A CARTESIAN, CELL-BASED APPROACH FOR ADAPTIVELY-REFINED SOLUTIONS OF THE EULER AND NAVIER-STOKES EQUATIONS

William J. Coirier*
NASA Lewis Research Center, Cleveland, Ohio

Kenneth G. Powell**
The University of Michigan, Ann Arbor, Michigan

ABSTRACT

A Cartesian, cell-based approach for adaptively-refined solutions of the Euler and Navier-Stokes equations in two dimensions is developed and tested. Grids about geometrically complicated bodies are generated automatically, by recursive subdivision of a single Cartesian cell encompassing the entire flow domain. Where the resulting cells intersect bodies, N-sided "cut" cells are created using polygon-clipping algorithms. The grid is stored in a binary-tree data structure which provides a natural means of obtaining cell-to-cell connectivity and of carrying out solution-adaptive mesh refinement. The Euler and Navier-Stokes equations are solved on the resulting grids using a finite-volume formulation. The convective terms are upwinded: A gradient-limited, linear reconstruction of the primitive variables is performed, providing input states to an approximate Riemann solver for computing the fluxes between neighboring cells. The more robust of a series of viscous flux functions is used to provide the viscous fluxes at the cell interfaces. Adaptively-refined solutions of the Navier-Stokes equations using the Cartesian, cell-based approach are obtained and compared to theory, experiment and other accepted computational results for a series of low and moderate Reynolds number flows.

I. INTRODUCTION

For complicated geometries, unstructured grids can be easier to generate than structured grids, which is directly responsible for their increasing popularity. Traditionally unstructured grids, with volume grids comprised of triangles in two-dimensions and tetrahedra in three-dimensions, are typically generated by first discretizing the bounding surfaces and then filling the volume grid by, say, an advancing front method, or, by triangulating a cloud of points. Both approaches are based upon a user specified surface discretization which is closely coupled to the volume grid generation by requiring the specified faces on the boundary surfaces to be faces of cells in the volume grid. In the approach considered here, the volume grid and surface description are not strongly coupled in this manner. The computational boundaries are described functionally, and are "cut" out of the automatically generated, Cartesian-cell based volume grid, yielding N-sided cells near the boundaries. This yields a computational grid comprised mostly of square, unit aspect ratio (Cartesian) conservation volumes with polygonal boundary cells. The ability to automatically create the volume grid and the cut, boundary cells gives the Cartesian-cell approach its utility, but adds some complexity to the numerics and resulting computer code. Since the cell geometry and hence the cell-to-cell connectivity for all cells is *not* known apriori, a unique data structure is needed to describe the conservation volumes. Indeed, it is this complication that sets the Cartesian-cell based approach apart from most of the traditional unstructured grid approaches that are becoming more and more prevalent today.

The Cartesian, cell-based approach has been used for solving a variety of problems, and variants of the approach have proven to be quite useful for computing both unsteady [6,7,26,27,5,10,25] and steady

* coirier@lerc.nasa.gov

** powell@engin.umich.edu

[11,8,9,16,15,19,22] adaptively-refined solutions to the Euler equations and for solutions to the transonic full potential equation [33]. Recently, the usefulness of the automated gridding and adaptive mesh refinement capabilities of the solver have been demonstrated for adaptively-refined solutions of the Euler equations about large aircraft configurations [23]. The automated mesh generation combined with the adaptive-mesh refinement capability of this approach offers the potential of obtaining accurate solutions with minimal user intervention. The work shown here investigates the extension of this capability to solving the compressible Navier-Stokes equations in an upwinded, finite-volume framework.

II. GRID GENERATION PROCEDURE

The grid is generated by the recursive subdivision of a single cell, and during the creation of the grid, the hierarchical relation between newly created cells and their parents are stored in a binary tree data structure. The cut cells, which are the background Cartesian cells *cut* into polygons, are created automatically using many concepts borrowed from computer graphics applications. Since they are hierarchically related to their Cartesian parents, they are also stored in the tree. This procedure of cell cutting is a subject unto itself and its robustness is absolutely crucial for the utility of this approach. The cell cutting methodology used here is based upon a polygon clipping algorithm [31], where a subject polygon (the body of interest) is "clipped" against a convex clipping polygon, which for this case is the Cartesian cell. The subject polygon can be formulated to include arbitrary functional descriptions of the bounding faces (i.e.-geometric description), which is used here. This particular clipping operation yields the logical and operation between the clipping polygon and the subject polygon. This operation will yield the correct cells when the subject polygon describes the outer boundary of a flow domain, but needs to be modified slightly when the subject polygon describes an inner boundary. When this is the case, the region needed is recovered from the clipping operation using a list directed vertex insertion procedure. Details behind the grid generation are presented in more detail in [12]. Once a suitable geometric description of the computational boundaries is made, the grid generation is automatic, and since the procedure is recursive and tree based, it is also efficient. An example of the grid generation is shown in Figure 1, where a coarse, base grid in a flow passage representative of the cooling passage within a turbine blade is shown. The geometry corresponds to that in [29], and is a projection onto the x-y plane of the geometry located along a curved surface about the turbine blade mean chord line. The grid contains 2640 cells and was generated in 218 seconds on an IBM RS6000 Model 560 workstation. The geometry definition is made using a linearly represented continuous outer boundary and 14 cooling fins.

The use of this particular grid and data structure easily allows adaptive mesh refinement, and away from bodies yields smooth grids. The binary tree data structure provides a logical means of finding cell-to-cell connectivity by logic based tree traversals and allows a straightforward means to perform mesh refinement and coarsening via tree branch growth and pruning. There are many niceties afforded by this data structure and grid setup. The grid hierarchy is amenable to multi-grid [17] and provides a natural means of domain decomposition that appears well suited for coarse grain parallel computations.

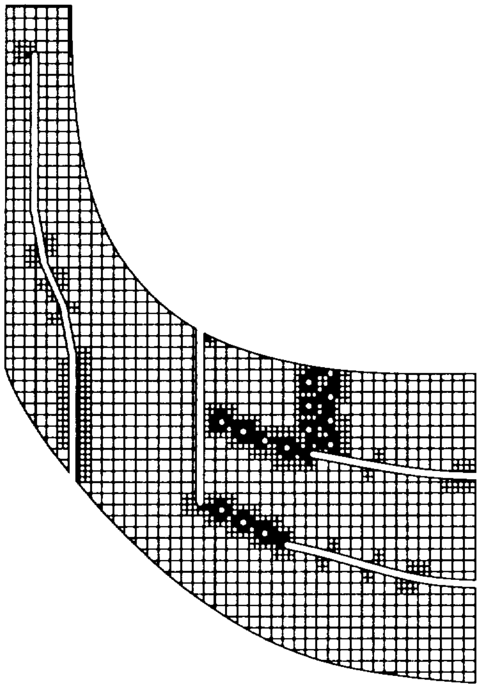

Figure 1 Representative
Turbine Cooling Passage Grid

III. SOLUTION OF THE EULER EQUATIONS USING A CARTESIAN,CELL-BASED APPROACH

The Euler equations are solved using a cell-centered, finite-volume, upwinding approach. A limited, linear reconstruction of cell-averaged data is used to provide input to a numerical flux function, yielding the flux through cell-to-cell interfaces. The numerical fluxes are computed in an upwind fashion using an appropriate approximate Riemann solver. These fluxes are then used to perform a flux balance upon the conservation volume, which is then used to advance the conserved variables in time. The procedure follows standard practice for a finite-volume scheme. The solution procedure can be broken up into 3 stages; reconstruction, flux construction and then evolution to steady-state. A more detailed description of the procedures is shown in [12].

The variation of the cell primitive variables in each cell is reconstructed using a linear reconstruction procedure, in the spirit of MUSCL interpolation, based on the Minimum-Energy reconstruction presented by Barth [3]. The Minimum-Energy reconstruction minimizes the Frobenius norm of the differences between the cell averages of the reconstructing polynomial and the cell averages of the support set. This reconstruction procedure is K-exact, in the sense that if a linear function is cell averaged upon the mesh, the reconstructed polynomial returns the same, exact, linear function. The Minimum-Energy reconstruction procedure provides the framework of extending the order of reconstruction, but only a linear reconstruction is considered here.

This reconstruction of cell averaged data does not preclude the introduction of new extrema: There is no means to ensure that the reconstructed solution is bounded by the data used to perform the reconstruction. To enforce this, the reconstruction is limited by evaluating the cell averaged data of the support cells used, reducing the reconstructed gradient to achieve monotonicity of the data. This will in turn guarantee monotonicity of the solution if the numerical flux function is a positivity preserving function (which for an upwind scheme is sufficiently implied by positivity of the dissipation matrix), and provided that a proper choice is made for the time step. The concept of restricting the local solution to be bounded by its immediate neighbors is based upon a discrete interpretation of a local maximum principle, and has been used to evaluate the stencils obtained for a model equation of the viscous terms of the Navier-Stokes equations, in [12]. The limiting procedure implemented here is based upon the procedure shown by Barth and Jespersen [4]. To ensure monotonicity of the reconstruction at cell interfaces, the solution is required to be bounded by the data used to perform the reconstruction. A single limiter, $\Phi = min(\phi_j)$ is found for all the primitive variables and applied to the reconstructions.

The inviscid numerical fluxes may be computed using a variety of approximate Riemann solvers: It is a simple matter to supply a different numerical flux function by replacing the approximate Riemann solver in the flow solver. For the work shown here, most of the computations have been performed using the AUSM scheme of Liou and Steffen [21]. This novel flux function combines the efficiency of flux vector splitting with the accuracy of flux difference splitting. The derivation and use of this flux function is available in [21].

For simplicity, the semi-discrete form of the equations are advanced in time using a multi-stage scheme. A spatially varying time step is used, and is indeed quite necessary, since there is typically a many order variation in cell size across the mesh due to cell refinement and cutting. A generic multi-stage scheme is used to advance the solution from the n-th to the (n+1)-th time level.

III.a SOLUTION ADAPTIVE MESH REFINEMENT

The Cartesian, cell-based approach gains its strength primarily from two features; the ability to compute flows about complicated geometries where the initial grid is obtained automatically, and by the inherent ease in which adaptive mesh refinement can be performed. Adaptive mesh refinement is an attempt to improve the quality of a solution by adding cells locally where an increased resolution is desired, and by possibly removing cells where the current resolution is unnecessarily too high. This feature, coupled with the automated means of mesh generation, attempts to yield grid converged solutions about geometrically complicated domains with minimal user intervention.

Each level of adaptive mesh refinement is comprised of two stages. In the first stage, refinement criteria are constructed for all cells on the mesh, and then in the second stage, cells are tagged for refinement or coarsening based on this criteria. After the mesh is enriched, a new calculation is made, converging the solution to a steady, and hopefully more accurate solution. This process of refining the grid and converging the solution on the new grid is repeated in an automated fashion, a set number of times, until a given level of refinement is achieved.

The refinement criteria and grading procedure used here are based upon that presented in [14,15]. Unless otherwise noted, no changes to the form of the refinement criteria or the selection levels are made. The procedure of refinement and coarsening of the cells is based upon a statistical description of the cell size weighted velocity divergence and curl. The local velocity divergence is used to detect compressive phenomena, while the velocity curl is used to detect shear. Each of these is weighted by the local cell size so that smaller cells contribute less to the overall weighting, as suggested in [32].

III.b VALIDATION OF THE EULER SOLVER

The Cartesian, cell-based approach has been demonstrated extensively for a wide variety of inviscid flows [11,8,9,16,15,19,22,23] and has been assessed for accuracy [13]. In [13] the approach was shown to compute transonic flows with a global second-order accuracy and a local accuracy between first- and second-order. The adaptively refined solutions about a transonic single-component airfoil and a sub-critical, multi-component airfoil are shown here, demonstrating the adaptive-mesh and mesh generation capability.

III.c AGARD CASE 06: TRANSONIC FLOW PAST A RAE 2822 AIRFOIL

This case corresponds to the same geometry and free-stream conditions as Test Case 06 in the collection of inviscid flow test cases[1]. Adaptive mesh refinement is performed for four levels beyond the base grid level. Figure 2 shows the final, adapted grid and Figure 3 shows the Mach number contours at the final refinement level.The solution compares well with the computed results, tabulated in [1]. Figure 4 shows the computed surface Mach number and the computed results from [1].

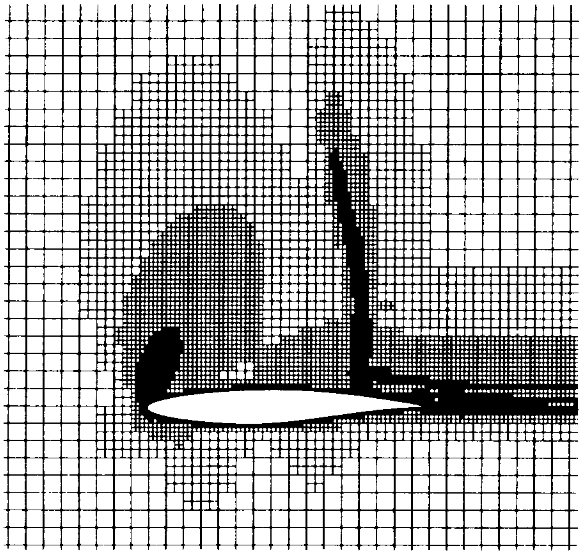

Figure 2 RAE 2822 Level 4 Adapted Grid

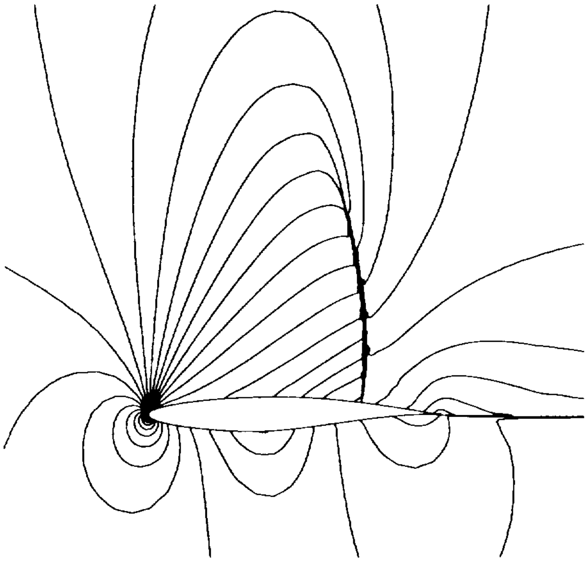

Figure 3 RAE 2822 Level 4 Adapted Mach Number Contours

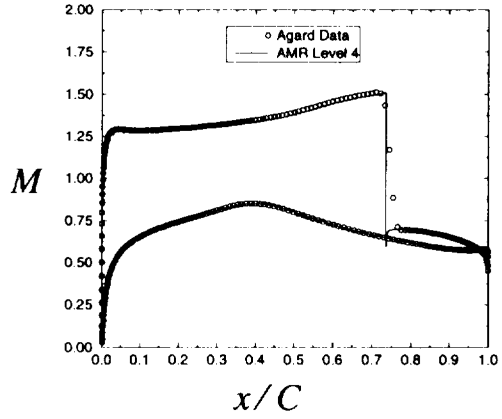

Figure 4 Comparison to Computed Surface Mach Numbers from AGARD 211

III.d SUDDHOO-HALL MULTI-ELEMENT AIRFOIL

This four-element airfoil has been included in a series of test cases at the 1994, ICASE/NASA LaRC Workshop on Adaptive Grid Methods. The geometry corresponds to that obtained by a conformal mapping technique [30], which has also yielded surface pressure data. The airfoil geometry has been curve fit using a cubic spline and made available on Mosaic (http://www.icase.edu/workshops/adapt) by the workshop organizers. The free-stream conditions corresponding to this case are $M_\infty = 0.2$ and $\alpha = 0°$. The mesh is generated automatically and adaptive-mesh refinement performed for 3 levels beyond the base grid. The computed solution on the final grid is compared to the potential flow solution in Figure 7 while the final grid and pressure contours are shown in Figure 5 and Figure 6.

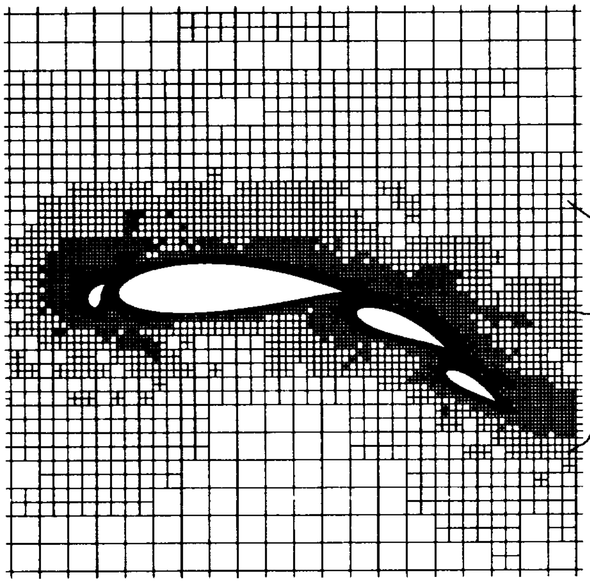

Figure 5 Level 3 Adapted Grid

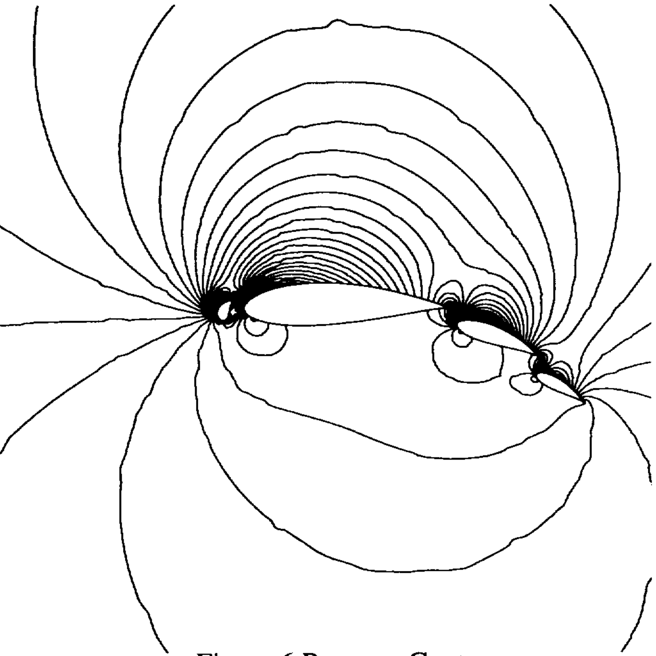

Figure 6 Pressure Contours

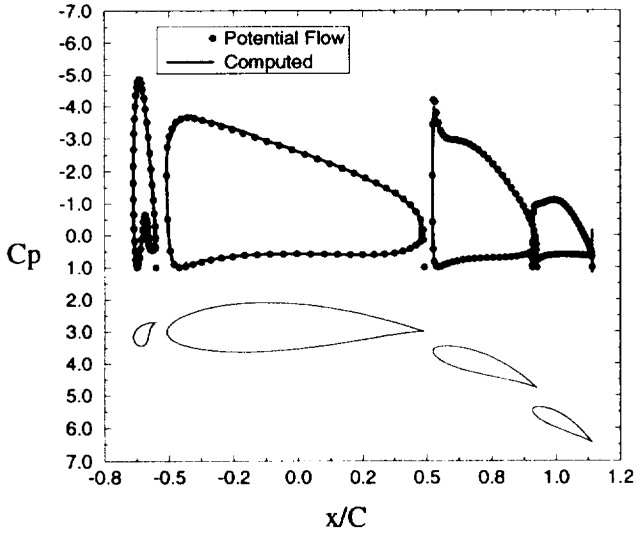

Figure 7 Computed Pressure Coefficients

IV. VISCOUS FLUX FORMULATION

The essence of the cell-centered viscous flux formulation is the reconstruction of the gradients of the velocity and temperature at the cell interfaces from the cell-averaged data of the cells around it. Once this has been completed, a numerical quadrature is performed over the cell faces, yielding the desired viscous fluxes through the interfaces. For cell-centered schemes, there are prevalently two separate classes of the viscous gradient reconstructions, which have been analyzed in [12]. The first and most widely used class is based upon an application of the divergence theorem to a co-volume surrounding the face where the flux is desired. These types of reconstructions are classified as Green-Gauss type reconstructions. Four of these types have been analyzed in [12], and are delineated amongst themselves by the reconstruction co-volumes and the procedures used to obtain the data at the vertices of the co-volumes. A different class of gradient reconstructions based upon expanding a polynomial about the face midpoint and then differentiating the polynomial to obtain the gradients has been proposed by Mitchell [24]. Linear and quadratic reconstructing polynomials are found using a Lagrangian type of interpolation, and are also examined in detail in [12].

There, the six schemes were analyzed for quality and positivity by local Taylor-series expansions of the stencils created for Laplace's equation on grids representative of the Cartesian approach. It was shown the importance of the K-exactness of the interface gradient reconstructions. The analysis showed that for arbitrarily distorted grids, the only means to obtain a conservative, first-order accurate discretization requires that the gradients be found from quadratically-preserving functions. Importantly, it was shown that if the gradient reconstruction procedure is not at least linearity-preserving, stencils can be obtained that preclude grid convergence, and will actually have an error that *increases* with mesh refinement. This type of truncation error is termed here as being mesh divergent. Very importantly, robustness of all of the schemes upon the distorted meshes caused by mesh refinement was gauged using a positivity analysis. Non-positivity of a reconstruction scheme can inhibit convergence and violates the discrete interpretation of the continuous maximum principle entertained by elliptic solutions.

To summarize the analytical results presented in [12], of the four Green-Gauss type reconstructions analyzed, three were shown to be either mesh divergent, produced either extremely non-positive stencils or gave solutions that on certain topologies yielded stencils that were completely or partially decoupled from the neighboring cells. The fourth reconstruction type is commonly known as a diamond-path reconstruction using a linearity-preserving weighting. This reconstruction is named so because it uses a diamond-shaped polygon to perform the gradient reconstruction. The four vertices of this polygon are formed by the two cell centroids that share the face to be reconstructed about, and the two vertices that are at the face endpoints. The data at the centroids is known exactly, but the data at the vertices must be arrived at by some interpolating procedure. In [20] a linearity-preserving weighting is derived which finds data at a vertex using the local cells about it. Since the Green-Gauss reconstruction procedure can only preserve linear functions, and can only do this if the data supplied to it is linearity-preserving, this weighting procedure guarantees the reconstruction of a gradient that is linearly K-exact. A means of obtaining higher-order preserving weightings was shown in [12], but for the linearity-preserving reconstruction scheme using the divergence integral, this higher-order weighting is unnecessary. Although the analysis showed that this reconstruction procedure was not positive on general meshes and could yield inconsistent stencils, it was seen to represent the better of the Green-Gauss class of reconstructions.

Of the polynomial reconstruction types, a Lagrangian type of interpolation procedure is used to reconstruct a polynomial based at the face midpoint, which is then differentiated to obtain the gradients. The reconstruction requires the selection of either three cells (for the linear reconstruction) or six cells (for the quadratic), which then necessitates the inversion of a Vandermonde-type matrix. In [24] it is suggested to base the selection of these support cells to provide a centroid of the resulting polygon to be closest to the face midpoint. In [12] it was shown that this criterion does not always yield the best stencil, and for the linear reconstruction scheme, yielded some extremely poor stencils on certain grid topologies. The analysis showed that the quadratic reconstruction was the only procedure that gave first-order accurate stencils for

Laplace's equation on arbitrarily distorted meshes. The analysis also indicated that this procedure could give the most non-positive stencils.

Since the analysis presented no clear choice as to the best procedure of the two classes of reconstructions, Green-Gauss or polynomial based, the two schemes were both used to compute a series of low to moderate Reynolds number, adaptively-refined solutions using the Cartesian approach. Although neither scheme guaranteed positive stencils, the diamond-path scheme could be viewed as representing the more positive, yet less accurate of the two. In [12], a discrete accuracy and positivity analysis on the grids showed that the inconsistency incurred by the diamond-path schemes is low while the computed results from both schemes were nearly identical. Globally this inconsistency incurred by the diamond-path scheme is small due in a large part by the regularity of the Cartesian grids. The quadratic polynomial based scheme guaranteed consistency, which was also shown computationally in the discrete accuracy analysis, but yielded the most non-positive stencils, which also made the quadratic scheme the least robust. For arbitrarily cut cells, neither scheme was as robust as would be desired, but the diamond-path scheme was shown to be the more positive. An important conclusion from this study is that the current viscous flux functions for cell centered schemes rely heavily upon grid smoothness and orthogonality to obtain accuracy and positivity. For the Cartesian grids, these properties do not hold at refinement boundaries, and are extremely violated near cut cells. This has negative implications for the smoothness of aerodynamic parameters that rely upon derivative quantities at walls, such as skin friction and heat transfer, and can have a detrimental effect upon con vergence. Regardless of these comparatively negative findings, the approach can still prove to be useful, and can give accurate, automatically gridded and adaptively-refined solutions of the Navier-Stokes equations for low and moderate Reynolds number flows. The following computations illustrate adaptively refined solutions using the Cartesian-cell approach with the diamond path, linearity-preserving viscous flux function.

IV.a LAMINAR DRIVEN CAVITY FLOW

The laminar flow inside a square, driven cavity is computed and compared to the computed results of Ghia, et. al. [18]. In [18], an incompressible formulation of the Navier-Stokes equations were solved using an implicit multi-grid method, where tabulated u- and v-velocity data is supplied along the lines through the geometric center of the cavity. To compare with these incompressible results, the Mach number used here is taken to be $M_{lid} = 0.1$. Two Reynolds numbers were computed and compared to the tabulated results.

IVa.1 Re=100

A uniform base grid of 1024 cells (32 by 32) is generated, and three levels of adaptive mesh refinement beyond the base grid are obtained. Adaptive mesh refinement improves the solution slightly, but the initial solution is quite good. Figure 8 and Figure 9 show the computed u- and v-velocity profiles along vertical and horizontal lines through the geometric center of the cavity for the diamond path scheme. Figure 10 shows the final adapted grid and Figure 11 shows contours of u-velocity. Particle paths, showing the primary and secondary vortices is shown in Figure 12.

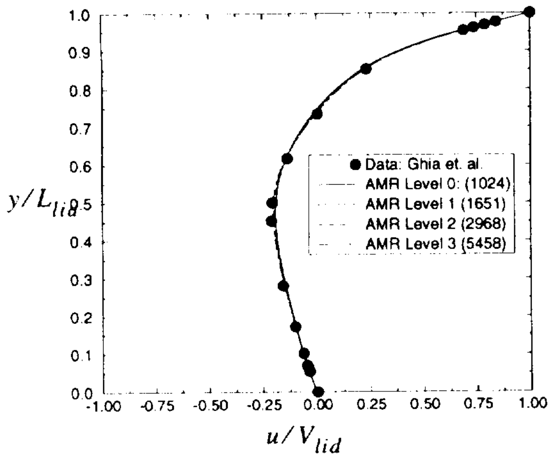

Figure 8 u-velocity Along Vertical Line Through Geometric Center

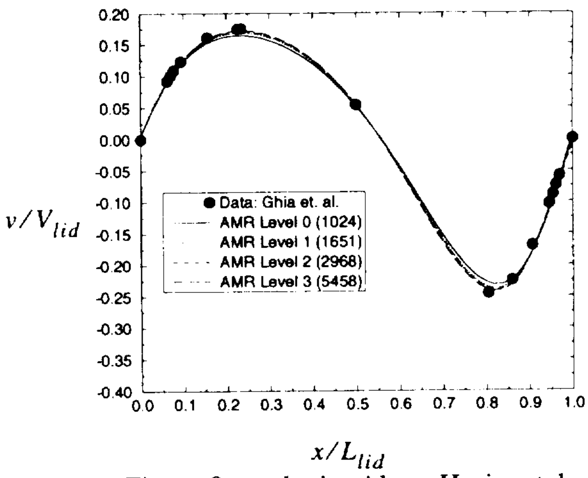

Figure 9 v-velocity Along Horizontal Line Through Geometric Center

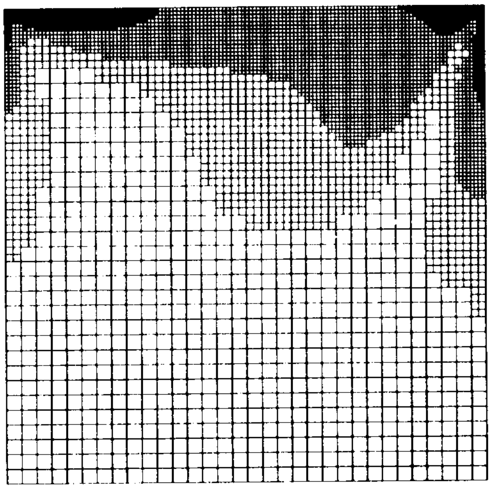

Figure 10 Refinement Level 3 Adapted Grid, Re=100 Case

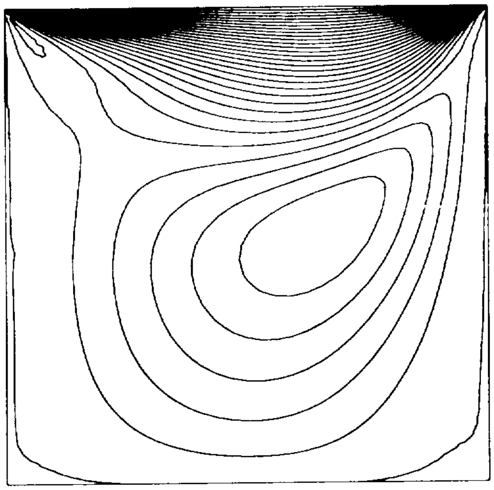

Figure 11 Refinement Level 3, u-velocity Contours

As seen in the adapted grid, the refinement strategy has added points near the lid and has resolved the grid near the upper corners, where there are singularities in the u-velocity. There are secondary vortices situated in the lower corners of the cavity: These vortices are not isolated by the refinement strategy, which indicates that a better strategy more suited for viscous flows might be needed. Overall, though, the solution is predicted well.

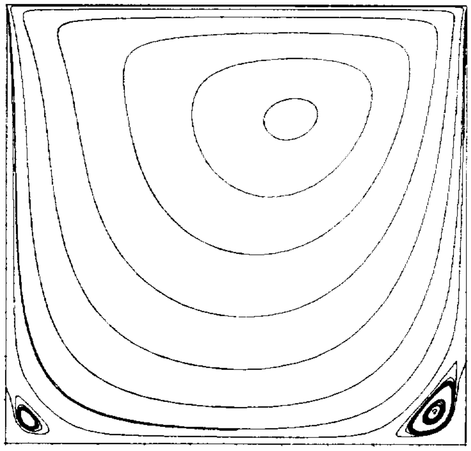

Figure 12 Particle Paths:

IVa.2 Re=400

This case is similar to the previous, although the Reynolds number is now 400. A coarse base grid is generated, and three levels of adaptive mesh refinement are performed beyond the base level. A comparison to the computational data of Ghia[18] is shown for the u- and v-velocities on lines through the geometric center in Figure 13 to Figure 14.

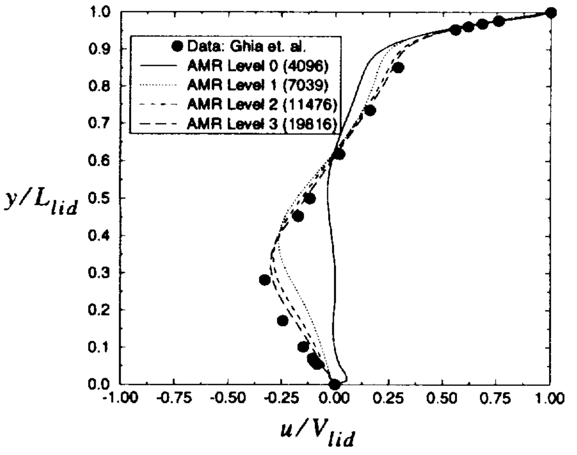

Figure 13 u-velocities on Vertical Line Through Geometric Center, Re=400

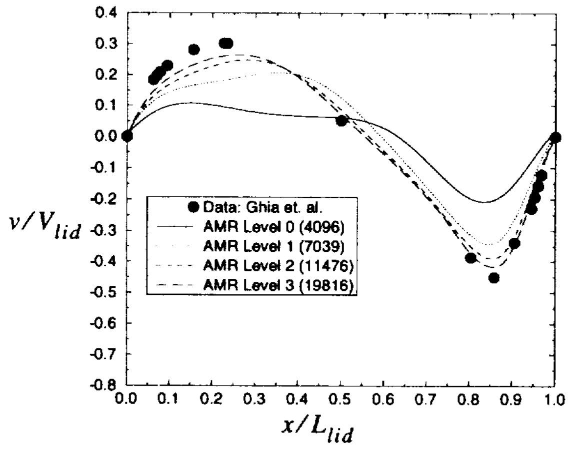

Figure 14 v-velocities on Horizontal Line Through Geometric Center, Re=400

Here, the solution on the coarse, base grid is poor, although the solution is improved through the adaptive mesh refinement. Figure 15, Figure 16 and Figure 17 show the adapted grid, u-velocity contours and particle paths in the cavity.

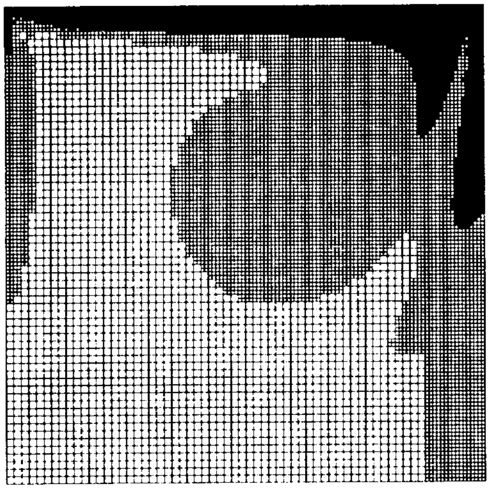

Figure 15 Final Adapted Grid, Re=400 Driven Cavity

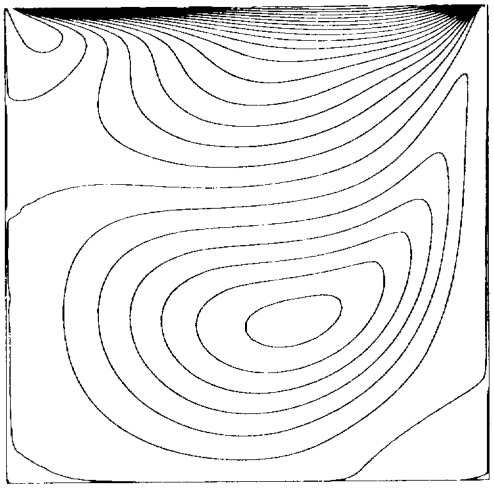

Figure 16 u-velocity Contours, Final Adapted

The solution obtained is good on the final grids, and the mesh refinement is shown to have improved the solution, even though the refinement strategy has not been derived with viscous flows in mind.

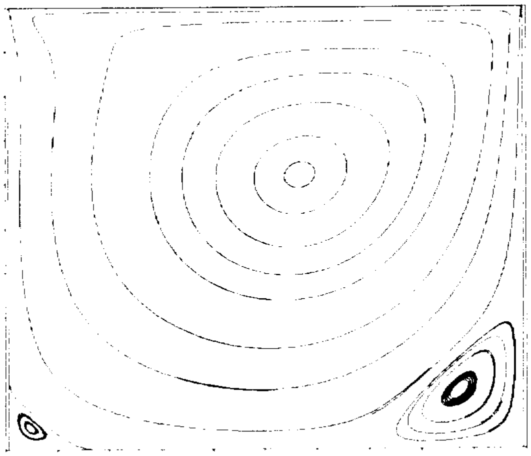

Figure 17 Particle Paths, Re=400 Driven Cavity

IV.b LAMINAR FLOW OVER A BACKWARD FACING STEP

The laminar flow over a backwards facing step at two Reynolds numbers is used to validate the solver. The computed results are compared to the experimental data of [2] at the laminar Reynolds numbers. A parabolic velocity profile is specified at the inflow, and the exit pressure is specified. This ensures that the proper pressure gradient and mass flow is imposed on the flow.

IVb.1 Re=100

Adaptive mesh refinement is made for three levels of refinement beyond the coarse, base grid. Figure 18 shows the grid and Figure 19 shows the effect of adaptive mesh refinement at a location corresponding to 2.55 step heights downstream of the step. Comparisons are made at other locations of the flow in [12]: The results compare equally well, and are not shown here. The agreement with the experimental data is good, and the adaptive mesh refinement improves the solution quality with each refinement.

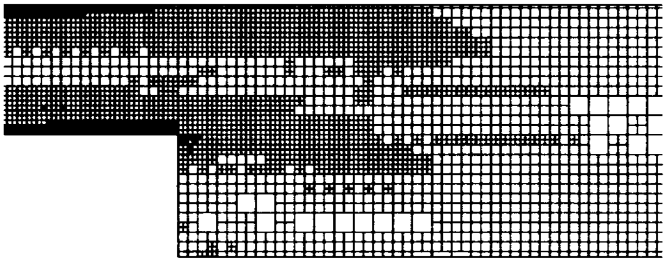

Figure 18 Portion of Adapted Grid at Refinement Level 3: Close-up Near Step

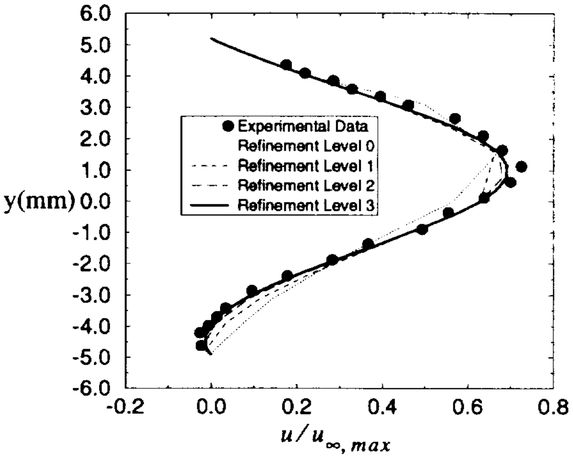

Figure 19 Comparison of Adapted Solutions at x/S=2.55. Re=100

IVb.2 Re=389

This case is identical to the previous case, but the Reynolds number is Re=389. The computed velocity profiles through refinement at x/S=2.55 are shown in Figure 20. As before, the adaptive mesh refinement automatically improves the solution quality, and the final refinement level solution compares well with experiment.

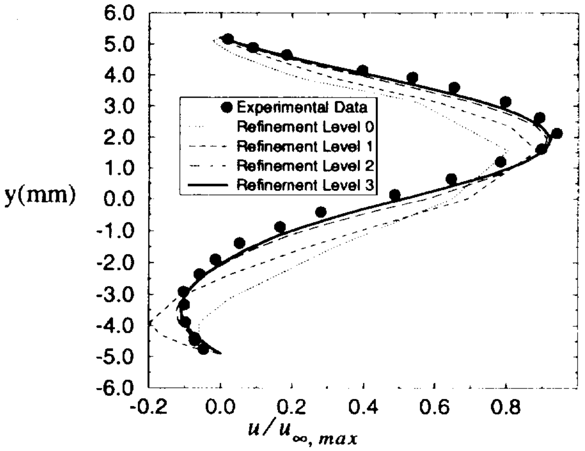

Figure 20 Comparison of Adapted Solutions at X/S=2.55. Re=389

IV.c DEVELOPING LAMINAR FLOW OVER A FLAT PLATE

The developing, laminar flow over a flat plate which is aligned with the free-stream is computed and compared to the theoretical solution. A uniform flow is imposed ahead of the plate, and the flow is allowed to develop along the plate. Conditions are set so that the Reynolds number based on plate length is 10,000. The Mach number is taken to be $M_\infty = 0.2$ which eliminates the need for any compressibility transformation to compare to theory. When a poorly refined base grid is specified, the coarser solutions are improved successively by the adaptive mesh refinement. Figure 21 shows the close up of a base grid near the leading edge where the initial resolution of the grid is made according to an estimated normal velocity scale variation deduced from theory. Figure 22 and Figure 23 show the effect of adaptive mesh refinement upon the u- and v-velocity profiles while Figure 24 shows the skin friction through refinement.

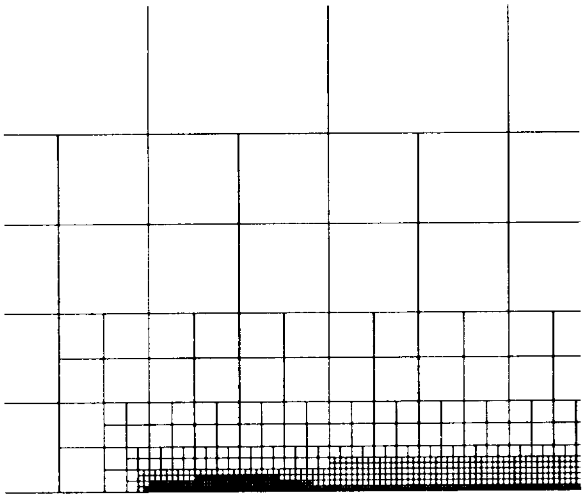

Figure 21 Close-up of Base Level Grid: Length Scale Smoothing: $\Delta u = 0.2$

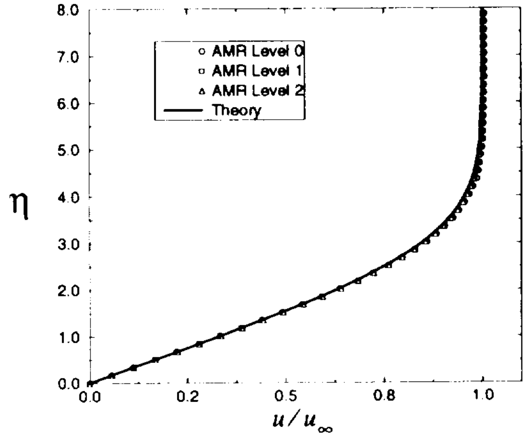

Figure 22 u-velocity Profiles: Effect of Adaptive Mesh Refinement at $Re_x = 8000$, Diamond Path Reconstruction.

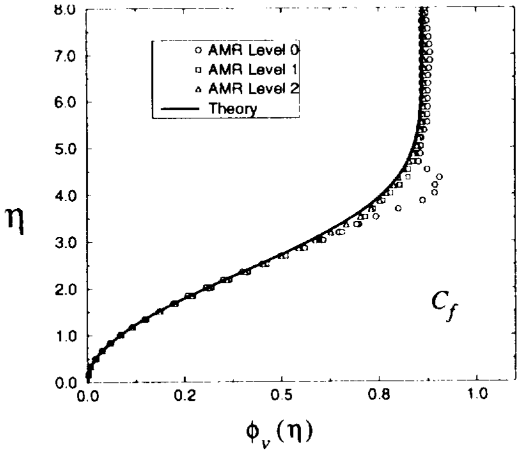

Figure 23 v-velocity Profiles: Effect of Adaptive Mesh Refinement at $Re_x = 8000$, Diamond Path

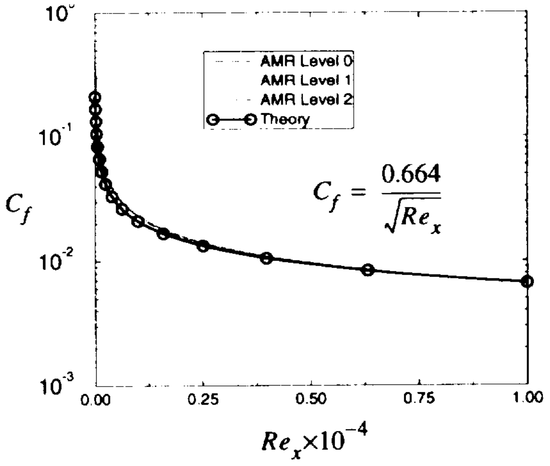

Figure 24 Skin Friction Through Adaptive Mesh Refinement, Diamond Path Scheme.

The smoothness of the skin friction in Figure 24 is due to the smoothness of the grid, since the root cell of the grid system has been located so that no cut cells are introduced along the surface of the flat plate. In [12], the plate is rotated 30° about the base axes and the identical flow is computed, bringing to light the effect of introducing cut cells along the plate boundary. The mean flow quantities are predicted well, but the skin friction is shown to be very oscillatory. This appears to be unavoidable with the current state of the art of the viscous flux functions. There is no flux function that guarantees both positivity and accuracy for cell centered schemes. Regardless of this finding, the mean flow quantities are predicted well, and the approach can be used to compute adaptively-refined solutions of the Navier-Stokes equations about complex geometries, as is indicated in the next case.

IV.d LAMINAR FLOW THROUGH A BRANCHED DUCT WITH COOLING FINS

To demonstrate the approach for complex geometries, the flow in a stylized duct is computed. This duct geometry corresponds to an experiment conducted at NASA LeRC designed to simulate, in a simplified manner, the flow in the cooling passages of a turbine blade [28]. The calculations shown here in no way try to simulate the experiment: The experimental conditions correspond to a turbulent flow, while the calculations shown here are laminar. A schematic of the geometry and flow is shown in Figure 25.

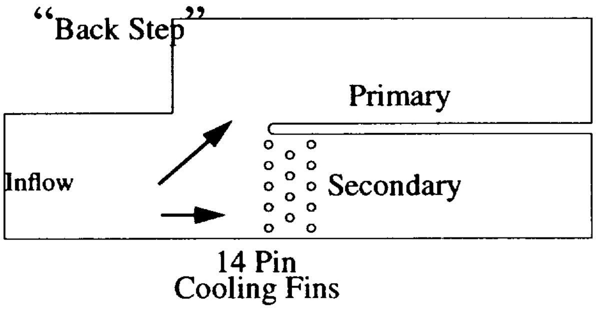

Figure 25 Schematic of Branched Duct Geometry

A fully developed profile is introduced at the inflow, and the flow is diverted into the primary passage by the blockage introduced by the pin fins in the secondary passage. Two different Reynolds numbers based on pin diameter and maximum velocity in the fully developed inflow profile were computed in [12]. The lower Reynolds number results are only shown here for brevity, where the Reynolds number based on maximum inflow velocity and pin fin diameter is Re=25. Only one level of adaptive-mesh refinement beyond the base grid level was obtained, due to positivity problems in the rear stagnation region of one of the pin fins. The final adapted grid and contours of total velocity are shown Figure 26 and Figure 27.

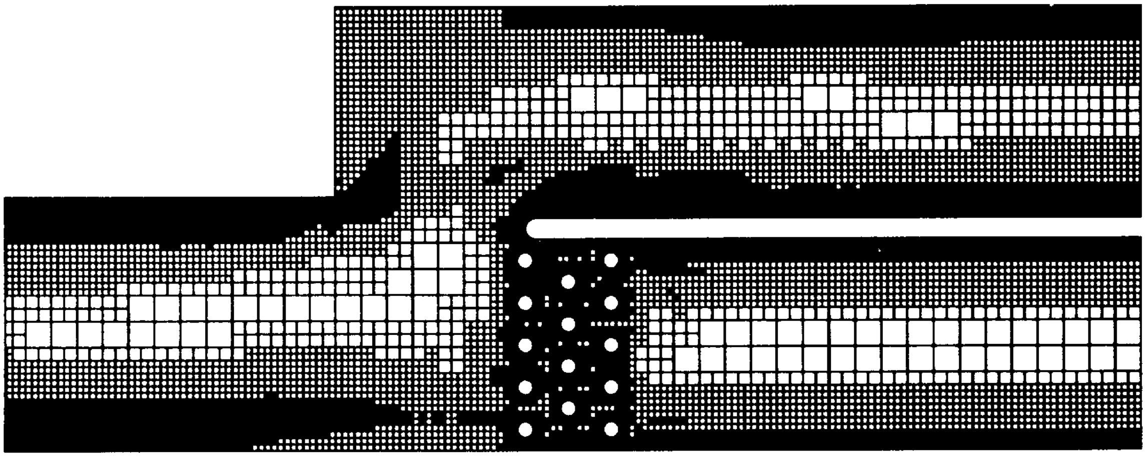

Figure 26 Close up of Final Adapted Grid, Lower Reynolds Number Case

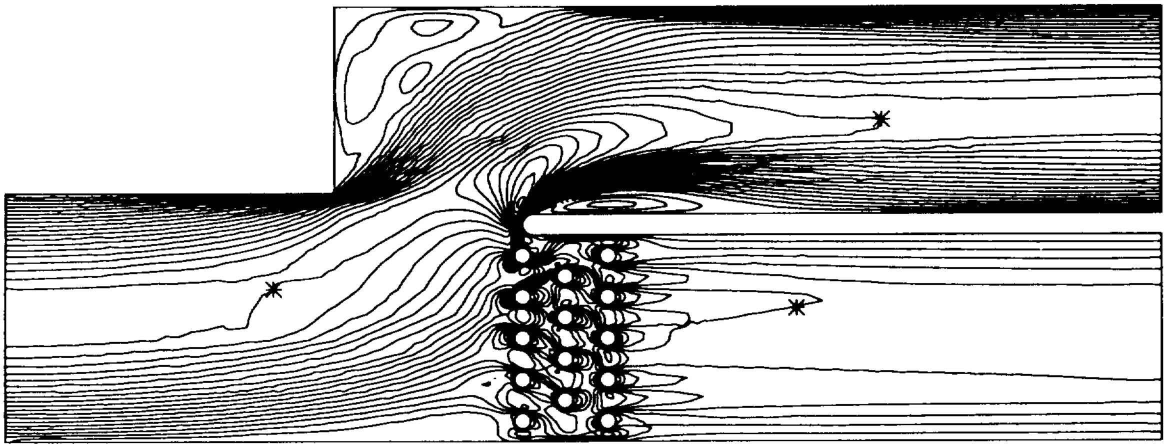

Figure 27 Close up of Final Grid Level Total-velocity Contours.

The basic flow features predicted here correspond to those in the experiment, although some important features are grossly under-resolved, such as the individual pin-fin wakes. The primary passage separation and reattachment along the splitter plate and the separation anchored at the back step portion are both properly predicted, as well as the upstream influence of the pin blockage upon the lower wall flow. Although many levels of refinement were not achieved, the larger scale flow features were adequately predicted and were improved by the mesh refinement procedure.

V. CONCLUSIONS

An adaptively-refined, finite-volume solution procedure for the Euler and Navier-Stokes equations using a Cartesian, cell-based approach has been presented, demonstrated and validated. The Cartesian, cell-based grid generation procedure is automated, and is able to generate grids about complicated geometries without user intervention. The grid generation strategy is based upon the recursive, isotropic subdivision of a Cardomain, polygonal cells are "cut" out of the background Cartesian mesh using a highly modified polygon clipping algorithm. The hierarchy of the grid generation process is stored in a binary tree, which provides a natural means of finding cell-to-cell connectivity (via logical tree traversals) and provides a straightforward means of adaptively refining the grid (via tree branch growth and pruning). Extension of this approach to the Navier-Stokes equations shows promise by providing a means of obtaining automated, adaptively-refined solutions upon domains where the grid has also been automatically generated.

A finite-volume, upwind-based scheme has been selected for treatment of the convective terms in the Euler/Navier-Stokes equations and has been implemented in the Cartesian, cell-based framework. Extension of the Cartesian approach for solving the Navier-Stokes equations has necessitated a careful investigation of candidate viscous flux formulations, which has been summarized here. The two viscous flux formulations that were investigated represent, respectively, a divergence theorem (Green-Gauss) based reconstruction procedure, and a quadratic polynomial based reconstruction procedure. Analysis and practice indicated that the Green-Gauss based procedure, where data at the face vertices was found in a linearity-preserving manner, was adequate and was also the more robust of the two, although neither could guarantee positivity on arbitrarily distorted meshes. This non-positivity reduced the robustness of the solver, although useful results can still often be obtained. The Green-Gauss based scheme, commonly referred to as the diamond-path reconstruction using a linearity-preserving weighting, has been used here to compute adaptively-refined solutions to a variety of low and moderate Reynolds number flows. Comparisons were made to accepted computational results, to experimental data and to theory for a range of flows, where the Cartesian, cell-based approach is shown to accurately predict these flows.

ACKNOWLEDGEMENTS

For the multi-element airfoil, the authors gratefully acknowledge the ICASE workshop organizers for the geometry, Dr. Kyle Anderson of NASA LaRC for the Suddhoo-Hall reference and Dr. Darren DeZeeuw of The University of Michigan for the electronic form of the actual pressure coefficient data. The geometry for the coolant passage and the branched duct was graciously supplied by Dr. Erlendur Steinthorsson of ICOMP. The first author gratefully acknowledges the discussions with Dr. P.C.E. Jorgenson and Mr. C.J. Steffen, Jr..

REFERENCES

[1] AGARD Subcommittee C. Test Cases for Inviscid Flow Field Methods. AGARD Advisory Report 211, 1986.

[2] B.F. Armaly, F. Durst, J.C.F. Pereira, and B. Schonung. Experimental and Theoretical Investigation of Backward-Facing Step Flow. *Journal of Fluid Mechanics*, 127:473–496, 1983.

[3] T. J. Barth and P. O. Frederickson. Higher Order Solution of the Euler Equations on Unstructured Grids Using Quadratic Reconstruction. AIAA Paper 90-0013, 1990.

[4] T.J. Barth and D.C. Jespersen. The Design and Application of Upwind Schemes on Unstructured Meshes. AIAA Paper 89-0366, 1989.

[5] S.A. Bayyuk, K.G. Powell, and B. van Leer. An Algorithm For the Simulation of 2-D Unsteady Inviscid Flows Around Arbitrarily Moving and Deforming Bodies of Arbitrary Geometry. AIAA 11th Computational Fluid Dynamics Conference Proceedings, 1993.

[6] M. J. Berger and P. Colella. Local Adaptive Mesh Refinement for Shock Hydrodynamics. Technical Report UCRL-97196, Lawrence Livermore National Laboratory, 1987.

[7] M.J. Berger. Adaptive mesh refinement for hyperbolic partial differential equations. *Journal of Computational Physics*, 53:484–512, 1984.

[8] M.J. Berger and R.J. LeVeque. An Adaptive Cartesian Mesh Algorithm for the Euler Equations in

Arbitrary Geometries. AIAA Paper 89-1930-CP, 1989.

[9] M.J. Berger and R.J. LeVeque. A Rotated Difference Scheme for Cartesian Grids in Complex Geometries. *Computing Systems in Engineering*, 1:305–311, 1990.

[10] Y.L. Chiang. *Simulation of Unsteady Inviscid Flow On An Adaptively Refined Cartesian Grid.* Ph.D. thesis, The University of Michigan, Department of Aerospace Engineering, 1992.

[11] D. K. Clarke, M. D. Salas, and H. A. Hassan. Euler Calculations for Multielement Airfoils Using Cartesian Grids. *AIAA Journal*, 24, 1986.

[12] W.J. Coirier. *An Adaptively-Refined, Cartesian, Cell-Based Scheme for the Euler and Navier-Stokes Equations*. Ph.D. thesis, The University of Michigan, Department Aerospace Engineering. Also published as NASA TM 106754, 1994.

[13] W.J. Coirier and K.G. Powell. An Accuracy Assessment of Cartesian-Mesh Approaches for the Euler Equations. AIAA Paper 93-3335-CP, 1993.

[14] D. DeZeeuw and K. G. Powell. An Adaptively-Refined Cartesian Mesh Solver for the Euler Equations. AIAA 10th Computational Fluid Dynamics Conference Proceedings, 1991.

[15] D. DeZeeuw and K.G. Powell. Euler Calculations of Axisymmetric Under-Expanded Jets by an Adaptive-Refinement Method. AIAA Paper 92-0321, 1992.

[16] D. DeZeeuw and K.G. Powell. An Adaptively Refined Cartesian Mesh Solver for the Euler Equations. *Journal of Computational Physics*, 104(1):56–58, 1993.

[17] D.L. DeZeeuw. *A Quadtree-Based Adaptively-Refined Cartesian-Grid Algorithm for Solution of the Euler Equations*. Ph.D. thesis, The University of Michigan, Department of Aerospace Engineering, 1993.

[18] U. Ghia, K.N. Ghia, and C.T. Shin. High-Re Solutions for Incompressible Flow Using the Navier-Stokes Equations and a Multigrid Method. *Journal of Computational Physics*, 48:387–411, 1982.

[19] C. Gooch and H. Oksuzoglu. Extension of State-Vector Splitting to the Navier-Stokes Equations. AIAA Paper 93-3374-CP, 1993.

[20] D.G. Holmes and S.D. Connell. Solution of the 2D Navier-Stokes Equations on Unstructured Adaptive Grids. AIAA Paper 89-1932-CP, 1989.

[21] M.S. Liou and C.J.Steffen, Jr. A New Flux Splitting Scheme. *Journal of Computational Physics*, 107:23–39, 1993.

[22] J. Melton, F. Enomoto, and M. Berger. Three-Dimensional Automatic Cartesian Grid Generation for Euler Flows. AIAA Paper 93-3386-CP, 1993.

[23] J.E. Melton, M.J. Berger, M.A. Aftosmis, and M.D. Wong. 3D Applications of a Cartesian Grid Euler Method. AIAA 33rd Aerospace Science Meeting, AIAA Paper 95-0853, 1995.

[24] C.R. Mitchell and R.W. Walters. K-Exact Reconstruction for the Navier-Stokes Equations on Arbitrary Grids. AIAA Paper 93-0536, 1993.

[25] R. Pember, J. Bell, P. Colella, W. Crutchfield, and M. Welcome. Adaptive Cartesian Grid Methods for Representing Geometry in Inviscid Compressible Flow. AIAA Paper 93-3385-CP, 1993.

[26] J.J. Quirk. *An Adaptive Grid Algorithm for Computational Shock Hydrodynamics*. Ph.D. thesis, Cranfield Institute of Technology, College of Aeronautics, 1991.

[27] J.J. Quirk. An Alternative to Unstructured Grids for Computing Gas Dynamic Flows Around Arbitrarily Complex Two-Dimensional Bodies. ICASE Report 92-7, 1992.

[28] L.M. Russell, D.R. Thurman, P.S. Simoni, S.A. Hippensteele, and P.E. Poinsatte. Measurements and Computational Analysis of Heat Transfer and Flow in a Simulated Turbine Blade Internal Cooling Passage. AIAA Paper AIAA-93-1797, 1993.

[29] P.H. Snynder and R.J.Roelke. The Design of an Air Cooled Metallic High Temperature Radial Turbine. AIAA Paper 88-2872, 1988.

[30] A. Suddhoo and I.M. Hall. Test Cases for the Plane Potential Flow past Multi-Element Aerofoils. *Aeronautical Journal*, pages 403–414, 1985.

[31] I.E. Sutherland and G.W. Hodgman. Reentrant Polygon Clipping. *Communications of the ACM, Graphics and Image Processing*, 17(1):32–42, 1974.

[32] G. Warren, W. K. Anderson, J. Thomas, and S. Krist. Grid Convergence for Adaptive Methods. AIAA 10th Computational Fluid Dynamics Conference Proceedings, 1991.

[33] D. P. Young, R. G. Melvin, M. B. Bieterman, and J. E. Bussoletti. A Locally Refined Rectangular Grid Finite Element Method: Application to Computational Fluid Dynamics and Computational Physics. *Journal of Computational Physics*, 92:1–66, 1991.