A TVD-MUSTA scheme for hyperbolic conservation laws

Yousef Hashem Zahran

Abstract

The aim of this paper is to develop an improved version of the Multi-Stage (MUSTA) approach to the construction of upwind fluxes that avoid the solution of the Riemann Problem (RP) in the conventional manner. We propose to use the second order TVD flux as a building block in the MUSTA scheme instead of the first order flux used in the original MUSTA. The numerical solution is advanced by TVD Runge-Kutta method. The new MUSTA scheme improves upon the original MUSTA and TVD schemes in terms of better convergence, higher overall accuracy, better resolution of discontinuities and find its justification when solving very complex systems for which the solution of the RP is costly or unknown. A way to extend this scheme to two dimensional systems of hyperbolic conservation laws is presented. In this paper we also extend the scheme in the framework of high order WENO methods. Numerical results suggest that our scheme is superior to the original schemes. This is specially so for long time evolution problems containing both smooth and non-smooth features.

1 Introduction

Numerical methods for solving nonlinear systems of hyperbolic conservation laws via finite volume methods require, as the building block, a monotone numerical flux. The choice of the building block has a profound influence on the properties of the resulting schemes. There are essentially two approaches for providing a monotone

Received by the editors February 2007.
Communicated by A. Bultheel.
1991 *Mathematics Subject Classification :* 65M06.
Key words and phrases : conservation laws, MUSTA fluxes, upwind schemes, Runge-Kutta methods, TVD schemes, discrete finite volume schemes, Euler equations.

numerical flux. The simplest approach utilizes a symmetric stencil and does not explicitly make use of wave propagation information in the construction of the numerical flux. This approach gives rise to centred schemes. A more refined approach utilizes wave propagation information contained in the differential equations to construct the numerical flux. This is done through the exact or approximate solution of the RP. Due to the fact that wave propagation information is used, these methods are called upwind methods.

Generally, upwind methods are more accurate than centred methods while the centred methods are simple, efficient but also quite diffusive as compared to the upwind methods.

It is thus desirable to construct a numerical flux that has the accuracy of upwind fluxes and simplicity of centred fluxes.

A recent approach for the construction of numerical fluxes that combine good properties of the upwind fluxes and simplicity of centred fluxes is the Multi-Stage (MUSTA) approach [6], [7], [8]. The key idea behind MUSTA is to compute a numerical flux via a multi-stage predictor-corrector procedure, using a simple flux at each stage. In essence, the MUSTA approach may be regarded as an approximate Riemann solver in which the predictor stage opens the Riemann fan without making use of knowledge of the structure of the solution of the RP. In addition, the information extracted from the opened Riemann fan is precisely the information required for the evaluation of the intercell numerical flux sought. The resulting upwind MUSTA schemes are found to be considerably more accurate than centred methods and comparable to upwind methods. In [7], [8] Toro proposed MUSTA method based on applying the first order FORCE flux for both the predictor and corrector steps. In [7] Toro introduced a generalization of FORCE flux, called GFORCE, that is a weighted average of Lax-Friedrich and Lax-Wendroff fluxes. For linear equations, this flux reduces identically to Godunov first order upwind method. Then Toro incorporated GFORCE in the framework of MUSTA approach resulting in a version that is called GMUSTA. Also, Toro implemented the numerical flux in the framework of high order finite volume WENO methods for multi-dimensional nonlinear hyperbolic systems.

The aim of this paper is to develop an improved version of MUSTA scheme. We propose to use the second order TVD flux [11], instead of first order flux, for both the predictor and corrector steps. Also, implement the numerical flux in framework of high order finite volume WENO methods The resulting schemes improve upon the original schemes in terms of better convergence, higher overall accuracy and better resolution of discontinuities in linearly degenerate fields, such as contact discontinuities.

The results demonstrate that our scheme is good alternative to current centred methods and conventional upwind methods as applied to very complex hyperbolic systems for which the solution of the RP is costly or unknown. Numerical results suggest that our scheme is superior to the original schemes. This is specially so for long time evolution problems containing both smooth and non-smooth features. The rest of the paper is organized as follows. In section 2, we briefly review the framework for constructing finite volume schemes and review the upwind and centred fluxes and TVD Runge- Kutta method for time stepping. In section 3 we review the MUSTA scheme. In section 4 we present the modified MUSTA using the second order scheme

[11]. In section 5 we show some numerical results for the scalar and system cases. In section 6 the extension to two dimensional problems is presented. Conclusions is presented in section 7.

2 Numerical Schemes

In this section we review the construction of high order finite volume schemes for hyperbolic systems in conservation form

$$U_t \ + F(U)_x \ = 0, \tag{2.1}$$

along with initial and boundary conditions. Here $U(x,t)$ is a vector of m components, the conserved variables and $F(U)$ is the physical flux vector. There are essentially two ways to discretize (2. 1). The first way is to develop fully discrete schemes. Consider a control volume in $x-t$ space $[x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}] \times [t^n, t^{n+1}]$, of dimensions $\Delta x = x_{j+\frac{1}{2}} - x_{j-\frac{1}{2}}$, $\Delta t = t^{n+1} - t^n$. Integrating (2.1) with respect to x and t over the volume we obtain

$$U_j^{n+1} = U_j^n - \frac{\Delta t}{\Delta x} \ \left[F_{j+\frac{1}{2}} - F_{j-\frac{1}{2}} \right] \tag{2.2}$$

where U_j^n is the space average of the solution in the cell $I_j = [x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}]$ at t^n and the flux $F_{j+\frac{1}{2}}$ is the time average of the physical flux at the cell interface $x_{j+\frac{1}{2}}$:

$$U_j^n = \frac{1}{\Delta x} \int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} U(x,t^n)dx \qquad\qquad F_{j+\frac{1}{2}} = \frac{1}{\Delta t} \int_{t^n}^{t^{n+1}} F(U(x_{j+\frac{1}{2}}))\, dt, \tag{2.3}$$

Another way to discretize (2.1) is to keep the time variable t continuous and consider semi-discrete schemes. Integrating (2.1) with respect to x only we obtain the following system of ordinary differential equations :

$$\frac{d}{dt}(U_j(t)) = \frac{-1}{\Delta x} \ \left[F_{j+\frac{1}{2}} - F_{j-\frac{1}{2}} \right] = L_j(U) \tag{2.4}$$

where

$$U_j(t) = \frac{1}{\Delta x} \int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} U(x,t)dx, \qquad\qquad F_{j+\frac{1}{2}} = F(U(x_{j+\frac{1}{2}},t)), \tag{2.5}$$

The description of the scheme is complete when a proper non-oscillatory flux is chosen. Next we briefly review possible upwind and centred fluxes which can be used.

2.1 Upwind fluxes

Upwind fluxes utilise wave propagation information contained in the differential equations to construct the numerical flux. The upwind method defines the numerical flux $F_{j+\frac{1}{2}}$ in terms of the solution, if available, of the corresponding RP :

$$\left.\begin{array}{l} U_t \ + F(U)_x \ = 0, \\ U(x,0) = \begin{cases} U_j^n, & x < x_{j+\frac{1}{2}} \\ U_{j+1}^n, & x > x_{j+\frac{1}{2}} \end{cases} \end{array}\right\} \tag{2.6}$$

The so called Riemann fan in the $x-t$ plane consists of $(m+1)$ constant states separated by m wave families, each one associated with a real eigenvalue $\lambda^{(k)}$. The Godunov numerical flux is found by first evaluating $U_{j+\frac{1}{2}}(x/t)$ at $x/t = 0$, that is along the t-axis, and then evaluating the physical flux vector $F(U)$ in (2.6), namely

$$F_{j+\frac{1}{2}}^{God} = F(U_{j+\frac{1}{2}}(0)) \tag{2.7}$$

When used in scheme (2. 2) this flux leads to a first order monotone upwind scheme. The exact solution will invariably involve at least one iterative procedure and thus in practice, whenever possible, one uses approximate solvers.

2.2 Centred fluxes

Centred fluxes contain no explicit wave propagation information. This makes them simple, efficient and applicable to very complex equations but also very diffusive as compared to upwind fluxes. Commonly, the numerical fluxes can be computed explicitly as algebraic functions of the initial condition in (2.6), namely

$$F_{j+\frac{1}{2}} = F_{j+\frac{1}{2}}(U_j^n, U_{j+1}^n) \tag{2.8}$$

2.2.1 First order fluxes

The most well known centred monotone flux is the Lax-Friedrich flux give by

$$F_{j+\frac{1}{2}}^{LF} = \frac{1}{2}[F(U_j^n) + F(U_{j+1}^n)] - \frac{1}{2}\frac{\Delta x}{\Delta t}(U_{j+1}^n - U_j^n) \tag{2.9}$$

This flux leads to a monotone first order accurate fully discrete scheme. Another centred flux is the first order centred (FORCE) flux introduced in [7,8] given by

$$F_{j+\frac{1}{2}}^{FORCE} = \frac{1}{4}\left\{F(U_j^n) + 2F(U_{j+\frac{1}{2}}^{n+\frac{1}{2}}) + F(U_{j+1}^n) - \frac{\Delta x}{\Delta t}(U_{j+1}^n - U_j^n)\right\} \tag{2.10a}$$

where

$$U_{j+\frac{1}{2}}^{n+\frac{1}{2}} = \frac{1}{2}[U_j^n + U_{j+1}^n] - \frac{1}{2}\frac{\Delta t}{\Delta x}[F(U_{j+1}^n) - F(U_j^n)] \tag{2.10b}$$

2.2.2 Second order fully discrete TVD scheme

In this section we review the second order TVD flux presented in [11]. The numerical flux of the second order fully discrete scheme [11] can be written in the two step flux:

$$\left.\begin{aligned} U^{YOU}_{j+\frac{1}{2}} &= \tfrac{1}{8}\left\{U^n_{j+2} + 3U^n_{j+1} + 3U^n_j + U^n_{j-1}\right\} - \tfrac{1}{8}\tfrac{\Delta t}{\Delta x}\left\{F^n_{j+2} + F^n_{j+1} - F^n_j - F^n_{j-1}\right\} \\ F^{YOU}_{j+\frac{1}{2}} &= F(U^{YOU}_{j+\frac{1}{2}}) \end{aligned}\right) \tag{2.11}$$

where $F_j = F(U_j)$. The scheme being second order accurate, is not TVD. It can be made TVD by the following procedure:

Consider a system of hyperbolic conservation laws in the form (2.1) and $A = \partial F/\partial U$ is the Jacobian matrix. The assumption that (2.1) is hyperbolic implies that $A(U)$ has real eigenvalues $\{\lambda^\ell(U)\}$ and a complete set of right eigenvectors $R^\ell(U), \ell = 1, 2, \ldots, m$. Hence the matrix

$$R(U) = (R^1, R^2, ...R^m) \tag{2.12a}$$

is invertible. Thus

$$R^{-1}AR = diag(\lambda^\ell) \tag{2.12b}$$

Here $diag(\lambda^\ell)$ denotes a diagonal matrix with diagonal elements λ^ℓ. Let $U_{j+\frac{1}{2}}$ denote some symmetric average of U_j and U_{j+1}(see Roe [4]). Let $\lambda^\ell_{j+\frac{1}{2}}$, R_{j+}, R^{-1}_{j+} denote the quantities λ^ℓ, R, R^{-1} evaluated at $U_{j+\frac{1}{2}}$. Define

$$\alpha_{j+\frac{1}{2}} = R^{-1}_{j+\frac{1}{2}}\Delta_{j+\frac{1}{2}}U, \qquad \Delta_{j+\frac{1}{2}}U = R_{j+\frac{1}{2}}\alpha_{j+\frac{1}{2}}$$

Now the TVD flux can be written in the form

$$F_{j+\frac{1}{2}} = F(U^{YOU}_{j+\frac{1}{2}}) - \frac{1}{2}R_{j+\frac{1}{2}}\Psi_{j+\frac{1}{2}} \tag{2.13}$$

where $\Psi_{j+\frac{1}{2}}$ is a limiter vector whose elements denoted by $\psi^\ell_{j+\frac{1}{2}}, \ell = 1, 2, \ldots, m$ and

$$\psi^\ell_{j+\frac{1}{2}} = \left|\lambda^\ell_{j+\frac{1}{2}}\right|(1 - \phi^\ell_{j+\frac{1}{2}})\alpha^\ell_{j+\frac{1}{2}} \tag{2.14a}$$

where

$$\phi^\ell_{j+\frac{1}{2}} = \phi^\ell_{j+\frac{1}{2}}\,(r^\ell_{j+\frac{1}{2}}) \tag{2.14b}$$

$$r^\ell_{j+\frac{1}{2}} = \frac{\alpha^\ell_{j-\frac{1}{2}}}{\alpha^\ell_{j+\frac{1}{2}}} \tag{2.14c}$$

The limiter function $\phi\,(r)$ is given by [5]

$$\phi\,(r) = \text{Max}\{0, \text{Min}(Qr, 1), \text{Min}(r, Q)\} \qquad 1 \le Q \le 2 \tag{2.14d}$$

If $Q = 1$, we get

$$\phi\,(r) = \text{Max}\{0, \text{Min}(r, 1)\} \tag{2.15a}$$

It is called minimod limiter and if $Q = 2$ we get

$$\phi\,(r) = \text{Max}\{0, \text{Min}(2r, 1), \text{Min}(r, 2)\} \tag{2.15b}$$

it is called superbee limiter.

2.3 Time Discretization

In current MUSTA schemes the numerical solution of the semi-discrete finite volume (2.4) is advanced in time by means of a TVD Runge-Kutta methods developed in [1]. Usually, the following third order three stage method is used (here we dropped the index j)

$$\begin{aligned} U^{(1)} &= U^n + \Delta t\, L(U^n) \\ U^{(2)} &= \frac{3}{4}U^n + \frac{1}{4}U^{(1)} + \frac{1}{4}\Delta t\, L(U^{(1)}) \\ U^{n+1} &= \frac{1}{3}U^n + \frac{2}{3}U^{(2)} + \frac{2}{3}\Delta t\, L(U^{(2)}) \end{aligned} \tag{2.16}$$

where $L(u)$ denotes the spatial operator (right hand side of (2.4)) at the appropriate level. In [3], it has been shown that, even with a very nice second order TVD spatial discretization, if the time discretization is by a non-TVD but linearly stable Runge-Kutta method, the result may be oscillatory. Thus it would always be safer to use TVD Runge-Kutta methods for hyperbolic problems.

3 MUSTA Numerical Fluxes

In general, the superior accuracy of upwind fluxes results from the opening of the RP and picking up a single value at the interface $x_{j+\frac{1}{2}}$. Complete (exact or approximate) Riemann solvers recognize all waves in the Riemann fan and therefore provide good resolution of delicate features of the flow, such as contact discontinuities. Centred fluxes can be regarded as very rough Riemann solvers in which Riemann fan is not opened at all. As a result the resolution of the solution by centred fluxes is very poor.

A very simple and general approach to the construction of numerical fluxes, which combines the simplicity of centred fluxes and the good accuracy of the upwind methods, is the Multi-Stage (MUSTA) approach. The key idea of the original MUSTA is to open the Riemann fan by evolving in time the initial data U_j^n and U_{j+1}^n in (2.6) via the governing equations.

In the MUSTA flux approach we first evolve in time the data states, for a number of stages, using a predictor flux function $F^{(P)}$. In this manner, starting with $U_j^{(0)} = U_j^n$ and $U_{j+1}^{(0)} = U_{j+1}^n$, after k stages we have the evolved data $U_j^{(k)} and\ U_{j+1}^{(k)}$. The sought numerical flux is computed by using a corrector flux function $F^{(C)}(V, W)$, in which the arguments are evolved states i. e.,

$$F_{j+\frac{1}{2}}^{MUSTA} = F^{(C)}(U_j^{(k)}, U_{j+1}^{(k)}) \tag{3.1}$$

In [8], Toro proposed MUSTA scheme based on applying the first order FORCE flux (2.11) for both the predictor and corrector steps, namely

$$F^{(P)}(U, V) = F^{(C)}(U, V) = F_{j+\frac{1}{2}}^{FORCE}(U, V) \tag{3.2}$$

An algorithm of MUSTA scheme works as follows. The multi-staging is started by setting s=0, $U_j^{(0)} = U_j^n$ and $U_{j+1}^{(0)} = U_{j+1}^n$. We then do the following:

Step 1 : Flux evaluation

$$\left.\begin{array}{l} F_j^{(s)} = F(U_j^{(s)}), \quad F_{j+1}^{(s)} = F(U_{j+1}^{(s)}), \\ U_{j+\frac{1}{2}}^{(s)} = \frac{1}{2}\left(U_j^{(s)} + U_{j+1}^{(s)}\right) - \frac{1}{2}\frac{\Delta t}{\Delta x}\left(F_{j+1}^{(s)} - F_j^{(s)}\right), \qquad F_M^{(s)} = F(U_{j+\frac{1}{2}}^{(s)}), \\ F_{j+\frac{1}{2}}^{(s)} = \frac{1}{4}\left(F_j^{(s)} + 2F_M^{(s)} + F_{j+1}^{(s)}\right) - \frac{\Delta x}{\Delta t}\left(U_{j+1}^{(s)} - U_j^{(s)}\right) \end{array}\right\} \tag{3.3a}$$

If the prescribed number of stages k has been reached, STOP.
Otherwise
Step 2 : Opening of Riemann fan

$$U_j^{(s+1)} = U_j^{(s)} - \frac{\Delta t}{\Delta x}\left(F_{j+\frac{1}{2}}^{(s)} - F_j^{(s)}\right), \quad U_{j+1}^{(s+1)} = U_{j+1}^{(s)} - \frac{\Delta t}{\Delta x}\left(F_{j+1}^{(s)} - F_{j+\frac{1}{2}}^{(s)}\right), \tag{3.3b}$$

Step 3 : Go to Step 1 Toro noticed that the number of stages between 3 and 4 gives numerical results are comparable with those from the most accurate of fluxes, namely, the first order Godunov upwind flux used in conjunction with the exact Riemann solver.

4 High order MUSTA flux

The MUSTA first order scheme presented in [7,8] may be extended to high order of accuracy by substituting the use of first order FORCE by high order but non-oscillatory flux.

4.1 TVD-MUSTA scheme

In this section we propose to use the second order TVD flux (2.11), instead of first order FORCE flux, in the predictor and corrector steps. The extension of the scheme can be carried out by two algorithms which are summarized in the following:
Algorithm 1
Step 1 : Flux evaluation

$$\left.\begin{array}{l} F_j^{(s)} = F(U_j^{(s)}), \quad F_{j+1}^{(s)} = F(U_{j+1}^{(s)}), \\ U_{j+\frac{1}{2}}^{(s)} = \frac{1}{8}\left(U_{j+2}^{(s)} + 3U_{j+1}^{(s)} + 3U_j^{(s)} + U_{j-1}^{(s)}\right) + \frac{1}{8}\frac{\Delta t}{\Delta x}\left(F_j^{(s)} - F_{j+1}^{(s)} - F_{j+2}^{(s)} + F_{j-1}^{(s)}\right), \\ F_{j+\frac{1}{2}}^{(s)} = F(U_{j+\frac{1}{2}}^{(s)}) - \frac{1}{2}R_{j+\frac{1}{2}}^{(s)}\Psi_{j+\frac{1}{2}}^{(s)} \end{array}\right\} \tag{4.1a}$$

If the prescribed number of stages k has been reached, STOP.
Otherwise
Step 2 : Opening of Riemann fan

$$U_j^{(s+1)} = U_j^{(s)} - \frac{\Delta t}{\Delta x}\left(F_{j+\frac{1}{2}}^{(s)} - F_j^{(s)}\right), \quad U_{j+1}^{(s+1)} = U_{j+1}^{(s)} - \frac{\Delta t}{\Delta x}\left(F_{j+1}^{(s)} - F_{j+\frac{1}{2}}^{(s)}\right), \tag{4.1b}$$

Step 3 : Go to Step 1

Algorithm 2
Step 1 : Flux evaluation

$$\left.\begin{array}{l} F_j^{(s)} = F(U_j^{(s)}), \quad F_{j+1}^{(s)} = F(U_{j+1}^{(s)}), \\ U_{j+\frac{1}{2}}^{(s)} = \frac{1}{8}\left(U_{j+2}^{(s)} + 3U_{j+1}^{(s)} + 3U_j^{(s)} + U_{j-1}^{(s)}\right) + \frac{1}{8}\frac{\Delta t}{\Delta x}\left(F_j^{(s)} - F_{j+1}^{(s)} - F_{j+2}^{(s)} + F_{j-1}^{(s)}\right), \\ F_{j+\frac{1}{2}}^{(s)} = F(U_{j+\frac{1}{2}}^{(s)}) \end{array}\right\} \tag{4.2a}$$

If the prescribed number of stages k has been reached Go to step 4
Otherwise
Step 2 : Opening of Riemann fan

$$U_j^{(s+1)} = U_j^{(s)} - \frac{\Delta t}{\Delta x}\left(F_{j+\frac{1}{2}}^{(s)} - F_j^{(s)}\right), \quad U_{j+1}^{(s+1)} = U_{j+1}^{(s)} - \frac{\Delta t}{\Delta x}\left(F_{j+1}^{(s)} - F_{j+\frac{1}{2}}^{(s)}\right), \tag{4.2b}$$

Step 3 : Go to Step 1
Step 4

$$F_{j+\frac{1}{2}}^{(s)} = F(U_{j+\frac{1}{2}}^{(s)}) - \frac{1}{2}R_{j+\frac{1}{2}}^{(s)}\Psi_{j+\frac{1}{2}}^{(s)} \tag{4.2c}$$

STOP
The first algorithm uses the TVD flux in each stage while the second algorithm uses the oscillatory flux in each stage and then use the TVD procedure in the final step. Although the use of the first algorithm increase the computational cost (time) of the scheme, our experiments show that in some cases it is necessary in order to avoid spurious oscillations. Moreover, in some cases the use of first algorithm improves overall accuracy of the numerical solution. Therefore, in this paper we always carry out the extension by the first algorithm. It should be noted that the centred scheme considered above requires the computation of a time step Δt to be used, such that stability of the numerical method is ensured. Our way of choosing condition for the above schemes Δt is

$$CFL = \max_j\left(S_j^n \frac{\Delta t}{\Delta x}\right)$$

Here S_j^n is the maximum propagation speed in I_j at time level n. For our scheme the optimal stability condition of CFL is

$$CFL \leq 1$$

4.2 WENO- MUSTA

In this section we use the TVD-MUSTA flux constructed in the previous subsection as a building block in the state of art WENO methods. Here we use the fifth order WENO reconstruction [2].

5 Numerical Results

In this section, we test our schemes proposed here and compare them with the schemes presented in [7] and [8]. For the time discretization we use, throughout, the third order TVD Runge-Kutta method. An important issue is the choice of test problems. We would like to emphasize here the importance of using really long time evolution problems with solutions consisting of discontinuities and smooth parts.

We compare the following schemes:

1- MUSTORO: it is the original MUSTA.
2- GMUSTORO: it is the generalized MUSTA presented in [7]
3- WENTORO: it is GMUSTA with WENO reconstruction
4- MUSTYOU it is the second order MUSTA presented here with the semi-discrete finite volume (2.4).
5- MUSTYOU1: it is MUSTYOU in which the corrector step is second order TVD flux and the predictor step is first order flux.
6- WENYOU: it is MUSTYOU with fifth order WENO reconstruction.

For all the computations we use the superbee limiter (2.15b) and the number of stages $k = 3$.

5.1 Scalar equations

We study the performance of our schemes by applying them to the following problems:

Example 1. We consider the equation (2.1) with the smooth initial condition

$$u(x, 0) = \sin^4 \pi x \tag{5.1}$$

defined on $[-1, 1]$ and periodic boundary conditions. The results obtained at $t = 1$, are shown in tables 1 and 2. It can be seen that MUSTYOU and MUSTYOU1 have smaller error size than both MUSTA and GMUSTA schemes and their order of accuracy is quite better. This due to the use of second order TVD flux in MUSTYOU and MUSTYOU1 instead of first order FORCE and GFORCE fluxes in MUSTA and GMUSTA schemes. Comparing MUSTYOU and MUSTYOU1 schemes we observe that the use of second order fluxes in both corrector and predictor steps improves the accuracy and convergence properties of the scheme. We note that WENYOU scheme is even more than fourth order accurate while the WENTORO scheme is approximately third order accurate. As expected the WENYOU scheme is the most accurate in both error sizes and order of accuracy.

N	MUSTORO	MUSTORO	GMUSTORO	GMUSTORO	WENTORO	WENTORO
	L_1 error	L_1 order	L_1 error	L_1 order	L_1 error	L_1 order
80	4.4115E-3		2.6807E-3		1.4394E-4	
160	1.4953E-3	1.56	7.5727E-4	1.823	1.4310E-5	3.33
320	6.2572E-4	1.257	1.8404E-4	2.041	1.8336E-6	2.96
640	2.2519E-4	1.474	5.4346E-5	1.7598	2.0666E-7	3.15

Table 1 $t = 1$

N	MUSTYOU	MUSTYOU	MUSTYOU1	MUSTYOU1	WENYOU	WENYOU
	L_1 error	L_1 order	L_1 error	L_1 order	L_1 error	L_1 order
80	6.489E-4		1.3458E-3		1.3368E-4	
160	9.936E-5	2.707	2.5607E-4	2.394	9.0628E-6	3.883
320	1.5568E-5	2.674	5.17104E-5	2.308	3.736E-7	4.600
640	2.4027E-6	2.696	1.03052E-5	2.327	1.8266E-8	4.3544

Table 2 $t = 1$

Example 2 We now consider the equation (2.1) for $m = 1$ with the initial condition [3]

$$u(x,0) = \begin{cases} \frac{1}{6}[G(x,z-\delta) + G(x,z+\delta) + 4G(x,z)], & -0.8 \le x \le -0.6 \\ 1, & -0.4 \le x \le -0.2 \\ 1 - |10(x-0.1)| & 0 \le x \le 0.2 \\ \frac{1}{6}[F(x,a-\delta) + F(x,a+\delta) + 4F(x,a)], & 0.4 \le x \le 0.6 \\ 0, & \text{otherwise} \end{cases} \tag{5.2}$$

with periodic boundary condition on $[-1, 1]$.
Where $G(x,z) = \exp(-\beta(x-z)^2)$, $\quad F(x,a) = \{\max(1-\alpha^2(x-a)^2, 0\}^{\frac{1}{2}}$.

The constants are taken as $a = 0.5$, $z = -0.7$, $\delta = 0.005$, $\alpha = 10$ *and* $\beta = (\log 2)/36\delta^2$. This initial condition consists of several shapes which are difficult for numerical methods to resolve correctly. Some of these shapes are not smooth and the other are smooth but very sharp. We compute the solution at $t = 8$ and very long time $t = 2000$. For comparison, firstly, we compute the solution at $t = 8$ on the mesh of 200 cells and CFL = 0.8. Figures 1-4 show the results of GMUSTORO, MUSTYOU, WENTORO and WENYOU schemes respectively. The full line is the exact solution and the symbols are the numerical solutions.

Figure 1 shows the performance of GMUSTORO scheme. As seen in the figure, the results look satisfactory for smooth parts, contact discontinuities and peaks. It is obvious that the results obtained by MUSTYOU, in figure 2, shows an improvement in the whole graphs. The results in figure 4 obtained with WENTYOU is superior to those obtained with all the other methods To illustrate the long time behaviour of the resulting schemes, figures 5-8 show the results using GMUSTORO, MUSTYOU, WENTORO and WENYOU schemes respectively for output time $t = 2000$. The MUSTYOU scheme is better than the original GMUSTORO. As expected the WENTORO and WENYOU schemes, from figures 7 and 8 produce the most accurate results (particularly WENYOU) for all parts of the solution, including the square pulse.

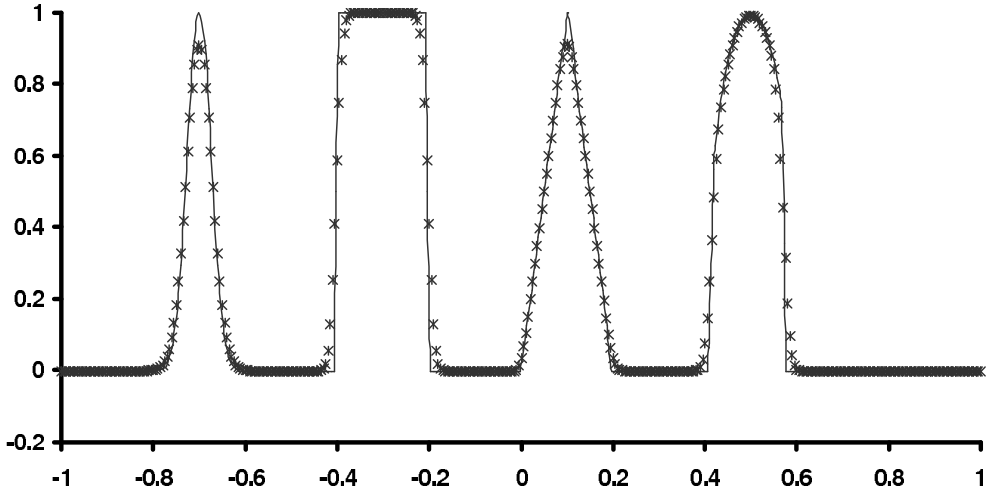

Figure 1. Solution of equation (5.2) by using GMUSTORO scheme at $t = 8$.

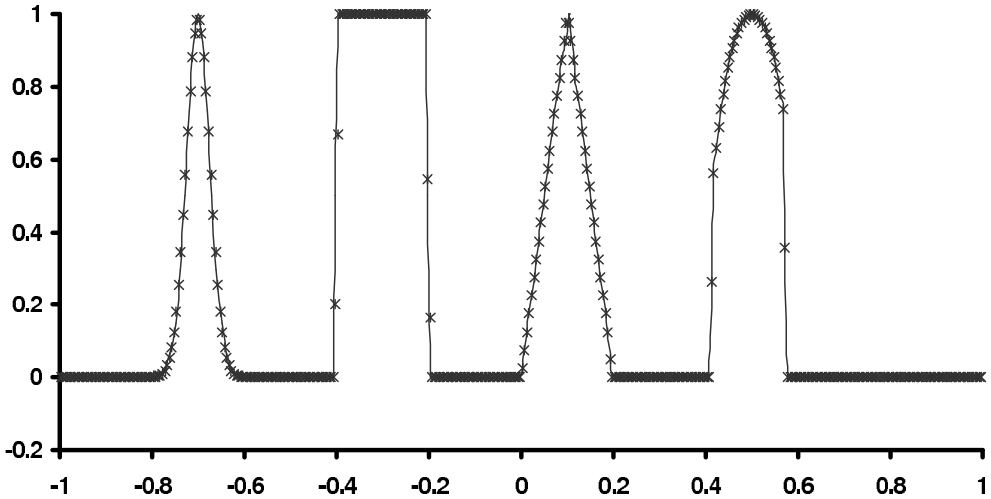

Figure 2. Solution of equation (5.2) by using MUSTYOU scheme at $t = 8$.

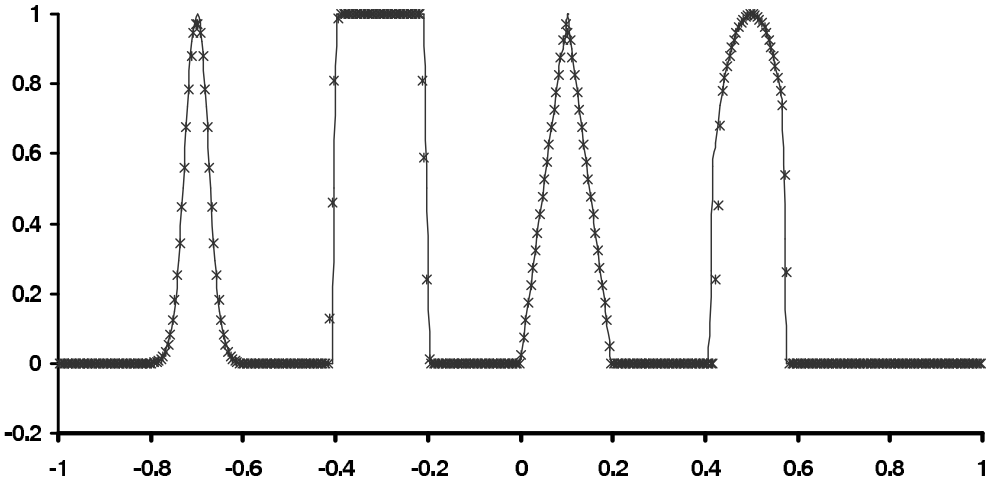

Figure 3. Solution of equation (5.2) by using WENTORO scheme at $t = 8$.

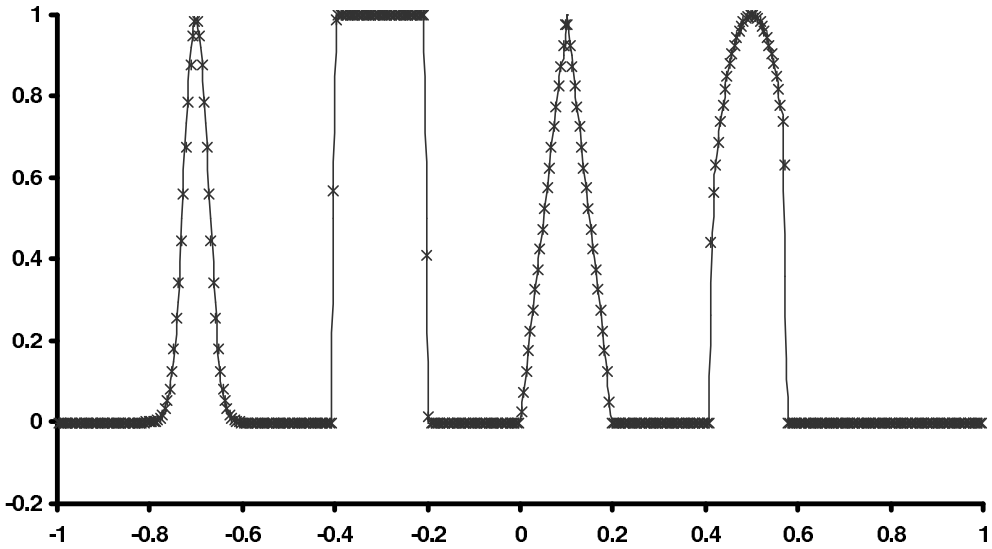

Figure 4. Solution of equation (5.2) by using WENTYOU scheme at $t = 8$.

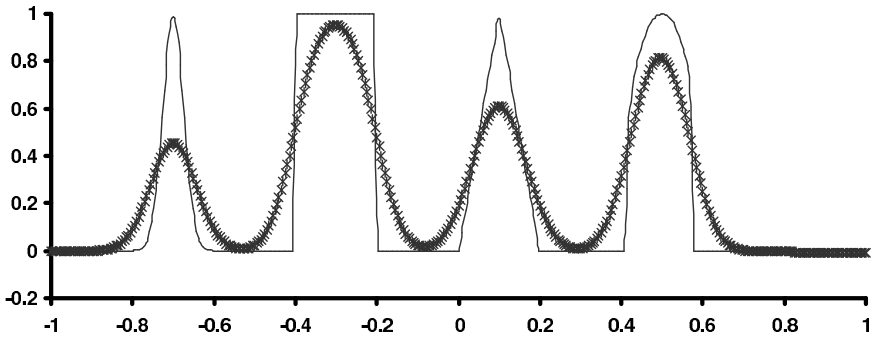

Figure 5. Solution of equation (5.2) by using GMUSTORO scheme at $t = 2000$.

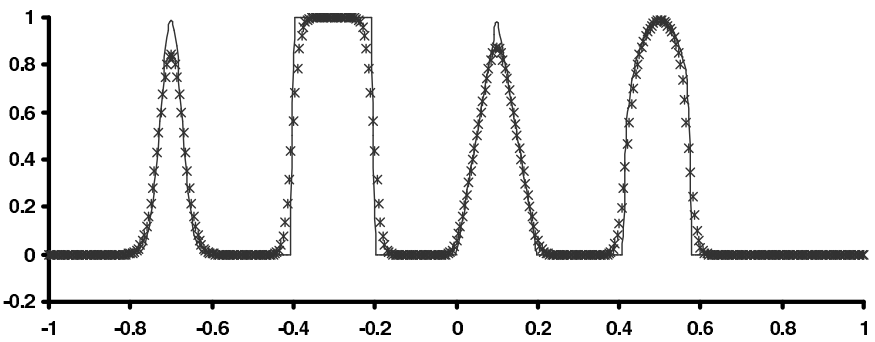

Figure 6. Solution of equation (5.2) by using MUSTYOU scheme at $t = 2000$.

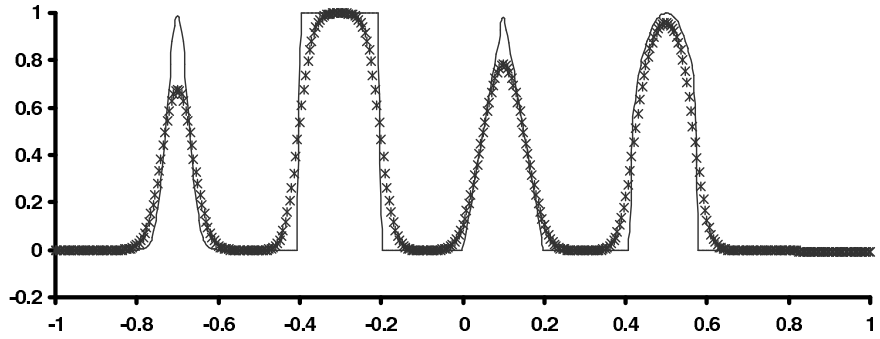

Figure 7. Solution of equation (5.2) by using WENTORO scheme at $t = 2000$.

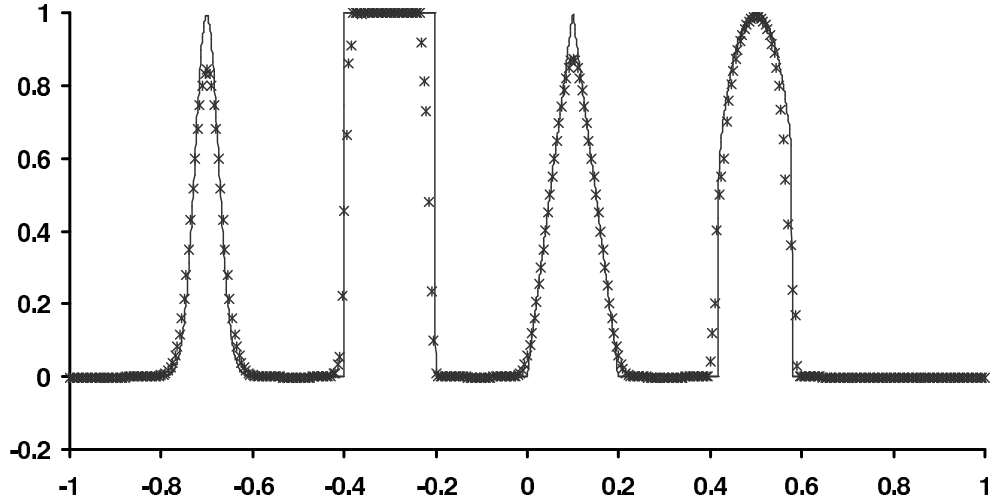

Figure 8. Solution of equation (5.2) by using WENYOU scheme at $t = 2000$.

5.2 Systems of equations

We apply our new schemes to the system of Euler equations of gas dynamics

$$U_t + F(U)_x = 0, \tag{5.3}$$

where $U = (\rho, \rho u, E)^T$ and $F(U) = (\rho u, \rho u^2 + P, u(E+P))^T$. where ρ is the density, u is the velocity, P is the pressure, $E = \frac{1}{2}\rho u^2 + \frac{P}{(\gamma-1)}$ is the total energy and γ is the ratio of specific heats, taken as 1.4 here.

Example 3 To show the advantages of our methods, we will solve a problem with a rich smooth structure and a shock wave. A typical example for this is the problem of shock interaction with entropy waves [3]. We solve the Euler equations (5.3) with a moving Mach = 3 shock interacting with sine waves in density; i. e., initially [3]

$$\begin{array}{ll} (\rho_L, u_L, P_L) = (3.857143, 2.629369, 10.3333), & \text{for } x < -4 \\ (\rho_R, u_R, P_R) = (1 + 0.2\sin 5x,\ 0,\ 1), & \text{for } x > -4 \end{array} \tag{5.4}$$

The flow contains physical oscillations which have to be resolved by the numerical method. We compute the solution at $t = 1.8$. Figures 9-12 show the computed density by GMUSTORO, MUSTYOU, WENTORO and WENYOU schemes respectively against the reference solution [3]. Here we use 200 grid points and the CFL =

0.8. It is clear that the results obtained by MUSTYOU and WENYOU schemes are better than MUSTORO and WENTORO schemes. Comparing the results in the figures with the results shown in [3] (see figures 11-14) we observe that our schemes MUSTYOU and WENYOU schemes are more accurate than the others and is less expensive because we use here 200 cells versus 400 cells in [3].

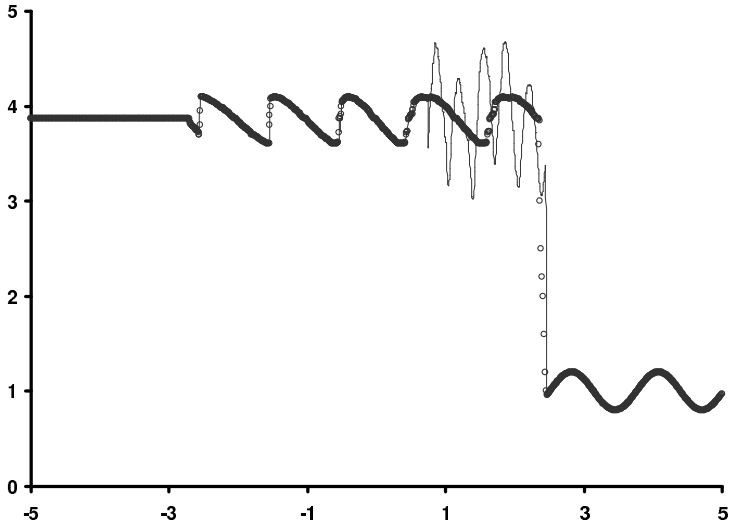

Figure 9. Solution of equation (5.4) using GMUSTORO scheme.

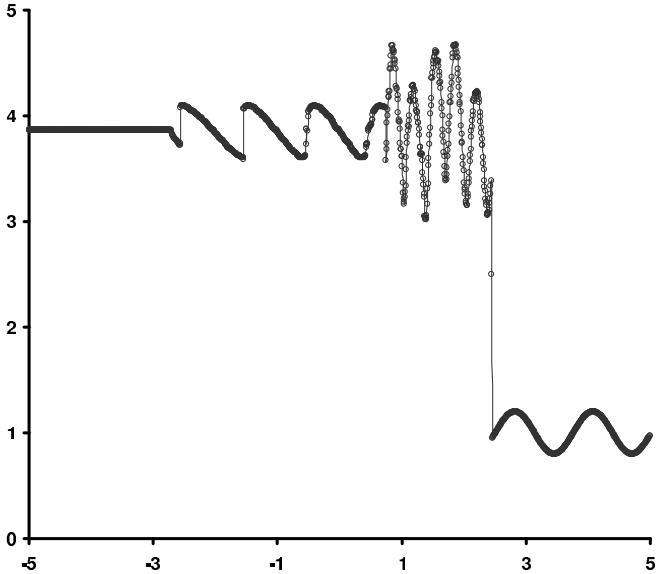

Figure 10. Solution of equation (5.4) using MUSTYOU scheme.

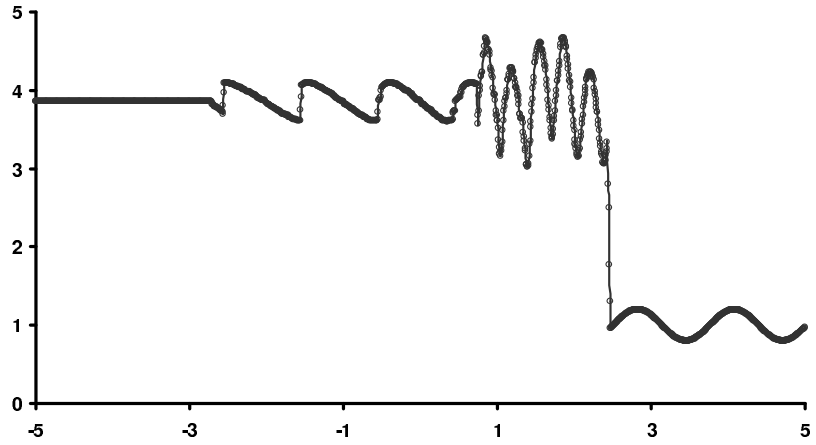

Figure 11. Solution of equation (5.4) using WENTORO scheme.

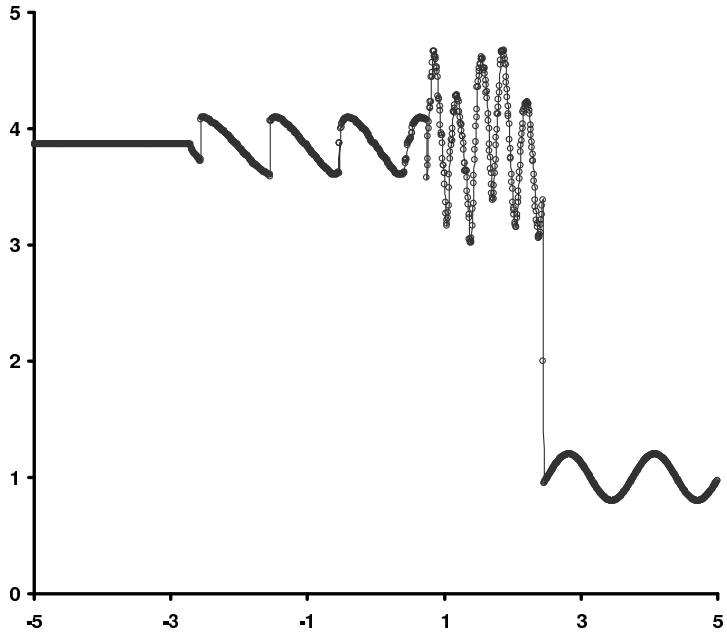

Figure 12. Solution of equation (5.4) using WENYOU scheme.

5.3 Computing time and Efficiency of the schemes

Here we discuss the computational efficiency of the schemes. Table 3 shows the CPU times for all schemes applied to Euler equations with initial data (5.4) on the mesh of 200 cells and same CFL = 0.8 and K= 3. It is noticed from the table that MUSTORO and GMUSTORO are the fastest schemes because they are Riemann solver free. The MUSTYOU and MUTYOU1 are around 1.5 times slower than Toros schemes. This is due to the use of TVD fluxes instead of first order ones. Considering the improvements in accuracy over the Toros schemes, this additional computational costs is not significant. Note that the fastest WENO schemes, WENTORO, is around five times slower than the MUSTORO. This is due to the use of expensive WENO reconstruction. The WENYOU is around 21% slower than WENTORO. However, again this difference in the speed is more than compensated by the

improvement in accuracy. Finally, we observe that MUSTYOU is only around 12% slower than MUTYOU1. From the numerical results presented here this difference is more than compensated by the improvement in accuracy.

scheme	MUSTORO	GMUSTORO	WENTORO	MUSTYOU	MUSTYOU1	WENYOU
CPU time	12	13	59	18	16	69

Table 3

6 Extension to multidimensional problems

The present schemes can be applied to multidimensional problems by means of space operator splitting. As an example we consider the two dimensional, Euler equations

$$U_t + [F(U)]_x + [G(U)]_y = 0 \tag{6.1}$$

where $U = (\rho, \rho u, \rho v, E)^T$, $F(U) = (\rho u, P + \rho u^2, \rho uv, u(P + E))^T$, $G(U) = (\rho v, \rho uv, P + \rho v^2, v(P + E))^T$
There are several versions of space splitting. Here we take the simplest one, whereby the two dimensional problem (6.1) is replaced by the sequence of two one-dimensional problems

$$U_t + [F(U)]_x = 0 \tag{6.2a}$$
$$U_t + [G(U)]_y = 0 \tag{6.2b}$$

If the data U^n at time level n for problem (6.1) are given, the solution U^{n+1} at time level $n + 1$ is obtained in the following two steps:
a) solve equation (6.2a) with data U^n to obtain an intermediate solution $\bar{U}^{n+1}$ (x-sweep);
b) solve equation (6.2b) with data $\bar{U}^{n+1}$ to obtain the complete solution U^{n+1} (y-sweep);
For three dimensional problems there is an extra z-sweep.

6.1 Double Mach reflection problem

The governing equation for this problem is the two dimensional Euler equations (6.1). The computational domain is $[0, 4] \times [0, 1]$. The reflecting wall lies at the bottom of the computational domain starting from $x = \frac{1}{6}$. Initially a right moving Mach 10 shock is positioned at $(x, y) = (\frac{1}{6}, 0)$ and makes 60° angle with the x-axis. For the bottom boundary, the exact post-shock condition is imposed from $x = 0$ to $x = \frac{1}{6}$ and a reflective boundary condition is used for the rest of the x-axis. At the top boundary of the computational domain, the data is set to describe the exact motion of the Mach 10 shock; consult [10] for a detailed discussion of this problem.

Figures 13 shows the computed density by MUSTYOU scheme on the 480×120 and 960×240 cells We observe that the scheme produces the flow pattern generally accepted in the present literature [10] as correct. All discontinuities are well resolved and correctly positioned.

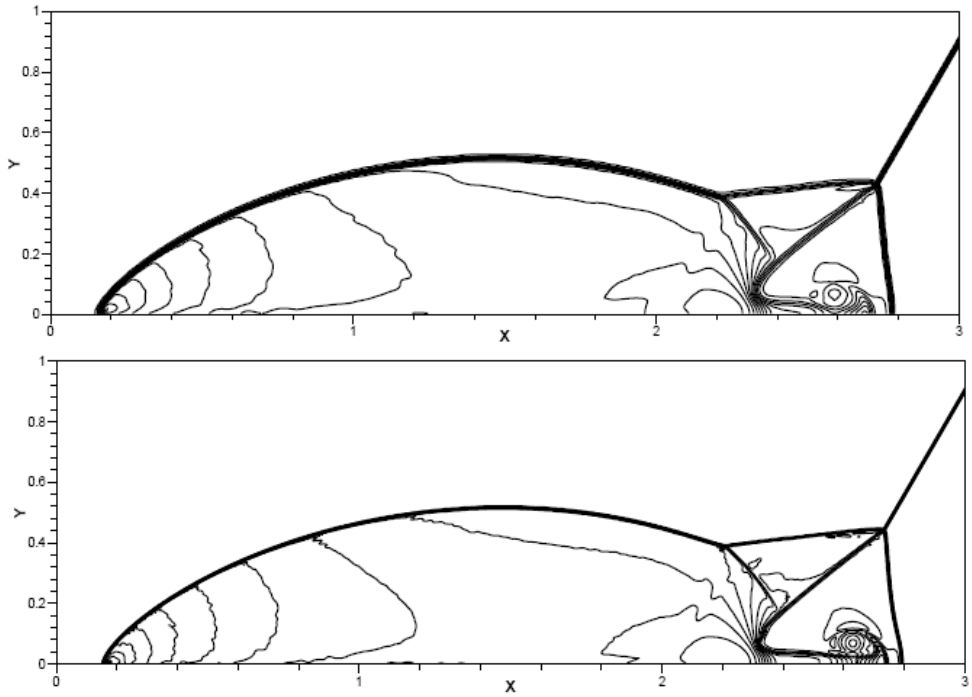

Figure 13. Double Mach refection problem for the MUSTYOU method. Meshes of (top) 480×120 and 960×240 (down)

7 Conclusions

In this paper, we proposed to use a second order TVD flux as a building block in MUSTA approach instead of the first order flux used in the original MUSTA. The numerical solution is advanced by TVD Runge-Kutta method. The resulting scheme improves upon the original MUSTA scheme in terms of better convergence, higher overall accuracy, better resolution of discontinuities and find its justification when solving very complex systems for which the solution of the RP is costly or unknown. Also, we extend the the scheme in the framework of high order WENO methods. We apply the proposed schemes to one and two dimensional problems.

References

[1] S. Gotlieb and C. W. Shu. *Total variation diminishing Runge-Kutta schemes*, Math. comp. 67(1998) pp. 73-85.

[2] Jiang G. S. and Shu C. W. *Efficient implementation of Weighted ENO schemes*, J. Comut. Phys. 126(1996) pp. 202-228

[3] J. Qiu and C-W. Shu. *On the construction, comparison, and local characteristic decomposition for the high order central WENO schemes*, J. Comput. Physics 183(2002) pp. 187-209.

[4] Roe P. L. *Approximate Riemann solvers parameter vectors and difference schemes*, J. Comput. Phys. 43 (1981) 357-372

[5] Sweby P. K. *High resolution schemes using flux limiters for hyperbolic conservation laws.* , SIAM, J. Num. Anal., 21 (1984) pp995-1011.

[6] Titarev V. A., Toro E. F. *MUSTA schemes for multi-dimensional hyperbolic systems: analysis and improvements*, Int. J. Numer. Meth. Fluid 49 (2005) pp 117-147.

[7] Toro E. F., Titarev V. A. *MUSTA fluxes for system of conservation laws*, J. Cpmput. Phys. 216 (2006) V2 pp. 403-429.

[8] Toro E. F., Titarev V. A. *Multi- Stage predictor-corrector fluxes for hyperbolic equations*, Technical Report N10303-NPA, Isaac Newton Institute for Mathematical sciences, University of Cambridge, U. K. (2003).

[9] Toro E. F. and S. J. Billett. *Centered TVD schemes for hyperbolic conservation laws*, IMA J. Numerical Analysis, 20 (2000) pp. 47-79.

[10] Woodward P. and Colella P. *The numerical solution of two dimensional fluid flow with strong waves*, J. Comput. Phys. 54(1984) pp 115-173.

[11] Yousef H. Zahran. *A Family of TVD second order schemes of nonlinear scalar conservation laws*, Comptes rendus de l'Acad. Bulgare des Sci. 56 (2003) No. 4 pp. 15-22.

Physics and Mathematics Department
Faculty of Engineering
Port Fouad- Port Said
POB No. 42523
Egypt
E-Mail : yousef_hashem_zahran@yahoo. com