

UCLA
COMPUTATIONAL AND APPLIED MATHEMATICS

**Essentially Nonoscillatory Postprocessing
Filtering Methods**

**F. Lafon
S. Osher**

**September 1991
CAM Report 91-21**

**Department of Mathematics
University of California, Los Angeles
Los Angeles, CA. 90024-1555**

Essentially Nonoscillatory Postprocessing Filtering Methods

F. Lafon*

C.E.A - CEL-V BP 27

94195 Villeneuve Saint Georges, France

& S. Osher†

Department of Mathematics - University of California

Los Angeles, CA 90024-1555

September 3, 1991

Abstract

High order accurate centered flux approximations used in the computation of numerical solutions to nonlinear partial differential equations produce large oscillations in regions of sharp transitions. In this paper, we present a new class of filtering methods denoted by ENO-LS (Essentially Nonoscillatory Least Squares) which constructs an upgraded filtered solution that is close to the physically correct weak solution of the original evolution equation. Our method relies on the evaluation of a least squares polynomial approximation to oscillatory data using a set of points which is determined via the ENO framework.

Numerical results are given in one and two space dimensions for both scalar and systems of hyperbolic conservation laws. Computational running time, efficiency and robustness of the method are illustrated in various examples such as Riemann initial data for both Burgers' and Euler's equations of gas dynamics. In all standard cases the filtered solution appears to converge numerically to the correct solution of the original problem. Some interesting results based on nonstandard central difference schemes, which exactly preserve entropy, and have been recently shown generally not to be weakly convergent to a solution of the conservation law, are also obtained using our filters.

*Research supported by ONR grant N00014-86-K-0691 and NSF grant DMS 88-11863 .

†Research supported by ONR grant N00014-86-K-0691, DARPA grant in the ACMP program and NSF grant DMS 88-11863.

1 Introduction

Numerical improvements in the computation of high order accurate numerical solutions to nonlinear hyperbolic conservation laws have been recently obtained. Hence, following Total Variation Diminishing (TVD) schemes, the Essentially Nonoscillatory (ENO) method has been introduced and proved to be very efficient in the computation of high accurate numerical solutions for several types of physical problems including Computational Fluid Dynamics (CFD) problems or front propagation using the Hamilton-Jacobi framework. However, these high accurate methods use a lot of computational time. For that reason, filtering methods were developed, beginning in the late eighties. The first one, described by B. Engquist and B. Sjogreen in [1], uses simple TVD and conservation properties to correct nonphysical spurious oscillations from one time step to another. The correction step consists in pushing numerical data points up or down to an acceptable level while preserving conservation. In [5], we presented a new class of filtering methods of any order of accuracy. Our method relies on switching fluxes at locations in which spurious oscillations are detected. This method was observed to be very efficient and its cost relatively low since high order ENO fluxes are evaluated at a few points only- a central difference method is used most often.

In this paper, we investigate some interesting computational properties of centered schemes after numerical oscillations have developed and propagated for some time. We define a new class of filtering methods that can be applied to highly oscillatory numerical solutions. This relies on the construction of an ENO stencil of points $([2, 7, 8])$ which is fitted with high degree polynomials from a least squares procedure. Our numerical filter is capable of smoothing oscillations having large amplitude and high frequency, but without removing sharp singularities which are crucial components of these solutions. Furthermore, the filtered solution seems to retain the oscillatory solution properties of the unfiltered schemes in some special "entropy preserving" cases as defined in [3] by J. Goodman and P.D. Lax, and in [6] by J. Liu and D. Levermore. We investigate some numerical examples using several centered approximations in order to illustrate the former property. The main conclusion indicates that for standard central differences, our filtered solution always converges accurately to the strong limit, whereas the predicted oscillatory behavior is retained even after using our filter in the examples of [3, 6].

Our main test problems will be inviscid Burgers' equation and the inviscid Euler equations of gas

dynamics. We first consider Burgers' equation:

$$U_t + \left(\frac{U^2}{2}\right)_x = 0, \quad (1)$$

with smooth initial condition $U(x, 0) = U_0(x)$, $U_0 \in C^\infty(0, 1)$, and periodic boundary conditions. We will discuss the main properties of the numerical solution obtained from some schemes based on the approximate fluxes:

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

$$F_{j+\frac{1}{2}} = \frac{7}{24}(U_{j+1}^2 + U_j^2) - \frac{1}{24}(U_{j+2}^2 + U_{j-1}^2), \quad (3)$$

$$F_{j+\frac{1}{2}} = \frac{37}{120}(U_{j+1}^2 + U_j^2) - \frac{8}{120}(U_{j+2}^2 + U_{j-1}^2) + \frac{1}{120}(U_{j+3}^2 + U_{j-2}^2), \quad (4)$$

$$F_{j+\frac{1}{2}} = \frac{1}{6}(U_{j+1}^2 + U_j U_{j+1} + U_j^2), \quad (5)$$

$$F_{j+\frac{1}{2}} = \frac{1}{2}U_j U_{j+1}. \quad (6)$$

The fluxes (2,3,4) are just standard central differencing of second, fourth and sixth order of accuracy, respectively; while (5,6) are the interesting examples analyzed in [6] and in [3], respectively. The oscillatory solution obtained from any of these fluxes is then corrected by the ENO-LS method, preserving the formal order of accuracy.

The Euler equations of gas dynamics:

$$U_t + (F(U))_x = 0, \quad (7)$$

$$U(x, 0) = U_0(x),$$

are to be solved for $t > 0$ and x in some interval Ω with appropriate boundary conditions, where

$$F(U) = vU + (0, p, vp)^T$$

and $U = (\rho, q, p)$, ρ is density, q is momentum, v is velocity, and p is the pressure. In this work, we use conventional second, fourth and sixth order central differencing with ENO-LS post processing applied to Euler's equations. See [4] for an analysis of the oscillatory Von Neumann-Richtmyer scheme approximating Euler's equation, [9].

2 The ENO-LS Filter, Algorithms:

The ENO-LS method mimics the construction of ENO polynomials but without involving the evolution equations. In short, we follow the algorithm just below:

Algorithm 2.1 • 1.) Compute N times the numerical solution of

$$\begin{aligned} U_t + A(U)_x &= 0 \\ U(x, 0) &= U_0(x) \end{aligned}$$

i.e we let $V_j^0 = U_0(x_j)$, and compute $V_j^N = I(V^0, N\Delta t)$, for all $j = 1, \dots, n$, where I is the solution semigroup operator that transforms the initial pointwise data V_j^0 to V_j^N after N iteration time steps.

• 2.) Filter the numerical solution computed in step 1.) by an iteration procedure similar to Jacobi or Gauss-Seidel elliptic solvers; first let:

$$W_j^0 = V_j^N$$

for all $j = 0, \dots, n$, then make use of primitive variables:

$$U_{j+\frac{1}{2}}^m = \sum_{i=0}^j W_i^m \Delta x_i, \quad (8)$$

and finally construct a sequence $\{U_{j+\frac{1}{2}}^m\}_{m=0, \dots, M}$ so that

$$U_{j+\frac{1}{2}}^{m+1} = E(U^m, U^{m+1}), \quad (9)$$

where M is defined from the stopping criterion:

$$\|W^{m+1} - W^m\| \leq \varepsilon,$$

for $m = 0, \dots, M - 1$, where ε is a small parameter of order Δx^α and

$$W_j^m = \frac{U_{j+\frac{1}{2}}^m - U_{j-\frac{1}{2}}^m}{\Delta x_j}. \quad (10)$$

Finally, let $\bar{V}_j^N = W_j^M$, for $j = 1, \dots, n$.

- 3.) Go to step 1.) unless $t = t_{max}$.

We notice that relation (10) and the use of primitive variables (8) implies the conservation property of the sequence $\{W_j\}_{j=1, \dots, n}$, i.e the resulting finite difference scheme is always in conservation form. Moreover, the number of correction steps M can be initially fixed so that the ratio $\frac{N}{M}$ is as large as desired. The operator E is a non trivial linear combination of $U_{j+\frac{1}{2}}^m, U_{j+\frac{1}{2}}^{m+1}$, for some j , as in point Jacobi or Gauss Seidel method. In the Jacobi procedure we have:

$$U_{j+\frac{1}{2}}^{m+1} = E(U_{\frac{1}{2}}^m, U_{\frac{3}{2}}^m, \dots, U_{j-\frac{1}{2}}^m, U_{j+\frac{1}{2}}^m, \dots, U_{n-\frac{1}{2}}^m),$$

and either

$$\begin{cases} U_{j+\frac{1}{2}}^{m+1} = E(U_{\frac{1}{2}}^{m+1}, \dots, U_{j-\frac{1}{2}}^{m+1}, U_{j+\frac{1}{2}}^m, \dots, U_{n-\frac{1}{2}}^m) & \text{if } j \text{ varies from } 0 \text{ to } n, \text{ or} \\ U_{j+\frac{1}{2}}^{m+1} = E(U_{\frac{1}{2}}^m, \dots, U_{j-\frac{1}{2}}^m, U_{j+\frac{1}{2}}^{m+1}, \dots, U_{n-\frac{1}{2}}^{m+1}) & \text{in the reverse direction;} \end{cases}$$

for the Gauss-Seidel method.

An important property of the ENO-LS algorithm comes from the fact that the corrected solution \bar{V}_j^N satisfies a conservation equation. To see that, we first use the relation (10) which can be rewritten as:

$$W_j^{m+1} = W_j^m + \frac{(E^{j+\frac{1}{2}}(U_{j+\frac{1}{2}-r_+}, \dots, U_{j+\frac{1}{2}+s_+}) - U_{j+\frac{1}{2}}^m) - (E^{j-\frac{1}{2}}(U_{j-\frac{1}{2}-r_-}, \dots, U_{j-\frac{1}{2}+s_-}) - U_{j-\frac{1}{2}}^m)}{\Delta x_j} \quad (11)$$

and then discuss the construction of the least squares process. In (11), the pair of integers (r_{\pm}, s_{\pm}) limit the width of the stencil used in the evaluation of the least squares polynomial. The appropriate stencil is defined as in the ENO algorithms (refer to [7], [8]). We briefly indicate the main steps leading to such a polynomial: We first compute the divided differences table of W^m and define the ENO stencil of points in the region which is the smoothest for successive space derivatives of W^m . We denote the ENO stencil in the set $(x_{j-r}, \dots, x_{j+s})$, where $r + s = p - 1$, and p is the number of data points that we want to take into account in the evaluation of the least squares polynomial. Hence if we denote this polynomial by

$$P^{j+\frac{1}{2}}(x) = \sum_{i=0}^q Y_i \varphi_i(x), \quad (12)$$

where $(\varphi_0, \dots, \varphi_q)$ are the basis functions of some polynomial space of degree q , then the unknown coefficients $Y = (Y_0, \dots, Y_q)$ are solutions of the linear system:

$$C^{j+\frac{1}{2}}Y = F^{j+\frac{1}{2}},$$

where $C^{j+\frac{1}{2}}$ is a $(q+1) \times (q+1)$ square matrix, $F^{j+\frac{1}{2}}$ is a $q+1$ column vector, and both can be computed from the basis functions:

$$\begin{aligned} C_{k,l}^{j+\frac{1}{2}} &= \sum_{i=-r}^s \varphi_k(x_{j+\frac{1}{2}+i})\varphi_l(x_{j+\frac{1}{2}+i}), \\ F_k^{j+\frac{1}{2}} &= \sum_{i=-r}^s U_{j+\frac{1}{2}+i}\varphi_k(x_{j+\frac{1}{2}+i}). \end{aligned}$$

The updated value $U_{j+\frac{1}{2}}^{m+1}$ follows from letting $x = x_{j+\frac{1}{2}}$ on the previously constructed LS polynomial:

$$U_{j+\frac{1}{2}}^{m+1} = P^{j+\frac{1}{2}}(x_{j+\frac{1}{2}}) = \sum_{i=0}^q \{(C^{j+\frac{1}{2}})^{-1} F^{j+\frac{1}{2}}\}_i \varphi_i(x_{j+\frac{1}{2}}).$$

The global conservation feature follows provided that we write:

$$U_{j+\frac{1}{2}}^{m+1} = \sum_{t=-r}^s \alpha_{j+\frac{1}{2}+t}^{j+\frac{1}{2}} U_{j+\frac{1}{2}+t}^{m+1},$$

where $l = 0$ or 1 depending on whether the Jacobi or the Gauss-Seidel method is used, and

$$\alpha_{j+\frac{1}{2}+t}^{j+\frac{1}{2}} = \sum_{i=0}^q \sum_{k=0}^q D_{i,k}^{j+\frac{1}{2}} \varphi_i(x_{j+\frac{1}{2}})\varphi_k(x_{j+\frac{1}{2}+t}),$$

where $D^{j+\frac{1}{2}} = (C^{j+\frac{1}{2}})^{-1}$.

Note first that the coefficients $\{\alpha_{j+\frac{1}{2}+t}^{j+\frac{1}{2}}\}_{t=-r, \dots, s}$ form a sequence of bounded real numbers, and second, that the basis functions $(\varphi_0, \dots, \varphi_q)$ can be appropriately chosen as a sequence of Chebyshev polynomials or some other set of orthogonal polynomials. The last choice permits us to compute directly the coefficients $\alpha_{j+\frac{1}{2}+t}^{j+\frac{1}{2}}$ without inverting the matrix $C^{j+\frac{1}{2}}$:

$$\alpha_{j+\frac{1}{2}+t}^{j+\frac{1}{2}} = \sum_{i=0}^q \frac{\varphi_i(x_{j+\frac{1}{2}})\varphi_i(x_{j+\frac{1}{2}+t})}{C_{i,i}^{j+\frac{1}{2}}}.$$

This yields a fast algorithm since matrix inversions are no longer needed.

We now present an extension of algorithm 2.1 to two space dimensions. The simplest possible extension would be to apply the previous algorithm in two sweeps. The first one will freeze one coordinate and correct the oscillatory solution with respect to the other free variable. The second one will simply reverse the role of each variable. This method, while simple, has difficulties near curved shocks. We shall use instead a fully two dimensional ENO-LS filtering algorithm. The latter will provide the construction of least squares polynomials in two space dimensions using a set of points which is chosen as the intersection of one dimensional ENO stencils of data points in each separate direction. The algorithm below describes our procedure.

Algorithm 2.2 • 1.) Compute N times the numerical solution of

$$\begin{aligned} U_t + A(U)_x + B(U)_y &= 0 \\ U(x, y, 0) &= U_0(x, y) \end{aligned}$$

using a very simple numerical method and then let $V_{i,j}^0 = U_0(x_j, y_i)$, and $V_{i,j}^N = I(V^0, N\Delta t)$, for all $j = 1, \dots, n_1$, and $i = 1, \dots, n_2$, where I is the solution semigroup operator that we have already encountered in previous algorithm.

• 2.) Let $W_{i,j}^0 = V_{i,j}^N$, and filter M times the primitive variable

$$U_{i+\frac{1}{2}, j+\frac{1}{2}}^m = \sum_{l,k=0}^{i,j} W_{l,k}^m \Delta x_k \Delta y_l, \quad (19)$$

by a Jacobi or point Gauss-Seidel iteration procedure, i.e perform:

$$U_{i+\frac{1}{2}, j+\frac{1}{2}}^{m+1} = E(U^m, U^{m+1}),$$

for positive integers m . Iterate as long as $\|W^{m+1} - W^m\| \leq \epsilon$, $m = 0, \dots, M-1$, and finally let the filtered solution: $\bar{V}_{i,j}^N = W_{i,j}^M$, for $j = 1, \dots, n_1$, $i = 1, \dots, n_2$.

The construction of the two dimensional least squares polynomial is as follows:

• 2.1) At the location $(x_{i+\frac{1}{2}}, y_{j+\frac{1}{2}})$, compute the ENO stencils of points $\{(x_{i-r_x}, y_j), \dots, (x_{i+1+s_x}, y_j)\}$, and $\{(x_i, y_{j-r_y}), \dots, (x_i, y_{j+1+s_y})\}$, for $r_x + s_x + 1 = n_1$ and $r_y + s_y + 1 = n_2$, where n_1 and n_2 are the preset maximum number of points along the x and y axis. These sets of

points form two segments crossing at (x_i, y_j) . Then, define the x and y ENO stencils of points along these y and x segments, respectively. The two dimensional ENO stencil is taken as the intersection of the union of the predefined x and y one dimensional ENO stencils (refer to figure (1)). The least squares polynomial is then simply defined on this two dimensional ENO stencil and is set to $P^{(i+\frac{1}{2}, j+\frac{1}{2})}(x, y) = \sum_{k=0}^q Y_k \varphi_k(x, y)$. Again, the unknowns coefficients Y_k , $k = 1, \dots, p$ are computed by solving the linear system $C^{(i+\frac{1}{2}, j+\frac{1}{2})} Y = F^{(i+\frac{1}{2}, j+\frac{1}{2})}$.

- 2.2) Let $U_{i+\frac{1}{2}, j+\frac{1}{2}}^{m+1} = P^{(i+\frac{1}{2}, j+\frac{1}{2})}(x_{i+\frac{1}{2}}, y_{j+\frac{1}{2}})$.
- 2.3) Recover the conservative solution

$$W_{i,j}^{m+1} = \frac{U_{i+\frac{1}{2}, j+\frac{1}{2}}^{m+1} + U_{i-\frac{1}{2}, j-\frac{1}{2}}^{m+1} - U_{i+\frac{1}{2}, j-\frac{1}{2}}^{m+1} - U_{i-\frac{1}{2}, j+\frac{1}{2}}^{m+1}}{\Delta x \Delta y}, \quad (14)$$

for $m = 0, \dots, M - 1$.

- 3.) Go to step 1.) unless $t = t_{max}$.

An example of two dimensional ENO stencil is given in figure 1. The main interesting feature of such construction is based on the localization of the least oscillatory part of the solution within the much larger rectangle $[x_{j-r_x}, x_{j+s_x}] \times [y_{i-r_y}, y_{i+s_y}]$.

In section 4, we investigate the two dimensional Burgers' equation and study numerical propagation of a shock along the radial axis. With centered fluxes, some spurious numerical oscillations propagate in the direction of the flow; however the two dimensional ENO-LS filtering method is able to filter all these oscillations while still giving the correct location of the curved shock wave.

Next, we consider hyperbolic systems of conservation laws exemplified by the inviscid Eulers' equations of compressible gas dynamics:

$$\begin{aligned} U_t + F(U)_x &= 0, \\ U(x, 0) &= U_0(x) \end{aligned}$$

for which there exists a complete set of real eigenvectors and eigenvalues; i.e $\nabla F(U) = P^{-1} \Lambda P$, where Λ is a diagonal matrix with entries $\lambda_1 \leq \lambda_2 \dots \leq \lambda_d$, and P is a matrix whose columns define a complete set of eigenvectors of the system. Note that the eigenvalues can be multiple, which is the case for the Euler equations of gas dynamics in two space dimensions. Using the eigenvector decomposition, a field by

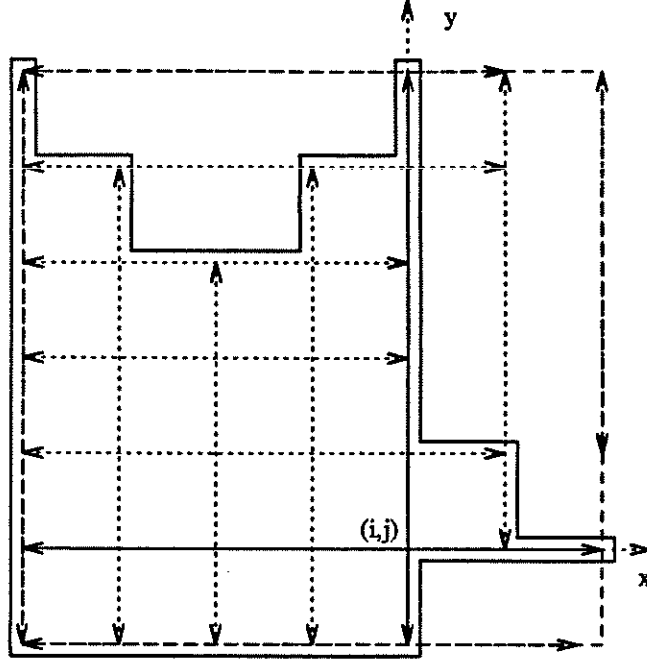


Figure 1: 2D ENO-LS stencil.

field approach is used, i.e the fluxes will be corrected in each fields (our filter sometimes failed to remove oscillations near strong shocks when applied to the conserved quantities). Briefly, we proceed as follows:

Algorithm 2.3 (Field by field ENO-LS method) • 2-1.)

Let

$W^m = (W_1^m, \dots, W_q^m)^T$, and define the primitive variable $V_{j+\frac{1}{2}}^m = \sum_{i=1}^j W_i^m \Delta x$.

- 2-2.) Compute the left and right eigenvectors $L_{j+\frac{1}{2}}^{m,(k)}$, $R_{j+\frac{1}{2}}^{m,(k)}$, for $k = 1, \dots, d$ (d is the number of equations) of the Roe matrix (see e.g. [2]) $A(W_j^m, W_{j+1}^m)$, and decompose the primitive vector $V_{j+\frac{1}{2}}^m$ along each characteristic field:

$$\begin{aligned} \sigma_{j+\frac{1}{2}}^{m,(k)} &= L_{j+\frac{1}{2}}^{m,(k)} \cdot V_{j+\frac{1}{2}}^m \\ Q_{j+\frac{1}{2}}^m &= \sum_{k=1}^d \sigma_{j+\frac{1}{2}}^{m,(k)} R_{j+\frac{1}{2}}^{m,(k)}, \end{aligned}$$

for which we have frozen the index j in order to get the same decomposition for all neighboring points $x_{j+\frac{1}{2}+l(k)}$, for $l(k) = -r(k), \dots, +s(k)$ involved in the calculation of the least squares polynomial which is constructed in step 2-3.).

- 2-3.) Select an ENO stencil of p points, i.e. $\{x_{j-r(k)}, \dots, x_{j+s(k)}\}$, for $r(k) + s(k) + 1 = p$; and then define the least squares polynomial $P^{(j+\frac{1}{2})}$ of degree q so that:

$$\sigma_{j+\frac{1}{2}}^{m+1,(k)} = P^{(j+\frac{1}{2})}(\sigma_{j+\frac{1}{2}-r(k)}^{m,(k)}, \dots, \sigma_{j+\frac{1}{2}+s(k)}^{m,(k)}; x_{j+\frac{1}{2}}).$$

- 2-4.) Transform back the filtered field by field solution to the primitive vector:

$$V_{j+\frac{1}{2}}^{m+1} = \sum_{k=1}^d \sigma_{j+\frac{1}{2}}^{m+1,(k)} R_{j+\frac{1}{2}}^{m,(k)},$$

and finally recover the desired physical variables:

$$W_j^{m+1} = \frac{V_{j+\frac{1}{2}}^{m+1} - V_{j-\frac{1}{2}}^{m+1}}{\Delta x}. \quad (15)$$

- 2-5.) Iterate until the stopping criterion

$$\|W^{m+1} - W^m\| \leq \varepsilon,$$

for $m = 0, \dots, M - 1$, is reached; and finally let $\bar{U}_j^N = W_j^M$.

Note that this algorithm can be extended to two space dimensions by correcting separately oscillatory fields involved in the x and y fluxes, respectively. Moreover, this algorithm does not make use of the evolution equations but does use the eigenfunction decomposition in order to track efficiently the propagation of spurious oscillations.

To conclude this section, we indicate that we mainly supposed that the numerical oscillations always propagate with the flow speed along local characteristic fields and that the amplitudes of such oscillations are not too large so that the oscillatory solution does not become unbounded, e.g., negative density and/or pressure is not allowed. In all our numerical experiments, we had to turn the filter on not only to recover an acceptable final solution but also to reduce the amplitude and frequency of spurious oscillations during the calculations. Hence, we usually preset the value of the ratio $\frac{N}{M}$ in the numerical experiments just after singularities have developed.

3 Numerical Convergence Study:

In this section, we investigate the numerical convergence of the approximate solution computed via the ENO-LS algorithm given an initial oscillatory solution which has been evaluated from one of the standard centered fluxes (2), (3), (4); or from one of the "entropy conserving" fluxes (5) or (6). As test problem, we study first the numerical evolution of Burgers' equation (1) in one space dimension:

$$U_t + \left(\frac{U^2}{2}\right)_x = 0,$$

with initial condition $U(x, 0) = \sin 2\pi x$, in the domain $[0, 1]$, and extend the solution by periodicity outside 0 and 1. A shock wave develops at $t = \frac{2}{\pi}$ at the point $x = \frac{1}{4}$.

We first display in figure (2) the solution using 100 cells and a CFL condition of $\frac{1}{2}$ for a second order centered difference (CD) scheme using the flux (2), and then correct that solution by applying 7 iterations of the ENO-LS algorithm. Note first that the exact solution is perfectly recovered with exact location of the singularity because of the use of primitive variables in algorithm 2.1; second, these results are obtained for the set of coefficients $(p, q) = (13, 2)$, which implies that the corrected solution is only second order accurate in smooth transition areas; third, the upper right plot displays the corrected solution when both (3,2)CD and (3,2)ENO schemes are sequentially used. In that case, we correct by 5 ENO iterations after every 25 CD steps. This method for filtering oscillations is also very efficient for one dimensional problems (refer to table 1); however, many more ENO iterations are needed in two dimensions. Basically, 10 ENO iterations are needed after every 10 CD steps in order to recover an accurate solution for the two dimensional Burgers' example 4.3, which is a quite expensive technique compared to the overall cost induced by the ENO-LS method.

We shall discuss the results obtained with larger values of q in section 4 in which we investigate numerical order of accuracy of the filtered solutions from the ENO-LS algorithm. Also, a similar study is investigated as the number of evaluation points p increases.

In figure (3), we plot the numerical solution before and after the filter steps when $n = 1000$. Again, convergence to the physical solution is reached within 10 Gauss Seidel iterations. Note that this number increases by a factor of nearly 2 when Jacobi iterations are used.

In figures (4) and (5), we show the filtered solutions which were initially computed using the standard (3,4)CD and (3,6)CD schemes (fluxes (3) and 4)), respectively. Note that the corrected solution is well reconstructed in smooth regions while a smearing of the shock over about 10 cells is observed. This

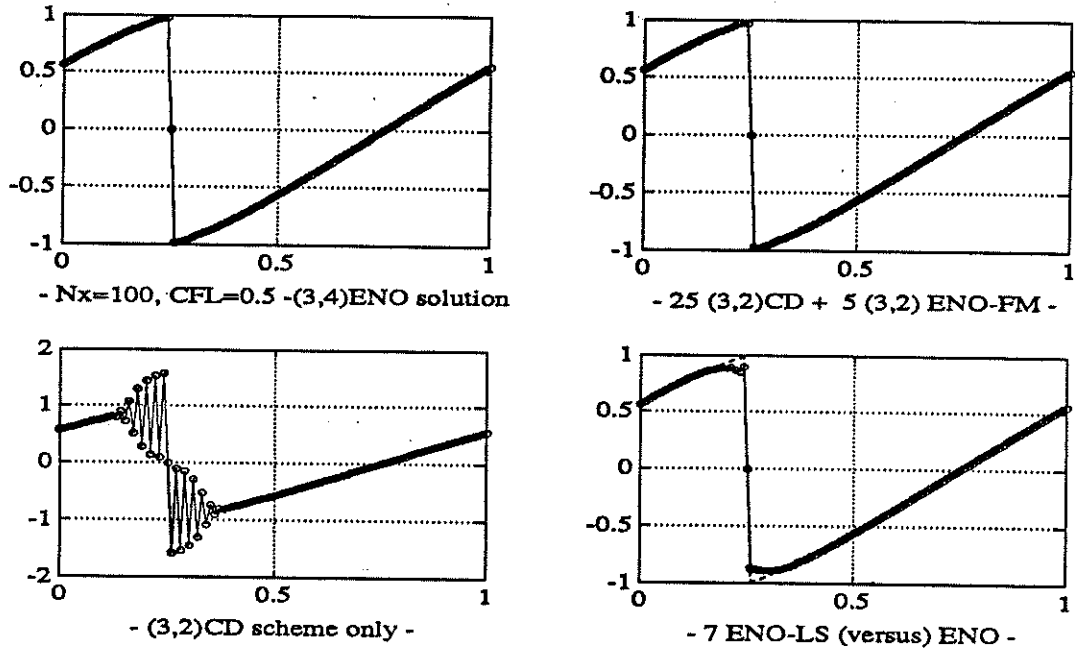


Figure 2: (3,2)CD scheme + ENO-LS method - 100 cells.

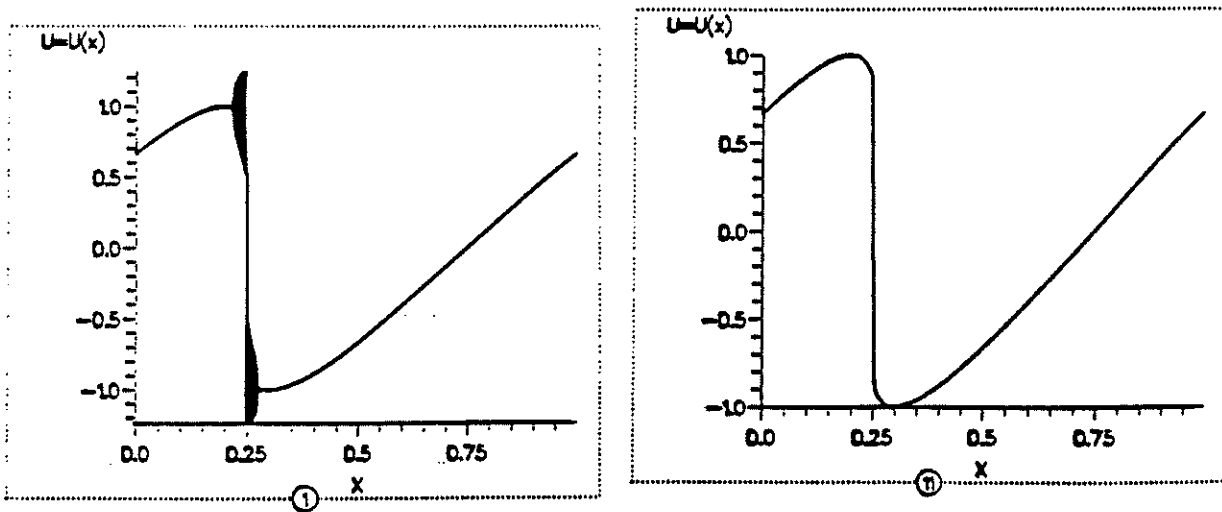


Figure 3: (3,2)CD scheme + ENO-LS method - 1000 cells.

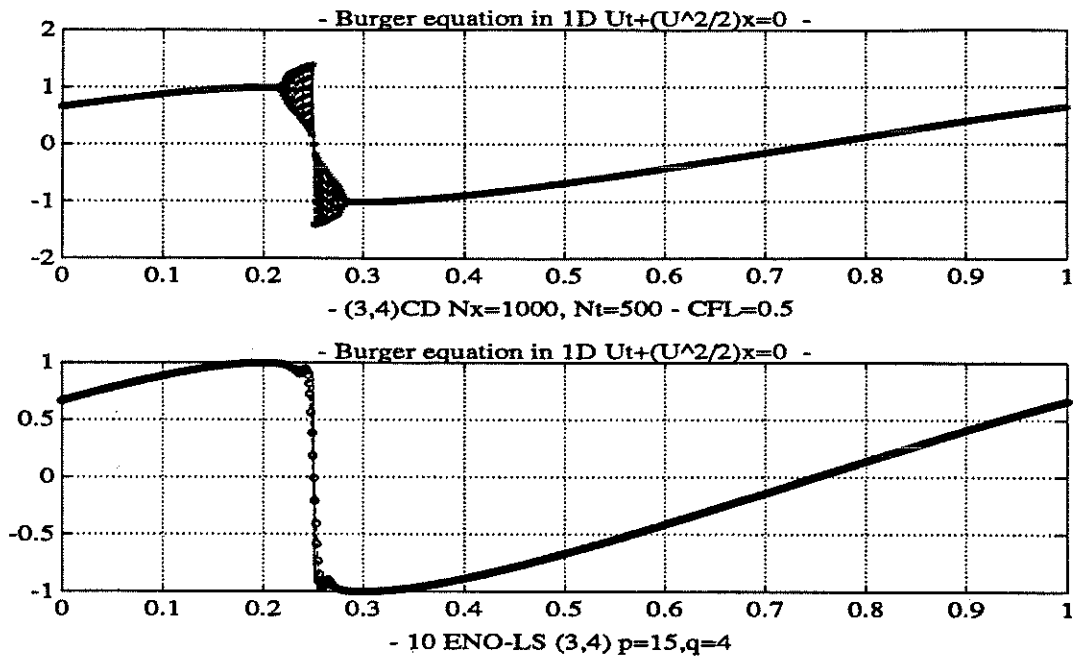


Figure 4: (3,4)CD scheme + 10 ENO-LS method - 1000 cells.

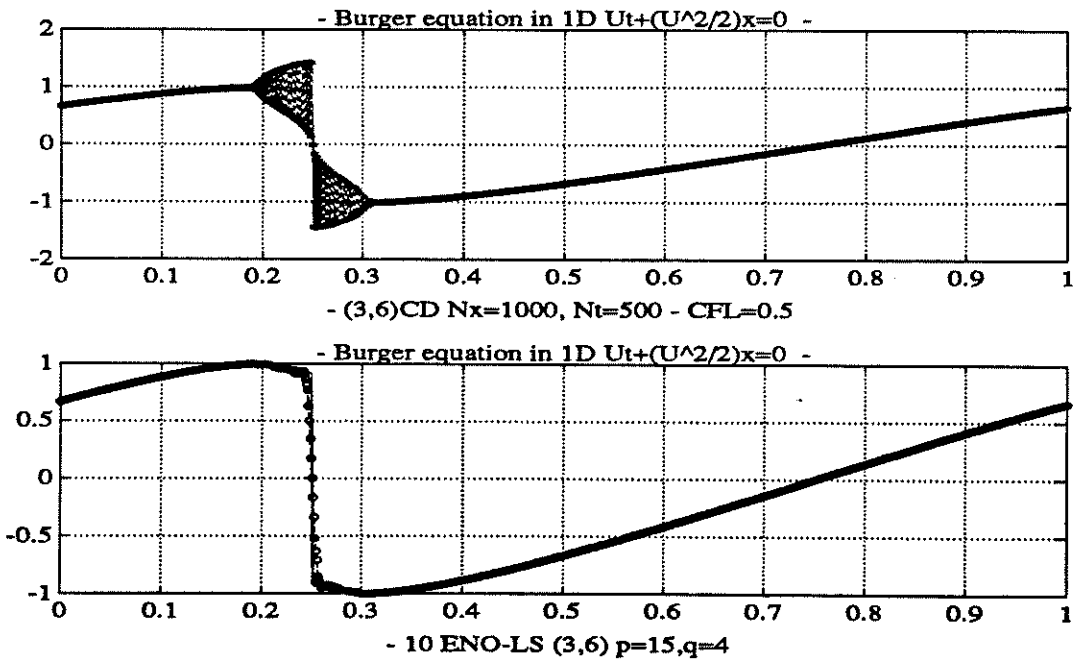


Figure 5: (3,6)CD scheme + 10 ENO-LS method - 1000 cells.

smearing can be primarily explained from the high (fourth) degree least squares polynomials ($q = 4$) used in those experiments.

Our second numerical test is devoted to showing that the filtered solution computed via the numerical flux (5):

$$F_{j+\frac{1}{2}} = \frac{1}{6}(U_{j+1}^2 + U_j U_{j+1} + U_j^2).$$

does not converge to the physical solution. It was observed in [6] that this scheme preserves mass and entropy (which in this case is taken to be $\frac{1}{2}U^2$). It was also shown there that the numerical solution does not converge to a weak solution of the original problem. Basically, as the mesh size tends to zero, some spurious oscillations are still visible near the shock and cannot be removed. We ran the last two previous experiments but used the approximate flux (5). We plot the numerical results in the figures (6) and (7) when $n = 100$ and $n = 1000$, respectively. For 100 cells (figure (6)), the oscillations near the shock are all smoothed out; however, the location of the shock is smeared over about 10 cells. This results were obtained for the same set of coefficients $(p, q) = (13, 2)$. Indeed, the solution is not well reconstructed. As the number of cells increases, the filtered solution is well reconstructed except near the shock. Numerical oscillations are still visible and cannot be removed. Note that the results visualized in figure (7) are given after 16 Gauss Seidel iterations which is much more than we needed when standard centered differences are taken.

Our final Burgers' equation test deals with a similar convergence failure property to the correct physical solution when the flux approximation (6) is implemented. The numerical flux is:

$$F_{j+\frac{1}{2}} = \frac{1}{2}U_j U_{j+1}.$$

This scheme again conserves both mass and entropy (this time the entropy is taken to be $\log U_j$), and it was shown in [3] that the numerical solution does not converge to a weak limit of the original problem as the stepsize Δx tends to zero. In the numerical experiments, we noticed that the amplitude of the oscillations grew very fast and was rapidly becoming unbounded. The results are plotted in the figures (8) and (9) for $n = 100$ and $n = 1000$, respectively. Note that, in the case $n = 100$, the filtered solution is again not very well reconstructed, and for 1000 cells, some oscillations are still visible after 20 ENO-LS iterations on the right of the shock. Moreover these oscillations could not be removed, even after many additional filtering iterations.

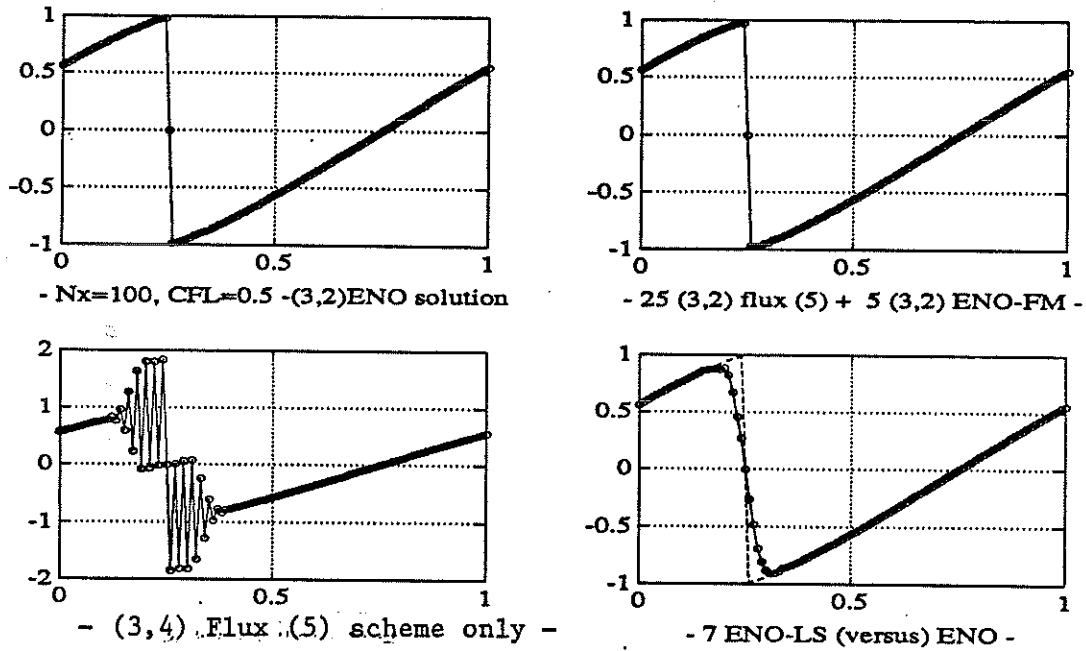


Figure 6: Flux (5) + ENO-LS method - 100 cells.

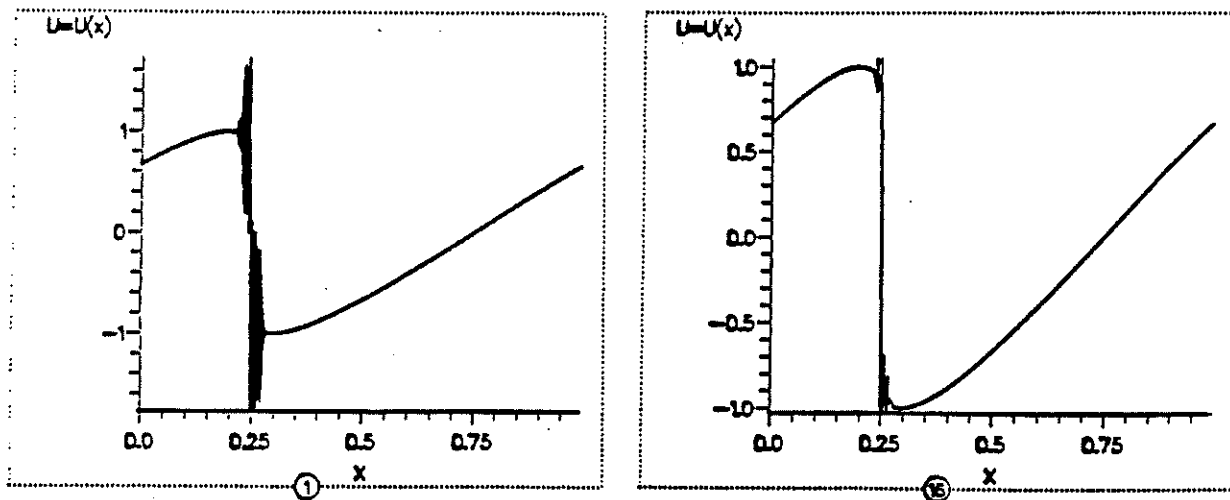


Figure 7: Flux (5) + ENO-LS method - 1000 cells.

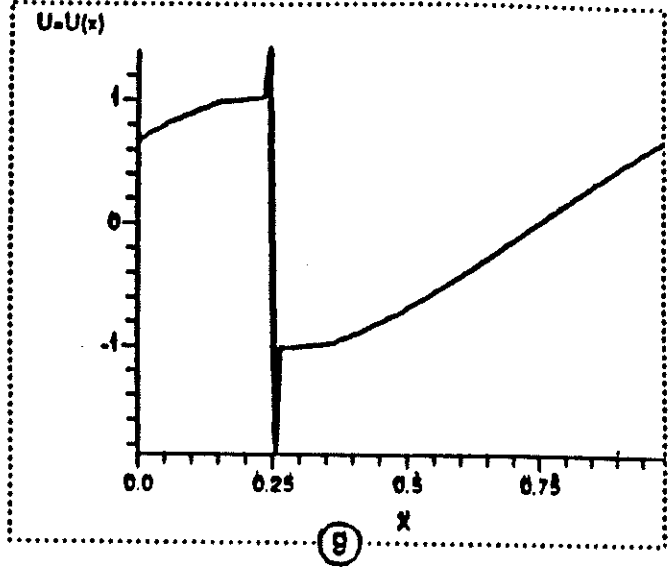
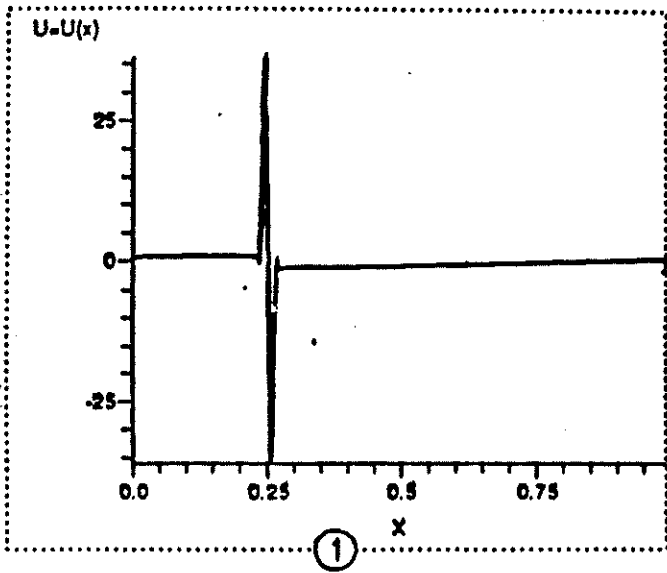


Figure 8: Flux (6) + ENO-LS method - 100 cells.

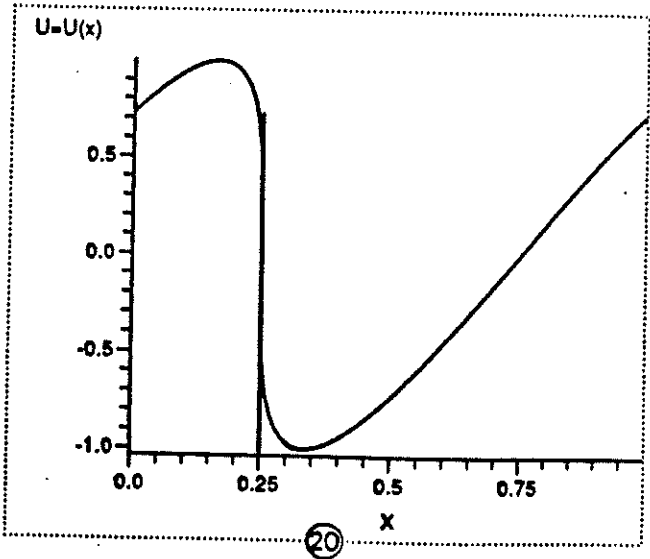
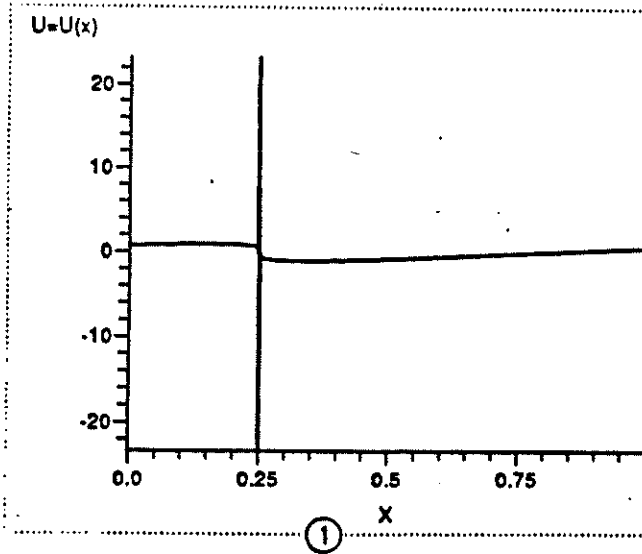


Figure 9: Flux (6) + ENO-LS method - 1000 cells.

Thus our techniques have been observed to construct converging sequences of filtered numerical solutions towards the expected solution given initial oscillatory data that has been computed from a central difference flux approximation. Our hope is now to show that our method is not only robust but is also fast. This is the topic of last section.

4 Time Efficiency of ENO-LS Algorithms:

In this section, we want to test the ENO-LS filtering method for several test problems involving nonlinear hyperbolic systems of conservation laws. We will focus our attention on comparing precisely the CPU times of the ENO-LS method versus more classical and filtered methods. Among them, we will consider the straightforward central difference (CD) method, the expensive ENO (Essentially Nonoscillatory) technique [7, 8], and our FM scheme (Filtering Method) [5]. We will run three examples for 1D and 2D Burgers' equation, and for Eulers' equations of compressible gas dynamics.

4.1 1D Burgers':

We first compare the time efficiency of several numerical schemes for the 1D Burgers' equation (1). In table (1), we show the average computational time per iteration for 100 mesh points for the CD, ENO, and FM schemes with several correction angles (see [5]), and for various combinations, performing alternatively some centered difference and some ENO steps. The notation CD+ENO (40,5,20) simply means that the calculation starts with 40 CD steps, followed by 5 full ENO iterations, and back to the centered scheme; the last 20 steps of the calculations are finally performed by the ENO method in order to recover an acceptable solution. Note that all coefficients displayed in this table are tuned so that the numerical solution is high order accurate in smooth regions and no spurious oscillations are detected.

The contents of this table need a few comments. First of all, the fastest algorithm is the one based on central differences. This is indeed not surprising since only one Fortran instruction is needed in the coding of the approximate flux. Second, postprocessing a numerical solution computed from the CD scheme by an ENO method can be very efficient for lower orders. For a second order method, only one ENO iteration after every 40 CD steps has to be implemented in order to reduce sufficiently the amplitude of oscillations. Note however that for the fourth order method, 7 ENO iterations were needed every after only 20 CD steps. In fact, if these spurious oscillations are not regularly cut off, the final ENO iterations may not recover the correct numerical solution. Third, the ENO-LS filtering method is the most costly

Type of Scheme	Order of Accuracy	CPU time $\times 10^{-2}$	# of corrections, comments
CD	(3,2)	0.56	CD = Centered Differences
CD	(3,4)	0.72	-
ENO-RF	(3,2)	1.11	RF = Roe Fix, Entropy fix
ENO-RF	(3,4)	1.68	-
FM	(3,2)	0.76	FM = Filtering Method, 3% Corrections
FM	(3,4)	1.07	4% Corrections
FM	(3,2)	1.40	100% Corrections = full ENO
FM	(3,4)	2.04	100% Corrections
CD+ENO (40,5,20)	(3,2)	0.74	40CD, 5ENO, 20 final ENO
CD+ENO (40,1,10)	(3,2)	0.58	40CD, 1ENO, 10 final ENO
CD+ENO (20,7,10)	(3,4)	0.99	20CD, 7ENO, 10 final ENO
ENO-LS (7,2)	(2)	1.84	$(p, q) = (7, 2)$, LU Inversion
ENO-LS (7,2)	(2)	0.65	Orthogonal polynomials
ENO-LS (7,3)	(3)	2.5	LU Inversion
ENO-LS (7,3)	(3)	0.89	Orthogonal polynomials

Table 1: CPU times of CD, ENO, FM and ENO-LS methods.

Type of Scheme	(q,p)	# of iterations	L^1 and L^∞ orders.
CD (3,2)	(2,5)	12	1.81 and 1.63
CD (3,2)	(2,9)	9	1.75 and 1.50
CD (3,2)	(2,13)	7	2.19 and 1.9
CD (3,4)	(3,5)	15	1.59 and 1.40
CD (3,4)	(3,9)	12	2.41 and 2.33
CD (3,4)	(3,13)	7	2.80 and 2.79

Table 2: Local L^1 and L^∞ order of accuracy.

when full LU inversion of the $C^{j+\frac{1}{2}}$ matrix is performed. However, when orthogonal basis functions are introduced, the ENO-LS method is competitive with respect to the fast CD scheme. Note moreover that ENO-LS correction steps have to be performed at a few times only. Finally, after running many experiments, we noticed that if the ratio $\frac{p}{q}$ becomes too large, then shocks have a tendency to spread over a large number of cells. Again, there is a compromise that needs to be reached for fast convergence; this depends on the large value of the ratio $\frac{p}{q}$, and the approximation of the shape near shocks for which p needs to be slightly smaller. Note that in most of experiments, the ratio $\frac{p}{q} \in [4, 6]$ was optimal.

Now, we want to measure the order of accuracy of the ENO-LS solution. To do so, we measure from computations the L^1 and L^∞ errors in the slabs $[0.10, 0.24]$ and $[0.26, 0.30]$. Numerical orders are shown in table 2.

Several comments about these results are now discussed. First of all, as the number of evaluation points increases, the better the quality of the results and the faster convergence is reached. Indeed, we have to pay the price of higher computations which are required to construct the coefficients involved in the $C^{j+\frac{1}{2}}$ matrix and in the vector $F^{j+\frac{1}{2}}$. On the other hand, faster calculations can be obtained provided

that the values of p and q differ only slightly, i.e $p = q + l$, $l = 1, 2, 3, \dots$, but local accuracy becomes obviously poor. Again, the optimal ratio seems to belong to the interval [4, 6].

4.2 Example 2. Euler Equations of Gas Dynamics:

The second test problem is devoted to the Euler equations of gas dynamics in one space dimension. We consider the initial condition given in example 8 of [8], that is:

$$\begin{aligned} \rho &= 3.857143, q = 2.629369, p = 10.3333333 \text{ when } x < -4 \\ \rho &= 1 + \varepsilon \sin 5x, q = 0, p = 1. \text{ when } x \geq -4 \end{aligned}$$

When $\varepsilon = 0$, a pure Mach 3 shock is moving to the right from the initial discontinuity $x = 4$. When $\varepsilon = 0.2$, we not only have a Mach 3 shock propagating to the right, but have as well a succession of weaker rarefaction and shock waves propagating to the left. Numerical results for the ENO and FM methods of high order of accuracy can be found in [8] and in [5], respectively. We ran the same problem using 240 CD iterations and then plotted the results in figure (10). Next, we correct this highly oscillatory numerical solution by performing 7 ENO-LS correction steps. In this experiment, we use $(p, q) = (15, 3)$. The filtered solution is visualized in figure (11). Note that the pressure, velocity, and entropy are quite well reconstructed, whereas the density is not perfectly recovered near the strong shock for which some physical oscillations should remain (refer to [10, 11]). However, we should note the remarkable improvement obtained from the solutions displayed in figures (10) and (11).

In figures (12) and (13), we use in sequence 50 CD steps and only one ENO iteration for second and fourth order methods before correcting the final oscillatory solution. The oscillatory solution is now postprocessed by 3 ENO-LS iterations. The filtered numerical results now look fairly similar to those shown in [8, 5]

4.3 Example 3. 2D Burgers' Equation.

The last example is devoted to the two dimensional Burgers' equation to be solved in the square domain $[-1, 1] \times [-1, 1]$ with initial condition $U_0(x, y) = \sin 2\pi r$, where $r = \sqrt{x^2 + y^2}$, for $r \leq \frac{1}{4}$, and $U_0(x, y) = 0$ outside the disc $r = \frac{1}{4}$. In figure (14), we visualize the solutions obtained by the (3,2)CD scheme, followed by 4 iterations of ENO-LS correction steps. In that experiment, we use $(p_x, p_y, q) = (6, 6, 2)$, so that the local rectangle in which the two dimensional ENO-LS stencil of points is taken contains 36

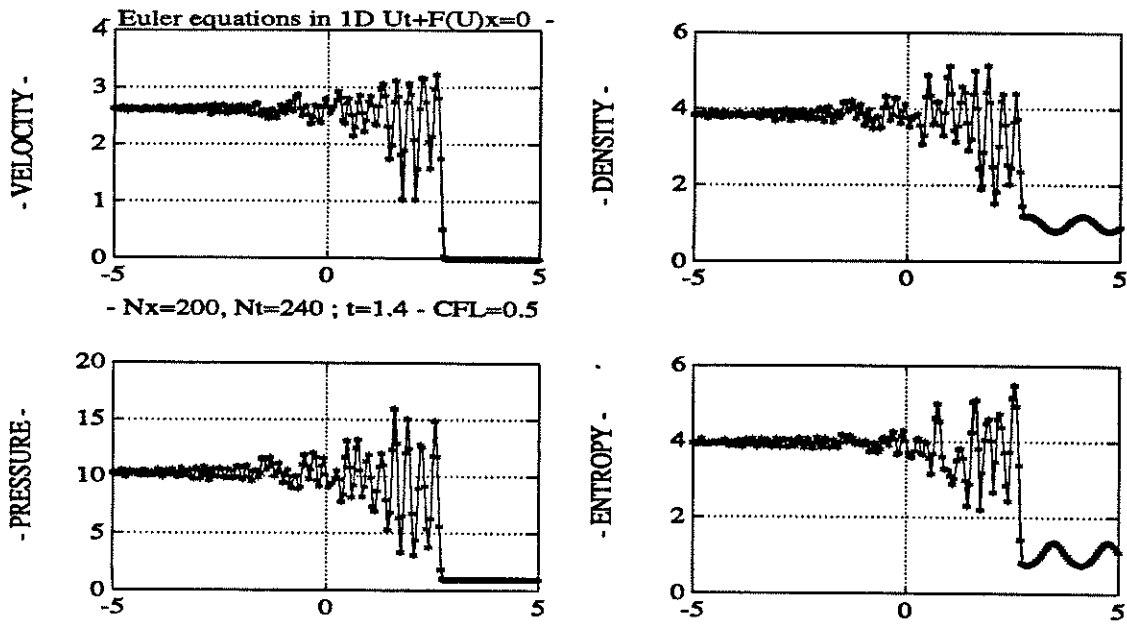


Figure 10: (3,2)CD scheme.

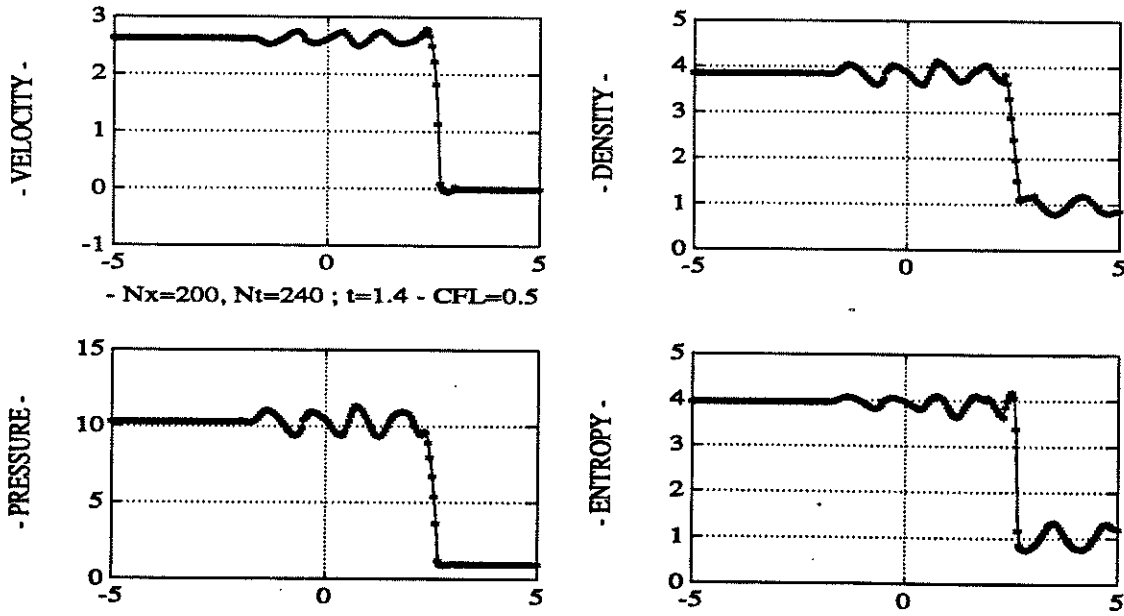
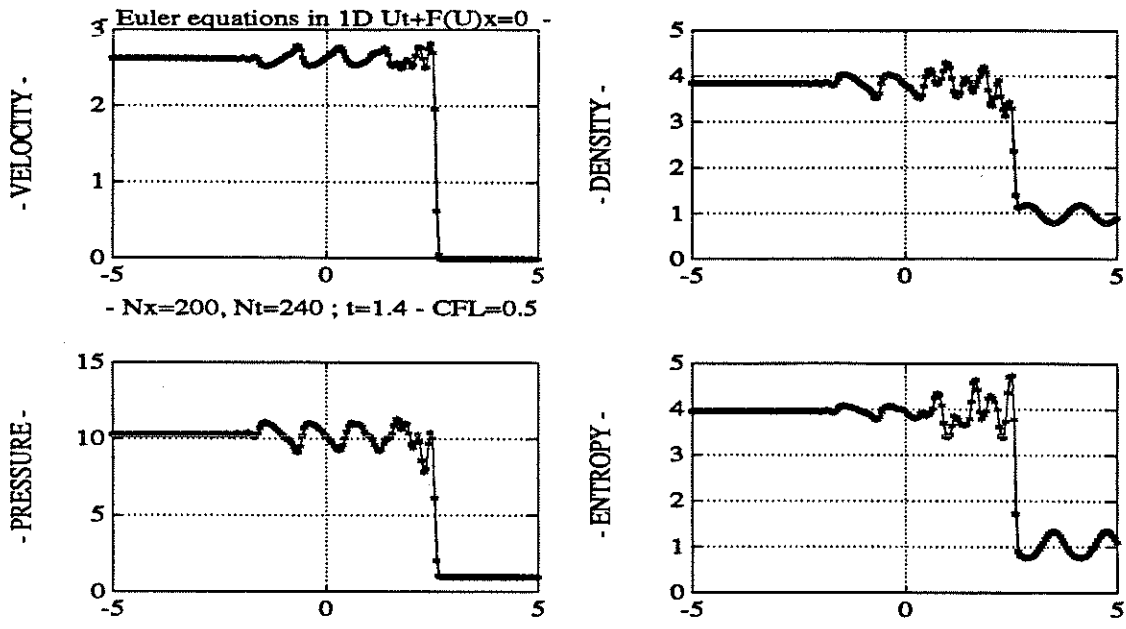
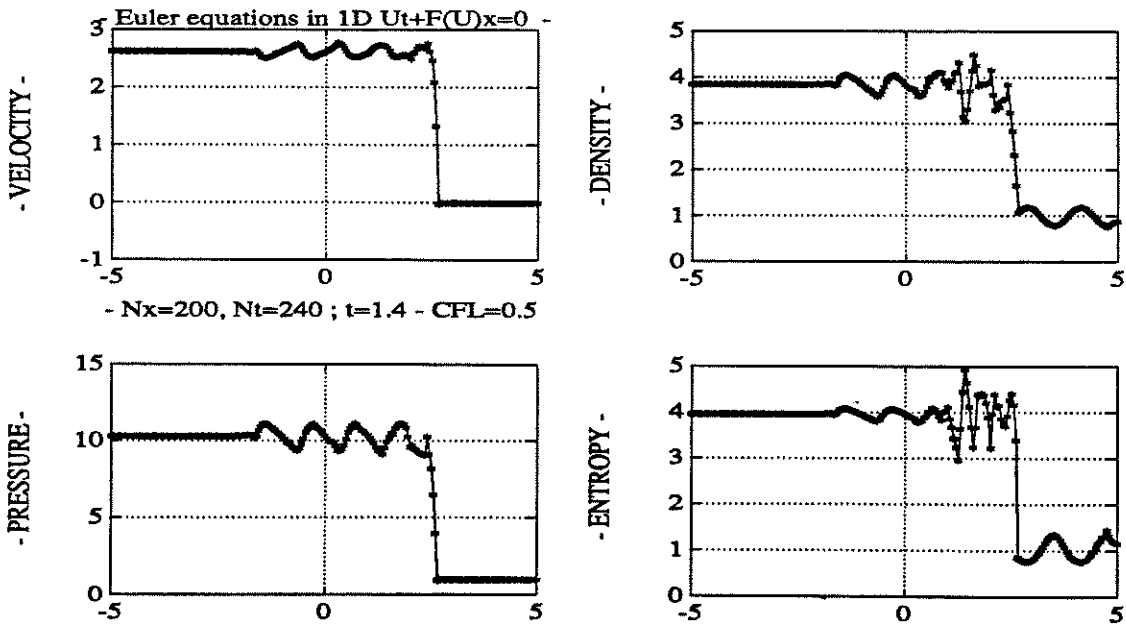


Figure 11: (3,2)CD scheme + 7 ENO-LS method $(p,q)=(13,3)$.



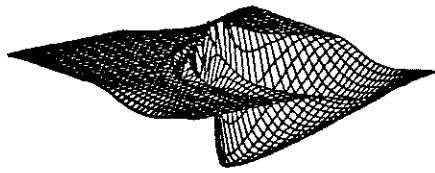
- (3,2)CD+ENO (50,1) + 5 final (ENO-LS) (p,q)=(13,3) -

Figure 12: (3,2)(CD,ENO) schemes + 3 ENO-LS method.

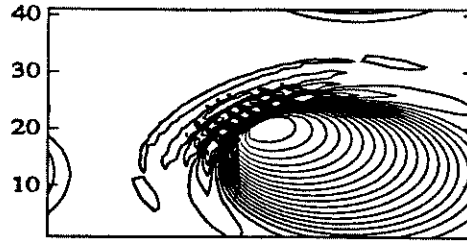


- (3,4)CD+ENO (50,1) + 3 final (ENO-LS) (p,q)=(13,4) -

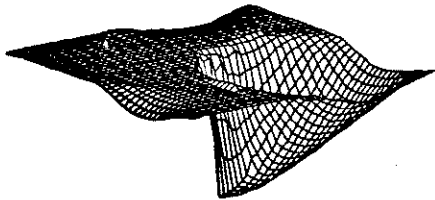
Figure 13: (3,4)(CD,ENO) schemes + 3 ENO-LS method.



- (3,4) CD on Burgers' equation -



- Nx=Ny=40, Nt=100 -



- (3,2)CD, (ENO-LS) Jacobi steps -

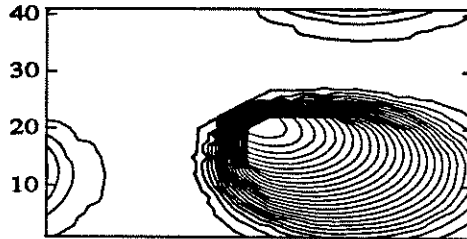


Figure 14: (3,2)CD scheme + 2D ENO-LS method.

mesh points. Note that a speed up factor of nearly 1.5 is obtained by using the two dimensional ENO-LS method instead of the one dimensional splitting version. In this numerical example, the amplitude of the numerical oscillations which propagate radially away from the initial shock location was approximately $\frac{1}{3}$ of the strength of the shock, yet the ENO-LS method recovered the nonoscillatory accurate solution quite well .

References

- [1] Engquist B., P. Lotstedt, B. Sjogreen, Math. Comput., 52, 509 (1989).
- [2] Harten A., B. Engquist, S. Osher, S. Chakravarthy, J. Comput. Phys., 71, 231 (1987).
- [3] Goodman, J. and P.D. Lax, Comm. Pure Appl. Math., 41, 591 (1988).
- [4] Hou T., P.D Lax, Comm. Pure Appl. Math., 44, 1 (1991).
- [5] Lafon F., S. Osher, "High order filtering methods for approximating hyperbolic systems of conservation laws", ICASE Report, 90-25, 1990 J. Comput. Phys. to appear, (1992).
- [6] Levermore D., J. Liu, to appear.
- [7] Shu C-W, S. Osher, J. Comput. Phys., 77, 439 (1988).
- [8] Shu C-W, S. Osher, J. Comput. Phys., 83, 32 (1989).
- [9] Von Neumann J., R.D. Richtmyer, J. Appl. Phys., 21, 380 (1950).
- [10] Zang T., M. Hussaini, D. Bushnell, AIAA J., 22, 13 (1984).
- [11] Zang T., M. Kopriva, M. Hussaini, "Pseudospectral Calculation of Shock Turbulence Interactions", ICASE Report 83-14, 1983.

