Optimization of MUSCL scheme by dispersion and dissipation

Leng Yan[1], Li Xinliang[1*], Fu Dexun[2] & Ma Yanwen[2]

（1 LHD Institute of Mechanics, CAS, Beijing，100190）

（2 LNM Institute of Mechanics, CAS, Beijing，100190）

Abstract A second-order monotonicity-preserving optimized MUSCL scheme (OMUSCL2) is developed based on dispersion and dissipation optimization and monotonicity-preserving technique. The new scheme (OMUSCL2) has simple expression and is easy to be used in CFD codes. Compared with the original second-order or third-order MUSCL scheme, the new scheme shows nearly the same CPU costs and higher resolution to shockwaves and small-scale waves. We test the new scheme through a set of one-dimensional and two-dimensional tests, including the Shu-Osher problem, the sod problem, the Lax problem, two dimension double Mach reflection and RAE2822 transonic airfoil test. All numerical tests show that, compared with original MUSCL schemes, the new scheme has less dispersion error, less dissipation error and higher resolution.

Key Words MUSCL scheme, monotonicity-preserving, resolution, dissipation/dispersion error, optimization

Introduction

Computational fluid dynamics (CFD) plays an important role in the aerospace engineering, and to develop high solution methods is still a major work in CFD. Generally speaking, the resolution of numerical solution is the ability to describe flow characteristics which we are interested in. High resolution scheme [1] generally refers to the numerical solution of this scheme can give a sharp and vivid picture to the flow characteristics which are in the range of interesting physical scale. This scale usually contains small-scale flow structure which is difficult to simulation correctly. For shock waves, the resolution means the numerical shock is sharp and the parameters (such as density, velocity, and pressure) have no or small oscillation through shock. When the shock wave is generated in the flow field, characteristic scale of flow structure has greatly discrepancy among different regions [2]. The characteristic scale of inviscid shock is zero, while the characteristic scale of flow is finite. Furthermore, flow parameters are discontinuous through shock. All of these lead to great difficulties in numerical calculation. We require the scheme has high resolution and strong ability to capture shock for multi-scale complex flow (e.g. turbulence) with shock. In addition, numerical solution should be free from non-physical high frequency oscillation near the shock, and different scale physical parameters can't be polluted through shock wave. Therefore, it's necessary to develop high resolution schemes for engineering application.

Numerical simulation of shock has made significant progress since 1980s. In 1983, Harten [1] introduced the concept of total variation diminishing (TVD), and got a

second order TVD scheme. Based on the Harten's TVD concept and condition [1], limiters can be defined to restore the TVD property of the scheme and to prevent the non-physical oscillations near the discontinuities. TVD limiters are bounded non-linear functions obeying Harten's TVD condition. These limiters ensure that any reconstructed values at any time don't lie outside the range of the initial data [3]. Sweby [4] proposed a series of second order TVD schemes using the flux limiters. In addition, Van Leer [5] developed a method called MUSCL (monotonic upstream-centered scheme for conservation law). This method first extrapolated interface values by using the cell averages, then obtain flux through flux splitting technique. This avoids under and overshoots phenomena and leads to a maximum principle on the discrete solution. MUSCL methods are one of the most popular second-order or third-order finite volume methods. Although TVD schemes show highly efficient and stable shock capturing ability, the order in local extreme point is only first-order for satisfying the TVD property. To avoid this drawback, Harten [6] introduced the concept of essentially non-oscillatory (ENO). Then many researchers have constructed high-order ENO [6] [7] and WENO [7] [8] schemes. There are also the total variation bounded (TVB) [9] method and monotonicity-preserving (MP) [10] method. However, more stencil points are used in high-order (more than three-order) schemes, and this limits the flexibility in complex geometries.

Finite volume method (FVM) is widely used in engineering applications due to its simplicity for complex geometries and build-in conservative property. To keep flexible for complex geometries, the scheme stencil in most FVM codes are four points, i.e. four points are used to compute the face value $U_{I+1/2}$. For engineering CFD codes, MUSCL scheme is one of the most popular schemes to compute $U_{I+1/2}$. This scheme is modified from the base schemes by use limiter techniques. The base schemes of MUSCL are second-order central scheme, second-order upwind scheme, and third-order upwind scheme or Fromm scheme. Although MUSCL scheme have many good properties, it still has the room for optimization. For example, Four points are used to compute $U_{I+1/2}$, and only *three points* are used to compute *the Lift* and *the Right* face values $U^L_{I+1/2}$, $U^R_{I+1/2}$. The message of one point is not used in the computation of $U^L_{I+1/2}$ and $U^R_{I+1/2}$, i.e. the message of this point is *wasted*. Additional, the base scheme (such as ordinary third-order upwind scheme) can be optimized by using the dispersion and dissipation optimization techniques [11, 12].

In this work, based on dispersion and dissipation optimization and monotonicity-preserving technique, a second order optimized MUSCL scheme (OMUSCL2) is proposed. Compared with the classical second or third order MUSCL scheme, the new scheme has less dissipation and dispersion and thus has higher resolution for shockwave and small scale waves. The new scheme has the same stencil and nearly the same computational cost as that of the classical MUSCL scheme, thus it is easy to be used or be migrated in the finite volume CFD code.

The paper is organized as follows. Section 2 focuses on numerical method, including a review of MUSCL method, TVD conditions and the construction of OMUSCL2. In section 3 we extend the new scheme to the Euler equations. Section 4 is devoted to numerical results and we make

comparisons with classical MUSCL method. All tests show that OMUSCL2's resolution is better than original MUSCL method.

1 **Description of the numerical scheme**

1.1 **The scalar conservation law**

In this section, we start with the description in the one-dimensional case. Consider the scalar hyperbolic conservation law given by

$$\frac{\partial u}{\partial t}+\frac{\partial f(u)}{\partial x}=0. \tag{1}$$

For simplicity, we assume the grids points x_j are uniform. That is $x_{j+1}-x_j=h,\quad x_{j+1/2}=x_j+h/2$. Defined $I_j=[x_{j-1/2},x_{j+1/2}]$ is a uniform partition of the solution domain in space. The semi-discrete conservative scheme of (1) is

$$\frac{\partial u}{\partial t}+\frac{1}{h}(\hat{f}_{j+1/2}-\hat{f}_{j-1/2})=0, \tag{2}$$

where $\hat{f}_{j+1/2}$ is the numerical flux. The details of how to get it will be described in the following. Defining

$$L(u)=-\frac{1}{h}(\hat{f}_{j+1/2}-\hat{f}_{j-1/2}), \tag{3}$$

then (2) can be written as

$$\frac{\partial u}{\partial t}=L(u). \tag{4}$$

In this paper, (4) is discrete in time by TVD Runge- Kutta scheme [13].

$$\begin{aligned} u^{(1)}&=u^n+\Delta tL(u^n)\\ u^{(2)}&=\frac{3}{4}u^n+\frac{1}{4}u^{(1)}+\frac{1}{4}\Delta tL(u^{(1)})\\ u^{n+1}&=\frac{1}{3}u^n+\frac{2}{3}u^{(2)}+\frac{2}{3}\Delta tL(u^{(2)}) \end{aligned} \tag{5}$$

1. 2 **A short review on MUSCL and TVD property**

1.2.1 **Van Leer's MUSCL method**

In 1979, Van Leer [5] proposed MUSCL method. Considered Eq. (1) and its semi-discrete form (2), we can get many different semi-discrete schemes after splitting flux, such as second-order central scheme, second-order or third-order upwind scheme. Different schemes which depend on the expression of $\hat{u}_{j+1/2,L}$ and $\hat{u}_{j+1/2,R}$ can be written in unified form:

$$\hat{f}_{j+1/2}=f^{+}(\hat{u}_{j+1/2,L})+f^{-}(\hat{u}_{j+1/2,R}). \tag{6}$$

A more general form is

$$\begin{aligned} \hat{u}_{j+1/2,L}&=u_j+\frac{1}{4}[(1-k)\delta_x^{-}+(1+k)\delta_x^{+}]u_j\\ \hat{u}_{j+1/2,R}&=u_{j+1}-\frac{1}{4}[(1-k)\delta_x^{+}+(1+k)\delta_x^{-}]u_{j+1} \end{aligned} \tag{7}$$

$k=-1$ is upwind scheme; $k=1$ is central scheme and $k=0$ is Fromm scheme. They are all second-order accurate. While $k=1/3$ is a third-order accurate scheme. If we use this scheme simply, it will produce numerical oscillation. Van Leer [5] improve (7) to

$$\begin{aligned} \hat{u}_{j+1/2,L}&=u_j+\frac{1}{4}[(1-k)\bar{\delta}_x^{-}+(1+k)\bar{\delta}_x^{+}]u_j\\ \hat{u}_{j+1/2,R}&=u_{j+1}-\frac{1}{4}[(1-k)\bar{\delta}_x^{+}+(1+k)\bar{\delta}_x^{-}]u_{j+1} \end{aligned} \tag{8}$$

where

$$\begin{aligned} &\bar{\delta}_x^{+}u_j=\min\mathrm{mod}(\delta_x^{+}u_j,b\delta_x^{-}u_j)\\ &\bar{\delta}_x^{-}u_j=\min\mathrm{mod}(\delta_x^{-}u_j,b\delta_x^{+}u_j)\,.\\ &1\le b\le 3,\ \ k=-1,\ \ 1,\ \ 0,\ \ \frac{1}{3} \end{aligned}$$

Reducing significant changes in $\hat{u}_{j+1/2,L}$ and $\hat{u}_{j+1/2,L}$ to improve the ability of capturing shock.

1.2.2 **TVD scheme**

In 1959, Godunov [14] proposed monotonicity scheme. Then Jenning [15] proposed monotonicity–pre serving scheme by extending the concept of monotone to non-linear scheme. The numerical solution avoids oscillation near the shock using monotonicity or monotonicity-preserving scheme.

TVD which is first introduced by Harten means total variation diminishing. The basic idea is to apply the characteristic which is total variation diminishing of differential equations in constructing difference scheme. In order to construct

TVD scheme, Harten gave a sufficient condition. That is, if scheme can be written as

$$u_j^{n+1} = u_j^n + C_{j+1/2}^n(u_{j+1}^n - u_j^n) - D_{j+1/2}^n(u_j^n - u_{j-1}^n) \quad (9a)$$

and for any j satisfies

$$C_{j+1/2}^n \geq 0, D_{j+1/2}^n \geq, C_{j+1/2}^n + D_{j+1/2}^n \leq 1, \quad (9b)$$

the scheme is TVD scheme.

1. 3 Optimized MUSCL scheme (OMUSCL2) by controlling dispersion and dissipation

1.3.1 Fourier analysis of dispersion and dissipation

We discrete the scalar hyperbolic equation (1) on the stencil $[x_{j-2}, x_{j-1}, x_j, x_{j+1}, x_{j+2}]$. Clearly, five points can construct 4th order scheme at most. Here *we use five points to construct a second-order scheme*, then getting a spatial discrete expression with *two free coefficients*:

$$h\frac{\partial f}{\partial x} = (-a - 3b + \frac{1}{2}) f_{j-2} + (3a + 8b - 2) f_{j-1} + (-3a - 6b + \frac{3}{2}) f_j + af_{j+1} + bf_{j+2} \quad (10)$$

Where a and b are *free coefficients*.

In order to analyze the dispersion and dissipation errors quantitatively, we consider model problem

$$\frac{\partial u}{\partial t} + c\frac{\partial u}{\partial x} = 0, c = const > 0,$$

with initial condition:

$$u(x,0) = e^{ikx}.$$

The exact solution is $u(x,t) = e^{ik(x-ct)}$. If we let F_j/h to approximate $\left(\frac{\partial u}{\partial x}\right)_j$, then model problem is written as

$$\frac{\partial u_j}{\partial t} + c\frac{F_j}{h} = 0, c = const > 0$$

with initial condition:

$$u(x_j,0) = e^{ikx_j}.$$

Suppose different equation's exact solution has the form $u(x_j,t) = \hat{u}(t)e^{ik(x_j - ct)}$ then we get $F_j = K_e \cdot \hat{u}(t)e^{ik(x_j - ct)}$, $k_e = k_r + ik_i$. The solution is $u(x_j,t) = e^{-c\frac{k_r}{h}t} e^{i(kx_j - c\frac{k_i}{h}t)}$. Note that for the exact solution, we have $k_r = 0, k_i = kh$. So the dispersion and dissipation error can be reflected by the functions k_r and k_i. [16]

For (10) the functions of dispersion and dissipation are expression as ($kh = \alpha$)

$$K_r = (\frac{1}{2} - a - 2b)\cos(2\alpha) + (4a + 8b - 2)\cos(\alpha) + (\frac{3}{2} - 3a - 6b) \quad (11)$$

$$K_i = (a + 4b - \frac{1}{2})\sin(2\alpha) + 2(-a - 4b + 1)\sin(\alpha)$$

Define:

$$\eta = 1 - 2a - 4b$$
$$\xi = 2a + 8b - 1,$$

Then, Eq. (11) is rewritten as

$$K_r = \eta(\cos^2(\alpha) - 1)^2$$
$$K_i = (1 - \xi + \xi\cos(\alpha))\sin(\alpha) \quad , \quad (12)$$

Here η, ξ is the dissipation and dispersion coefficient, respectively, i.e. the dissipation property of the scheme (10) is determined only by the free parameter η , and the dispersion property of scheme (10) is determined only by the free parameter ξ . Since ξ and η are independent parameters, we can *optimize the dissipation and dispersion* property of scheme (10) *independently*.

Specially, when η=0, ξ=−1/3 , scheme (10) is a fourth-order central scheme.

1.3.2 Optimization for dissipation and dispersion coefficients

Equation (12) requires $\eta \geq 0$ since the dissipation must be positive. In the application of specific physical problem it can be given an appropriate value to avoid over smoothed discontinuities. So it's a controllable parameter. Here we choose $\eta = 0.2$ determined by tests. The following optimized mainly for dispersion coefficient using the best square approximation method.

The exact value of dispersion is $k_i = \alpha$, by the best square approximation we obtain

$$F(\xi)=\int_0^x(\theta-(1-\xi+\xi\cos(\alpha))\sin(\alpha))^2 d\theta$$
$$=\frac{x^3}{3}+\frac{\xi^3}{8}(x-\frac{1}{4}\sin 4x)+\frac{(1-\xi)^2}{2}(x-\frac{1}{2}\sin 2x)$$
$$+\frac{\xi}{2}(x\cos 2x-\frac{1}{2}\sin 2x)+2(1-\xi)x\cos x$$
$$-2(1-\xi)\sin x+\frac{2}{3}\xi(1-\xi)\sin^3 x \quad x\in[0,2\pi]$$

In table 1, for given x value, ξ value is presented when $F(\xi)$ obtains the minimum value. Figure 1 shows the dispersion curve with different ξ value and exact value. Fig. 1 shows that the dispersion curve will approach exact value better in the range of high wave number as ξ decreasing. But from fig. 2, we know it can't be decreased unlimited. So we choose ξ in the range of $\max|(\alpha-ki(\alpha)|\le 0.05$. Finally we select $\xi=-0.55$ when $\max|(\alpha-ki(\alpha)|=0.05$.

As we have addressed above, the scheme is 4th order central scheme when $\xi=-1/3$. Seen from fig. 3, the region is increased by 50% through choosing new ξ value. That is, for the same resolution, our grid's number is 2/3 of the original fourth-order scheme.

Table 1 x and ξ values for minimum $F(\xi)$ value

x	ξ
$\pi/2$	–0.49269
$2\pi/3$	–0.35613
$5\pi/6$	–0.07565

1.3.3 **Improve shock capturing ability by adding TVD limiter**

After above works, we have defined $\eta=0.2$ and $\xi=-0.55$. So far the scheme is linear which can't calculate shock. The following content is to add limiter. Flux can be split into $f=f^+ + f^-$, satisfying $\frac{df^+}{du}\ge 0$ and $\frac{df^-}{du}\le 0$. We consider Eq. (1) only for the case of $\frac{df}{du}=c=const>0$. For the case $c<0$, the scheme is easy to be developed due to the symmetry.

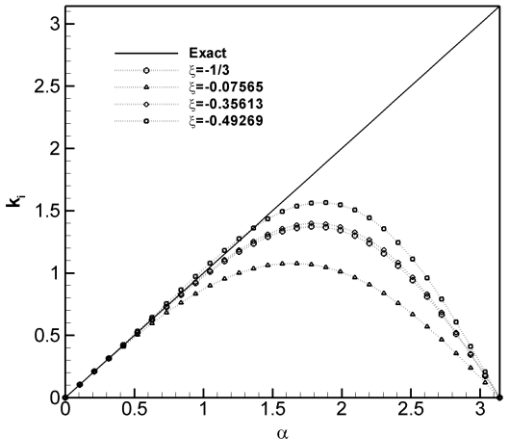

Fig. 1 Dispersion curves for different ξ

The stencil in the reconstruction of $U_{j+1/2}$ in classical MUSCL and the new scheme are shown in fig. 4. Classical MUSCL reconstructs left and right state by using three points. Totally four points are used to compute $U_{j+1/2}$, and the message in one additional point is not used, i.e. one point's message is wasted. This inspires us to reconstruct the left and right state by all four points:

$$U^L_{j+1/2}=U^L_{j+1/2}(U_{j-1},U_j,U_{j+1},U_{j+2})$$
$$U^R_{j+1/2}=U^R_{j+1/2}(U_{j-1},U_j,U_{j+1},U_{j+2})$$

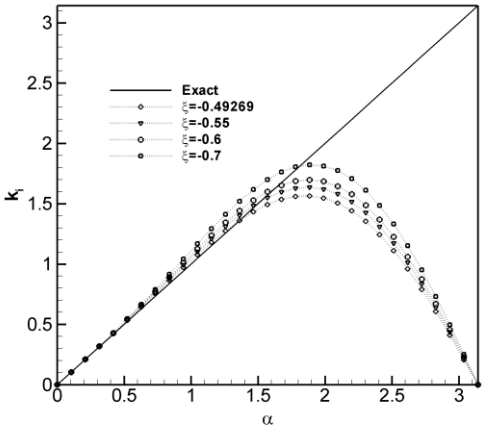

Fig. 2 Dispersion curves by decreasing ξ value

So, the new scheme uses the same stencil points in the computation of $U_{j+1/2}$ as that used in classical MUSCL scheme.

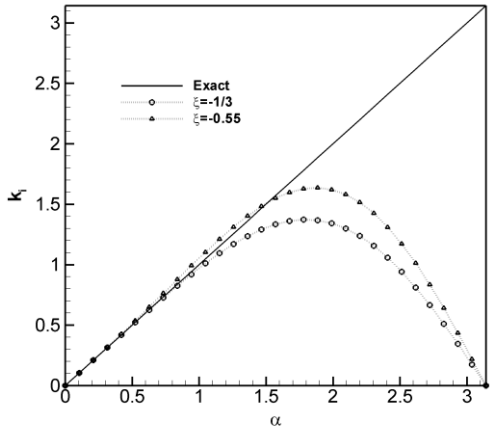

Fig. 3 Compared $\xi = -1/3$ to $\xi = -0.55$

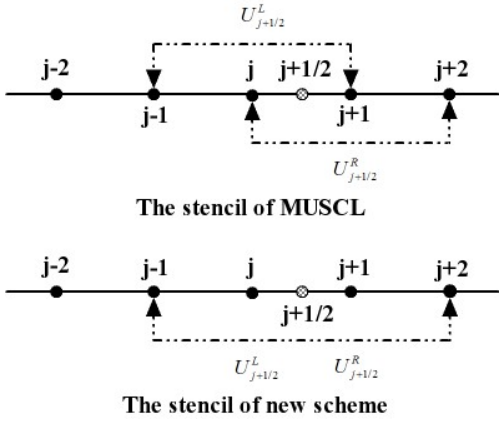

Fig. 4 Stencils for the reconstruction of $U_{j+1/2}$

According to the above idea, from (10) we obtain

$$\hat{f}^+_{j+1/2} = \frac{\xi-\eta}{4} f^+_{j-1} + \frac{3\eta-\xi+2}{4} f^+_j + \frac{2-3\eta-\xi}{4} f^+_{j+1} + \frac{\eta+\xi}{4} f^+_{j+2} \tag{13}$$

where $\xi = -0.55$ and $\eta = 0.2$, which are chosen by above optimization process.

Based on the limiter technique [17], scheme (13) can be rewritten as a first-order upwind part and a correction part, and then limiter factor are used in the correction part to keep TVD property. Now, the schemes are:

$$\hat{f}^+_{j+1/2} = f^+_j + \frac{1}{\phi_T}\tilde{\phi}_{j+1/2}(f^+_{j+1} - f^+_j)$$
$$\phi_{j+1/2} = 1-\eta+\frac{\eta+\xi}{2}\frac{1}{r_{j+3/2}}+\frac{\eta-\xi}{2}r_{j+1/2} \tag{14}$$

Here

$$r_{j+1/2} = \frac{f^+_j - f^+_{j-1}}{f^+_{j+1} - f^+_j}.$$

Reference [18] demonstrated that this method is identical to the TVD method and gave the range of $\phi_{j+1/2}$ is

$$\begin{cases} 0 \le \phi_{j+1/2} \le \dfrac{2}{1-\nu} \\ 0 \le \phi_{j+1/2} \le \dfrac{2r_{j+1/2}}{\nu} \end{cases}$$

where ν is CFL number, and the common value is $0 \le \nu \le 1$.

So，finally we choose:

$$\tilde{\phi}_{j+1/2} = \max(0, \min(2, \phi_{j+1/2}, 2r_{j+1/2}))$$

Now we obtain OMSCL2 scheme in the scalar hyperbolic conservation equation. The following will extend it to the Euler equations.

2 Extension to the Euler equation

In this section, we propose an extension of the OMUSCL2 method to the Euler equations. The one-dimension Euler equations of gas dynamics can be written as the following conservative form

$$\frac{\partial U}{\partial t} + \frac{\partial F(U)}{\partial x} = 0$$

Where

$$U = (\rho, \rho u, E)^T$$

is the vector of conservative variables and

$$F(U) = (\rho u, \rho u^2 + p, u(E+p))^T$$

is the vector of flux.

With the idea of MUSCL and TVD, we get the second-order optimized scheme (OMUSCL2). In order to apply to the finite volume method easily and to compare with MUSCL method, we rewrite it as similar as MUSCL method. Here we summarize how to apply this algorithm to the finite volume method.

- First, we use ***OMUSCL2*** scheme to compute the left and right state at the face of the control volume $U^L_{j+1/2}, U^R_{j+1/2}$:

$$
\begin{aligned}
&U^{L}_{j+1/2} = U_j + \frac{1}{2}\tilde{\phi}^{L}_{j+1/2}\Box(U_{j+1} - U_j)\\
&\tilde{\phi}^{L}_{j+1/2} = \max(0, \min(2, \phi^{L}_{j+1/2}, 2r^{L}_{j+1/2}))\\
&\phi^{L}_{j+1/2} = 0.8 - 0.175\frac{1}{r^{L}_{j+3/2}} + 0.375 r^{L}_{j+1/2}\\
&r^{L}_{j+1/2} = \frac{U_j - U_{j-1}}{U_{j+1} - U_j}, r^{L}_{j+3/2} = \frac{U_{j+1} - U_j}{U_{j+2} - U_{j+1}};
\end{aligned}
\qquad (15)
$$

$$
\begin{aligned}
&U^{R}_{j+1/2} = U_{j+1} - \frac{1}{2}\tilde{\phi}^{R}_{j+1/2}\Box(U_{j+1} - U_j)\\
&\tilde{\phi}^{R}_{j+1/2} = \max(0, \min(2, \phi^{R}_{j+1/2}, 2r^{R}_{j+1/2}))\\
&\phi^{R}_{j+1/2} = 0.8 - 0.175\frac{1}{r^{R}_{j-1/2}} + 0.375 r^{R}_{j+1/2}\\
&r^{R}_{j+1/2} = \frac{U_{j+2} - U_{j+1}}{U_{j+1} - U_j}, r^{R}_{j-1/2} = \frac{U_{j+1} - U_j}{U_j - U_{j-1}};
\end{aligned}
$$

- Then, the fluxes in the face of control volume can be computed by using flux technique, such as Steger-Warming splitting [19], Van Leer splitting [20], Roe [21] or AUSM [22] method.

3 Numerical tests

3.1 One dimension problems

In this section, we use this new scheme to solve some one-dimensional and two-dimensional tests on the purpose of comparing with the original MUSCL schemes.

In the following tests, original second-order MUSCL scheme (MUSCL2), third-order MUSCL scheme (MUSCL3) and the new scheme (OMUSCL2) are used to compute the face values, and Steger-Warming [19] method for one dimension tests and AUSM-PW [23] [24] method for two dimension tests with characteristic-wise are used to compute the flux, the third-order TVD type Runge-Kutta method are used for time advance.

3.1.1 Several convergence studies for the advection equations [8] [25]

We solve the following equation on the domain $[-1,1]$ with periodic boundary conditions.

$$
\begin{cases}
u_t + u_x = 0\\
u(x,0) = \sin(\pi x)
\end{cases}
$$

The computed L_1 error and order of accuracy are listed in Table 2. The error was measured at $t = 1$ with the CFL number is equal to 0.001. Where OMUSCL2 represents the current second-order optimized scheme; MUSCL2 and MUSCL3 represents the second-order and the third-order MUSCL scheme, respectively.

The results in table 2 tell us that the OMUSCL2 scheme gives the second-order accuracy, which meets the designation order. While the L_1 order of MUSCL2 and MUSCL3 is less than their theoretical order.

Table 2 L_1 error and order with $u(x,0) = \sin(\pi x)$

Method	N	L_1 error	L_1 order
MUSCL2	10	0.3515	-
	20	0.1316	1.42
	40	4.8703E-02	1.43
	80	1.4067E-02	1.79
	160	3.8667E-03	1.86
	320	1.0453E-03	1.89
MUSCL3	10	0.1867	-
	20	7.8412E-02	1.25
	40	1.8282E-02	2.10
	80	4.1645E-03	2.13
	160	8.7331E-04	2.25
	320	1.7762E-04	2.30
OMUSCL2	10	0.1910	-
	20	6.6878E-02	1.51
	40	2.1127E-02	1.66
	80	5.5547E-03	1.93
	160	1.3730E-03	2.02
	320	3.4528e-04	2.00

3.1.2 Shu-Osher problem [13]

This test represents that Mach 3 shock interacts with a density disturbance. And this is a good model to test the scheme's resolution for both shocks and fine scale waves. The governing equations are one-dimensional Euler equations and solved on the spatial domain $x \in [0,10]$. The initial conditions are

$$
\begin{cases}
\rho = 3.857143,\ u = 2.629369,\ p = 10.333333 & when\ x < 1\\
\rho = 1 + 0.2\sin(5x),\ u = 0,\quad p = 1 & when\ x \geq 1
\end{cases}
$$

We compute the solution up to $t = 1.8$ with 400 points. Since the real exact solution is unknown, the "exact" solution

here is obtained by 4000 points. In fig. 5 we compare the results for the OMUSCL2 and MUSCL schemes. fig. 5b is local enlarged plot of fig 5a. The figure shows clearly that OMUSCL2 has better resolution than original MUSCL schemes, especially, in the region of high wave number.

3.1.3 **One-dimensional Sod problem [26]**

The governing equations are one-dimensional Euler equations, and the computation domain is $x \in [0,1]$. The initial conditions are

$$\begin{cases} \rho = 1, & u = 0,\ p = 1 \quad when\ x < 0.5 \\ \rho = 0.125,\ u = 0,\ p = 0.1 & when\ x \geq 0.5 \end{cases}$$

We compute the solutions up to $t = 0.14$ with 200 points and compared with the exact solution, where the exact solution are computed by use an exact (Godunov) Riemann solver.

The density and velocity distribution obtained by MUSCL schemes and OMUSCL2 are shown in figs. 6-7. Where figs. 6b and 6c are locally enlarged plots of fig. 6; figs. 7b and 7c are locally enlarged plots of fig. 7. These figures show that the new scheme produces a better result than original MUSCL schemes.

3.1.4 **One-dimensional Lax problem [27][28]**

The governing equations are 1D Euler equations and solved on the spatial domain $x \in [0,2]$. The initial conditions are

$$\begin{cases} \rho = 0.445,\ u = 0.698,\ p = 3.528 & when\ x < 1 \\ \rho = 0.5, \quad u = 0, \quad p = 0.571 & when\ x \geq 1 \end{cases}$$

We compute the solution up to $t = 0.32$ with 100 points. The density distribution obtained by MUSCL and OMUSCL2 are shown in fig. 8, where fig. 8b is the local enlarged plot. From these figures we can observe that the best solutions are given by OMUSCL2, and the original MUSCL method has more dissipation around the discontinuities and non-physical oscillatory.

3.2 Two dimension problem

3.2.1 Double Mach reflection problem[29]

The governing equations are two-dimensional Euler equations, and the computational domain for this problem is chosen to be $[0,4]\times[0,1]$. Only the region $[0,3]\times[0,1]$ is used during the computing. The reflecting wall lies at the bottom of the computational domain starting from $x = 1/6$. Initially a right-moving Mach 10 shock in air ($\gamma = 1.4$) is positioned at $x = 1/6, y = 0$, and makes an 60° angle with the x-axis. For the bottom boundary, the region from $x = 0$ to $x = 1/6$ is always assigned the initial values. The boundary with $x > 1/6$ on the x-axis is taken to be a reflecting boundary. At the top boundary of our computational domain, the flow values are set to describe the exact motion of the Mach 10 shock. The problem was run with a CFL number of 0.6 and the results are shown at a simulation time of 0.2. The grid resolution was 960×240 points.

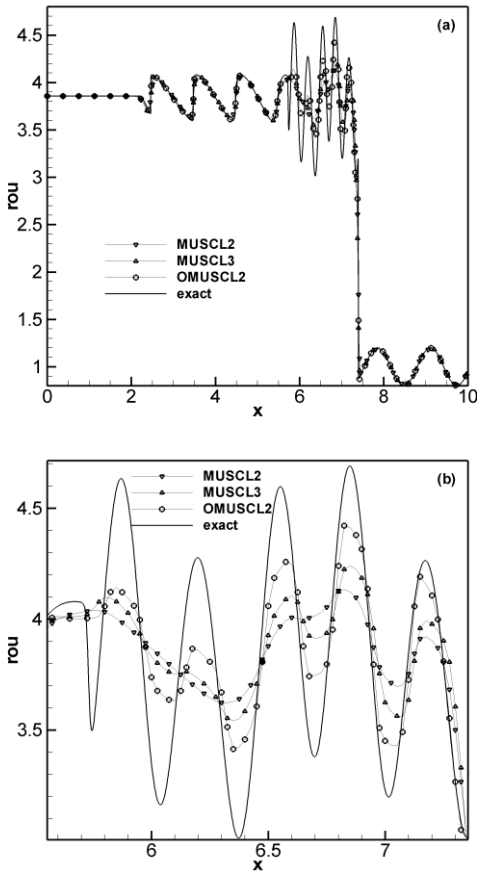

Fig. 5 Plots of density at t=1.8. $N = 400$

The density distribution obtained by OMUSCL2 is shown in fig. 9, where fig. 9c is the local enlarged plot. The result obtained by MUSCL3 is shown in fig 10. From fig. 9a, we can see that both Mach stems and shocks in this problem are properly captured. Compared these figures, it's clear that the new scheme (OMUSCL2) achieves a high-resolution in numerical solution, especially in the region near the Mach stems. The new scheme can capture the rollup of the slip lines which emanate from the head of head clearly. This result also shows that the dissipation of the new scheme is much smaller than that of MUSCL3.

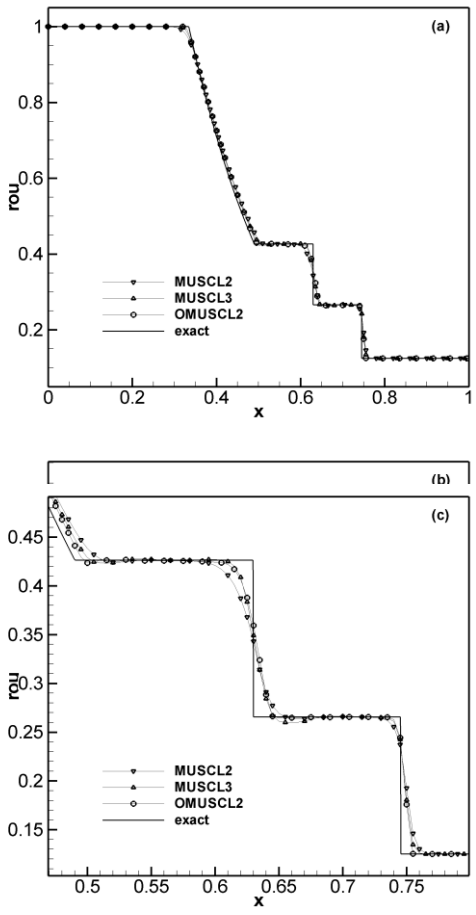

Fig. 6 Plots of density at t=0.14, $N = 200$

3.2.1 RAE2822 transonic airfoil [30]

The transonic flow over a RAE2822 airfoil is a classical validation test of CFD codes. This airfoil is transonic supercritical airfoil and there's a shock in the leeward side. Numerical solution of the shock position is sensitive to the numerical methods. So it's a good example to verify the shock resolution of numerical methods.

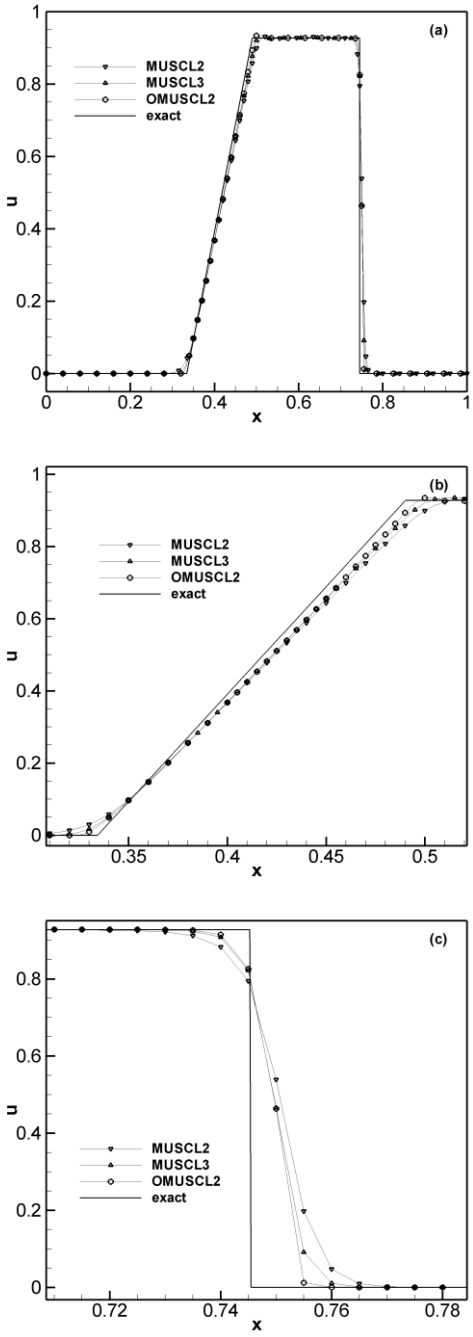

Fig. 7 Plots of velocity at t=0.14, $N = 200$

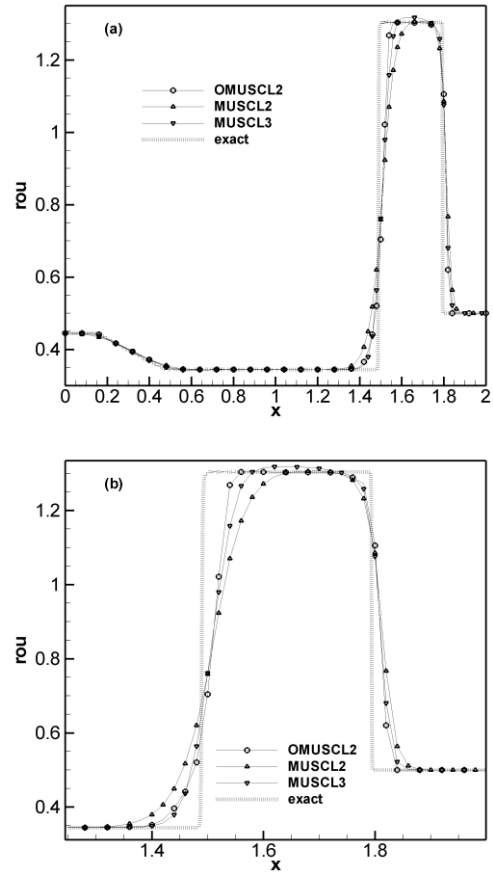

Fig. 8 Plots of velocity at t=0.32, $N=100$

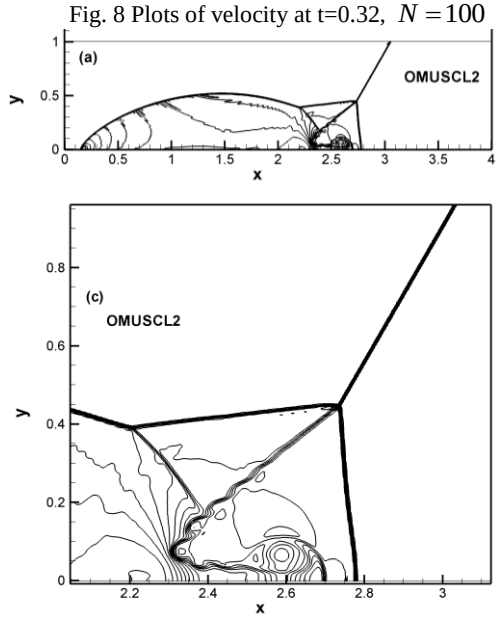

Figure 9 Contours of density (from 1.731 to 20.92 with 30 equally spaced contours), using OMUSCL2

The grid we use is the grid providing by J.W. Slater from NASA Web [31]. It is a multi-block C-grid, and the total mesh number is 369×69. Grid around the airfoil shown in figure 11. The free stream flow conditions are $Ma_\infty=0.729$; the chord-based Reynolds number is $\mathrm{Re}=6.5\times10^6$; angle of attack is 2.31° .The surface pressure coefficient for the comparison results from [30] (or downloads from the website [31]).

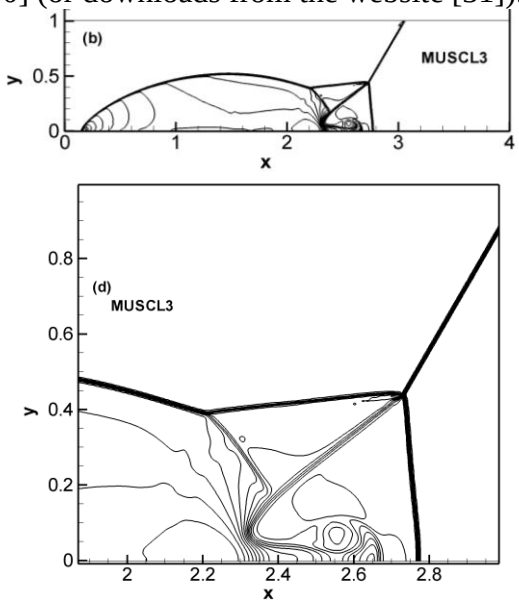

Figure 10 Contours of density (from 1.731 to 20.92 with 30 equally spaced contours), using MUSCL3

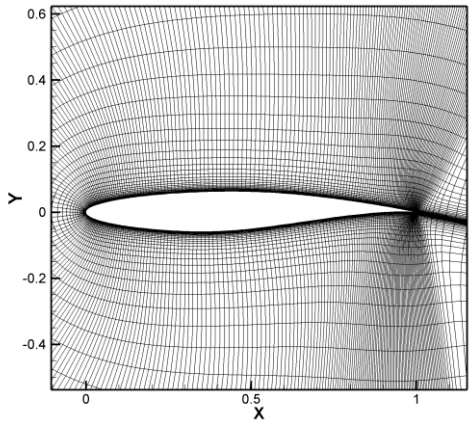

Fig. 11 RAE2822 computational grids

Figure 12 shows the distribution of the pressure coefficient on the surface of the airfoil. This figure shows that the result of OMUSCL2 is most close to the experiential data, and is better than MUSCL3 and MUSCL2. Epically, the shock's location computed by OMUSCL2 agrees very well to the experiential data. That shows that OMUSCL2 has a higher shock resolution

and lower dissipation than original second-order or third-order MUSCL schemes.

Table 3 gives the CPU time per 1000 steps in this test. The CPU is Intel i7-920 at 2.66GHz. This table shows that OMUSCL2's computational cost is nearly the same as that of the MUSCL schemes. The new scheme's CPU cost is only 8% more than that of that of MUSCL2 and only 4% more than that of MUSCL3. Therefore, we need not to pay much more CPU cost when we change original MUSCL scheme to OMUSCL2 in CFD codes.

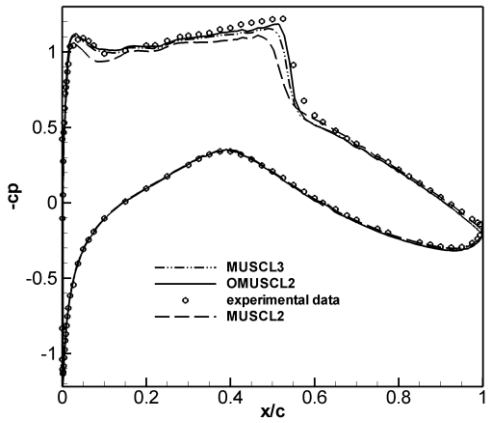

Fig. 12 Distribution of the pressure coefficient on the surface of airfoil

Table 3 CPU time comparison

Scheme	Computational efficiency (per 1000 steps)
MUSCL2	207s
MUSCL3	215s
OMUSCL2	224s

4 Conclusion

A second-order monotonicity-preserving optimized MUSCL scheme (OMUSCL2) is developed based on dispersion and dissipation optimization and monotonicity-preserving technique. The new scheme (OMUSCL2) has simple expression and is easy to be used in CFD codes. Compared with the original second-order or third-order MUSCL scheme, the new scheme shows nearly the same CPU costs and higher resolution to shockwaves and small-scale waves.

We test the new scheme through a set of one-dimensional and two-dimensional tests, including the Shu-Osher problem, the sod problem, the Lax problem, two dimension double Mach reflection and RAE2822 transonic airfoil test. All numerical tests show that, compared with original MUSCL schemes, the new scheme has less dispersion and less dissipation error and higher resolution.

Thanks to Professor Ren Yuxin in Tsinghua University for the helpful discussion of the optimization method. Thanks to the Supercomputing Center of Chinese Academy of Sciences (SCCAS) and Shanghai Supercomputing Center (SSC) for proving the computing time. This work was supported by the National Nature Science Foundation of China (Nos.10632050, 10872205, 11072248), the 973 project (Grant Nos. 2009CB724100), the 863 program (No. 2009AA010A139), and project of CAS INFO-115-B0.

参考文献

1 Harten A. High resolution schemes for hyperbolic conservation laws. J. Comput. Phys., 1983,49: 357-393

2 Fu D X. Direct numerical simulation of compressible turbulence. Beijing: Science Press, 2010

3 M. C., M.T., Compact third-order limiter functions for finite volume methods. J. Comput. Phys., 2009, 228: 4118-4145

4 Sweby P K. High resolution schemes using flux limiters for hyperbolic conservation laws. SIAMJ Numer Anal, 1984, 21: 995-1011

5 Van Leer B. Towards the ultimate conservation difference scheme V: A second-order sequel to Godunov's Method. J. Comput. Phys., 1979,32:101-136

6 Harten A, Engquist B. Osher S, et al. Uniformly high order accurate essentially non-oscillatory shock-capturing schemes Ⅲ. J. Comput. Phys., 1987, 71: 231-303

7 Serna S, Marquina A. Powe ENO methods: A fifth-order accurate weighted power ENO method. J. Comput. Phys., 2004, 194: 632-658

8 Jiang G S, Shu C W. Efficient implementation of weighted ENO

schemes. J Comp Phys, 1996, 126: 202-228

9 Shu C W. TVB uniformly high-order schemes for conservation laws. Math Comput, 1987, 49: 105-121

10 Suresh A, Huynh H T. Accurate monotonicity-preserving schemes with Runge-Kutta time stepping. J. Comput. Phys., 1997, 136: 83-99

11 C. K. W. Tam and J. C. Webb, Dispersion-relation-preserving finite difference schemes for computational acoustics, *J. Comput. Phys.* 107, 262 (1993).

12 S. K. Lele, Compact finite difference schemes with spectral-like resolution, *J. Comput. Phys.* **103**, 16 (1992).

13 Shu C W., Osher S. Efficient implementation of essentially non-oscillatory schemes. J. Comput. Phys., 1989, 83: 32-78

14 Godunov C K. Difference method for computing the discontinuity in fluid dynamics. Math Sbornik, 1959, 47(3): 271

15 Jennings G. Discrects shock. Comm Pore & Appl Math. 1974, 27: 25-37

16 Fu D X, Ma Y w, Computational Fluid Dynamics (in China). Beijing: High Education Press, 2002

17 He Z W, Li X L, Fu D X, Ma Y w, A 5th order monotonicity-preserving upwind compact difference scheme. Science China. 2011, 54: 1-12

18 Daru V, Tenaud C. High order one-step monotonicity-preserving schemes for unsteady compressible flow calculations. J. Comput. Phys., 2004, 193: 563-594

19 Steger J L., Warming R F., Flux vector splitting of the inviscid gasdynamic equations with applications to finite difference methods. J. Comput. Phys., 1981, 40: 263-293

20 Van Leer B., Flux-vector splitting for the Euler equations. Technical Report ICASE 82-30, NASA Langley Research Center, USA, 1982

21 Roe P. L., Approximate Riemann solvers, parameter vectors and difference schemes. J. Comput. Phys., 1981, 43: 357-372

22 Liou M.S. Steffen J.C.J., A new flux splitting scheme. J. Comput. Phys. 1993,107:23-39

23 K. H. Kim and O. H. Rho. An improvement of AUSM schemes by introducing the pressure-based weight function, Comput. Fluids 27(3), 311 (1998)

24 K. H. Kim and O. H. Rho. An improvement of AUSM schemes by introducing the pressure-based weight function, in The Fifth Annual Conference of the Computational Fluid Dynamics Society of Canada (CFD 97), Vol. 5, pp. 14-33-14-38, 1997

25 Shu C W, Dinshaw S. Balsara. Monotonicity preserving weighted essentially non-oscillatory schemes with increasingly high order of accuracy. J. Comput. Phys., 2000, 160: 405-452

26 Sod G A. A survey of several finite difference mrthods for system of non-linear hyperbolic conservation law. J. Comput. Phys., 1978, 21(1): 1-31

27 Shu C W. Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws. NASA/CR-97-206253, ICASE Report No. 97-65, 1997

28 E.F. Toro, Riemann Solvers and Numerical Methods for Fluid Dynamics, Second ed., Springer, Berlin, Germany, 199

29 P. Woodward, P. Collela, The numerical simulation of two-dimensional fluid flow with strong shocks. J. Comput. Phys. 1984, 54: 115-173

30 Cook P. H., M. A. McDonal, M. C. P. Firmin, "Aerofoil RAE2822-Pressure Distributions, and Boundary Layer and Wake measurements," Experimental Data base for Computer Program Assessment, AGARD Report AR 138, 1979

31 http://www.grc.nasa.gov/WWW/wind/valid/raetaf/raetaf.html