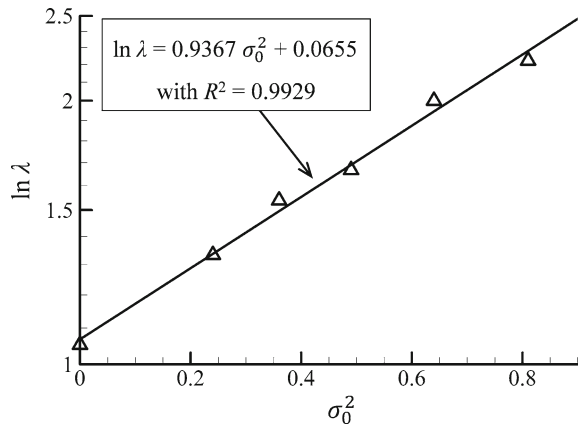
detonation to evaluate the effect of particle size distribution on detonation cell size. From the above results, the capacities of the CESE method in polydisperse two-phase detonation simulations under the Eulerian–Lagrangian framework is demonstrated as well.

References

1. Shen, H., & Parsani, M. (2017). The role of multidimensional instabilities in direct initiation of gaseous detonations in free space. *Journal of Fluid Mechanics, 813*, R4.
2. Shi, L. S., Uy, K. C. K., & Wen, C. Y. (2020). The re-initiation mechanism of detonation diffraction in a weakly unstable gaseous mixture. *Journal of Fluid Mechanics, 895*, A24.
3. Wang, G., Zhang, D. L., Liu, K. X., & Wang, J. T. (2010). An improved CE/SE scheme for numerical simulation of gaseous and two-phase detonations. *Computers & Fluids, 39*(1), 168–177.
4. Shen, H., Liu, K. X., & Zhang, D. L. (2011). Three-dimensional simulation of detonation propagation in a rectangular duct by an improved CE/SE scheme. *Chinese Physics Letters, 28*(12), 124705.
5. Shi, L. S., Shen, H., Zhang, P., Zhang, D. L., & Wen, C. Y. (2017). Assessment of vibrational non-equilibrium effect on detonation cell size. *Combustion Science and Technology, 189*(5), 841–853.
6. Uy, K. C. K., Shi, L. S., & Wen, C. Y. (2020). Numerical analysis of the vibration-chemistry coupling effect on one-dimensional detonation stability. *Aerospace Science and Technology, 107*, 106327.
7. Uy, K. C. K., Shi, L. S., Hao, J., & Wen, C. Y. (2020). Linear stability analysis of one-dimensional detonation coupled with vibrational relaxation. *Physics of Fluids, 32*(12), 126101.
8. Wang, F., Weng, C. S., Wu, Y. W., Bai, Q. D., Zheng, Q., & Xu, H. (2020). Numerical research on kerosene/air rotating detonation engines under different injection total temperatures. *Aerospace Science and Technology, 103*, 105899.
9. Zhang, Z., Wen, C. Y., Liu, Y., Zhang, D. L., & Jiang, Z. (2019). Application of CE/SE method to gas-particle two-phase detonations under an Eulerian-Lagrangian framework. *Journal of Computational Physics, 394*, 18–40.
10. McBride, B. J. (2002). *NASA Glenn coefficients for calculating thermodynamic properties of individual species*. National Aeronautics and Space Administration, John H. Glenn Research Center.
11. Yu, S. T., Chang, S. C., Yu, S. T., & Chang, S. C. (1997). Treatments of stiff source terms in conservation laws by the method of space-time conservation element/solution element. In *35th Aerospace Sciences Meeting and Exhibit*.
12. Park, S. J., Yu, S. T., Lai, M. C., Chang, S. C., & Jorgenson, C. (1999). Numerical calculation of unstable detonations by the method of space-time conservation element and solution element. In *37th Aerospace Sciences Meeting and Exhibit*.
13. Wang, B., He, H., & Yu, S. T. J. (2005). Direct calculation of wave implosion for detonation initiation. *AIAA Journal, 43*(10), 2157–2169.
14. Wang, J. T., Liu, K. X., & Zhang, D. L. (2009). An improved CE/SE scheme for multi-material elastic–plastic flows and its applications. *Computers & Fluids, 38*(3), 544–551.
15. Wang, G., Zhu, H. Y., Sun, Q. H., Zhang, D. L., & Liu, K. X. (2011). An improved CE/SE scheme and its application to dilute gas–particle flows. *Computer Physics Communications, 182*(8), 1589–1601.
16. Wen, C. Y., Saldívar Massimi, H., & Shen, H. (2018). Extension of CE/SE method to non-equilibrium dissociating flows. *Journal of Computational Physics, 356*, 240–260.
17. Uy, K. C. K., Shi, L. S., & Wen, C. Y. (2019). Prediction of half reaction length for H_2–O_2-Ar detonation with an extended vibrational nonequilibrium Zel'dovich−von Neumann−Döring (ZND) model. *International Journal of Hydrogen Energy, 44*(14), 7667–7674.
18. Ropp, D. L., & Shadid, J. N. (2009). Stability of operator splitting methods for systems with indefinite operators: Advection–diffusion–reaction systems. *Journal of Computational Physics, 228*(9), 3508–3516.
19. Ouyang, X., Li, X., Yan, H., Mo, F., & Zhao, C. (2008). Preparation and characterization of nanosized TiO_2 powders by gaseous detonation method. *Materials Science and Engineering: B, 153*(1–3), 21–24.

20. Zhang, F., Murray, S., & Gerrard, K. (2006). Aluminum particles-air detonation at elevated pressures. *Shock Waves, 15*(5), 313–324.
21. Zhang, F., Gerrard, K., & Ripley, R. C. (2009). Reaction mechanism of aluminum-particle-air detonation. *Journal of propulsion and power, 25*(4), 845–858.
22. Meda, L., Marra, G., Galfetti, L., Severini, F., & De Luca, L. (2007). Nano-aluminum as energetic material for rocket propellants. *Materials Science and Engineering: C, 27*(5–8), 1393–1396.
23. Shen, H., Wang, G., Liu, K. X., & Zhang, D. L. (2012). Numerical simulation of liquid-fueled detonations by an Eulerian-Lagrangian model. *International Journal of Nonlinear Sciences and Numerical Simulation, 13*(2), 177–188.
24. Crowe, C., Schwarzkopf, J., Sommerfeld, M., & Tsuji, Y. (2011). Multiphase flows with droplets and particles. CRC Press.
25. Zhang, Z., Wen, C. Y., Liu, Y., Zhang, D. L., & Jiang, Z. (2020). Effects of different particle size distributions on aluminum particle–air detonation. *AIAA Journal, 58*(7), 3115–3128.
26. Zhang, F. (2009). Detonation of gas-particle flow. In *Shock wave science and technology reference library* (Vol. 4, pp. 87–168). Springer.
27. Teng, H., & Jiang, Z. (2013). Effects of different product phases in aluminum dust detonation modeling. *Science China Physics, Mechanics and Astronomy, 56*(11), 2178–2185.
28. Teng, H., & Jiang, Z. (2013). Numerical simulation of one-dimensional aluminum particle-air detonation with realistic heat capacities. *Combustion and Flame, 160*(2), 463–472.

Chapter 9
Other Applications

The CESE method has been applied to a wide range of scientific and engineering problems since its inception in the 1990s. Although solving CFD problem is the primary goal of the CESE method, this general approach is actually applicable to a variety of PDE systems with physical backgrounds different from fluid dynamics. This chapter mainly introduces the application highlights of CESE in several representative fields.

9.1 Hypersonic Aerodynamics

Typical high-speed aerodynamic problems can be solved by using the CESE methods. Successful examples of such applications include the supersonic flow over a blunted flat plate [1], Mach reflection of shock waves [2], unsteady viscous flows in rocket nozzles [3], counterflow jets for the reduction of aerothermal load on spacecrafts [4, 5], and the flow over roughness elements in a supersonic boundary layer [6].

With the increasing number of Mars exploration missions in recent years, the research interest of hypersonic aerothermodynamics has been revived. In a typical atmospheric-entry flight, the aircraft will encounter hypersonic gas flow, and the flow field is characterized by strong shock waves and thermochemical non-equilibrium effects. CFD has become a useful tool for in-depth understanding of complex flow physics and for complementary experimental prediction of aerodynamic characteristics.

Using a 2D axisymmetric CESE solver based on hybrid meshes, Wen et al. [7–9] conducted a systematic study of hypersonic chemically reacting non-equilibrium flows over spheres. Three different working gases: nitrogen, air, and carbon dioxide were used in their simulations. The thermochemical non-equilibrium effects considered therein include the vibrational energy relaxation of gas molecules as well as the dissociation and recombination reactions in the gas flows. The physically consistent coupled vibration–chemistry–vibration (CVCV) [10] two-temperature model was

employed. Here, the performance of the CESE method are shown by three different flow cases. The working gas, sphere radius, and the CESE simulation results [9] of the dimensionless shock stand-off distances (defined as the ratio of the stand-off distance to the sphere radius) of each case are listed in Table 9.1, along with the experimental and theoretical results of Wen and Hornung [11]. The CESE results are shown in good agreement with experimental and theoretical results. Further investigation of the shape of shock waves is presented in Fig. 9.1, in which the numerical density contours are overlaid on the experimental finite fringe differential interferograms. The shapes of the simulated bow shocks with the CESE method match the experimental results well.

Another important thermochemical non-equilibrium process in an atmospheric-entry flow is the ionization in a high-temperature shock layer. Massimi et al. [12] conducted a numerical investigation of hypersonic ionized air flows over rounded nose geometries, using a 2D axisymmetric CESE solver based on hybrid meshes. The multi-species NS equations were solved with the two-temperature seven-species Park model for ionization reactions. The weakly ionized air flow in the Radio Attenuation Measurement (RAM-C II) flight test at 71 km altitude was simulated and compared with experiments and numerical results of Candler and MacCormack [13]. The CESE results [12] of the maximum electron number densities along the direction normal to the vehicle's surface agree well with the flight test measurements and the numerical results of Candler and MacCormack [13].

Table 9.1 The working gases, sphere radii, and dimensionless shock stand-off distances (last three columns) in different flow cases

Case no	Gas	Radius (inch)	Experiment [11]	Theory [11]	Simulation [9]
1	N_2	1.5	0.100	0.095	0.100
2	Air	1.0	0.105	0.093	0.095
3	CO_2	2.0	0.088	0.084	0.087

(a) Case 1   (b) Case 2   (c) Case 3
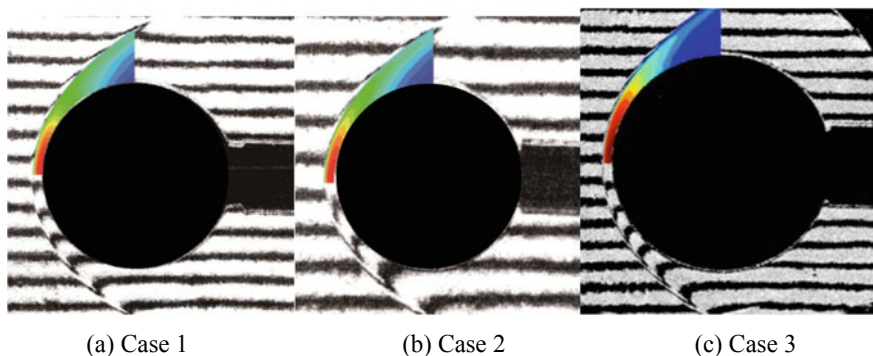
Fig. 9.1 Comparison of experimental shapes of the bow shocks with the numerical predictions by CESE. Courtesy of C. Y. Wen [9]

9.2   Aeroacoustics

Thanks to the rapid development of CFD, it is now expected to solve aeroacoustics problems through high-resolution numerical simulations of unsteady compressible Euler or NS equations. However, in many computational aeroacoustic problems, the coexistence of shock waves and small disturbances (acoustic waves) imposes stringent requirements on numerical methods.

The sound–shock interaction problem was used by Wang et al. [14] to examine the accuracy of the CESE method for aeroacoustic problems involving shock waves. Loh et al. [15] further considered three selected problems, namely a linear benchmark problem, instability waves on a free shear layer, and shock–vortex interactions, to illustrate the feasibility of using CESE as a tool for computational aeroacoustics (CAA). Good numerical results with high resolution and low dispersion were demonstrated by their second-order CESE scheme. Additionally, the authors also pointed out that the non-reflecting boundary condition, which plays an important role in CAA, is much simpler to implement in the CESE method than in traditional methods (see [15, 16]). To deal with practical multi-dimensional CAA problems where large disparity in grid cell size is inevitable, Yen et al. [17, 18] modified the Courant number insensitive (CNI)-CESE scheme [17] and the local time-stepping CESE scheme [18] to improve the numerical efficiency. A 3D simulation was performed for the propagation of a Gaussian acoustic pulse and a vorticity wave embedded in a Mach 0.5 mean flow. Once again, the CESE results were in good agreement with the corresponding analytical solution in preserving both the form and amplitude of the waves [17]. Yen [18] further conducted a series of 3D CESE simulations for the Helmholtz resonator problem. By using a second-order CESE scheme, excellent agreement between the linear acoustic theory and the CESE solution was achieved.

The CESE method has also been employed as a numerical tool for CAA problems in practical engineering applications related to engines and aircrafts. To shed light upon an aeroacoustic resonance phenomenon often encountered by convergent–divergent nozzles under transonic conditions, Loh and Zaman [19, 20] developed an axisymmetric NS solver using the CESE method and conducted a numerical investigation. A 3D CESE NS solver was also implemented by Loh et al. [15] and applied to compute the screech noise generated by an under-expanded supersonic jet. Kim et al. [21] simulated the supersonic unsteady flow over an open cavity to investigate the mixing-enhancement and flame-holding capability in the scramjet engine. The CESE method solving 2D NS equations successfully captured the self-sustained oscillations in the supersonic cavity flows. The computed frequencies and amplitudes of the pressure oscillations compare favourably with the theoretical and experimental data. Cheng et al. [22] also studied subsonic and supersonic flows over open-cavity geometries by an unsteady NS solver based on the CESE method. When compared with experimental and analytical data, the generation and propagation of flow-induced acoustic waves were faithfully captured, and satisfactory results were obtained for the oscillation frequency of the dominant mode.

9.3   Solid Dynamics

In recent years, high-resolution discontinuity-capturing numerical methods originally devised for CFD problems have been utilized to solve nonlinear solid-dynamic problems with Eulerian formulations. Among these methods, the CESE method has attracted considerable attention from researchers in computational solid dynamics because of its simple logic, high accuracy, and ability to capture the behaviours of nonlinear waves.

Wang et al. [23] investigated numerically two high-velocity impact problems in elastic–plastic materials: (1) Taylor copper-bar-impact problem and (2) the penetration of a long rod (tungsten heavy alloy) into a steel target. In this study, the CESE method combined with the level-set technique was adopted to solve Eulerian governing equations that describe the solid dynamics and to trace material interfaces. Chen et al. [24] extended the CESE solver developed by Wang et al. [23] to simulate high-velocity impact problems involving elastic–plastic flows, high strain rates, and spall fractures. Excellent agreement was observed between their simulation of aluminium plates colliding with stainless-steel plates and the experimental data.

As known, the governing equations of elastic waves in solids can be properly written as a set of fully coupled first-order hyperbolic PDEs, including the conservation laws of mass and momentum as well as the rate-type constitutive relations for materials, by treating the density, velocity, and stress components as primitive unknowns. Therefore, the CESE method, which is suitable for hyperbolic systems, can be intuitively applied to simulate linear and nonlinear waves in elastic solids. The CESE simulations of resonant standing waves arising from a time-harmonic external axial load and compression waves arising from a bi-material collinear impact was demonstrated by Yu et al. [25]. Another study on the propagation and reflection of extensional waves in an abruptly stopped elastic rod using the CESE method was carried out by Yang et al. [26]. Moreover, Chen et al. [27] performed CESE simulations to study planar-wave expansion from a point source in an anisotropic solid of cubic symmetry (gallium arsenide) and Yang et al. [28] extended the CESE simulations to stress waves in solids of hexagonal symmetry, illustrating wave propagation in a heterogeneous solid composed of three blocks of beryl with different lattice orientations. Lately, Lowe et al. [29] conducted a comprehensive study of nonlinear longitudinal waves (including both weak and strong shocks, rarefactions, and contact discontinuities) in tapered elastic rods using the CESE method as a numerical tool. In the above studies, the CESE simulations effectively captured waves in a variety of solid materials and provided results consistent with the available analytical solutions.

9.4   Magnetohydrodynamics

The plasma flows in aerospace applications and astrophysics can be described and solved generally by the magnetohydrodynamic (MHD) equations, which combine the NS equations for fluid dynamics and the Maxwell equations for electromagnetics. Due to the divergence-free constraint usually imposed for the magnetic field $\boldsymbol{B}$ in the computational MHD problems, some special treatments are added into the MHD numerical schemes to enforce $\nabla \cdot \boldsymbol{B} = 0$. Since the proposal of the CESE method, researchers have been attempting to solve MHD problems accurately and efficiently with the CESE method.

Zhang et al. [30, 31] used the CESE method to study MHD benchmark problems, including an MHD shock-tube problem and an MHD vortex problem. Their CESE results without additional treatments for $\nabla \cdot \boldsymbol{B} = 0$ compared favourably with previously reported reference solutions. In fact, Zhang et al. [30, 31] performed the simulations with the baseline CESE scheme and the CESE scheme in conjunction with a special treatment to maintain $\nabla \cdot \boldsymbol{B} = 0$. Nevertheless, no obvious difference was observed.

Feng et al. [32] developed a numerical platform based on the CESE method to investigate solar–interplanetary physics and space weather. Three dimensional MHD equations were solved with the adaptive mesh refinement (AMR) to better resolve flow features that have spatial scales many orders of magnitude smaller than the vast size of solar–interplanetary space. This CESE-based MHD-simulation approach was validated through the numerical study of solar corona and solar wind and comparison with observation data. Recently, Yang et al. [33, 34] extended this CESE-based MHD solver to a high-order version [33] and an upwind version [34] based on the CESE modifications of Shen et al. [35] and Shen and Wen [36], respectively. Numerical tests of MHD-vortex and MHD-blast-wave problems demonstrated the accuracy improvement of the CESE results by applying both the high-order and the upwind extension. Furthermore, a new strategy to keep the magnetic field fundamentally divergence-free was proposed by Yang et al. [33], taking the advantage of the CESE method that the spatial derivatives of the magnetic field are treated as marching variables in the algorithm.

Notably, the successful combination of MHD modelling and CESE simulation was elaborated in a recent monograph of Feng [37]. For more details about the CESE method for MHD, readers are recommended to refer to Feng [37] and the references therein.

References

1. Fedorov, A., & Tumin, A. (2004). Evolution of disturbances in entropy layer on blunted plate in supersonic flow. *AIAA Journal, 42*(1), 89–94.
2. Tseng, T., & Yang, R. J. (2005). Simulation of the Mach reflection in supersonic flows by the CE/SE method. *Shock Waves, 14*(4), 307–311.

3. Chang, I. S., Chang, C. L., & Chang, S. C. (2005). Unsteady Navier-Stokes rocket nozzle flows. In *41st AIAA/ASME/SAE/ASEE Joint Propulsion Conference and Exhibit*.

4. Chang, C. L., Venkatachari, B. S., & Cheng, G. (2006). Effect of counterflow jet on a supersonic reentry capsule. In *42nd AIAA/ASME/SAE/ASEE Joint Propulsion Conference and Exhibit*.

5. Venkatachari, B. S., Ito, Y., Cheng, G., & Chang, C. L. (2011). Numerical investigation of the interaction of counterflowing jets and supersonic capsule flows. In *42nd AIAA Thermophysics Conference*.

6. Chang, C. L., & Choudhari, M. M. (2011). Hypersonic viscous flow over large roughness elements. *Theoretical and Computational Fluid Dynamics, 25*(1), 85–104.

7. Shen, H., Wen, C. Y., & Saldívar Massimi, H. (2014). Application of CE/SE method to study hypersonic non-equilibrium flows over spheres. In *19th AIAA International Space Planes and Hypersonic Systems and Technologies Conference*.

8. Massimi, H., Shen, H., & Wen, C. Y. (2017). Study of hypersonic dissociating flows over spheres using the space-time CE/SE method. In *30th International Symposium on Shock Waves* (Vol. 1). Springer.

9. Wen, C. Y., Saldívar, H., & Shen, H. (2018). Extension of CE/SE method to non-equilibrium dissociating flows. *Journal of Computational Physics, 356*, 240–260.

10. Knab, O., Fruehauf, H. H., & Messerschmid, E. (1995). Theory and validation of the physically consistent coupled vibration-chemistry-vibration model. *Journal of Thermophysics and Heat Transfer, 9*(2), 219–226.

11. Wen, C. Y., & Hornung, H. (1995). Non-equilibrium dissociating flow over spheres. *Journal of Fluid Mechanics, 299*, 389–405.

12. Massimi, H., Shen, H., & Wen, C. Y. (2015). Numerical simulation of ionized hypersonic flows using the space-time CE/SE method. In *20th AIAA International Space Planes and Hypersonic Systems and Technologies Conference*.

13. Candler, G. V., & MacCormack, R. W. (1991). Computation of weakly ionized hypersonic flows in thermochemical nonequilibrium. *Journal of Thermophysics and Heat Transfer, 5*(3), 266–273.

14. Wang, X. Y., Chang, S. C., & Jorgenson, P. (1999). Accuracy study of the space-time CE/SE method for computational aeroacoustics problems involving shock waves. In *38th Aerospace Sciences Meeting and Exhibit*.

15. Loh, C. Y., Hultgren, L. S., & Chang, S. C. (2001). Wave computation in compressible flow using space-time conservation element and solution element method. *AIAA journal, 39*(5), 794–801.

16. Chang, S. C., Loh, C., Yu, S. T., Himansu, A., Wang, X. Y., Jorgenson, P., Chang, S. C., Loh, C., Yu, S. T., & Himansu, A. (1997). Robust and simple non-reflecting boundary conditions for the space-time conservation element and solution element method. In *13th Computational Fluid Dynamics Conference*.

17. Yen, J., Duell, E., & Martindale, W. (2006). CAA using 3D CESE method with a simplified Courant number insensitive scheme. In *12th AIAA/CEAS Aeroacoustics Conference (27th AIAA Aeroacoustics Conference)*.

18. Yen, C. C. (2011). Demonstration of a multi-dimensional time-accurate local time stepping CESE method. In *17th AIAA/CEAS Aeroacoustics Conference (32nd AIAA Aeroacoustics Conference)*.

19. Loh, C. Y., & Zaman, K. (2002). Numerical investigation of transonic resonance with a convergent-divergent nozzle. *AIAA Journal, 40*(12), 2393–2401.

20. Zaman, K., Dahl, M., Bencic, T., & Loh, C. (2002). Investigation of a 'transonic resonance' with convergent–divergent nozzles. *Journal of Fluid Mechanics, 463*, 313–343.

21. Kim, C. K., Yu, S. T. J., & Zhang, Z. C. (2004). Cavity flows in a scramjet engine by the space-time conservation and solution element method. *AIAA Journal, 42*(5), 912–919.

22. Cheng, G. C., Olcmen, S., Venkatachari, B. S., Brooker, B. T., & Chang, S. C. (2018). Computational study of subsonic and supersonic acoustic cavity flows using CESE method. In *2018 AIAA/CEAS Aeroacoustics Conference*.

23. Wang, J. T., Liu, K. X., & Zhang, D. L. (2009). An improved CE/SE scheme for multi-material elastic–plastic flows and its applications. *Computers & Fluids, 38*(3), 544–551.

24. Chen, Q. Y., Wang, J. T., & Liu, K. X. (2010). Improved CE/SE scheme with particle level set method for numerical simulation of spall fracture due to high-velocity impact. *Journal of Computational Physics, 229*(19), 7503–7519.

25. Yu, S. T. J., Yang, L., Lowe, R. L., & Bechtel, S. E. (2010). Numerical simulation of linear and nonlinear waves in hypoelastic solids by the CESE method. *Wave Motion, 47*(3), 168–182.

26. Yang, L., Lowe, R. L., Yu, S. T. J., & Bechtel, S. E. (2010). Numerical solution by the CESE method of a first-order hyperbolic form of the equations of dynamic nonlinear elasticity. *Journal of Vibration and Acoustics, 132*(5).

27. Chen, Y. Y., Yang, L., & Yu, S. T. J. (2011). Simulations of waves in elastic solids of cubic symmetry by the conservation element and solution element method. *Wave Motion, 48*(1), 39–61.

28. Yang, L., Chen, Y. Y., & Yu, S. T. J. (2011). Velocity-stress equations for waves in solids of hexagonal symmetry solved by the space-time CESE method. *Journal of Vibration and Acoustics, 133*(2).

29. Lowe, R. L., Lin, P. H., Yu, S. T. J., & Bechtel, S. E. (2016). An Eulerian model for nonlinear waves in elastic rods, solved numerically by the CESE method. *International Journal of Solids and Structures, 94*, 179–195.

30. Zhang, M., Yu, S. T. J., Lin, S. C., Chang, S. C., & Blankson, I. (2004). Solving magneto-hydrodynamic equations without special treatment for divergence-free magnetic field. *AIAA Journal, 42*(12), 2605–2608.

31. Zhang, M., Yu, S. T. J., Lin, S. C. H., Chang, S. C., & Blankson, I. (2006). Solving the MHD equations by the space–time conservation element and solution element method. *Journal of Computational Physics, 214*(2), 599–617.

32. Feng, X. S., Xiang, C. Q., Zhong, D. K., Zhou, Y. F., Yang, L. P., & Ma, X. P. (2014). SIP-CESE MHD model of solar wind with adaptive mesh refinement of hexahedral meshes. *Computer Physics Communications, 185*(7), 1965–1980.

33. Yang, Y., Feng, X. S., & Jiang, C. W. (2017). A high-order CESE scheme with a new divergence-free method for MHD numerical simulation. *Journal of Computational Physics, 349*, 561–581.

34. Yang, Y., Feng, X. S., & Jiang, C. W. (2018). An upwind CESE scheme for 2D and 3D MHD numerical simulation in general curvilinear coordinates. *Journal of Computational Physics, 371*, 850–869.

35. Shen, H., Wen, C. Y., Liu, K. X., & Zhang, D. L. (2015). Robust high-order space–time conservative schemes for solving conservation laws on hybrid meshes. *Journal of Computational Physics, 281*, 375–402.

36. Shen, H., & Wen, C. Y. (2016). A characteristic space–time conservation element and solution element method for conservation laws II. Multidimensional extension. *Journal of Computational Physics, 305*, 775–792.

37. Feng, X. S. (2019). *Magnetohydrodynamic modeling of the solar corona and heliosphere.* Springer.

Chapter 10
Summary

In this book, the space–time CESE method is introduced as a novel and distinctive numerical approach to solving conservation equations in engineering sciences. Also, from a historical perspective, the development of the CESE method since the 1990s has been reviewed and the remarkable improvements and extensions of the CESE method in the past few years has been summarized. Various applications of the CESE method have been presented to emphasize its numerical performance under different scenarios, including many research topics in engineering, such as compressible multi-fluid flows and detonations.

The most elegant part of the CESE method might be the non-dissipative a scheme described in Chap. 2, where the unknown variable u and its spatial derivative u_x can be obtained directly from the conservation laws. However, for the purpose of shock capturing, two families of CESE schemes have been developed on the basis of a scheme. The first family is called the central CESE scheme, including the a–α and CNI schemes. In these schemes, the spatial derivatives are updated by specially designed procedures, where artificial dissipation can be added. The central CESE schemes avoid any characteristic-based techniques (i.e., they are free of Riemann solver). The second family is the upwind CESE schemes, in which the upwind flux solver (e.g., Riemann solver) can be utilized, but it just plays a supporting role. By combining the ideas in the original CESE method and the conventional upwind FVM, the upwind CESE scheme managed to retain the forms of discretized equations for u and u_x in the a scheme, while the numerical dissipation can be introduced in a reasonable manner. It is worth noting that the dissipation of the upwind CESE scheme is naturally insensitive to the CFL number.

Two important issues about the CESE method, namely the CESE schemes on unstructured meshes and the high-order CESE schemes, have been discussed in Chaps. 4 and 5, respectively. The CESE method is naturally multi-dimensional and compatible with unstructured meshes. A high-order extension of the CE/SE method in both space and time can be achieved by storing and updating high-order derivatives

(e.g., u_{xx} and u_{xxx}) at each solution point, without enlarging the stencil or adding stages in time integration.

The CESE method has low dissipation and high compactness. When applied to the simulations of complex physical processes, the CESE method can catch shock waves, contact discontinuities, fine structures, and small disturbances, with high resolution and strong robustness. Therefore, the CESE method demonstrates good performances in the numerical simulations of wave-propagation problems (e.g., shock waves, acoustic waves, detonation waves, stress waves in solid, and the electromagnetic waves), interfacial instabilities, and shock–bubble/droplet interactions. In many hot research areas including hypersonic aerodynamics, shock dynamics, detonation, aeroacoustics, MHD, and solid dynamics, the CESE method proves to be suitable and shows a good development prospect.

In recent years, an important topic in CESE research has been its application to highly nonuniform and high-aspect-ratio computational meshes, which are often required in simulations of boundary layers. In particular, the CESE method has been recently applied to aerodynamic heating problems and direct numerical simulation of laminar or turbulent flows. It is also worth noting that the CESE schemes are mainly applied to unsteady problems due to their explicit nature in time with the associated restriction on time step size. In order to solve steady problems efficiently, the extensions to implicit time-stepping CESE schemes are under developing. Further analyses of the numerical properties of CESE schemes will be very useful to solidify their mathematical foundation. The modified equation analysis and the spectral property analysis for the upwind CESE scheme are in progress now.

Along with the continuous improvement of high-performance computing, numerical methods emerge one after another. Many challenging numerical simulations can be accomplished today. However, none of the existing numerical methods can be accurate, robust, and efficient under all situations. In this sense, the CESE method may not be superior to the existing methods, but it does provide an alternative approach which deserves more evaluations and further investigations. The authors wish this book can familiarize the readers with the CESE method.

Appendix

1. The *a-α* CESE solver for solving the 1D scalar convection problem.

```c
#include <stdio.h>
#include <math.h>
#include <stdlib.h>
#include <time.h>
#define end_time 2.0            // Computing time.
#define L1-1.0      // Left end of the domain.
#define L2 1.0              // Right end of the domain.
#define N 200            // Mesh number.
#define Sf 0.8          // CFL number.
#define MAX(x,y) (((x)>(y))?(x):(y))
#define MIN(x,y) (((x)<(y))?(x):(y))
double U1[N+3],U2[N+3],Ux1[N+3],Ux2[N+3],Ut[N+3],F[N+3],Ft[N+3];
double const dx=(double(L2-L1))/double(N);      //Mesh size

double avg(double x1,double x2)
{
      double x,aa;
      aa=2.0;//6*fabs(x1-x2)/(fabs(x1)+fabs(x2)+1e-10);
      x=(pow(fabs(x1),aa)*x2+pow(fabs(x2),aa)*x1)/
            (pow(fabs(x1),aa)+pow(fabs(x2),aa)+1e-20);
      return x;
}// Compute weighted average of x1 and x2.

double CFL(double U[N+3])
{
      int i;
      double maxvel,vel;
      maxvel=1e-10;
      for(i=1;i<=N+1;i++)
      {
            vel=1.0;
            if (vel>maxvel)maxvel=vel;
      }
      return Sf*dx/maxvel;
}// Compute timestep.
```

```
void Bound (double U[N+3],double Ux[N+3])
{
      U[0]=U[N];
      U[N+2]=U[2];
      Ux[0]=Ux[N];
      Ux[N+2]=Ux[2];
}// Apply boundary condition.

void U2Fu(double U[N+3],double Fu[N+3])
{
      int i;
      for(i=0;i<N+3;i++)
      {
            Fu[i]=U[i];
      }
}// Compute flux.

void ComputeTemporalDerivatives(double U[N+3],double Ux[N+3])
{
      int i;
      for(i=0;i<=N+2;i++)
      {
            Ut[i]=-Ux[i];
            Ft[i]=Ut[i];
      }
}// Compute the temporal derivatives of U and F.

void    CESE_TimeMarch(double    U[N+3],double    Ux[N+3],double
Fu[N+3],double
Fut[N+3],double Unew[N+3],double Uxnew[N+3],double dt,int ishalf)
{
      int i,I;
      double Uxplus,Uxminus,Uleft,Uright,Fleft,Fright;
      for(i=1-ishalf;i<=N+1;i++)
      {
            I=i+ishalf;

            Uleft=U[I-1]+dx*Ux[I-1]/4;
            Uright=U[I]-dx*Ux[I]/4;
            Fleft=Fu[I-1]+dt*Fut[I-1]/4;
            Fright=Fu[I]+dt*Fut[I]/4;

            Unew[i]=0.5*(Uleft+Uright+dt/dx*(Fleft-Fright));
            Uxminus=2*(Unew[i]-U[I-1]-dt*Ut[I-1]/2)/dx;
            Uxplus=2*(U[I]+dt*Ut[I]/2-Unew[i])/dx;
            Uxnew[i]=avg(Uxminus,Uxplus);
      }
}// Scheme marching from integer-step to half-step, or from half-
step to integer-step.

void Initialize(double U[N+3],double Ux[N+3])
{
      int i;
      for(i=0;i<=N+2;i++)
      {
```

```
                if(i>=(N/4+1)&&i<=(3*N/4+1))
                {
                        U[i]=1;
                }
                else
                {
                        U[i]=0;
                }
                Ux[i]=0;
        }
}// Initialization.

void Output(double U[N+3],double t)
{
        int i;
        double x;
        FILE *fp;
        fp=fopen("result.txt","w+");
        for(i=1;i<=N+1;i++)
        {
                x=L1+(i-1)*dx;
                fprintf(fp,"%20.5e\t%20.5e\n",x,U[i]);
        }
        fclose(fp);
}// Output result.

void CESE_Solver()
{
        int Nstep=0;
        double dt,t=0;
        Initialize(U1,Ux1);
        while(t<end_time &&Nstep>=0)
        {
                dt=CFL(U1);
                t=t+dt;
                Nstep++;
                printf("Nstep=%d,dt=%e,t=%e\n",Nstep,dt,t);
                U2Fu(U1,F);
                ComputeTemporalDerivatives(U1,Ux1);
            CESE_TimeMarch(U1,Ux1,F,Ft,U2,Ux2,dt,1);//the 1st half
time step
                U2Fu(U2,F);
                ComputeTemporalDerivatives(U2,Ux2);
            CESE_TimeMarch(U2,Ux2,F,Ft,U1,Ux1,dt,0);//the 2nd half
time step
                Bound(U1,Ux1);
        }
        Output(U1,t);
}// a-α CESE solver.

int main(void)
{
        CESE_Solver();
        return 1;
}
```

2. The *a-α* CESE solver for solving the Sod-shock problem.

```
#include <stdio.h>
#include <math.h>
#include <stdlib.h>
#include <time.h>
#define end_time 0.1              // Computing time.
#define L1 0.0                    // Left end of the domain.
#define L2 2.0                    // Right end of the domain.
#define L0 1.0
#define N 200                     // Mesh number.
#define Sf 0.8                    // CFL number.
#define GAMA 1.4             // Ratio of specific heats.
#define rou1 1.0
#define u1 0.0
#define p1 1.0
#define rou2 0.125
#define u2 0.0
#define p2 0.1                            // Initial conditions
#define MAX(x,y) (((x) > (y))?(x):(y))
#define MIN(x,y) (((x)<(y))?(x):(y))
double U1[N+1][3],U2[N+1][3],Ut[N+1][3],F[N+1][3],Ux1[N+1][3],
Ux2[N+1][3], Ft[N+1][3];
double const dx=(double(L2-L1))/double(N);        // Mesh size.

double CFL(double U[N+1][3])
{
      int i;
      double maxvel=1e-4,vel,p,u;
      for(i=1;i<=N-1;i++)
      {
            u=U[i][1]/U[i][0];
            p=(GAMA-1)*(U[i][2]-0.5*U[i][0]*u*u);
            vel=sqrt(GAMA*p/U[i][0])+fabs(u);
            if (vel > maxvel)maxvel=vel;
      }
      return Sf*dx/maxvel;
}// Compute timestep.

double avg(double x1,double x2)
{
      double x,aa=1.0;
      x=(pow(fabs(x1),aa)*x2+pow(fabs(x2),aa)*x1)/
            (pow(fabs(x1),aa)+pow(fabs(×2),aa)+1e-20);
      return x;
}// Compute weighted average of x1 and x2.

void Init()
{
      int i,k;
      double x;
      for(i=0;i<=N;i++)
      {
            x=L1+i*dx;
            if(x<=L0)
```

```
                {
                        U1[I][0]=rou1;
                        U1[I][1]=rou1*u1;
                        U1[I][2]=p1/(GAMA-1)+rou1*u1*u1/2;
                }
                else
                {
                        U1[I][0]=rou2;
                        U1[I][1]=rou2*u2;
                        U1[I][2]=p2/(GAMA-1)+rou2*u2*u2/2;
                }
        }
        for(i=0;i<=N;i++)
        {
                for(k=0;k<3;k++)
                {
                        Ux1[i][k]=0;
                }
        }
}// Initialization.

void Bound (double U[N+1][3])
{
        for(int k=0;k<3;k++)
        {
                if(k==1)
                {
                        U[N][k]=0;
                        Ux1[N][k]=Ux1[N-1][k];
                        U[0][k]=0;
                        Ux1[0][k]=Ux1[1][k];
                }
                else
                {
                        U[N][k]=U[N-1][k];
                        Ux1[N][k]=0;
                        U[0][k]=U[1][k];
                        Ux1[0][k]=0;
                }
        }
}// Apply boundary condition.

void U2F(double U[N+1][3],double F[N+1][3])
{
        int i;
        double u,p;
        for(i=0;i<N+1;i++)
        {
                u=U[i][1]/U[i][0];
                p=(GAMA-1)*(U[i][2]-0.5*U[i][1]*U[i][1]/U[i][0]);
                F[i][0]=U[i][1];
                F[i][1]=U[i][0]*u*u+p;
                F[i][2]=(U[i][2]+p)*u;
        }
}// Compute flux.
```

```
void ComputeDerivative(double U[N+1][3],double Ux[N+1][3])
{
      int i;
      double u;
      for(i=0;i<=N;i++)
      {
          u=U[i][1]/U[i][0];
          Ut[i][0]=-Ux[i][1];
          Ut[i][1]=-(GAMA-3)*u*u*Ux[i][0]/2-(3-GAMA)*u*Ux[i][1]
-(GAMA-1)*Ux[i][2];
           Ut[i][2]=-((GAMA-1)*u*u*u-GAMA*u*U[i][2]/U[i][0])*
Ux[i][0]-(GAMA*U[i][2]/U[i][0]-3*(GAMA-1)/2*u*u)*Ux[i][1]-
GAMA*u*Ux[i][2];
           Ft[i][0]=Ut[i][1];
           Ft[i][1]=(GAMA-3)*u*u*Ut[i][0]/2+(3-GAMA)*u*Ut[i][1]
+(GAMA-1)*Ut[i][2];
           Ft[i][2]=((GAMA-1)*u*u*u-GAMA*u*U[i][2]/U[i][0])*
Ut[i][0]+(GAMA*U[i][2]/U[i][0]-3*(GAMA-1)/2*u*u)*Ut[i][1]
+GAMA*u*Ut[i][2];
      }
}// Compute the derivatives of F.

void  CESE_TimeMarching(double  U[N+1][3],  double  Ux[N+1][3],
double Fu[N+1][3],double Fut[N+1][3], double Unew[N+1][3], double
Uxnew[N+1][3], double dt, int ishalf)
{
      int i,j,I;
      double Uleft,Uright,Fleft,Fright,Ux_minus,Ux_plus;
      for(i=1-ishalf;i<N;i++)
      {
           I=i+ishalf;
           for(j=0;j<3;j++)
           {
                Uleft=U[I-1][j]+dx*Ux[I-1][j]/4;
                Uright=U[I][j]-dx*Ux[I][j]/4;
                Fleft=F[I-1][j]+dt*Ft[I-1][j]/4;
                Fright=F[I][j]+dt*Ft[I][j]/4;
                Unew[i][j]=0.5*(Uleft+Uright+dt/dx*(Fleft
-Fright));
                Ux_minus=2*(Unew[i][j]-U[I-1][j]-dt*
Ut[I-1][j]/2)/dx;
                Ux_plus=2*(U[I][j]+dt*Ut[I][j]/2-Unew[i][j])/dx;
                Uxnew[i][j]=avg(Ux_minus,Ux_plus);
           }
      }
}// Scheme marching from integer-step to half-step, or from half-
step to integer-step.

void CESE_1D_Solver()
{
      int i,comput_num;
      double dt,t=0;
      FILE *fp;
      comput_num=0;
      Init();
```

```
        while(t<end_time)
        {
                comput_num++;
                dt=CFL(U1);
                t=t+dt;
                U2f(U1,f);
                ComputeDerivative(U1,Ux1);
                CESE_TimeMarching(U1,Ux1,F,Ft,U2,Ux2,dt,1);
                U2f(U2,f);
                ComputeDerivative(U2,Ux2);
                CESE_TimeMarching(U2,Ux2,F,Ft,U1,Ux1,dt,0);
                Bound(U1);
        }
}// a-α CESE solver.

void Output(double U[N+1][3])
{
        int i;
        double x;
        FILE *fp;
        double rou,u,p;
        fp=fopen("result.txt","w+");
        for(i=0;i<N+1;i++)
        {
                x=L1+i*dx;
                rou=U[i][0];
                u=U[i][1]/U[i][0];
                p=(GAMA-1)*(U[i][2]-0.5*U[i][0]*u*u);
                fprintf(fp,"%20.5e\t%20.5e\t%20.5e\t%20.5e\n",
x,rou,u,p);
        }
        fclose(fp);
}// Output result.

int main()
{
        CESE_1D_Solver();
        Output(U1);
        return 1;
}
```