Computers Fluids Vol. 20, No. 4, pp. 399–409, 1991
Printed in Great Britain.

0045-7930/91 $3.00 + 0.00

A FINITE-VOLUME METHOD FOR THE EULER EQUATIONS ON ARBITRARY LAGRANGIAN–EULERIAN GRIDS

J. Y. TRÉPANIER, M. REGGIO,† H. ZHANG and R. CAMARERO
Ecole Polytechnique de Montréal, C.P. 6079, Montréal, Québec, Canada H3C 3A7

(Received 12 September 1990: in revised form 14 May 1991)

Abstract—This study presents a finite-volume method for the solution of 2-D/axisymmetric Euler equations using triangular moving grids. The flow simulation is carried out using Roe's approximate Riemann solver. The importance of the implicit treatment of the space conversation laws, based on geometric analysis, is emphasized. The procedure for reconstructing Roe's method for moving meshes is described and validated.

1. INTRODUCTION

Since the turn of the century, when Richardson presented his unfortunately unstable scheme, to this last decade, a long but straight route has been established for developing numerical CFD procedures. In this field, gas dynamics has constituted a major application. Until recently, most of the development of Euler solvers has been undertaken in a finite-volume framework, where the domain discretization is carried out using a logical mesh, with a quadrilateral (or hexahedron) used as the basic computational cell.

In order to tackle increasingly complex problems, the trend is to employ unstructured meshes with a triangle as the supporting element. In parallel to this finite-element-type discretization, the idea of an adaptive mesh has evolved because local grid enrichment is an appropriate way to achieve an appropriate degree of accuracy with an affordable number of grid nodes.

As life is not only non-linear but dynamic as well, the next step in the quest for new frontiers is to consider those problems associated with unsteady flows, those with moving boundaries being the most interesting. Applications range from internal flows with pistons, valves, rotor–stator interaction and external flows around oscillating airfoils, among others. For these cases, an entirely new strategy has to be developed based on a time-dependent grid.

In this work, we present a finite-volume methodology applied to the solution of 2-D/axisymmetric Euler equations on arbitrary moving grids. The discrete form of these basic equations will follow a procedure developed for general moving meshes described elsewhere [1]. The main characteristic of such a formulation relies on the fact that it respects in an intrinsic way the geometric conservation law (GCL), and on its applicability to any scheme initially developed for static grids. The present treatment may be differentiated from more standard procedures not only by the novel manner in which the GCL is implemented, but also by the fusion of Lagrangian and Eulerian steps into one single step.

Among the various modern schemes, the approximate Riemann solver introduced by Roe [2–4], and widely applied by others [5–8], has been selected to perform the numerical simulations in a moving grid situation. The spatial discretization employs unstructured meshes based on triangles, and a cell-centered approach.

The proposed methodology has been verified using a fundamental test consisting of the arbitrary motion of a grid in a constant-flow field. Basic examples are presented for simple geometries, and their analytical and computed results are compared.

†To whom all correspondence should be addressed.

2. GOVERNING SYSTEM

The Euler equation for 2-D/axisymmetric compressible flows in a general moving reference frame may be written in integral form as

$$\frac{\partial}{\partial t}\int_{V(t)} U\eta \, \mathrm{d}V + \oint_{S(t)} \mathbf{n}\cdot\mathbf{F}\eta \, \mathrm{d}S = \int_{V(t)} \mathbf{g}\eta \, \mathrm{d}V, \tag{1}$$

where U is the vector of dependent variables, $\mathbf{F}$ is the flux tensor, $\mathbf{n}$ is the outward unit vector normal to the boundary $S(t)$, which encloses the time-dependent volume $V(t)$. The source term $\mathbf{g}$ includes the external source flux g_e from the physics and the extra term g_a resulting from the axisymmetric formulation. The variable η accounts for the flow geometry. In axisymmetric flows, η is taken as the distance y from the axis, while in plane flow cases, it is taken as 1. In the latter case, $g_a = 0$. The open forms of U, $\mathbf{F}$ and f_a are [9]:

$$U = \begin{bmatrix} \rho \\ \rho\mathbf{u} \\ \rho E \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} \rho(\mathbf{u}-\mathbf{w}) \\ \rho(\mathbf{u}-\mathbf{w})\mathbf{u} + \mathbf{I}p \\ \rho(\mathbf{u}-\mathbf{w})E + \mathbf{u}\rho \end{bmatrix}, \quad g_a = \begin{bmatrix} 0 \\ \mathbf{e}_y p/y \\ 0 \end{bmatrix}, \tag{2}$$

where, ρ is the density, $\mathbf{u}$ is the fluid velocity, $\mathbf{w}$ is the velocity of the boundary of the volume, E is the specific energy, p is the pressure, $\mathbf{I}$ is the unit tensor and $\mathbf{e}_y$ is the unit radial vector. The cases $\mathbf{w} = \mathbf{u}$ corresponds to a Lagrangian system, and $\mathbf{w} = 0$ is a Eulerian one. In the present formulation, $\mathbf{w}$ is arbitrarily specified.

Equations (1) and (2) represent the conservation of mass, momentum and energy. In addition, an equation of state is considered in the general form

$$\mathbf{p} = p(\rho, e), \tag{3}$$

where $e = \frac{1}{2}\mathbf{u}\cdot\mathbf{u}$ is the specific internal energy. In the case of an ideal gas, equation (3) becomes

$$\mathbf{p} = (\gamma - 1)\rho e, \tag{4}$$

where γ is a constant representing the ratio of specific heat capacities of the fluid.

3. IMPLICIT GCL

3.1. The geometric laws

The GCL [10], which establish the relations for the conservation of surfaces (SCL) and volumes (VCL) of the control cells used in computational fluid dynamics, play key roles in the flow simulation, particularly when moving grids are used. If these laws are violated, a misrepresentation of the convective velocities or extra sources is encountered [1, 9].

The GCL stem from the evident principle that arbitrarily deforming or moving grids do not affect the flow field. They may thus be derived from, say, the continuity equation and properly selected flow fields. The VCL is obtained by selecting a constant fluid density and a zero velocity field, which results in

$$\frac{\partial V}{\partial t} - \oint_B \mathbf{w}\cdot\mathrm{d}S = 0. \tag{5}$$

The SCL, on the other hand, is obtained by assuming a uniform flow oriented in an arbitrary direction $\mathbf{a}$ in a non-moving mesh. The resulting, well-known analytical expression is

$$\oint_B \mathbf{a}\cdot\mathrm{d}S = 0 \quad \text{iff} \oint_B \mathrm{d}S = 0. \tag{6}$$

The discrete form of equation (5) from time t_n to $t_n + \Delta t$ gives

$$V^{n+1} - V^n \sum_i \int_{t_n}^{t_n+\Delta t} \int_{B_i} \mathbf{w}\cdot\mathrm{d}S \, \mathrm{d}t = \sum_i \Delta V_i, \tag{7}$$

where the superscript n denotes the value at time t_n, and the value of ΔV_i indicates the volumetric increase (or decrease) along the fact B_i. This equation states that, during a time interval, the

increased volume in a control cell equals the summation of the volumetric increases along its faces, and so is called the VCL.

The discrete form of equation (6) is

$$\sum_i \mathbf{a} \cdot S_i = 0. \tag{8}$$

This equation states that each cell must be closed, and is called the SCL. Although apparently trivial, equation (8) is fundamental for geometrically induced oscillations. The key point in maintaining the SCL in finite-volume schemes (and also in finite-difference formulations), is to evaluate exactly the surface vectors.

The importance of the VCL has long been recognized [10]. According to this procedure, the volume conservation is imposed by updating the volumes of control cells *explicitly* from equation (7). These updated volumes can be different depending on the discrete methods used for their computation. Specifically, they may differ from geometrically computed volumes. This situation is illustrated in the following example.

Consider a unit square, as shown in Fig. 1, with velocities defined at its vertices. Let us assign to nodes 1, 2, 3 and 4, the following Cartesian velocity components, respectively: [0,0; 1,0; 1,1; 0,1]. If the explicit volume conservation (EVCL) is implemented by applying a linear interpolation of the nodal velocities along each face, and if we consider a time step $\Delta t = 1$, the computed volume increment ΔV, given by the summation ΔV_1 and ΔV_2, equals 2. However, it can be clearly seen that the *actual* geometric volume increment is composed of the addition of ΔV_1, ΔV_2, ΔV_3 and ΔV_4, which equals 3.

In light of this result, and considering the intrinsic nature of the problem, the GCL must be satisfied using a geometric rationale. The fundamental idea, denoted by IGCL, is based on the evaluation of the volumetric changes along elementary faces of a control volume as it moves. The use of this methodology leads to a procedure that is independent of the flow scheme, and must be the link with the physical laws for a proper implementation of the geometric conservation.

3.2. The implicit approach

The motion of the grid may be specified on the nodes in terms of either the velocity **w** or the time-dependent position **r**. Without loss of generality, the nodal velocities are specified as constants (in magnitude and direction) during each time step, and then the new and old nodal positions are related by

$$\mathbf{r}^{n+\lambda} - \mathbf{r}^n = \lambda \mathbf{w}\, \Delta t, \tag{9}$$

where λ is a normalized time >0, but $\leqslant 1$, to indicate the new time location.

The main objective of the IGCL procedure is to calculate the volumetric increment of a cell, $V^{n+1} - V^n$, formed by the elementary ΔV surface increments. The shape of the sides, as well as the size of these ΔV, is not crucial as long as equation (7) is satisfied. What is most important is

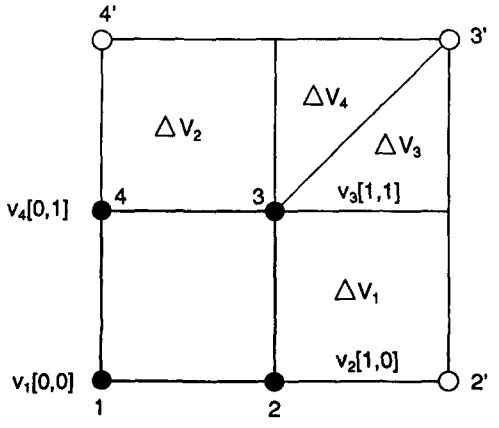

Fig. 1. Volume increments after the EVCL and IVCL.

the way in which the facial cell increments are computed and assembled to yield the volumetric increments of the cells. In order to meet this requirement, a unique way of defining the volumetric increment of the cell as a function of the individual ΔV has been proposed [1]. The procedure may be summarized as follows.

The motion of a typical control cell is characterized by the motion of its nodes. Each time that a node moves, a corresponding facial volume increment is obtained. After all the nodes are moved in a given sequence, the total volume increment is obtained by adding the individual facial increments. In axisymmetric/2-D cases (but not in 3-D cases) the order of the node movement is not important. Ultimately, it can be shown that for an axisymmetric domain the facial volume increment along a side with vertices 1 and 2 is defined by

$$\Delta V = \frac{\Delta t}{6}(2y_1^{n+1/2} + y_2^{n+1/2})\mathbf{w}_1 \times \Delta\mathbf{r}_{12}^n + \frac{\Delta t}{6}(2y_2^{n+1/2} + y_1^{n+1/2})\mathbf{w}_2 \times \Delta\mathbf{r}_{12}^n + \frac{\Delta t}{4}(2y_1^{n+2/3} + y_2^{n+2/3})\mathbf{w}_1 \times \mathbf{w}_2. \tag{10}$$

In equation (10) y_1, y_2 and $\mathbf{w}_1, \mathbf{w}_2$ denote, respectively, the radial coordinates and the velocity of nodes 1 and 2. The symbol $\Delta\mathbf{r}_{12}^n$ stands for the side vector evaluated at time n. Note that the superscripts of the radial coordinates y are not the same time level.

Associated with the facial volume increment ΔV, the relevant facial velocity w_n during a time step Δt is defined by [1, 9]

$$w_n = \frac{\Delta V}{S\,\Delta t}, \tag{11}$$

where $S(\Delta\mathbf{r}_{12})$ is the length of the side from $\mathbf{r}_1$ to $\mathbf{r}_2$. From this expression, one may see that w_n is not an independent geometric quantity like ΔV, but depends on the value of S. The area S may be evaluated at time t_n or t_{n+1}, as a linear combination of these. From the IGCL point of view, the resulting velocity is always consistent.

4. THE FLUX DEFINITION

The basic problem of any finite volume procedure is the calculation of the "specific" flux $f = \mathbf{n} \cdot \mathbf{F}$ at the faces of an element. The type of scheme will be defined as a consequence of the strategy chosen to evaluate this specific flux. If we consider, for example, the 1-D hyperbolic system of conservation laws,

$$U_t = f(U)_x,$$

with the semi-discrete conservation form

$$\frac{\partial U_1}{\partial t} = \frac{(f_{1+1/2} - f_{i-1/2})}{\Delta x},$$

the problem is to evaluate the numerical fluxes f at the $i + \frac{1}{2}$ and $i - \frac{1}{2}$ positions. Among the diverse possibilities, a large class of upwind-biased schemes can be considered based on the following relations [11]:

$$\sum \Delta f^-_{i+1/2} = f_{i+1/2} - f_i$$

$$\sum \Delta f^+_{i+1/2} = f_{i+1} - f_{i+1/2},$$

where the symbols Σ^- and Σ^+ denote summation over the negative and positive wave speeds. According to the above relations, the flux $f_{i+1/2}$ is given by

$$f_{i+1/2} = \tfrac{1}{2}[(f_i + f_{i+1}) - |\Delta f_{i+1/2}|]$$

with

$$|\Delta f_{i+1/2}| = \sum \Delta f^{+}_{i+1/2} - \sum \Delta f^{-}_{i+1/2}. \tag{12}$$

The class of methods that yield the terms in the summation based on the initial-value problem; i.e.

$$U_t = f_x(U)$$

and

$$U = \begin{cases} U_L, & x < 0, \\ U_R, & x > 0, \end{cases}$$

are known as Riemann solvers, which differ in the way in which this information is used. Some procedures evaluate the second terms of equation (12) from the exact solution of the Riemann problems [12], some will look for the solution of an approximate Riemann problem [2], and others will compute the correction on the basis of an approximate solution of the Riemann problem [13]. Also, the flux-splitting methodology can be invoked to calculate such terms. In the present approach, Roe's flux-difference scheme [2, 3] has been selected to carry out the numerical simulations using moving grids.

4.1. The Roe scheme on static grids

Before its application to Lagrangian–Eulerian grids, a brief account on the use of Roe's method on static meshes will be given. Although similar developments may be found in the literature [2, 3, 14], this description will serve to introduce the "flux variables" that will be used in the next section. The keystone of the Roe scheme is the introduction of an average state $\bar{A}$, which approximates the Jacobian $A = \partial f/\partial U$. This Roe matrix is such that for any given left and right pair of states (U_L, U_R):

(i) $\bar{A}(U_L, U_R) \to A(U)$ as U_L and $U_R \to U$.
(ii) $\bar{A}(U_L, U_R)$ has real eigenvalues and a complete set of linearly independent eigenvectors.
(iii) $f_R - f_L = \bar{A}(U_R - U_L)$.

Using these properties, Roe's average variables are given by

$$\bar{\rho} = \sqrt{\rho_R \rho_L}$$

$$\bar{\Phi} = \frac{\sqrt{\rho_R}\Phi_R + \sqrt{\rho_L}\Phi_L}{\sqrt{\rho_R} + \sqrt{\rho_L}}, \quad \bar{\Phi} = \bar{u}, \bar{h}.$$

Once all the averaged properties are obtained, the corresponding $\bar{A}$ is found and the linearized Riemann problem, expressed as

$$U_t = \bar{A}\frac{\partial U}{\partial x},$$

is considered at each interface. The exact solution of this approximate problem can be given in terms of the right eigenvector as

$$\Delta U = \sum_{j=1}^{4} \alpha_j \mathbf{e}_j,$$

where the α_js may be determined by multiplying this equation by each left eigenvector $\mathbf{le}_i$. Noting that $\mathbf{le}_i \cdot \mathbf{e}_j = \delta_{ij}$, the respective α_js are defined by

$$\alpha_j = \mathbf{le}_j \cdot \Delta U.$$

According to property (iii) above, when this vector ΔU is multiplied by the matrix $\bar{A}$, the components of the vector flux increment are obtained, which are simply the product of each component of U and the corresponding eigenvalues, i.e.

$$\Delta f = \sum \alpha_j \lambda_j \mathbf{e}_j. \tag{13}$$

In this form, the flux increment is regarded as a combination of "stretchings" by factors equal to the wave speeds λ and wave strengths α along the eigenvector axes. In 2-D the eigenvalues and the eigenvectors can be found in Ref. [14].

As a preamble to a discussion of the Roe scheme for moving grids, we introduce some useful "flux variables". First, we define the following:

$$Q = u_n S\,\Delta t, \quad C = cS\,\Delta T, \quad P = pS\,\Delta t, \tag{14}$$

where u_n represents the velocity normal to a given face S, c is the speed of sound, p is the pressure and Δt is the time step.

Second, we define the total flux $fS\,\Delta t$ across a face during a time step Δt as

$$F(Q) = QU + \begin{bmatrix} 0 \\ nP \\ u_n P \end{bmatrix}. \tag{15}$$

Finally, the "flux eigenvalues" of $\lambda S\,\Delta t$ are introduced. In a 2-D framework these are

$$\Lambda^{\mathrm{F}} = \begin{bmatrix} \Lambda_1 \\ \Lambda_2 \\ \Lambda_3 \\ \Lambda_4 \end{bmatrix}^{\mathrm{F}} = \begin{bmatrix} Q + C \\ Q - C \\ Q \\ Q \end{bmatrix}. \tag{16}$$

Using these definitions, and equations (12) and (13), the Roe flux for non-moving grids can be written as

$$F^{\mathrm{F}}(Q) = \frac{1}{2}\left[F_{\mathrm{R}}^{\mathrm{F}}(Q) + F_{\mathrm{L}}^{\mathrm{F}}(Q) - \sum_{i=1}^{4} \alpha_i |\Lambda_i^{\mathrm{F}}| \mathbf{e}_i \right], \tag{17}$$

where the superscript F has been used to denote a static or fixed mesh and the subscripts R and L characterize the right and left states at a given interface.

Once the flux variables are known, the properties U are advanced to the new time position $n+1$ by

$$(U^{n+1} - U^n)V^n = -\sum_{k=1}^{N\,\text{sides}} F^{\mathrm{F}}(Q_k) + gV\,\Delta t. \tag{18}$$

The method chosen to evaluate the right and left states at a given interface determines the order of the scheme. In its simplest form, the flow variables are considered to be piecewise constants on each cell, and the right and left states then correspond to these values. This leads to a first-order scheme. A second-order scheme may be obtained by using piecewise linear interpolation, instead of a piecewise constant, for the calculation of the fluxes at each cell interface. However, in order to avoid oscillations the interpolation must be performed together with a flux-limiting technique [7].

As usual, and depending on the time level in which the flow variables on the r.h.s. are assumed, the nature of the method will be defined. In the present formation, an explicit approach has been followed.

5. SCHEME FOR MOVING GRIDS

From a physical point of view, the grid motion only affects the convective variables. In particular, the Q term defined in equations (14) changes to

$$Q^{\mathrm{M}} = (u_n - w_n)S\,\Delta t. \tag{19}$$

From a mathematical point of view, the grid motion only modifies the eigenvalues of the Jacobian A. In 2-D the set of eigenvalues becomes $\lambda_1, \lambda_2, \lambda_3, \lambda_4 = u_n - w_n + c, u_n - w_n - c, u_n - w_n, u_n - w_n$. The λ_i eigenvalues "feel" the motion of the grid, but the wave strengths α_i and the eigenvectors $\mathbf{e}_i$ remain unchanged with respect to a static mesh formulation.

In order to calculate the new convective terms and eigenvalues, the velocity w_n of the face of a control volume is required. This velocity has to be specified in terms of the geometry, otherwise

the referred quantities will not be correctly defined. If the facial velocity w_n given in equation (11) is introduced, the convective term Q^M becomes

$$Q^M = u_n S\,\Delta t - \Delta V = Q^F - \Delta V,$$

where ΔV corresponds to the volumetric increment along a given face in equation (10). Using this definition of Q^M together with equation (15), the leading term of equation (17) is now

$$\tfrac{1}{2}[F_R^M(Q) + F_L^M(Q)] = \tfrac{1}{2}[(F_R^F(Q) - \Delta V U_R) + (F_L^F(Q) - \Delta V U_L)].$$

Thus, it is possible to imagine right and left fluxes defined by

$$F_R^M = F_R^F - \Delta V U_R$$

and

$$F_L^M = F_R^F - \Delta V U_L.$$

Accordingly, the new set of "flux" eigenvalues Λ^M is

$$\Lambda^M = \begin{bmatrix} \Lambda_1 \\ \Lambda_2 \\ \Lambda_3 \\ \Lambda_4 \end{bmatrix}^M = \begin{bmatrix} Q + C - \Delta V \\ Q - C - \Delta V \\ Q - \Delta V \\ Q - \Delta V \end{bmatrix}. \tag{20}$$

It is therefore possible to characterize the Roe flux for a moving grid in a manner analogous to that of a static one, i.e.

$$F^M(Q) = \frac{1}{2}\left[F_R^M(Q) + F_L^M(Q) - \sum_{i-1}^{4} \alpha_i |\Lambda_i^M| \mathbf{e}_i \right], \tag{21}$$

In this case, the U properties are advanced to the new time position $n+1$ by

$$U^{n+1}V^{n+1} - U^nV^n = -\sum_{k=1}^{N\,\text{sides}} F(Q_k^M) + \mathbf{g}V\,\Delta t. \tag{22}$$

From a practical point of view, and according to the adopted explicit approach, equation (22) may be written in the following convenient form:

$$U^{n+1} = \frac{V^n}{V^{n+1}}\left[U^n + \frac{1}{V^n}\left(-\sum_{k=1}^{N\,\text{sides}} F_k(Q_k^M) + \mathbf{g}V^n\,\Delta t \right) \right]. \tag{23}$$

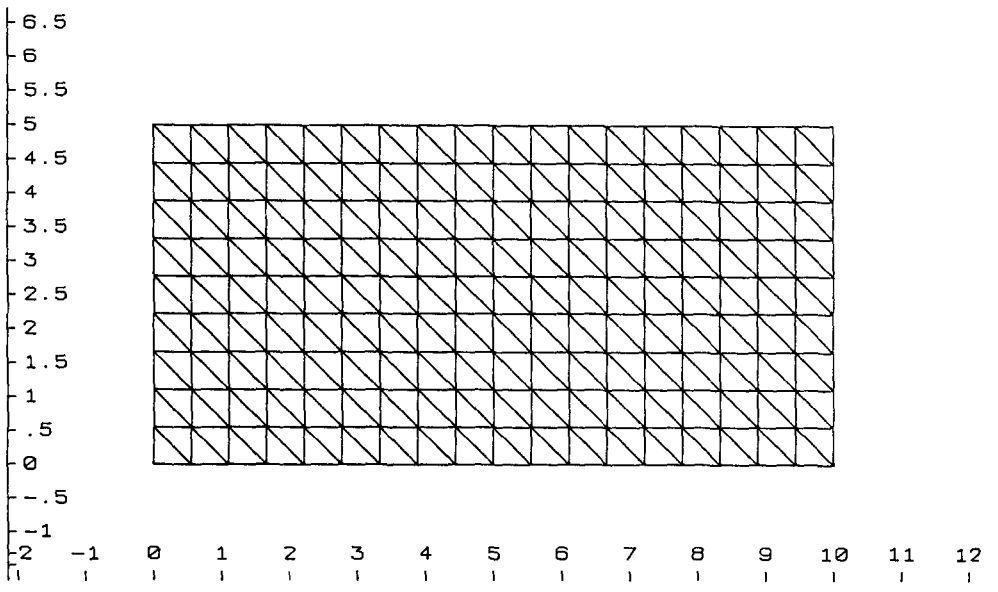

Fig. 2. Initial grid pattern for a rectangular domain.

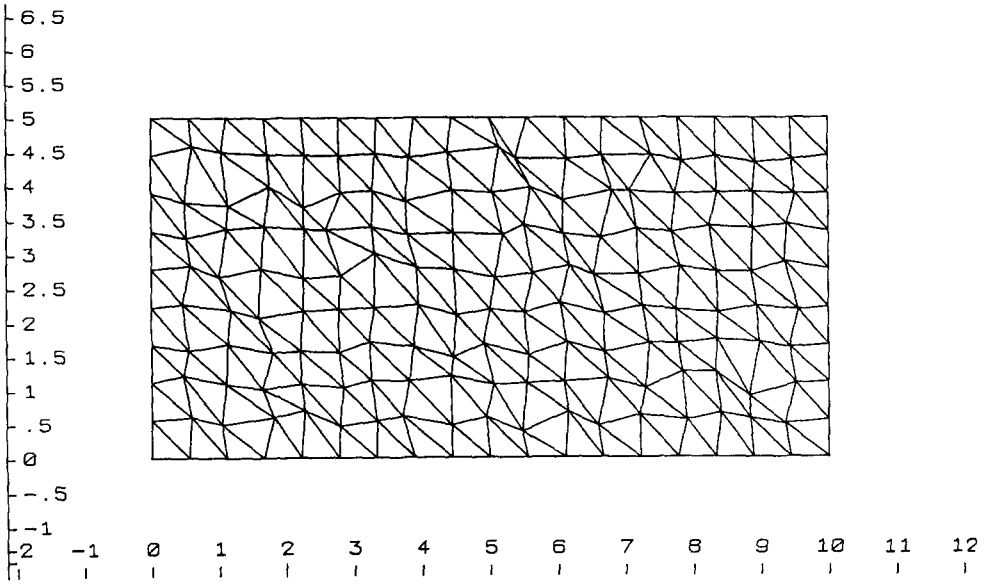

Fig. 3. Grid pattern after random motion.

According to this formulation the influence of the grid motion is only felt through the volume increments ΔV along the control volume faces. Once the original Q_k terms for static grids have been modified to $Q_k - \Delta V_k$, the U^{n+1} variables may be computed with the same code as for non-moving grids. The only remaining difference is a final scaling by the factor V^n/V^{n+1}. Since the ΔVs are computed exactly, this scheme keeps the order of the basic static grid scheme.

6. NUMERICAL TESTS

In order to validate the proposed procedure, four basic tests were carried out using an explicit first-order implementation.

6.1. Random grid motion

The first test has been devised to study the grid motion independence of the code. In fact, for a rectangular domain discretized according to the mesh illustrated in Fig. 2, a zero velocity field was imposed. Then the grid was moved at random. Figure 3 shows the grid pattern after 25 time steps. Figure 4 illustrates the computed isovalues of the $\rho - 1$ field, using single precision. The maximum level reached for these values is of the order of 10^{-5}, which verifies that the GCL is correctly implemented and that the grid motion does not affect the flow.

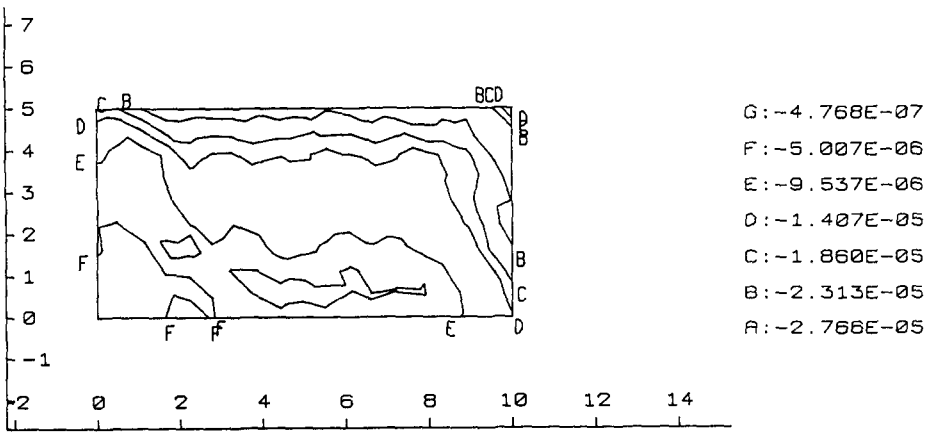

Fig. 4. $(\rho - 1)$ isovalues after random motion of the grid.

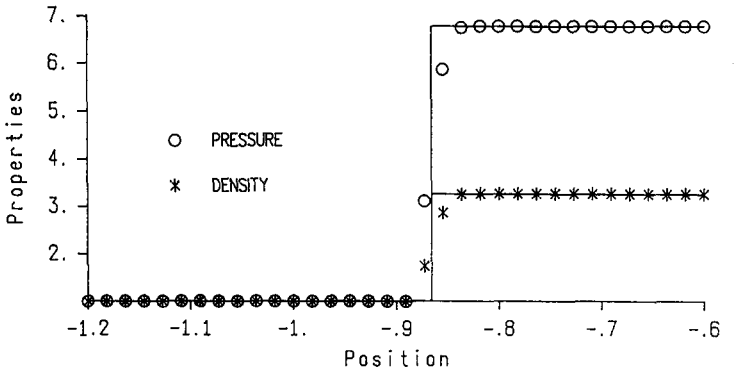

Fig. 5. Comparison of computed and analytical pressure distributions for the compression case; $v_p = 2, t = 0.3$.

6.2. 1-D compression and expansion

The second and third tests considered the 1-D compression and expansion of a gas. For both situations, the velocity of the grid nodes was set to vary linearly from the piston velocity for the node on this boundary, to zero for a node at the far field.

Figures 5 and 6 are comparisons, for the compression case, of the computed results and the analytical solutions for two different piston velocities. For both situations, with a dimensionless time and velocity piston of $t = 0.3$ and $v_p = 2$, respectively, and for $t = 0.6$ and $v_p = 1.0$, the agreement of the shock positions, and the intensities of the pressure and the densities, is good. The discrepancy in the values at the shock level is in part due to the size of the mesh, but also to the scheme, which is a first-order one.

The results for the expansion case are shown in Figs 7 and 8. Once again different kinematic and temporal conditions are used for comparison purposes. Figure 7 depicts numerical and analytical results for $v_p = 0.8$ and $t = 0.6$, while Fig. 8 shows the results for $v_p = 1.2$ and $t = 0.4$. For this test where the expansion waves are not able to reinforce one another as for compression waves, the concordance looks better.

6.3. 2-D probe

Once the tests establishing the correctness of the approach in 1-D has been carried out, an equivalent test was conducted in 2-D. In this test, parallel plates were separated by a 0.9 dimensionless distance, between which a rectangular probe 0.3 in width was placed halfway (Fig. 9). In this case a y-velocity component of zero was imposed, while the x-velocity component of the probe and of all grid nodes was constant.

The computed results for $v_p = 0.4$ and $t = 0.5$ are shown in Fig. 10 in terms of the pressure along the y-coordinate at four stations, at the probes head $x = x_p$, and at the positions $x = 0.05$, 0.25 and 0.45 in front of the probe.

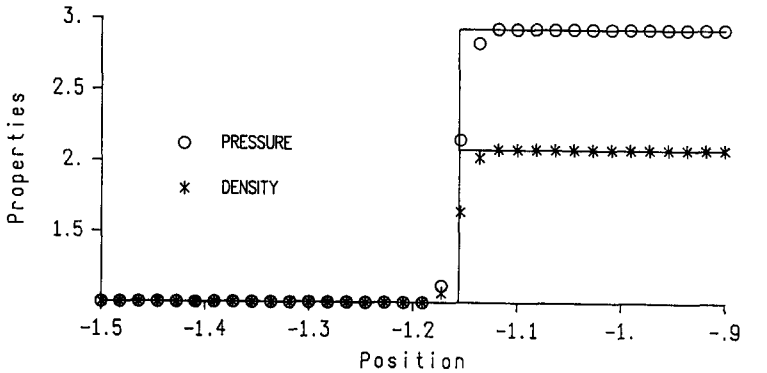

Fig. 6. Comparison of computed and analytical pressure distributions for the compression case; $v_p = 1, t = 0.6$.

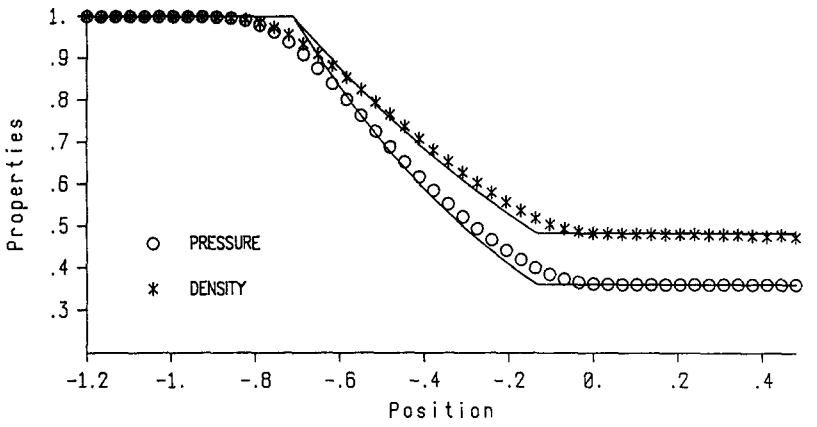

Fig. 7. Comparison of computed and analytical pressure distributions for the expansion case; $v_p = 0.8, t = 0.6$.

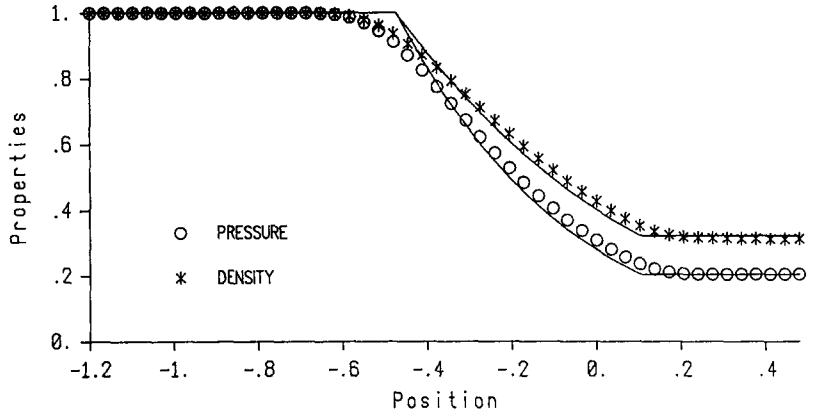

Fig. 8. Comparison of computed and analytical pressure distributions for the expansion case; $v_p = 1.2, t = 0.4$.

These results were compared with those of the inverse problem, i.e. a static probe, a non-moving grid and a fluid moving with a velocity equal to v_p. This comparison revealed what was expected: agreement at all positions. Because it is not possible to differentiate them graphically, only the first set of results is shown.

7. CONCLUSIONS

A finite-volume method based on Roe's scheme for the solution of 2-D/axisymmetric flows using a Lagrangian–Eulerian formulation has been described. The implicit space conservation for moving grids has been presented and the importance of this approach has been emphasized. The scheme for moving grids keeps the order of the basic static grid scheme. Thus, an extension to second-order accuracy can be obtained by using a piecewise linear reconstruction technique. Basic validation tests for 1-D and 2-D situations have been carried out using a first-order implementation of the scheme and the results indicates the correctness of the proposed approach.

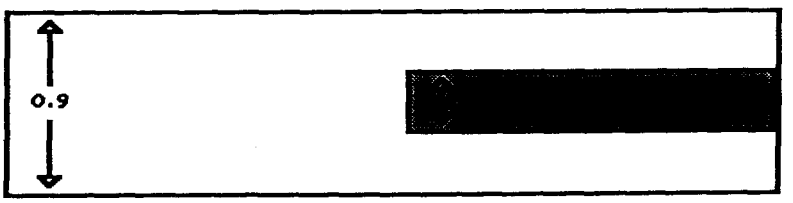

Fig. 9. Rectangular probe between parallel plates.

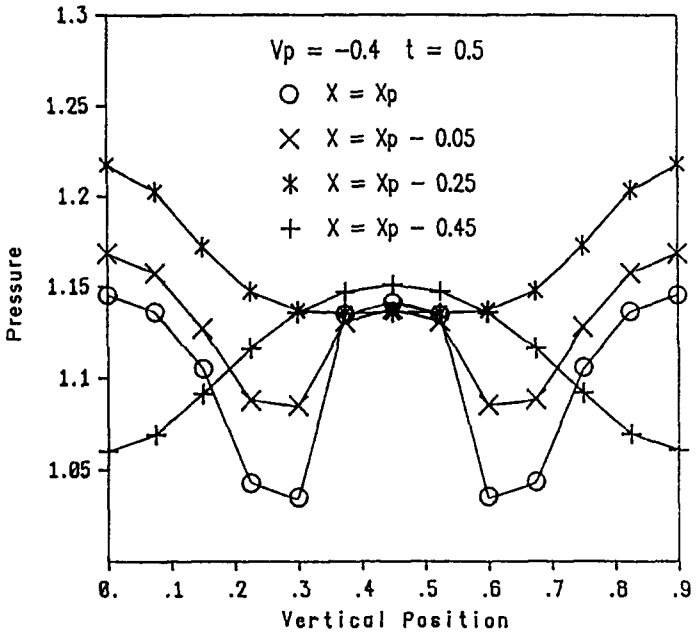

Fig. 10. Pressure distribution at different positions in front of the probe.

Acknowledgements—The authors thank the Natural Sciences and Engineering Research Council of Canada for financial support of this project in the form of an operating grant.

REFERENCES

1. H. Zhang, M. Reggio, J.-Y. Trépanier and R. Camarero, Implicit space conservation laws for multidimensional meshes. *Computers Fluids*. Submitted.
2. P. L. Roe, Approximate Riemann solvers, parameters vectors, and difference schemes. *J. Comput. Phys.* **43,** 357 (1981).
3. P. L. Roe and J. Pike, Efficient construction and utilization of approximate Riemann solutions. In *Computing Methods in Applied Sciences and Engineering*, Vol. 6 (Edited by R. Glowinski and J-L. Lion), pp. 499–518, North-Holland, Amsterdam (1984).
4. P. L. Roe, Characteristic-based schemes for the Euler equations. *A. Rev. Fluid Mech.* **18,** 337 (1986).
5. M. Brio and C. C. Wu, An upwinding differencing scheme for the equations of ideal magnetohydrodynamics. *J. Comput. Phys.* **75,** 400 (1988). Also, *Fluids* **8,** (1988).
6. D. Whitaker and B. Grosmann, Two-dimensional Euler computations on a triangular mesh using an upwind, finite-volume scheme. Paper AIAA-89-0470.
7. J. T. Barth and D. C. Jespersen, The design and application of upwind schemes on unstructured meshes. Paper AIAA-89-0366.
8. D. Whitaker, Solution algorithms for the two-dimensional Euler equation on unstructured meshes. Paper AIAA-90-0697.
9. M. Vinokur, An analysis of finite-difference and finite-volume formulations of conservation laws. *J. Comput. Phys.* **81,** 1 (1989).
10. P. D. Thomas and C. K. Lombard, Geometric conservation law and its application to flow computations on moving grids. *AIAA Jl* **17,** 1030 (1979).
11. S. Chakravarty, High-resolution upwind formulations for the Navier–Stokes equations. *Computational Fluid Dynamics,. VKI Lecture Series*, p. 05 (1988).
12. S. K. Godunov, Finite-difference method for numerical computations of discontinuous solutions of equations of fluid dynamics. *Mat. Sbornik* **47,** 271 (1959).
13. S. Osher, and S. Charkravarthy, High-resolution scheme and the entropy condition. *SIAM Jl Numer. Analysis* **21,** 955 (1984).
14. P. Glaister, A shock capturing scheme for body-fitted meshes. *Int. J. Numer. Meth. Fluids* **8,** 97 (1988).