AIAA JOURNAL
Vol. 36, No. 8, August 1998

# Essentially Nonoscillatory Euler Solutions on Unstructured Meshes Using Extrapolation

D. Stanescu* and W. G. Habashi†
*Concordia University, Montreal, Quebec H3G 1M8, Canada*

**An extension of the essentially nonoscillatory formulation via extrapolation on unstructured meshes to systems of conservation laws is presented. Higher-order accuracy is obtained using a piecewise polynomial reconstruction of the solution from its cell averages. To avoid spurious oscillations near discontinuities and maintain accuracy in smooth regions, the reconstruction procedure selects an adaptive stencil for each cell. The reconstruction via extrapolation allows the selection in one step of all of the cells in the required stencil for a given order. This leads to important savings in computer time and makes the essentially nonoscillatory schemes on unstructured meshes suitable for practical applications. The problems encountered when using triangular meshes, as well as possible ways to overcome them, are discussed. Numerical results and comparison of the efficiency against other schemes are also presented.**

## Introduction

THE numerical simulation of transient wavelike phenomena places specific demands on computational algorithms. In areas such as aeroacoustics, for example, it must not only allow the propagation of low-amplitude waves but also enable the representation of discontinuities present in the flowfield without generating numerical noise. The essentially nonoscillatory (ENO) schemes, developed by Harten et al.[1] in one space dimension, are valid candidates for such problems because, unlike the total-variation diminishing (TVD) schemes that revert to first order near discontinuities, they can be formally extended to an arbitrary order of accuracy.

The main idea underlying the construction of ENO schemes is the use of a reconstruction procedure that recovers high-order-accurate approximations to the pointwise values from cell averages. The reconstruction chooses, for each cell of the mesh, a stencil of neighboring cells over which the solution is smoothest in a certain sense and uses it to build a polynomial that approximates the solution over that cell. It is this adaptive stencil that enables the higher-order-accurate flux computation over smooth flow regions, while avoiding the Gibbs phenomenon near discontinuities.

The first implementation of the ENO concept for two-dimensional scalar equations is also due to Harten.[2] For the Euler equations, Casper and Atkins[3] present a two-dimensional finite volume formulation on structured quadrilateral grids. They use a two-dimensional reconstruction that is built as a tensor product of one-dimensional ones, a procedure that cannot be extended to unstructured grids. Early attempts at treating unstructured meshes using cell-centered control volumes have been made by Harten and Chakravarthy,[4] as well as Vankeirsbilck and Deconinck.[5] Although very general and robust, these algorithms are expensive because the cells in the stencil are selected one at a time, while testing all possible stencils for the minimization of solution derivatives. A further step has been taken by Abgrall,[6,7] who presents an implementation of an ENO scheme using node-centered control volumes and who also gives bounds on the reconstruction error. To remain efficient, his third-order scheme limits the number of potential stencils that have to be tested.[6,7]

A recent interpretation of the reconstruction procedure on uniform grids as being based on extrapolation, introduced by Suresh and Jorgenson,[8] offers new perspectives for the ENO reconstruction on unstructured meshes. In this method, given the stencil already chosen for a certain order $r$ of accuracy, all of the cells in the stencil for order $r+1$ can be selected at once. Results for the reconstruction of a piecewise smooth function given by Suresh and Jorgenson[8] show that the cost of the reconstruction algorithm is substantially reduced. Thus, this method seems to be a good choice for the design of ENO schemes for the Euler equations on unstructured meshes.

The following section presents the numerical formulation of a higher-order ENO scheme using cell-centered control volumes for the Euler equations. The reconstruction via an extrapolation procedure is then described. Numerical results using the present scheme are then given, and our conclusions end the paper.

## Finite Volume Formulation

The two-dimensional system of unsteady Euler equations can be written in conservation form as

$$\frac{\partial \boldsymbol{w}}{\partial t}+\frac{\partial \boldsymbol{f}}{\partial x}+\frac{\partial \boldsymbol{g}}{\partial y}=0, \qquad (x,y)\in\mathcal{D}, \qquad t>0 \tag{1}$$

subject to given initial conditions

$$\boldsymbol{w}(x,y,0)=\boldsymbol{w}_0(x,y), \qquad (x,y)\in\mathcal{D} \tag{2}$$

and with specified boundary conditions on $\partial\mathcal{D}$, the boundary of $\mathcal{D}$. The state vector $\boldsymbol{w}$ and the flux vector components $\boldsymbol{f}$ and $\boldsymbol{g}$ are given by

$$\boldsymbol{w}=\begin{bmatrix}\rho\\ \rho u\\ \rho v\\ \rho E\end{bmatrix}, \qquad \boldsymbol{f}=\begin{bmatrix}\rho u\\ \rho u^2+p\\ \rho uv\\ (\rho E+p)u\end{bmatrix}, \qquad \boldsymbol{g}=\begin{bmatrix}\rho v\\ \rho uv\\ \rho v^2+p\\ (\rho E+p)v\end{bmatrix} \tag{3}$$

where $\rho$ is the density, $u$ and $v$ are the two components of the velocity, $E$ is the specific total energy, and $p$ is the pressure. The equation of state

$$p=\rho(\gamma-1)[E-(u^2+v^2)/2] \tag{4}$$

where $\gamma$ is the ratio of the specific heats, relates the pressure to the other variables and closes the system.

To obtain weak solutions to Eq. (1), the computational domain $\mathcal{D}$ is divided into cells $\mathcal{C}_i$ that do not change with time, and the equation is integrated over each cell to give

$$\frac{\partial \bar{\boldsymbol{w}}_i}{\partial t}=-\frac{1}{\mathcal{A}_i}\int_{\partial\mathcal{C}_i}F_n[\boldsymbol{w}(x,y,t)]\,\mathrm{d}s \tag{5}$$

where $\mathcal{A}_i$ is the area of $\mathcal{C}_i$, $F_n=\boldsymbol{f}n_x+\boldsymbol{g}n_y$ is the projection of the flux on the normal to the cell boundary, and

$$\bar{\boldsymbol{w}}_i=\frac{1}{\mathcal{A}_i}\int_{\mathcal{C}_i}\boldsymbol{w}(x,y,t)\,\mathrm{d}A$$

Presented as Paper 97-0539 at the AIAA 35th Aerospace Sciences Meeting, Reno, NV, Jan. 6–9, 1997; received Feb. 13, 1997; revision received April 15, 1998; accepted for publication April 15, 1998. 

*Ph.D. Candidate, Computational Fluid Dynamics Laboratory, Department of Mechanical Engineering. Student Member AIAA.

†Professor and Director, Computational Fluid Dynamics Laboratory. Associate Fellow AIAA.

is the average of $\boldsymbol{w}$ over the cell $\mathcal{C}_i$. Considering a series of discrete time levels $t_1 < t_2 \cdots < t_n \cdots$ and denoting $\bar{\boldsymbol{w}}_i(t_n)$ by $\bar{\boldsymbol{w}}_i^n$, we write Eq. (5) to define a semidiscrete formulation in operator form:

$$\frac{\partial \bar{\boldsymbol{w}}_i^n}{\partial t} = (\mathcal{L}\boldsymbol{w}(t))_i \tag{6}$$

where $(\mathcal{L}\boldsymbol{w}(t))$ involves only spatial operations. Using the method of lines, we consider that along each $t = \text{const}$ line the right-hand side in Eq. (6) is constant; thus, it becomes an ordinary differential equation that can be integrated by appropriate numerical methods. Particularly attractive for the purpose of this study are the TVD Runge–Kutta methods devised by Shu and Osher.[9] The methods have the property that the discrete temporal operator does not increase the total variation of the solution, and they have been used to obtain all of the numerical results that follow. These methods can be written in the form

$$\bar{\boldsymbol{w}}_i^{(l)} = \sum_{m=0}^{l-1} \left[\alpha_{lm}\bar{\boldsymbol{w}}_i^{(m)} + \beta_{lm}\Delta t(\mathcal{L}\boldsymbol{w}^m)_i\right], \qquad l = 1, 2, \ldots, p$$
$$\bar{\boldsymbol{w}}_i^{(0)} = \bar{\boldsymbol{w}}_i^n, \qquad \bar{\boldsymbol{w}}_i^{(p)} = \bar{\boldsymbol{w}}_i^{n+1} \tag{7}$$

where the values of the coefficients $\alpha$ and $\beta$ and the integer $p$ determine the TVD properties and the order of accuracy of the method.

Thus, having chosen a time integration method (7) that achieves the desired accuracy in time, we turn our attention to the spatial discretization. To obtain an approximation of the exact operator $(\mathcal{L}w(t))$, let it be written as

$$-\frac{1}{\mathcal{A}_i}\int_{\partial\mathcal{C}_i} F_n[\boldsymbol{w}(x, y, t)]\,\mathrm{d}s = -\frac{1}{\mathcal{A}_i}\sum_k \int_{\partial\mathcal{C}_i^k} F_n[\boldsymbol{w}(x, y, t)]\,\mathrm{d}s \tag{8}$$

where the integral has been broken up into a sum over the edges $\partial\mathcal{C}_i^k$ of the cell $\mathcal{C}_i$. On each edge, the integral can be approximated to a desired order of accuracy $r$ using a Gauss quadrature

$$\int_{\partial\mathcal{C}_i^k} F_n[\boldsymbol{w}(x, y, t)]\,\mathrm{d}s = \sum_{g=1}^{G} W_g \tilde{F}_n[\boldsymbol{w}(x_g, y_g, t)] \tag{9}$$

where $W_g$ are the Gauss weights, with the number $G$ of Gauss points chosen to achieve the desired accuracy, i.e., $G = 1$ for $r = 1$ and 2 and $G = 2$ for $r = 3$ and 4. In Eq. (9), the numerical approximation to the flux $\tilde{F}_n$, which is defined as follows, has been introduced. Because the data represent averages of the state vector over the cells of the mesh, $\bar{\boldsymbol{w}}_i^n$, we need pointwise values on the edges that approximate $\boldsymbol{w}(x_g, y_g, t_n)$ to order $r$. We will use for this purpose the reconstruction polynomial representation of $\boldsymbol{w}$ over the cell $\mathcal{C}_i$, denoted by $R_i[\boldsymbol{w}(\cdot), r]$. The way this polynomial is obtained is described in the next section. Let $\mathcal{C}_*$ be the cell that has $\partial\mathcal{C}_i^k$ in common with $\mathcal{C}_i$. The flux is computed as

$$\tilde{F}_n[\boldsymbol{w}(x_g, y_g, t)] = \tilde{\mathcal{F}}_n^R\{R_i[\boldsymbol{w}(x_g, y_g, t), r], R_*[\boldsymbol{w}(x_g, y_g, t), r]\} \tag{10}$$

where $\tilde{\mathcal{F}}_n^R$ is a properly chosen Riemann solver.

As an alternative to this approximation, one can perform the integration of the reconstructed state vector for the two elements neighboring an edge and solve only one Riemann problem per edge using the average $\boldsymbol{w}$ values thus obtained. The cost of the Riemann solution being very low compared with that of the reconstruction, the advantage of this approach is, however, minimal.

## ENO Reconstruction

### Scalar Reconstruction

Let the coordinates of the centroid of cell $\mathcal{C}_i$ for which we want the reconstruction $R(\cdot)$ at time $t_n$ be denoted by $(x_0, y_0)$, and suppose that the function to be approximated is a scalar function. The extension to vector functions will be presented later. For an $r$th-order-of-accuracy scheme, the approximation polynomial is sought in the vector space $\mathbb{P}_m[X, Y]$ of polynomials of total degree $m = r - 1$:

$$R_i[\boldsymbol{w}(\cdot), r] = P(X, Y) = \sum_{l=0}^{m}\sum_{j+k=l} a_{jk}X^jY^k \tag{11}$$

It is obvious that, if one chooses as the basis of $\mathbb{P}_m$ monomials of the form $\{(x - x_0)^j(y - y_0)^k\}$, the coefficients $a_{jk}$ will be approximations to the derivatives of $\boldsymbol{w}$ at the centroid of the cell, scaled by the corresponding factor. Because the dimension of $\mathbb{P}_m$ is $M = (m + 1)(m + 2)/2$, we need a stencil of $M$ cells with indices $S_q^m$, $q = 1, 2, \ldots, M$, to build the interpolation polynomial. Requiring, to preserve conservation, that cell $\mathcal{C}_i$ be in the stencil and that the approximation polynomial preserve the average value of $\boldsymbol{w}$ over the stencil cells leads to the linear system

$$\sum_{l=0}^{m}\sum_{j+k=l} a_{jk}\langle X^jY^k\rangle_{S_q^m} = \bar{\boldsymbol{w}}_{S_q^m}^n \tag{12}$$

where

$$\langle X^jY^k\rangle_{S_q^m} = \frac{1}{\mathcal{A}_{S_q^m}}\int_{\mathcal{C}_{S_q^m}} X^jY^k\,\mathrm{d}A \tag{13}$$

The stencil will be called admissible if the associated matrix is nonsingular. In that case the solution of the preceding system yields the coefficients and, hence, the required polynomial.

We turn now to the selection of the cells in the stencil. For a first-order approximation ($m = 0$), the stencil will be the cell for which we perform the reconstruction itself, $S^0 = \{i\}$. For higher-order approximations, we build the stencils successively. We start from a stencil for a degree $m$ polynomial $S^m$ and a computed reconstruction obtained by solving (12) in the form

$$R_i[\boldsymbol{w}(\cdot), m + 1] = \sum_{l=0}^{m}\sum_{j+k=l} a_{jk}{}^m X^jY^k \tag{14}$$

Let the set of cells that have one edge in common with at least one cell in $S_q^m$ and do not belong to this stencil be denoted by $P^m$. (We will call this set the potential stencil.) For each cell $p \in P^m$, the difference between the average value of $\boldsymbol{w}$ and the average value of the reconstruction polynomial extrapolated to cell $p$ is given by

$$D_p = \bar{\boldsymbol{w}}_p - \frac{1}{\mathcal{A}_p}\int_{\mathcal{C}_p} R_i[\boldsymbol{w}(\cdot), m + 1]\,\mathrm{d}A \tag{15}$$

Denoting $h$ to be a convenient measure of the mesh size, $D_p$ will be of order $\mathcal{O}(h^{r+1})$ if the cell $p$ and all of the cells in the stencil $S^m$ lie in a smooth region. Otherwise, this difference will be $\mathcal{O}(h^1)$. To select the smoothest stencil, we compute $D_p$ for all cells $p \in P^m$ and add the $m + 3$ cells that have the smallest value to the stencil $S^m$ to obtain the stencil $S^{m+1}$. If this is not an admissible stencil, then the next best stencil is selected.

### State Vector Reconstruction

An extension of the described procedure to the state vector $\boldsymbol{w}$ of the Euler equations can be made by performing the reconstruction componentwise. In this approach, a stencil is selected for each of the four components of the state vector. An even faster method would be to select only one stencil for all of the components, based on one of the components, i.e., the density, or a derived variable, i.e., the Mach number. However, one has to consider that the pressure is not among the conserved variables. Recovery of the pressure field from polynomial approximations of the state vector can easily lead to nonphysical results because the pressure is a nonlinear function of the components and the pointwise values of the four components of the state vector do not obey the conservation law as do the cell averages. This remark has been made by Harten et al.[1] and Harten and Chakravarthy.[4]

A possible solution is to base the reconstruction on another set of variables that are smoother than the conserved ones. Such a set can be either the vector of primitive variables $\boldsymbol{U} = \{\rho, u, v, p\}^T$ or a conveniently defined set of characteristic variables. In the first case, the pressure is among the reconstructed variables, and because in two-dimensional flows the second option supposes an ad hoc choice of the direction of wave propagation, it is the method we used in this study. For $r \le 2$, the average values for $\boldsymbol{U}$ can be directly obtained from the average values of $\boldsymbol{w}$, which is not true for orders of accuracy $r \ge 3$. The computation of cell averages for the components of $\boldsymbol{U}$

can in this case be done by a local transformation (see Ref. 1 for a one-dimensional example). To reconstruct $\boldsymbol{w}$ in cell $\mathcal{C}_i$, we consider the locally defined transformation

$$\boldsymbol{w} = T\left(\bar{\boldsymbol{w}}_i^n\right)\boldsymbol{U}, \qquad \boldsymbol{U} = T^{-1}\left(\bar{\boldsymbol{w}}_i^n\right)\boldsymbol{w} \tag{16}$$

where the matrix $T$ is the Jacobian matrix

$$T = \frac{\partial \boldsymbol{w}}{\partial \boldsymbol{U}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ u & \rho & 0 & 0 \\ v & 0 & \rho & 0 \\ \dfrac{u^2+v^2}{2} & \rho u & \rho v & \dfrac{1}{\gamma-1} \end{bmatrix} \tag{17}$$

Because the transformation is linear, we have that

$$\bar{\boldsymbol{U}}_i^n = T^{-1}\left(\bar{\boldsymbol{w}}_i^n\right)\bar{\boldsymbol{w}}_i^n \tag{18}$$

The componentwise reconstruction is applied to the values $\bar{\boldsymbol{U}}_i^n$ defined by Eq. (18), and the inverse transformation is used to compute the polynomial coefficients for the conservative variables.

**Implementation Issues**

Because the number of neighbors that belong to the potential stencil $P^m$ is not large enough for $m = 0$ (there are only three edge neighbors for regular interior cells, and one needs to select two new cells in $S^{m+1}$), the method just described has to be modified in this case. To allow for a larger choice, the two cells in $S^{m+1}$ are selected one at a time, and after each selection the potential stencil $P^m$ is increased with the neighbors of the selected cell. Thus, the algorithm becomes identical with the Harten and Chakravarthy[4] one for second-order accuracy.

A main problem of the method is related to the condition number of the matrix associated with the linear system (12). Even for very isotropic meshes, there usually exist some selected stencils that, due to their spatial arrangement, lead to degenerate reconstructions. For these stencils, the condition number of the matrix is several orders of magnitude higher than for the ordinary stencils. Furthermore, the ratio cannot be changed by scaling or by the use of other basis functions or coordinate systems, although the magnitude of the condition number itself changes. To avoid the degenerate reconstructions obtained in such a case, we used the fact that the coefficients of the polynomial become very large and highly oscillatory in such a case. An oscillation norm, following Suresh and Jorgenson,[8] is set up as

$$O_m = \frac{1}{(m+1)(m+2)/2} \sum_{l=1}^{m} \sum_{j+k=l} |a_{jk}| \tag{19}$$

and the reconstruction was stopped when it increased with $m$. This is, however, a possible source for loss of accuracy. A more recent solution to this problem, using a generalization of the Newton form of the Lagrange interpolation polynomial to multiple dimensions, based on Mühlbach expansions, has been introduced by Abgrall et al.[10]

## Numerical Results

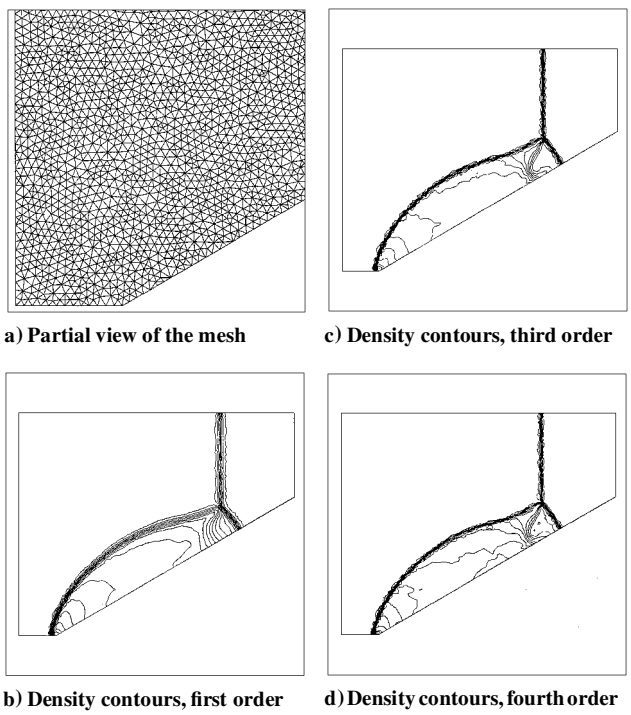
The preceding numerical scheme has been implemented in a computer code that offers the possibility to use spatial approximations of order $r = 1$–$4$. The Riemann solver used for all of the simulations was the exact solver of Godunov,[11] although other options are available. The order of the temporal approximation has been chosen accordingly, except for the fourth-order scheme, for which the same third-order time marching scheme as for $r = 3$ has been used. The time step has been computed as

$$\Delta t = C \cdot \min_i \left\{ \mathcal{A}_i \Big/ \left[ \sum_k \int_{\partial \mathcal{C}_i^k} (v_n + a)\, \mathrm{d}s \right] \right\} \tag{20}$$

where $v_n$ is the normal velocity, $a$ is the sound speed, and the value of the constant $C$ is given by the stability limit of the Runge–Kutta scheme. The code uses a dual data structure (based on both edges and cells) and could thus be fully parallelized. (The flux computation is done using the edge and the reconstruction using the cell data structure.) This section presents the results obtained for a series of test cases that are well documented in the literature.

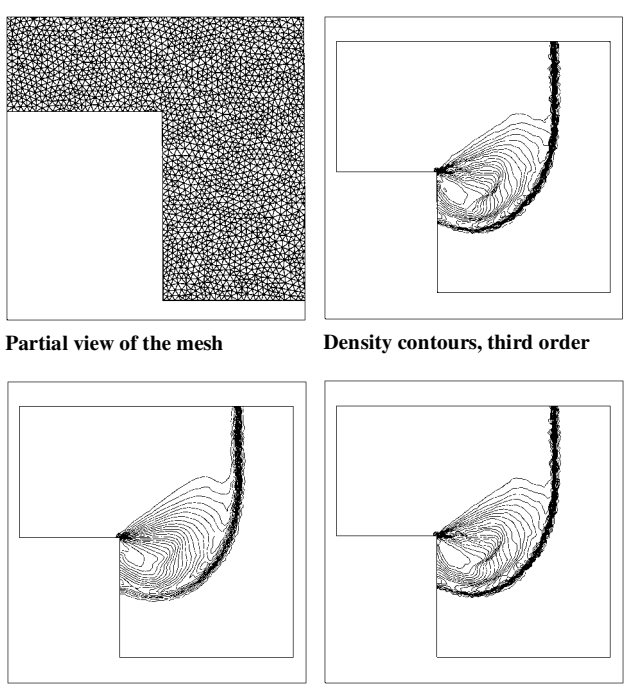
**Double Mach Reflection**

The first problem is the reflection of a shock wave ($M = 5.5$ in this case) from a 30-deg ramp. This gives rise to a self-similar solution that has been widely studied analytically and numerically (see Ref. 12 and references therein). In general, a very fine mesh is used to obtain a correct solution. A partial view of the mesh used in this case is shown in Fig. 1a; the full mesh has 7075 nodes and 13,825 cells. The shock is positioned 0.1 length units before the wedge at $t = 0$. Density contours obtained at $t = 0.18$ are given in Figs. 1b–1d. Figures 2 and 3 show shock diffraction and the forward-facing step, respectively. For comparison with Fig. 1, we also present results obtained with the method originally developed by Harten and Chakravarthy,[4] using a third-order approximation (Fig. 4). Figure 4 also shows the advantage of using the primitive



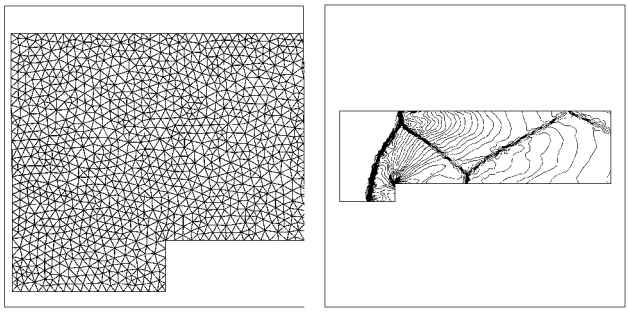
a) Partial view of the mesh
c) Density contours, third order
b) Density contours, first order
d) Density contours, fourth order

**Fig. 1 Double Mach reflection.**

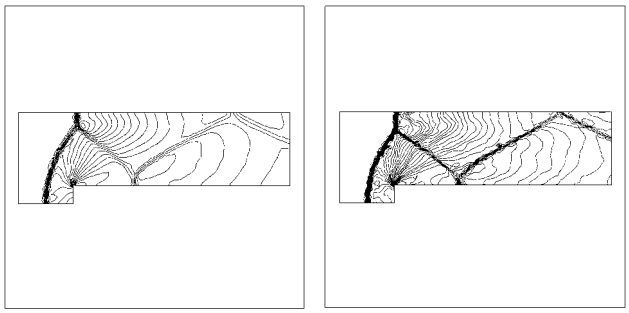


Partial view of the mesh
Density contours, third order
Density contours, first order
Density contours, fourth order

**Fig. 2 Shock diffraction.**

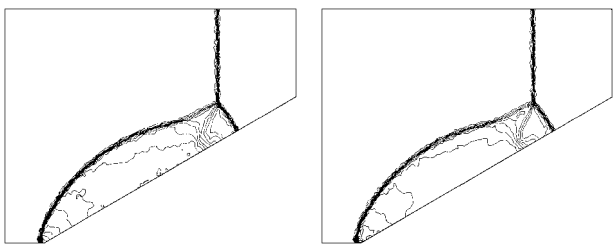


Partial view of the mesh | Density contours, second order



Density contours, first order | Density contours, third order

**Fig. 3 Forward-facing step.**



Reconstruction in conservative variables | Reconstruction in primitive variables

**Fig. 4 Double Mach reflection: results obtained by the third-order Harten–Chakravarthy[4] method.**

variables for the reconstruction procedure, the discontinuity emanating from the triple point being better resolved in that case.

#### Shock Diffraction

This problem represents the diffraction of a $M = 5.5$ shock wave around a 90-deg corner. The computational domain has a length of 1.1 on the $x$ axis; the computation is initialized with the shock at $x = 0.1$. The state at the right of the shock is the undisturbed flow state ($\rho = 1.4$, $p = 1.0$, and $u = v = 0$), whereas the left-hand state is the postshock state. Density contours at time $t = 0.14$ are displayed in Fig. 2, along with a partial view of the mesh.

#### Mach 3 Flow over a Forward-Facing Step

This is another test case used to test the accuracy of higher-order methods (for a complete description, see Ref. 12). Although the steady state (reached at $t = 12$) is not particularly interesting, the shock structure at earlier times is very complex. The results (Fig. 3) show this structure at $t = 4$, obtained on a rather coarse mesh (2698 nodes and 5158 elements). No special treatment has been applied at the corner of the step. It can be noticed that the position of the Mach stem on the upper wall and its length are more correctly computed by the third-order method. Also, the spurious Mach reflection on the lower wall is much less pronounced. However, the resolution of the contact discontinuity that propagates downstream of the upper wall Mach stem does not improve substantially through the use of the third-order approximation. Table 1 gives the relative CPU time of the present method compared with that of the cheapest method proposed by Harten and Chakravarthy[4] in which the value of the last derivative is used to compare the stencils. (This avoids solving the linear system when comparing stencils because only the forward elimination phase is needed to compute the last derivative.) The times include reconstruction, flux computation, and update of the cell averages. (The actual CPU time needed for one time step of the third-order present method on one processor of a SGI 4D/300 machine is 98 s.)

**Table 1 Relative CPU times for the forward step problem**

| CPU time | Order 3 | Order 4 |
|---|---|---|
| Present | 100 | 318.3 |
| Ref. 4 | 626.7 | 2956.9 |

As can be seen, the present fourth-order approximation is even cheaper than a third-order solution using the Harten and Chakravarthy[4] method, the cost of which becomes prohibitive for fourth and higher orders.

## Conclusions

A higher-order method for the Euler equations has been presented. The method has the capability to treat general mesh topologies and is shown to be very effective when compared with other ENO methods of the same accuracy. Whereas the issue of unstructured mesh methods cannot be considered solved, because of the problems touched upon herein and because more involved interpolation techniques must be used, the present method is a step forward toward feasible higher-order approximations on general meshes.

## Acknowledgments

The support received from the Natural Sciences and Engineering Research Council of Canada and the Fonds pour la Formation de Chercheurs et l'Aide à la Recherche of Quebec is gratefully acknowledged.

## References

[1] Harten, A., Engquist, B., Osher, S., and Chakravarthy, S., "Uniformly High Order Accurate Essentially Non-Oscillatory Schemes, III," *Journal of Computational Physics*, Vol. 71, No. 2, 1987, pp. 231–303.

[2] Harten, A., "Preliminary Results on the Extension of ENO Schemes to Two-Dimensional Problems," *Nonlinear Hyperbolic Problems: Proceedings of a Research Workshop Held in St. Etienne, January 13–17, 1986*, No. 1270, Lecture Notes in Mathematics, edited by C. Carasso, Springer–Verlag, Berlin, 1987, pp. 23–40.

[3] Casper, J., and Atkins, H. L., "A Finite-Volume High-Order ENO Scheme for Two-Dimensional Hyperbolic Systems," *Journal of Computational Physics*, Vol. 106, No. 1, 1993, pp. 62–76.

[4] Harten, A., and Chakravarthy, S., "Multi-Dimensional ENO Schemes for General Geometries," Inst. for Computer Applications in Science and Engineering, Rept. 91-76, Sept. 1991.

[5] Vankeirsbilck, P., and Deconinck, H., "Higher Order Upwind Finite Volume Schemes with ENO-Properties for General Unstructured Meshes," AGARD Rept. 787, May 1992.

[6] Abgrall, R., "Design of an Essentially Non-Oscillatory Reconstruction Procedure on Finite-Element Type Meshes," Institut National de la Recherche en Informatique et en Automatique, Rept. 1584, Jan. 1992.

[7] Abgrall, R., "On Essentially Non-Oscillatory Schemes on Unstructured Meshes: Analysis and Implementation," *Journal of Computational Physics*, Vol. 114, No. 1, 1994, pp. 45–58.

[8] Suresh, A., and Jorgenson, P. C. E., "Essentially Nonoscillatory (ENO) Reconstructions via Extrapolation," AIAA Paper 95-0467, Jan. 1995.

[9] Shu, C.-W., and Osher, S., "Efficient Implementation of Essentially Non-Oscillatory Shock-Capturing Schemes," *Journal of Computational Physics*, Vol. 77, No. 2, 1988, pp. 439–471.

[10] Abgrall, R., Friedrich, O., and Sonar, T., "Efficient ENO Schemes: From Mühlbach Expansions to Multiresolution Analysis," *Computational Fluid Dynamics'96: Proceedings of the Third ECCOMAS CFD Conference, Paris, Sept. 1996*, edited by J.-A. Désidéri, Wiley, Chichester, England, UK, 1996, pp. 931–936.

[11] Godunov, S. K., "Difference Method for the Numerical Computation of Discontinuous Solutions of Equations of Hydrodynamics," *Matematicheskii Sbornik*, Vol. 47, No. 3, 1959, pp. 271–306.

[12] Woodward, P., and Colella, P., "The Numerical Simulation of Two-Dimensional Fluid Flow with Strong Shocks," *Journal of Computational Physics*, Vol. 54, No. 1, 1984, pp. 115–173.

D. S. McRae
*Associate Editor*