Table 1 Summary of quadrature formulas for $I_1(x)$

$W(x)$	$(1-x^2)^{-1/2}$	$(1-x^2)^{1/2}$	$\sqrt{(1-x)/(1+x)}$	$\sqrt{(1+x)/(1-x)}$
$P_N(x)$	$\cos N\theta$	$\sin(N+1)\theta/\sin\theta$	$C\sin(2N+1)(\theta/2)/\sin(\theta/2)$	$C\cos(2N+1)(\theta/2)/\cos(\theta/2)$
$Q_N(x)$	$-\frac{\pi}{2}\sin N\theta/\sin\theta$	$\frac{\pi}{2}\cos(N+1)\theta$	$-\frac{\pi}{2}C\cos(2N+1)(\theta/2)/\cos\theta$	$-\frac{\pi}{2}C\sin(2N+1)(\theta/2)/\sin(\theta/2)$
ξ_i	$\cos[(2i-1)\pi/2N]$, $i=1,2,\ldots,N$	$\cos[i\pi/(N+1)]$, $i=1,2,\ldots,N$	$\cos[2i\pi/(2N+1)]$, $i=1,2,\ldots,N$	$\cos[(2i-1)\pi/(2N+1)]$, $i=1,2,\ldots,N$
e_i	$\frac{\pi}{N}(1-\xi_i^2)^{1/2}$	$\frac{\pi}{N+1}(1-\xi_i^2)^{1/2}$	$\frac{2\pi}{2N+1}(1-\xi_i^2)^{1/2}$	$\frac{2\pi}{2N+1}(1-\xi_i^2)^{1/2}$
x_j	$\cos(j\pi/N)$, $j=1,2,\ldots,N-1$	$\cos[(2j-1)\pi/2(N+1)]$, $j=1,2,\ldots,N+1$	$\cos[(2j-1)\pi/(2N+1)]$, $j=1,2,\ldots,N$	$\cos[2j\pi/(2N+1)]$, $j=1,2,\ldots,N$

$\theta=\cos^{-1}x$; $C=\Gamma(N+1/2)/N!\sqrt{\pi}$; $\Gamma(m)$ is the gamma function with argument m.

Table 2 Comparison of computed and exact solutions of

$$I_2(x)=\int_{-1}^{1}\frac{\xi^4 d\xi}{\sqrt{1-\xi^2}\,(\xi-x)^2},\quad -1<x+1$$

x_j	Exact $I_2(x_j)$	Computed $I_2(x_j)$
	For $N=6$	
0.72741239	6.55771764	6.55771764
0.26621648	2.23874179	2.23874179
−0.26621648	2.23874179	2.23874179
−0.72741239	6.55771764	6.55771764

whose roots are

$$x_j=\cos[j\pi/(N+1)],\ j=1,2,\ldots,N$$

the roots of $Q_N'(x)$, unfortunately, do not seem to be extractable in closed form for an arbitrary value of N. Therefore, they must be determined numerically, say, by using the bisection method. We may note, however, that since $Q_N(x)$ is a polynomial, all of whose roots lie in the interval $-1<x<+1$, $Q_N'(x)$, which will be a polynomial of one degree less, will also have all of its roots in the interval $-1<x<+1$. Indeed, between two consecutive roots of $Q_N(x)$ will lie a root of $Q_N'(x)$.

V. Numerical Example of $I_2(x)$

For illustration, we choose to evaluate numerically the integral

$$I_2(x)=\int_{-1}^{1}\frac{\xi^4 d\xi}{\sqrt{1-\xi^2}(\xi-x)^2}=\pi(3x^2+0.5)\qquad(15)$$

whose exact solution is also noted above. The $Q_N'(x)$ for the weight function $(1-x^2)^{-1/2}$ is

$$Q_N'(x)=-\frac{\pi}{4}[(N+1)\sin(N-1)\theta-(N-1)\sin(N+1)\theta]/\sin^3\theta$$

where $\theta=\cos^{-1}x$. Let us choose $N=6$, for which we have

$$Q_6'(x)=-\pi(80x^4-48x^2+3)$$

The computed and exact values of $I_2(x)$ are shown in Table 2 and, as expected, they are identical. In passing, we note that integrands having any of the other three weight function, 2–4 in Sec. IV, can always be reinterpreted to have $(1-x^2)^{1/2}$ as its weight function.

VI. Conclusions

Stark's quadrature formula for Cauchy integrals has been extended in a simple manner to evaluate double-pole singular integrals, which occur in linear lifting surface theory.

References

[1]Mangler, K. W., "Improper Integrals in Theoretical Aerodynamics," British Aeronautical Research Council, London, R&M 2424, 1951.

[2]Stark, V. J. E., "A Generalised Quadrature Formula for Cauchy Integrals," *AIAA Journal*, Vol. 9, Sept. 1971, pp. 1854–1855.

[3]Bera, R. K., "The Numerical Evaluation of Cauchy Integrals," *International Journal of Mathematical Education in Science and Technology*, (to be published).

Grid Embedding Technique Using Cartesian Grids for Euler Solutions

R. A. Mitcheltree*
North Carolina State University, Raleigh, North Carolina
M. D. Salas†
NASA Langley Research Center, Hampton, Virginia
and
H. A. Hassan‡
North Carolina State University, Raleigh, North Carolina

Introduction

NUMERICAL solution of the Euler equations is typically carried out by discretization of the flowfield and then solution of the resulting set of coupled equations for each node. Construction of these grids with the requisite smoothness and point clustering remains one of the most difficult tasks associated with the solution process. This is especially true for complex configurations.

The introduction of finite volume formulation made it possible to satisfy the wall boundary conditions, in the integral sense, irrespective of the shape of the boundary. However, a

Received May 26, 1987; revision received Sept. 25, 1987.

*Research Assistant, Mechanical and Aerospace Engineering. Student Member AIAA.

†Head, Theoretical Aerodynamics Branch, Transonic Aerodynamics Division. Associate Fellow AIAA.

‡Professor, Mechanical and Aerospace Engineering. Associate Fellow AIAA.

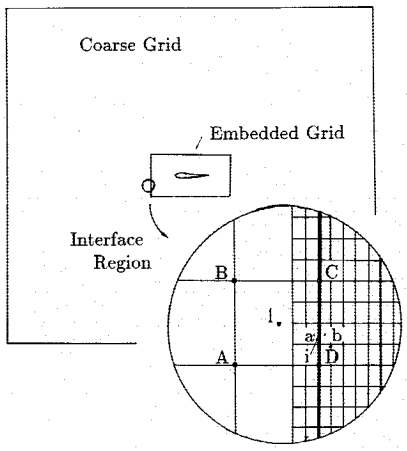

Fig. 1 Close-up of interface region.

smoothly varying grid with a minimum of skewness is required for accurate solutions.

A Cartesian grid involves grid lines which are everywhere orthogonal to one of the coordinate axes. Hence grid generation is simple. Since the grid is not aligned with the body, however, additional body boundary considerations are required. These considerations are easily and logically implemented for finite volume formulations of the Euler equations, as demonstrated in two-dimensions by Clark et al.[1] and in three-dimensions by Gaffney et al.[2]

Use of a Cartesian grid results in unnecessarily fine resolution extending to the far field boundaries. Thus, the objective of this work is to increase the efficiency of Cartesian grids through the use of a grid embedding scheme capable of arbitrary refinement ratios.

Usab and Murman[3] first implemented grid embedding for the Euler equations using mapped grids. Additional significant work has been done by Berger and Jameson[4] and Allmaras and Barron.[5]

Grid embedding techniques are applied to the solutions of the Euler equations on Cartesian grids for the NACA 0012 airfoil and multielement SKF1.1 airfoil at transonic speeds. Based on comparisons with solutions on fine grids, it is shown that the scheme presented results in reduced memory requirements and substantial reduction in computational time without loss of accuracy.

Embedding Formulations

As shown in Fig. 1, grid embedding involves a grid which along some line is abruptly refined in space. This line is referred to as the interface between the coarse global grid and the fine or embedded inner grid. The refinement ratio at an interface location is defined as the ratio of the number of inner grid cells per coarse grid cell.

A global grid covers the entire computational domain and a rectangular fine grid is overlayed in the immediate region of the airfoil. The strategy is to solve an initial boundary-value problem on each grid. In general, the solution proceeds by first advancing the inner embedded grid one step, or a small number of steps. Fluxes determined for the inner grid at the interface are stored and then reused as the global grid is then advanced. Finally, bilinear interpolation relates the new global grid solution back to the fine grid points at the interface and the process is repeated.

Time stepping for both grids is handled separately by a Runge-Kutta scheme similar to that of Ref. 6. This is a four-stage method with second and fourth difference damping as well as residual smoothing. (Residual smoothing is discussed in Ref. 7.) Far field boundary conditions are calculated using the method of characteristics.

Communication between the global grid and the embedded grid occurs at their interface. Handling this interface requires new flux and damping formulations for each grid in this region. Consideration must be given to conservation, stability and accuracy when deriving these formulations.

For the embedded grid, the flux and damping formulations hinge on the definition of two embedded grid cells outside of the interface. These cells are defined and updated via bilinear interpolation.

The interpolation first defines values at the corners of the large cell. Corners A and B in Fig. 1 are computed as the average of the four neighboring global grid points. Corners C and D are computed as the average of the four neighboring embedded cells. The flow value W at location a is then computed by

$$W_a = W_1 + \frac{\partial W}{\partial x}\Delta x + \frac{\partial W}{\partial y}\Delta y + \frac{\partial^2 W}{\partial x\,\partial y}\Delta x \Delta y$$

where

$$\frac{\partial W}{\partial x} = \frac{1}{2}(W_C + W_D - W_B - W_A)/\Delta X,$$

$$\frac{\partial W}{\partial y} = \frac{1}{2}(W_B + W_C - W_A - W_D)/\Delta Y$$

$$\frac{\partial^2 W}{\partial x\,\partial y} = (W_C - W_A + W_B - W_D)/\Delta X \Delta Y \qquad (1)$$

This interpolation formula uses information from embedded cells within the interface. Berger[8] has observed that any interface scheme which couples the fine grid points across the interface can give rise to an oscillating wave emanating from the interface into the fine grid. That instability was not observed in the present work.

Computation of the flux across the interface for the inner grid iteration then uses one of the two additional points defined outside of the interface. The flux at i, Fig. 1, is determined from the values of the flow variables estimated at i by

$$W_i = (1/2)\,(W_a + W_b) \qquad (2)$$

Allmaras and Barron[5] have shown that there are no flux formulations at the interfaces that are both conservative and second order accurate spatially. Therefore, a conservative first order accurate formulation is used.

Damping for the embedded grid cells involves a blending of second and fourth differences, as discussed in Ref. 6. Therefore, the two additional cells outside of the interface are sufficient to define the damping term for all interior cells.

Following the embedded grid step, the global grid values are updated. The flux across the interface for an adjacent global grid cell is the sum of the inner grid cell fluxes across the interface. These fluxes were stored during the embedded grid step and then reused. This represents a conservative flux formulation. Conservation is especially important in the neighborhood of shocks, as discussed by Berger.[8]

Damping on the global grid, as with the inner grid, involves second and fourth differences; two global grid cells inside the interface are required. These values are computed as the volume weighted average of the values at the inner grid cells contained within each larger cell.

At the end of the global grid step, the far field boundary conditions are applied. The process then repeats as the two embedded grid cells outside the interface are updated by the bilinear interpolation.

Results

Two supercritical configurations have been examined. Results are presented in the form of comparisons between embedded and nonembedded solutions and comment is made on the associated savings realized.

The first case considered is that of flow past an NACA 0012 airfoil at a Mach number of 0.8 and angle of attack of 1.25 deg.

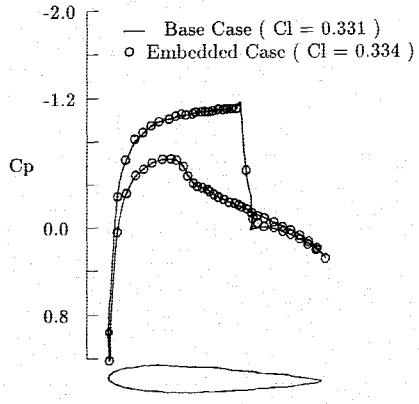

Fig. 2 Pressure distribution on the NACA 0012 airfoil, $M = 0.8$, $\alpha = 1.25$ deg.

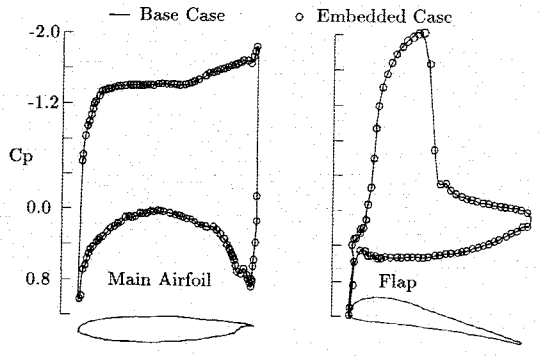

Fig. 3 Pressure distribution on the flapped SKF1.1 airfoil, $M = 0.76$, $\alpha = 3$ deg.

The base grid (no embedding) consists of 93 × 79 mesh cells. When embedding is employed, a global grid of 63 × 47 cells is used. Within this grid, a 59 × 51 cell fine grid is embedded. As shown in Fig. 1, which is drawn to scale, the interface is placed about 0.5–0.6 chords from the body. Although all global grid cells require memory allocation, only those cells outside the embedded region are computed. At certain interface locations for this grid, the refinement ratio is 4 to 1. Ratios of 3 to 1 and 2 to 1 also appear. Because the lift coefficient depends on the location of the outer boundaries,[9] calculations are carried out with outer boundary extents in all directions at 5 chords and then 10 chords. Extrapolation, as in Ref. 9, is then used to eliminate the dependence of lift coefficient C_l on boundary extent. Both grids predict about the same value of C_l when the effects of grid extent are removed (0.331 for the base grid vs 0.334 for the embedded grid). Figure 2 shows a comparison of pressure coefficient C_p distribution for the two grids when the outer boundary is placed at 10 chords. For this case, embedding results in a 21% reduction in memory requirements and a 60% reduction in CPU time.

The sensitivity of the above results to the location of the interface is discussed next. The interface for the above NACA 0012 case was placed 23% closer to the airfoil surface, i.e., to $y/c = 0.39$ where c is the chord length. The resulting change in C_l and convergence rate was less than 1%. This contrasts with the findings of Allmaras and Barron[5] which suggest that when the interface intersects a shock, small modifications in interface location produce large effects on convergence and accuracy.

An advantage of Cartesian grids is the ease with which they handle complex geometries. To illustrate this diversity, the flapped SKF1.1 airfoil is considered. Calculations are presented for the case of Mach number 0.76 and 3 deg angle of attack. The base grid is 166 × 101. The embedded grid consists of a global grid of 56 × 45 cells within which is embedded a 144 × 81 cell fine grid. The refinement ratios vary as high as 8 to 1. The extent of the outer boundary is 10 chords. Figure 3 compares C_p distributions for the main airfoil on the base grid ($C_l = 1.66$) and the embedded grid ($C_l = 1.64$). For this case, embedding results in 15% savings in memory and 45% savings in CPU time. As in the NACA 0012 case discussed earlier, the interface intersects the shock. Again, this arrangement does not introduce any problems in convergence or solution accuracy.

In conclusion, a robust embedding scheme which allows arbitrary refinement is developed for cell-centered finite volume formulations. When used in conjunction with the Runge-Kutta time stepping scheme and Cartesian grids, the scheme results in significant reduction in memory and computation time requirements without loss of accuracy.

Acknowledgments

This work was supported in part by NASA Cooperative Agreement NCCI-22.

References

[1]Clarke, D. K., Salas, M. D., and Hassan, H. A., "Euler Calculations for Multi-Element Airfoils Using Cartesian Grids," *AIAA Journal*, Vol. 24, March 1986, pp. 353–358.

[2]Gaffney, R. L., Jr., Salas, M. D., and Hassan, H. A., "Euler Calculations for Wings Using Cartesian Grids," AIAA Paper 87-0356, Jan. 1987.

[3]Usab, W. J. and Murman, E. M., "Embedded Mesh Solutions of the Euler Equations Using a Multiple Grid Method," *Proceedings of the AIAA Computational Fluid Dynamic Conference*, Danvers, MA, AIAA Paper 83-1946, July 1983.

[4]Berger, M. J. and Jameson, A., "Automatic Adaptive Grid Refinement for the Euler Equations," *AIAA Journal*, Vol. 23, No. 4, April 1985, pp. 561–568.

[5]Allmaras, S. R. and Barron, J. R., "Embedded Mesh Solutions of the 2-D Euler Equations: Evaluation of Interface Formulations," AIAA Paper 86-0509, Jan. 1986.

[6]Jameson, A., Schmidt, W., and Turkel, E., "Numerical Solution of the Euler Equations by Finite Volume Methods Using Runge-Kutta Time Stepping Schemes," AIAA Paper 81-1259, June 1981.

[7]Jameson, A. J. and Baker, T. J., "Solution of the Euler Equations for Complex Configurations," AIAA Paper 83-1929, July 1983.

[8]Berger, M. J., "On Conservation at Grid Interfaces," ICASE Report No. 84-43, NASA Langley Research Center, Sept. 1984.

[9]Thomas, J. L. and Salas, M. D., "Far-Field Boundary Conditions for Transonic Lifting Solutions to the Euler Equations," AIAA Paper 85-0020, Jan. 1985.

Buckling Analysis of Sandwich Columns of Linearly Varying Thickness

N. Paydar*
Purdue University, Indianapolis, Indiana

Nomenclature

d = diameter of an inscribed circle in a honeycomb cell
D_o = flexural stiffness of the column at $x = 0$
E = modulus of elasticity of the face material
G' = core effective shear modulus
G_c = shear modulus of the core material
H = nondimensional measure of core thickness ($= h/h_o$)
h = core thickness
h_o = core thickness at $x = 0$
K = core geometric factor
k = critical load parameter
ℓ = length of the column

Received Nov. 20, 1986; revision received Oct. 9. 1987.

*Assistant Professor of Mechanical Engineering, School of Engineering and Technology.