NASA/CR-1998-208459
ICASE Report No. 98-32

Weighted Essentially Non-Oscillatory Schemes on Triangular Meshes

Changqing Hu and Chi-Wang Shu
Brown University, Providence, Rhode Island

Institute for Computer Applications in Science and Engineering
NASA Langley Research Center
Hampton, VA

Operated by Universities Space Research Association

National Aeronautics and
Space Administration

Langley Research Center
Hampton, Virginia 23681-2199

Prepared for Langley Research Center under
Contract NAS1-19480 and NAS1-97046

July 1998

WEIGHTED ESSENTIALLY NON-OSCILLATORY SCHEMES ON TRIANGULAR MESHES*

CHANGQING HU† AND CHI-WANG SHU‡

Abstract. In this paper we construct high order weighted essentially non-oscillatory (WENO) schemes on two dimensional unstructured meshes (triangles) in the finite volume formulation. We present third order schemes using a combination of linear polynomials, and fourth order schemes using a combination of quadratic polynomials. Numerical examples are shown to demonstrate the accuracies and robustness of the methods for shock calculations.

Key words. weighted essentially non-oscillatory schemes, unstructured mesh, high order accuracy, shock calculations.

Subject classification. Applied and Numerical Mathematics

1. Introduction. ENO (essentially non-oscillatory) schemes (Harten, Osher, Engquist and Chakravarthy [16], Shu and Osher [28, 29]) have been successfully applied to solve hyperbolic conservation laws and other convection dominated problems, for example in simulating shock turbulence interactions, Shu and Osher [29], Shu, Zang, Erlebacher, Whitaker and Osher [30], and Adams and Shariff [2]; in the direct simulation of compressible turbulence, Shu, Zang, Erlebacher, Whitaker and Osher [30], Walsteijn [35], and Ladeinde, O'Brien, Cai and Liu [20]; in solving the relativistic hydrodynamics equations, Dolezal and Wong [8]; in shock vortex interactions and other gas dynamics problems, Casper and Atkins [6] and Erlebacher, Hussaini and Shu [10]; in incompressible flow calculations, E and Shu [9] and Harabetian, Osher and Shu [13]; in solving the viscoelasticity equations with fading memory, Shu and Zeng [31]; in semi-conductor device simulation, Fatemi, Jerome and Osher [11] and Jerome and Shu [17], [18]; in image processing and level set methods, Osher and Sethian [24], Sethian [26], and Siddiqi, Kimia and Shu [32]. The original ENO paper by Harten, Osher, Engquist and Chakravarthy [16] was for a one dimensional finite volume formulation. Later, this finite volume formulation of ENO schemes has been extended to two dimensional structured meshes by Harten [14] and by Casper [5], and to unstructured triangular meshes by Abgrall [1], Harten and Chakravarthy [15], and Sonar [34]. Finite volume ENO schemes based on a staggered grid and Lax-Friedrichs formulation were given in Bianco, Puppo and Russo [4]. Although finite difference versions of ENO schemes [28, 29] are more efficient for multidimensional calculations, finite volume schemes have the advantage of easy handling of complicated geometry by arbitrary triangulations.

WENO schemes were developed later to improve upon ENO schemes, in Liu, Osher and Chan [23] and Jiang and Shu [19]. Advantages of WENO schemes over ENO include the smoothness of numerical fluxes, better steady state convergence, and better accuracy using the same stencils. Levy, Puppo and Russo [22] designed one dimensional finite volume WENO schemes based on a staggered grid and Lax-Friedrichs formulation.

For a review of ENO and WENO schemes, see [27].

*Research was supported in part by ARO grant DAAG55-97-1-0318, NSF grants DMS-9500814, ECS-9627849 and INT-9601084, NASA Langley grant NAG-1-1145 and Contract Nos. NAS1-19480 and NAS1-97046 while the second author was in residence at ICASE, NASA Langley Research Center, Hampton, VA 23681-2199, and AFOSR grant F49620-96-1-0150.

†Division of Applied Mathematics, Brown University, Providence, RI 02912. E-mail: hu@cfm.brown.edu

‡Division of Applied Mathematics, Brown University, Providence, RI 02912. E-mail: shu@cfm.brown.edu

Recently, Friedrich [12] constructed WENO schemes on unstructured meshes using a co-volume formulation as in Abgrall [1]. The WENO schemes in [12] only achieve the same order of accuracy as the corresponding ENO schemes when the same set of stencils is considered. This is not optimal as was known in Jiang and Shu [19] for structured meshes.

In this paper, we present higher order WENO schemes on triangular meshes when using the same set of ENO stencils. We will construct third order schemes using a combination of two-dimensional linear polynomials, and fourth order schemes using a combination of two-dimensional quadratic polynomials.

We will first sketch the procedure to construct the high order linear schemes. The formulation at this stage is important to accommodate nonlinear WENO weights later. We then describe the third and fourth order WENO schemes. Numerical examples will be given, to demonstrate the accuracy and resolution of the constructed schemes. Concluding remarks are included at the end.

2. Finite Volume Formulation. In this paper we solve the following two-dimensional conservation law:

$$\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} + \frac{\partial g(u)}{\partial y} \;=\; 0 \tag{2.1}$$

using the finite volume formulation. Computational control volumes are simply triangles.

Taking the triangle $\triangle_i$ as our control volume, we formulate the semi-discrete finite volume scheme of equation (2.1) as:

$$\frac{d}{dt}\bar{u}_i(t) + \frac{1}{|\triangle_i|}\int_{\partial\triangle_i} F\cdot n\, ds \;=\; 0 \tag{2.2}$$

where $\bar{u}_i(t)$ is the cell average of u on the cell $\triangle_i$, $F = (f, g)^T$, and n is the outward unit normal of the triangle boundary $\partial\triangle_i$.

The line integral in (2.2) is discretized by a q-point Gaussian integration formula,

$$\int_{\Gamma_k} F\cdot n\, ds \;\approx\; |\Gamma_k| \sum_{j=1}^{q} \omega_j F(u(G_j,t))\cdot n \tag{2.3}$$

and $F(u(G_j,t))\cdot n$ is replaced by a numerical flux. The simple Lax-Friedrichs flux is used in all our numerical experiments, which is given by

$$F(u(G_j,t))\cdot n \;\approx\; \frac{1}{2}\left[\left(F(u^-(G_j,t)) + F(u^+(G_j,t))\right)\cdot n - \alpha\left(u^+(G_j,t) - u^-(G_j,t)\right)\right] \tag{2.4}$$

where α is taken as an upper bound for the eigenvalues of the Jacobian in the n direction, and u^- and u^+ are the values of u inside the triangle and outside the triangle (inside the neighboring triangle) at the Gaussian point.

Since we are constructing schemes up to fourth order accuracy, two point Gaussian $q = 2$ is used, which has $G_1 = cP_1 + (1-c)P_2$, $G_2 = cP_2 + (1-c)P_1$, $c = \frac{1}{2} + \frac{\sqrt{3}}{6}$ and $\omega_1 = \omega_2 = \frac{1}{2}$ for the line with end points P_1 and P_2.

3. Reconstruction and Linear Schemes. Let P^k denote the set of two-dimensional polynomials of degree less than or equal to k. The reconstruction problem, from cell averages to point values, is as follows: given a smooth function u, and a triangulation with triangles $\{\triangle_0, \triangle_1, ..., \triangle_N\}$, we would like to construct,

for each triangle $\triangle_i$, a polynomial $p(x,y)$ in P^k that has the same mean value as u on $\triangle_i$, and is an $(n+1)$-th order approximation to u on the cell $\triangle_i$. The mean value of a function $u(x,y)$ on a cell $\triangle_i$ is defined as

$$\bar{u}_i \equiv \frac{1}{|\triangle_i|}\int_{\triangle_i} u(x,y)dxdy. \tag{3.1}$$

In order to determine $K = \frac{(k+1)(k+2)}{2}$ degrees of freedom in a k-th degree polynomial p, we need to use the information of at least K triangles. In addition to $\triangle_i$ itself, we take its $K-1$ neighboring cells, and we rename these K triangles as $S_i = \{\Omega_1, \Omega_2, ..., \Omega_K\}$, S_i is called a stencil for the triangle $\triangle_i$. If we require that p has the mean value $\bar{u}_j$ on Ω_j for all $1 \le j \le K$, we will get a $K \times K$ linear system. If this linear system has a unique solution, S_i is called an *admissible* stencil. Of course, in practice, we also have to worry about any ill conditioned linear system even if it is invertible. For linear polynomials $k = 1$, a stencil formed by $\triangle_i$ and two of its neighbors is admissible for most triangulations.

3.1. Third order reconstruction. To construct a third order linear scheme (a scheme is called linear if it is linear when applied to a linear equation with constant coefficients) as a starting point for the WENO procedure, we need a quadratic polynomial reconstruction. It seems that the most robust way is the least square reconstruction suggested by Barth and Frederickson [3]. For the control volume of triangle $\triangle_0$, see Fig. 3.1, let $\triangle_i, \triangle_j, \triangle_k$ be its three neighbors, and $\triangle_{ia}, \triangle_{ib}$ be the two neighbors (other than $\triangle_0$) of $\triangle_i$, and so on. We determine the quadratic polynomial p^2 by requiring that p^2 has the same cell average as u on $\triangle_0,$, and also p^2 has the same cell average as u on each triangle in the set

$$\{\triangle_i, \triangle_{ia}, \triangle_{ib}, \triangle_j, \triangle_{ja}, \triangle_{jb}, \triangle_k, \triangle_{ka}, \triangle_{kb}\},$$

but only in a *least-square* sense (as this is an over-determined system). Notice that some of the neighbors' neighbors ($\triangle_{ia}$, $\triangle_{ib}$, $\triangle_{ja}, ...$) may coincide. For example, $\triangle_{ib}$ might be the same as $\triangle_{ja}$. This, however, does not affect the least square procedure to determine p^2.

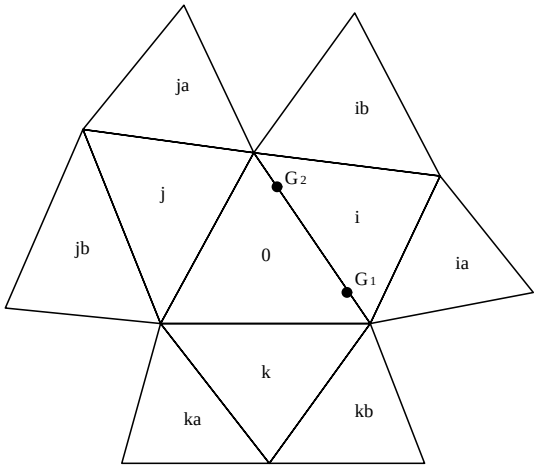

FIG. 3.1. *A typical stencil.*

A key step to build a high order WENO scheme based on lower order polynomials is carried out in the following. We want to construct several linear polynomials whose weighted average will give the same result as the quadratic reconstruction p^2 at each quadrature point (the weights are different for different quadrature points).

We first build the following 9 linear polynomials by agreeing with the cell averages of u on the following stencils: p_1 (on triangles: $0, j, k$), p_2 (on triangles: $0, k, i$), p_3 (on triangles: $0, i, j$), p_4 (on triangles: $0, i, ia$), p_5 (on triangles: $0, i, ib$), p_6 (on triangles: $0, j, ja$), p_7 (on triangles: $0, j, jb$), p_8 (on triangles: $0, k, ka$), and p_9 (on triangles: $0, k, kb$).

For each quadrature point (x^G, y^G), we want to find the linear weights such that the linear combination of these p_s

$$R(x, y) = \sum_{s=1}^{9} \gamma_s p_s(x, y) \tag{3.2}$$

satisfies $R(x^G, y^G) = p^2(x^G, y^G)$ with p^2 defined before using the least squares procedure. Notice that there are in general 10 equations to be satisfied in order to make $R(x^G, y^G) = p^2(x^G, y^G)$ hold for any u. This is because both R and p^2 depend on the 10 cell averages of u on the 10 triangles in the stencil. Note that there are situations when ia, ib, ja, jb, ka, kb might not be distinct, in these cases, we simply discard some of the p_s, or just set the corresponding coefficient γ_s to zero. For example, if $ib = ja$, we will just use $p_1, p_2, p_3, p_4, p_5, p_7, p_8, p_9$ and discard p_6. In this case there is one fewer coefficient but also one fewer condition to satisfy for $R(x^G, y^G) = p^2(x^G, y^G)$, as there is one fewer triangle in the stencil.

To illustrate the detail of the procedure, we will take (x^G, y^G) as the first quadrature point on side i, the interface of $\triangle_0$ and $\triangle_i$, i.e. point G_1 in Fig. 3.1. Without loss of generality we will assume that ia, ib, ja, jb, ka, kb are distinct. In order to make $R(x^G, y^G) = p^2(x^G, y^G)$ for any u, first we use the 6 equalities for coefficients of the triangles ia, ib, ja, jb, ka, kb (i.e. to satisfy $R(x^G, y^G) = p^2(x^G, y^G)$ for the case when u has a cell average 1 on the triangle ia and cell averages 0 on all other triangles in the stencil, etc.), This will give us 6 equations. We then will use any two of the three equalities for the coefficients of triangles i, j, k which will give us two additional equations. Since p_1, p_2, p_3 are linearly dependent and from the consistency (all polynomials are constants if u is a constant function), we claim that the other two equalities (for cell 0 and one of i, j, k) will be satisfied automatically if the above 8 equations are satisfied. Numerical experiments for various triangulations confirm our claim.

We now have obtained a linear system of 8 equations with 9 unknowns, hence we still have one free parameter (one of $\gamma_1, \gamma_2, \gamma_3$). We would like to use this freedom to get a set of non-negative solutions, which is important for the WENO procedure to be developed later for shock calculations. Unfortunately, it turns out that, for many triangulations, it is impossible to do so. Some regrouping is needed and will be discussed later. Without worrying about positivity for the moment, we can choose the free parameter in any fashion. We will take $\gamma_1 = 0$ for side i (similarly $\gamma_2 = 0$ for side j, and $\gamma_3 = 0$ for side k). We thus have obtained all the combination coefficients γ_k $(k = 1, ..., 9)$ in (3.2) such that $R(x^G, y^G) = p^2(x^G, y^G)$ for any u, i.e. the linear polynomial $R(x, y)$ in (3.2) approximates any smooth function u to third order accuracy at the point (x^G, y^G).

3.2. Fourth order reconstruction. To construct a fourth order linear scheme as a starting point for the WENO procedure, we need a cubic polynomial reconstruction, which has 10 degrees of freedom. We thus only consider the case where ia, ib, ja, jb, ka, kb are distinct in the stencil (see Fig. 3.1). We construct the cubic polynomial p^3 by requiring that its cell average agrees with that of u on each triangle in the 10-triangle stencil shown in Fig. 3.1. It seems that for most triangulations this reconstruction is possible.

Again, the key step to build a high order WENO scheme based on lower order polynomials is carried out in the following. We would like to construct several quadratic polynomials whose weighted average will give the same result as the cubic reconstruction p^3 at each quadrature point (the weights are different for

different quadrature points). The following 6 quadratic polynomials are constructed by having the same cell averages as u on the corresponding triangles:
q_1 (on triangles: $0, i, ia, ib, k, kb$), q_2 (on triangles: $0, i, ia, ib, j, ja$),
q_3 (on triangles: $0, j, ja, jb, i, ib$), q_4 (on triangles: $0, j, ja, jb, k, ka$),
q_5 (on triangles: $0, k, ka, kb, j, jb$), q_6 (on triangles: $0, k, ka, kb, i, ia$).

For each quadrature point (x^G, y^G), we would like to find the linear weights such that the linear combination of these q_s

$$Q(x, y) = \sum_{s=1}^{6} \gamma_s q_s(x, y) \tag{3.3}$$

satisfies $Q(x^G, y^G) = p^3(x^G, y^G)$ for all u.

The above 6 quadratic polynomials are linearly dependent, but any 5 of them are linearly independent. We thus set one of the coefficients γ_s to be 0, and determine the other 5 to satisfy $Q(x^G, y^G) = p^3(x^G, y^G)$. This looks like an over-determined problem, as there are 10 equations to satisfy but only 5 coefficients. Taking for example $(q_2, q_3, q_4, q_5, q_6)$ for the linear combination, we need 10 equations for the 10 cell averages in the stencil, but similar to the third order case, numerical experiments for various triangulations show that there are actually only 5 independent equations among these 10, which means that we need only to establish 5 equations by the equalities of the coefficients of cell averages at any 5 triangles from the stencil, say, (ib, ja, jb, ka, kb), and the other 5 equations from triangles $(0, i, j, k, ia)$ will be satisfied automatically.

Since we can pick any one of the γ_s to be 0, we have 6 different possible combinations. We also can take any average of these 6 combinations to obtain a new combination with the same accuracy.

3.3. Accuracy test for the linear schemes. From the third and the fourth order reconstructions, we can now obtain the third and the fourth order linear schemes for (2.1) by replacing $u^-(G_j, t)$ in (2.4) with the reconstructed values $R(x^G, y^G)$ in (3.2) or $Q(x^G, y^G)$ in (3.3), respectively. Similarly, $u^+(G_j, t)$ is replaced with the reconstructed values in the neighboring cell $\triangle_i$.

For the temporal discretization, the third order TVD Runge-Kutta scheme of Shu and Osher [28] is used. For the fourth order scheme, we use $\Delta t = (\Delta x)^{4/3}$ to achieve fourth order accuracy in the time, but only for the examples of the accuracy test.

Example 3.1. Two-dimensional linear equation:

$$u_t + u_x + u_y = 0 \tag{3.4}$$

with the initial condition $u_0(x, y) = \sin(\frac{\pi}{2}(x + y))$, $-2 \le x \le 2$, $-2 \le y \le 2$, and periodic boundary conditions.

We first use uniform triangular meshes which are obtained by adding one diagonal line in each rectangle, shown in Fig. 3.2 for the coarsest case $h = \frac{2}{5}$. In Table 3.1, the accuracy results are shown for both the third order scheme (from the combination of linear polynomials) and the fourth order scheme (from the combination of quadratic polynomials), where h is the length of the rectangles, at $t = 2.0$. The errors presented are those of the cell averages of u.

We then use non-uniform meshes, shown in Fig. 3.3 for the coarsest case $h = h_0 = 1$, where h is just an average length. The refinement of the meshes is done in a uniform way, namely by cutting each triangle into 4 smaller similar ones. The accuracy result is shown in Table 3.2.

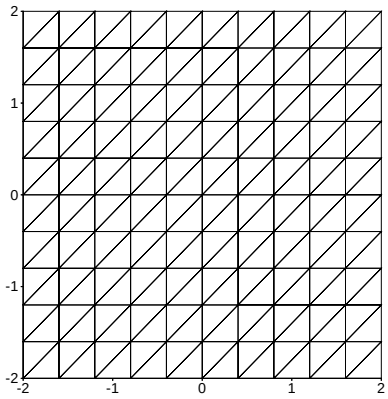

FIG. 3.2. *Uniform mesh with $h = \frac{2}{5}$ for accuracy test.*

TABLE 3.1
Accuracy for 2D linear equation, uniform meshes, linear schemes.

	P^1 (3rd order)				P^2 (4th order)			
h	L^1 error	order	L^∞ error	order	L^1 error	order	L^∞ error	order
2/5	1.80E-01	—	2.79E-01	—	1.40E-02	—	2.17E-02	—
1/5	2.81E-02	2.68	4.37E-02	2.68	9.11E-04	3.94	1.41E-03	3.94
1/10	3.65E-03	2.95	5.72E-03	2.93	5.57E-05	4.03	8.72E-05	4.02
1/20	4.60E-04	2.99	7.22E-04	2.99	3.43E-06	4.02	5.39E-06	4.02
1/40	5.76E-05	3.00	9.05E-05	3.00	2.12E-07	4.02	3.34E-07	4.01
1/80	7.21E-06	3.00	1.13E-05	3.00	1.32E-08	4.01	2.07E-08	4.01

Example 3.2. Two-dimensional Burgers' equation:

$$u_t + \left(\frac{u^2}{2}\right)_x + \left(\frac{u^2}{2}\right)_y = 0 \tag{3.5}$$

with the initial condition $u_0(x, y) = 0.3+0.7\sin(\frac{\pi}{2}(x+y))$, $-2 \le x \le 2$, $-2 \le y \le 2$, and periodic boundary conditions.

We first use the same uniform triangular meshes as in Example 3.1, shown in Fig. 3.2 for the coarsest case $h = \frac{2}{5}$. In Table 3.3, the accuracy results are shown for both the third order scheme and the fourth order scheme, at $t = 0.5/\pi^2$ when the solution is still smooth. The errors presented are those of the point values at the 6 quadrature points of each triangle.

We then use the same non-uniform meshes as in Example 3.1, shown in Fig. 3.3 for the coarsest case. The accuracy result is shown in Table 3.4.

Example 3.3. Two-dimensional vortex evolution problem for the Euler equations. See [27] for a description of this problem. We consider the compressible Euler equations of gas dynamics:

$$\xi_t + f(\xi)_x + g(\xi)_y = 0 \tag{3.6}$$

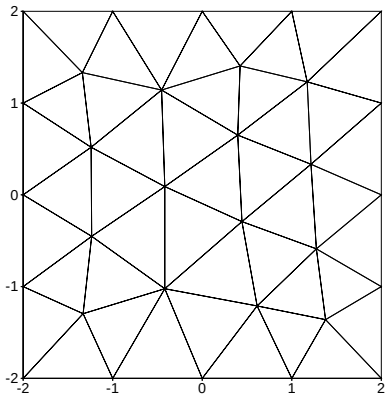

FIG. 3.3. *Non-uniform mesh with $h = 1$ for accuracy test.*

TABLE 3.2
Accuracy for 2D linear equation, non-uniform meshes, linear schemes.

	P^1 (3rd order)				P^2 (4th order)			
h	L^1 error	order	L^∞ error	order	L^1 error	order	L^∞ error	order
$h_0/2$	1.21E-01	—	2.25E-01	—	4.95E-03	—	1.73E-02	—
$h_0/4$	1.81E-02	2.74	3.74E-02	2.59	2.90E-04	4.09	1.42E-03	3.61
$h_0/8$	2.36E-03	2.94	5.39E-03	2.80	2.21E-05	3.71	8.32E-05	4.09
$h_0/16$	3.00E-04	2.98	7.19E-04	2.91	1.29E-06	4.10	5.09E-06	4.03
$h_0/32$	3.78E-05	2.99	9.40E-05	2.94	7.76E-08	4.06	3.16E-07	4.01
$h_0/64$	4.75E-06	2.99	1.22E-05	2.95	4.75E-09	4.03	1.95E-08	4.02

where

$$\xi = (\rho, \rho u, \rho v, E),$$

$$f(\xi) = (\rho u, \rho u^2 + p, \rho uv, u(E + p)),$$

$$g(\xi) = (\rho v, \rho uv, \rho v^2 + p, v(E + p)).$$

Here ρ is the density, (u, v) is the velocity, E is the total energy, p is the pressure, and

$$E = \frac{p}{\gamma - 1} + \frac{1}{2}\rho(u^2 + v^2)$$

with $\gamma = 1.4$.

The mean flow is $\rho = 1$, $p = 1$, and $(u, v) = (1, 1)$. We add, to the mean flow, an isentropic vortex (perturbations in (u, v) and the temperature $T = \frac{p}{\rho}$, no perturbation in the entropy $S = \frac{p}{\rho^\gamma}$):

$$(\delta u, \delta v) = \frac{\epsilon}{2\pi} e^{0.5(1-r^2)}(-\overline{y}, \overline{x})$$

TABLE 3.3
Accuracy for 2D Burgers' equation, uniform meshes, linear schemes.

	P^1 (3rd order)				P^2 (4th order)			
h	L^1 error	order	L^∞ error	order	L^1 error	order	L^∞ error	order
2/5	2.67E-02	—	7.75E-02	—	8.63E-03	—	2.18E-02	—
1/5	3.65E-03	2.87	1.16E-02	2.74	6.08E-04	3.83	1.70E-03	3.68
1/10	4.60E-04	2.99	1.52E-03	2.93	3.97E-05	3.94	1.16E-04	3.87
1/20	5.75E-05	3.00	1.91E-04	2.99	2.51E-06	3.98	7.37E-06	3.98
1/40	7.18E-06	3.01	2.38E-05	3.01	1.57E-07	4.00	4.62E-07	4.00
1/80	8.96E-07	3.00	2.97E-06	3.00	9.83E-09	4.00	2.89E-08	4.00

TABLE 3.4
Accuracy for 2D Burgers' equation, non-uniform meshes, linear schemes.

	P^1 (3rd order)				P^2 (4th order)			
h	L^1 error	order	L^∞ error	order	L^1 error	order	L^∞ error	order
$h_0/2$	1.69E-02	—	7.95E-01	—	3.96E-03	—	1.88E-02	—
$h_0/4$	2.23E-03	2.92	1.23E-02	2.69	2.87E-04	3.79	2.17E-03	3.12
$h_0/8$	2.84E-04	2.97	1.69E-03	2.86	1.90E-05	3.92	1.81E-04	3.58
$h_0/16$	3.57E-05	2.99	2.22E-04	2.93	1.20E-06	3.99	1.34E-05	3.77
$h_0/32$	4.48E-06	2.99	3.00E-05	2.89	7.57E-08	3.99	1.00E-06	3.74
$h_0/64$	5.63E-07	2.99	4.26E-06	2.82	4.75E-09	4.00	7.57E-08	3.72

$$\delta T = -\frac{(\gamma-1)\epsilon^2}{8\gamma\pi^2} e^{1-r^2}, \qquad \delta S = 0,$$

where $(\overline{x}, \overline{y}) = (x-5, y-5)$, $r^2 = \overline{x}^2 + \overline{y}^2$, and the vortex strength $\epsilon = 5$.

The computational domain is taken as [0,10]×[0,10], and periodic boundary conditions are used.

It is clear that the exact solution of the Euler equation with the above initial and boundary conditions is just the passive convection of the vortex with the mean velocity.

The reconstruction procedure is applied to each component of the solution ξ. We first compute the solution to $t = 2.0$ for the accuracy test. The meshes are the same as those in Example 3.1 suitably scaled for the new spatial domain. The accuracy results are shown in Table 3.5 for the uniform meshes and Table 3.6 for the non-uniform meshes. The errors presented are those of the cell averages of ρ.

We then fix the mesh at $h = \frac{1}{8}$ (uniform) and compute the long time evolution of the vortex. Fig. 3.4 is the result by the third order scheme at $t = 0$ and after 1, 5 and 10 time periods, and Fig. 3.5 is the result by the fourth order scheme. We show the line cut through the center of the vortex for the density ρ. It is easy to see the difference between the third and fourth order schemes. The fourth order scheme gives almost no dissipation even after 10 periods, while the dissipation is quite noticeable for the long time results of the third order scheme.

TABLE 3.5
Accuracy for 2D Euler equation of smooth vortex evolution, uniform meshes, linear schemes.

	P^1 (3rd order)				P^2 (4th order)			
h	L^1 error	order	L^∞ error	order	L^1 error	order	L^∞ error	order
1	1.65E-02	—	2.60E-01	—	5.26E-03	—	7.89E-02	—
1/2	6.31E-03	1.39	1.21E-01	1.10	7.36E-04	2.84	1.62E-02	2.28
1/4	1.31E-03	2.27	2.53E-02	2.26	5.40E-05	3.77	1.03E-03	3.98
1/8	2.21E-04	2.57	4.66E-03	2.44	2.32E-06	4.54	5.36E-05	4.26
1/16	2.98E-05	2.89	6.44E-04	2.86	1.10E-07	4.40	2.48E-06	4.43
1/32	3.77E-06	2.98	8.23E-05	2.97	6.37E-09	4.11	1.25E-07	4.31

TABLE 3.6
Accuracy for 2D Euler equation of smooth vortex evolution, non-uniform meshes, linear schemes.

	P^1 (3rd order)				P^2 (4th order)			
h	L^1 error	order	L^∞ error	order	L^1 error	order	L^∞ error	order
$h_0/2$	1.81E-02	—	2.98E-01	—	7.00E-03	—	8.16E-02	—
$h_0/4$	7.74E-03	1.28	1.44E-01	1.05	1.18E-03	2.57	1.61E-02	2.34
$h_0/8$	1.67E-03	2.21	2.47E-02	2.54	8.17E-05	3.85	1.31E-03	3.62
$h_0/16$	2.86E-04	2.55	4.79E-03	2.37	4.70E-06	4.12	1.10E-04	3.57
$h_0/32$	3.94E-05	2.86	7.95E-04	2.59	2.68E-07	4.13	7.73E-06	3.83
$h_0/64$	5.07E-06	2.96	1.25E-04	2.67	1.56E-08	4.10	5.99E-07	3.69

4. WENO Reconstruction and WENO Schemes. In this section, we will introduce non-linear weights to make the resulting schemes suitable for shock computations. To ensure stability near shocks, we need non-negative weights, thus we need non-negative linear weights to start with. Unfortunately, for most triangulations, there are negative coefficients in the combinations for both the third and fourth order schemes constructed above. We fix this problem by grouping the linear polynomials for the third order scheme, and by recombining the six combinations of quadratic polynomials for the fourth order scheme.

4.1. Grouping linear polynomials for the third order scheme. We will use the same notations as in the previous section. In (3.2), by consistency, $\sum_{s=1}^{9} \gamma_s = 1$. We want to group these 9 linear polynomials into 3 new linear polynomials such that each new linear polynomial is still a second order approximation to u and the new combination coefficients (of the three new linear polynomials) are positive. We also want the stencils corresponding to the three new linear polynomials to be reasonably separated, so that when shocks are present, not all stencils will contain the shock under normal situations.

The grouping we will introduce in the following works for most triangulations. There are however cases when it does give some negative coefficients with very small magnitudes, but it does not seem to affect the stability of WENO schemes built upon them.

For the first quadrature point on side i (G_1 in Fig. 3.1), Group 1 contains $p_2(0,k,i)$, $p_4(0,i,ia)$, and $p_5(0,i,ib)$, the combination coefficient is $\gamma_2+\gamma_4+\gamma_5$; Group 2 contains $p_3(0,i,j)$, $p_6(0,j,ja)$, and $p_7(0,j,jb)$; the combination coefficient is $\gamma_3+\gamma_6+\gamma_7$; Group 3 contains $p_1(0,j,k)$, $p_8(0,k,ka)$, and $p_9(0,k,kb)$; the combination coefficient is $\gamma_1+\gamma_8+\gamma_9$.

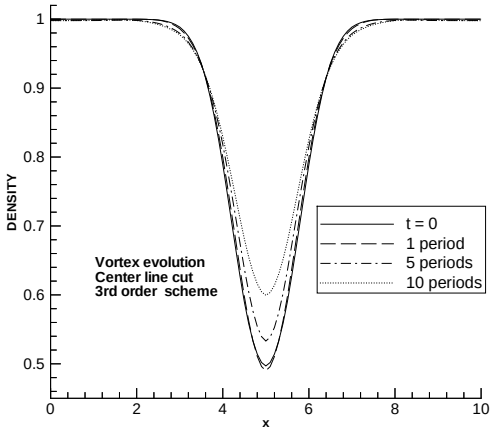

FIG. 3.4. *2D vortex evolution: third order linear scheme.*

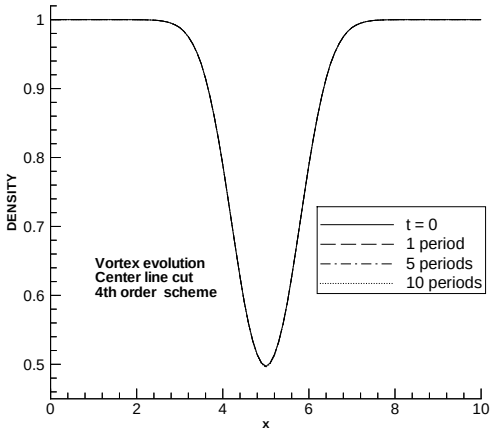

FIG. 3.5. *2D vortex evolution: fourth order linear scheme.*

For the second quadrature point on side i, (G_2 in Fig. 3.1), Group 1 contains $p_3(0,i,j)$, $p_4(0,i,ia)$, and $p_5(0,i,ib)$, the combination coefficient is $\gamma_3+\gamma_4+\gamma_5$; Group 2 contains $p_2(0,k,i)$, $p_8(0,k,ka)$, and $p_9(0,k,kb)$; the combination coefficient is $\gamma_2+\gamma_8+\gamma_9$; Group 3 contains $p_1(0,j,k)$, $p_6(0,j,ja)$, and $p_7(0,j,jb)$; the combination coefficient is $\gamma_1+\gamma_6+\gamma_7$.

We can do the same thing for the other two sides (j,k). The resulting polynomial $\tilde{R}(x,y)$

$$\tilde{R}(x,y)=\sum_{s=1}^{3}\tilde{\gamma}_s\tilde{p}_s(x,y) \tag{4.1}$$

is equivalent to $R(x,y)$ and in most cases the coefficients $\tilde{\gamma}_s$ are non-negative.

4.2. New combination of quadratic polynomials for the fourth order scheme. From section 3.2, we can construct 6 combinations of quadratic polynomials which are fourth order approximations to u. For each quadrature point, we have 6 groups of combination coefficients for q_1,q_2,q_3,q_4,q_5,q_6. To combine

these 6 groups to a new one with non-negative coefficients for the q_s's, assuming that A is the 6×6 matrix formed by the above 6 groups of combination coefficients, we want to find $x \in \Re^6$, $x = (x_1, x_2, x_3, x_4, x_5, x_6)^T$ such that

$$\begin{aligned} x_1 + x_2 + x_3 + x_4 + x_5 + x_6 &= 1 \\ Ax &\geq 0 \end{aligned}$$

which can be solved by using a linear programming procedure

$$\begin{cases} \max_{-1 \leq x_i \leq 1} (x_1 + x_2 + x_3 + x_4 + x_5 + x_6) \\ \text{subject to } Ax \geq 0 \end{cases}$$

provided that the object (maximum) is positive.

Unfortunately, there are meshes such that the object = 0 for some triangles. Due to this, our fourth order schemes have rather severe restrictions on the type of triangulations to which it can apply. Further investigation is needed to improve this. Currently, we only consider meshes which give a positive maximum (e.g. nearly uniform triangles).

4.3. Smoothness indicators and non-linear weights. We finally come to the point of smooth indicators and nonlinear weights. For this we follow exactly as in Jiang and Shu [19]. For a polynomial $p(x, y)$ with degree up to k, we define the following measurement for smoothness

$$S = \sum_{1 \leq |\alpha| \leq k} \int_{\triangle} |\triangle|^{|\alpha|-1} (D^\alpha p(x, y))^2 dxdy \tag{4.2}$$

The non-linear weights are then defined as:

$$\omega_j = \frac{\tilde{\omega_j}}{\sum_i \tilde{\omega_i}}, \qquad \tilde{\omega_i} = \frac{C_i}{(\epsilon + S_i)^2} \tag{4.3}$$

where C_i is the i-th coefficient in the linear combination of polynomials (after grouping or re-combination), S_i is the measurement of smoothness of the i-th polynomial $p_i(x, y)$, and ϵ is a small positive number which we take as $\epsilon = 10^{-3}$ for all the numerical experiments in this paper. The numerical results are not very sensitive to the choice of ϵ in a range from 10^{-2} to 10^{-6}. In general, larger ϵ gives better accuracy for smooth problems but may generate small oscillations for shocks. Smaller ϵ is more friendly to shocks.

4.4. Extension to the Euler systems. There are two ways to extend the previous results to systems. One is to do so component by component. This is easy to implement and cost effective, and it seems to work well for the third order scheme. We will use component-wise methods for all numerical examples with the third order scheme. Another extension method is by the characteristic decomposition. We will give a brief description in the following.

Let us take one side of the triangle which has the outward unit normal (n_x, n_y). Let A be some average Jacobian at one quadrature point,

$$A = n_x \frac{\partial f}{\partial u} + n_y \frac{\partial g}{\partial u}. \tag{4.4}$$

For Euler systems, the Roe's mean matrix [25] is used. Denote by R the matrix of right eigenvectors and L the matrix of left eigenvectors of A. Then the scalar triangular WENO scheme can be applied to each of the characteristic fields, i.e. to each component of the vector $v = L\,u$. With the reconstructed point values v, we define our reconstructed point values u by $u = R\,v$.

TABLE 5.1
Accuracy for 2D linear equation, uniform meshes, WENO schemes.

	P^1 (3rd order)				P^2 (4th order)			
h	L^1 error	order	L^∞ error	order	L^1 error	order	L^∞ error	order
2/5	2.66E-01	—	4.30E-01	—	1.38E-02	—	2.94E-02	—
1/5	8.11E-02	1.71	1.93E-01	1.16	1.80E-03	2.94	2.74E-03	3.42
1/10	2.65E-02	1.62	6.16E-02	1.65	8.87E-05	4.34	1.46E-04	4.23
1/20	2.68E-03	3.31	8.77E-03	2.81	4.34E-06	4.35	7.11E-06	4.36
1/40	1.44E-04	4.22	4.88E-04	4.17	2.30E-07	4.24	3.71E-07	4.26
1/80	8.05E-06	4.16	2.40E-05	4.35	1.34E-08	4.10	2.12E-08	4.13

TABLE 5.2
Accuracy for 2D linear equation, non-uniform meshes, WENO schemes.

	P^1 (3rd order)				P^2 (4th order)			
h	L^1 error	order	L^∞ error	order	L^1 error	order	L^∞ error	order
$h_0/2$	2.79E-01	—	5.28E-01	—	1.77E-02	—	6.41E-02	—
$h_0/4$	8.43E-02	1.73	2.32E-01	1.19	8.85E-04	4.32	3.07E-03	4.38
$h_0/8$	2.53E-02	1.74	7.47E-02	1.64	4.08E-05	4.44	1.43E-04	4.42
$h_0/16$	2.24E-03	3.50	1.14E-02	2.71	1.82E-06	4.49	6.37E-06	4.49
$h_0/32$	1.18E-04	4.25	6.83E-04	4.06	8.95E-08	4.35	3.36E-07	4.25
$h_0/64$	6.21E-06	4.25	3.15E-05	4.44	4.92E-09	4.19	2.00E-08	4.07

4.5. Parallel implementation. As an explicit method, the WENO schemes constructed above are easily implemented on an IBM SP-2 parallel computer. The parallel efficiency is over 90% when 16 processors are used. Most of the numerical examples with large meshes in the next section are obtained with the SP-2 parallel computer using 16 processors at the Center for Fluid Mechanics of Brown University.

5. Numerical Examples. We will implement the third and fourth order WENO schemes developed in the previous sections to some two-dimension test problems. First, the same accuracy tests as in the linear weights case are given for linear, Burgers' equation and the smooth vortex problem. Next, some non-smooth problems for two-dimensional Euler equations are tested.

5.1. Accuracy test for triangular WENO schemes. We use the same examples as in section 3.3 to test the accuracy of third and fourth order triangular WENO schemes constructed in the previous section.

Example 5.1. Two-dimensional linear equation as defined in Example 3.1, (3.4). The accuracy results are shown in Table 5.1 for the uniform meshes and in Table 5.2 for the non-uniform meshes. We can see that the correct orders of accuracy are obtained by the third and fourth order WENO methods.

Example 5.2. Two-dimensional Burgers' equation as defined in Example 3.2, (3.5). The accuracy results are shown in Table 5.3 for the uniform meshes and in Table 5.4 for the non-uniform meshes. We can see that the correct orders of accuracy are again obtained by the third and fourth order WENO methods.

TABLE 5.3
Accuracy for 2D Burgers' equation, uniform meshes, WENO schemes.

	P^1 (3rd order)				P^2 (4th order)			
h	L^1 error	order	L^∞ error	order	L^1 error	order	L^∞ error	order
2/5	2.76E-02	—	8.18E-02	—	8.64E-03	—	2.106-02	—
1/5	4.63E-03	2.58	1.20E-02	2.77	6.05E-04	3.84	1.73E-03	3.60
1/10	6.97E-04	2.73	2.16E-03	2.47	3.94E-05	3.94	1.18E-04	3.87
1/20	7.12E-05	3.29	1.90E-04	3.51	2.50E-06	3.98	7.42E-06	3.99
1/40	7.63E-06	3.22	2.36E-05	3.01	1.57E-07	3.99	4.63E-07	4.00
1/80	9.08E-07	3.07	2.96E-06	3.00	9.83E-09	4.00	2.89E-08	4.00

TABLE 5.4
Accuracy for 2D Burgers' equation, non-uniform meshes, WENO schemes.

	P^1 (3rd order)				P^2 (4th order)			
h	L^1 error	order	L^∞ error	order	L^1 error	order	L^∞ error	order
$h_0/2$	2.01E-02	—	9.16E-02	—	4.18E-03	—	2.376-02	—
$h_0/4$	3.85E-03	2.38	1.80E-02	2.35	2.90E-04	3.85	2.61E-03	3.18
$h_0/8$	5.79E-04	2.73	3.39E-03	2.41	1.85E-05	3.97	1.92E-04	3.77
$h_0/16$	5.34E-05	3.44	3.55E-04	3.26	1.18E-06	3.97	1.35E-05	3.83
$h_0/32$	5.12E-06	3.38	2.95E-05	3.59	7.45E-08	3.99	9.99E-07	3.76
$h_0/64$	5.82E-07	3.14	4.23E-06	2.80	4.67E-09	4.00	7.56E-08	3.72

To demonstrate the application for shock computations, we continue the the above calculation to $t = 5/\pi^2$ when discontinuities develop. Fig. 5.1 is the result for $h = 1/20$ of a uniform mesh. Fig. 5.2 is the result for $h = h_0/16$ of a non-uniform mesh. We can see that the shock transitions are sharp and non-oscillatory.

Example 5.3. Two-dimensional vortex evolution problem as defined in Example 3.3. The accuracy results are shown in Table 5.5 for the uniform meshes and in Table 5.6 for the non-uniform meshes. We can see that the correct orders of accuracy are again obtained by the third and fourth order WENO methods.

Fig. 5.3 and Fig. 5.4 are the results for the long time evolution of the vortex. These results are similar to those obtained with the linear schemes in Fig. 3.4 and Fig. 3.5.

5.2. Riemann problems of Euler equations. The two dimensional triangular WENO methods are applied to one dimensional shock tube problems. We consider the solution of the Euler equations (3.6) in a domain of $[-1, 1] \times [0, 0.2]$ with a triangulation of 101 vertices in the x-direction and 11 vertices in the y-direction. The velocity in the y-direction is zero, and periodic boundary condition is used in the y-direction. The mesh is shown in Fig. 5.5. The pictures shown below are obtained by extracting the data along the central cut line for 101 equally spaced points.

We consider the following Riemann type initial conditions:

$$(\rho, u, p) = \begin{cases} (\rho_L, u_L, p_L) & \text{if } x \le 0 \\ (\rho_R, u_R, p_R) & \text{if } x > 0. \end{cases}$$

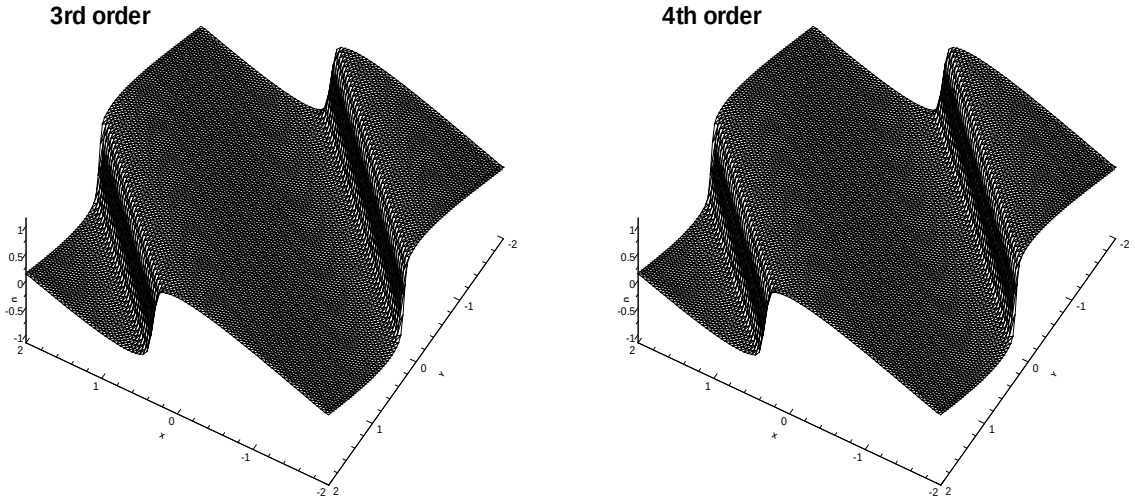

FIG. 5.1. *2D Burgers' equation: $t = 5/\pi^2$, uniform mesh.*

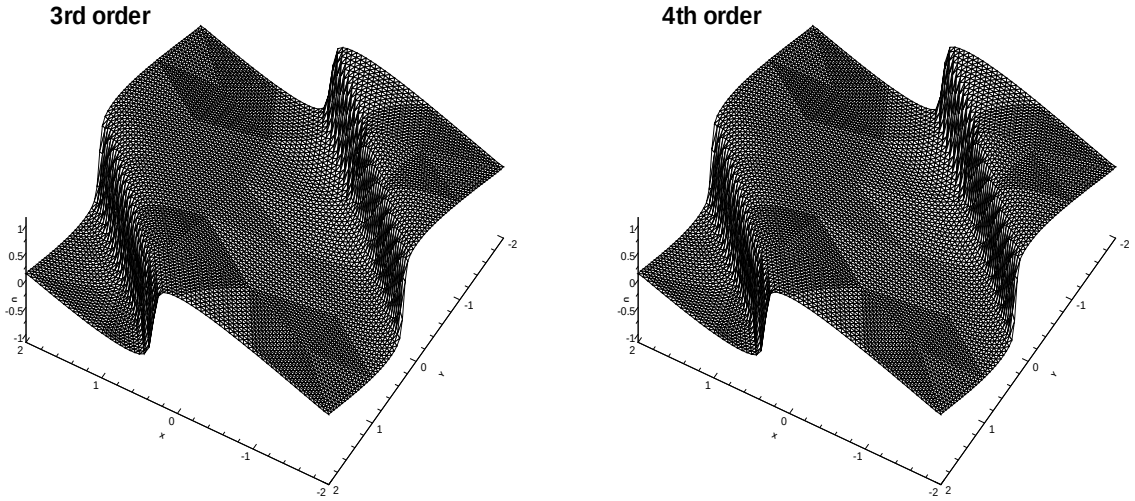

FIG. 5.2. *2D Burgers' equation: $t = 5/\pi^2$, non-uniform mesh.*

The first test case is Sod's problem [33]. The initial data is

$$(\rho_L, u_L, p_L) = (1, 0, 1),\ (\rho_R, u_R, p_R) = (0.125, 0, 0.1).$$

Density at $t = 0.40$ is shown in the first 3 plots in Fig. 5.6.

The second test case is the Riemann problem proposed by Lax [21]:

$$(\rho_L, u_L, p_L) = (0.445, 0.698, 3.528),\ (\rho_R, u_R, p_R) = (0.5, 0, 0.571).$$

Density at $t = 0.26$ is shown in the last 3 plots in Fig. 5.6.

We can observe a better resolution of the fourth order scheme over the third order one, and also a less oscillatory result from the characteristic version of the fourth order scheme over the component version.

TABLE 5.5
Accuracy for 2D Euler equation of smooth vortex evolution, uniform meshes, WENO schemes.

	P^1 (3rd order)				P^2 (4th order)			
h	L^1 error	order	L^∞ error	order	L^1 error	order	L^∞ error	order
1	1.87E-02	—	2.95E-01	—	1.30E-02	—	2.05E-01	—
1/2	1.01E-02	0.89	2.09E-01	0.50	2.50E-03	2.38	4.45E-02	2.49
1/4	2.78E-03	1.86	6.37E-02	1.71	1.79E-04	3.80	3.29E-03	3.76
1/8	6.47E-04	2.10	3.05E-02	1.06	6.92E-06	4.69	1.96E-04	4.07
1/16	8.74E-05	2.89	8.14E-03	1.91	2.03E-07	5.09	4.95E-06	5.31
1/32	7.10E-06	3.62	5.66E-04	3.85	7.83E-09	4.70	1.96E-07	4.66

TABLE 5.6
Accuracy for 2D Euler equation of smooth vortex evolution, non-uniform meshes, WENO schemes.

	P^1 (3rd order)				P^2 (4th order)			
h	L^1 error	order	L^∞ error	order	L^1 error	order	L^∞ error	order
$h_0/2$	2.12E-02	—	3.33E-01	—	1.84E-02	—	2.14E-01	—
$h_0/4$	1.28E-02	0.73	2.27E-01	0.55	2.80E-03	2.69	3.43E-02	2.64
$h_0/8$	3.84E-03	1.74	6.85E-02	1.73	2.12E-04	3.72	6.57E-03	2.38
$h_0/16$	8.32E-04	2.21	3.02E-02	1.18	1.09E-05	4.28	5.91E-04	3.48
$h_0/32$	1.26E-04	2.72	5.64E-03	2.42	3.76E-07	4.86	1.97E-05	4.91
$h_0/64$	1.16E-05	3.44	6.19E-04	3.19	1.66E-08	4.50	6.78E-07	4.86

5.3. A Mach 3 wind tunnel with a step. This problem is from [36]. We solve the Euler equations (3.6) in a wind tunnel of 1 length unit wide and 3 length units long. The step is 0.2 length units high and is located 0.6 length units from the left end of the tunnel. Initially, a right-going Mach 3 flow is used. Reflective boundary conditions are applied along the walls of the tunnel and inflow and outflow boundary conditions are used at the entrance and the exit.

The corner of the step is a singularity point. [36] uses an assumption of nearly steady flow in the region near the corner to fix this singularity. In this paper, we do not modify our method near the corner, instead we adopt the same technique as the one used in [7], namely refining the mesh near the corner and using the same scheme in the whole domain.

We use the third order scheme for this problem. Four meshes have been used, see Fig. 5.7. For the first mesh, the triangle size away from the corner is roughly equal to a rectangular element case of $\Delta x = \Delta y = \frac{1}{40}$, while it is one-quarter of that near the corner. For the second mesh, the triangle size away from the corner is the same as in the first mesh, but it is one-eighth of that near the corner. The third mesh has a triangle size of $\Delta x = \Delta y = \frac{1}{80}$ away from the corner, and it is one-quarter of that near the corner. The last mesh has a triangle size of $\Delta x = \Delta y = \frac{1}{160}$ away from the corner, and it is one-half of that near the corner. Fig. 5.8 is the contour picture for the density at time $t = 4.0$. It is clear that with more triangles near the corner the artifacts (spurious boundary layers downstream) from the singularity decrease significantly.

5.4. Double Mach reflection. This problem is also from [36]. We solve the Euler equations (3.6) in a computational domain of $[0, 4] \times [0, 1]$. A reflecting wall lies at the bottom of the domain starting from

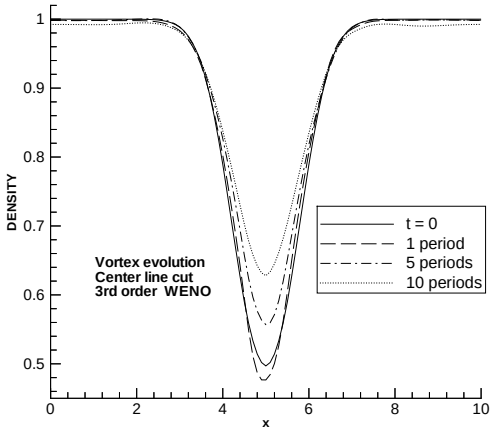

FIG. 5.3. *2D vortex evolution: third order WENO scheme.*

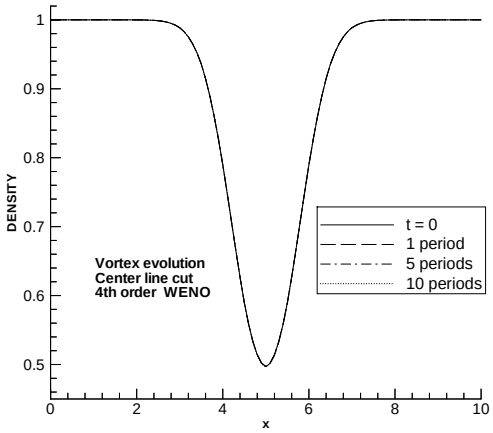

FIG. 5.4. *2D vortex evolution: fourth order WENO scheme.*

$x = \frac{1}{6}$. Initially a right-moving Mach 10 shock is located at $x = \frac{1}{6}, y = 0$, making a 60° angle with the x-axis. The reflective boundary condition is used at the wall, while for the rest of the bottom boundary (the part from $x = 0$ to $x = \frac{1}{6}$, the exact post-shock condition is imposed. At the top boundary, the flow values are set to describe the exact motion of the Mach 10 shock. The results shown are at $t = 0.2$.

We test both the third and the fourth order schemes. Four triangle sizes are used, they are roughly equal to rectangular element cases of $\Delta x = \Delta y = \frac{1}{50}$, $\Delta x = \Delta y = \frac{1}{100}$, $\Delta x = \Delta y = \frac{1}{200}$, and $\Delta x = \Delta y = \frac{1}{400}$ respectively. For the third order scheme, we use both uniform triangular mesh (equilateral triangles) and locally refined triangular mesh (the refined region has the above triangle sizes, Fig. 5.9 shows the region $[0, 2] \times [0, 1]$ of such a mesh of $\Delta x = \Delta y = \frac{1}{50}$ locally). For the fourth order, we use uniform triangular mesh only. For the cases of $\Delta x = \Delta y = \frac{1}{200}$ and $\Delta x = \Delta y = \frac{1}{400}$, we present both the picture of whole region ($[0, 3] \times [0, 1]$) and a blow-up region around the double Mach stems. All pictures are the density contours with 30 equally spaced contour lines from 1.5 to 21.5. We can clearly see that the fourth order scheme captures

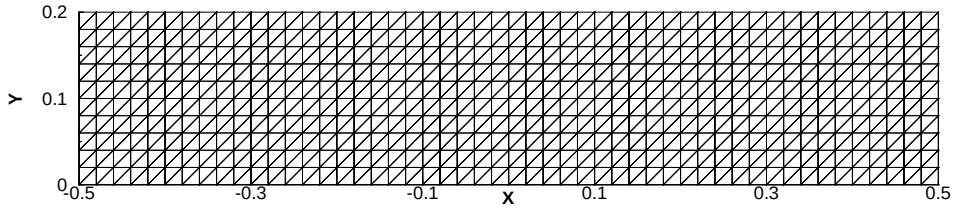

FIG. 5.5. *Mesh for the Riemann problems.*

the complicated flow structure under the triple Mach stem much better than the third order scheme. We refer to [7] for similar results obtained with discontinuous Galerkin methods.

6. Concluding Remarks. We have presented the development of third and fourth order WENO schemes based on linear and quadratic polynomials for 2D triangle meshes. Accuracy and stability issues are considered in the design of the schemes and verified numerically. Numerical examples show the improvement of resolution using high order schemes.

REFERENCES

[1] R. ABGRALL, *On essentially non-oscillatory schemes on unstructured meshes: analysis and implementation*, Journal of Computational Physics, v114 (1994), pp.45–58.

[2] N. ADAMS AND K. SHARIFF, *A high-resolution hybrid compact-ENO scheme for shock-turbulence interaction problems*, Journal of Computational Physics, v127 (1996), pp.27–51.

[3] T. BARTH AND P. FREDERICKSON, *High order solution of the Euler equations on unstructured grids using quadratic reconstruction*, AIAA Paper No. 90-0013.

[4] F. BIANCO, G. PUPPO AND G. RUSSO, *High order central schemes for hyperbolic systems of conservation laws*, SIAM Journal on Scientific Computing, to appear.

[5] J. CASPER, *Finite-volume implementation of high-order essentially non-oscillatory schemes in two dimensions*, AIAA Journal, v30 (1992), pp.2829–2835.

[6] J. CASPER AND H. ATKINS, *A finite-volume high-order ENO scheme for two dimensional hyperbolic systems*, Journal of Computational Physics, v106 (1993), pp.62–76.

[7] B. COCKBURN AND C.-W. SHU, *The Runge-Kutta discontinuous Galerkin method for conservation laws V: multidimensional systems*, Journal of Computational Physics, v141 (1998), pp.199-224.

[8] A. DOLEZAL AND S. WONG, *Relativistic hydrodynamics and essentially non-oscillatory shock capturing schemes*, Journal of Computational Physics, v120 (1995), pp.266–277.

[9] W. E AND C.-W. SHU, *A numerical resolution study of high order essentially non-oscillatory schemes applied to incompressible flow*, Journal of Computational Physics, v110 (1994), pp.39–46.

[10] G. ERLEBACHER, Y. HUSSAINI AND C.-W. SHU, *Interaction of a shock with a longitudinal vortex*, Journal of Fluid Mechanics, v337 (1997), pp.129–153.

[11] E. FATEMI, J. JEROME AND S. OSHER, *Solution of the hydrodynamic device model using high order non-oscillatory shock capturing algorithms*, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, v10 (1991), pp.232–244.

[12] O. FRIEDRICHS, *Weighted essentially non-oscillatory schemes for the interpolation of mean values on unstructured grids*, Journal of Computational Physics, to appear.

[13] E. Harabetian, S. Osher and C.-W. Shu, *An Eulerian approach for vortex motion using a level set regularization procedure*, Journal of Computational Physics, v127 (1996), pp.15–26.

[14] A. Harten, *Preliminary results on the extension of ENO schemes to two dimensional problems*, in Proceedings of the International Conference on Hyperbolic Problems, Saint-Etienne, 1986.

[15] A. Harten and S. Chakravarthy, *Multi-dimensional ENO schemes for general geometries*, ICASE Report No. 91-76 (1991)

[16] A. Harten, B. Engquist, S. Osher and S. Chakravarthy, *Uniformly high order essentially non-oscillatory schemes, III*, Journal of Computational Physics, v71 (1987), pp.231–303.

[17] J. Jerome and C.-W. Shu, *Energy models for one-carrier transport in semiconductor devices*, in IMA Volumes in Mathematics and Its Applications, v59, W. Coughran, J. Cole, P. Lloyd and J. White, editors, Springer-Verlag, 1994, pp.185–207.

[18] J. Jerome and C.-W. Shu, *Transport effects and characteristic modes in the modeling and simulation of submicron devices*, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, v14 (1995), pp.917–923.

[19] G. Jiang and C.-W. Shu, *Efficient implementation of weighted ENO schemes*, Journal of Computational Physics, v126 (1996), pp.202–228.

[20] F. Ladeinde, E. O'Brien, X. Cai and W. Liu, *Advection by polytropic compressible turbulence*, Physics of Fluids, v7 (1995), pp.2848–2857.

[21] P. D. Lax, *Weak solutions of nonlinear hyperbolic equations and their numerical computation*, Communications in Pure and Applied Mathematics, v7 (1954), pp.159–193.

[22] D. Levy, G. Puppo and G. Russo, *Central WENO schemes for hyperbolic systems of conservation laws*, Mathematical Modelling and Numerical Analysis, submitted.

[23] X.-D. Liu, S. Osher and T. Chan, *Weighted essentially non-oscillatory schemes*, Journal of Computational Physics, v115 (1994), pp.200–212.

[24] S. Osher and J. Sethian, *Fronts propagating with curvature-dependent speed: algorithms based on Hamilton-Jacobi formulation*, Journal of Computational Physics, v79 (1988), pp.12–49.

[25] P. L. Roe, *Approximate Riemann solvers, parameter vectors, and difference schemes*, Journal of Computational Physics, v43 (1981), pp.357–372.

[26] J. Sethian, *Level Set Methods: Evolving Interfaces in Geometry, Fluid Dynamics, Computer Vision, and Material Science*, Cambridge Monographs on Applied and Computational Mathematics, Cambridge University Press, New York, New York, 1996.

[27] C.-W. Shu, *Essentially Non-Oscillatory and Weighted Essentially Non-Oscillatory Schemes for Hyperbolic Conservation Laws*, in *Advanced Numerical Approximation of Nonlinear Hyperbolic Equations*, A. Quarteroni, Editor, Lecture Notes in Mathematics, CIME subseries, Springer-Verlag, to appear. ICASE Report 97-65.

[28] C.-W. Shu and S. Osher, *Efficient implementation of essentially non-oscillatory shock capturing schemes*, Journal of Computational Physics, v77 (1988), pp.439–471.

[29] C.-W. Shu and S. Osher, *Efficient implementation of essentially non-oscillatory shock capturing schemes II*, Journal of Computational Physics, v83 (1989), pp.32–78.

[30] C.-W. Shu, T.A. Zang, G. Erlebacher, D. Whitaker, and S. Osher, *High order ENO schemes applied to two- and three- dimensional compressible flow*, Applied Numerical Mathematics, v9 (1992), pp.45–71.

[31] C.-W. Shu and Y. Zeng, *High order essentially non-oscillatory scheme for viscoelasticity with fading*

memory, Quarterly of Applied Mathematics, v55 (1997), pp.459-484.

[32] K. Siddiqi, B. Kimia and C.-W. Shu, *Geometric shock-capturing ENO schemes for subpixel interpolation, computation and curve evolution*, Computer Vision Graphics and Image Processing: Graphical Models and Image Processing (CVGIP:GMIP), v59 (1997), pp.278-301.

[33] G. Sod, *A survey of several finite difference methods for systems of nonlinear hyperbolic conservation laws*, Journal of Computational Physics, v27 (1978), pp.1–32.

[34] T. Sonar, *On the construction of essentially non-oscillatory finite volume approximations to hyperbolic conservation laws on general triangulations: polynomial recovery, accuracy and stencil selection*, Comput. Methods Appl. Mech. Engrg. v140 (1997) pp.157–181.

[35] F. Walsteijn, *Robust numerical methods for 2D turbulence*, Journal of Computational Physics, v114 (1994), pp.129–145.

[36] P. Woodward and P. Colella, *The numerical simulation of two-dimensional fluid flow with strong shocks*, Journal of Computational Physics, v54 (1984) pp.115–173.

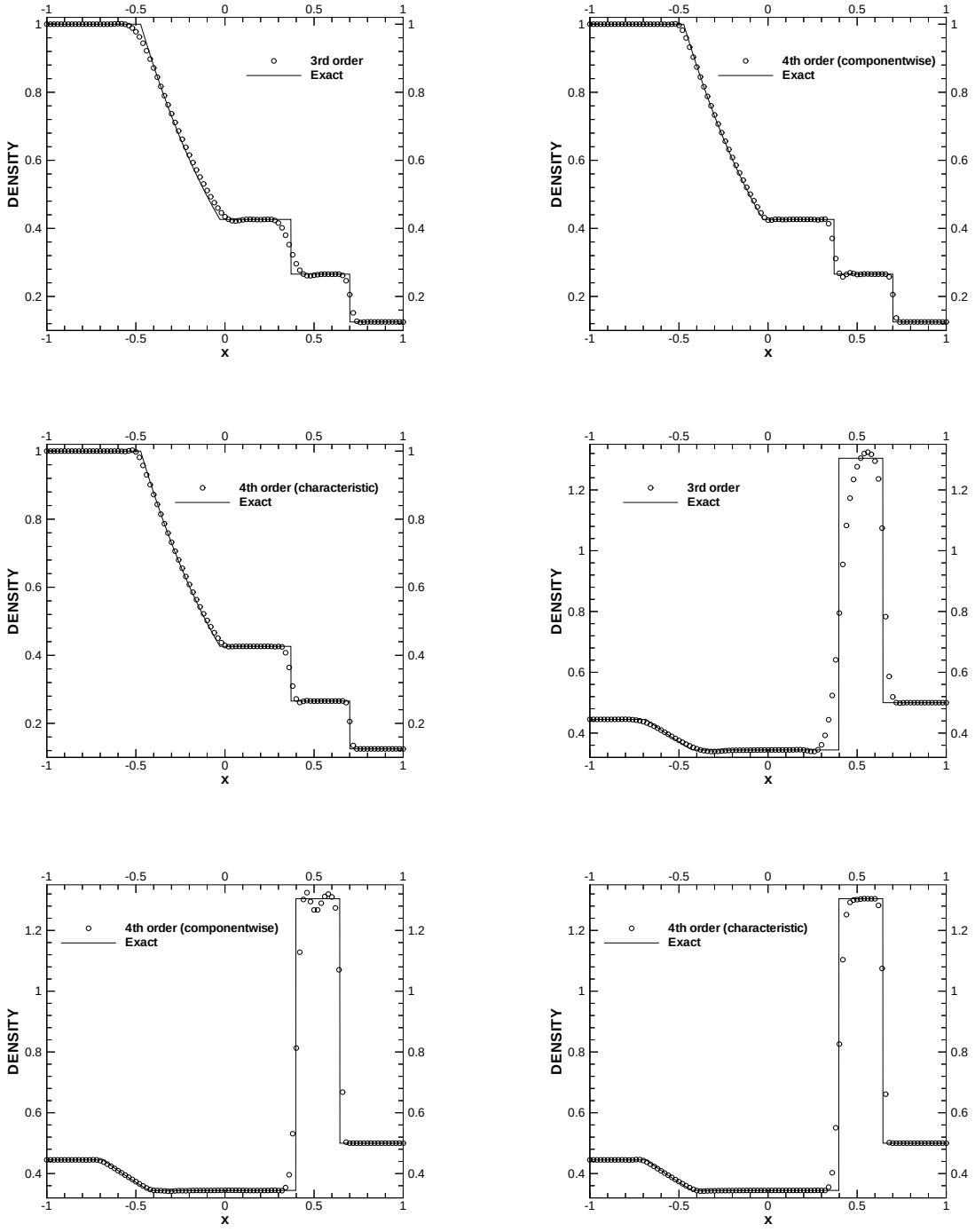

FIG. 5.6. *Riemann problems of Euler equations: density*

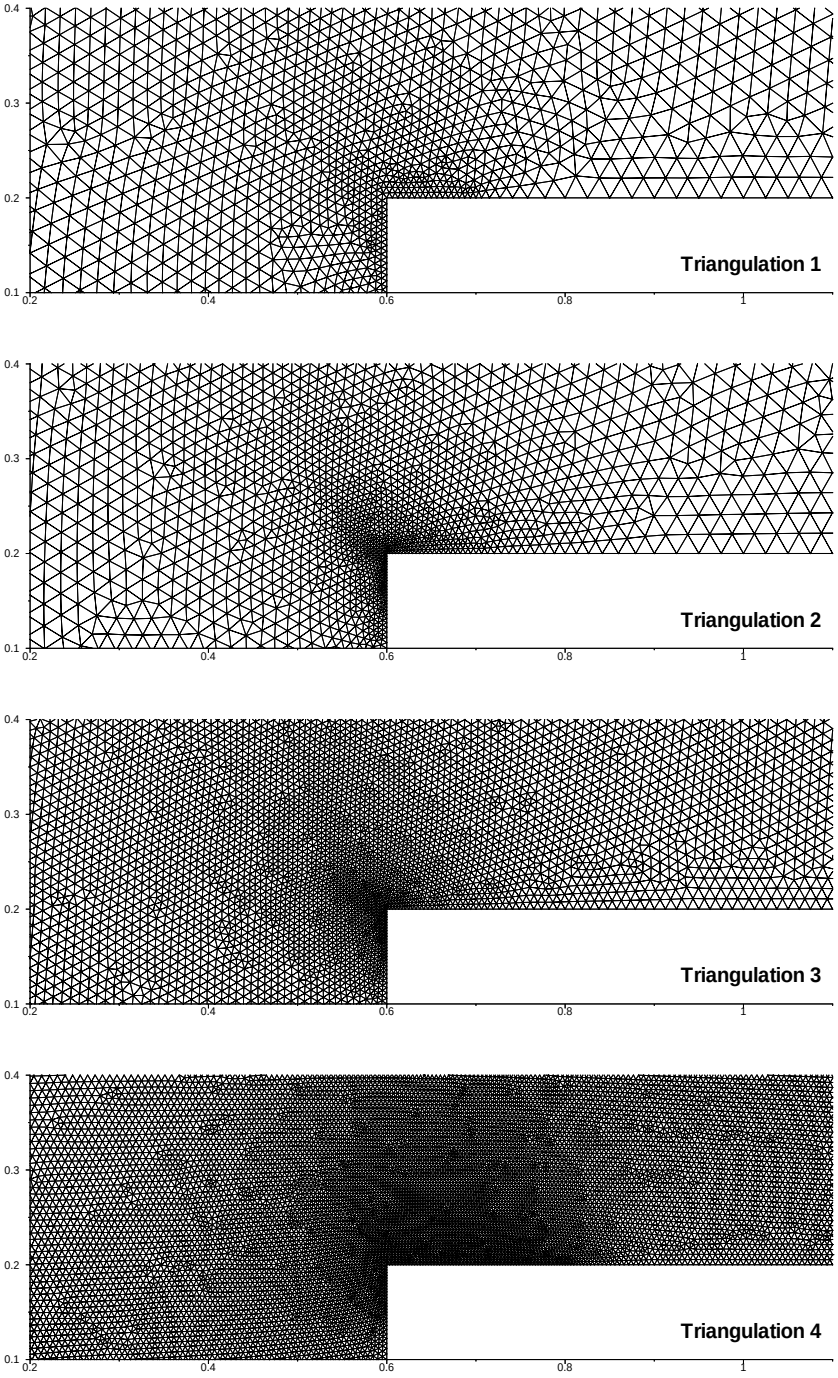

FIG. 5.7. *Triangulations for the forward step problem: part near the corner.*

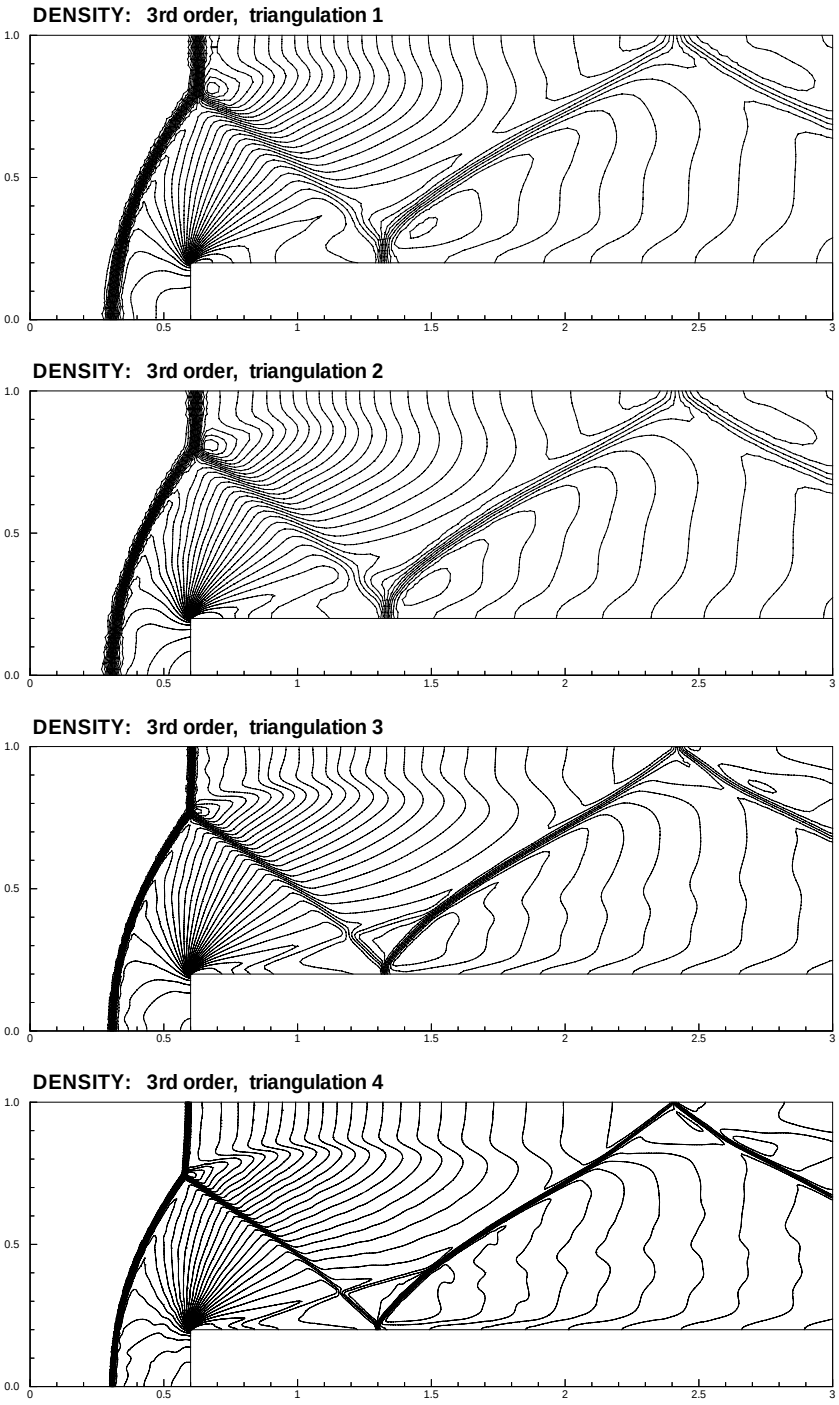

FIG. 5.8. *Forward step problem: 30 contours from 0.32 to 6.15*

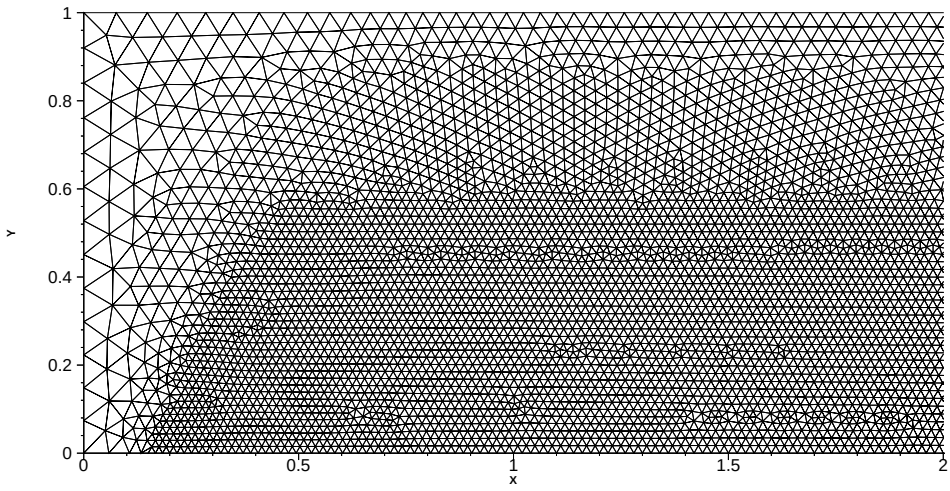

FIG. 5.9. *Triangulation for the double Mach reflection*

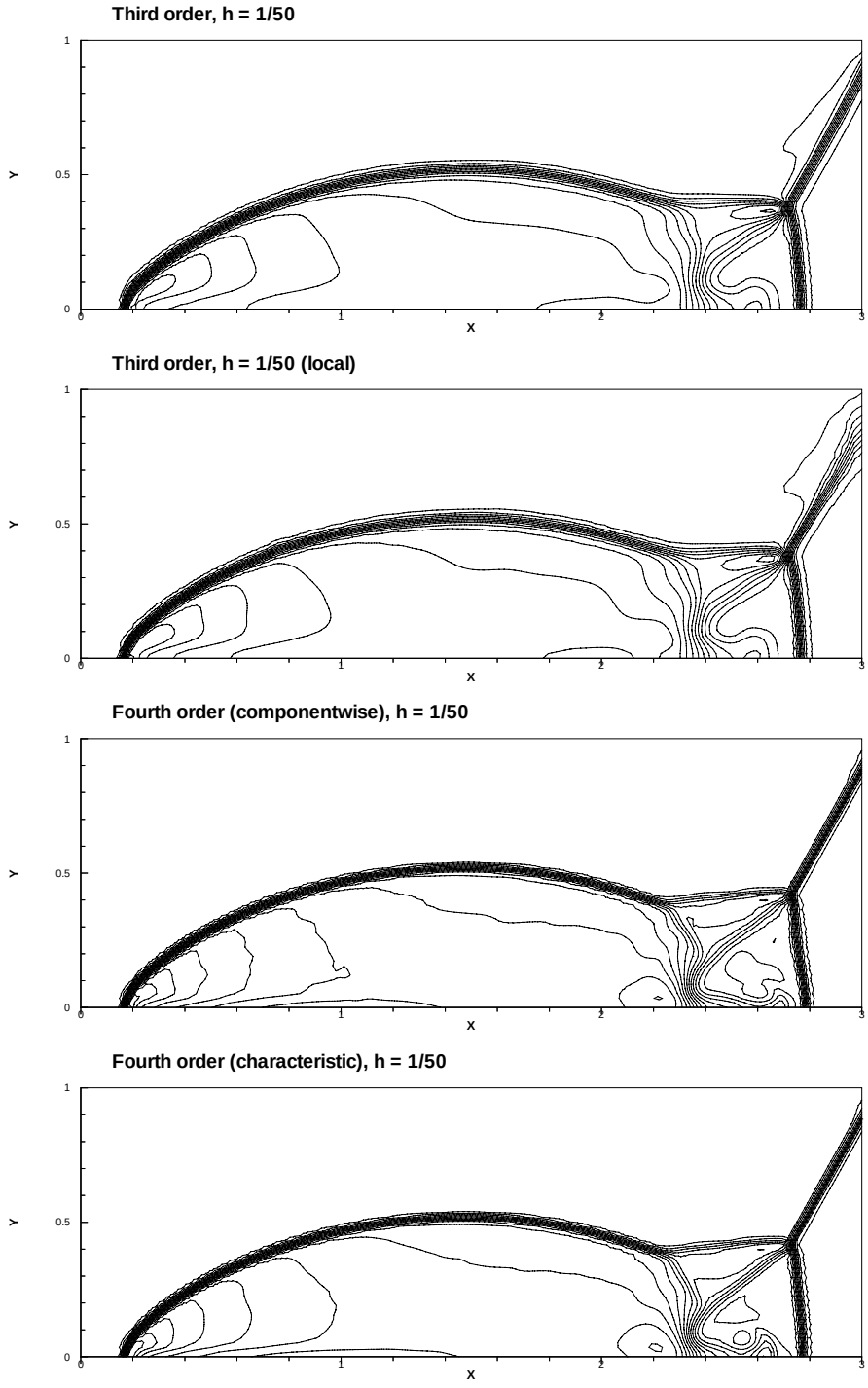

FIG. 5.10. *Double Mach reflection:* $h = \frac{1}{50}$, $t = 0.2$

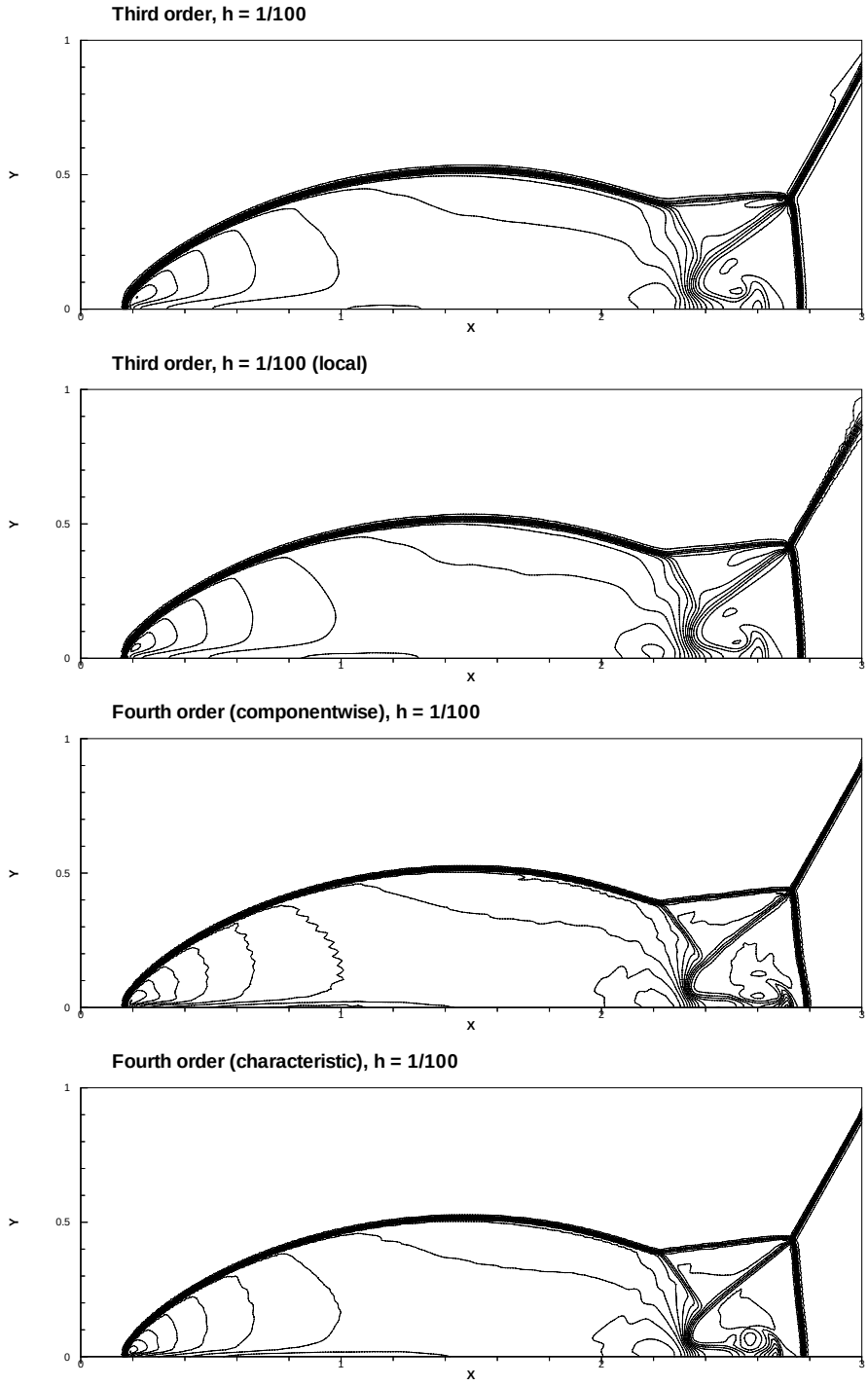

FIG. 5.11. *Double Mach reflection:* $h = \frac{1}{100}$, $t = 0.2$

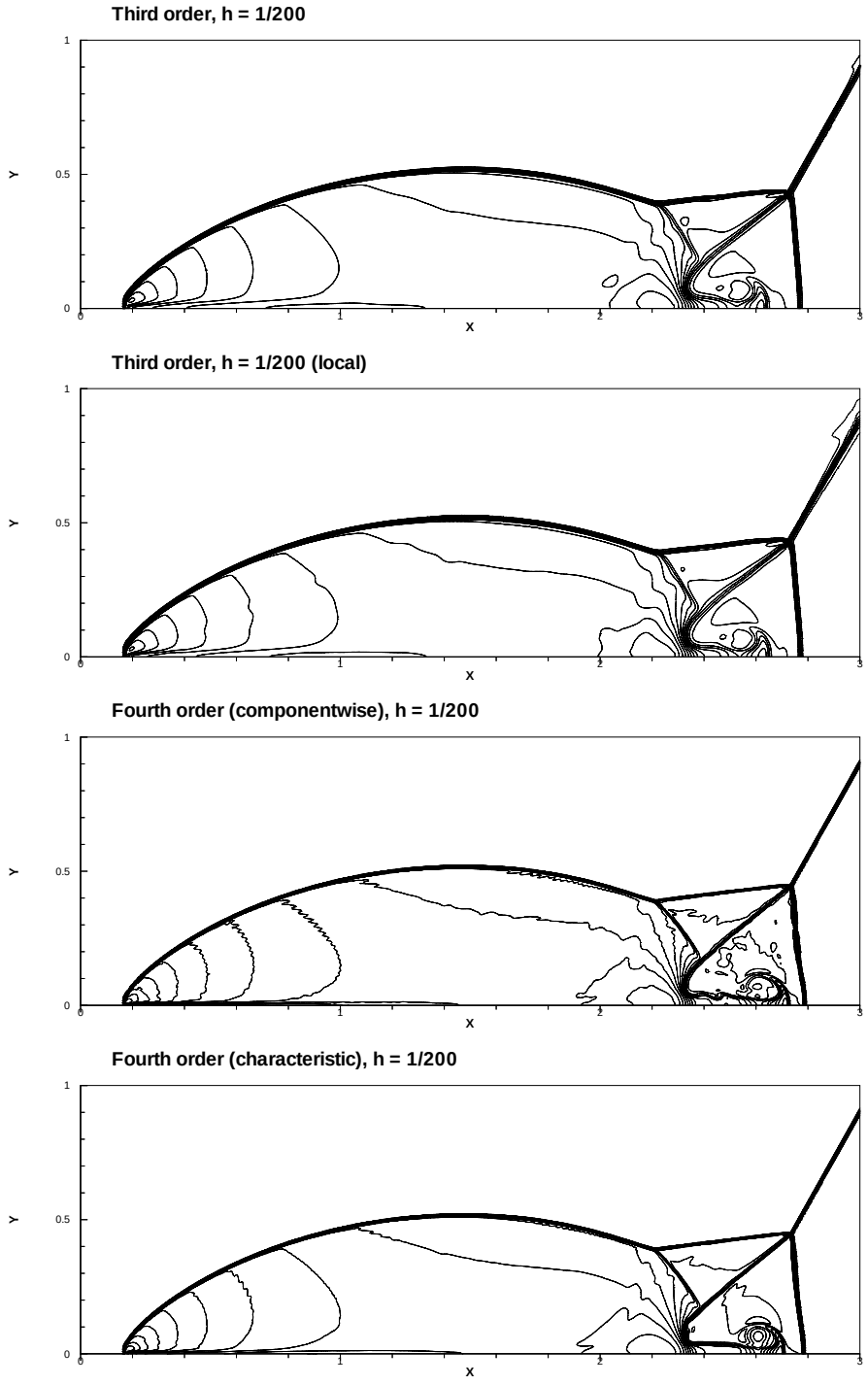

FIG. 5.12. *Double Mach reflection:* $h = \frac{1}{200}$, $t = 0.2$

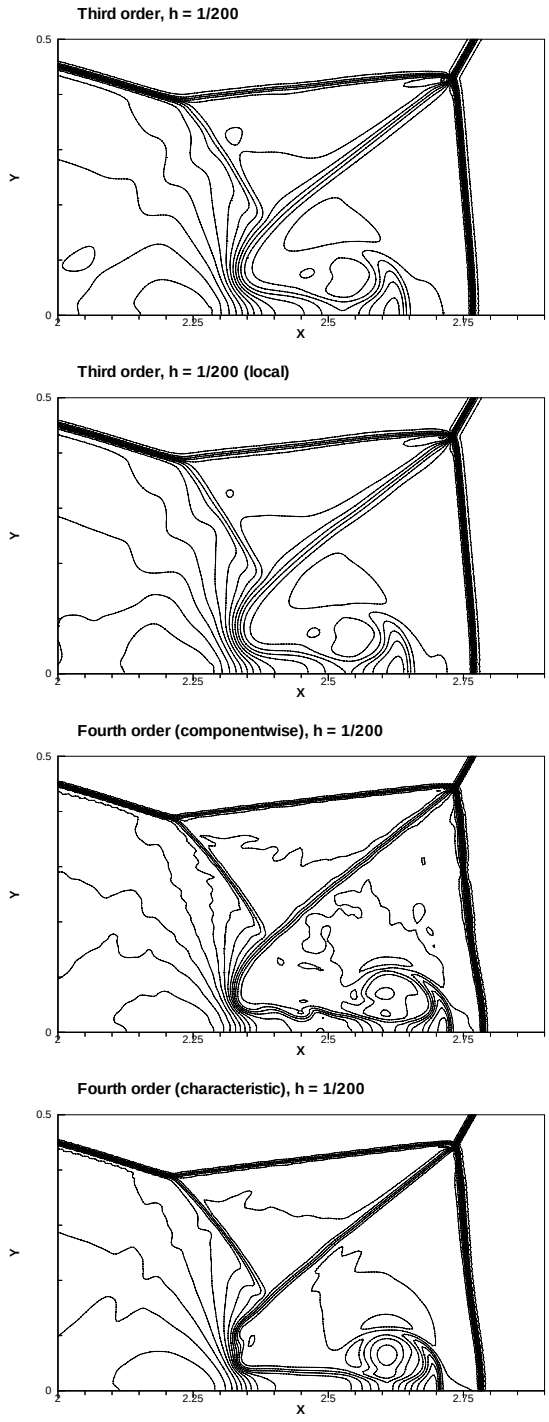

FIG. 5.13. *Double Mach reflection:* $h = \frac{1}{200}$, $t = 0.2$ *(blow-up)*

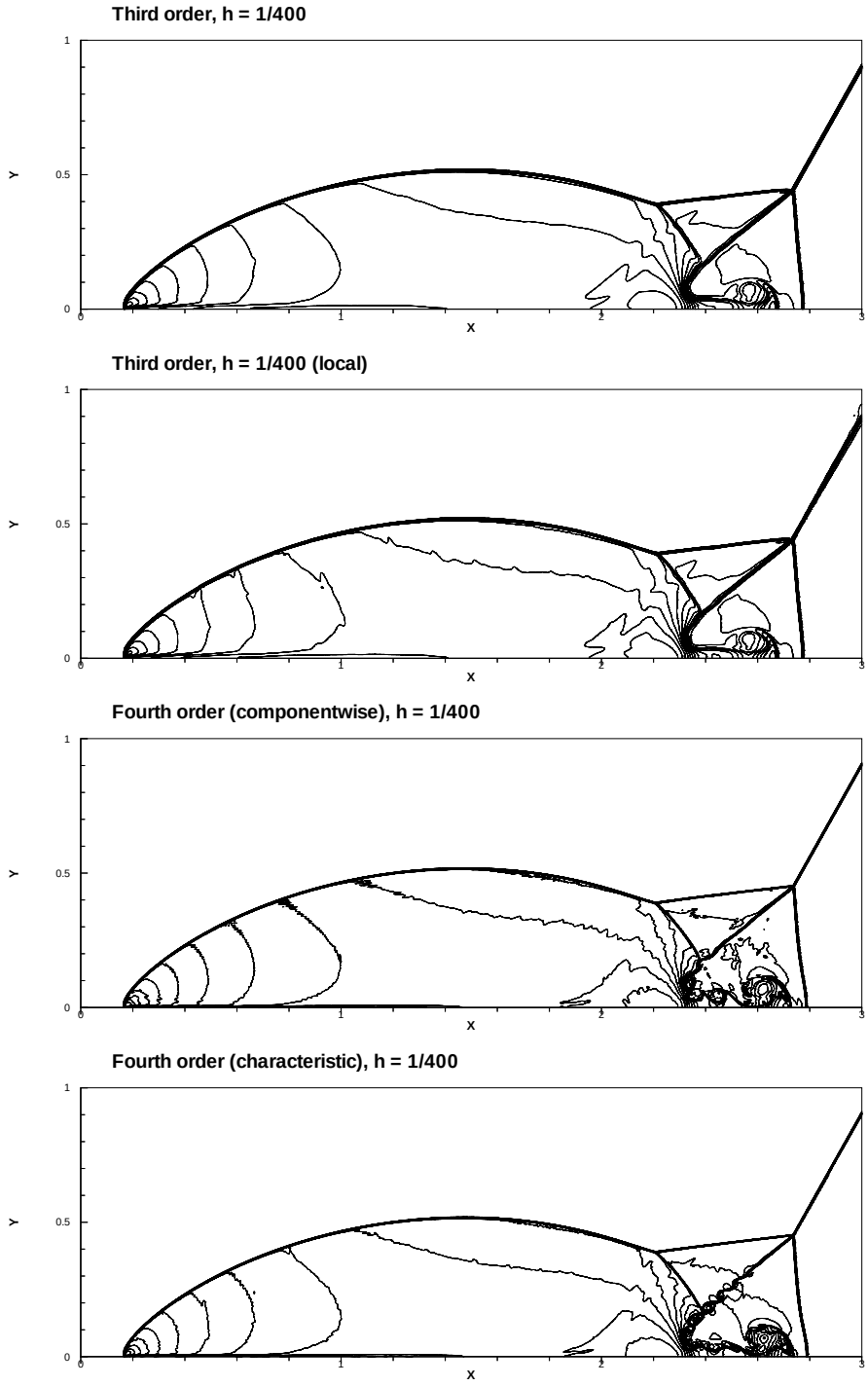

FIG. 5.14. *Double Mach reflection:* $h = \frac{1}{400}$, $t = 0.2$

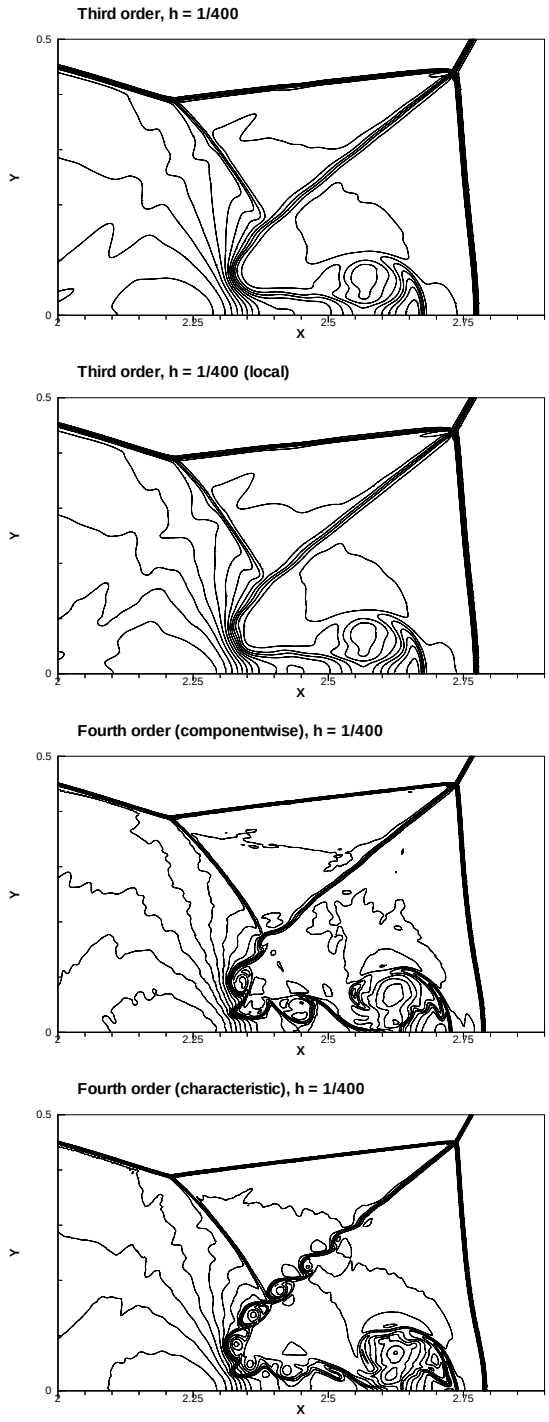

FIG. 5.15. *Double Mach reflection:* $h = \frac{1}{400}$, $t = 0.2$ *(blow-up)*