The Method of Space-Time Conservation Element and Solution Element -- A New Paradigm for Numerical Solution of Conservation Laws

Sin-Chung Chang [1], Sheng-Tao Yu [2], Ananda Himansu [3],
Xiao-Yen Wang [4], Chuen-Yen Chow [4] and Ching-Yuen Loh [3]

[1] NASA Lewis Research Center, Cleveland, OH 44135
[2] NYMA Technology, Inc., NASA LeRC, Cleveland, OH 44135
[3] Cleveland Telecomm. Corp., NASA LeRC, Cleveland, OH 44135
[4] University of Colorado, Boulder, CO 80309

The Method of Space-Time Conservation Element and Solution Element — A New Paradigm for Numerical Solution of Conservation Laws

Sin-Chung Chang*   Sheng-Tao Yu[†]   Ananda Himansu[‡]   Xiao-Yen Wang[§]

Chuen-Yen Chow[¶]   Ching-Yuen Loh[‖]

Abstract

In this paper, we review the basic principles of the method of space-time conservation element and solution element for solving the conservation laws in one and two spatial dimensions. The present method is developed on the basis of local and global flux conservation in a space-time domain, in which space and time are treated in a unified manner. In contrast to the modern upwind schemes, the approach here does not use the Riemann solver and the reconstruction procedure as the building blocks. Therefore, the logic and rationale are considerably simpler. The present approach has yielded high-resolution shocks, rarefaction waves, acoustic waves, vortices, ZND detonation waves and shock/acoustic waves/vortices interactions. Moreover, since no directional splitting is employed, numerical resolution of two-dimensional calculations is comparable to that of the one-dimensional calculations.

1   Introduction

Recently, Chang and coworkers [1–9] have reported a new numerical framework for solving conservation laws, namely, the Method of Space-Time Conservation Element and Solution Element, or the CE/SE method for short. This method is distinguished by the simplicity of its conceptual basis – flux conser-

vation in space and time. The method has been illustrated in the context of conservation laws in fluid dynamics. In describing the present method, the original reports took the most direct approach. That is, starting from the basic integral equations, the algebraic details and the mathematical analyses were presented in a systematic way, such that all information needed to implement a computer program was provided. That approach made it clear that the present method was developed from fundamentals. It is not an incremental improvement of a previously existing method.

The CE/SE method, however, can also be described in a more intuitive manner by focusing on its unique space-time discretization. The essence of the method may be grasped from a simple delineation of the inherent space-time geometry. With it, the rigorous algebraic details and mathematical proofs, though not superfluous, take on more of the flavor of confirming the intuitively obvious facts. In this review, the CE/SE method will be illustrated by the geometric approach. For mathematical details, the reader is referred to the original papers.

In addition, we wish to take a somewhat comparative approach in this review, to clarify the differences between the CE/SE method and traditional schemes. In particular, we wish to pique the interest of readers familiar with the modern upwind schemes. Toward this end, we shall place the present method into the context of the projection-upwinding-evolution approach advocated by van Leer [10,11].

In this introduction, to assist the reader in getting a flavor of the present method, particularly in its role as a tool to solve initial-value problems, several remarks which will be fully explained later are given here: (i) Although the differential form is also considered, the main emphasis of the present

*NASA Lewis Res. Center, Cleveland, OH 44135
[†]NYMA Technology, Inc., NASA Lewis Res. Center
[‡]Cleveland Telecomm. Corp., NASA Lewis Res. Center
[§]University of Colorado, Boulder, CO 80309
[¶]University of Colorado, Boulder, CO 80309
[‖]Cleveland Telecomm. Corp., NASA Lewis Res. Center

method is on solving the integral form of the transport equations in the space-time domain. (ii) The present method was designed to conserve space-time flux locally and globally. Conventional finite-volume schemes, however, concentrate on the spatial flux calculation; temporal evolution is usually treated by finite-differencing or semi-integration. The unified treatment of space and time in the present method cannot be overemphasized. (iii) For isentropic flows, the present method can be used to construct explicit solvers that are non-dissipative (neutrally stable) for all Courant numbers ≤ 1. Also, these solvers are two-way time-marching schemes, i.e., each forward marching scheme can be inverted to become a backward marching scheme. In other words, the marching variables at the $(n-1)$th time level can be determined in terms of those at the nth time level. (iv) A zigzagging marching strategy in the space-time domain is employed, such that flow information at each interface separating two conservation elements can be evaluated without interpolation or extrapolation. In particular, no Riemann solver is needed in calculating interfacial fluxes. (v) The flow solution structure is not calculated through a reconstruction procedure. Instead, the gradients of flow variables are treated as independent unknowns, and they are not influenced by the flow properties in neighboring elements at the same time level. This is in full compliance with the flow physics of the initial value problem. (vi) For flows in multiple spatial dimensions, no directional splitting is employed. The two and three-dimensional spatial meshes employed by the present method are built from triangles and tetrahedrons, respectively. Note that triangles and tetrahedrons, respectively, are also the simplest building blocks for two- and three-dimensional unstructured meshes.

At first glance, the special features of the CE/SE method, such as the staggered mesh and the treatment of flow gradient values as unknowns, may seem cumbersome. Closer examination, however, shows that they are requirements for a faithful discrete counterpart to the conservation laws. The resulting schemes are simple and yield accurate results. As a concrete illustration of these features, a FORTRAN program is listed in the appendix. It is a CE/SE solver for an extended Sod's shock tube problem [12], in which the shock tube problem is extended by imposing a non-reflecting boundary condition at each end of the computational domain.

The challenge of the non-reflective boundary condition is no less difficult than that of capturing the shock and the contact discontinuity. First, the flow under consideration is subsonic throughout and the treatment of the non-reflecting boundary condition for a subsonic flow is more difficult than that for a supersonic flow. Second, this difficulty is exacerbated by the existence of a shock and a contact discontinuity, which must be allowed to exit the domain without reflection. We remark that traditional non-reflective boundary conditions, such as the characteristics-based, the radiation (asymptotic), and the buffer-zone conditions, are based on the assumption that the flow is continuous. In spite of these difficulties, the present solver is capable of generating highly accurate non-reflecting solutions using a uniform time-step size from the beginning of time-marching (refer to Sec. 6 for a full discussion of the numerical result). The main loop in this CE/SE program contains only 39 FORTRAN statements. A single "if" is used to identify the time-levels at which to activate the boundary conditions. The program contains no other Fortran conditional statements such as "if", "amax", and "amin" that are used so often in programs of modern upwind methods.

Computer programs based on the CE/SE method have been developed for calculating flows in one and two spatial dimensions. Numerous results were obtained [13–22], including various shock tube problems, the ZND detonation waves, the implosion and explosion problem, shocks over a forward-facing step, acoustic waves, and shock/acoustic wave interactions. The method can clearly resolve shock/acoustic wave interactions wherein the difference of the magnitude between acoustic wave and shock could be up to six orders. In two-dimensional flows, the reflected shock is as crisp as the leading shock. From the evidence of these results, the CE/SE method has proved to be a promising numerical framework for solving fluid dynamics problems. In addition, Scott [23-27] has developed an implicit steady-state version of the space-time method for simulating steady-state incompressible flows, including viscous boundary layers and developing pipe flows.

The remainder of the paper is organized as follows. In Sec. 2, we present the general aspects of the space-time methods. A new space-time integral form of conservation laws will be described. In Sec. 3, a brief review at the conceptual level of the modern upwind schemes is presented. In Sec. 4, we present the CE/SE method for solving flow equations in one spatial dimension. In Sec. 5, the extensions of the CE/SE method in two and three spatial

dimensions, without directional-splitting, are illustrated. In Sec. 6, numerical examples calculated by the present method are shown. We then offer the concluding remarks.

2 The Space-Time Methods

The conventional finite volume methods for simulating conservation laws were formulated according to flux balance over *a fixed spatial domain*. As such, the conservation laws state that the rate of change of the total amount of a substance contained in a fixed spatial domain V is equal to the flux of that substance across the boundary of V. Let the density of the substance be u and its spatial flux be $\vec{f}$; then the convectional conservation law can be expressed as:

$$\frac{d}{dt}\int_V u\,dV = -\int_{S(V)} \vec{f}\cdot\vec{ds}, \qquad (2.1)$$

where $S(V)$ is the boundary of V, and $\vec{ds} = d\sigma\,\vec{n}$ with $d\sigma$ and $\vec{n}$, respectively, being the area and the outward unit normal vector of a surface element on $S(V)$. The finite volume methods concentrate on the evaluation of the right hand side of Eq. (2.1). The left hand side of Eq. (2.1) is usually discretized by a finite difference method, such as the Runge Kutta method.

Alternatively, an integration method for space-time flux balance could be employed:

$$\int_V u\,dV \Big|_{t_1}^{t_2} = -\int_{t_1}^{t_2} dt \int_{S(V)} \vec{f}\cdot\vec{ds}. \qquad (2.2)$$

This integration method, however, bears strong resemblance to the method of finite-differencing the temporal derivative term. Algebraically, one can always find its counterpart in the finite difference methods. Therefore, we refer to this method as semi-integration. Perhaps, the usefulness of this method lies in depicting a clearer picture of the space-time flux balance as compared to the methods of finite-differencing the time derivative term.

As shown in Fig. 2.1(a), due to the fixed spatial domain, the shape of the space-time Conservation Elements (CEs) in one spatial dimension must be rectangular. In addition, these elements must stack up exactly on top of each other in the time direction, i.e., no staggering of these elements in time is allowed. For equations in two spatial dimensions, as depicted in Fig. 2.1(b), a conservation element is

a uniform-cross-section cylinder in space-time, and again no staggering in time is allowed. This arrangement results in vertical interfaces extended in the direction of time evolution between adjacent CEs. Across these interfaces, flow information travels in both directions. Therefore, in calculating the interfacial flux, upwind biasing (or a Riemann solver) becomes necessary.

In the following subsections, we shall first state a non-conventional integral equation for conservation laws, in which space and time are treated in a unified manner. This space-time integral equation allows a space-time geometry of the CEs in the present CE/SE method which is different from that used in the conventional finite volume methods and shown in Fig. 2.1. As a result, the present CE/SE method is able to capture shocks without using a Riemann solver.

In Secs. 2.2 and 2.3, we shall discuss the non-dissipative property of the isentropic conservation laws. In Sec. 2.4, the breakdown of the non-dissipative property due to the entropy increasing processes will be discussed. We then conclude this section by posing several criteria for an ideal numerical analogue of the conservation laws. For convenience, the discussions in this section are based on one-dimensional flow equations.

2.1 Governing Equations

The one-dimensional unsteady Euler equations of a perfect gas can be expressed as

$$\mathbf{U}_t + \mathbf{F}_x = 0, \qquad (2.3)$$

where

$$\mathbf{U} = \begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix} = \begin{pmatrix} \rho \\ \rho v \\ \rho E \end{pmatrix}, \qquad (2.4)$$

$$\mathbf{F} = \begin{pmatrix} f_1 \\ f_2 \\ f_3 \end{pmatrix} = \begin{pmatrix} \rho v \\ \rho v^2 + p \\ (\rho E + p)v \end{pmatrix}, \qquad (2.5)$$

with ρ, v, p, and E being the density, velocity, static pressure, and specific total energy, respectively. By definition, $E = v^2/2 + e$, where e is the specific internal energy. The equation of state is $p = (\gamma - 1)\rho e$ with γ being the specific heat ratio.

As shown in Fig. 2.2, let $x_1 = x$, and $x_2 = t$ be the coordinates of a two-dimensional Euclidean

3

space E_2. Then Eq. (2.3) can be expressed as the divergence-free conditions

$$\bigtriangledown \cdot \vec{h}_m = 0, \qquad m = 1, 2, 3, \qquad (2.6)$$

where $\vec{h}_m = (f_m, u_m)$, $m = 1, 2, 3$, are the *space-time* mass, momentum, and energy current-density vectors, respectively. Equation (2.6) is valid everywhere in E_2 for continuous and isentropic flow solutions. For flows with shock waves, we must use the more fundamental form of the conservation laws:

$$\oint_{S(R)} \vec{h}_m \cdot \vec{ds} = 0, \qquad m = 1, 2, 3. \qquad (2.7)$$

Here $S(R)$ is the boundary of a space-time region R, and $\vec{ds} = d\sigma\,\vec{n}$ with $d\sigma$ and $\vec{n}$, respectively, being the area and the outward unit normal vector of a surface element on $S(R)$. Note that (i) because $\vec{h}_m \cdot \vec{ds}$ is the space-time flux of $\vec{h}_m$ leaving R through $\vec{ds}$, Eq. (2.7) simply states that the total space-time flux of $\vec{h}_m$ leaving R through its boundary vanishes; (ii) all mathematical operations can be carried out as though E_2 were an ordinary two-dimensional Euclidean space; (iii) because $S(R)$ is a simple closed curve in E_2, the "surface" integration in Eq. (2.7) can be converted into a line integration [6], i.e.,

$$\oint_{S(R)} \vec{h}_m \cdot \vec{ds} = \oint_{S(R)}^{c.c.} (f_m\,dt - u_m\,dx) = 0 \qquad (2.8)$$

where the notation *c.c.* indicates that the line integration is carried out in the counterclockwise direction; and (iv) Eq. (2.6) is valid only for smooth flows, and it can be derived from Eq. (2.7) using Gauss' divergence theorem.

Here, we offer two remarks about Eq. (2.7): (i) Unlike Eqs. (2.1) and (2.2), the present formulation does not impose any constraint on the shape of the CEs in the space-time domain. This is a crucial difference that, at the conceptual level, separates the present method from the conventional finite volume methods. (ii) Equation (2.7) must be satisfied over any bounded sub-domain R of E_2. As a result, if the total space-time fluxes leaving R through a subset of the boundary $S(R)$ are known, the total space-time fluxes leaving through the rest of $S(R)$ can be deduced. Perhaps, a marching scheme can be built by a careful selection of the space-time geometry of the CEs. In the following discussions, we offer some thoughts about this envisioned marching scheme.

2.2 Marching Forward and Backward

A time-marching scheme for solving the Euler equations is valid because the Jacobian matrix $\mathbf{A} = \partial \mathbf{F} / \partial \mathbf{U}$ has the real eigenvalues v and $v \pm c$, where c is the sonic speed. Because $\mathbf{F}_x = \mathbf{A}\mathbf{U}_x$, Eq. (2.3) can be diagonalized by multiplying it from the left by the left eigenvector matrix $\mathbf{M}$ of $\mathbf{A}$, i.e.,

$$\mathbf{M}\mathbf{U}_t + \mathbf{M}\mathbf{A}\mathbf{M}^{-1}\mathbf{M}\mathbf{U}_x = 0, \qquad (2.9)$$

where $\mathbf{M}^{-1}$ is the inverse of $\mathbf{M}$. Let $\partial \hat{\mathbf{U}} = \mathbf{M}\partial \mathbf{U}$ be the characteristic variable vector. Then we have

$$\hat{\mathbf{U}}_t + \mathbf{\Lambda}\hat{\mathbf{U}}_x = 0, \qquad (2.10)$$

where $\mathbf{\Lambda} = \mathbf{M}\mathbf{A}\mathbf{M}^{-1}$ is a diagonal matrix composed of the eigenvalues a_m ($m = 1, 2, 3$) of $\mathbf{A}$. Eq. (2.10) is the classical system of characteristic compatibility equations, from which one can derive the Riemann invariants along the characteristics.

From the solution of the Method Of Characteristics (MOC), one knows that the flow solution has a finite domain of dependence at a previous time and is completely determined by it. One can calculate the flow solution along the characteristics for the whole space-time domain if the initial conditions are specified at every point in space. Moreover, because the Riemann invariant is constant along any characteristic without regard to the marching direction, given the flow solution at a certain time, one can *trace backward* along the characteristics to obtain the solution in the past. Thus, an ideal numerical analogue of the Euler equations should be able to march forward to obtain the flow solution at a new time level, and from there march backward to recover the solution at the starting time.

Moreover, because the marching procedure can be initiated from any initial-data curve in the space-time domain (as long as it nowhere has a characteristic direction), the nature of the forward-and-backward marching of the Euler equations is not limited to the time evolution. Therefore, an ideal numerical analogue of the isentropic convection equations should possess no preferred marching direction in the E_2 space, except insofar as grid-lines are selected.

For flows in multiple spatial dimensions, one can also derive the divergence-free condition, similar to Eq. (2.6), and its integral form similar to Eq. (2.7), as the governing equations. Although the equations cannot be diagonalized as in the one-

4

dimensional case, the equation set is nonetheless hyperbolic. Therefore, the abovementioned forward-and-backward marching nature is still valid.

2.3 Space-Time Inversion

Another important property of Eq. (2.3) is its invariance under space-time inversion. Let (x_o, t_o) be a fixed point in E_2, and let $x' = 2x_o - x$ and $t' = 2t_o - t$. Note that (x', t') is the image of (x, t) and vice-versa under the space-time inversion with respect to (x_o, t_o). By the coordinate transformation, Eq. (2.3) is equivalent to

$$\mathbf{U}_{t'} + \mathbf{F}_{x'} = 0. \tag{2.11}$$

Let $\mathbf{U}^o(x, t)$ be a smooth function of x and t, and let $\mathbf{U} = \mathbf{U}^o$ satisfy Eq. (2.3). Then the above invariance property of Eq. (2.3) and the fact that $\mathbf{F}$ is function of $\mathbf{U}$ imply that $\mathbf{U} = \mathbf{U}^o(2x_o - x, 2t_o - t)$ is also a solution to Eq. (2.3). This property of the smooth solutions to Eq. (2.3) should be shared by an ideal numerical analogue of Eq. (2.3).

We remark that a numerical analogue that is stable and also preserves the above invariance property under space-time inversion must be non-dissipative, or neutrally stable. This non-dissipative nature of a numerical analogue of the convection equations reflects the fact that a smooth solution to Eq. (2.3) does not dissipate with time. For the one and two-dimensional scalar convection equations, this conclusion is established in [2, 5].

For flow equations in multiple spatial dimensions, the invariance property under space-time inversion is still valid. An ideal numerical analogue should preserve this property, and it should be neutrally stable.

2.4 The Shock Capturing Scheme

Two issues need to be addressed by a shock capturing scheme : (i) the flow properties are not continuous across the shock, and (ii) the flow entropy increases across the shock.

To address the first issue, an integral, rather than differential, form of the governing equations must be employed. Based on the conservation of mass, momentum, and energy, the flow properties on one side of a shock are determined by the flow properties on the other side. Hence, flux conservation should be the only principle used to relate the two distinct flow states. Other methods, based on the differential equations such as the MOC, should not be considered.

The second issue pertains to the fact that the evolution of a flow with shocks is an entropy increasing process, and it is irreversible. However, there is no constraint built into the Euler equations for flow solutions to obey the second law of thermodynamics. Therefore, an additional constraint must be imposed when flows with shocks are modeled using the Euler equations. In computational fluid dynamics, this additional constraint can be conveniently implemented by numerical treatment. For example, in a finite-differencing setting, Lax [28] showed that the discretized Euler equations must satisfy an entropy condition in order to capture shocks successfully. Essentially, an even-order artificial damping should be added to the discretized equation such that spurious oscillations near the shock can be suppressed and a physically sensible solution can be obtained.

In Sec. 4, we shall present the method of adding the artificial damping in the present scheme for capturing shocks. We shall show that within any one time step, the added artificial damping only influences the calculation of the flow gradient; the calculation of the flow properties is not changed. In addition, a specific parameter is employed to allow direct control of the amount of the artificial damping. When the simulations of isentropic flows are of interest, the artificial damping can be turned off.

2.5 An Ideal Numerical Analogue

From the above discussions, it is seen that a smooth solution to the Euler equations, Eq. (2.3), has the following important properties: (i) it does not dissipate with time; (ii) its value at any point (x, t) has a finite domain of dependence at an earlier time; and (iii) it is completely determined by the initial data at a given time. In the light of these properties, we remark that (i) a solution to a dissipative numerical scheme will dissipate with time; (ii) the value of a solution to an implicit scheme at any point (x, t) depends on all the initial data and all the boundary data up to the time t; and (iii) a scheme involving more than two time levels requires the specification of the initial data at more than one time level. Therefore, we conclude that an ideal numerical analogue to Eq. (2.3), in addition to having the

properties discussed in Secs. 2.2 and 2.3, should be *neutrally stable, explicit*, and involving only *two time levels*.

By adding an artificial dissipation term, an ideal solver of Eq. (2.3) can be extended to model flows with shocks. We want to emphasize that the artificial dissipation in an ideal numerical method should occur only in shock capturing; without added artificial damping, there should be no other source of numerical dissipation.

Furthermore, in an ideal Navier Stokes solver, the above guidelines of modeling the Euler equations should be applied to the discretization of the convective terms of the Navier Stokes equations. We note that, stripped of any added artificial terms, traditional numerical schemes are in general not free from inherent numerical dissipation. For flows at large Reynolds numbers, numerical dissipation may overwhelm the physical dissipation and cause a complete distortion of the solution. Because an ideal analogue of Eq. (2.3) has no numerical dissipation, when it is applied to discretize the convective terms of the Navier Stokes equations, the Navier-Stokes solver has the property that the numerical dissipation of its solutions approaches zero as the physical dissipation approaches zero. æ

3 Modern Upwind Schemes in the Space-Time Domain

In this section, we consider the modern upwind schemes, namely those developed based on the Godunov scheme. In this context, we want to clarify the differences between the CE/SE method and the conventional shock-capturing schemes. The structure of the modern upwind schemes will be described using van Leer's projection-upwinding-evolution formulation. In this regard, Huynh [29] has given a particularly lucid description. In general, the upwind schemes consists of three steps: (i) a projection or reconstruction step, in which the flow property distribution within each cell is approximated by polynomial curve fitting; (ii) an upwind step involving the solution of a Riemann problem to calculate the spatial fluxes at cell interfaces; and (iii) the temporal evolution step, in which the flow properties at the next time step are determined by either finite-differencing or by space-time flux conservation. Note that a space-time splitting form of the conservation laws, i.e., either Eq. (2.1) or Eq. (2.2), is employed

in the upwind schemes.

The original Godunov scheme employs the simplest reconstruction – the piecewise constant function, the most computationally expensive upwind scheme – the exact Riemann solver, and a simple time-marching method – an averaging procedure. Since the Godunov method is only first-order accurate, all modern upwind schemes are endeavors to improve the efficiency and accuracy of the method. To this end, modern upwind schemes utilize higher order reconstruction methods, more efficient Riemann solvers, and various time marching schemes. The basic structure of the modern upwind approach, however, has not changed.

3.1 The Evolution Step

In the present paper, a second-order upwind scheme presented by Huynh [29] is used to represent this class of schemes. As shown in Fig. 3.1, the E_2 space is divided into rectangular CEs. In each CE, the known column matrix of flow properties U_j^n is located at the center of the bottom boundary, and the unknown to be solved for, U_j^{n+1}, is at the center of the top boundary. The space-time flux conservation over the CE using Eq. (2.2) can be expressed as

$$U_j^{n+1}\Delta x - U_j^n\Delta x$$
$$+F_{j+1/2}^{n+1/2}\Delta t - F_{j-1/2}^{n+1/2}\Delta t = 0. \qquad (3.1)$$

Since a second-order scheme is used, a linear distribution of U and F along the respective boundary segments is assumed. Therefore, as shown in Eq. (3.1), the integration of the flow properties along the respective boundary segment is equal to the values at the midpoint multiplied by the length of the line segment. In Eq. (3.1), the value of U_j^n is known, and we need to evaluate $F_{j-1/2}^{n+1/2}$ and $F_{j+1/2}^{n+1/2}$ to determine U_j^{n+1}. This equation serves to track the temporal evolution of the solution, and thus forms the evolution step in van Leer's formulation. However, in order to calculate $F_{j-1/2}^{n+1/2}$ and $F_{j+1/2}^{n+1/2}$, the use of Eq. (3.1) must be preceded by a reconstruction step and an upwind step.

Temporal evolution schemes other than Eq. (3.1) could be used, in which, however, the picture of the space-time flux conservation is less clear. For instance, the Runge-Kutta (RK) method could be used for time marching. In each intermediate RK step, a reconstruction step and an upwind step are

6

performed to calculate the spatial fluxes. The unbalanced spatial flux is then used as the inhomogeneous term in the RK method. After several intermediate steps of the RK method, the flow solution at the new time level is obtained.

In the space-time setting of Eq. (3.1), a possibility of solution discontinuity must be allowed between U_j^{n+1} and U_j^n. Its exact location, however, is unclear. In Huynh's approach, the solution is assumed to be continuous within the lower half of each CE. As such, if $(U_x)_j^n$ is known, $(U_t)_j^n$ can be obtained by assuming

$$(F_x)_j^n = A_j^n (U_x)_j^n \qquad (3.2)$$

and

$$(U_t)_j^n = - (F_x)_j^n . \qquad (3.3)$$

Here (i) A_j^n is the matrix A (which is a function of U) evaluated with $U = U_j^n$, and (ii) Eqs. (3.2) and (3.3) are the numerical analogues of the relation $F_x = AU_x$ and the differential equation, Eq. (2.3), respectively. It follows from Eqs. (3.2) and (3.3) that one can use the second-order Taylor polynomial to calculate $U_{j\pm1/2}^{n+1/2}$:

$$U_{j\pm1/2}^{n+1/2} = U_j^n + (U_t)_j^n \frac{\Delta t}{2} \pm (U_x)_j^n \frac{\Delta x}{2} \qquad (3.4)$$

if the value of $(U_x)_j^n$ is known. This leads to the reconstruction step in the upwind schemes.

3.2   The Reconstruction Step

As a common practice in upwind schemes, flow properties at neighboring nodes are employed for a polynomial curve-fit to calculate $(U_x)_j^n$. The difficulty with such a procedure has always lain in fitting a polynomial curve across a possible solution discontinuity, which results in spurious oscillations. To suppress the spurious overshoot of $(U_x)_j^n$, scheme developers introduced *predetermined constraints* to the flow distribution, based on the expected nature of the solution. Van Leer [10, 11] suggested that the reconstruction procedure must preserve the monotonicity of the flow property distribution. An extension of this idea is the Total Variational Diminishing (TVD) method [30]. Another approach is the Essentially Non-Oscillatory (ENO) method [31] based on a strategy of stencil nodal selection. In general, these complex procedures are combinations of limiting the slopes of the flow properties and sharpening the jump conditions whenever a shock is detected.

Usually, these methods are effective in suppressing oscillations near shocks.

The drawback of these complex procedures is that the special properties, such as monotonicity, are not universal, and they are not prescribed by the conservation laws. For example, due to the existence of a source term, the flow field of a detonation wave is not monotonic nor does it display a TVD property. In this case, one must resort to other ideas to constrain the slopes.

3.3   The Upwind Step

Once $(U_x)_j^n$ is determined, the flow property distribution inside a CE can be expressed by the Taylor polynomial, Eq. (3.4). As such, flow properties at the immediate two sides of a cell interface, $\left(U_{j+1/2}^{n+1/2}\right)_R$ and $\left(U_{j+1/2}^{n+1/2}\right)_L$, can be calculated. Refer to Fig. 3.2.

Note that $\left(U_{j+1/2}^{n+1/2}\right)_R$ and $\left(U_{j+1/2}^{n+1/2}\right)_L$ are distinct states, and one must reconcile the two states to yield a unique interfacial flux for the temporal evolution calculation described by Eq. (3.1). The simplest approach is to average the two states. This approach, however, leads to central differencing the spatial flux terms in Eq. (3.1), and results in numerical oscillations for flows with shocks.

In the Godunov scheme, flow properties in each cell at $t = t^n$ are assumed constant. By using U_j^n and U_{j+1}^n as the initial values, one can set up a Riemann (shock tube) problem at the interface of two adjacent CEs. With the aid of the known solution (a function of x/t) to this classical problem, the flux passing through the interface can be obtained by an integration over the time interval $t^n \le t \le t^{n+1}$ at $x = x_{j+1/2}$.

In the second-order modern upwind schemes, a distribution of the flow property inside each CE exists. Thus, inside a CE, the characteristics associated with the varying flow properties interact with one another. As such, the flow physics at the interface is much more complex than that of the Godunov scheme. In this case, there is no known analytical solution. As a result, it becomes a formidable task to calculate the evolution of the interfacial flux. When faced with this difficulty, the usual approach is to consider only the left and right interface midpoint values, $\left(U_{j+1/2}^{n+1/2}\right)_L$ and $\left(U_{j+1/2}^{n+1/2}\right)_R$, and use

them to yield a unique value for the flux at that same point in the space-time domain. The calculation is usually based on the combination of left and right fluxes with upwind biasing, and the procedure is usually termed as an approximate Riemann solution. The idea of tracking the temporal evolution of flow solution at the interface, such as is used in the Godunov scheme, is discarded.

What is actually involved in practice is the use of the characteristic-based techniques. In the classical MOC, to march one step forward along a characteristic, one has to assume that the characteristic can be approximated locally as a straight line segment with its slope (an eigenvalue of A) either explicitly determined by the initial flow states (a first-order linearization), or iteratively determined by the combination of the initial flow state and the final flow state (a second-order linearization).

In the approximate Riemann solver, a similar procedure is performed, in which the coefficient matrix A and its eigenvalues are evaluated based on a specially chosen "averaged flow state." Following this, the flux calculation can be performed based on the characteristic-value splitting. In the modern upwind schemes, the efficiency and accuracy of the approximate Riemann solver hinge on a careful choice of this averaged flow state. As an example, we refer the reader to [29] for a detailed discussion of the linearization procedure in Roe's flux splitting scheme [32] based on the "Roe state."

The drawback of this treatment is that the order of accuracy of the resultant scheme is usually less than the order of the polynomial employed in the reconstruction. This is because the flux calculation is based on an approximated Riemann problem, in which the flow property variation inside each CE is not directly taken into account in calculating the interfacial flux. In addition, the linearization process in the flux splitting also involves approximation. For these reasons, modern upwind schemes are often referred to as *high-resolution* instead of high-order, even when a third- or fourth-order polynomial was used in the reconstruction.

Moreover, the logic of the upwind biasing is largely based on the characteristic-value splitting, which is sensible only when it is used to describe the space-time evolution of a smooth flow. The application of this method in a situation without space-time evolution and with the possible existence of a shock is difficult to justify.

In addition, the above characteristics-based technique is applicable only to flows in one spatial dimension. For flows in multiple spatial dimensions, the MOC is more complex and has its own inherent space-time geometries. Therefore, it is not amenable to any numerical method using a predetermined lattice stencil in the space-time domain. As a result, directional splitting is employed for solving flow equations in multiple spatial dimensions. This practice causes deterioration of numerical resolution in multi-dimensional flows. Furthermore, because source terms have no preferred direction, the directional splitting approach poses significant difficulties in constructing an approximate Riemann solver.

3.4 Summary of Distinguishing Features

To recapitulate the distinguishing features of the modern upwind schemes, we provide the following remarks: (i) In the reconstruction step, the flow structure is estimated by a curve fit among neighboring cells. The use of a polynomial curve fit over a possible solution discontinuity at cell interfaces is theoretically unjustifiable, and also leads to spurious oscillations. A *priori* knowledge of the flow solution, such as a TVD property, is then used to suppress these oscillations. (ii) The need to solve the Riemann problem in the modern upwind schemes is due to the choice of a fixed spatial mesh in the space-time domain as shown in Fig. 2.1. As a result of the choice, flow information propagates in both directions through the interface, and the upwind-biasing becomes necessary in calculating the interfacial flux. (iii) Only the flow properties immediately adjacent to an interface are used to calculate the interfacial flux. Due to the variation of the flow properties inside each CE, the actual evolution of the interfacial flux is much more complex. (iv) Given two distinct flow states in contact, the calculation of the interfacial flux is carried out by using characteristics-based splitting, which is valid only for describing the space-time evolution of a smooth flow. (v) For flows in multiple spatial dimensions, directional splitting is used to implement one-dimensional characteristic flux splitting. This practice causes deterioration of numerical resolution and difficulties in solving conservation laws with source terms.

4 The CE/SE Method in One Spatial Dimension

In the setting of evolution and reconstruction, the following distinguishing features of the CE/SE method result in a simpler and more consistent numerical flow model: (i) A space-time discretization is chosen for the flux conservation such that there is no Riemann problem to be solved at the cell interface. (ii) To avoid imposing predetermined constraints on the flow solutions such as monotonicity and TVD, the flow properties and their gradients are treated as unknowns. For smooth flows, the unknowns are completely determined by flux conservation, and the resultant numerical procedure can march forward and backward in time. (iii) For flows with shocks, an adjustable artificial damping is added to the discretized equations such that the numerical entropy condition is satisfied.

In the following, we shall first discuss the preliminaries of the CE/SE method. We then present the evolution step for calculating flow properties, the reconstruction step for obtaining the flow property gradient, and the shock capturing method for resolving shocks.

4.1 Preliminaries

In Fig. 4.1, we illustrate the nodes, denoted by dots (filled circles), where the unknowns are located. The space and time intervals between neighboring nodal lines are respectively denoted by $\Delta x/2$ and $\Delta t/2$. There is a Solution Element (SE) associated with each node (j, n). Let the $SE(j, n)$ be the interior of the space-time region bounded by a dashed line depicted in Fig. 4.2. It includes a horizontal line segment, a vertical line segment, and their immediate neighborhood. Between SEs, discontinuities are allowed. As will be shown immediately, for the Euler equations, Eq. (2.3), which have no source terms, the actual size of the neighborhood does not matter. However, for other equations with source terms, because these sources may be distributed over the entire computational domain, solution elements must be constructed such that they can fill the entire domain. In this case the neighborhood may be chosen such that a SE looks like the rhombus depicted in Fig. 4.8(b). Inside a SE, the flow properties are assumed continuous.

Within a SE, the flow property vector $\mathbf{U}$ and the flux vector $\mathbf{F}$ are approximated by their discretized counterparts $\mathbf{U}^*$ and $\mathbf{F}^*$. Since a second-order scheme is desired, piecewise linear distributions $\mathbf{U}^*$ and $\mathbf{F}^*$ are assumed. For (x, t) in $SE(j, n)$, we assume that

$$\mathbf{U}^*(x, t; j, n) = \mathbf{U}_j^n + (\mathbf{U}_x)_j^n (x - x_j) + (\mathbf{U}_t)_j^n (t - t^n) \quad (4.1)$$

and

$$\mathbf{F}^*(x, t; j, n) = (\mathbf{F})_j^n + (\mathbf{F}_x)_j^n (x - x_j) + (\mathbf{F}_t)_j^n (t - t^n). \quad (4.2)$$

Here $\mathbf{F}_j^n$ is the column matrix $\mathbf{F}$ (which is a function of $\mathbf{U}$) evaluated with $\mathbf{U} = \mathbf{U}_j^n$.

The expansion coefficients $(\mathbf{U}_t)_j^n$, $(\mathbf{F}_x)_j^n$ and $(\mathbf{F}_t)_j^n$ in Eqs. (4.1) and (4.2) will be expressed as functions of the independent unknowns $\mathbf{U}_j^n$ and $(\mathbf{U}_x)_j^n$ of the present scheme by assuming Eqs. (3.2) and (3.3), and

$$(\mathbf{F}_t)_j^n = \mathbf{A}_j^n (\mathbf{U}_t)_j^n, \quad (4.3)$$

which is the numerical analogue of the relation $\mathbf{F}_t = \mathbf{A}\mathbf{U}_t$. Furthermore, because $\vec{h}_m = (f_m, u_m)$, $m = 1, 2, 3$, we shall assume that for $m = 1, 2, 3$,

$$\vec{h}_m^*(x, t; j, n) = (f_m^*(x, t; j, n), u_m^*(x, t; j, n)), \quad (4.4)$$

where u_m^* and f_m^*, $m = 1, 2, 3$, are the components of the column matrices $\mathbf{U}^*$ and $\mathbf{F}^*$, respectively.

At this juncture, note that: (i) In an alternative approach to be presented in Sec. 4.3, the differential condition Eq. (3.3) is not assumed. Rather it arises naturally as a result of flux conservation and Eqs. (4.1) and (4.2). (ii) Hereafter, the components of the column matrices $\mathbf{U}_j^n$, $(\mathbf{U}_x)_j^n$, $(\mathbf{U}_t)_j^n$, $\mathbf{F}_j^n$, $(\mathbf{F}_x)_j^n$ and $(\mathbf{F}_t)_j^n$, will be denoted by $(u_m)_j^n$, $(u_{mx})_j^n$, $(u_{mt})_j^n$, $(f_m)_j^n$, $(f_{mx})_j^n$ and $(f_{mt})_j^n$, $m = 1, 2, 3$, respectively.

4.2 The Evolution and Reconstruction Steps

For smooth flows, the calculations of $\mathbf{U}_j^n$ and $(\mathbf{U}_x)_j^n$ are determined by requiring fluxes to be conserved over space-time Conservation Elements (CEs). As depicted in Figs. 4.3(a) and 4.3(b), two CEs, denoted by $CE_-(j, n)$ and $CE_+(j, n)$, are associated with every mesh point (j, n). A glance over Figs. 4.1, 4.2, and 4.3 reveals that the set of $CE_\pm(j, n)$ over all mesh points (j, n) do not overlap among themselves and can fill the entire space-time computational domain.

For each (j, n), the following discrete analogue to the space-time flux conservation, Eq. (2.7), is imposed:

$$\oint_{S(CE_-(j,n))} \vec{h}_m^* \cdot \vec{ds} = 0, \quad m = 1, 2, 3, \quad (4.5)$$

and

$$\oint_{S(CE_+(j,n))} \vec{h}_m^* \cdot \vec{ds} = 0, \quad m = 1, 2, 3. \quad (4.6)$$

According to Figs. 4.2, 4.3a, and 4.3b, we have the following observations: (i) The edges $\overline{CB}$ and $\overline{CD}$ of $CE_-(j,n)$ lie in $SE(j-1/2, n-1/2)$; (ii) The edges $\overline{AB}$ and $\overline{AD}$ of $CE_-(j,n)$ lie in $SE(j,n)$; (iii) The edges $\overline{ED}$ and $\overline{EF}$ of $CE_+(j,n)$ lie in $SE(j+1/2, n-1/2)$; (iv) The edges $\overline{AD}$ and $\overline{AF}$ of $CE_+(j,n)$ lie in $SE(j,n)$. As a result, with the aid of the numerical counterpart of Eq. (2.7), and Eqs. (3.2), (3.3) and (4.1)–(4.4), we conclude that Eq. (4.5) leads to three relations involving the independent unknowns $\mathbf{U}_j^n$, $(\mathbf{U}_x)_j^n$, $\mathbf{U}_{j-1/2}^{n-1/2}$, $(\mathbf{U}_x)_{j-1/2}^{n-1/2}$, and Eq. (4.6) leads to another three relations, involving $\mathbf{U}_j^n$, $(\mathbf{U}_x)_j^n$, $\mathbf{U}_{j+1/2}^{n-1/2}$, and $(\mathbf{U}_x)_{j+1/2}^{n-1/2}$. Assuming that the unknowns at the mesh points $(j-1/2, n-1/2)$ and $(j+1/2, n-1/2)$ are given, the six components of $\mathbf{U}_j^n$ and $(\mathbf{U}_x)_j^n$ can be determined by the above six relations. In the following, it will be shown that the procedure for solving these relations can be divided into two sequential steps: an evolution step followed by a reconstruction step.

Note that the space-time flux of $\vec{h}^*$ leaving $CE_-(j,n)$ through $\overline{AD}$ and that leaving $CE_+(j,n)$ through $\overline{AD}$ are evaluated using the same unknowns, i.e., $\mathbf{U}_j^n$ and $(\mathbf{U}_x)_j^n$. Thus, these two space-time fluxes are each the negative of the other. As a result, a combination of Eq. (4.5) and Eq. (4.6) imply that

$$\oint_{S(CE(j,n))} \vec{h}_m^* \cdot \vec{ds} = 0, \quad m = 1, 2, 3, \quad (4.7)$$

where $CE(j,n)$ (see Fig. 4.3(c)) is the union of $CE_-(j,n)$ and $CE_+(j,n)$. Here, as explained above, the fluxes of $\vec{h}_m^*$ leaving $CE(j,n)$ through $\overline{CD}$, $\overline{CB}$, $\overline{ED}$, and $\overline{EF}$ can be evaluated in terms of $\mathbf{U}_{j\pm1/2}^{n-1/2}$ and $(\mathbf{U}_x)_{j\pm1/2}^{n-1/2}$.

The flux of $\vec{h}_m^*$ leaving $\overline{BF}$, however, is a function of $\mathbf{U}_j^n$ only. This conclusion can be reached from the following observations: (i) $dt = 0$ along $\overline{BF}$, (ii) $\overline{BF}$ lies in $SE(j,n)$, (iii) u_m^*, $m = 1, 2, 3$, is linear in x on $\overline{BF}$, and (iv) the mesh point (j,n), i.e., point

A, is the mid-point of $\overline{BF}$. As a matter of fact, with the aid of Eq. (2.8), it can be shown that the flux of $\vec{h}_m^*$ leaving $CE(j,n)$ through $\overline{BF}$ is simply $(u_m)_j^n \Delta x$. In other words, Eq. (4.7) implies that $\mathbf{U}_j^n$ can be determined explicitly in terms of $\mathbf{U}_{j\pm1/2}^{n-1/2}$ and $(\mathbf{U}_x)_{j\pm1/2}^{n-1/2}$. This is the *evolution* step for the present marching scheme.

After obtaining $\mathbf{U}_j^n$, $\mathbf{F}_j^n$ and $\mathbf{A}_j^n$ can be determined because they are functions of $\mathbf{U}_j^n$ only. As a result, by applying either Eq. (4.5) or Eq. (4.6) (only one of these two equations is independent after Eq. (4.7) is used), one can obtain a system of three linear equations with the three unknowns being the three components of $(\mathbf{U}_x)_j^n$. In other words, $(\mathbf{U}_x)_j^n$ can be determined in terms of $\mathbf{U}_j^n$, $\mathbf{U}_{j\pm1/2}^{n-1/2}$, and $(\mathbf{U}_x)_{j\pm1/2}^{n-1/2}$ by solving either Eq. (4.5) or Eq. (4.6). This is the *reconstruction* step of the present marching scheme.

Recall that the reason leading to the decoupling of the evaluation of $\mathbf{U}_j^n$ from that of $(\mathbf{U}_x)_j^n$ is that the mesh point (x_j, t^n) is located at the mid-point of $\overline{BF}$. For a space-time mesh with non-uniform spatial mesh intervals, the above decoupling can still be achieved, with some modifications in the procedure. We shall explain these modifications using a special nonuniform-spatial-interval mesh. From this description, the reader can easily infer the modifications required for a general nonuniform-spatial-interval mesh.

Consider the space-time mesh depicted in Fig. 4.4. Here, we assume that the spatial mesh intervals to the right and the left of the vertical mesh line $j = 0$ are separately uniform and have the lengths Δx and $\Delta x'$, respectively. Spatial mesh intervals are separated from one another by solid vertical mesh lines. Each spatial mesh interval is divided into two subintervals of equal length by a dashed vertical line. For this space-time mesh, non-uniformity of spatial intervals occurs around the mesh line $j = 0$. Thus, we are interested in the marching scheme solving for $\mathbf{U}_0^n$ and $(\mathbf{U}_x)_0^n$. Note that the mesh point $(0, n)$ is not located at the intersection of mesh lines. Instead, it is located at the mid-point of $\overline{BF}$ (see Fig. 4.5). Refer to Figs. 4.4–4.6 for the mesh point $(0, n)$ and its associated SEs and CEs (for $n = 1/2, 3/2, ...$). As a result, in this case too, $\mathbf{U}_0^n$ can be calculated by using Eq (4.7) only. Once $\mathbf{U}_0^n$ is obtained, $(\mathbf{U}_x)_0^n$ can be calculated by either Eq. (4.5) or Eq. (4.6).

10

4.3 An Alternative Construction

As described in [6], the above CE/SE scheme for isentropic flows can be constructed from a different flux-balance perspective. In this construction, the locations of mesh points (dots in Fig. 4.7) are identical to that in the original construction (refer to Fig. 4.1). The computation domain, however, is divided into rhombic regions in the interior and triangular regions at the boundary. Each region is associated with a mesh point (j, n) and serves as a conservation element, as denoted by CE$'(j, n)$ (refer to Fig. 4.8(a)). The solution element associated with point (j, n) is the interior of CE$'(j, n)$, and is denoted by SE$'(j, n)$ (refer to Fig. 4.8(b)).

For any (x, t) in SE$'(j, n)$, $\mathbf{U}(x, t)$, $\mathbf{F}(x, t)$ and $\vec{h}_m(x, t)$ are approximated by their discretized counterparts: $\mathbf{U}^*(x, t; j, n)$, $\mathbf{F}^*(x, t; j, n)$, and $\vec{h}_m^*$ as defined in Eqs. (4.1), (4.2), and (4.4), respectively. Here, the marching scheme requires that for each mesh point (j, n), the space-time flux of $\vec{h}_m^*$ leaving the boundary of the space-time rhombus CE$'(j, n)$ vanishes, i.e.,

$$\oint_{S(CE'(j,n))} \vec{h}_m^* \cdot \vec{ds} = 0, \quad m = 1, 2, 3. \quad (4.8)$$

By using Eqs. (4.1), (4.2) and (4.4), we have, for $m = 1, 2, 3$,

$$\nabla \cdot \vec{h}_m^*(x, t; j, n) = (f_{mx})_j^n + (u_{mt})_j^n, \quad (4.9)$$

for any (x, t) in SE$'(j, n)$. By using Gauss' divergence theorem and the fact that the right-hand-side of Eq. (4.9) is a constant within SE$'(j, n)$, Eq. (4.8) implies that

$$(f_{mx})_j^n + (u_{mt})_j^n = 0, \quad m = 1, 2, 3, \quad (4.10)$$

i.e., $(\mathbf{U}_t)_j^n + (\mathbf{F}_x)_j^n = \mathbf{O}$, which is Eq. (3.3). Here, however, Eq. (4.8) is not an imposed assumption as it was in the original construction. Instead, it is the result of the flux conservation condition Eq. (4.8).

In the present construction, the interaction between two neighboring SE's lies in the balance of space-time flux over the oblique interface that separates them. Refer to Fig. 4.9. Note that distinct flow states exist along the two sides of the interface. As would be done in the derivation of the Rankine-Hugoniot relation, we impose the space-time-flux-balance condition across the interface, i.e., the space-time flux leaving CE$'(j - 1/2, n - 1/2)$ through $\overline{B'C'}$ should be the space-time flux entering CE$'(j, n)$ through $\overline{BC}$.

In order to avoid the ambiguity of a double-valued function, SE's are defined such that an interface separating two rhombic CE's does not belong to any of the two adjacent SE's. As depicted in Fig. 4.9, the values of $\vec{h}_m^*$ along $\overline{BC}$ and $\overline{B'C'}$ are evaluated using information from SE$'(j, n)$ and SE$'(j - 1/2, n - 1/2)$, respectively. Moreover, by using Eqs. (4.9) and (4.10), and Gauss' divergence theorem, we have

$$\oint_{S(\triangle ABC)} \vec{h}_m^* \cdot \vec{ds} = 0, \quad m = 1, 2, 3, \quad (4.11)$$

and

$$\oint_{S(\triangle A'B'C')} \vec{h}_m^* \cdot \vec{ds} = 0, \quad m = 1, 2, 3. \quad (4.12)$$

Equation (4.5) follows immediately from Eqs. (4.11) and (4.12) and the interface flux balance condition. Similarly, Eq. (4.6) can also be obtained. As a result, the marching scheme constructed using the alternative approach is identical to that constructed with the original approach.

Note that, as a result of Eq. (4.11), the flux of $\vec{h}_m^*$ leaving $\triangle ABC$ through the oblique line segment $\overline{BC}$ is balanced by the flux of $\vec{h}_m^*$ leaving $\triangle ABC$ through the horizontal line segment $\overline{AB}$ (where $dt = 0$) and the vertical line segment $\overline{AC}$ (where $dx = 0$). It follows that, in general, flux integration along an oblique line segment can be turned into an equivalent and simpler problem that involves only flux integration along a horizontal line segment and a vertical line segment.

4.4 Special Features of the CE/SE Method for Isentropic Flows

We conclude the discussion of the CE/SE method for isentropic flows by the following remarks: (i) Since the space-time fluxes at the interfaces of CEs cancel each other, the local conservation conditions, Eqs. (4.5) and (4.6), lead to a global conservation condition

$$\oint_{S(V')} \vec{h}_m^* \cdot \vec{ds} = 0, \quad m = 1, 2, 3, \quad (4.13)$$

where V' is the union of any combination of CE$_+$s and CE$_-$s. Similarly, as a result of Eq. (4.8), Eq. (4.13) is also valid in the alternative construction if V' is the union of any combination of CE's. (ii) The present marching scheme is a two-level explicit scheme. Only one set of initial conditions at a single

time level is needed to start the computation, and the numerical solution is completely determined by the initial conditions. This is in full compliance with the flow physics. (iii) Backward marching schemes can also be constructed using Eqs. (4.5) and (4.6). For each mesh point (j, n), they also imply

$$\oint_{S(CE_+(j-1/2,n+1/2))} \vec{h}_m^* \cdot \vec{ds} = 0, \qquad (4.14)$$

and

$$\oint_{S(CE_-(j+1/2,n+1/2))} \vec{h}_m^* \cdot \vec{ds} = 0. \qquad (4.15)$$

Note that the rectangular space-time regions to the left and the right of the mesh point (j, n) depicted in Fig. 4.10 are $CE_+(j - 1/2, n + 1/2)$ and $CE_-(j + 1/2, n + 1/2)$, respectively. Previously, we noted that Eqs. (4.5) and (4.6) show that U_j^n and $(U_x)_j^n$ are determined in terms of $U_{j\pm1/2}^{n-1/2}$ and $(U_x)_{j\pm1/2}^{n-1/2}$. Here, Eqs. (4.14) and (4.15) show that U_j^n and $(U_x)_j^n$ can be determined in terms of $U_{j\pm1/2}^{n+1/2}$ and $(U_x)_{j\pm1/2}^{n+1/2}$. In other words, the same set of local conservation conditions that was used to construct the forward-marching scheme can also be used to construct the backward-marching scheme. (iv) Recall the invariance property of the Euler equations under space-time inversion that was discussed in Sec. 1.2. Let

$$U = U^o(x, t) \qquad (4.16)$$

be a smooth solution to Eq. (2.3). For this solution, we have

$$U_x = U_1^o(x, t) \equiv \frac{\partial U^0(x, t)}{\partial x}, \qquad (4.17)$$

for any x and t. Previously, we showed that

$$U = U^o(2x_o - x, 2t_o - t) \qquad (4.18)$$

is also a solution to the Euler equations, Eq. (2.3). For the above solution, Eq. (4.18), the spatial gradients of the flow properties can be expressed as

$$\begin{aligned}
U_x &= \frac{\partial U^o(2x_o - x, 2t_o - t)}{\partial(2x_o - x)} \frac{d(2x_o - x)}{dx}, \\
&= -U_1^o(2x_o - x, 2t_o - t). \qquad (4.19)
\end{aligned}$$

The present numerical analogue to the Euler equations, Eq. (2.3), also possesses the invariance property under space-time inversion. Let $(U^o)_j^n$, and $(U_1^o)_j^n$ be given at all mesh points (j, n). Let

$$U_j^n = (U^o)_j^n \quad ; \qquad (U_x)_j^n = (U_1^o)_j^n \qquad (4.20)$$

be a solution to Eqs. (4.5) and (4.6). With the aid of the equations provided in the previous sections, it can be shown that

$$\begin{aligned}
U_j^n &= (U^o)_{2j_o-j}^{2n_o-n} \quad ; \\
(U_x)_j^n &= -(U_1^o)_{2j_o-j}^{2n_o-n} \qquad (4.21)
\end{aligned}$$

are also a solution to Eqs. (4.5) and (4.6). Here, $(2j_o - j, 2n_o - n)$ is the image of (j_o, n_o) and vice-versa, under space-time inversion with respect to (j_o, n_o) (see Fig. 4.11). Note that Eqs. (4.20) and (4.21) are the numerical counterparts of Eqs. (4.16–17) and (4.18–19), respectively.

It has been shown by numerical experiments that the present scheme is neutrally stable in the interior of the computational domain up to at least a thousand time steps when the Courant number does not exceed unity. In these numerical experiments simulating a shock-tube problem, the computational domain was allowed to grow with time, so that the undisturbed fluid state could always be prescribed at the computational boundaries as the exact solution. As a matter of fact, by using an analysis similar to that given at the end of Sec. 6 in [6], one can show that the linearized form of the present numerical analogue is neutrally stable when the Courant number does not exceed unity. The above conclusions are consistent with a remark made in Sec. 2.3, i.e., a numerical analogue of the Euler equations that is stable and also preserves the invariance property under space-time inversion must be neutrally stable. The Courant number mentioned above, denoted by CFL, is defined at each time level n for 1D flow computations as

$$CFL = \max_j \left[\frac{(|v_j^n| + |c_j^n|) \Delta t}{\Delta x} \right],$$

where c is the sonic speed. Analogous Courant numbers can be defined for 2D flow computations.

To summarize, the present numerical scheme preserves the forward-backward marching nature and the space-time inversion invariance property of the Euler equations. The present scheme also meets the requirements of an ideal numerical analogue set forth in Sec. 2.4, i.e., it should be neutrally stable, explicit, and involving only two time levels. The present scheme can be considered as a nonlinear extension of the a scheme, i.e., the inviscid ($\mu = 0$) version of the a-μ scheme described in [6]. Because the a scheme is neutrally stable, generally one would expect that a nonlinear extension of such a scheme is numerically unstable. The present scheme appears to be an exception to this common wisdom.

12

4.5 The Shock-Capturing Scheme

The above marching scheme for isentropic flows can be expressed as

$$\mathbf{U}_j^n =$$
$$\mathbf{H}\left(\mathbf{U}_{j-1/2}^{n-1/2}, \mathbf{U}_{j+1/2}^{n-1/2}, (\mathbf{U}_x)_{j-1/2}^{n-1/2}, (\mathbf{U}_x)_{j+1/2}^{n-1/2}\right), \quad (4.22)$$

and

$$(\mathbf{U}_x)_j^n =$$
$$\mathbf{H}_x\left(\mathbf{U}_{j-1/2}^{n-1/2}, \mathbf{U}_{j+1/2}^{n-1/2}, (\mathbf{U}_x)_{j-1/2}^{n-1/2}, (\mathbf{U}_x)_{j+1/2}^{n-1/2}\right) \quad (4.23)$$

Here $\mathbf{H}$ and $\mathbf{H}_x$ are column-matrix functions. Their explicit forms can be obtained from Eqs. (5.20)-(5.29) in [6] with the assumption that the viscosity $\mu = 0$. In the construction of the shock-capturing scheme, the local conservation condition, Eq. (4.7), is again assumed. Because Eq. (4.22) follows directly from Eq. (4.7), the former is incorporated into the shock-capturing scheme without modification. As a result, given the same $\mathbf{U}_{j\pm1/2}^{n-1/2}$ and $(\mathbf{U}_x)_{j\pm1/2}^{n-1/2}$, the shock-capturing scheme shares with the non-dissipative scheme the same zero-order terms on the right sides of Eqs. (4.1) and (4.2). In addition, the shock-capturing scheme observes a global conservation condition that is also a direct result of Eq. (4.7), i.e., for any space-time region that is the union of any combination of the CEs of the type depicted in Fig. 4.3(c), the total flux of $\vec{h}_m^*$, $m = 1, 2, 3$, leaving its boundary vanishes.

The shock-capturing scheme is obtained by modifying Eq. (4.23). To proceed, let

$$(\mathbf{U}')_{j\pm1/2}^n = \mathbf{U}_{j\pm1/2}^{n-1/2} + \frac{\Delta t}{2}(\mathbf{U}_t)_{j\pm1/2}^{n-1/2}, \quad (4.24)$$

i.e., $(\mathbf{U}')_{j\pm1/2}^n$ is a first-order Taylor's approximation of $\mathbf{U}$ at the mesh point $(j \pm 1/2, n)$. Thus,

$$(\mathbf{U}_x^c)_j^n = \frac{(\mathbf{U}')_{j+1/2}^n - (\mathbf{U}')_{j-1/2}^n}{\Delta x} \quad (4.25)$$

is a central-difference approximation of $\mathbf{U}_x$ at the mesh point (j, n). In the shock-capturing scheme, Eq. (4.23) is replaced by

$$(\mathbf{U}_x)_j^n = (1 - 2\epsilon)(\mathbf{H}_x)_j^n + 2\epsilon(\mathbf{U}_x^c)_j^n, \quad (4.26)$$

where $(\mathbf{H}_x)_j^n$ denotes the expression on the right side of Eq. (4.23), and ϵ is a real number. Note that $(\mathbf{U}_x^c)_j^n$ is defined in terms of a central-difference approximation. Generally, numerical dissipation is introduced as a result of using such an approximation. On the other hand, $(\mathbf{H}_x)_j^n$ represents the solution from a non-dissipative scheme. The right side of Eq. (4.26) is a weighted averaged of $(\mathbf{H}_x)_j^n$ and $(\mathbf{U}_x^c)_j^n$ with the weight factor of $1 - 2\epsilon$ and 2ϵ, respectively. Therefore, one may heuristically conclude that the numerical dissipation associated with the shock-capturing scheme can be increased by increasing the value of ϵ. This conclusion is verified by numerical experiments. As shown in [6], the stability domain of the shock-capturing scheme is defined by

$$CFL \leq 1 \quad \text{and} \quad 0 \leq \epsilon \leq 1. \quad (4.27)$$

Note that Eq. (4.26) can also be expressed as

$$(\mathbf{U}_x)_j^n = (\mathbf{H}_x)_j^n + 2\epsilon(D\mathbf{U})_j^n, \quad (4.28)$$

or

$$(\mathbf{U}_x)_j^n = (\mathbf{U}_x^c) + (2\epsilon - 1)(D\mathbf{U})_j^n, \quad (4.29)$$

where

$$(D\mathbf{U})_j^n \equiv (\mathbf{U}_x^c)_j^n - (\mathbf{H}_x)_j^n. \quad (4.30)$$

According to Eq. (4.28), $(\mathbf{U}_x)_j^n$ for the shock-capturing scheme is the sum of the non-dissipative term $(\mathbf{H}_x)_j^n$ and the dissipative term $2\epsilon(D\mathbf{U})_j^n$. The latter provides the necessary entropy-increasing condition within the stability domain defined by Eq. (4.27). Also it is seen from Eqs. (4.28) and (4.29) that $(\mathbf{U}_x)_j^n$ reduces to $(\mathbf{H}_x)_j^n$ and $(\mathbf{U}_x^c)_j^n$ in the cases of $\epsilon = 0$ and $\epsilon = 0.5$, respectively.

We remark that there is a slight difference between the shock-capturing scheme defined above and that defined by Eqs. (4.24)-(4.28) in [6]. Hereafter, the latter is referred to as the simplified shock-capturing scheme. Both schemes assume Eq. (4.22). However, Eq. (4.26) is replaced by

$$(\mathbf{U}_x)_j^n = (1 - 2\epsilon)(\mathbf{H}_x')_j^n + 2\epsilon(\mathbf{U}_x^c)_j^n \quad (4.31)$$

in the simplified scheme. Here,

$$(\mathbf{H}_x')_j^n = (\mathbf{U}_x^c)_j^n - (d\mathbf{U})_j^n \quad (4.32)$$

with

$$(d\mathbf{U})_j^n = \frac{1}{2}\left[(\mathbf{U}_x)_{j+1/2}^{n-1/2} + (\mathbf{U}_x)_{j-1/2}^{n-1/2}\right] - \frac{\mathbf{U}_{j+1/2}^{n-1/2} - \mathbf{U}_{j-1/2}^{n-1/2}}{\Delta x}. \quad (4.33)$$

Note that $(d\mathbf{U})_j^n$ is the difference of two numerical approximations to $(\mathbf{U}_x)_j^{n-1/2}$: (i) an average of $(\mathbf{U}_x)_{j+1/2}^{n-1/2}$ and $(\mathbf{U}_x)_{j-1/2}^{n-1/2}$, and (ii) a central-difference approximation in terms of $\mathbf{U}_{j+1/2}^{n-1/2}$ and $\mathbf{U}_{j-1/2}^{n-1/2}$. Also, as a result of Eqs. (3.2), (3.3), (4.24),

13

(4.25), (4.32), and (4.33), $(H'_x)^n_j$ is a simple explicit function of $U^{n-1/2}_{j\pm1/2}$ and $(U_x)^{n-1/2}_{j\pm1/2}$. On the other hand, as explained earlier, the evaluation of $(H_x)^n_j$ requires the solution of a system three linear equations at each mesh point (j,n).

It follows from Eqs. (4.31) and (4.32) that, in the simplified scheme, Eqs. (4.28) and (4.29) should be replaced by

$$(U_x)^n_j = (H'_x)^n_j + 2\epsilon(dU)^n_j \qquad (4.34)$$

and

$$(U_x)^n_j = (U^c_x)^n_j + (2\epsilon - 1)(dU)^n_j, \qquad (4.35)$$

respectively.

The reader is reminded that Eq. (4.31) was originally derived [6] from a "natural generalization" of the $a - \epsilon$ scheme, which is a solver for a scalar convection equation with a constant convection speed a. In a paper to be published, the conditions under which $(H_x)^n_j$ can be approximated by $(H'_x)^n_j$ will be given.

By comparison with the shock-capturing scheme defined by Eqs. (4.22) and (4.26), the simplified scheme is more computationally efficient. Yet the numerical results are almost identical to those generated by the former except when ϵ is very small (≤ 0.03). Note that the stability domain of the simplified scheme is approximately

$$CFL \leq 1 \quad \text{and} \quad 0.03 \leq \epsilon \leq 1. \qquad (4.36)$$

The two shock-capturing schemes described above generally can capture shocks with high resolution and without generating substantial numerical oscillations near shock if $0.3 \leq \epsilon \leq 0.8$. To further damp out these oscillations, $(U^c_x)^n_j$ in Eq. (4.29) (which is equivalent to Eq. (4.26)) and Eq. (4.35) (which is equivalent to Eq. (4.31)) can be modified using a weighting procedure [6]. Let

$$(U_{x\pm})^n_j = \pm\frac{(U')^n_{j\pm1/2} - U^n_j}{\Delta x/2}. \qquad (4.37)$$

Because $(U')^n_{j\pm1/2}$ and U^n_j are the numerical analogues of U at the mesh points $(j\pm1/2,n)$ and (j,n), respectively, $(U_{x+})^n_j$ and $(U_{x-})^n_j$ are two numerical analogues of U_x at the mesh point (j,n), with one being evaluated from the right and another from the left. It follows from Eqs. (4.25) and (4.37) that

$$(U^c_x)^n_j = \frac{1}{2}\left[(U_{x+})^n_j + (U_{x-})^n_j\right], \qquad (4.38)$$

i.e., $(U^c_x)^n_j$ is the simple average of $(U_{x+})^n_j$ and $(U_{x-})^n_j$. The nonlinear weighting function is defined as

$$W(x_-, x_+; \alpha) = \frac{|x_+|^\alpha x_- + |x_-|^\alpha x_+}{|x_+|^\alpha + |x_-|^\alpha} \qquad (4.39)$$

where x_-, x_+, and α are real variables with $|x_+| + |x_-| > 0$ and $\alpha \geq 0$. Note that (i) $W(x_-, x_+; 0)$ is the simple average of x_- and x_+, and (ii) $W(x_-, x_+; 1)$ and $W(x_-, x_+; 2)$ are used in the slope-limiters proposed by van Leer [11] and van Albada et al. [33], respectively.

Recall that $(u_{mx\pm})^n_j$ denotes the m-th component of $(U_{x\pm})^n_j$. Let

$$(u^w_{mx})^n_j = W\left[(u_{mx-})^n_j, (u_{mx+})^n_j, \alpha\right] \qquad (4.40)$$

Then, as shown in [6], numerical oscillations near shocks can be suppressed very efficiently if $(U^c_x)^n_j$ is replaced by $(U^w_x)^n_j$, i.e., the column matrix formed by $(u^w_{mx})^n_j$, $m = 1, 2, 3$, if $\alpha \geq 1$.

4.6 Concept of Dual Space-Time Mesh and Development of New Implicit Solvers

In addition to being used in the Euler solvers described in this section, the uniform space-time mesh depicted in Fig. 4.1 was also used in the simplest CE/SE scheme, i.e., the a scheme referred to earlier. The latter is a solver for the pure convection equation

$$\frac{\partial u}{\partial t} + a\frac{\partial u}{\partial x} = 0, \qquad (4.41)$$

where a is a constant convection speed.

The mesh depicted in Fig. 4.1 is staggered in time, i.e., the mesh points that have the same spatial location appear only at alternating time levels. In Fig. 4.12, the mesh depicted in Fig. 4.1 is superimposed on another staggered mesh, with the mesh points of the latter being marked by filled triangular symbols. Hereafter, in this subsection, (i) the mesh depicted in Fig. 4.1, with its mesh points marked by dots, is referred to as the original mesh, (ii) the mesh with its mesh points marked by triangles is referred to as the alternate mesh, and (iii) the combination of the above two meshes, i.e., that depicted in Fig. 4.12, is referred to as the dual mesh. As shown in Fig. 4.13, a CE of a mesh point marked by a triangle may coincide with a CE of another mesh point marked by a dot.

Obviously, through a similar procedure, one can construct the a scheme or any CE/SE Euler solver using the alternate mesh. As a matter of fact, one can even combine two independent a schemes, one constructed on the original mesh, and the other on the alternate mesh. This combined explicit scheme, referred to as the dual a scheme, has two completely decoupled solutions.

In [8], the construction of two implicit solvers for the convection-diffusion equation

$$\frac{\partial u}{\partial t} + a\frac{\partial u}{\partial x} - \mu\frac{\partial^2 u}{\partial x^2} = 0 \qquad (\mu \geq 0) \qquad (4.42)$$

was described. These two solvers are referred to as the a-$\mu(I1)$ and a-$\mu(I2)$ schemes, respectively. Here "I" stands for "implicit", and is used to distinguish these schemes from the explicit a-μ scheme described in [1, 4, 6].

In the case that $\mu = 0$, both the a-$\mu(I1)$ and the a-$\mu(I2)$ schemes reduce to the non-dissipative (explicit) dual a scheme. As a result, these two schemes have the important property that their numerical dissipation approaches zero as the physical dissipation approaches zero. This property provides better simulation of nearly inviscid flows (i.e., large-Reynolds-number flows), and ensures that numerical dissipation will not overwhelm physical dissipation. Furthermore, as can be inferred from several discussions in [6], the amplification factors arrived at by a von Neumann analysis of the dual a scheme are identical to those of the Leapfrog scheme. In other words, for the special case $\mu = 0$, the amplification factors of the a-$\mu(I1)$ and the a-$\mu(I2)$ schemes reduce to those of the Leapfrog scheme.

In case that $\mu > 0$, both a-$\mu(I1)$ and a-$\mu(I2)$ schemes become implicit. This is consistent with the fact that, for $\mu > 0$, the value of a solution to Eq. (4.42) at any point (x, t) depends on the initial data and all the boundary data up to the time t. In other words, an implicit scheme should be used to solve an initial/boundary-value problem, such as one involving Eq. (4.42). This requirement becomes more important as the diffusion term in Eq. (4.42) becomes more dominant.

Furthermore, for both a-$\mu(I1)$ and a-$\mu(I2)$ schemes, the solution at the mesh points marked by dots is coupled with that at the mesh points marked by triangles if $\mu > 0$. Also, it was shown in [8] that, in the pure diffusion case (i.e., when $a = 0$), the principal amplification factors of both a-$\mu(I1)$ and a-$\mu(I2)$ schemes are identical to the amplification factor of the Crank-Nicolson scheme. Note that the latter has only one amplification factor.

Finally, note that both a-$\mu(I1)$ and a-$\mu(I2)$ schemes are stable if the Courant number $|\nu| \equiv |a|\,\Delta t\,/\,\Delta x \leq 1$. Also, both schemes have second-order accuracy in space and time if ν is held constant. This is numerically confirmed in [8].

5   The Euler Solvers for Multiple Spatial Dimensions

5.1   The 2D Euler Solvers

In Sec. 4, it was established that there were only *two* sets of independent marching variables, i.e., (i) $(u_m)_j^n$, $m = 1, 2, 3$, and (ii) $(u_{mx})_j^n$, $m = 1, 2, 3$, at each mesh point (j, n), if Eqs. (3.2), (3.3) and (4.1)–(4.4) are assumed. As a result, it requires *two* sets of conservation conditions, i.e., Eqs. (4.5) and (4.6) to construct the 1D non-dissipative Euler scheme. As a prerequisite to Eqs. (4.5) and (4.6), *two* CEs, i.e., $CE_-(j, n)$ and $CE_+(j, n)$ are defined for each mesh point (j, n).

The 2D CE/SE non-dissipative Euler solver [5, 7] was constructed using the same set of design principles that was used to construct its 1D counterpart. The differences between them stem entirely from the fact that there is one more spatial dimension to be considered in the 2D solver. In this section, only the basic geometric structures of the 2D solver will be described. For other details, the reader is referred to [5].

The 2D unsteady Euler equations of a perfect gas [5, 7] consist of *four* independent equations, $m = 1$, 2, 3, 4, instead of the *three* equations applicable to 1D flow. Also, in the 2D case, there are two spatial components of the gradient of each u_m (i.e., u_{mx} and u_{my}, where x and y are Cartesian coordinates for the 2D space). This is in contrast to the 1D case, in which, for each u_m, there is only one spatial component of the gradient of u_m (i.e., u_{mx}).

In the development of the 2D non-dissipative Euler solver and its extensions [5], a set of equations that is a natural 2D extension of Eqs. (3.2), (3.3) and (4.1)–(4.4) is assumed. As a result, there are *three* sets of independent marching variables at each mesh point (j, k, n) (see Figs. 5.1 and 5.2 for the lo-

cations of the mesh points. The reader is referred to [5, 7] for the definitions of the spatial mesh indices j and k). They are $(u_m)_{j,k}^n$, $(u_{mx})_{j,k}^n$ and $(u_{my})_{j,k}^n$, $m = 1, 2, 3, 4$. It follows that it requires *three* sets of conservation conditions (each set comprises four conditions, corresponding to $m = 1, 2, 3, 4$) at each mesh point to construct the 2D non-dissipative solver. Therefore, as a prerequisite, one must define *three* conservation elements for each mesh point. The construction of these CEs, which is the most intriguing part of the development of the 2D CE/SE Euler solver, will be described in what immediately follows.

Consider a spatial domain formed by congruent triangles (see Fig. 5.1). The center of each triangle is marked by either an empty circle or a filled circle. The distribution of these empty and filled circles is such that if the center of a triangle is marked by a filled (empty) circle, then the centers of the three neighboring triangles with which the first triangle shares a side are marked by empty (filled) circles. As an example, point G, the center of the triangle $\triangle BDF$, is marked by a filled circle while points A, C and E, the centers of the triangles $\triangle BFM$, $\triangle BJD$ and $\triangle DLF$, respectively, are marked by empty circles. These centers are the spatial projections of the space-time mesh points used in the 2D solver [5, 7].

To specify the exact locations of the mesh points in space-time, one must also specify their temporal coordinates. In the 2D CE/SE development, again we assume that the mesh points are located at the time levels $n = 0 \pm 1/2$, ± 1, $\pm 3/2$, ..., with $t = n \Delta t$ at the nth time level. Furthermore, we assume that the spatial projections of the mesh points at a whole-integer (half-integer) time level are the points marked by empty (filled) circles in Fig. 5.1.

Let the triangles depicted in Fig. 5.1 lie on the time level $n = 0$. Then those points marked by empty circles are the mesh points at this time level. On the other hand, those points marked by filled circles are not the mesh points at the time level $n = 0$. They are only the spatial projections of the mesh points at half-integer time levels.

Points A, C and E, which are depicted in Figs. 5.1 and 5.2(a), are three mesh points at the time level $n = 0$. Point G', which is depicted in Fig. 5.2(a), is a mesh point at the time level $n = 1/2$. Its spatial projection at the time level $n = 0$ is point G. Because point G is not a mesh point, it is not marked by a filled circle in the space-time plots given in Figs. 5.2(a)–(c). Hereafter, only a mesh point, e.g., point

G', will be marked by a filled or empty circle in a space-time plot.

The conservation elements associated with point G' are defined to be the space-time quadrilateral cylinders $GFABG'F'A'B'$, $GBCDG'B'C'D'$, and $GDEFG'D'E'F'$ that are depicted in Fig. 5.2(a). Here (i) points B, D and F are the vertices of the triangle with point G being its center (centroid) (see also Fig. 5.1), and (ii) points A', B', C', D', E' and F' are on the time level $n = 1/2$ with their spatial projections on the time level $n = 0$ being points A, B, C, D, E and F, respectively.

Point G' is a mesh point at a half-integer time level. For a mesh point at a whole-integer time-level, the conservation elements associated with it can be constructed in a similar fashion. As an example, consider Fig. 5.2(b). Here points $B'(B'')$, $I'(I'')$, $J'(J'')$, $K'(K'')$, $D'(D'')$, $G'(G'')$ and $C'(C'')$ are on the time level $n = 1/2$ ($n = 1$) with their spatial projections on the time level $n = 0$, respectively, being the points B, I, J, K, D, G and C that are depicted in Fig. 5.1. Point C'' is a mesh point at the time level $n = 1$. By definition, the conservation elements associated with point C'' are the quadrilateral cylinders $C'J'K'D'C''J''K''D''$, $C'D'G'B'C''D''G''B''$ and $C'B'I'J'C''B''I''J''$.

The CEs associated with point G' and those associated with point C'' are depicted in Figs. 5.2(a) and 5.2(b), respectively. The relative positions of these six CEs in the global space-time mesh are depicted in Fig. 5.2(c).

Recall that, in the development of the 1D non-dissipative Euler solver, a pair of diagonally opposite vertices of each $\text{CE}_\pm(j, n)$ (see Figs. 4.3(a) and (b)) are assigned as mesh points. Furthermore, the boundary of each $\text{CE}_\pm(j, n)$ is a subset of the union of the SEs associated with the two diagonally opposite mesh points of this CE. In the 2D development, as seen from Figs. 5.2(a)–(c), two diagonally opposite vertices of each CE are also assigned as mesh points. In the following, we shall define the SEs such that even in the 2D case, the boundary of a CE is again a subset of the union of the SEs associated with the two diagonally opposite mesh points of this CE.

As an example, the SE associated with point G' is depicted in Fig. 5.3(a). It is the union of three vertical rectangles (i.e., $G''B''BG$, $G''D''DG$ and $G''F''FG$), a horizontal hexagon (i.e., $A'B'C'D'E'F'$) and their immediate neighborhood.

Note that points G'', B'', D'' and F'' are on the time level $n = 1$ and their spatial projections on the time level $n = 0$ are points G, B, D and F, respectively. As another example, the SE associated with point C''' is depicted in Fig. 5.3(b). Again, it is the union of three vertical rectangles (i.e., $C'''D'''D'C'$, $C'''B'''B'C'$ and $C'''J'''J'C'$), a horizontal hexagon (i.e., $K''D''G''B''I''J''$) and their immediate neighborhood. Note that points C''', D''', B''' and J''' are on the time level $n = 3/2$, and their spatial projections on the time level $n = 0$ are points C, D, B and J, respectively. The definition of the SE of any mesh point at a half-integer (whole-integer) time level is similar to the definition of the SE of the point $G'(C'')$.

As depicted in Fig. 5.2(b), one of the CEs associated with point C''' is the space-time quadrilateral cylinder $C'D'G'B'C''D''G''B''$. Among the vertices of this CE, only points C'' and G' are mesh points. From Figs. 5.3(a) and (b), it is seen that (i) three of the faces of this CE, i.e., $G'G''D''D'$, $G'B'B''G''$ and $G'D'C'B'$ are subsets of the SE of point G', and (ii) the other three faces, i.e., $C''B''B'C'$, $C''C'D'D''$ and $C''D''G''B''$ are subsets of the SE of point C''. As a result, by assuming that the flux of each $\vec{h}_m^*$ ($m = 1, 2, 3, 4$) is conserved over this CE, one can impose four conditions involving only the independent marching variables at the mesh points C'' and G'. Similarly, by using the flux conservation conditions over the other two CEs associated with point C''', one can obtain eight other conditions that relate the independent marching variables at the mesh points C'', I' and K'. Using the above 12 conditions, the 12 independent marching variables, i.e., u_m, u_{mx} and u_{my}, $m = 1, 2, 3, 4$, at the mesh point C''' can be determined in terms of the independent marching variables at the mesh points G', I' and K'. Similarly, the independent marching variables at the mesh point G' (see Fig. 5.2(a)) can be determined in terms of those at the mesh points A, C and E. By considering the mesh point $C''(G')$ as a typical mesh point at a whole-integer (half-integer) time level, the reader can understand how the 2D non-dissipative Euler solver was constructed [5, 7].

The non-dissipative Euler solver is only one of several 2D solvers described in [5, 7]. The latter document includes the 2D extensions of all but one of the 1D solvers described in [6]. The only exception is the 2D extension of the 1D Navier-Stokes solver. This will be dealt with in a separate paper. Also, because of the similarity in their design, each of the 2D extensions shares with its 1D version virtually the same

fundamental characteristics. As an example, the 2D non-dissipative Euler solver is neutrally stable, explicit, and involves only two time levels during a single time step. It also preserves the forward-backward marching nature and the space-time inversion invariance property of the 2D unsteady Euler equations. These are the same properties that characterize the 1D non-dissipative Euler solver.

The discussion of the 2D Euler solvers is concluded with the following remarks:

1. Because (i) the spatial geometric structure embedded in the CE/SE 2D space-time mesh is constructed from triangles, and (ii) triangles are the simplest polygon in the 2D space, the CE/SE solvers described in [5] can easily be modified and extended to solve flow problems with complex geometries.

2. Several 2D CE/SE solvers using nonuniform mesh have been developed [18-22]. Some of the numerical results generated with these solvers will be presented in Sec. 6.

3. Recall that the 1D non-dissipative Euler solver can be constructed using an alternative perspective in which the CEs and SEs have the shape of a rhombus (see Figs. 4.7–4.9). A similar alternative perspective can also be used to construct the 2D non-dissipative Euler solver. In this construction, the 2D counterpart of the rhombus referred to above is a space-time region with 12 faces.

To visualize the region referred to in remark (c), above, consider Figs. 5.3(a) and 5.4. Let points A' and G'' be joined by a line segment $\overline{A'G''}$. Because (i) points F' and A' are at one time level, and (ii) points F'' and G'' are at another time level, for any point P on the line segment $\overline{A'G''}$ (see Fig 5.4), there is one and only one point P' on $\overline{F'F''}$ such that P and P' have the same temporal coordinates. One face of the space-time region is generated by the line segment $\overline{PP'}$ as point P moves from one end of $\overline{A'G''}$ to the other end. Note that, by its definition, the intersection of this face and a plane of constant t is a straight line. However, it can be shown that the intersection of the face and a plane of constant x (constant y) is a hyperbola on a y-t (x-t) plane. Thus, in general, this face is not a plane. It is a "hyperbolic" surface. The same conclusion also follows from the fact that (i) the face contains four points A', G'', F'' and F', and (ii) generally one can not

find a plane in a 3D space-time that contains four arbitrary points in this space-time.

Similarly, one can generate the other eleven faces which, respectively, contain the following quads of points (i) A', G'', B'', B', (ii) C', G'', B'', B', (iii) C', G'', D'', D', (iv) E', G'', D'', D', (v) E', G'', F'', F', (vi) A', G, F, F', (vii) A', G,B, B', (viii) C', G, B, B', (ix) C', G, D, D', (x) E', G, D, D' and (xi) E', G, F, F'. Because the boundary of the space-time region is formed by 12 "hyperbolic" surfaces, it may be referred to as a "hyperbolic dodecahedron".

Point G' is a mesh point at a half-integer time level. Similarly, for a point at a whole-integer time level, eg., the point C'' depicted in Fig. 5.3(b), one can construct a corresponding "hyperbolic dodecahedron". For the current space-time mesh, the hyperbolic dodecahedron associated with a mesh point at a half-integer level and that at a whole-integer level are images of each other under space-time inversion. They are also congruent to each other. Note that these hyperbolic dodecahedra are constructed such that they can fill the entire computational domain without gaps or overlaps. Each of them plays the roles of both a CE and a SE in the 2D case, just as is done by a rhombus in the 1D case.

5.2   The Basis for a 3D Euler Solver

We indicate the discretization of space-time which forms the basis for a 3D Euler solver currently under development. The extension of the CE/SE method to three spatial dimensions follows reasoning similar to that used when extending the 1D solver to the 2D case (see the previous subsection). In the 3D case, the unsteady Euler equations of a perfect gas consist of five independent equations, $m = 1, 2, 3, 4, 5$. There are *three* spatial components of the gradient of each u_m (i.e., u_{mx}, u_{my} and u_{mz}, where x, y and z are Cartesian coordinates for the 3D space). When piecewise linear variation with space and time are assumed for the numerical solution, as is done in the 1D and 2D cases, and after the differential equation is assumed valid at each mesh point, there remain *four* sets of independent marching variables at each mesh point. It follows that *four* sets of conservation conditions are required at each mesh point to construct the non-dissipative 3D solver. Hence, *four* conservation elements must be defined for each mesh point. Just as a triangle was the polygon sharing its bounding edges with three neighbors, so a tetrahedron is the polyhedron sharing its bounding surfaces with *four* neighbors.

In the 2D case, referring to Figs. 5.1 and 5.2(a), $GFAB$, $GBCD$ and $GDEF$ are the spatial projections of the CEs associated with G'. The CEs in the 3D case can be constructed in analogous fashion. Consider the tetrahedron $ABCD$ with centroid G, and the tetrahedron $ABCP$ with centroid H, depicted in Fig. 5.5. They share the face ABC. The polyhedron $GABCH$ is then defined as the spatial projection of a CE associated with a point G'. The CE is thus a right cylinder in space-time, with $GABCH$ as its spatial base. The point G is the spatial image of the mesh point G', which is displaced temporally from G by half a time step.

In similar fashion, three additional CEs associated with the mesh point G' can be constructed by considering in turn three tetrahedra that share with $ABCD$ one of its other three faces. Thus the numerical solution at G' can be determined from a knowledge of the solution at the four mesh points (one of which is H) which are the centroids of the tetrahedra sharing a face with $ABCD$. This forms the basis of a non-dissipative 3D Euler solver.

Just as the structured mesh of Fig. 5.1 is obtainable by sectioning the parallelograms of Fig. 5.1 into triangles, so it is possible to construct a structured mesh of tetrahedra by sectioning a mesh of parallelepipeds. Details of the construction will be given in a future paper.

6   Computational Examples

6.1   Shock Tube with Non-Reflecting Computational Boundaries

The CE/SE computational results for the extended shock-tube problem described in Sec. 1 are presented in this subsection.

Flow of an ideal gas with specific-heat ratio $\gamma = 1.4$ is considered in an infinite shock-tube. The initial condition, at time $t = 0$, is $(\rho, v, p) = (1, 0, 1)$ if $x < 0$, and $(\rho, v, p) = (0.125, 0, 0.1)$ if $x > 0$. Here, ρ, v, p denote the density, velocity, and pressure of the fluid, respectively. A uniform space-time mesh with $\Delta x = 0.01$ and $\Delta t = 0.004$ (corresponding to a maximum Courant number of about 0.88) is used over the computational domain defined by $-0.5 \le x \le 0.5$ and $t \ge 0$.

The numerical results are generated using the FORTRAN program listed in the Appendix. The program implements the 1D simplified shock-capturing scheme (see Sec. 4.5). The settings $\epsilon = 0.5$ and $\alpha = 1$ are used for the artificial dissipation parameters. Note that the results are obtained without the need of any local mesh-refinement techniques or any time-step tuning.

The non-reflecting boundary conditions used are (i) $\mathbf{U}_j^n = \mathbf{U}_{j-1/2}^{n-1/2}$ and $(\mathbf{U}_x)_j^n = (\mathbf{U}_x)_{j-1/2}^{n-1/2}$ if (j, n) is a mesh point on the right boundary, and (ii) $\mathbf{U}_j^n = \mathbf{U}_{j+1/2}^{n-1/2}$ and $(\mathbf{U}_x)_j^n = (\mathbf{U}_x)_{j+1/2}^{n-1/2}$ if (j, n) is a mesh point on the left boundary. The reasons why such trivial extrapolations can serve so well as non-reflecting boundary conditions in the CE/SE method are explained in a separate paper [40].

Figs. 6.1–6.3 show the numerical solution (triangular data points) compared with the analytical solution (unbroken line) at three different times, namely, $t = 0.2$, 0.4 and 0.6. It is seen that excellent agreement is obtained between the numerical results and the analytical solution. In particular, as seen in Fig. 6.1, the shock wave discontinuity is resolved almost within one mesh interval and the contact discontinuity is resolved in four mesh intervals. Fig. 6.2 shows that by $t = 0.4$, the numerically computed shock wave has passed cleanly through the right boundary, with no spurious reflections. Similarly, Fig. 6.3 shows that by $t = 0.6$, the contact discontinuity has passed through the right boundary, while the expansion region has partially passed through the left boundary. Agreement with the exact solution continues to be excellent.

6.2   Convection-Diffusion Examples

The CE/SE computations described in this subsection were originally presented in [8], where an implicit CE/SE solver for the convection-diffusion equation $u_t + au_x - \mu u_{xx} = 0$ ($\mu \geq 0$) was developed. The solver, termed the a-$\mu(I1)$ solver and previously referred to in Sec. 4.6, is an extension of the a scheme, which is the solver for $u_t + au_x = 0$ referred to at the end of Sec. 4.4. See Sec. 4.6 for a brief discussion of how the construction of the a-$\mu(I1)$ solver ensures that numerical dissipation does not overwhelm physical dissipation in large-Reynolds-number flow computations. The examples below help show that the scheme is accurate over the whole Reynolds number range, from pure diffusion

to convection-dominated solutions.

Pure Diffusion. We consider a special case of the convection-diffusion equation with $a = 0$ and $\mu = 1$, in the domain $0 \leq x \leq 1$ and $t \geq 0$. The initial/boundary conditions completing the problem specification are (i) $u(0, t) = u(1, t) = 0$ for $t \geq 0$, (ii) $u(x, 0) = 2x$ for $0 \leq x \leq 0.5$, and (iii) $u(x, 0) = 2(1 - x)$ for $0.5 \leq x \leq 1$. The solution $u(x, t)$ exhibits the diffusive decay of the initial sawtooth shape. An exact series solution is available, see for e.g. p.15 of [34]. For the CE/SE computation, uniform mesh intervals $\Delta x = 0.02$ and $\Delta t = 0.005$ are used. Fig. 6.4 shows the time-slice at $t = 0.05$, comparing numerical and exact solutions, and also showing the error scaled with the peak exact value at that time level. The maximum error magnitude is seen to be about 0.5% of the peak solution value. At $t = 1$ (not shown), when the peak solution value has dwindled to about 4×10^{-5}, the maximum error magnitude is about 0.15% of the peak solution value.

Boundary Layer, Re = 100. We next consider the problem defined for the convection-diffusion equation in the domain $0 \leq x \leq 1$ and $t \geq 0$ by the conditions (i) $u(0, t) = 0$ for $t \geq 0$, (ii) $u(1, t) = 1$ for $t \geq 0$, and (iii) $u(x, 0) = x$ for $0 \leq x \leq 1$. The 'steady-state' or time-asymptotic limit of the solution is $u(x, \infty) = [\exp(ax/\mu) - 1] / [\exp(a/\mu) - 1]$. The case $a = 1$, $\mu = 0.01$ (i.e. $Re = 100$) is considered, which leads to a steady-state boundary layer at $x = 1$. Uniform mesh intervals $\Delta x = 0.0025$ and $\Delta t = 0.002$ are used, so that the Courant number is 0.8. Fig. 6.5 shows the computed and exact steady-state limits, together with the error. The boundary layer is seen to be well resolved, with the maximum magnitude of the error being about 1% of the solution peak.

6.3   Shock Wave in a Constant-Temperature Bath

We next take up a flow problem proposed by Pember [35]. Consider the one-dimensional Euler equations with a special heat transfer term in the energy equation:

$$\mathbf{U}_t + \mathbf{F}_x + \mathbf{H} = 0, \qquad (6.1)$$

where the flow properties $\mathbf{U}$ and the flux $\mathbf{F}$ are as defined in Eqs. (2.4) and (2.5). The source term is defined as $\mathbf{H} = (0, 0, K\rho(T - T_o))^T$. The function of the source term is to force a constant tempera-

ture T_o upon the whole flow field. The equilibrium counterpart of the relaxation system is

$$\frac{\partial \rho}{\partial t} + \frac{\partial \rho v}{\partial x} = 0, \tag{6.2}$$

$$\frac{\partial \rho v}{\partial t} + \frac{\partial \rho v^2 + p^*}{\partial x} = 0, \tag{6.3}$$

where the pressure $p^*(\rho) = (\gamma - 1)\rho e_o$, with e_o as the internal energy of the gas at $T = T_o$. We conducted the calculation with 201 grid nodes over the x interval $0 \le x \le 1$. The conductivity K in the flow system was set be to 10^8, 10^{12}, and 10^{16}. The relaxation time $\epsilon = 1/K$ is under-resolved for all three cases. Essentially, we get the same result for different Ks. Figures 6.6(a)–(d) show the flow properties at $t = 0.3$. The flow field contains a right-moving shock and a left-moving rarefaction. The numerical solution is in agreement with that presented by Jin [36]. For details of the treatment of source terms in the present CE/SE method, we refer the reader to [16].

6.4 Diffraction of a Shock Wave around a Wedge

According to the experimental shadowgraph results shown in [37], when a plane shock wave of $M_s = 1.3$ is moving over the beginning of a finite wedge of semi-vertex angle $\theta = 26.565°$, an ordinary Mach reflection is generated. As the shock wave passes the base, the flow separates to form vortex sheets at the sharp corners. Further interaction produces an increasingly elaborate pattern of shock waves, slip lines and vortices.

As reported in [18, 21], this flowfield is simulated using the CE/SE Euler solver. By virtue of the symmetry in the solution, attention is restricted to the upper half of the domain. The extent of the computational domain is set based on an estimation from Fig. 522 in [37]. The shock wave is at $x = -0.5$ at $t = 0$. The numerical boundary condition imposed on the vertical wall of the wedge is described in [20]. Numerical solutions at eight time levels ($t = 0.725, 0.9075, 1.2125, 1.55, 1.825, 2.1375, 2.4875,$ and 2.9475), obtained by using two subdomains with 321×89 and 209×34 mesh points, and with $\Delta t = 0.0025$, are shown in Fig. 6.7. It should be pointed out that the upper and lower walls of the channel shown in the shadowgraphs of [37] are actually further apart than the top and bottom edges of the shadowgraphs. Therefore some flow phenomena that are seen in Fig. 6.7, in the region near the upper

wall, are lost in shadowgraphs, especially at the 4th time level. Comparisons of the computed solutions with experimental pictures of [37] have shown an excellent agreement in general flow features except for those phenomena induced by the effect of viscosity. The shock waves, slip lines and vortices are captured very well.

6.5 Implosion/Explosion of Polygonal Shock Waves in a Box

The 2-D CE/SE Euler solver has been used in [22] to solve a problem studied in [38], concerning the implosion/explosion of a polygonal shock wave in a square box. In addition to the early stage of the implosion/explosion process, the later development of the process, which was not studied in [38], is also simulated in [22]. The computation further demonstrates the robustness of the CE/SE Euler scheme in handling discontinuous flows.

A uniformly distributed 241x241 grid is utilized in the computational domain, which is a square defined by $-2 \le x \le 2$ and $-2 \le y \le 2$. The initial shock wave configuration is a polygon, the geometric center of which coincides with that of the square. Inside the polygon is a low pressure region, with a pressure ratio of 10 across the shock. The radius of the circumscribed circle of the polygon is selected to be $0.8\sqrt{3}$ for all shapes of the polygon. In the numerical scheme, the two parameters ϵ and α are set to be 0.5 and 1 respectively, everywhere in the computational domain for all cases, and the maximum Courant number is always kept at a value of 0.9.

In one set of computations, the early flowfield is studied for polygonal shock waves with initial shapes of an equilateral triangle, a square, and a pentagon. The density contour plots at different time levels are shown in Fig. 6.8. Wave patterns similar to those captured in Figs. 1–5 of [38] using a TVD method on a 359x359 grid are clearly shown in the CE/SE solutions, displaying detailed features such as Mach stems and the newly-developed smaller polygons.

In another computation, the implosion/explosion of a hexagonal shock wave is simulated until the re-implosion of the shock wave is observed in the box. More complex flow phenomena can be seen in the density contour plots of Fig. 6.9, including the reflections of shock waves, shock-shock interaction, and shock-contact surface interaction. It is interesting to note that the shape of the contact surface cen-

tered at the origin of the box remains unchanged even after the passage of shock waves.

6.6 Examples from Computational Aeroacoustics

The CE/SE computational examples we describe in the next two subsections were reported in [9] and [15]. The investigations in [9] and [15] found the CE/SE Euler scheme to be capable of handling the complete spectrum of flows, from small-amplitude linear acoustic waves, all the way to nonlinear or even discontinuous waves (shocks). Through numerical experiments in computational aeroacoustics, the following salient properties of the CE/SE Euler scheme emerge: (i)The CE/SE scheme possesses very low dispersion error and yields high resolution results comparable to that of a high order compact difference scheme, although nominally the CE/SE scheme is only of 2nd order accuracy. (ii) In general, the numerical non-reflecting boundary condition applicable to the CE/SE scheme is genuinely multi-dimensional, and can be implemented in a simple and elegant way without resorting to the complexities of characteristic forms or buffer zones. (iii) The CE/SE scheme is both a CFD (Computational Fluid Dynamics) and a CAA (Computational Aeroacoustics) scheme, capable of handling continuous and discontinuous flows. It thus represents a unique numerical technique for flows where sound waves and shocks and their interactions are important.

It is well-known that in CAA, the non-reflecting boundary condition plays a dominant role in the final numerical results. In general, there are three ways to impose the non-reflecting boundary conditions, namely,

(i) to apply 1-D characteristic variables (Riemann invariants) in the direction normal to the boundary,

(ii) to minimize spurious numerical reflections from the boundaries by inserting a buffer zone with increased numerical damping,

(iii) to apply an asymptotic analytical solution at the boundaries.

In the new CE/SE scheme, none of the above complex treatments of non-reflecting boundary conditions is needed. Instead, a simple non-reflecting

boundary condition is:

$$(u_{mx})_{j,k}^n = (u_{my})_{j,k}^n = 0, \qquad m = 1, 2, 3, 4,$$

while $(u_m)_{j,k}^n$ is defined by simple extrapolation from the interior neighbors. In general, the consequent reflection amounts to about 1% of the strength of the incident waves.

6.7 Acoustic Waves Generated by a Flat-Plate Loudspeaker

In our first example, a continuous wave problem is considered. The physical domain is a rectangle, as shown as in Fig. 6.10. A mesh of 200×200 nodes is employed over a domain of length 300 in x and 200 in y. The background mean flow is a uniform one in the x direction, with Mach number $M = 0.5$. The mean flow is given by

$$u_0 = M, \quad v_0 = 0, \quad \rho_0 = 1, \quad p_0 = \frac{1}{\gamma}, \quad \gamma = 1.4$$

where the subscript 0 identifies mean flow quantities. At the center of the inlet boundary, a small vertical flat plate of size 1/10 of the height of the domain is vibrating in the x direction, generating a perturbation to the mean flow that is given by

$$u^* = 0.01 \sin\left(\frac{\pi t}{5}\right)$$

Fig. 6.10 illustrates the isobars at time $t = 600$. For clear display, 800 uniform contour levels are used. In Fig. 6.10, the well-known acoustic directivity pattern of a flat plate loudspeaker is clearly seen, and practically no reflection at the four boundaries is observed.

6.8 Acoustic Pulse / Shock-Wave Interaction

In order to demonstrate the capability of the CE/SE scheme to handle interactions of acoustic waves and shock waves, we describe an example of a weak acoustic pulse wave passing through a strong shock. We use a mesh of 200×200 nodes. The domain is centered at the origin $(0,0)$, with an extent of $-100 \le x \le 100$ and $-100 \le y \le 100$. A steady oblique shock at a position along a diagonal of the computational domain is precalculated to form part of the initial condition for the computation. The

initial conditions of an isolated acoustic pulse are superimposed on this precalculated shock to form the initial condition of the given problem. The data upstream and downstream of the shock are respectively

$$u_0 = 2.378056, \quad v_0 = 0,$$
$$\rho_0 = 1 \quad \text{and} \quad p_0 = 0.7142857;$$

and

$$u_0 = 2.1017481, \quad v_0 = 0.4062729,$$
$$\rho_0 = 1.5807555 \quad \text{and} \quad p_0 = 1.3713613.$$

A weak acoustic pulse propagating across a strong shock is considered. An acoustic pulse, initially centered at $(x_0, y_0) = (-75, 0)$, with initial data

$$u^* = v^* = 0, \qquad p^* = \rho^* = \epsilon e^{-a\left[(x-x_0)^2 + (y-y_0)^2\right]}$$

is superimposed on the mean flow, where the initial pulse amplitude $\epsilon = 0.001$ and $a = (\ln 2)/9$. It is observed that the oblique shock strength is three orders of magnitude larger than the initial amplitude of the acoustic pulse. The pulse propagates in all directions with the speed of sound, while being carried downstream by the mean flow. During the computation, the non-reflecting boundary condition described above is enforced at all the four sides of the computational domain.

For such an interaction between a weak (linear) wave and a discontinuous wave, the theoretical exact solution is not available. However, the numerical results obtained with the CE/SE scheme demonstrate physically plausible phenomena. Fig. 6.11 illustrates the isobars at various time steps. At first, the acoustic pulse is blown downstream and propagates freely. As the pulse collides with the strong oblique shock, the shock is practically unaffected, while the acoustic pulse ring is distorted in its passage through the shock, due to different speeds of sound and flow velocities on either side of the oblique shock.

In other examples described in [9] and [15], the interactions of a strong (i.e. nonlinear) acoustic pulse, and of weak and strong vortical and entropy pulses with a strong shock were computed. Currently, the CE/SE method is being applied to benchmark problems in CAA, and has proved to be exceptionally accurate.

7 Summary and Conclusions

In the present article, we reviewed the method of space-time conservation element and solution element (the CE/SE method, for short) for the numerical solution of conservation laws. We described several CE/SE schemes for computing fluid flows, and touched upon other CE/SE schemes and extensions. Our descriptions emphasized the geometry of the space-time discretization.

We examined salient general properties of solutions to the Euler equations for time-dependent compressible flow. From this, we arrived at requirements for a scheme to be an ideal Euler solver. An ideal solver for smooth flows must be neutrally stable, explicit and two-level, and must be such that the discrete equations are invariant under space-time inversion. The CE/SE non-dissipative Euler solvers for isentropic flows meet all these requirements. In the present article, we described the non-dissipative 1D and 2D Euler solvers in terms of the conservation of piecewise linear space-time fluxes over discrete space-time volumes. Thus, given the space-time discretization, the schemes have a simple specification in terms of flux conservation. When shock waves are present in the solution, numerical dissipation must be introduced into numerical schemes in a controllable fashion, to model the irreversibility in the exact solution. We described the shock-capturing 1D Euler solver, which is a modification of the non-dissipative solver. The added numerical dissipation has a simple geometric description and a straightforward generalization to the 2D case. The Navier-Stokes solvers, not described here, reduce to the non-dissipative solvers when the physical viscosity vanishes, and hence the latter is never overwhelmed by numerical dissipation.

We contrasted the structure and properties of the CE/SE 1D Euler solvers with those typical of high-resolution finite-volume upwind schemes. The key strategies that enable the CE/SE schemes to avoid the limitations of the upwind schemes are: (i) The more general form of the conservation laws, i.e., the integral form, is cast in a form in which space and time are treated on an equal footing. This gives flexibility in the shape of the space-time conservation elements, which is useful for defining CEs when, for e.g., sources are present in the CE. (ii) A staggered space-time mesh is employed. This results in the simplest stencil. It also obviates the need for interpolation of fluxes at the interface between CEs. Thus, there is no need for an approximate Riemann

solver. Hence, characteristics-based upwind-biasing methods, which are complicated and strictly valid only for smooth solutions, are avoided. There is thus also no compromise in the symmetry of treatment of the spatial fluxes. This also has implications for flows in multiple spatial dimensions. For the computation of such flows, upwind techniques must use directional splitting with its attendant difficulties. The CE/SE method in multiple spatial dimensions, on the other hand, does not involve any directional splitting. (iii) The flow property gradient is treated as an additional unknown in the CE/SE schemes. Therefore, there is no need for reconstruction of the flow gradient by polynomial curve fitting over neighboring mesh points, and for the subsequent use of complicated flux limiters. (iv) Space-time fluxes are conserved at both the local and global level. The condition of flux conservation, rather than any extrapolation, links the solution at a mesh point with its neighbors at the previous time level. This emphasis on the integral conservation law is critical for accurate flow simulations, particularly if they involve long marching times and/or regions of rapid change (e.g., boundary layers and shocks).

We reproduced here several numerical results obtained with various CE/SE flow solvers. The results included a demonstration of extremely simple yet highly effective non-reflecting boundary conditions for the extended Sod's shock-tube problem. The CE/SE solver for the scalar convection-diffusion equation was shown to be accurate in all Reynolds number regimes. The CE/SE solver for the Euler equations with source terms representing heating/cooling was also shown to be accurate, even when the source term is of the order of 10^{16}. We reproduced numerical solutions obtained with the 2D CE/SE Euler solver, including the process of diffraction of a shock wave around a wedge and the implosion/explosion of a polygonal shock wave in a box, as well as computational aeroacoustic phenomena involving the interaction of strong shocks and weak acoustics. The results reproduced here are only some of the difficult problems readily solved with CE/SE schemes; see [1–9] and [13–22] for more examples.

We remark here that the CE/SE schemes developed thus far are characterized by simplicity, generality of applicability and second-order accuracy in space and time. The simplest possible stencils are employed. The 2D spatial mesh is constructed from triangles, and the 3D spatial mesh will be constructed from tetrahedra. Triangles and tetrahedra are the simplest polytopes in 2D and 3D, respec-

tively. The 1D and 2D Euler solvers bear a remarkable resemblance to the solvers of the 1D and 2D scalar convection-diffusion equations, respectively, with the discrete equations in the former two being matrix versions of the scalar equations in the latter two. All of the above schemes are characterized by virtually the same properties. Furthermore, the viscous flow solvers are designed to reduce to the respective non-dissipative solvers when the physical viscosity vanishes. The CE/SE method thus represents a new unified framework for the numerical solution of conservation laws. The concept of the dual space-time mesh, that was explained in Sec. 4.6, plays a key role in the development of implicit viscous solvers. It is also essential in the application of the CE/SE method to 2D unstructured meshes, when the triangles in space can not be segregated into two sets which are dual to each other. A 2D Euler solver which uses an unstructured mesh, and a 2D implicit Navier-Stokes solver are currently under development.

```fortran
c     The CE/SE 1D Euler Solver solves
c     the extended Sod's shock-tube problem
c
      implicit real*8(a-h,o-z)
      parameter (nxd=1000)
      dimension  q(3,nxd), qn(3,nxd),
     *           qx(3,nxd), qt(3,nxd),
     *           s(3,nxd), vxl(3), vxr(3),
     *           xx(nxd)
c
c     nx must be an odd integer.
      nx = 101
      it = 100
      dt = 0.4d-2
      dx = 0.1d-1
      ga = 1.4d0
      rhol = 1.d0
      ul = 0.d0
      pl = 1.d0
      cl = 1.d0
      rhor = 0.125d0
      ur = 0.d0
      pr = 0.1d0
      cr = 1.d0
      ia = 1
c
      nx1 = nx + 1
      nx2 = nx1/2
      hdt = dt/2.d0
      tt = hdt*dfloat(it)
      qdt = dt/4.d0
      hdx = dx/2.d0
      qdx = dx/4.d0
      dtx = dt/dx
      a1 = ga - 1.d0
      a2 = 3.d0 - ga
      a3 = a2/2.d0
      a4 = 1.5d0*a1
      u2l = rhol*ul
      u3l = pl/a1 + 0.5d0*rhol*ul**2
      u2r = rhor*ur
      u3r = pr/a1 + 0.5d0*rhor*ur**2
      do 5 j = 1,nx2
      q(1,j) = rhol
      q(2,j) = u2l
      q(3,j) = u3l
      q(1,nx2+j) = rhor
      q(2,nx2+j) = u2r
      q(3,nx2+j) = u3r
      do 5 i = 1,3
      qx(i,j) = 0.d0
      qx(i,nx2+j) = 0.d0
5     continue
c
      open (unit=8,file='for008')
      write (8,10) tt,it,ia,nx
      write (8,20) dt,dx,ga
      write (8,30) rhol,ul,pl,cl
      write (8,40) rhor,ur,pr,cr
c
      do 400 i = 1,it
      m = nx + i - (i/2)*2
      do 100 j = 1,m
      w2 = q(2,j)/q(1,j)
      w3 = q(3,j)/q(1,j)
      f21 = -a3*w2**2
      f22 = a2*w2
      f31 = a1*w2**3 - ga*w2*w3
      f32 = ga*w3 - a4*w2**2
      f33 = ga*w2
      qt(1,j) = -qx(2,j)
      qt(2,j) = -(f21*qx(1,j) + f22*qx(2,j)
     *          + a1*qx(3,j))
      qt(3,j) = -(f31*qx(1,j) + f32*qx(2,j)
     *          + f33*qx(3,j))
      s(1,j) = qdx*qx(1,j) + dtx*(q(2,j)
     *         + qdt*qt(2,j))
      s(2,j) = qdx*qx(2,j) + dtx*
     *         (f21*(q(1,j) + qdt*qt(1,j))
     *         + f22*(q(2,j)+ qdt*qt(2,j))
     *         + a1*(q(3,j) + qdt*qt(3,j)))
      s(3,j) = qdx*qx(3,j) + dtx*
     *         (f31*(q(1,j) + qdt*qt(1,j))
     *         + f32*(q(2,j) + qdt*qt(2,j))
     *         + f33*(q(3,j) + qdt*qt(3,j)))
100   continue
      if (i.ne.(i/2)*2) goto 150
      do 120 k = 1,3
      qx(k,1)= cl*qx(k,1)
      qx(k,nx1) = cr*qx(k,nx)
      qn(k,1) = q(k,1)
      qn(k,nx1) = q(k,nx)
120   continue
150   j1 = 1 - i + (i/2)*2
      mm = m - 1
      do 200 j = 1,mm
      do 200 k = 1,3
      qn(k,j+j1) = 0.5d0*(q(k,j) + q(k,j+1)
     *          + s(k,j) - s(k,j+1))
      vxl(k) = (qn(k,j+j1) - q(k,j)
     *        - hdt*qt(k,j))/hdx
      vxr(k) = (q(k,j+1) + hdt*qt(k,j+1)
     *        - qn(k,j+j1))/hdx
      qx(k,j+j1) =
     *      (vxl(k)*(dabs(vxr(k)))**ia
     *      + vxr(k)*(dabs(vxl(k)))**ia)/
```

```
*            ((dabs(vxl(k)))**ia
*          + (dabs(vxr(k)))**ia +1.d-60)
200   continue
      m = nx1 - i + (i/2)*2
      do 300 j = 1,m
      do 300 k = 1,3
      q(k,j) = qn(k,j)
300   continue
400   continue
c
      m = nx1 -it + (it/2)*2
      mm = m - 1
      xx(1) = -0.5d0*dx*dfloat(mm)
      do 500 j = 1,mm
      xx(j+1) = xx(j) + dx
500   continue
      do 600 j = 1,m
      x = q(2,j)/q(1,j)
      z = a1*(q(3,j) - 0.5d0*x**2*q(1,j))
      y = x/dsqrt(ga*z/q(1,j))
      write (8,50) xx(j),q(1,j),x,y,z
600   continue
c
      close (unit=8)
10    format(' t = ',f8.4,' it = ',i4,
*            ' ia = ',i4,' nx = ',i4)
20    format(' dt = ',f8.4,' dx = ',f8.4,
*            ' gamma = ',f8.4)
30    format(' rhol = ',f8.4,' ul = ',f8.4,
*            ' pl = ',f8.4,' cl = ',f8.4)
40    format(' rhor = ',f8.4,' ur = ',f8.4,
*            ' pr = ',f8.4,' cr = ',f8.4)
50    format(' x =',f8.4,' rho =',f8.4,
*            ' u =',f8.4,' M =',f8.4,
*            ' p =',f8.4)
      stop
      end
```

References

[1] S.C. Chang and W.M. To, "A New Numerical Framework for Solving Conservation Laws – The Method of Space-Time Conservation Element and Solution Element", NASA TM 104495, August 1991.

[2] S.C. Chang, "On an Origin of Numerical Diffusion: Violation of Invariance Under Space-Time Inversion", Proceedings of the 23rd Modeling and Simulation Conference, April 30 - May 1, 1992, Pittsburgh, PA, William G. Vogt and Marlin H. Mickle eds., Part 5, pp. 2727-2738. Also published as NASA TM 105776.

[3] S.C. Chang and W.M. To, "A Brief Description of a New Numerical Framework for Solving Conservation Laws – The Method of Space-Time Conservation Element and Solution Element", Proceedings of the 13th International Conference on Numerical Methods in Fluid Dynamics, July 6-10, 1992, Rome, Italy, M. Napolitano and F. Sabetta, eds. Also published as NASA TM 105757.

[4] S.C. Chang, "New Developments in the Method of Space-Time Conservation Element and Solution Element – Applications to the Euler and Navier-Stokes Equations", Presented at the Second U.S. National Congress on Computational Mechanics, August 16-18, 1993, Washington D.C. Published as NASA TM 106226.

[5] S.C. Chang, X.Y. Wang and C.Y. Chow, "New Developments in the Method of Space-Time Conservation Element and Solution Element – Applications to Two-Dimensional Time-Marching Problems", NASA TM 106758, December 1994.

[6] S.C. Chang, J. Comput. Phys., 119 (1995), pp. 295-324.

[7] S.C. Chang, X.Y. Wang and C.Y. Chow, "The Method of Space-Time Conservation Element and Solution Element – Applications to One-Dimensional and Two-Dimensional Time-Marching Flow Problems", AIAA Paper 95-1754, in A Collection of Technical Papers, 12th AIAA CFD Conference, June 19-22, 1995, San Diego, CA, pp. 1258-1291. Also published as NASA TM 106915.

[8] S.C. Chang, X.Y. Wang, C.Y. Chow and A. Himansu, "The Method of Space-Time Conservation Element and Solution Element – Development of a New Implicit Solver", Proceedings of the Ninth International Conference on Numerical Methods in Laminar and Turbulent Flow, July 10-14, 1995, Atlanta, GA. Also published as NASA TM 106897.

[9] S.C. Chang, C.Y. Loh and S.T. Yu, "Computational Aeroacoustics via a New Global Conservation Scheme", to appear in the Proceedings of the 15th International Conference on Numerical Methods in Fluid Dynamics, June 24-28, 1996, Monterey, CA.

[10] B. van Leer, J. Comput. Phys., 23 (1977), pp. 263-275.

[11] B. van Leer, *J. Comput. Phys.*, **23** (1977), pp. 276-299.

[12] G.A. Sod, *J. Comput. Phys.*, Vol. 27 (1978), pp. 1-31.

[13] C.Y. Loh, S.C. Chang, J.R. Scott and S.T. Yu, "Application of the Method of Space-Time Conservation Element and Solution Element to Aeroacoustics Problems", Proceedings of the 6th International Symposium of CFD, September 1995, Lake Tahoe, NV.

[14] C.Y. Loh, S.C. Chang, J.R. Scott and S.T. Yu, "The Space-Time Conservation Element Method – A New Numerical Scheme for Computational Aeroacoustics", AIAA Paper 96-0276, presented at the 34th AIAA Aerospace Sciences Meeting, January 15-18, 1996, Reno, NV.

[15] C.Y. Loh, S.C. Chang and J.R. Scott, "Computational Aeroacoustics via the Space-Time Conservation Element / Solution Element Method", AIAA Paper 96-1687, presented at the 2nd AIAA/CEAS Aeroacoustics Conference, May 6-8, 1996, State College, PA.

[16] S.T. Yu and S.C. Chang, "Treatments of Stiff Source Terms in Hyperbolic Conservation Systems by the Method of Space-Time Conservation Element and Solution Element", AIAA Paper 97-0435, presented at the 35th AIAA Aerospace Sciences Meeting, January 1997, Reno, NV.

[17] X.Y. Wang, C.Y. Chow and S.C. Chang, "Application of the Space-Time Conservation Element and Solution Element Method to Shock-Tube Problem", NASA TM 106806, December 1994.

[18] X.Y. Wang, "Computational Fluid Dynamics Based on the Method of Space-Time Conservation Element and Solution Element", Ph.D. Dissertation, 1995, Department of Aerospace Engineering, University of Colorado, Boulder, CO.

[19] X.Y. Wang, C.Y. Chow and S.C. Chang, "Application of the Space-Time Conservation Element and Solution Element Method to Two-Dimensional Advection-Diffusion Problems", NASA TM 106946, June 1995.

[20] X.Y. Wang, C.Y. Chow and S.C. Chang, "High Resolution Euler Solvers Based on the Space-Time Conservation Element and Solution Element Method", AIAA Paper 96-0764, presented at the 34th AIAA Aerospace Sciences Meeting, January 15-18, 1996, Reno, NV.

[21] X.Y. Wang, C.Y. Chow and S.C. Chang, "Numerical Simulation of Flows Caused by Shock-Body Interaction", AIAA Paper 96-2004, presented at the 27th AIAA Fluid Dynamics Conference, June 17-20, 1996, New Orleans, LA.

[22] X.Y. Wang, C.Y. Chow and S.C. Chang, "An Euler Solver Based on the Method of Space-Time Conservation Element and Solution Element", to appear in the Proceedings of the 15th International Conference on Numerical Methods in Fluid Dynamics, June 24-28, 1996, Monterey, CA.

[23] J.R. Scott and S.C. Chang, "A New Flux Conserving Newton's Method Scheme for the Two-Dimensional, Steady Navier-Stokes Equations", NASA TM 106160, June 1993.

[24] J.R. Scott, "A New Flux-Conserving Numerical Scheme for the Steady, Incompressible Navier-Stokes Equations", NASA TM 106520, April 1994.

[25] J. R. Scott and S.C. Chang, *Comp. Fluid Dyn.*, Vol. 5 (1995), pp. 189-212.

[26] J.R. Scott and S.C. Chang, "The Space-Time Solution Element Method – A New Numerical Approach for the Navier-Stokes Equations", AIAA Paper 95-0763, presented at the 33rd AIAA Aerospace Sciences Meeting, January 9-12, 1995, Reno, NV.

[27] J. R. Scott, "Further Development of a New, Flux-Conserving Newton Scheme for the Navier-Stokes Equations", NASA TM 107190, March 1996.

[28] P. Lax, "Hyperbolic Systems of Conservation Laws and the Mathematical Theory of Shock Waves," *SIAM*, Philadelphia, Pennsylvania (1973).

[29] H.T. Huynh, *SIAM J. Numer. Anal.*, **32**, 5 (1995), pp. 1565-1619.

[30] A. Harten, *J. Comput. Phys.*, Vol. 49 (1983), pp. 357-393.

[31] A. Harten, B. Engquist, S. Osher, and S. Chakravarthy, *J. Comput. Phys.*, Vol. 71 (1987), pp. 231-303.

[32] P.L. Roe, *Ann. Rev. Fluid Mech.*, Vol. 18 (1986), pp. 337-365.

[33] G.D. van Albada, B. van Leer, and W.W. Roberts, *Astronom. Astrophys.*, Vol. 108 (1982), p.76.

[34] G.D. Smith, *Numerical Solution of Partial Differential Equations: Finite Difference Methods*, 3rd ed., 1985. Oxford Univ. Press, New York.

[35] R.B. Pember, *SIAM J. Appl. Math.*, **53**, 5 (1993), pp. 1293-1330.

[36] S. Jin, *J. Comput. Phys.*, **122** (1995), pp. 51-67.

[37] H. Oertel, Sr., 1966 *Stossrohre*, Vienna: Springer-Verlag.

[38] T. Aki and F. Higashino, AIP Conference Proceedings 208, 1989.

[39] S.T. Yu, Y.-L.P. Tsai and K.C. Hsieh, "Simulating Waves in Flows by Runge-Kutta and Compact Difference Schemes", AIAA Journal, Vol. **33**, No. 3, pp. 421-429. Also presented as AIAA Paper 92-3210.

[40] S.C. Chang, A. Himansu, C.Y. Loh, X.Y. Wang, S.T. Yu and P.C.E. Jorgenson, "Robust and Simple Non-Reflecting Boundary Conditions for the Space-Time Conservation Element and Solution Element Method", AIAA Paper 97-2077, presented at the 13th AIAA CFD Conference, June 29-July 2, 1997, Snowmass, CO.
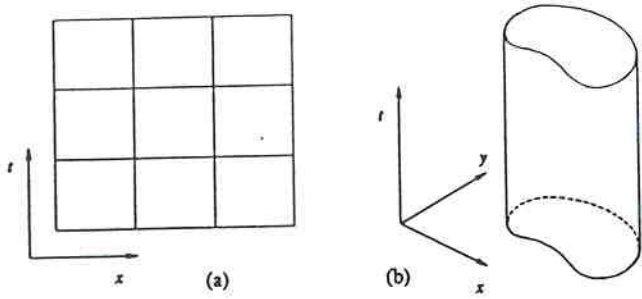
Figure 2.1 — Space-time conservation elements for methods using a fixed spatial domain: (a) one space dimension (b) two space dimensions
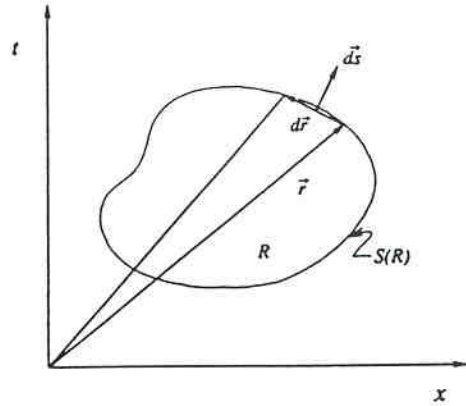
Figure 2.2 — A space-time conservation element with an arbitrary space-time domain
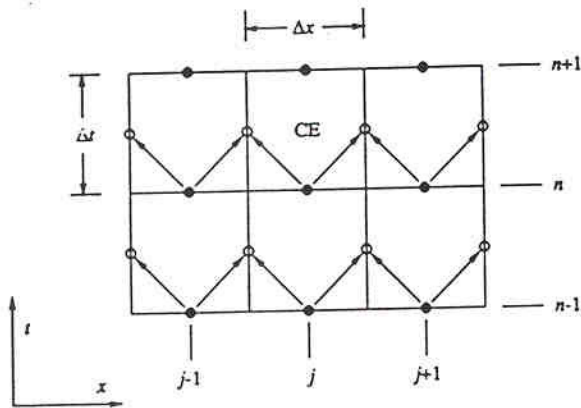
Figure 3.1 — The direction of information transfer (shown by arrows) during spatial flux estimation in an upwind scheme
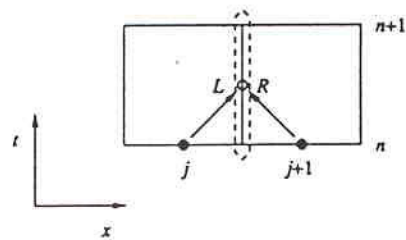
Figure 3.2 — Estimating interface solution value from left (L) and right (R), in an upwind scheme
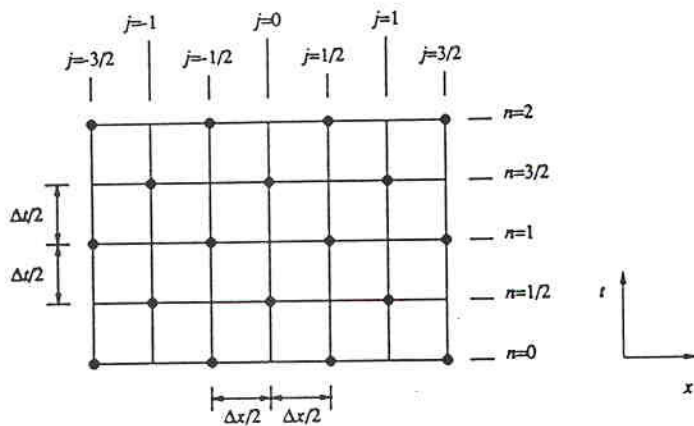
Figure 4.1 — The space-time mesh
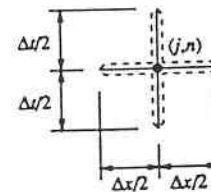
Figure 4.2 — SE(j,n)

(a) CE_(j,n)   (b) CE_+(j,n)   (c) CE(j,n)
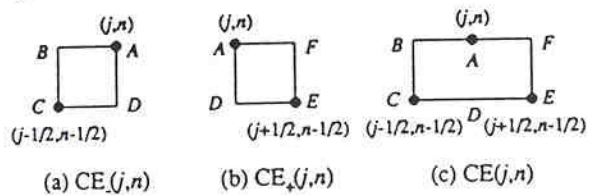
Figure 4.3 — The CEs associated with the mesh point (j,n)

28
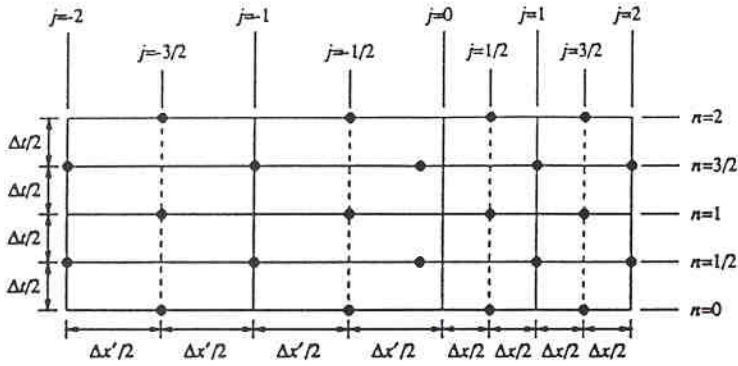
Figure 4.4 — A space-time mesh with nonuniform spatial mesh intervals
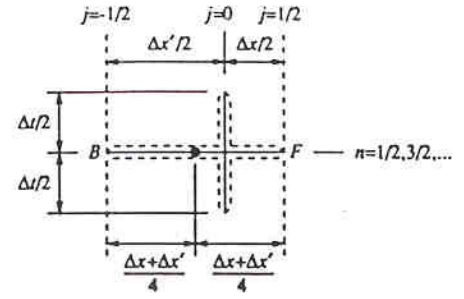
Figure 4.5 — The mesh point $(0,n)$ (denoted by a dot) and SE$(0,n)$ for $n=1/2,3/2,...$

(a) CE$_-(0,n)$ (b) CE$_+(0,n)$ (c) CE$(0,n)$
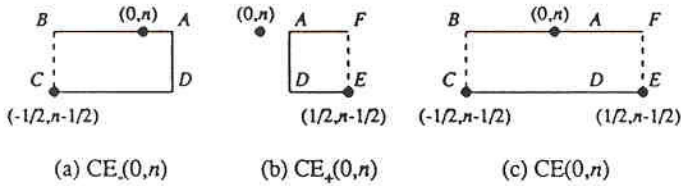
Figure 4.6 — The CEs associated with the mesh point $(0,n)$, $n=1/2,3/2,...$
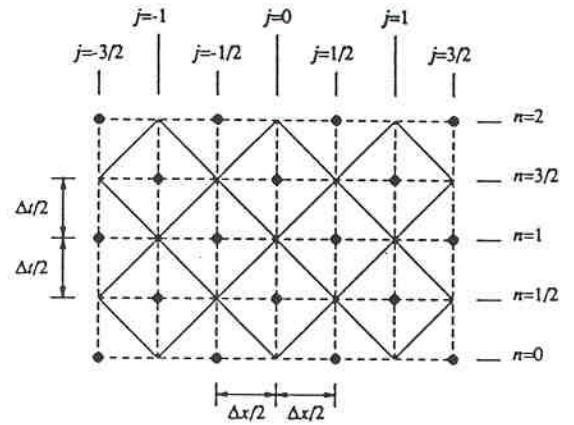
Figure 4.7 — The alternative type of SEs and CEs

(a) CE$'(j,n)$ (b) SE$'(j,n)$
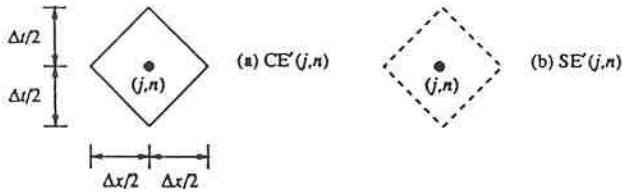
Figure 4.8 — CE$'(j,n)$ and SE$'(j,n)$ for an interior mesh point (j,n)
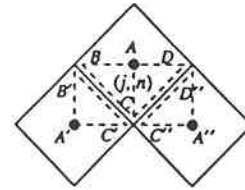
Figure 4.9 — Three neighboring CEs (SEs) of alternative type
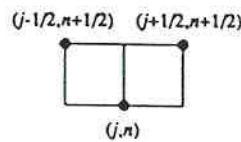
Figure 4.10 — The backward-marching conservation conditions
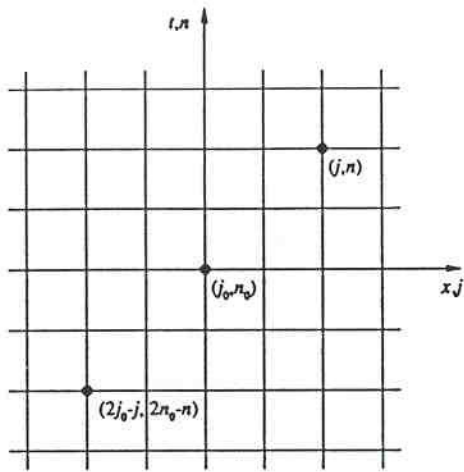
Figure 4.11 — The mesh points (j,n) and $(2j_0\text{-}j,2n_0\text{-}n)$ are images of each other under space-time inversion with respect to (j_0,n_0)

Figure 4.12 — The dual space-time mesh

Figure 4.13 — A rectangular space-time region shared by $CE_-(1/2,1/2)$ and $CE_+(0,1/2)$
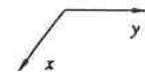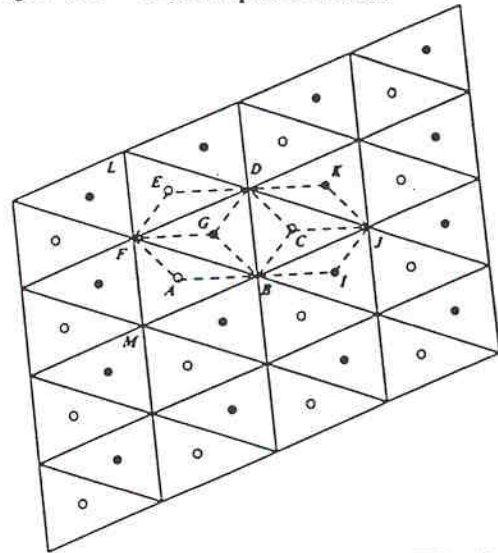
Figure 5.1 — A spatial domain formed from congruent triangles, showing the spatial projections of the mesh points
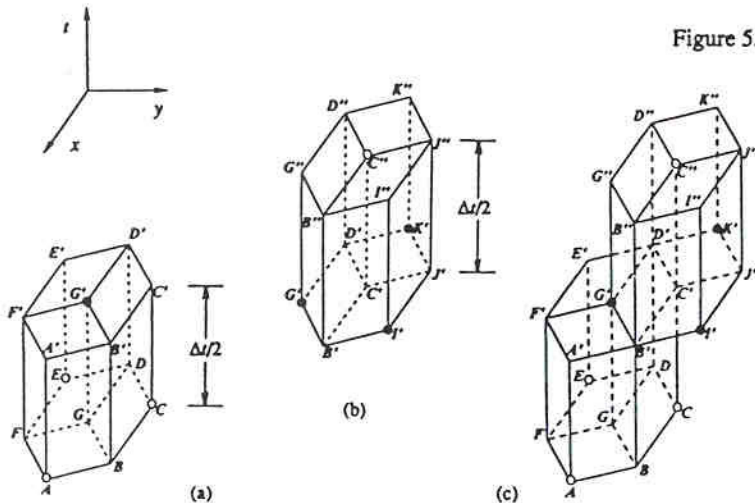
Figure 5.2 — (a) The CEs associated with G' (b) the CEs associated with C'' (c) The relative positions of the CEs of successive time steps
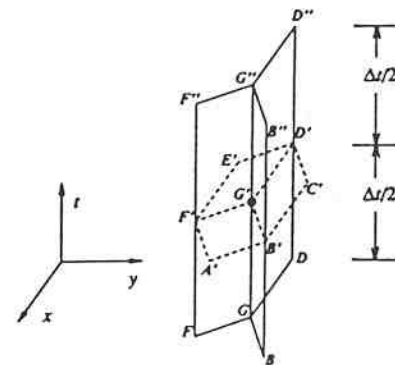
Figure 5.3(a) — The SE associated with the point G'

30
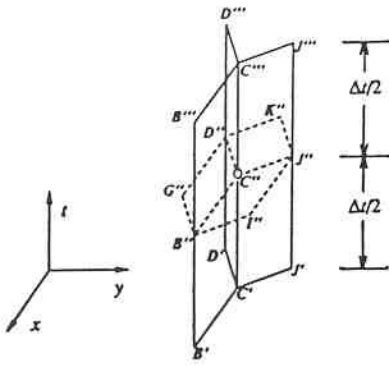
Figure 5.3(b) — The SE associated with the point C''
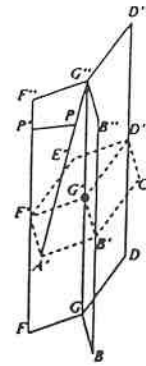
Figure 5.4 — The SE of alternative type associated with G'
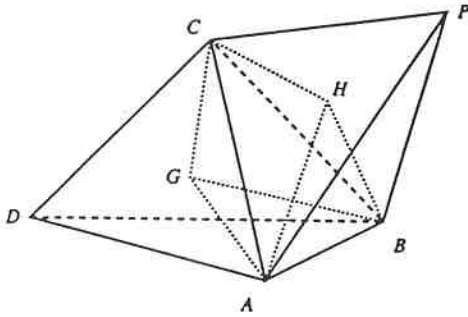
Figure 5.5 — Spatial projection of part of a 3D space-time mesh,
showing the construction of a CE

Fig. 6.1 The CE/SE solution of the extended Sod's problem at $t=0.2$
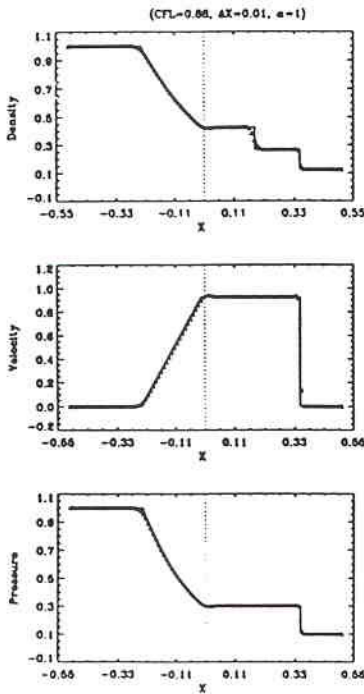
Fig. 6.2 The CE/SE solution of the extended Sod's problem at $t=0.4$

31
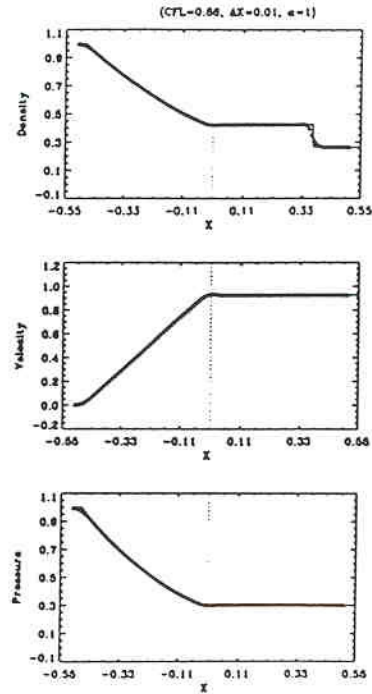
Fig. 6.3 The CE/SE solution of the extended Sod's problem at $t=0.6$

(CFL=0.66, ΔX=0.01, $a=1$)
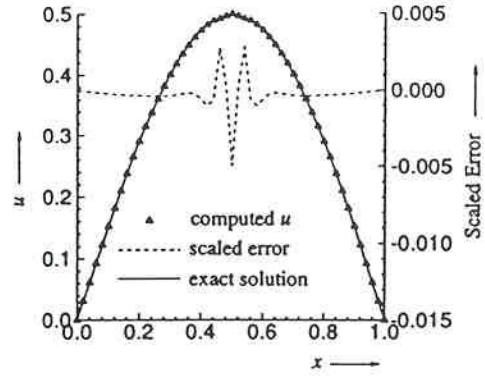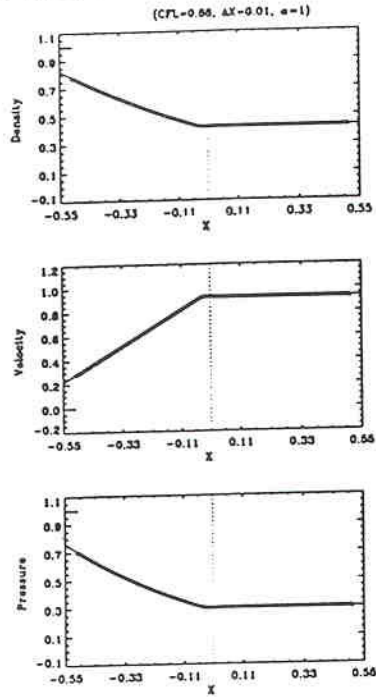
Figure 6.4 — Implicit a-μ Scheme : Pure Diffusion
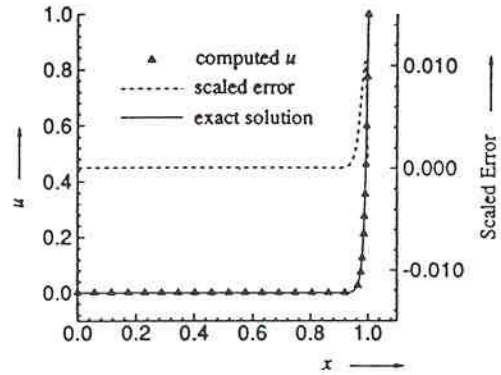
Figure 6.5 — Implicit a-μ Scheme : Boundary Layer, Re = 100
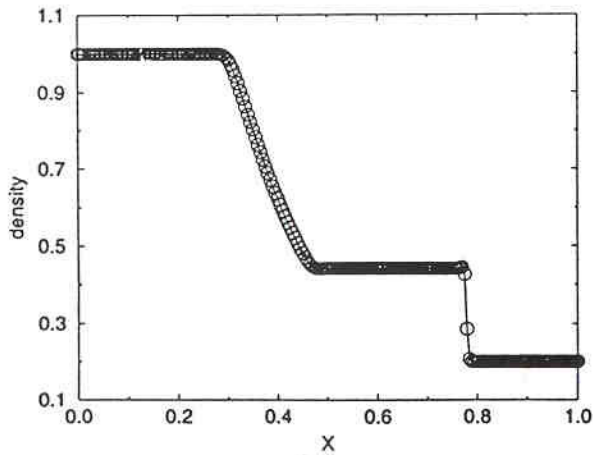
Figure 6.6(a) — Shock in a constant temperature bath with $K = 10^{16}$: density distribution at $t = 0.3$
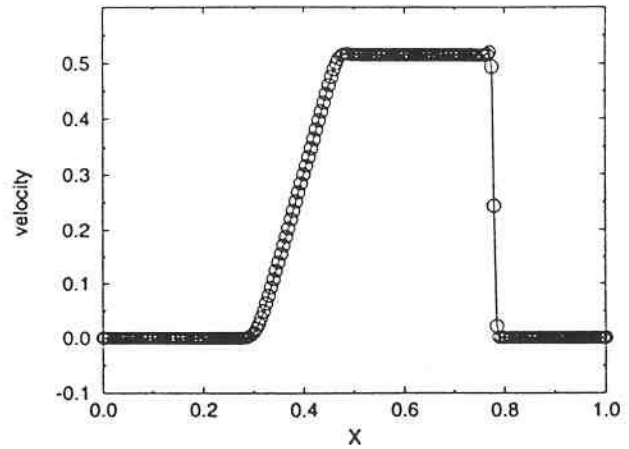
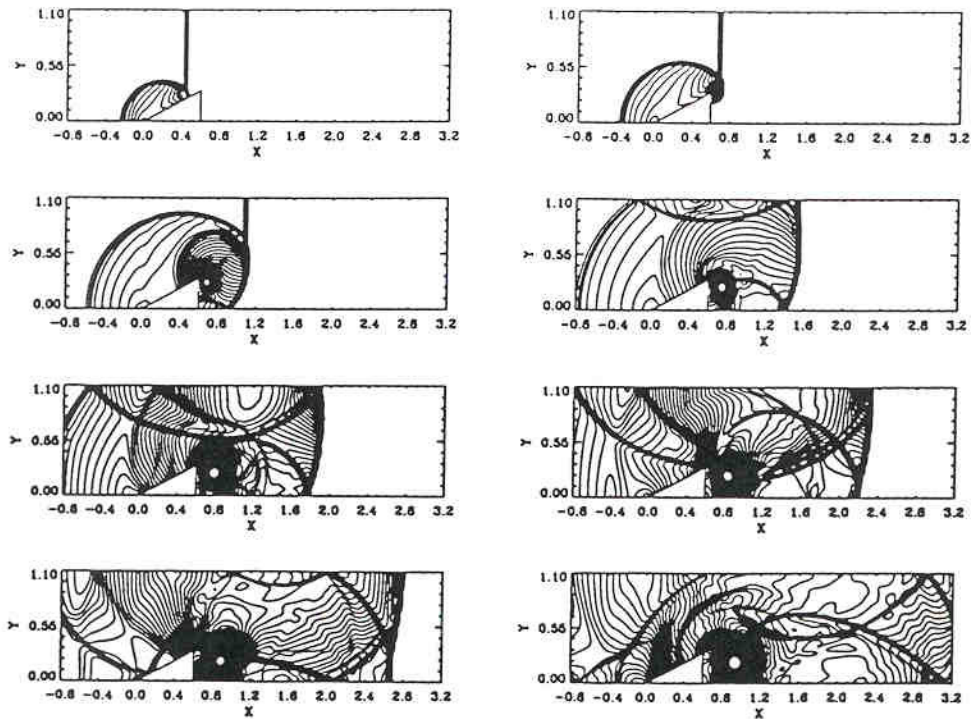Figure 6.6(b) — Shock in a constant temperature bath with $K = 10^{16}$: velocity distribution at $t = 0.3$

32
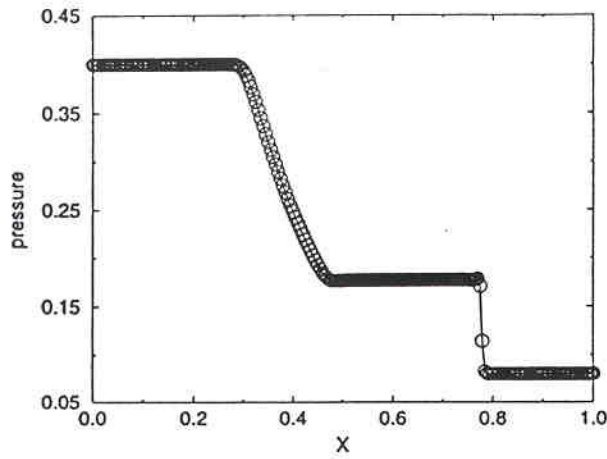
Figure 6.6(c) — Shock in a constant temperature bath with $K = 10^{16}$: pressure distribution at $t = 0.3$
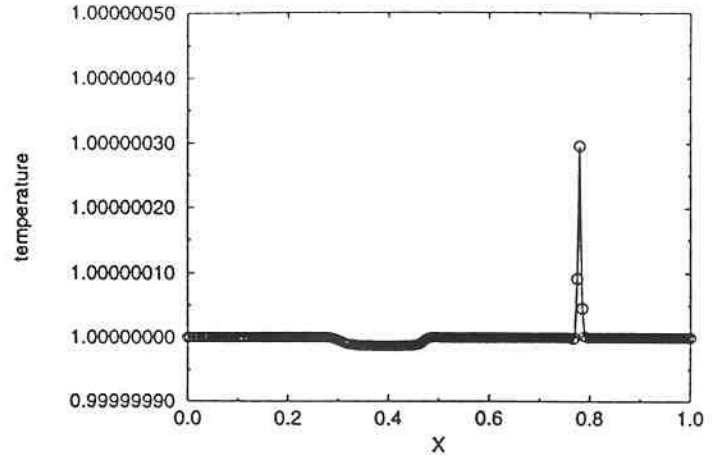
Figure 6.6(d) — Shock in a constant temperature bath with $K = 10^{16}$: temperature distribution at $t = 0.3$

Fig. 6.7 Diffraction of a shock wave around a wedge: Density contours of the CE/SE solution
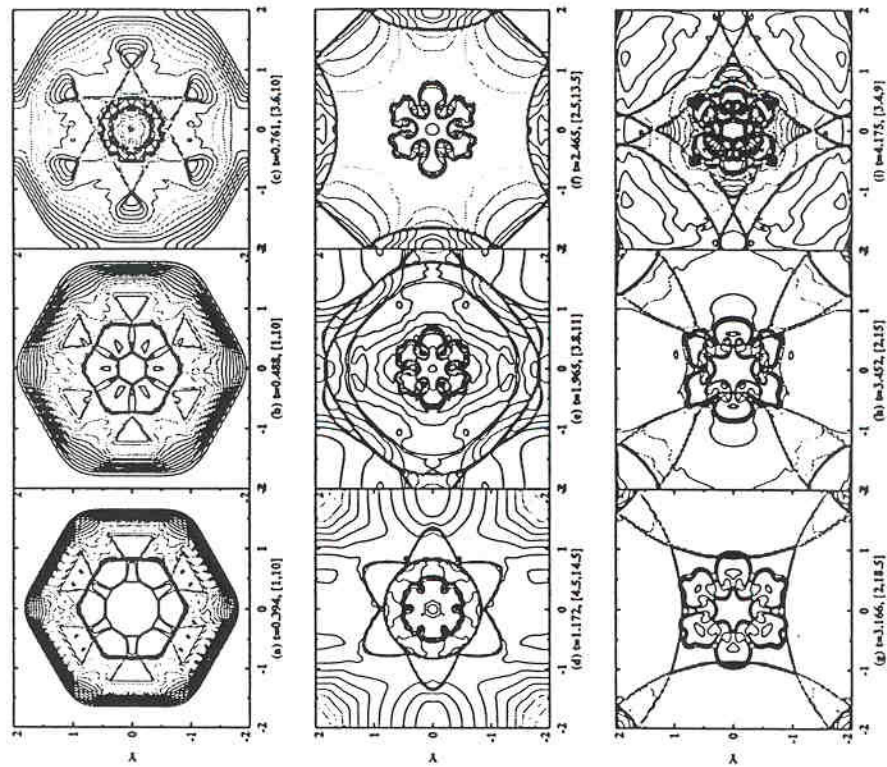
Fig. 6.9 Density contours for implosion/explosion of a hexagonal shock in a square box
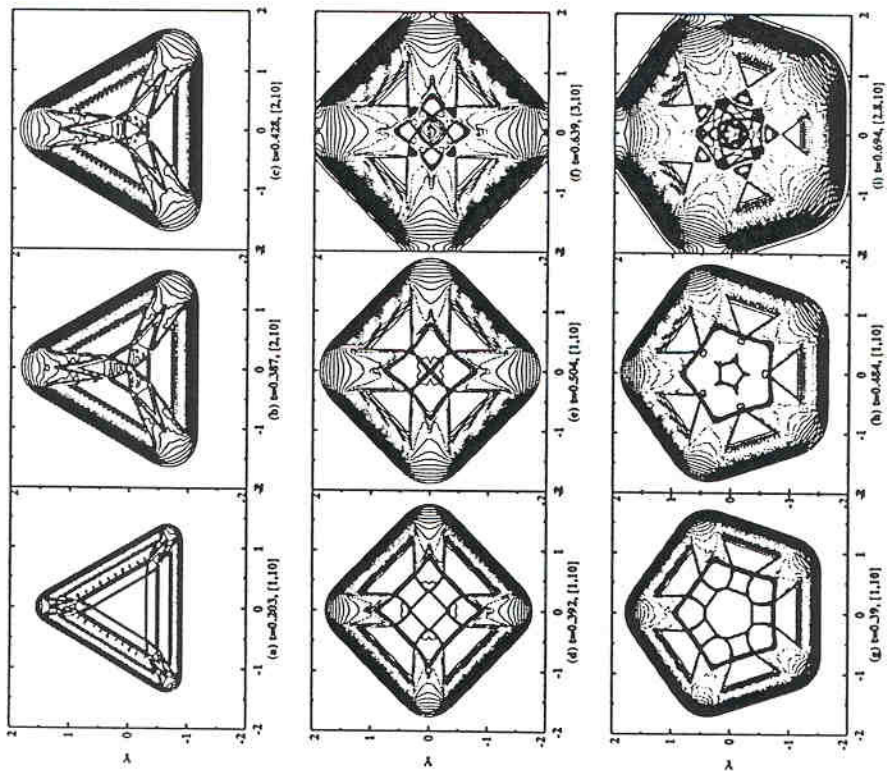

Fig. 6.8 Density contours for implosion/explosion of a polygonal shock in a square box
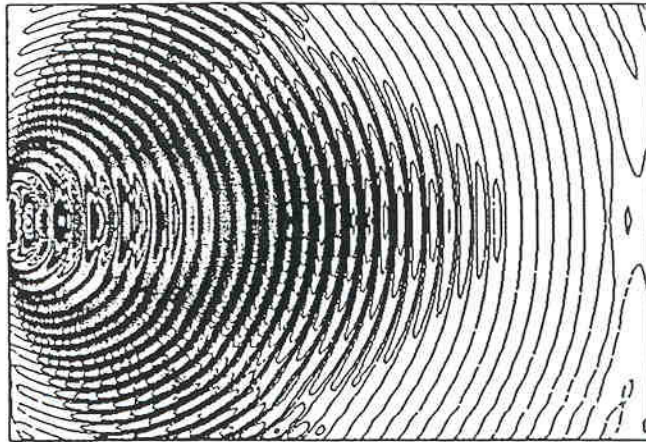
34

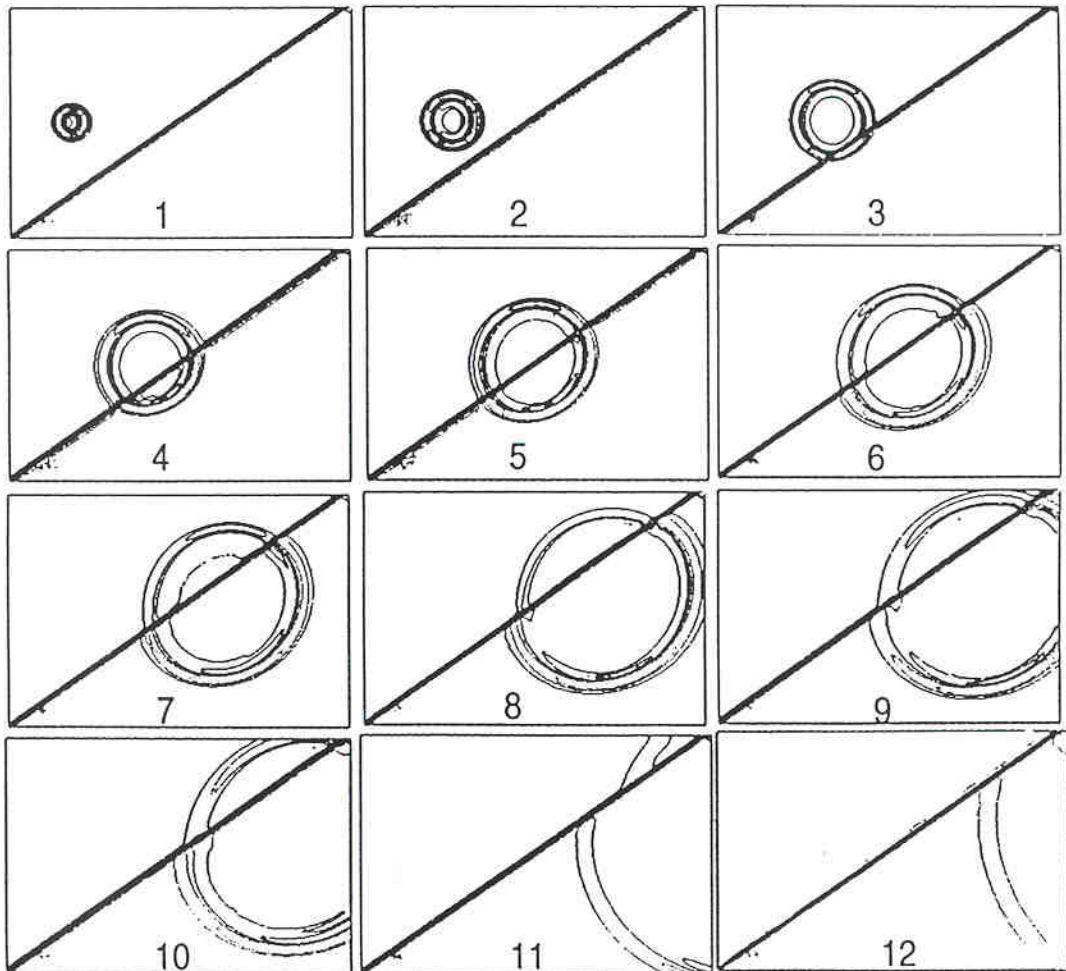Fig. 6.10 Isobars for acoustic waves generated by a flat-plate loudspeaker

Fig. 6.11 Isobars for interaction of an acoustic pulse with a shock wave