Available online at www.sciencedirect.com
SCIENCE DIRECT®
Journal of Computational Physics 189 (2003) 305–329
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

An artificially upstream flux vector splitting scheme for the Euler equations

M. Sun *, K. Takayama

Shock Wave Research Center, Institute of Fluid Science, Tohoku University, Katahira 2-1-1, Aoba, Sendai 980, Japan

Received 20 October 2002; received in revised form 2 April 2003; accepted 2 April 2003

Abstract

A new method is proposed to split the flux vector of the Euler equations by introducing two artificial wave speeds. The direction of wave propagation is adjusted by these two wave speeds. If they are set to be the fastest wave speeds in two opposite directions, the method leads to the HLL approximate Riemann solver devised by Harten, Lax and van Leer, which indicates that the HLL solver is a vector flux splitting scheme as well as a Godunov-type scheme. A more accurate scheme that resolves 1D contact discontinuity is further proposed by carefully choosing two wave speeds so that the flux vector is split to two simple flux vectors. One flux vector comes with either non-negative or non-positive eigenvalues and is easily solved by one-side differencing. Another flux vector becomes a system of two waves and one, two or three stationary discontinuities depending on the dimension of the Euler equations. Numerical flux function for multi-dimensional Euler equations is formulated for any grid system, structured or unstructured. A remarkable simplicity of the scheme is that it successfully achieves one-sided approximation for all waves without recourse to any matrix operation. Moreover, its accuracy is comparable with the exact Riemann solver. For 1D Euler equations, the scheme actually surpasses the exact solver in avoiding expansion shocks without any additional entropy fix. The scheme can exactly resolve stationary 1D contact discontinuities, and it avoids the carbuncle problem in multi-dimensional computations. The robustness of the scheme is shown in 1D test cases designed by Toro, and other 2D calculations.

1. Introduction

Upwind numerical methods are to discretize hyperbolic equations according to the direction of wave propagation. There are basically two approaches for determining upwind directions, namely the flux vector splitting (FVS) approach and the Godunov approach. A comprehensive review of the methods can be found elsewhere (e.g. [6,18]).

The Godunov-type approach uses either exact or approximate Riemann solutions between two adjacent states to calculate the flux through the interface between them. Most Godunov-type schemes have been

* Corresponding author. Tel.: +81-22-217-5285; fax: +81-22-217-5324.
E-mail addresses: sun@ceres.ifs.tohoku.ac.jp (M. Sun), takayama@ifs.tohoku.ac.jp (K. Takayama).

0021-9991/03/$ - see front matter
doi:10.1016/S0021-9991(03)00212-2

proven to be robust and accurate for the 1D Euler equations. For multi-dimensional Euler equations, the extension based on 1D Riemann solver, which obviously neglects the shear wave that only exists in multi-dimensions, contains a large amount of empiricism and must therefore remain suspect, although these schemes have been successfully applied to many practical problems. Quirk [12] reported that many Godunov-type schemes contain subtle flaws that can cause spurious solutions, all in two dimensions. Carbuncle instability is one of the flaws, and has been extensively investigated. It was noticed that the Riemann solvers that explicitly take into account the presence of a contact surface, suffer from the carbuncle instability, while the FVS schemes exhibit absence of the instability [11].

The FVS approach achieves upwinding by decomposing the flux vector into positive and negative components according to the sign of eigenvalues of the coefficient matrix. The identification of upwind directions is done with less effort than in the Godunov-type methods. The upwinding is also realized for multi-dimensional flows. However, as compared with Godunov-type schemes, the FVS results in poorer resolution of discontinuity, particularly contact discontinuity. Most upwind schemes, either Godunov-type or FVS methods, have difficulties in resolving the sonic point, and produce a spurious expansion shock there.

In this paper, a new method to construct simple, robust and accurate upwind schemes is proposed for solving the Euler equations. In view of the merits and demerits of Godunov-type and FVS approaches, we are excited about the method that overcomes all above-mentioned demerits. The paper is organized as follows. Section 2 introduces some preliminary knowledge related to this paper. Section 3 is devoted to the new scheme. The basic idea is first described, and then the equivalence of the scheme in a simple case to the HLL Riemann solver is shown in Section 3.1. Most attention is paid on the scheme that may resolve all waves in the Euler equations in Section 3.2. Multi-dimensional extension is shown in Section 3.3. High-order extension of scheme is discussed in Section 4. Extensive numerical tests are given in Section 5. Finally, the scheme is summarized in Section 6.

2. Preliminaries

Consider the one-dimensional system of conservation laws for ideal-gas flows

$$\mathbf{U}_t + \mathbf{F}_x = 0, \tag{1}$$

where $\mathbf{U}, \mathbf{F}$ are vectors of conservative quantities and fluxes, respectively,

$$\mathbf{U} = \begin{pmatrix} \rho \\ \rho u \\ \rho e \end{pmatrix}, \quad \mathbf{F} = \begin{pmatrix} \rho u \\ \rho uu + p \\ \rho eu + pu \end{pmatrix}. \tag{2}$$

The flux vector can be rewritten as

$$\mathbf{F} = u\mathbf{U} + \mathbf{P}, \tag{3}$$

where $\mathbf{P} = (0, p, pu)^{\mathrm{T}}$. Flux vector (3) is generally used in this paper.

For the numerical solution of (1), we shall consider a conservative scheme

$$\mathbf{U}_i^{n+1} = \mathbf{U}_i^n - \lambda(\hat{\mathbf{F}}_{i+1/2} - \hat{\mathbf{F}}_{i-1/2}), \tag{4}$$

where $\lambda = \Delta t/\Delta x$ is the ratio of time step Δt to grid size Δx. $\hat{\mathbf{F}}_{i+1/2}$ is often referred to as *numerical flux*. The numerical flux vector is given by two neighboring cells or left and right cells,

$$\hat{\mathbf{F}}(\mathbf{U}_{i+1/2}) = \hat{\mathbf{F}}(\mathbf{U}_i, \mathbf{U}_{i+1}) = \hat{\mathbf{F}}(\mathbf{U}^{\mathrm{L}}, \mathbf{U}^{\mathrm{R}}), \tag{5}$$

and it must satisfy the consistency condition

$$\hat{\mathbf{F}}(\mathbf{U},\mathbf{U}) = \mathbf{F}(\mathbf{U}). \tag{6}$$

System of equations (1) can be expressed in a quasi-linear form

$$\mathbf{U}_t + \mathbf{A}\mathbf{U}_x = 0, \tag{7}$$

where matrix $\mathbf{A} = \partial\mathbf{F}/\partial\mathbf{U}$. If the eigenvalues of $\mathbf{A}$ are real and the eigenvectors exist, the system of equations is hyperbolic. Denote $\mathbf{R}$ as the matrix whose columns are right eigenvectors of $\mathbf{A}$. Then one has

$$\mathbf{A} = \mathbf{R}\Lambda\mathbf{R}^{-1},$$

where Λ is the matrix of eigenvalues. The diagonal matrix of eigenvalues Λ is

$$\Lambda = \text{diagonal}(u - c,\ u,\ u + c). \tag{8}$$

3. Artificially upstream flux vector splitting (AUFS)

The fundamental idea of the present scheme is to split flux vector (3) as following:

$$\mathbf{F} = (1 - M)[(u - s_1)\mathbf{U} + \mathbf{P}] + M[(u - s_2)\mathbf{U} + \mathbf{P}], \tag{9}$$

where s_1, s_2 are two scalar constants, and it is easy to show that (9) is exactly (3) if

$$M = \frac{s_1}{s_1 - s_2}. \tag{10}$$

For the sake of clarity, flux vector (9) is rewritten as

$$\mathbf{F} = (1 - M)\mathbf{F_1} + M\mathbf{F_2}, \tag{11}$$

where two flux vectors $\mathbf{F_1}$ and $\mathbf{F_2}$ have a similar form

$$\mathbf{F}_{1,2} = (u - s)\mathbf{U} + \mathbf{P}, \tag{12}$$

where s corresponds to s_1, s_2 for $\mathbf{F_1}$ and $\mathbf{F_2}$, respectively. Therefore, these two flux vectors are different from the original $\mathbf{F}$ by a term $-s\mathbf{U}$. Their Jacobian matrixes become

$$\mathbf{A}_{1,2} = \frac{\partial\mathbf{F}_{1,2}}{\partial\mathbf{U}} = \mathbf{A} - s\mathbf{I},$$

and obviously their corresponding matrixes of eigenvalues become

$$\Lambda_{1,2} = \text{diagonal}\ (u - s - c,\ u - s,\ u - s + c). \tag{13}$$

Upwind schemes are to discretize the Euler equations basically according to the direction of wave propagation, or the sign of the eigenvalues. An excellent merit of the present splitting is that we can change the eigenvalues in (13) by varying the scalar value s, an *artificially* introduced wave speed. We may choose some appropriate values of s_1, s_2 to simplify the upwinding discretization for the Euler equations. It is emphasized that for any constant s, the corresponding flux vector can be evaluated by (12) without recourse to any matrix operation, and this property greatly simplifies the scheme. In this paper, we will discuss two ways to choose constants s_1, s_2 in two following sections.

The present idea of introducing two wave speeds was somehow stimulated by the work of Sokolov et al. [15], in which they noticed that the direction of wave propagation is relative to the frame of observation.

3.1. The AUFS method that is equivalent to the HLL Riemann solver

An intuitive choice of s_1 and s_2 is to set

$$s_1 \geqslant u + c, \quad s_2 \leqslant u - c,$$

such that eigenvalues in (13) are either non-positive or non-negative. Then one gets the numerical flux vectors, according to the wave direction

$$\hat{\mathbf{F}}_1 = \mathbf{U}^{\mathrm{R}}(u^{\mathrm{R}} - s_1) + \mathbf{P}^{\mathrm{R}}, \quad \hat{\mathbf{F}}_2 = \mathbf{U}^{\mathrm{L}}(u^{\mathrm{L}} - s_2) + \mathbf{P}^{\mathrm{L}}, \tag{14}$$

since all eigenvalues of $\mathbf{\Lambda}_1$ are non-positive, and those of $\mathbf{\Lambda}_2$ are non-negative. Only subsonic flows are considered here, and one-side differencing of supersonic flows will be readily taken into account in evaluating wave speed s_1 and s_2. Substituting (14) to (11), one obtains the intercell flux

$$\hat{\mathbf{F}} = -\frac{s_2}{s_1 - s_2}[\mathbf{U}^{\mathrm{R}}(u - s_1) + \mathbf{P}^{\mathrm{R}}] + \frac{s_1}{s_1 - s_2}[\mathbf{U}^{\mathrm{L}}(u - s_2) + \mathbf{P}^{\mathrm{L}}].$$

Rearranging the terms on the right-hand side, and using (3), one gets

$$\hat{\mathbf{F}} = \frac{s_1\mathbf{F}^{\mathrm{L}} - s_2\mathbf{F}^{\mathrm{R}}}{s_1 - s_2} + \frac{s_1 s_2}{s_1 - s_2}(\mathbf{U}^{\mathrm{R}} - \mathbf{U}^{\mathrm{L}}), \tag{15}$$

which is identical to the HLL solver theoretically proposed by Harten et al. [6]. The HLL solver is made practical by Davis [2] and Einfeldt [3]. They devised the flux following the Godunov approach, and assuming a wave configuration that consists of two waves separating three constant states. The HLL solver is an approximate Godunov-type scheme, and we show here that the solver is a vector flux splitting scheme as well. Interestingly enough the solver, like a flux vector splitting scheme, does not produce carbunlce phenomenon [11].

Physical interpretation of the HLL solver is sketched in Fig. 1, which is conceptually different from the original wave configuration considered by Harten et al. [6]. In subsonic flows, the Euler equations contain left and right traveling waves on the laboratory frame, which makes upwinding difficult. However, all waves can propagate in the same direction, observed from the frame moving at a speed of s_1 or s_2. A limit case is shown in Fig. 1, in which one wave speed becomes zero. One-sided approximation can be easily done for these unidirectional waves on the two moving frames.

In order to determine completely the numerical fluxes in (15) we need to provide an algorithm for computing those two wave speeds. A direct wave speed estimate gives, suggested by Davis [2]

$$s_1 = \max(0, u^{\mathrm{L}} + c^{\mathrm{L}}, u^{\mathrm{R}} + c^{\mathrm{R}}), \quad s_2 = \min(0, u^{\mathrm{L}} - c^{\mathrm{L}}, u^{\mathrm{R}} - c^{\mathrm{R}}), \tag{16}$$

where zero term is added to guarantee one-side differencing for supersonic flows. A more sophisticated estimate is based on the isentropic equations of gas dynamics presented by Toro et al. [19]. The exact solution for the speed u^* and the sound speed c^* between two isentropic waves can be found,

$$\begin{aligned} u^* &= \tfrac{1}{2}(u^{\mathrm{L}} + u^{\mathrm{R}}) + \tfrac{c^{\mathrm{L}} - c^{\mathrm{R}}}{\gamma - 1}, \\ c^* &= \tfrac{1}{2}(c^{\mathrm{L}} + c^{\mathrm{R}}) + \tfrac{1}{4}(\gamma - 1)(u^{\mathrm{L}} - u^{\mathrm{R}}), \end{aligned} \tag{17}$$

and then we choose

$$s_1 = \max(0, u^* + c^*, u^{\mathrm{R}} + c^{\mathrm{R}}), \quad s_2 = \min(0, u^{\mathrm{L}} - c^{\mathrm{L}}, u^* - c^*). \tag{18}$$

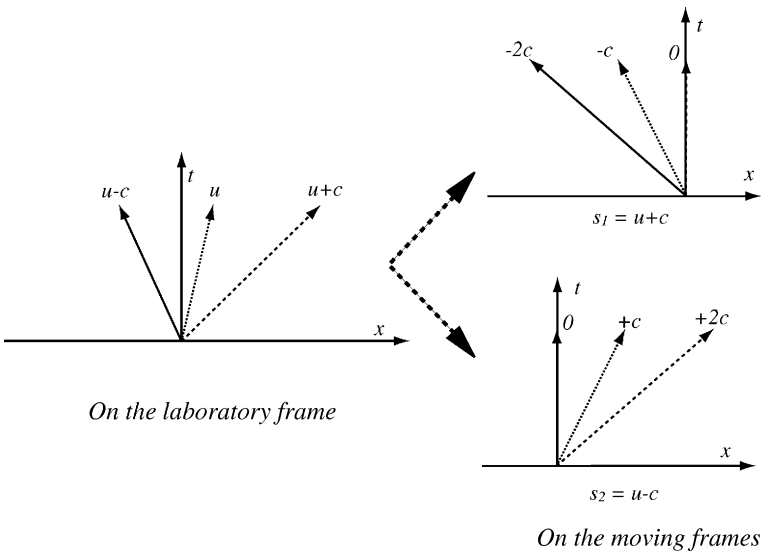

Fig. 1. Interpretation of the HLL solver from the viewpoint of AUFS.

The HLL solver is very efficient and robust for many inviscid applications, but adds excessive diffusion around contact discontinuities. To overcome the shortcoming, Toro et al. [19] proposed the HLLC solver (C stands for contact) by introducing a contact in the two-wave configuration of the HLL solver. Although the HLLC solver improves the accuracy around contact discontinuities, it encounters the problem of carbuncle instability as other Riemann solvers that explicitly resolve contact discontinuity [11] , and kinked Mach stems in shock reflection that will be shown in Section 5. In fact, it is this problem of kinked shock waves that forces us to find a more accurate and robust scheme.

Since this simple version of the AUFS method is the same as the HLL solver, its property and numerical performance will not be repeated in this paper. A comprehensive review of the HLL solver can be found in [18]. The AUFS method, hereinafter, is referred to as the method that can resolve 1D contact discontinuities and remove the carbuncle instability, to be presented in the following sections.

3.2. The AUFS method that resolves contact discontinuities

The AUFS method provides a framework for the design of numerical schemes by choosing two artificial wave speeds. In the previous section, we choose such wave speeds that the original flux vector of the Euler equations can be split to two vectors that contain only unidirectional waves. The fact that the derived scheme is equivalent to the robust HLL Riemann solver suggests the feasibility and reliability of the AUFS method. In order to resolve contact discontinuities, we propose another way to choose the wave speeds in this section.

Consider

$$s_1 = u,$$
$$s_2 = \begin{cases} u - c & \text{for } s_1 > 0, \\ u + c & \text{for } s_1 \leqslant 0, \end{cases} \tag{19}$$

and then two split flux vectors become

$$\mathbf{F}_1 = \mathbf{P}, \tag{20}$$

which consists of only pressure and velocity terms, and

$$\mathbf{F}_2 = \mathbf{U}(u - s_2) + \mathbf{P}. \tag{21}$$

The eigenvalues of the Jacobian matrix of flux vector $\mathbf{F}_1$ become $(-c, 0, +c)$, which are two isentropic waves and one stationary entropy wave. Following the Steger–Warming approach or formula (15), one gets

$$\hat{\mathbf{F}}_1 = \frac{1}{2}(\mathbf{P}^{\mathrm{L}} + \mathbf{P}^{\mathrm{R}}) + \delta\mathbf{U}. \tag{22}$$

Definition of $\delta\mathbf{U}$ will be given in (28) or (34).

The Jacobian matrix of the second flux vector $\mathbf{F}_2$ still maintains the property that eigenvalues are either all non-negative or all non-positive, so it can be easily discretized by one-side differencing, depending upon the sign of s_1,

$$\hat{\mathbf{F}}_2 = \mathbf{U}^{\alpha}(u^{\alpha} - s_2) + \mathbf{P}^{\alpha}, \tag{23}$$

where

$$\alpha = \begin{cases} L & \text{for } s_1 > 0, \\ R & \text{for } s_1 \leqslant 0. \end{cases} \tag{24}$$

Substituting (22) and (23) to (11), one obtains the final intercell flux

$$\hat{\mathbf{F}} = (1 - M)\left[\frac{1}{2}(\mathbf{P}^{\mathrm{L}} + \mathbf{P}^{\mathrm{R}}) + \delta\mathbf{U}\right] + M\left[\mathbf{U}^{\alpha}(u^{\alpha} - s_2) + \mathbf{P}^{\alpha}\right]. \tag{25}$$

Unlike the HLL solver that splits the Euler equations to two systems of three unidirectional waves, the AUFS scheme retains one system of the unidirectional waves in $\mathbf{F}_2$, and introduces another system of two symmetric sound waves and a stationary contact wave, $\mathbf{F}_1$, as sketched in Fig. 2. In multi-dimensions,

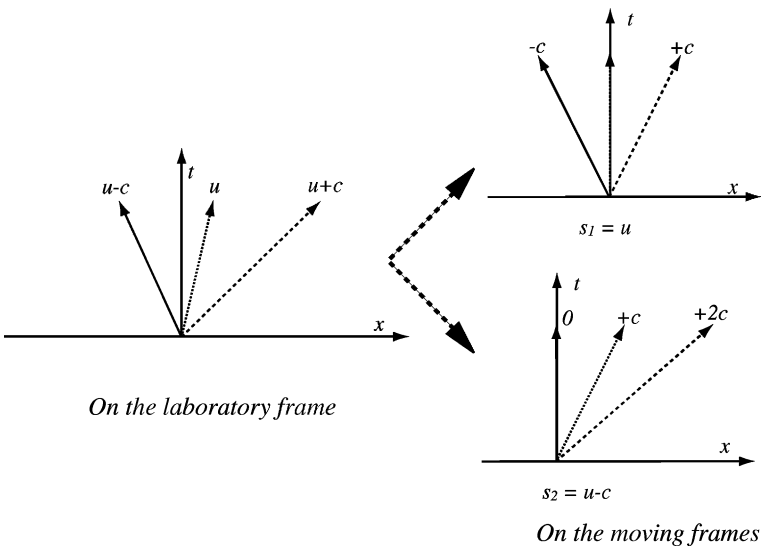

Fig. 2. Physical interpretation of the AUFS scheme.

although there are more than three waves, the Euler equations have only three wave speeds. The present splitting is therefore also valid for multi-dimensions.

The second term on the right-hand side of (22), $\delta\mathbf{U}$, represents the artificial viscosity. The performance of the scheme (25) should rely on the way to define $\delta\mathbf{U}$. A few possible choices for the discretization of $\delta\mathbf{U}$ are discussed here. First, we consider the Steger–Warming approach [16]. Since the eigenvalues of the Jacobian matrix of flux vector $\mathbf{F}_1$ becomes $(-c, 0, +c)$, let $\mathbf{\Lambda}_1^+ = \text{diagonal}(0, 0, +c^{\mathrm{L}})$, $\mathbf{\Lambda}_1^- = \text{diagonal}(-c^{\mathrm{R}}, 0, 0)$, one gets

$$\hat{\mathbf{F}}_1^{\mathrm{SW}} = \mathbf{R}^{\mathrm{L}}\mathbf{\Lambda}_1^+(\mathbf{R}^{\mathrm{L}})^{-1}\mathbf{U}^{\mathrm{L}} + \mathbf{R}^{\mathrm{R}}\mathbf{\Lambda}_1^-(\mathbf{R}^{\mathrm{R}})^{-1}\mathbf{U}^{\mathrm{R}}. \tag{26}$$

After some calculations, it becomes

$$\hat{\mathbf{F}}_1^{\mathrm{SW}} = \frac{1}{2}(\mathbf{P}^{\mathrm{L}} + \mathbf{P}^{\mathrm{R}}) + \frac{1}{2c^{\mathrm{L}}}\begin{pmatrix} p^{\mathrm{L}} \\ u^{\mathrm{L}}p^{\mathrm{L}} \\ H^{\mathrm{L}}p^{\mathrm{L}} \end{pmatrix} - \frac{1}{2c^{\mathrm{R}}}\begin{pmatrix} p^{\mathrm{R}} \\ u^{\mathrm{R}}p^{\mathrm{R}} \\ H^{\mathrm{R}}p^{\mathrm{R}} \end{pmatrix}, \tag{27}$$

where $H = (c^2/\gamma - 1) + \frac{1}{2}u^2$. Last two terms on the right-hand side of (27) correspond to $\delta\mathbf{U}$ in (22). It is seen that they introduce artificial diffusion around a contact discontinuity due to the difference of sound speeds. Replacing c^{L} and c^{R} by an intermediate sound speed $\bar{c}$, one gets that the last two terms become

$$\delta\mathbf{U}^{\mathrm{SW}} = \frac{1}{2\bar{c}}\begin{pmatrix} p^{\mathrm{L}} - p^{\mathrm{R}} \\ (pu)^{\mathrm{L}} - (pu)^{\mathrm{R}} \\ \frac{\bar{c}^2}{\gamma-1}(p^{\mathrm{L}} - p^{\mathrm{R}}) + \frac{1}{2}[(pu^2)^{\mathrm{L}} - (pu^2)^{\mathrm{R}}] \end{pmatrix}, \tag{28}$$

where $\delta\mathbf{U}^{\mathrm{SW}}$ denotes that (28) is derived following the Steger–Warming approach.

Interestingly enough (28) can also be obtained by using the HLL approximation (15) enforced by an isentropic condition. The eigenvalues of the Jacobian matrix of flux vector $\mathbf{F}_1$ becomes $(-c, 0, +c)$. Let $s_1 = c$ and $s_2 = -c$, substituting them to (15), one gets a formulation similar to (22)

$$\hat{\mathbf{F}}_1^{\mathrm{HLL}} = \frac{1}{2}(\mathbf{P}^{\mathrm{L}} + \mathbf{P}^{\mathrm{R}}) + \delta\mathbf{U}^{\mathrm{HLL}}, \tag{29}$$

where $\delta\mathbf{U}^{\mathrm{HLL}} = \frac{c}{2}(\mathbf{U}^{\mathrm{L}} - \mathbf{U}^{\mathrm{R}})$, namely

$$\delta\mathbf{U}^{\mathrm{HLL}} = \frac{c}{2}\begin{pmatrix} \rho^{\mathrm{L}} - \rho^{\mathrm{R}} \\ (\rho u)^{\mathrm{L}} - (\rho u)^{\mathrm{R}} \\ (\rho e)^{\mathrm{L}} - (\rho e)^{\mathrm{R}} \end{pmatrix}. \tag{30}$$

Since the entropy wave moves at a speed of zero, which is a stationary contact discontinuity, no artificial viscosity is required. In order to capture a stationary contact precisely, for $\delta p = 0$ and $\delta u = 0$, the artificial viscosity $\delta\mathbf{U}$ must be zero. However (30) introduces additional diffusion due to the density difference between left and right states. On the other hand, flux vector $\mathbf{F}_1$ in (20) contains only pressure and velocity terms, so density ρ should not appear in its numerical flux. If the density terms in (30) are removed by enforcing isentropic relations

$$\delta\rho = \frac{\delta p}{\bar{c}^2}, \quad \rho = \frac{p}{\bar{c}^2}, \tag{31}$$

one immediately gets less diffusive artificial viscosity, $\delta\mathbf{U}^{\mathrm{HLLI}}$, which is identical to that derived following the Steger–Warming approach (28):

$$\delta\mathbf{U}^{\mathrm{HLLI}} = \delta\mathbf{U}^{\mathrm{SW}}. \tag{32}$$

If the density terms in (30) are removed by enforcing isothermal relations instead of isentropic ones,

$$\delta\rho = \frac{\gamma\delta p}{\overline{c}^2}, \quad \rho = \frac{\gamma p}{\overline{c}^2}, \tag{33}$$

one gets

$$\delta\mathbf{U}^{\mathrm{HLLT}} = \frac{\gamma}{2\overline{c}} \begin{pmatrix} p^{\mathrm{L}} - p^{\mathrm{R}} \\ (pu)^{\mathrm{L}} - (pu)^{\mathrm{R}} \\ \frac{\overline{c}^2}{\gamma(\gamma-1)}(p^{\mathrm{L}} - p^{\mathrm{R}}) + \frac{1}{2}[(pu^2)^{\mathrm{L}} - (pu^2)^{\mathrm{R}}] \end{pmatrix}. \tag{34}$$

Both (34) and (32) can be used to approximate $\delta\mathbf{U}$ in (25).

In order to determine completely the numerical fluxes in (25) we need to provide an algorithm for computing $\overline{c}$, s_1 and s_2. Intermediate sound speed $\overline{c}$ and wave speed s_1 are simply set to be their algebraic averages

$$\overline{c} = (c^{\mathrm{L}} + c^{\mathrm{R}})/2 \tag{35}$$

and

$$s_1 = (u^{\mathrm{L}} + u^{\mathrm{R}})/2. \tag{36}$$

Numerical values of s_2 can be computed from

$$s_2 = \begin{cases} \min(0, u^{\mathrm{L}} - c^{\mathrm{L}}, u^* - c^*) & \text{for } s_1 > 0, \\ \max(0, u^* + c^*, u^{\mathrm{R}} + c^{\mathrm{R}}) & \text{for } s_1 \leqslant 0, \end{cases} \tag{37}$$

which is a combination of two speeds in (18) according to the sign of s_1. Another simple choice for s_2 is

$$s_2 = \begin{cases} \min(0, \min(u^{\mathrm{L}}, u^{\mathrm{R}}) - \max(c^{\mathrm{L}}, c^{\mathrm{R}})) & \text{for } s_1 > 0, \\ \max(0, \max(u^{\mathrm{L}}, u^{\mathrm{R}}) + \max(c^{\mathrm{L}}, c^{\mathrm{R}})) & \text{for } s_1 \leqslant 0. \end{cases} \tag{38}$$

Remark 1. A difficulty in devising flux (25) is how to approximate $\mathbf{F}_1$. On the frame moving a speed of u, as depicted in the upper right subfigure of Fig. 2, the system becomes two isentropic waves $\pm c$ and a stationary contact. We can certainly use one scalar diffusive coefficient for the isentropic waves because of their equal wave speeds. However, if the scalar coefficient is directly applied for solving $\mathbf{F}_1$ following the HLL approach, the same amount of viscosity is added to the stationary contact as well, as that built in (30). We notice that the lost accuracy can be recovered by enforcing the isentropic relations (31). In this way, the present approach employs a scalar diffusive coefficient, but the derived flux, as proved in (32), is the same as the characteristic splitting method that introduces the minimum diffusion to all characteristic variables. The gain is that one can achieve the characteristic splitting without using matrix operations. The upwinding property of the overall flux (25) will be proved in Remark 2.

Remark 2. Letting $\mathbf{W} = \mathbf{R}^{-1}\mathbf{U}$ and multiplying (7) by $\mathbf{R}^{-1}$, one obtains

$$\mathbf{W}_t + \mathbf{\Lambda}\mathbf{W}_x = 0, \tag{39}$$

which is a system of uncoupled scalar equations. We show here that the AUFS scheme is actually a first-order upwind method for all scalar equations, or it is a characteristic splitting method. Since the one-sided approximation is guaranteed for the supersonic flows by introducing term zero in (37), only subsonic case is considered here. Without losing generality, we only prove for $u > 0$. For the wave moving at the speed of $u - c$, the scheme gives $\hat{F}_1 = -cW^{\mathrm{R}}_{u-c}$, as implied in (26), and $\hat{F}_2 = 0$, so that

$$\hat{F}_{u-c} = \left(1 - \frac{u}{c}\right)(-cW^{\mathrm{R}}_{u-c}) = (u-c)W^{\mathrm{R}}_{u-c}.$$

Similarly for other two waves, the scheme gives

$$\hat{F}_{u} = \frac{u}{c}(cW^{\mathrm{L}}_{u}) = uW^{\mathrm{L}}_{u},$$

$$\hat{F}_{u+c} = \left(1 - \frac{u}{c}\right)(cW^{\mathrm{L}}_{u+c}) + \frac{u}{c}(2cW^{\mathrm{L}}_{u+c}) = (u+c)W^{\mathrm{L}}_{u+c}.$$

These three fluxes are just the numerical fluxes given by the first order upwind method for the corresponding scalar equations. Therefore, the stability condition of the scheme is the same as the first-order upwind scheme, i.e.,

$$\lambda \frac{\Delta t}{\Delta x} \leqslant 1,$$

where λ is the maximum value of absolute wave speeds.

Remark 3. The AUFS scheme is somehow a nonlinear variant of the Steger–Warming splitting method. However, numerical tests show that the AUFS scheme resolves much better the sonic point than the Steger–Warming scheme. A noticeable difference between two schemes is that the present scheme adopts wave estimate (37) or (38), which is developed in the Godunov-type schemes. We show here that the wave speed estimates implicitly introduce some amount of nonlinear diffusion only to the wave that potentially produces an expansion shock. In Remark 2, s_2 is simply assumed to be $u - c$. Wave estimates (37) and (38) give actually

$$s_2 = u - c - \epsilon,$$

where ϵ is a small value due to the minimum or maximum operator in the wave estimates. As done in Remark 2, one gets

$$\hat{F}_{u-c} = \left(1 - \frac{u}{c+\epsilon}\right)(-cW^{\mathrm{R}}_{u-c}) + \frac{u}{c+\epsilon}(\epsilon W^{\mathrm{L}}_{u-c}) = (u-c)W^{\mathrm{R}}_{u-c} + \epsilon M(W^{\mathrm{L}}_{u-c} - W^{\mathrm{R}}_{u-c}),$$

$$\hat{F}_{u} = \frac{u}{c+\epsilon}[(c+\epsilon)W^{\mathrm{L}}_{u}] = uW^{\mathrm{L}}_{u},$$

$$\hat{F}_{u+c} = \left(1 - \frac{u}{c+\epsilon}\right)(cW^{\mathrm{L}}_{u+c}) + \frac{u}{c+\epsilon}[(2c+\epsilon)W^{\mathrm{L}}_{u+c}] = (u+c)W^{\mathrm{L}}_{u+c}.$$

It is seen that nonlinear difference ϵ has no effect on the fluxes for waves moving at the speeds of u and $u + c$, but it introduces diffusion to the $u - c$ wave, which is the only wave that may create expansion shock for $u > 0$. Since s_2 used is the extreme value in (37) or (38), it is expected that ϵ is positive. Therefore, nonlinearly over-estimating wave speed s_2 mitigates the problem in resolving sonic points that troubles most upwind schemes. This is also confirmed by numerical tests in Section 5. The nonlinear effect of $\overline{c}$ and s_1 on numerical solutions is more complex, so simple algebraic averages (35) and (36) are used.

Remark 4. The scheme splits the flux vector, if high order terms are neglected, as that follows

$$\mathbf{F} = (1-M)\mathbf{P} + (u\mathbf{U} + M\mathbf{P}) = \begin{pmatrix} 0 \\ (1-M)p \\ (1-M)pu \end{pmatrix} + \begin{pmatrix} \rho u \\ \rho uu + Mp \\ \rho eu + Mpu \end{pmatrix}. \tag{40}$$

Substituting (19) to (10), one gets

$$M = \frac{|u|}{c},$$

which is the flow Mach number. In this sense, the scheme is a combination of the Mach number weighted average of pressure vector $\mathbf{P}$ and convective vector $u\mathbf{U}$. The convective vector only appears in $M\mathbf{F}_2$, and is solved by one-sided approximation according to the direction of s_1 or u as done in (23). This shows again that all waves moves at the speed of flow velocity are upwindingly solved. The present approach shares some similarities to wave-particle splitting [13], AUSM [9] and CUSP [7] schemes. However, the present approach, based on (9), provides a physically meaningful way to split the flux vector of the Euler equations as shown in Figs. 1 and 2. Mathematically, it is different from (40) by high-order terms that are not trivial in resolving discontinuities and realizes the characteristic splitting.

3.3. Extension to multi-dimensions

The Euler equations in two dimensions are given by

$$\mathbf{U}_t + \mathbf{F}_x + \mathbf{G}_y = 0, \tag{41}$$

where the conservative state vector $\mathbf{U}$ and the flux vector $\mathbf{F}$ and $\mathbf{G}$ are defined as

$$\mathbf{U} = \begin{Bmatrix} \rho \\ \rho u \\ \rho v \\ \rho e \end{Bmatrix}, \quad \mathbf{F} = \begin{Bmatrix} \rho u \\ \rho u^2 + p \\ \rho v u \\ \rho e u + p u \end{Bmatrix}, \quad \mathbf{G} = \begin{Bmatrix} \rho v \\ \rho u v \\ \rho v^2 + p \\ \rho e v + p v \end{Bmatrix}. \tag{42}$$

We consider here a numerical scheme that employs the finite volume method for any grid system. Given an interface with normal vector $\mathbf{n} = (n_x, n_y)$, and two adjacent states, it is required to provide a numerical flux through the interface. The Euler equations satisfy the rotational invariance property, (see, e.g. [18]),

$$n_x\mathbf{F} + n_y\mathbf{G} = \mathbf{T}^{-1}\mathbf{F}(\mathbf{T}\mathbf{U}).$$

Here $\mathbf{T}$ is the rotation matrix and $\mathbf{T}^{-1}$ is its inverse matrix

$$\mathbf{T} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & n_x & n_y & 0 \\ 0 & -n_y & n_x & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad \mathbf{T}^{-1} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & n_x & -n_y & 0 \\ 0 & n_y & n_x & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}. \tag{43}$$

We calculate $\tilde{\mathbf{U}} = \mathbf{T}\mathbf{U}$, $\tilde{\mathbf{F}} = \mathbf{F}(\mathbf{T}\mathbf{U})$,

$$\tilde{\mathbf{U}} = \mathbf{T}\mathbf{U} = (\rho, \rho\tilde{u}, \rho\tilde{v}, \rho e)^{\mathrm{T}},$$

$$\tilde{\mathbf{F}} = \mathbf{F}(\mathbf{T}\mathbf{U}) = (\rho\tilde{u}, \rho\tilde{u}^2 + p, \rho\tilde{u}\tilde{v}, \rho e\tilde{u} + p\tilde{u})^{\mathrm{T}},$$

where $\tilde{u}$ is the normal velocity through the interface $\tilde{u} = un_x + vn_y$, and $\tilde{v}$ the tangential velocity $\tilde{v} = -un_y + vn_x$. Thus, the Euler equations can be written as, in a one-dimensional form

$$\tilde{\mathbf{U}}_t + \tilde{\mathbf{F}}_n = 0. \tag{44}$$

It is generally known that the system (44) contains four wave speeds

$(\tilde{u}-c,\tilde{u},\tilde{u},\tilde{u}+c)$.

The AUFS method is able to split them to two flux vectors, one can be solved by one-side differencing, and another contains two isentropic waves and two stationary waves, $(-c,0,0,+c)$. In a similar way as done for 1D Euler equations, one gets

$$\begin{aligned}\hat{\tilde{\mathbf{F}}} &= (1-M)\tilde{\mathbf{F}}_1 + M\tilde{\mathbf{F}}_2 \\ &= (1-M)[\tfrac{1}{2}(\tilde{\mathbf{P}}^{\mathrm{L}}+\tilde{\mathbf{P}}^{\mathrm{R}})+\delta\tilde{\mathbf{U}}] + M[\tilde{\mathbf{U}}^{\alpha}(\tilde{u}^{\alpha}-s_2)+\tilde{\mathbf{P}}^{\alpha}],\end{aligned} \tag{45}$$

where $\tilde{\mathbf{P}}=(0,p,0,p\tilde{u})^{\mathrm{T}}$. Rotating back the flux, one gets the final numerical flux through the interface

$$\hat{\mathbf{F}} = \mathbf{T}^{-1}\hat{\tilde{\mathbf{F}}} = (1-M)\left[\frac{1}{2}(\mathbf{P}^{\mathrm{L}}+\mathbf{P}^{\mathrm{R}})+\delta\mathbf{U}\right] + M[\mathbf{U}^{\alpha}(\tilde{u}^{\alpha}-s_2)+\mathbf{P}^{\alpha}\Big], \tag{46}$$

where $\mathbf{P}=(0,pn_x,pn_y,p\tilde{u})^{\mathrm{T}}$. Artificial viscosity term $\delta\mathbf{U}$ is obtained, in a similar way to get (32)

$$\delta\mathbf{U} = \frac{1}{2\overline{c}}\begin{pmatrix} p^{\mathrm{L}}-p^{\mathrm{R}} \\ (pu)^{\mathrm{L}}-(pu)^{\mathrm{R}} \\ (pv)^{\mathrm{L}}-(pv)^{\mathrm{R}} \\ \frac{\overline{c}^2}{\gamma-1}(p^{\mathrm{L}}-p^{\mathrm{R}})+\frac{1}{2}[(pq^2)^{\mathrm{L}}-(pq^2)^{\mathrm{R}}] \end{pmatrix}, \tag{47}$$

where $q^2=u^2+v^2$.

The above procedure can be readily extended to three dimensions. We may rewrite the numerical flux in a more concise form valid for the Euler equations with any dimensions

$$\hat{\mathbf{F}} = (1-M)[\frac{1}{2}(\mathbf{P}^{\mathrm{L}}+\mathbf{P}^{\mathrm{R}})+\delta\mathbf{U}] + M[\mathbf{U}^{\alpha}(\tilde{u}^{\alpha}-s_2)+\mathbf{P}^{\alpha}], \tag{48}$$

where

$$\mathbf{P} = \begin{pmatrix} 0 \\ p\mathbf{n} \\ p\tilde{u} \end{pmatrix}, \quad \delta\mathbf{U} = \frac{1}{2\overline{c}}\begin{pmatrix} p^{\mathrm{L}}-p^{\mathrm{R}} \\ (p\mathbf{v})^{\mathrm{L}}-(p\mathbf{v})^{\mathrm{R}} \\ \frac{\overline{c}^2}{\gamma-1}(p^{\mathrm{L}}-p^{\mathrm{R}})+\frac{1}{2}[(p\mathbf{v}\cdot\mathbf{v})^{\mathrm{L}}-(p\mathbf{v}\cdot\mathbf{v})^{\mathrm{R}}] \end{pmatrix} \tag{49}$$

and $\tilde{u}=\mathbf{v}\cdot\mathbf{n}$. Symbols α, $\overline{c}$, s_1 and s_2 are defined as those in one-dimension, (24), (35), (36) and (37), respectively, but using normal velocities.

Remark 5. Isentropic relations are enforced in order to resolve the stationary entropy wave, which is accurate for 1D Euler equations. However, there are stationary shear waves in multi-dimensional flows, so that $\delta\mathbf{U}$ in (49) introduces some amount of dissipation to the shear waves. Note that the similar dissipation is built in the Steger–Warming splitting as well. In fact, the tangential component can be excluded from the normal component simply by replacing the velocity vector in (49) by the normal velocity vector; however it is found in our early tests that this attempt causes some carbuncle-like problems. It seems that some dissipation to shear waves is a different mechanism from the dissipative pressure term [10] in suppressing carbuncle instability. It will be shown in Section 5 that the present scheme passes all tests for inspecting carbuncle instability. Another controversial issue is related to the resolution of boundary layers in viscous flow computations. On the one hand, a scheme or Riemann solver that can resolve an isolated shear layer should be more accurate for the simulation of boundary layers, and this principle has been followed by many researchers. On the other hand, for the flows with high Reynolds numbers, the size of grid is often not fine enough for thin shear layers, for instance, the shear layer created in shock diffraction or Mach reflection. It has been reported that under-resolved numerical solutions contain spurious nonphysical vortices

in simulations of a shear layer [1]. Note that the under-resolved problem cannot be avoided by using a fine grid. The grid size is limited by available computer resources in practice; more importantly for the Euler equations numerical solutions are always under-resolved whatever the grid size is, if we consider the Euler equations be the Navier–Stokes equations with an infinitely large Reynolds number. Some amount of dissipation may be helpful to stabilize the simulation of thin shear layers. So we leave the dissipation to shear waves as it is in $\delta\mathbf{U}$. The effect of the dissipation on carbuncle instability, spurious vortices, and boundary layers is being under investigation.

4. High-order extension

The AUFS scheme provides a numerical flux based on two adjacent states besides an interface. The scheme can be extended to high-order accuracy following van Leer's MUSCL approach [20,21], which is routinely used in practice today especially on unstructured grids. In this section, a second-order extension for the two-dimensional Euler equations is described for any grid system. The technique can be readily extended to three dimensions.

The conservation laws are written for a discrete control volume as

$$\mathbf{U}_i^{n+1} = \mathbf{U}_i^n - \frac{\Delta t}{\Delta\Omega}\sum_{k=1}^{\text{faces}} \hat{\mathbf{F}}_k(\mathbf{M}^{\mathrm{L}}, \mathbf{M}^{\mathrm{R}}), \tag{50}$$

where $\hat{\mathbf{F}}_k(\mathbf{M}^{\mathrm{L}}, \mathbf{M}^{\mathrm{R}})$ is the flux vector through interface k, determined by primitive variables on two sides of the interface, $\mathbf{M}^{\mathrm{L}}$ and $\mathbf{M}^{\mathrm{R}}$. Primitive variables, $\mathbf{M}$, are (ρ, u, v, p). One may achieve second-order accuracy in both time and space by setting

$$\mathbf{M}^{\mathrm{L}} = \mathbf{M}_i^n + \frac{\Delta t}{2}(\mathbf{M}_t)_i^n + (\nabla\mathbf{M})_i^n \cdot (\mathbf{r}_k - \mathbf{r}_i)$$

and

$$\mathbf{M}^{\mathrm{R}} = \mathbf{M}_j^n + \frac{\Delta t}{2}(\mathbf{M}_t)_j^n + (\nabla\mathbf{M})_j^n \cdot (\mathbf{r}_k - \mathbf{r}_j).$$

A sketch of the interface is shown in Fig. 3. Time derivative $\mathbf{M}_t$ is determined by the quasi-linear Euler equations

$$\mathbf{M}_t = -(\mathbf{A}, \mathbf{B}) \cdot \nabla\mathbf{M},$$

where matrixes $\mathbf{A}$ and $\mathbf{B}$ are

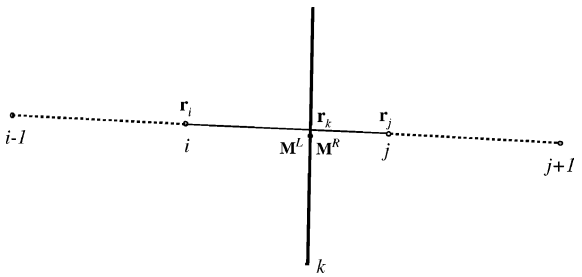

Fig. 3. Sketch of an interface: face k and its two neighboring volumes i and j; $i-1$ and $j+1$ are ghost points.

$$\mathbf{A} = \begin{pmatrix} u & \rho & 0 & 0 \\ 0 & u & 0 & 1/\rho \\ 0 & 0 & u & 0 \\ 0 & \gamma p & 0 & u \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} v & 0 & \rho & 0 \\ 0 & v & 0 & 0 \\ 0 & 0 & v & \rho \\ 0 & 0 & \gamma p & v \end{pmatrix}. \tag{51}$$

In order to suppress possible numerical oscillations, all gradients or slopes used are locally limited. Only the slopes in the direction between two control volumes $\mathbf{r}_{ij}$ are limited by the MINMOD limiter. Setting all gradients be zero, one gets the first order scheme; setting time derivatives be zero, one gets a scheme with second-order accuracy in space and first order in time.

Given the states of $\mathbf{M}^{\mathrm{L}}, \mathbf{M}^{\mathrm{R}}$ on both sides of the interface, the flux is calculated by the AUFS scheme (48), or other Riemann solvers. Two Riemann solvers are used for the comparison purpose, the exact one and the HLLC solver.

5. Numerical results

In this section we illustrate the performance of the AUFS scheme on the 1D and 2D Euler equations for ideal gases with $\gamma = 1.4$. For 1D tests, we solve a few initial value problems, which are generally tested by Toro [18]. Data consists of two constant states $(\rho, u, p)^{\mathrm{L}}$ and $(\rho, u, p)^{\mathrm{R}}$, separated by a discontinuity at position $x = x_0$, and are given in Table 1. The exact and numerical solutions are found in the spatial domain $0 \leqslant x \leqslant 1$ using 100 cells. The Courant number or CFL number is taken as 0.9. For each test problem, an initial location of discontinuity, x_0, and the output time are selected; these are stated in the caption of each figure. Numerical results of nearly all upwind schemes, e.g. [5,14], HLL, HLLC, AUSM, for the same test problems can be found in [18]. In this paper, for the sake of clarity, numerical results are compared only with the Godunov scheme using the exact Riemann solver, and the Steger–Warming FVS scheme, which are two representatives of the Godunov-type scheme and the FVS scheme.

Two 1D AUFS algorithms are tested in this section. Both algorithms are based on flux (25), intermediate sound speed (35) and first wave estimate (36). One algorithm employs dissipation (28) and second wave estimate (37), denoted by "AUFS", and the another follows (34) and (38) instead, denoted by "AUFST". All 1D results are obtained using the Fortran subroutines that are listed in Appendix A.

The solution of Test 1 consists of a right traveling shock wave, a contact wave, and left expansion waves with a sonic point inside. This test is devised to assess the entropy satisfaction property of numerical methods. Fig. 4 shows the results of the three schemes. The Godunov scheme gives an expansion shock at the sonic point, while the Steger–Warming scheme shows a bump there. The AUFS scheme resolves the sonic point more smoothly than other two schemes, because it implicitly introduces some amount of nonlinear viscosity by slightly over-estimating the wave speed s_2 around left traveling waves, as discussed in Remark 3. In comparison with the Godunov scheme, there is an additional point inside the expansion shock. The resolution of the contact wave is virtually identical for three schemes. The AUFS scheme even resolves the shock slightly sharper than other two schemes.

Table 1
Data for five test problems

Test	$(\rho, u, p)^{\mathrm{L}}$	$(\rho, u, p)^{\mathrm{R}}$
1	$(1.0, 0.75, 1.0)$	$(0.125, 0.0, 0.1)$
2	$(1.0, -2.0, 0.4)$	$(1.0, 2.0, 0.4)$
3	$(1.0, 0.0, 1000.0)$	$(1.0, 0.0, 0.01)$
4	$(5.99924, 19.5975, 460.894)$	$(5.99242, -6.19633, 46.095)$
5	$(1.0, -19.59745, 1000.0)$	$(1.0, -19.59745, 0.01)$

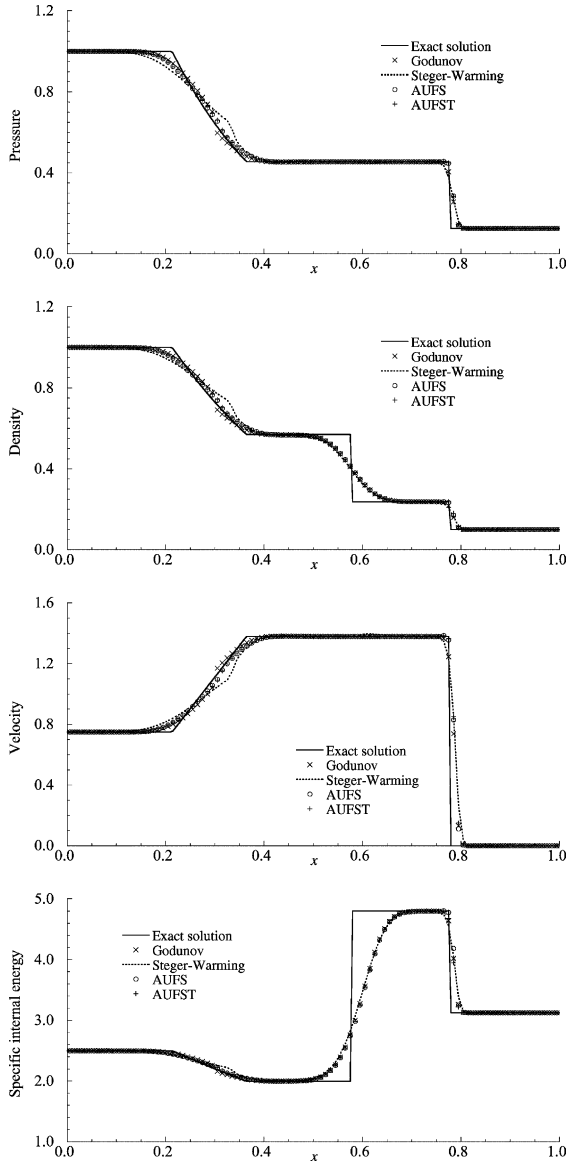

Fig. 4. Solutions of Test 1 with $x_0 = 0.3$. Exact solutions (solid line) and numerical solutions of the first-order schemes are compared at time 0.2 U.

Test 2 is a receding flow problem. The solution consists of two symmetric expansion waves; the central region is close to vacuum, which makes this problem a suitable test for accessing the performance of numerical methods for low-density flows. The results are shown in Fig. 5. Pressure and density results from three schemes agree similarly well with the exact solutions. Both AUFS algorithms give slightly more diffusive velocity and internal energy than other two schemes. The AUFS algorithm is less dissipative than the AUFST in this test that contains only isentropic waves. In view of the fact that the Godunov-type scheme with linearized Riemann solvers will fail for this test [4], the AUFS scheme gives reasonably good results.

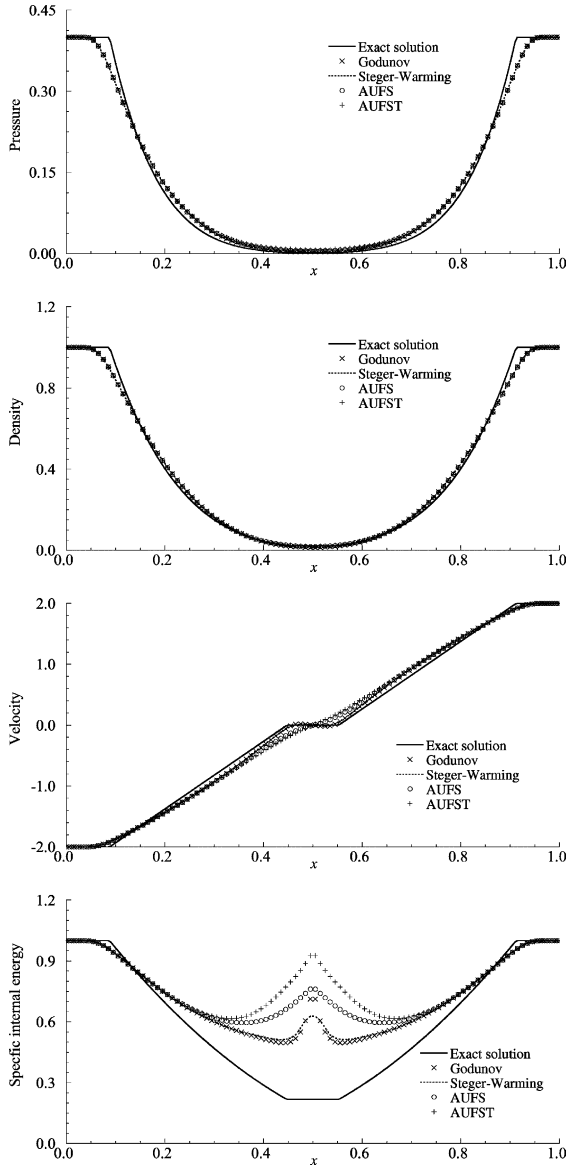

Fig. 5. Solutions of Test 2 with $x_0 = 0.5$. Exact solutions (solid line) and numerical solutions of the first-order schemes are compared at time 0.15 U.

The solution of Test 3 consists of a strong right running shock wave of shock Mach number 198, a contact surface and left expansion waves. It is designed to assess the robustness and accuracy of numerical methods. Fig. 6 shows the results for three schemes. The results from the AUFS scheme are virtually identical to those of the Godunov scheme. The Steger–Warming scheme has a non-physical bump around contact wave, which is clearly seen in pressure and velocity distributions.

Test 4, as Test 3, is also designed to access the robustness of numerical schemes. The solution of Test 4 consists of three strong discontinuities. It is an interaction of two strong impinging shock waves. The left

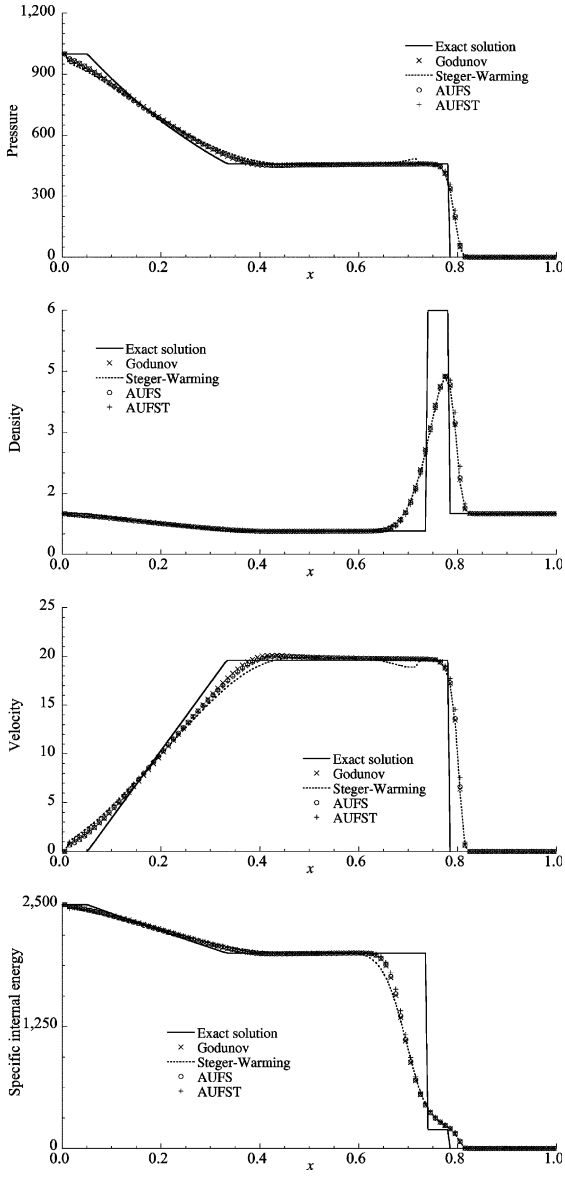

Fig. 6. Solutions of Test 3 with $x_0 = 0.5$. Exact solutions (solid line) and numerical solutions of the first-order schemes are compared at time 0.012 U.

shock wave moves to the right very slowly, which adds another difficulty to numerical methods. Fig. 7 shows the results for three schemes. Three schemes behave similarly well. The AUFS scheme resolves the right-traveling shock slightly sharper than the Godunov scheme, but gives a wider left-traveling shock. This is due to the fact that the AUFS scheme nonlinearly introduces more artificial viscosity to the wave moving opposite to the flow direction.

Test 5 is the similar to Test 3, except that a negative uniform background speed is added so as to obtain a stationary contact discontinuity. This test is devised to access the ability of numerical methods to resolve

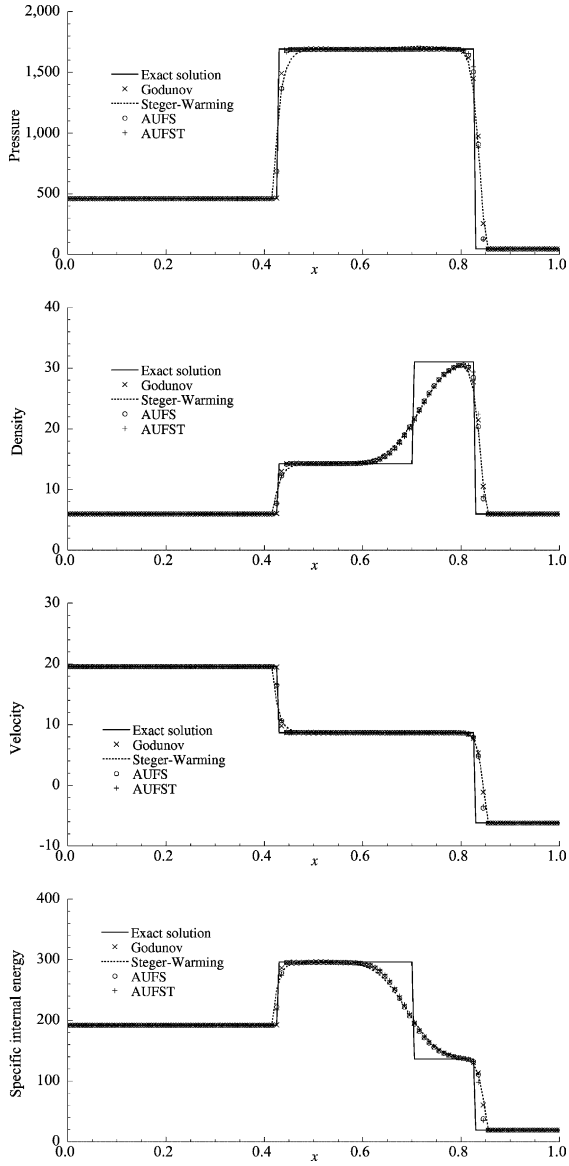

Fig. 7. Solutions of Test 4 with $x_0 = 0.4$. Exact solutions (solid line) and numerical solutions of the first-order schemes are compared at time 0.035 U.

the slowly moving contact wave as well as their robustness. Results are shown in Fig. 8. The AUFST algorithm does a better job than the AUFS algorithm that gives one peak behind the shock wave. The contact discontinuity is heavily smeared in the results of the Steger–Warming scheme. The AUFS and the Godunov schemes are clearly superior to the Steger–Warming scheme for this test problem. Artificial viscosity built in the AUFS scheme around a stationary contact is trivial, given by (28) or (34) that is zero under compatible conditions. The ability of the AUFS scheme to resolve contact discontinuity is more straightforwardly demonstrated in Fig. 9, in which initial conditions give an isolated contact wave. It is seen that the results of

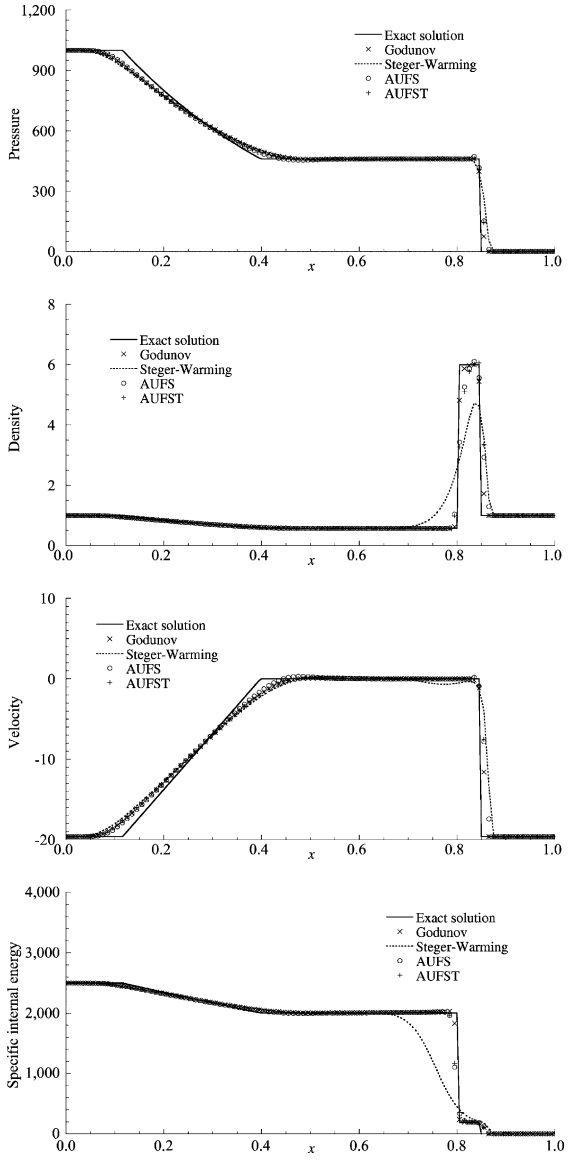

Fig. 8. Solutions of Test 5 with $x_0 = 0.8$. Exact solutions (solid line) and numerical solutions of the first-order schemes are compared at time 0.012 U.

the AUFS scheme are identical to those of the Godunov scheme for isolated stationary and moving contact waves.

In order to verify the performance of the AUFS scheme in resolving isolated shock waves, numerical results for solving a Mach 20 shock wave are given in Fig. 10. The AUFS scheme resolves a moving shock similarly well as the Godunov scheme, although it requires one or two cells to resolve a stationary shock wave. This is not a serious problem for multi-dimensional flows because numerical meshes are not aligned with shock front in general.

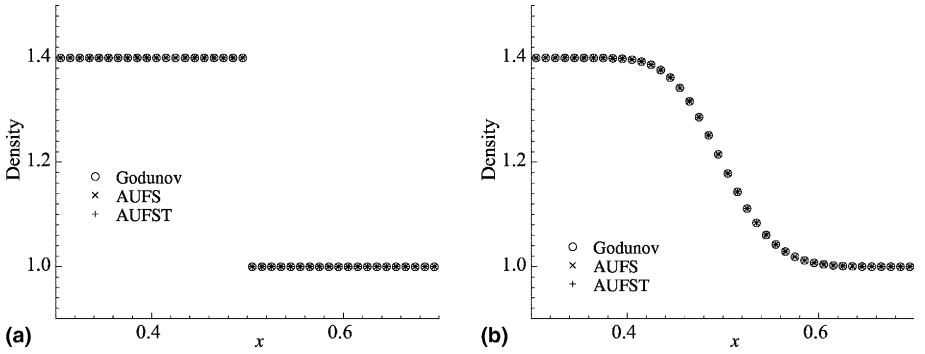

Fig. 9. Resolution of contact waves: (a) stationary contact wave; (b) moving contact wave.

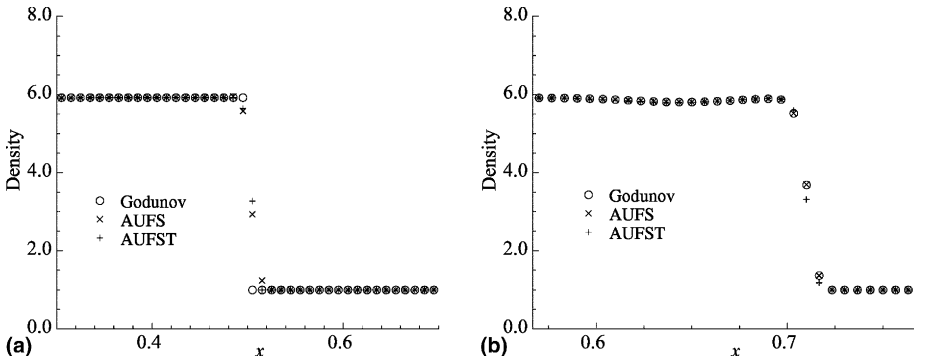

Fig. 10. Resolution of shock waves, $M_s = 20$: (a) stationary shock wave, 200 steps; (b) moving shock wave, 80 steps.

A few 2D problems are solved to illustrate the accuracy and robustness of the AUFS scheme. Numerical results are compared with the exact Riemann solver and the HLLC approximate solver. All numerical results are computed using the same limiter and boundary conditions for different solvers.

First test case is the propagation of a planar shock wave of $M_s = 6$ in a straight channel. The long channel is covered by a nominally uniform grid of 20×800 with unit spacing, the centerline of which is perturbed from that of a perfectly uniform mesh in the following manner, $\delta y = \pm 10^{-3}$, as sketched in Fig. 11(a), which would be undetectable at the correct scale. The case was first proposed and investigated by Quirk [12], and now becomes one of the significant examples demonstrating the carbuncle instability. It was reported that the Riemann solvers that explicitly capture the 1D contact wave suffer from this problem of odd-even decoupling, such as the exact solver, HLLC approximate solver [11,12]. The problem is illustrated in Figs. 11(b) and (c), in which the planar shock wave has been greatly distorted by using the exact or the HLLC solver, in comparison with the AUFS scheme shown in Fig. 11(d). The AUFS scheme exhibits absence of this carbuncle instability, and this behavior is similar to the AUSM scheme [8] and other FVS schemes. Notice that the two Godunov-type schemes not only resolve the shock structure erroneously but also give a slightly faster shock speed, as seen from the locations of shock waves. Second-order version of the schemes, extended following Section 4, may mitigate, but not solve the problem as shown in Figs. 11(e) and (f). Although a planar shock wave is recovered, many disturbances still prevail behind the shock wave in comparison with the uniform solution given from the AUFS scheme.

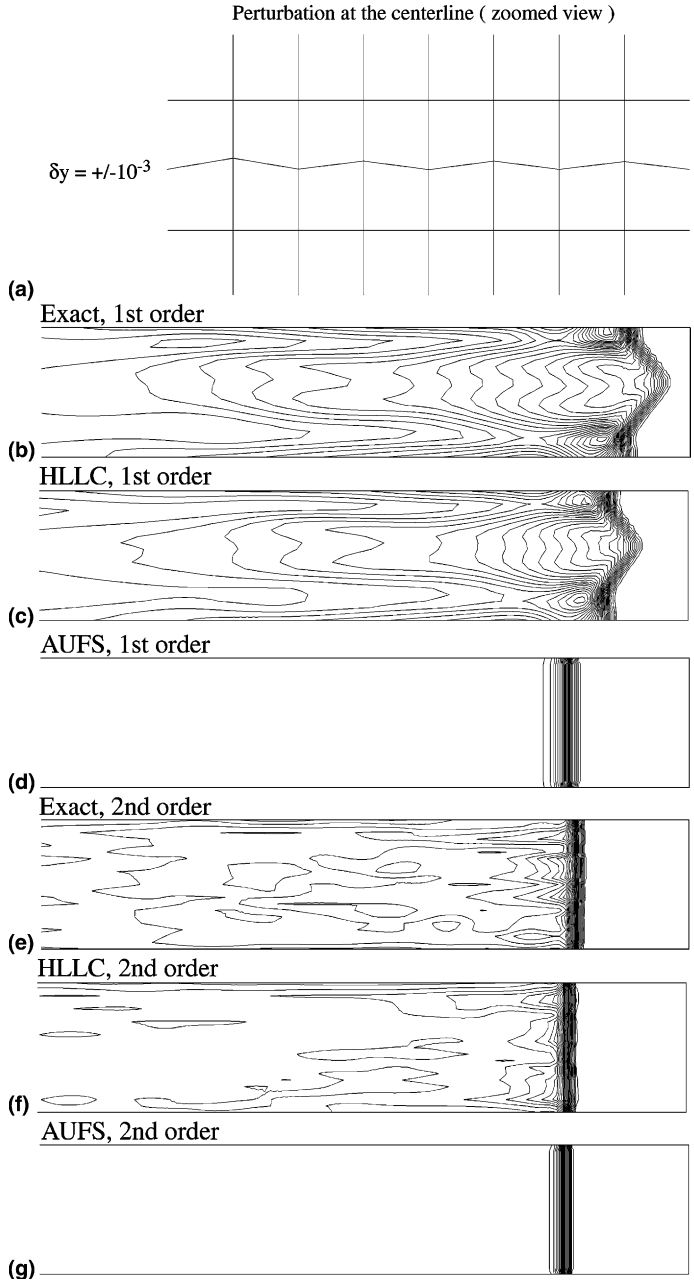

Fig. 11. A shock wave of $M_s = 6$ propagating in a channel: (a) perturbation in the vertical direction of $\delta y = \pm 10^{-3}$ is imposed at centerline.

Another problem of some Godunov-type schemes is that they produce kinked Mach stems (e.g. [12]) in the simulation of shock reflection over a straight wedge. A computational grid is shown in Fig. 12(a), in which only 1 of 16 grid lines in each dimension are plotted. In the results of the exact and the HLLC solver, the Mach stem is kinked, and some disturbances are visible behind the incident shock on the upper surface.

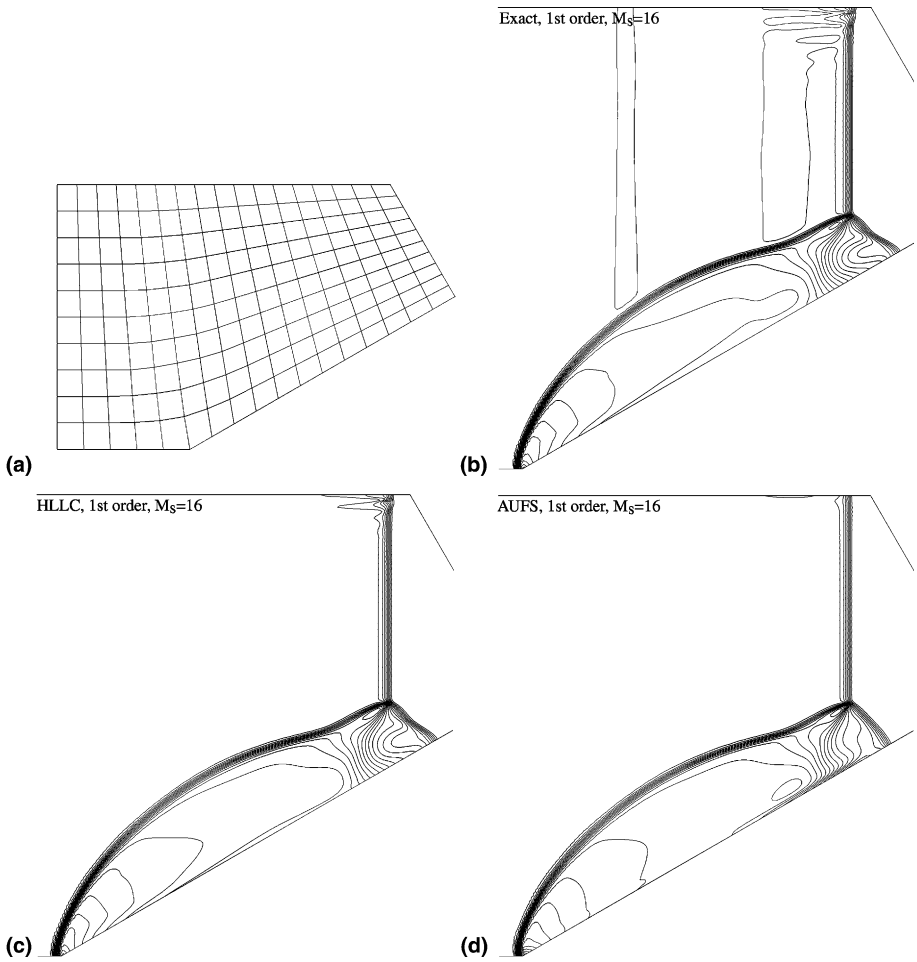

Fig. 12. Shock reflection: numerical results of first-order schemes.

The problem can be more serious using fine grids or other badly aligned grids. There is probably insufficient dissipation added via the shear waves that are neglected in all 1D Riemann solvers. The AUFS scheme gives a much better result as shown in Fig. 12(d). Similarly to the first case, a second-order version of the Godunov-type schemes fails to solve the problem completely. Kinked shock wave and post-shock disturbance are still visible in Fig. 13(b), compared with the clean solution given by the AUFS scheme shown in Fig. 13(a). A smaller grid size, 1/32 of that show in Fig. 12, is used for the second-order computations to be able to illustrate the difference. Judged from the thickness of shock waves and slipstreams in these 2D results, it is clear that the AUFS scheme, free of carbuncle instability, resolves shock waves and slipstreams similarly well as the exact Riemann solver does.

The test of shock reflection offers for inspecting the ability of the AUFS scheme to capture strong moving shock waves. Fig. 14 compares the results of a supersonic flow over a circular cylinder using a few first-order schemes. This test case has been commonly used for testing carbuncle instability. The Godunov and HLLC Riemann solvers resolve the bow shock, but pollute the flow regions behind the bow shock. For more stretched meshes, they can fail to correctly resolve the bow shock structure. The AUFS scheme again provides a clean flow field behind the bow shock. Fig. 15 shows another example that demonstrates the

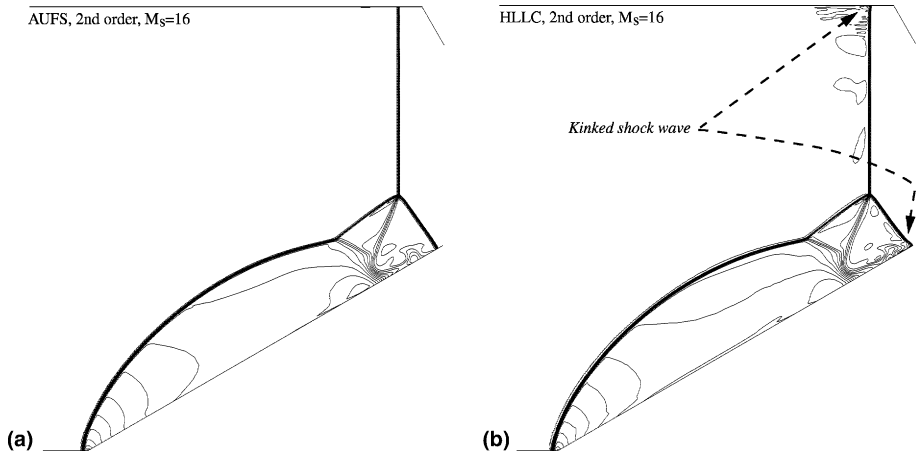

Fig. 13. Shock reflection: numerical results of second-order schemes.

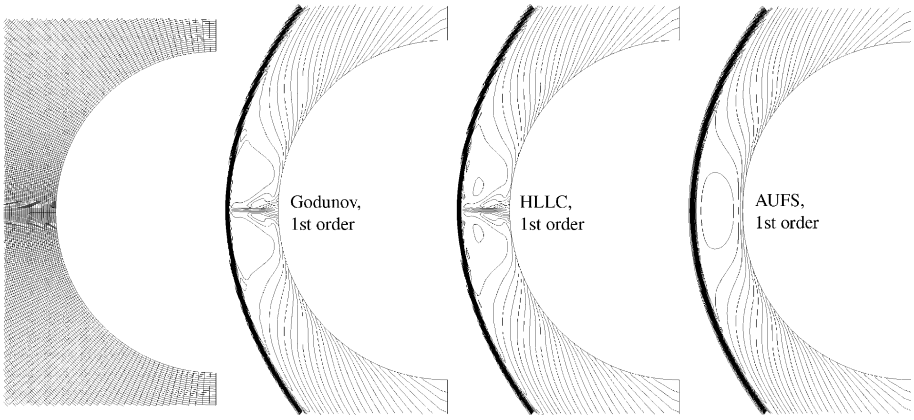

Fig. 14. Mach 20 flow over a circular cylinder: grid and density contours.

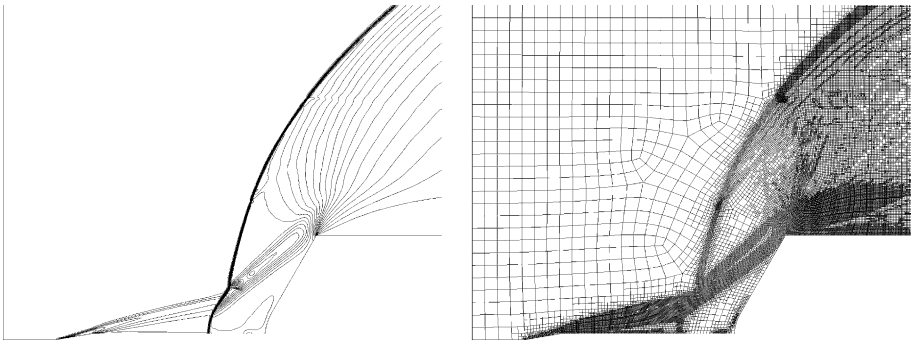

Fig. 15. Mach 7 flow over a spiked blunt body: density contours and the corresponding solution-adaptive unstructured grid.

robustness of the AUFS scheme in computing hypersonic flows with stationary shock waves and their interactions. It is a steady hypersonic flow of $M = 7$ over a blunt body with a long and thin spike. The result is obtained using the second-order AUFS scheme on a solution-adaptive unstructured grid [17]. The oblique shock, bow shock, expansion waves, and their strong interactions are well resolved.

6. Concluding remarks

We present a novel numerical scheme, named AUFS scheme, for solving the Euler equations. It would be impossible for the scheme to emerge without two great ideas. One idea is to split the flux vector based on two artificially introduced wave speeds, discussed in Section 3; another idea is to choose the flow velocity as one of the wave speeds, stated in Section 3.2. The AUFS scheme surpasses, to our knowledge, all existing upwind schemes, in one or few points that follow:

1. The scheme is simple. It realizes one-side differencing for all waves in the 1D and multi-dimensional flows without recourse to any matrix operation, so that it can be extended to more complex systems with less difficulty.
2. The scheme resolves sonic points smoothly. It implicitly and non-linearly introduces more artificial dissipation only to the wave that possibly creates an expansion shock, while leaving other waves untouched.
3. The scheme can resolve exactly 1D stationary contact discontinuities. In multi-dimensions, the scheme does not suffer from carbuncle instabilities, which trouble the Godunov-type schemes that may resolve well contact discontinuities.
4. The resolution and robustness of the scheme is, overall, comparable with the exact Riemann solver.

It should be noted that the present scheme for multi-dimensional flows introduces some amount of dissipation to shear waves, as the Steger–Warming characteristic splitting does. The role of the dissipation in overcoming carbuncle instability, and its effects, positive and negative, on the simulation of shear layers are under investigation, and will be reported soon.

Appendix A

This is a list of two Fortran source codes for solving 1D Euler equations for perfect gases. Given left and right states in primitive variables, both subroutines calculate the fluxes through the interface between them. Algorithms 1 and 2 correspond to AUFS and AUFST tested in Section 5, respectively.

Algorithm 1.

```
      subroutine flux_aufs1D(rl,ul,pl,rr,ur,pr,gam,fr,fru,fre)
c..Input: left states (rl,ul,pl)
c..Input: right states (rr,ur,pr)
c..Input: the ratio of specific heats (gam)
c..Output: fluxes (fr,fru,fre)
c..last modified on March 30, 2003 by Mingyu Sun
      gm1 = gam-1.
      al = sqrt(gam*pl/rl)
      ar = sqrt(gam*pr/rr)
c....part 1 (Calculating F1)
      ca = (al+ar)/2.
      f1r = (pl-pr)/2./ca
```

```
      f1ru = ((ul*pl-ur*pr)/ca+pl+pr)/2.
      f1re = (ca/gm1*(pl-pr)+(pl*ul*ul-pr*ur*ur)/2./ca+ul*pl+ur*pr)/2.
c....part 2 (Weighted average of F1 and F2)
c.isentropic wave estimate
      ustar = (ul+ur)/2.+(al-ar)/gm1
      cstar = (al+ar)/2.+gm1*(ul-ur)/4.0
c.upwinding
      s1 = (ul+ur)/2.
      if(s1.gt.0.)then
      s2 = amin1(0.0,ul-al,ustar-cstar)
      am = s1/(s1-s2)
      fr  = (1.-am)*f1r +am*(rl*(ul-s2))
      fru = (1.-am)*f1ru+am*(rl*ul*(ul-s2)+pl)
      fre = (1.-am)*f1re+am*((1./gm1*pl+rl*ul*ul/2.)*(ul-s2)+pl*ul)
      else
      s2 = amax1(0.0,ustar+cstar,ur+ar)
      am = s1/(s1-s2)
      fr  = (1.-am)*f1r +am*(rr*(ur-s2))
      fru = (1.-am)*f1ru+am*(rr*ur*(ur-s2)+pr)
      fre = (1.-am)*f1re+am*((1./gm1*pr+rr*ur*ur/2.)*(ur-s2)+pr*ur)
      end if
      return
      end
```

Algorithm 2.

```
      subroutine flux_aufst1D(rl,ul,pl,rr,ur,pr,gam,fr,fru,fre)
c..last modified on March 30, 2003 by Mingyu Sun
      gm1 = gam-1.
      al = sqrt(gam*pl/rl)
      ar = sqrt(gam*pr/rr)
c....part 1 (Calculating F1)
      ca = (al+ar)/2.
      f1r = gam*(pl-pr)/2./ca
      f1ru = (gam*(ul*pl-ur*pr)/ca+pl+pr)/2.
      f1re = (ca/gm1*(pl-pr)+gam*(pl*ul*ul-pr*ur*ur)/2./ca+ul*pl+ur*pr)/2.
c....part 2 (Weighted average of F1 and F2)
      s1 = (ul+ur)/2.
c.upwinding
      if(s1.gt.0.)then
      s2 = amin1(0.0,amin1(ul,ur)-amax1(al,ar))
      am = s1/(s1-s2)
      fr  = (1.-am)*f1r +am*(rl*(ul-s2))
      fru = (1.-am)*f1ru+am*(rl*ul*(ul-s2)+pl)
      fre = (1.-am)*f1re+am*((1./gm1*pl+rl*ul*ul/2.)*(ul-s2)+pl*ul)
      else
      s2 = amax1(0.0,amax1(ul,ur)+amax1(al,ar))
      am = s1/(s1-s2)
```

```
23.     fr  =(1.-am)*f1r +am*(rr*(ur-s2))
24.     fru =(1.-am)*f1ru+am*(rr*ur*(ur-s2)+pr)
25.     fre =(1.-am)*f1re+am*((1./gm1*pr+rr*ur*ur/2.)*(ur-s2)+pr*ur)
26.     end if
27.     return
28.     end
```

References

[1] D.L. Brown, M.L. Minion, Performance of under-resolved two-dimensional incompressible flow simulations, J. Comput. Phys. 122 (1995) 165.
[2] S.F. Davis, Simplified second-order Godunov-type methods, SIAM J. Sci. Statist. Comput. 9 (1988) 445.
[3] B. Einfeldt, On Godunov-type methods for gas dynamics, SIAM J. Numer. Anal. 25 (1988) 294.
[4] B. Einfeldt, C.D. Munz, P.L. Roe, B. Sjogreen, On Godunov-type methods near low densities, J. Comput. Phys. 92 (1991) 273.
[5] B. Engquist, S. Osher, One-sided difference approximations for nonlinear conservation laws, Math. Comput. 36 (1981) 321.
[6] A. Harten, P.D. Lax, B. van Leer, On upstream differencing and Godunov-type schemes for hyperbolic conservation laws, SIAM Rev. 25 (1983) 35.
[7] A. Jameson, Positive schemes and shock modeling for compressible flows, Intl. J. Numr. Methods Fluids 20 (1995) 743.
[8] M.S. Liou, A sequel to AUSM: AUSM$^+$, J. Comput. Phys. 129 (1996) 364.
[9] M.S. Liou, C.J. Steffen, A new flux splitting scheme, J. Comput. Phys 107 (1993) 23.
[10] M.S. Liou, Mass flux schemes and connection to shock instability, J. Comput. Phys. 160 (2000) 623.
[11] M. Pandolfi, D. D'Ambrosio, Numerical instabilities in upwind methods: analysis and cures for the "Carbuncle" phenomenon, J. Comput. Phys. 166 (2001) 271.
[12] J.J. Quirk, A contribution to the great Riemann solver debate, Intl. J. Numer. Methods Fluids 18 (1994) 555.
[13] S.V. Rao, S.M. Deshpande, A class of efficient kinetic upwind methods for compressible flows, Report 91 FM 11, Indian Institute of Science, 1991.
[14] P.L. Roe, Approximate Riemann solvers, parameter vectors, and difference schemes, J. Comput. Phys. 43 (1981) 357.
[15] I.V. Sokolov, E.V. Timofeev, J. Sakai, K. Takayama, Artificial wind—a new framework to construct simple and efficient upwind shock-capturing schemes, J. Comput. Phys. 181 (2002) 354.
[16] J.L. Steger, R.F. Warming, Flux vector splitting of the inviscid gasdynamic equations with applications to finite difference methods, J. Comput. Phys. 40 (1981) 263.
[17] M. Sun, K. Takayama, Conservative smoothing on an adaptive quadrilateral grid, J. Comput. Phys. 150 (1999) 143.
[18] E.F. Toro, Riemann Solvers and Numerical Methods for Fluid Dynamics, second ed., Springer, Berlin, 1999.
[19] E.F. Toro, M. Spruce, W. Speares, Restoration of the contact surface in the HLL-Riemann solver, Shock Waves 4 (1994) 25.
[20] B. van Leer, Towards the ultimate conservative difference scheme. III. Upstream-centered finite-difference schemes for ideal compressible flow, J. Comput. Phys. 23 (1977) 263.
[21] B. van Leer, Towards the ultimate conservative difference scheme. IV. A new approach to numerical convection, J. Comput. Phys. 23 (1977) 276.