(2)

SECU

AD-A243 977

IT DOCUMENTATION PAGE

1a. F UNCLASSIFIED	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution unlimited.
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION PEDA Corporation	6b. OFFICE SYMBOL (If applicable)
6c. ADDRESS (City, State and ZIP Code) 4151 Middlefield Road, Suite 107 Palo Alto, CA 94303	7a. NAME OF MONITORING ORGANIZATION Air Force Office of Scientific Research 7b. ADDRESS (City, State and ZIP Code) Directorate of Mathematical & Information Sciences, Bolling AFB DC 20332-6448
8a. NAME OF FUNDING/SPONSORING ORGANIZATION AFOSR	8b. OFFICE SYMBOL (If applicable) NM/NA
8c. ADDRESS (City, State and ZIP Code) Bolling AFB DC 20332-6448	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER F49620-85-C-0081 10 SOURCE OF FUNDING NOS. PROGRAM ELEMENT NO. 61102F; PROJECT NO. 2304; TASK NO. A1; WORK UNIT NO.

11. TITLE (Include Security Classification)
Accurate, Productive Aerodynamic Simulation on Patched Mesh Systems

12. PERSONAL AUTHOR(S)
Charles Lombard

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Yr., Mo., Day)	15. PAGE COUNT
Final	FROM 86 10 01 TO 91 09 30	91 10 18	50

16. SUPPLEMENTARY NOTATION

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB. GR.	CSCM, Upwind Methods, Navier-Stokes, Turbulence Modeling, Multiple Grids, Data Structures, Object-Based, Algebraic Grid Generation, Chemically Reacting Flow

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

In the fifth and final year of the program the research has completed defining data structures, object-based programming style, and tools for a new flexible approach to scientific programming and problem solving. Problems of program complexity associated with changing models and physics as well as with joined and disjoint multiple independent patched mesh domain decompositions for treating complex geometries and resolving captured flow structures can be systematically organized within the context of the directed graph programming concept being explored. Problems and parts of problems having geometric connectivity or its analogs such as association, hierarchy or precedence relationships are naturally exhibited and easily debugged or modified in the graph. The solution of problems is literally to traverse the graphs. For the emerging prototype aerodynamic simulation facility, the graphs which are to control grid generation, Navier-Stokes solution

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT	21. ABSTRACT SECURITY CLASSIFICATION
UNCLASSIFIED/UNLIMITED ☒ SAME AS RPT ☐ DTIC USERS ☐	UNCLASSIFIED

22a. NAME OF RESPONSIBLE INDIVIDUAL	22b TELEPHONE NUMBER (Include Area Code)	22c OFFICE SYMBOL
Dr. Len Sakell	(202) 767-4935	NM/NA

DD FORM 1473, 83 APR EDITION OF 1 JAN 73 IS OBSOLETE. UNCLASSIFIED

SECURITY CLASSIFICATION OF T· PAGE

19. Abstract Cont'd.

procedures, and scientific graphics are to be constructed with a graphical editor hosted in high performance graphics workstations.

The efficient global data structure for the system is a set of large linear arrays in which the data and parameterization associated with the independent quadrilateral blocks of mesh are sequentially stacked. The directed graph is to control procedures that point to and operate on the data structure. For physical and external domain boundaries, we implement both upwind characteristics based implicit computed boundary point procedures and, on internal patch boundaries, frozen boundary data that is obtained from bilinear interpolation of the solution on adjoining overlapping meshes. The very flexible approach admits segmented grids with steps and holes, relatively moving bodies and grid systems, zonal methods expressing different combinations of physics over the computational domain, and boundary conditions that are nonhomogeneous by type.

Additions and improvements to our upwind computational algorithms and tools considered over the three year program have included a consistent strongly conservative finite volume formulation with numerical flux satisfying our earlier TVD or more recent essentially non oscillatory (ENO) schemes. These differencing schemes improve the nonlinear stability of second order methods while maintaining near order of accuracy throughout. Consistent relative accuracy across disparate length scales of problems is achieved first through employing multiple independent grids fitting the local scales and topologies of elements of a problem, and second by a simplified 3-D solution adaptive mesh point redistribution procedure. To support the requirements of multiple patch mesh generation in nonsimple geometry, we have explored algebraic techniques including methods for the algebraic construction of smooth grid on 3-D surfaces of arbitrary analytical description. The combined methodology is demonstrated, with excellent resolution and practical amounts of computation – for the CSCM compressible Navier-Stokes algorithm solved on a Stardent high performance workstation in application to 3 - D multiple jet interaction control flow fields on a complex (HEDI) missile geometry. The technology has also been applied to a 3 – D skewed compound-ramp inlet with growing sidewalls and is now confronting the multifaceted problem of a multiple slot injection cooled aero-optical cavity flow embedded in a partially transpiration cooled missile forebody subject to a hypersonic stream.

In the area of numerical approaches to supporting physics, a significant improvement in the stability and convergence of implicit methods coupled with a procedure for determining variable gas properties in equilibrium (or otherwise reacting or mixing) flow has been achieved with a new exponentially decreasing under-relaxation strategy. Finally, two theoretically sound new comparatively computationally efficient Reynolds stress models of anisotropic turbulent flow are well into development. The first of these is a Universal Two Equation Model jointly parameterized in rotation and strain and expressed in a thin layer algebraic Reynolds stress approximation. The second is a compressible extension of an older Large Eddy Interaction Model directly predicting unsteady turbulence with a stochastic component as well as a time averaged field. The latter formulation is believed substantially better capable to address important highly nonlinear computational aerodynamics problems such as turbulent combustion, unsteady aerodynamics and the important aero-optical distortion jitter produced by large scale turbulent disturbances.

Accurate, Productive Aerodynamic Simulation on Patched Mesh Systems

C.K. Lombard, Joseph Oliger, J.Y. Yang
S.K. Hong, William Codding, Jorge Bardina
E. Venkatapathy, Gustav Nystrom, Raymond Luh

Abstract

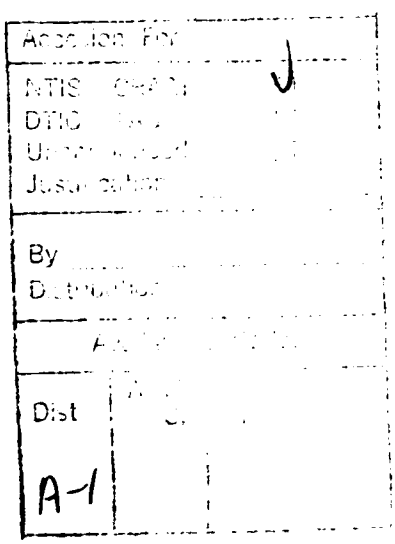
In the fifth and final year of the program the research has completed defining data structures, object-based programming style, and tools for a new flexible approach to scientific programming and problem solving. Problems of program complexity associated with changing models and physics as well as with joined and disjoint multiple independent patched mesh domain decompositions for treating complex geometries and resolving captured flow structures can be systematically organized within the context of the directed graph programming concept being explored. Problems and parts of problems having geometric connectivity or its analogs such as association, hierarchy or precedence relationships are naturally exhibited and easily debugged or modified in the graph. The solution of problems is literally to traverse the graphs. For the emerging prototype aerodynamic simulation facility, the graphs which are to control grid generation, Navier-Stokes solution procedures, and scientific graphics are to be constructed with a graphical editor hosted in high performance graphics workstations.

The efficient global data structure for the system is a set of large linear arrays in which the data and parameterization associated with the independent quadrilateral blocks of mesh are sequentially stacked. The directed graph is to control procedures that point to and operate on the data structure. For physical and external domain boundaries, we implement both upwind characteristics based implicit computed boundary point procedures and, on internal patch boundaries, frozen boundary data that is obtained from bilinear interpolation of the solution on adjoining overlapping meshes. The very flexible approach admits segmented grids with steps and holes, relatively moving bodies and grid systems, zonal methods expressing different combinations of physics over the computational domain, and boundary conditions that are nonhomogeneous by type.

Additions and improvements to our upwind computational algorithms and tools considered over the three year program have included a consistent strongly conservative finite volume formulation with numerical flux satisfying our earlier TVD or more recent essentially non oscillatory (ENO) schemes. These differencing schemes improve the nonlinear stability of second order methods while maintaining near order of accuracy throughout. Consistent relative accuracy across disparate length scales of problems is achieved first through employing multiple independent grids fitting the local scales and topologies of elements of a problem, and second by a simplified 3-D solution adaptive mesh point redistribution procedure. To support the requirements of multiple patch mesh generation in nonsimple geometry, we have explored algebraic techniques including methods for the algebraic construction of smooth grid on 3-D surfaces of arbitrary analytical description. The combined methodology is demonstrated, with excellent resolution and practical amounts

of computation – for the CSCM compressible Navier-Stokes algorithm solved on a Stardent high performance workstation in application to 3 - D multiple jet interaction control flow fields on a complex (HEDI) missile geometry. The technology has also been applied to a 3 – D skewed compound-ramp inlet with growing sidewalls and is now confronting the multifaceted problem of a multiple slot injection cooled aero-optical cavity flow embedded in a partially transpiration cooled missile forebody subject to a hypersonic stream.

In the area of numerical approaches to supporting physics, a significant improvement in the stability and convergence of implicit methods coupled with a procedure for determining variable gas properties in equilibrium (or otherwise reacting or mixing) flow has been achieved with a new exponentially decreasing under-relaxation strategy. Finally, two theoretically sound new comparatively computationally efficient Reynolds stress models of anisotropic turbulent flow are well into development. The first of these is a Universal Two Equation Model jointly parameterized in rotation and strain and expressed in a thin layer algebraic Reynolds stress approximation. The second is a compressible extension of an older Large Eddy Interaction Model directly predicting unsteady turbulence with a stochastic component as well as a time averaged field. The latter formulation is believed substantially better capable to address important highly nonlinear computational aerodynamics problems such as turbulent combustion, unsteady aerodynamics and the important aero-optical distortion jitter produced by large scale turbulent disturbances.



Accurate, Productive Aerodynamic Simulation on Patched Mesh Systems

1. Introduction

As major strides have been made in the past several years in the speed and memory of computers, similar advances have been achieved in details of computational technique for constructing finite difference meshes and for solving the compressible Euler or Navier – Stokes equations. The combined progress has been such that the CFD community is now challenged by the possibility of solving whole interacting flows of multicomponent systems. Topical examples are flows about aircraft and missiles, in turbomachines and rocket engines, and flows involving interactions with jets and wakes.

Such thinking is symptomatic of the fact that the research focus has shifted from 2 – D methods development and demonstrations to viable 3 – D computational design methodologies. An important facet of the new expectations and requirements for aerodynamic simulation is the development and incorporation of improved physical modeling and associated numerics such as for turbulence and chemically reacting flow. Such modeling and numerical simulation also opens the way to improved design possibilities for environments which tunnel testing cannot reproduce or effectively explore.

The shift poses two research challenges. The first is to bring a new generation of 3 – D algorithms to an effective fruition in terms of accuracy and computational efficiency. The second is to organize the problems of constructing meshes and solving the gasdynamic PDE's and/or supporting physics equations on geometrically complex domains in such a way as to minimize the work of problem setup and execution for each new geometry or mesh structure. This requires the development of computational systems to unify the many components of the simulation – principally grid generation, computation and graphics.

The present program contains both the above aspects in research aimed at substantially enhancing accuracy and productivity in the numerical simulation of aerodynamic flows in complex geometry. The program builds on and extends recent work in algebraic grid generation[1–3] for composite patched mesh systems and the CSCM upwind implicit algorithm[4–10] for solving the compressible Euler and Navier-Stokes equations. With the conservative upwind method we have successfully demonstrated[5,10] the capability to accurately capture weak reflecting shocks in compressible flow using solution adaptive grid refinement strategies with aligned overset mesh patches[5] like those proposed by Berger and Oliger[11] and with mesh point redistribution[10] as formulated by Nakahashi and Deiwert[12]. On conventional single grids, the implicit upwind method exhibits robust stability properties with well posed implicit characteristic procedures[4] at all external flow boundaries. However, the upwind method particularly excels in accuracy and stability in the vicinity of interior boundaries of overset multiple grids with only a minimal overlap at their interfaces as required to exchange data through interpolation[5].

The combined strategy of employing upwind methods with both composite and overset patched meshes is sufficient to treat problems of any geometric and flow structure complexity with desired accuracy and computational efficiency. For complex geometry and

flow structures, the often multiple grid decomposition of the domain is chosen to maintain locally simple connectivity and nonsingular mappings of the coordinates in quadrilateral blocks on which solution procedures may be easily and universally organized. Good grid decompositions result in treating issues of geometry and flow structure resolution locally so as to achieve desirable alignment and not waste mesh resources in uselessly propagated clusterings into regions of the flow domain where fine mesh spacing is not required.

Mesh generators and flow solvers can be programmed[9,11,12] in FORTRAN with a flexible global data structure comprising a linear array with pointers and parameter controlled boundary interface routines to handle general multiple mesh decompositions of the domain. However, the pointer and parameter data setup to direct the solver over particularly complex problems is tedious and invites mistakes that may be hard to track down. The problems are inherent in the lack of or unnatural structure of linear programming languages like FORTRAN that do not exhibit the connectivity of objects nor naturally provide such facility. New ways of directing programs and organizing data with relationships can substantially mitigate these difficulties.

Recent research by Oliger and his students has supported the possibility to write programs and execute them in a more natural way that directly represents the connectivity of objects. The approach is based on the directed graph which is to be implemented and exhibited in a workstation based graphical interface.[13] The graph is a natural representation of the macro data flow of the program. An important feature of the system is that it is recursive, i.e. nodes of the graph may represent pieces of the graph. To facilitate using programs with disparate data structures, filters are inserted between nodes to reformat the output (range) data of one process to be suitable for the input (domain) of the next. A language such as LIL, proposed by Goguen[14], is being developed for writing these filters and controlling processes. To minimize programming while permitting great flexibility, programming will be carried out and programs executed in the context of the UNIX environment. The Make facility of UNIX naturally lends itself to achieve the composition of these programs.

Based in the above technologies, a major purpose of the evolving research is to provide the basis for an advanced computational system for aerodynamic simulation. With the simulation facility we will generate multiple patch meshes for representative problems and direct the flow solver over the meshes with greater ease and productivity. The present final report describes a Graph-Object Based Programming paradigm being pursues at PEDA Corporation with a goal to effectively support writing generic programs in large scale scientific computation including zonal method and user interaction.

Complementary facets of the research are to seek further improvements in accuracy and efficiency of the upwind method and algebraic grid generation procedures that we have been studying.

Topics that we have pursued in the upwind methods area that derive from our recent research are, first, strongly conservative schemes with second order spatial accuracy in the complete 2-D and 3-D Taylor series' expansion. Here, second, these schemes are

expressed in available families of upwind biased finite volume computational cells rather than the classical data node centered cells. On such cells an interesting hierarchy of splittings and biased upwind methods are possible that are related to and extend the locally one dimensional higher order TVD schemes of Chakravarthy and Osher[15] and Harten[16]. Related flux limiting schemes including some based on the work of Yang, et.al.[8,21] have been explored.

Another research topic we have considered and one that has recently shown great promise, particularly for steady flow implementation with upwind schemes, is flow structure adaptive mesh redistribution. This technique requires no additional patches of mesh (with additional computational costs) to achieve the same order of magnitude improvement in resolution[10] as is achieved with mesh embedding and overset refinement[5].

Advances in the algebraic grid generation area described in the present final report emphasize extension to 3 – D. One important facet is the construction of smooth body conforming surface patches of curvilinear coordinate mesh. Our present mesh generation philosophy, which is suitable for both 2 –D and 3 – D, emphasizes techniques based on low order polynomial blending of parametric cubic and tension spline functions. These are used for both bounding and coordinate curves. Blended stretching functions are used to obtain desired mesh point distributions. The approach[2] represents a simplification of, and borrows many tools from, the transfinite interpolation method of Vinokur and Lombard[1]. The latter relates to the work of Eriksson[17] and to Eiseman and Smith[18]. For boundary surface meshes, our approach which suppresses the explicit dependence on boundary gradients of Hermite interpolation follows the work of Gordon[19].

Another development of the boundary blending philosophy which we are pursuing is patched mesh generation using isoparametric macro finite elements (Baker and Manhardt[20]). The basis functions for the elements can then be used for coordinate transformation (Mitchell and Wait[21]). A feature of the macro element approach which has recently been advanced in 2 – D by Oliger and Suhr[3] is the use of piecewise quadratics for boundary curves in a way that achieves C^1 continuity.

An important topic of the new Graph-Object-Based Programming system and that we have studied and employed as an illustration is how to represent the algebraic grid generation problem for composite meshes in a directed graph in such a way that the connectivity (continuity properties) are elicited and the procedure is made very efficient. A similar topic exists for gasdynamics in how to efficiently organize problem setup to direct the operationally explicit symmetric Gauss – Seidel implicit solution algorithm CSCM – S on the multiple, segmented mesh.

The CSCM – S algorithm has largely beer tested in supersonic and hypersonic applications. In the transonic regime, we have explored two three – dimensional problems of topical interest with the compressible Navier – Stokes code on segmented grids. The first of these is the SOCBT projectile at angle of attack and with computed base flow.[22] In the SOCBT problem that has been computed over many years with several different procedures, the CSCM method provided very good results and with computational time

reduced by about an order of magnitude relative to earlier methods. This gain in efficiency had been our experience with the nonlinear space marching technique in earlier simpler 2 – D and 1 – D model problems. The second problem was a development test configuration slender sub – orbital launch vehicle with non-constant diameter modular sections and longitudinally running protuberances. With this configuration we found we were able to explore the nonlinear aerodynamic coefficients space at selected angles of attack and Mach number with qualitative consistent results.[23] The matrix of runs required only about the same expenditure of computer time as had been typically utilized in solving a single projectile flow with older algorithms.

In the hypersonic regime the algorithm has been tested against otherwise computed multi-conic entry bodies.[24] In the flexible, multiple-patch, multiple-grid data structures allowing fitting all the local topologies and geometric length scales of a complex problem, the 3 –D CSCM code has most recently been applied and validated in the challenging new technology area of three dimensional jet interaction controls aerodynamics.[25,26,27,28]

Toward answering the needs of improved physical modeling for currently important high speed flows and combusting flows, and with technical direction, and we began placing more emphasis in the program on modeling and numerics of turbulent flows and chemically reacting flows. In the turbulence area the research builds on the work of Bardina[29] and Hong[30]. In the area of chemically reacting flow, we have continued the directions of Lombard and Nagaraj[31,32].

Finally, computer graphics naturally fits within the context of a complete simulation system. From an earlier effort we have a 2 –D graphics package programmed in a flexible multiple mesh data structure to mirror that of the solution procedure. This earlier graphics was written in the commercial DISSPLA graphics library available principally on main frames, such as the VAX and CRAY computers at NASA – Ames, on which we formerly relied. We have now rewritten the plot programs to the GKS open graphics standard which is inexpensive and widely available on PCs and workstations, such as the SUN's which we are now using. Some examples of grids, contour and velocity vector plots drawn with this system can be found associated with applications in Section 4a, Multiple – Grid Data Structure for CSCM.

In the following sections we provide additional details about the directed graph based programming paradigm and advances in computational methods, tools and technique that we have made.

References

1. Vinokur, Marcel and Lombard, C.K.: "Algebraic Grid Generation with Corner Singularities," *Advances in Grid Generation*, Vol. 5, Sponsored by ASME Fluids Engineering Div., Symposium on Grid Generation, 1983, ASME Fluids Engineering Conference, Houston TX.
2. Luh, R. C.-C., Nagaraj, N. and Lombard, C.K.: "Simplified Algebraic Grid Generation in Patched Mesh Systems," AIAA-87-0200, 1987.

3. Oliger, J. and Suhr, S.: "Component Grid Generation Using Isoparametric Macro Elements," to appear.

4. Lombard, C.K., Bardina, J., Venkatapathy, E. and Oliger, J.: "Multi-Dimensional Formulation of CSCM - An Upwind Flux Difference Eigenvector Split Method for the Compressible Navier-Stokes Equations," AIAA-83-1895, July 1983.

5. Lombard, C.K. and Venkatapathy, Ethiraj: "Implicit Boundary Treatment for Joined and Disjoint Patched Mesh Systems," AIAA 85-1503-CP, AIAA 7th Computational Fluid Dynamics Conference, Cincinnati, OH, July 1985.

6. Bardina, Jorge and Lombard, C.K.: "Three Dimensional CSCM Method for the Compressible Navier-Stokes Equations with Application to a Multi-Nozzle Exhaust Flowfield," AIAA Paper 85-1193, 1985.

7. Lombard, C.K.: "Recent Development in the Upwind Implicit Method for the Compressible Navier – Stokes Equations," Presented as AIAA Paper 85-1659, 1985.

8. Yang, J.Y., Lombard, C.K. and Bardina, Jorge: "Implicit Upwind TVD Schemes for the Euler Equations with Bidiagonal Approximate Factorization," Presented at the International Symposium of Computational Fluid Dynamics, Tokyo, Japan, 1985.

9. Venkatapathy, Ethiraj and Lombard, C.K.: "Application of Patched Meshes to Viscous and Inviscid Flows," Presented at the Sixth GAMM Conference on Numerical Methods in Fluid Mechanics, Sept. 1985.

10. Venkatapathy, E., Lombard, C.K., Bardina, J. and Luh, R.C.-C.: "Accurate Numerical Simulation of Supersonic Jet Exhaust Flow with CSCM on Adaptive Overlapping Grids," AIAA-87-0465, 1987.

11. Berger, Marsha J. and Oliger, Joseph: "Adaptive Mesh Refinement for Hyperbolic Partial Differential Equations," *J. of Computational Physics*, Vol. 53, No. 3, March 1984, pp. 484-512.

12. Nakahashi, K. and Deiwert, G.S.: "A Self Adaptive grid method with Application to Airfoil Flows," AIAA-85-1525, 1985

13. Oliger, J., Pichumani, R., Ponceleon, D.: "A Visual Object Oriented Unification System," *Center for Large Scale Scientific Computation, Stanford University*, Manuscript Classic-89-23

14. Goguen, J. A.: "Reusing and Interconnecting Software Components," IEEE *Computer*, pp. 16-28, Feb. 1986.

15. Chakravarthy, S.R. and Osher, Stanley: "A New Class of High Accuracy TVD Schemes for Hyperbolic Conservation Laws," AIAA Paper 85-0363, 1985.

16. Harten, A.: "High Resolution Schemes for Hyperbolic Conservation Laws," *J. Comp. Phys.*, Vol. 49, 1983, pp. 357-393.

17. Eriksson, Lars-Erik: "Three – Dimensional Spline – Generated Coordinate Transformations for Grids Around Wing – Body Configurations," *Numerical Grid Generation Techniques*, NASA CP 2166, 1980, pp. 253-264.

18. Eiseman, Peter R. and Smith, Robert E.: "Mesh Generation Using Algebraic Techniques," *Numerical Grid Generation Techniques*, NASA CP-2166, pp. 73-120, 1980.

19. Gordon, William J.: "An Operator Calculus for Surface and Volume Modeling," *Computer-Aided Geometry Modeling*, NASA CP 2272, 1983, pp. 1-5.

20. Baker, A.J. and Manhardt, P.D.: "Grid and Metric Generation of the Assembly of Locally Bi – Quadratic Coordinate Transformations," *Numerical Grid Generation Techniques*, NASA CP 2166, 1980, pp. 175-179.

21. Mitchell, A.R. and Wait, R.: *The Finite Element Method in PDE's*, John Wiley & Sons, NY, 1977.

22. Bardina, J., Lombard, C.K., and Luh, R.C.C.: "CSCM Three Dimensional Navier-Stokes Computational Aerodynamics for a Projectile Configuration at Transonic Velocities", AIAA 6th Applied Aerodynamics Conference, June, 1988, Williamsburg, Virginia

23. Bardina, J., Lombard, C.K., Yoon, S.: "Efficient Design Analysis for a new Launch Vehicle Using the 3-D CSCM Navier-Stokes Method", AIAA-89-0336, 27th Aerospace Sciences Meeting, Jan.'89, Reno, Nevada

24. Bardina, J., and Lombard, C.K.: Three Dimensional Hypersonic Flow Simulations with the CSCM Implicit Upwind Navier-Stokes Method"; AIAA-87-1114-CP, AIAA 8th Computational Fluid Dynamics Conference; June, 1987; Honolulu, Hawaii

25. Hong, S.K., Nystrom, G.A., Wang, D., Bardina, J., and Lombard, C.K.: "Simulation of 3-D Jet Interaction Flowfields with CSCM on Multiple Grids"; AIAA-89-2552; AIAA/ASME/SAE/ASEE 25th Joint Propulsion Conference; Monterey, CA; July, 1989

26. Lombard, C.K.; Hong, S.K.; Bardina, J.; Codding, W.H.; and Wang, D.: 'CSCM in Multiple Meshes with Application to High Resolution Flow Structure Capture in the multiple Jet Interaction Problem"; AIAA-90-2120, AIAA/ASME/SAE/ASEE 26th Joint Propulsion Conference; Orlando, FL; July, 1990

27. Hong, S.K., Bardina, J., Lombard, C.K., Wang, D., and Codding, W.H.; "A Matrix of 3-D Turbulent CFD Solutions for JI Control with Interacting Lateral and Attitude Thrusters"; AIAA-91-2099, AIAA/SAE/ASME, 27th Joint Propulsion Conference, Sacramento, CA, June, 1991

28. Lombard, C.K.; Hong, S.K.; Bardina, J.; Oliger, J.; Suhr, S.; "Simple efficient Parallel Computing with Asynchronous Implicit CFD Algorithms on Multiple Grid Domain Decompositions;" 12th International Conference on Numerical Methods in Fluid Dynamics, Oxford, England, July 1990

29. Bardina, J.: "Turbulence Modeling Based on Direct Simulation of the Navier-Stokes Equations," *First National Fluid Dynamics Congress*, AIAA-88-3747, 1988.

30. Hong, S.K. and Payne, F.R.: "Development of Large – Eddy Interaction Model for Inhomogeneous Turbulent Flows," AIAA-87-1248, Honolulu, Hawaii, 1987.

31. Lombard, C.K. and Nagaraj, N.: "Extensions of the CSCM Upwind Methodology for Nonequilibrium Reacting Gas Flows," *International Symposium on Computational Fluid Dynamics (ISCFD)*, Sydney, Australia, July, 1987

32. Nagaraj, N. and Lombard, C.K.: "Navier-Stokes Simulation of Real Gas Flows in Nozzles," AIAA-87-1291, Honolulu, Hawaii, 1987.

2. Graph-Object Based Programming and Algebraic Grid Generation

The goal to write generic programs for arbitrary collections of organized grids and other associated parameterizations of physical problems has led us to explore concepts of program and data abstraction in arbitrarily flexible directed graph representations. These generic programs achieve their specificity through input that may in many important cases be developed interactively at runtime. Here we discuss the concepts and illustrate the approach in the very relevant problem of interactive algebraic grid generation for arbitrarily topologically connected multiple block grids.

The directed graph data structure provides a variety of natural representations of complex systems dating back 100 years to the chemical process industry. These graphical representations of systems well encompass program structure and data relationships of computer software. The representations enable a powerful visually intuitive framework for an effective object based programming paradigm.

Distilled from the many flavors of object-based / object-oriented programming in the present firmament are some simple primitive concepts that support flexible, extensible generic large scale scientific software. Central to the concept is the idea of abstraction.

In the abstraction of programs two entities are identified – data or instance variables defining state and functionability or methods that operate on or change the state. The object concept is the identified duality between an abstract data type and the associated functionality that governs it.

The essence of object based programming is the identification of the important elements or kinds of data to be dealt with in a given program. It is implicitly assumed the program must be able to deal with these data elements in collections or multiplicative instances and often, most powerfully, in different combinations or subsets of instances for different parts of the functionality.

When the collection of data of a given kind is stored in an enumerated labeled data structure, then the abstraction of the directed graph – whereby subsets of instances of the collections are identified as the children of a graph defining a particular sub-problem to be computed – can be used as the flexible data element of a graph-object based generic programming paradigm.

Illustration in Multiple Block Grid Generation

The graph-object based programming paradigm is natural to generically organize computation associated with interconnections among modular systems such as multiple block grids.

With particular relevance to scientific computation, the same or readily associated supplementary Graph-Object Data Structures can be used to construct and represent the grids as is used to solve PDEs and render graphics or other procedures upon them.

A key to the utility of the concept for both comparatively straight forward, error free generic program organization, modification and extension is the natural way the concept

supports complex programming based in unique data trails. That is, different parts of a program are not changing different copies of data with the potential to create unforeseen side effects.

Also central to the concept and a key to the utility of object orientated programming is functional decomposition in pieces that govern or change generally one principle element or kind of data in a comparatively simple or predictable way. The latter property is particularly important for procedures based in user interaction where too many immediately available degrees of freedom can lead to confusion, frustration, slow learning and ineffective performance.

The latter was precisely the situation we encountered with our first research code for 2-D algebraic grid generation based in single step transfinite interpolation. With many governing parameters to set, it was extremely difficult to find effective combinations to realize good meshes. This experience led to a simplifying organization in sequential steps with separately identifiable data being impacted in easily conceptualized ways governed by only a few parameters at each step. This modified organization is expressed in the Five Algebraic Step algorithms, we have called FASTWO and FASTHREE for 2-D and 3-D respectively, and to be further described in the next section.

The original research codes based in these evolving algorithms was written in FORTRAN and suffered from the limitations and inefficiencies of the available data and program structure options, including too simple and rigid data types and awkward sequential run stream interaction through file based input.

The functional decomposition of the algorithm is exhibited in the directed graph of Figure 1. Here the steps of the algorithm are represented as children of the executive procedure ALGRID which is intended to be graphics based and user interaction driven.

As an example to illustrate the Graph-Object Based Programming paradigm applied in the FASTWO context we show how globally topologically singular but geometrically fully connected multiple block grid can be effectively developed within a generically flexible graphics based program organization.

The simple but very relevant elemental geometry chosen for the example is a quadrant of a circle exhibited in Figure 2a. Along with the functionality of the bounding curve, the domain can be specified by the three labeled data points that define the intersection of the bounding segments. These points are instances of a generic element of the program and accordingly will be allocated a Graph-Object Data Structure (GODS) in which the points can be stored and by which their data can be subsequently conveniently reference and accessed.

Another use of the directed graph in the paradigm is to provide a strongly visually comprehensible abstract representation of the flow of the key data elements through the pieces of functionality. The principle need to be served is a data abstraction that can provide the collection of instance variables of a given kind or kinds associated with the object being operated upon. Since the instance variables are referenceable through their GODS, this abstraction is available in a directed graph we call an Association Graph that

provides the operative connections among key data elements in a given functionality. In the graph the directed arcs point to the referenced GODS associated with referencing GODS from which the arcs emanate. In Figure 2b an association graph for Domain Specification references first its boundaries which reference their curve types and terminating corners.

The second step of the algorithm is to decompose the domain into topologically rectangular blocks which can be the basis for separately identifiable regular grids. Figure 3a illustrates the decomposition of the domain into two blocks that support a resulting grid topology providing balanced resolution, good conditioning and avoids the introduction of a pole at the center of the circle.

The second step may be further expressed to decompose the blocks into patches on which the geometry is slowly varying and readily representable by simple algebraic functions. This further decomposition is illustrated is Figure 3b. Note in the figures to minimize confusion we have used different graphical symbols associated with the number labels for the instance variables of the different elements – corners, sides and blocks or patches. The association graph for block decomposition is shown in Figure 3c.

The third step of the procedure assigns distributions of grid points on the boundaries of patches of the domain. As indicated in the abstraction of the association graph shown in Figure 4a, the distributions are accorded to segments of sides and reference prescribed grid spacing at either end of the distribution. Needed parameterization of the sides is also referenced.

The fourth step of the algebraic grid generation procedure is to fill in the interior mesh of the patches by a two boundary approach connecting opposite boundary data by polynomial coordinate curves, usually cubics derived from transfinite interpolation. In the present illustration developed ad hoc, we have used simple uniform boundary distributions Figure 4b and linear coordinate curves. The resulting trail grid shown in Figure 5 is not bad, however.

The fifth and final step of the procedure is to improve the quality of the grid through appropriate displacements of the patch boundary data and interpolation of the corrections onto the interior grid. Appropriate correction for the present example is illustrated in Figure 6a. The corrected grid shown in Figure 6b has significantly better balance than the trial grid and even as illustrated with primitive straightline generators has quite good conditioning.

As suggested in the generic association graph of Figure 6c, the displacements along sides can be defined functionally in terms of displacements specified at its corners. Note in the correction procedure, the sense of referencing data elements and resulting combinations of instance variables to be accessed is very different from the sense of referencing in other procedures. Yet the power of the graph data structures and programming abstraction makes all of this functionality very comprehensible and straightforward to organize with a tremendous economy of data definition and manipulation. Later a program written in this style will be similarly comparatively easy to modify and extend.

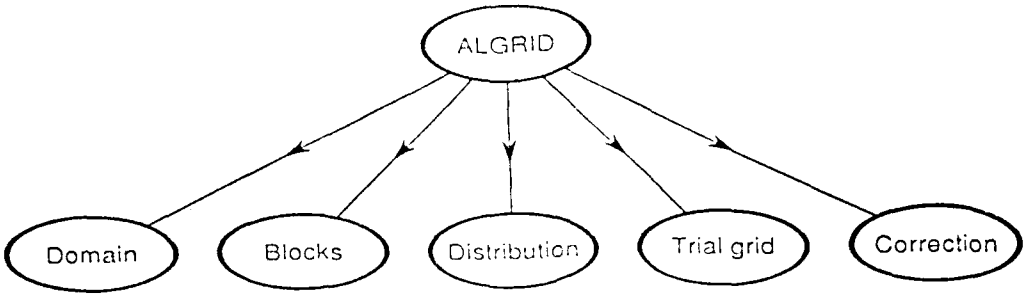


Figure 1. Directed graph representation of the Five Algebraic Step functional decomposition.

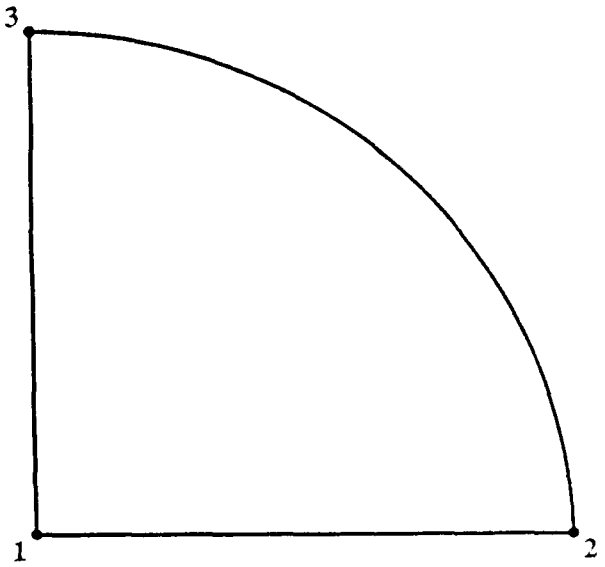


Figure 2a. Geometric domain of multiple block grid generation illustration.

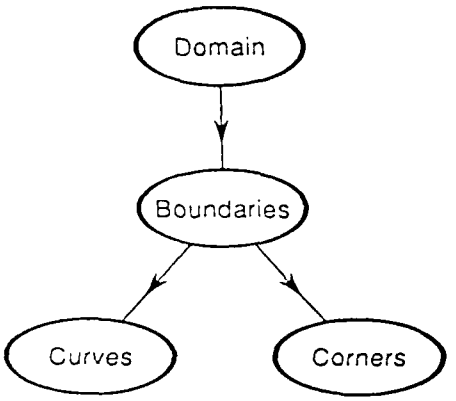


Figure 2b. Association Graph for the first step Domain Specification.

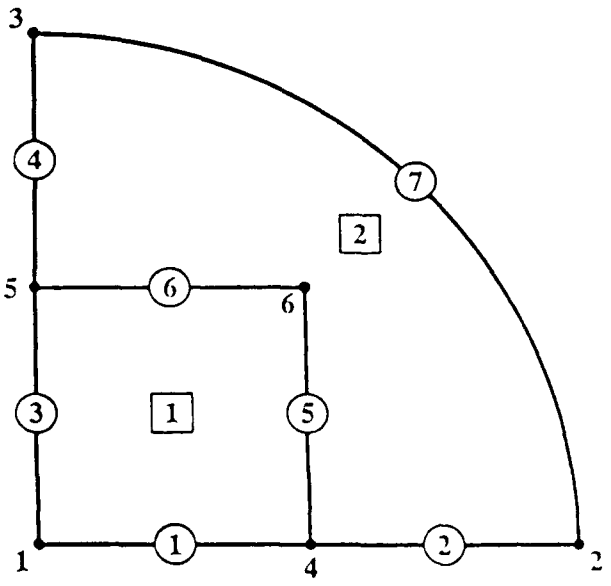


Figure 3a. Step two Domain Decomposition, here into topologically regular multiple grids.

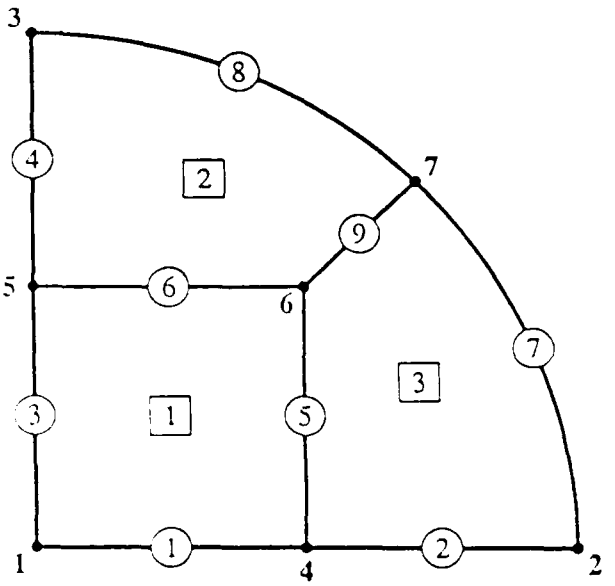


Figure 3b. Further decomposition of grid blocks into geometrically regular patches for grid generation.

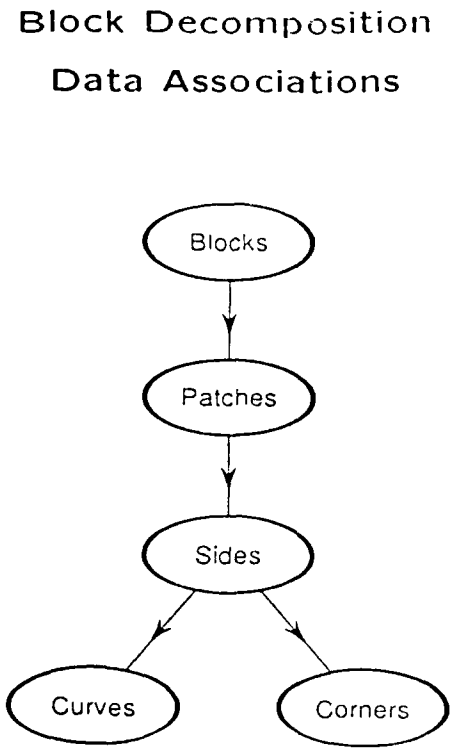


Figure 3c. Association graph for block decomposition.

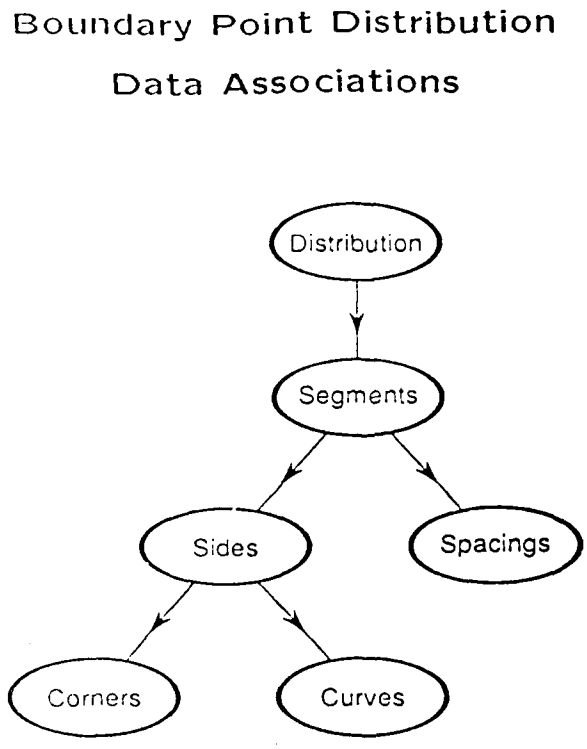


Figure 4a. Association graph for step three boundary data distribution.

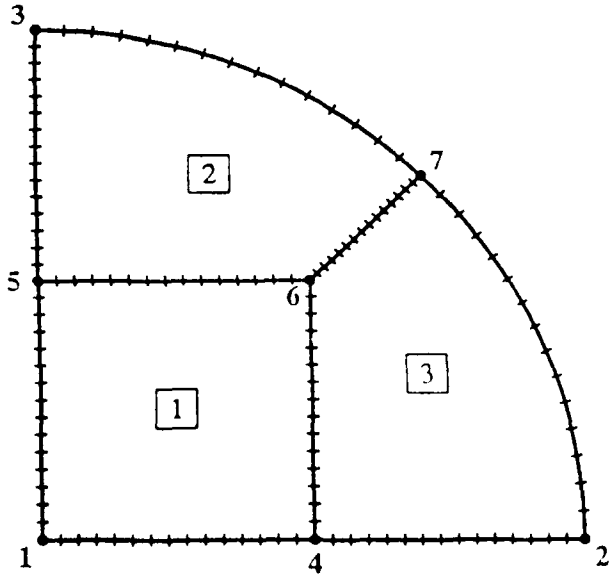


Figure 4b. Uniform boundary data distribution for illustrative problem.

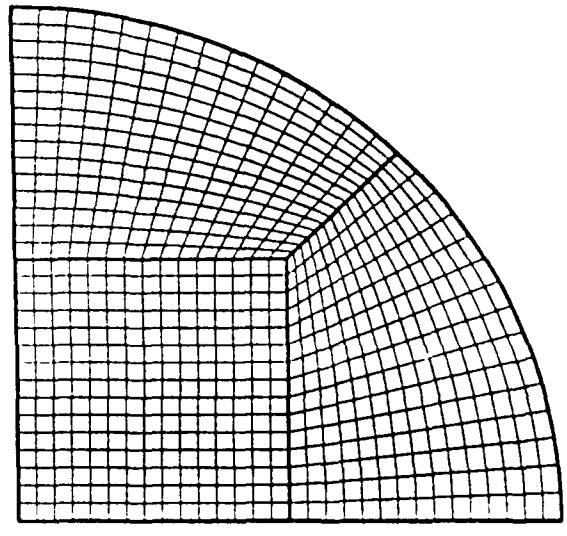


Figure 5. Step four trial grid illustrated with straight line generators.

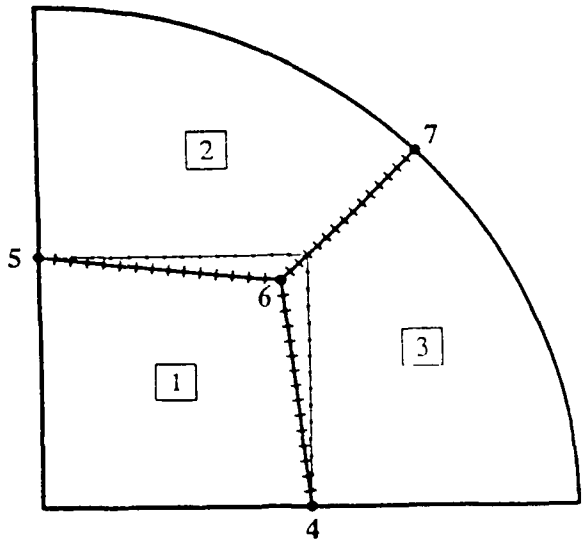


Figure 6a. Boundary data displacements associated with correcting location of corner 6.

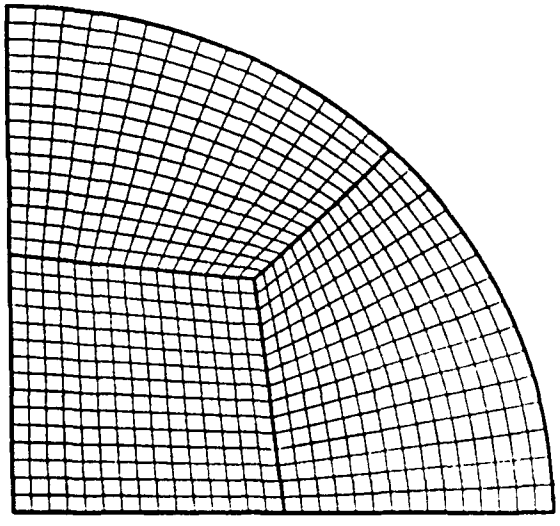


Figure 6b. Result of step five Grid Correction by interpolation of boundary data displacements.

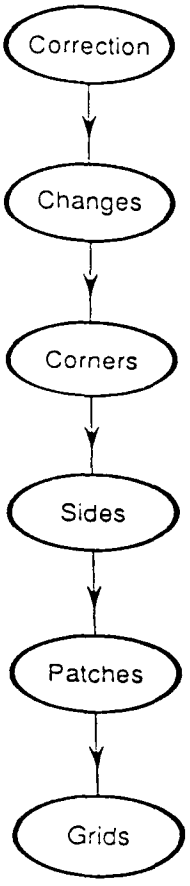


Figure 6c. Association graph for step five with very different sense of data associations from earlier steps of the functionality.

3. Computational Methods, Tools and Technique

Concluding advances that we have made in computational methods, tools and techniques, not otherwise described in previous annual reports and to be described below, are in the areas of three dimensional extensions of the five step algorithm for simple, efficient interactive algebraic grid generation for block meshes, and the companion multiple, segmented grid data structure for arbitrary geometries, with application to Navier-Stokes simulation of the Jet Interaction controls problem on a complex missile body

A. Multiple Patch-Multiple Grid Data Structure for CSCM

For the most part, and until quite recently except on a research level (e.g. references 4-11), finite difference codes have been configured to operate on a single rectangular grid of ordered points. For moderately complicated geometries of practical aerodynamics problems we've studied, such as a propulsion base flow, a hypersonic wedge-cavity problem with a single cavity and one inflow end cooling injector, and a hypersonic tunnel nozzle flow with slot injection cooled throat, source code with appropriate structuring could be effectively modified to integrate segmented sub-regions of a single grid. But for more complicated geometries the modifications required even for a comparatively well organized code, still become practically unmanageable.

From the research base[5-7,9-11] a flexible multiple-grid data structure for the CSCM implicit finite difference schemes[4-24] for the compressible Navier-Stokes equations has recently emerged[25-28] to more efficiently handle complex geometries. The new scheme is designed to handle geometries that may contain in principle arbitrary numbers of steps, cavities, embedded objects or holes. The computational space is subdivided into a number of grids. The flow solver on each grid is organized in a number of input-defined quadrilateral patches for difference operators and boundary procedures in each coordinate direction. An ensemble of commonly encountered boundary conditions for the characteristics based implicit boundary procedures is parameter addressable at patch boundaries. The data structure allows the user to break up a complicated geometry into a set of simple quadrilateral grids (and patches) on whose boundaries are specified boundary (or connectivity) conditions that are piecewise uniform by type.

A good multiple grid decomposition of the flow domain achieves efficient resolution of boundary and flow structures through local quasi-aligned meshing. The approach does not waste grid resources by propagating clusterings from one region into another where they are not needed. The multiple grid strategy affords the simplicity and universality

of data storage and program organization on local rectangles in computational space for global grid structures that may feature very complicated multiconnected singular coordinate topologies, e.g. about embedded bluff objects like a multi-element airfoil.

This scheme and its precursors involve at each time level performing integrations sequentially over the set of grids and exchanging information at interior grid boundaries[5] by interpolation of computed data from grids that the receiving grid boundaries overlay, or overlap in the case of boundaries of adjoining grids. While operationally explicit, the approach to handling boundary data with the CSCM upwind scheme with DDADI approximate factorization[4] is effectively implicit and unconditionally stable.

In global data store the grid rectangles (in two space dimensions/cubes in three) are represented as linear arrays stacked sequentially, grid upon grid with pointers to the starting addresses of data for each grid. Similarly, associated with the patch decomposition of the grids for each coordinate direction are stacked parameter arrays governing the coordinate index limits of patches, the types of boundary procedures, and parameters governing physical modeling such as turbulence, etc.

With the new data structure, the same code organization can be used to handle any two or three-dimensional geometry that one is willing to compute. It is the user specified input which defines the number of grids, the number of patches for each grid, boundary conditions on each patch, and the relations among various grids. To facilitate vectorization, regions of a grid domain that are not to be computed such as steps, holes, frozen boundaries, etc. are formally advanced with a zero right hand side of the difference equations.

The integration within each grid uses the CSCM method[4,24], which is an implicit upwind method that eigenvector splits the flux difference in the hyperbolic terms of the Navier-Stokes equations. At external domain boundaries we apply implicitly coupled mathematically well posed characteristics based boundary relations[4]. As we generally practice it, the exchange of data at interior grid-to-grid boundaries normally involves locally aligned composite grids overlapping by one grid cell and interpolation on the computed conservative variables[5] of the neighboring grid cell enclosing the boundary point. To rigorously maintain flux conservation at interior grid boundaries, we can alternatively[7], and at substantially more effort, interpolate fluxes.

We further describe the multiple-patch multiple-grid organization and demonstrate its mode of application in the context of a problem of topical interest – a complex three dimensional flow associated with the jet interaction control problem.

Application to Hypersonic Jet Interaction Control

The power of the multiple grid data structures has permitted the pioneering capture of the highly complex transverse Jet Interaction (JI) phenomenon in a realistic engineering environment involving both lateral and attitude thrusters injected from the multi-conic HEDI missile body. To resolve vastly different scales of flow structures in the multiple JI flow field and also to enable efficiently computing such flow fields on the high performance workstation environments, with particular advantage to support coarse grain parallelism locally embedded multiple grids with topologically adapted grid structures, are introduced for different parts of the computational area. From a matrix of 3-D compressible Navier-Stokes calculations for 15 different parametric combinations of jets, angles of attack, and flow conditions, a sequence of successive computations with progressive integration of numerical approaches shows the effect of the lateral, altitude and the combination of both thrusters on the HEDI external flow field at Mach number 8 with an angle of attack -8deg. Changes in the flow structure are represented in details exhibited in bow shock, location separation configurations, and in the shock/boundary-layer interaction region on the upstream side of the jet expansion. The results in particular highlight the effects on the tail attitude jet interaction flow with and without the upstream lateral thruster activated.

Introduction

A transverse jet injected form a body surface into an overlying main flow, especially when it is a supersonic or hypersonic flow, creates a complex Jet Interaction (JI) flow field. Recent experimental works, e.g.; (Sect.3) references 1-3, as well as some computational efforts[4-8] on this subject have provided an improved understanding of flow topology. From theory and our computational experience[9-10], we expect a bow shock to occur in the supersonic outer flow, a contact surface of viscous sheer layer at the interface of the two flows and another more complicated shock structure composed of weak (and possibly strong) shocks in the supersonic jet-exit domain. For jets entering transversely, such as we consider here, the outer flow is stagnated or approximately stagnated over the windward side of the jet near the exit, and controls the dynamics locally. The outer flow separates and passes around the jet on the sides, where depending upon the freestream Mach number, the lateral expansion of the jet is subject to the locally greatly reduced static pressure of the shock layer relative to the stagnation region. Thus, from the point of view of well known axial jet flow structures with patterns of embedded shocks, the existing jet could well be characterized as being overexpanded to windward but underexpanded with respect to the lateral and downwind directions.

The present results extend our earlier discussions[9,10] of the JI field over interceptor

missiles in hypersonic regime. Reference 9 deals with a lateral jet interacting with a Mach 10 flow over a biconic body[11] wherein we first attempted utilizing a curvilinear, three-dimensional grid which covers the local domain of the jet interaction flow field and follows approximately the bow shock with an optimum grid resolution. In Ref. 10, a CSCM technique extended to compute a matrix of 15 different calculations for combinations of attitude or lateral jets over the High Endoatmospheric Defense Interceptor (HEDI) missile is described.

Based on the computational techniques developed by the present authors over the past several years[9,10,12,13], we describe here the study of the JI flow fields for the HEDI vehicle with two attitude jets located near the base section and one lateral jet placed at the center of gravity of the body. In particular we have adopted a five-step computational approach in order to take account of varying influence of the upstream flow on the downstream attitude JI computational area when the lateral jet is also turned on. The five-step sequence consists of computing the following domain: (1) the tare solution over the HEDI body, (2) the attitude JI domain where the inflow is obtained from the tare solution, (3) the lateral JI field, (4) the flare section which is on the lee side of the lateral jet, and (5) the attitude JI field when the inflow is updated from the flare solution. The current paper will primarily focus on the changes in the attitude JI flow structure due to activation of the lateral thruster.

Comparison of two attitude JI fields with and without the lateral thruster exhibitis a significant impact of the lateral JI field on the attitude JI flow structure, represented through different sizes and formations of the bow shock and the extent of the vortical structure embedded in the separation zone.

Problem Description

In the current numerical simulation, the flow conditions and geometry are taken from the McDonnell-Douglas test and validation cases for the HEDI vehicle. The model is a spherical-blunted 23° tetracone configuration with a cylindrical/flared body. The presence of the tetracone forebody section introduces a complex asymmetry. A notched cavity is located at the beginning of the cylindrical middle section, and four nozzles are located at the middle of this section providing a strong lateral control jet. Two attitude thrusters are located in the rearward flared conical section, being situated at 45° off the lateral jet symmetry plane ($\phi = 0°$). A small cylindrical section follows at the end of the conical flare section. Figure 1 shows a sketch of a lateral view of the HEDI vehicle. Locations of the attitude and lateral jets are denoted in the same figure.

The freestream Mach number for the chosen condition is 8.0, and the angle of attack of the HEDI body is $-8°$ without roll angle: $M = 8.0, \alpha = -8°, \phi = 0°$. Also the Reynolds number is 3.5 x 10^6 per foot, and the freestream static pressure and density are 0.083 *psia* and 1.336 x 10^{-6} lb_m/in^3, respectively. A jet interaction scaling parameter is defined as $H = p_{0jet} A^* / q_\infty A_{base}$. Earlier, effects of different attitude jet strengths on the JI flow structure have been studied[10] with

a weak case ($H = 0.0068$), a medium case ($H = 0.03$), and a strong jet case ($H = 0.06$). Presently, the attitude jet strength is fixed with the medium case ($H = 0.03$).

The computational grids are constructed by using PEDA developed interactive algebraic grid generating procedures, FASTWO[14], and extensions to 3-D geometries with transfinite interpolation.

3-D CSCM Numerical Method

The numerical method used to compute the flow fields is the 3-D Conservative Supra Characteristic Method (CSCM)[12] for the compressible Navier-Stokes equations. This method combines the best features of data management and computational efficiency of space marching procedures with the generality and stability of time dependent Navier-Stokes procedures. Its robust stability derives from the combination of conservative implicit upwind flux difference splitting of the inviscid fluxes, a characteristics based procedure for allocation of changes in grid cells where eigenvalues switch sign, implicit second-order central differencing of the viscous fluxes, a relaxation scheme based on three-dimensional diagonally dominant approximate factorization, and well-posed characteristic-based implicit boundary approximations. Central to its accurate shock capturing capability, the CSCM conservative flux difference splitting maintains the Roe "property U".

The efficiency of the method is based on an implicit symmetric Gauss-Seidel "Method of Planes" relaxation scheme with alternating directional space marching sweeps along one coordinate direction, and an implicit block tridiagonal two-level DDADI diagonally dominant scheme along the other two directions. An effective Newton-Raphson inner iteration procedure provides improved accuracy and greatly accelerated convergence, and maintains conservation to a very high degree for single sweep marching in supersonic zones. The space marching alternating directional sweeps are von Neumann unconditionally stable for zones of subsonic, streamwise-separated, and reverse flows as well as supersonic flow. The method provides improved propagation of nonlinear effects in order to accelerate convergence to steady state, generally in about one order of magnitude fewer iterations than two level linearized implicit methods.

Recently, the 3-D CSCM code has been organized in input addressable data structures of multiple and segmented grids, which enable computation of arbitrary geometries without reprogramming. Accordingly, with natural multiple grid domain decompositions the approach is highly suitable for coarse grain parallel mapping onto concurrent computing architectures of moderate numbers of processors[13].

The turbulence model used in this study is a modified algebraic eddy viscosity model formulation based on the Baldwin-Lomax model[15]. This model uses a two layer mixing length formulation for the eddy viscosity; the inner layer uses the Prandtl-Van Driest formula with a lower bound value of 1/14 times the maximum viscosity value across the wake for near wake flows, while the

outer layer uses a function of the maximum value of the mean vorticity and total velocity difference across the wake. This model has been successfully applied in other complex simulations[9,12,16].

Expanding on how actual computations are done, first the external, jet-free flow fields are obtained over the entire HEDI body prior to the JI computation. Then the attitude JI solutions are tackled, followed by the lateral JI computations subsequently. Due to the big pressure drop on the lee side of the lateral jet, the entire flare section is re-computed with a more refined grid than the HEDI original external grid. The new, updated solutions on the flare grid are then incorporated in computing the new attitude JI area on the downstream side of the lateral thruster. In the following, these five steps are described in sequence along with boundary conditions and some pertinent results.

HEDI External Flow Computation

The large-scale external flow grid encircles half of the test body to completely capture the bow shock, shock layer, and the strong jet interaction separation. We applied the common bisymmetry assumption imposed by the body geometry and the freestream angle of attack in order to limit the computational domain to only half of the physical domain. The symmetry plane is through the jet and the axis of the test body. The external grid has 110x50x41 points, where 110 grid points are located in the streamwise direction, 50 grid points from the body wall toward the outer freestream boundary, and 41 grid points in the circumferential direction. Fig. 2 shows a view of the symmetry plane of the computational grid and Fig. 3 provides the contour plots of the normalized pressure with its freestream value through the 45°-plane. The bow shock is captured within the computational domain and the contour lines in the shock layer are smooth and free of instabilities. The expansion waves generated at the tetracone-cylinder junction and the oblique compression shock generated at the cylinder-flare corner are well captured. The effects of the freestream angle of attack on the blunt body shock are apparent from the asymmetry of the shock layer between the upper and the lower planes. For convenience, the attitude JI solutions are also imposed on the same figure with the same plotting scale, showing a relative physical domain of the attitude JI field. The calculated flow field of this HEDI body in the absence of jets is used to provide the external flow field of the jet interaction grid domain.

Attitude JI Problem

Two attitude jets are located at 31.7 inches from the virtual nose of the vehicle on two 45° planes off the HEDI body symmetry plane, and are injected from rectangular nozzles which are slanted forward by 30°. The attitude jet nozzle has a throat to exit plane area ratio of approximately 6 and a divergence angle of 15° longitudinally and 30° laterally. This 3-D nozzle has been simulated separately with the CSCM code. The jet exit plane of this nozzle is used as an inflow boundary in the jet interaction simulation. The mean flow conditions at this plane are approximately Mach 2 with static pressure about 250 times the freestream value.

(i) Grid Geometry

A multiple grid system employed in the present work is comprised of a moderately coarse outer region mesh and of a fine mesh near the immediate jet expansion area. In this work we present solutions for the jet interaction problem with three independent grids. The first grid fits the core region of the nozzle flow exit. The second grid wraps around the nozzle wall boundary layer and the jet-wall exit area. The third and largest scale grid fits the local jet/external flow strong interaction region including the bow shock and primary separation surface over the jet flow. The number of mesh points is (47x53x27), (47x149x20), and (51x65x69) for grid number 1, 2, and 3 respectively, totaling 436,052 points. Figs. 4 and 5 show this 3-D multiple grid system in the JI symmetry plane and the crossflow plane, respectively. The grid organization smoothly patches on a wraparound polar topology on the upwind side of the JI flow. The three grids are connected with one mesh cell of overlap at the interface between the grids to permit mutual exchange of computed data as boundary data.

The outer grid boundary follows a parabola, whose shape and distance from the wall are determined using a Schlieren photograph[11] as initial guide. The partitioning of the multiple grids into different patches, including details of the data exchange at overlapping boundary grids and the implementation in the CSCM code, is illustrated in Ref. 9. The CSCM algorithm is then applied to each one of the multiple grid patch domains during the solution relaxation procedure.

(ii) Solution Procedure and Results

No-slip boundary conditions for velocity and adiabatic wall conditions for temperature are imposed at the body wall. Symmetric boundary conditions are applied at the plane of symmetry, including the grid-pole in the symmetry plane of the third outer JI grid. This pole is common for all planes and is aligned with the attitude nozzle centerline. The boundary values at J_{max} of the outer grid (No. 3) which is the enveloping surface of the attitude JI grid are obtained from the HEDI tare solution, as indicated in Fig. 3.

In successive relaxations, the solutions are updated implicitly with boundary values fixed at the first and last plane of each grid. Inflow boundary values in the supersonic zone of the nozzle exit are taken from a separate computation dealing only with the attitude nozzle, while the inflow data of the last plane of grids 1 and 2 have values obtained from the overset grid 3. The inflow data of the first plane of grid 3 have values obtained from grids 1 and 2, and the inflow data of the last plane of grid 3 take values obtained from the external HEDI solution. All outflow values are extrapolated from the simulation of the interior domain. Fast relaxation is provided by the CSCM method and solutions are generally obtained after about 300 relaxation sweeps of the Navier-Stokes equations. It is noted that for these large problems one relaxation step takes approximately 40 minutes on a modest Stellar GS-1000 workstation, yielding solutions in something over a week of intermittent computations.

Figure 6 shows Mach contours in the attitude JI symmetry plane ($\phi = 45^o$) with the attitude thruster turned on. With the viscous effects incorporated, a structure of shock induced separations has occurred in the boundary layer upstream of the JI region. Over the upstream boundary layer separation a weak shock appears and intersects with the bow shock. The location of the bow shock and the size of the separated shear layer on the upstream side of the jet are dependent on the upstream flow conditions as well as the attitude jet strength: stronger jets push the shock upward. It also displays the barrel shock leaning forward due to the 30°-tilt of the nozzle centerline. The figure shows the crisp shock capture and flow turning that is realized with the upwind method.

Figure 7 depicts a velocity vector plot in the separated area near the jet upstream. Although weak, the counter-rotating vortical flows in the separated region are present underneath the weak compression waves as have been previously observed[9,10]. Then the wall pressure variation along the JI symmetry plane is plotted in Fig. 8, showing a double-peak structure on the upstream. In the absence of comparable measurements at the present time, however, it is merely mentioned that the overall trend is concurrent with previous simulations obtained for a biconic body[9].

Fig. 9 shows Mach number contours in the crossflow plane through the attitude nozzle. This figure shows the cross section of the barrel shock, the bow shock, and the extent of the secondary flow along the transverse direction within the boundary layer. The bow shock, however, is quite asymmetric with respect to the 45°-plane due to the high-speed flow settled near the upper symmetry plane. The bow shock extends laterally toward the bottom symmetry plane. These structures associated with strong viscous-inviscid interaction are in accord with the classical pattern described in the literature[1-8].

Lateral JI Problem

The lateral nozzle consists of four side-by-side rectangular nozzles with a longitudinal extent of 1.67 inches at the nozzle exit plane. The center of the lateral nozzle is located 18.4 inches from the virtual nose of the HEDI test vehicle. Again the nozzle flow is computed separately with a scaling parameter of 0.125, stagnation pressure of 225 *psia*, and stagnation temperature of $520°R$.

(i) Grid

The grid system has a resolution which is comparable to that of the earlier attitude JI simulation: 27x27x93 grid points for the jet core, 27x141x20 grid points for the wraparound wall layer immediately next to the jet exit, and 42x65x34 grid points for the outer region. The total number of grid points of the present multiple grid is 236,757 points. The computational grid in the jet symmetry plane is shown in Fig. 10.

(ii) Solution Procedure and Results

The boundary values at J_{max} of the outer grid (No. 3) are obtained from the HEDI tare solution. The solutions at the lateral nozzle exit plane are also frozen, where the average exit flow Mach number is 2.

The results have already been given in Ref. 10, and thus only the Mach contours in the symmetry plane are presented in Fig. 11. They show the overall flow pattern and the interaction of the body recompression shock with the bow shock, all consistent with the earlier work[9], the attitude JI results, and other related works[1-8]. One can also observe a quite vivid barrel shock structure with a Mach disc on top, and a thick shear layer in the shock/boundary-layer interaction region as well as the flow expansion and recompression on the leeward side. Other aspects of the flow structure are given in Ref. 10.

Flare Solutions

(i) Grid Geometry

After the lateral JI field is computed, the downstream side of the lateral JI grid over the HEDI flare region is recomputed in order to take account of the pressure drop in that area. A more refined grid is obtained from the external grid by simply putting more K and L lines over the lee side of the lateral jet. This grid is implemented to update the flow field on the downstream side of the lateral jet when the lateral jet is turned on, and has maximum indices as (J, K, L)=(26,79,61). Figure 12 displays the flare grid taken from the symmetry plane.

(ii) Solution Procedure and Results

Then the solutions in the first J plane of the flare grid is interpolated from the lateral JI solutions by searching for the closest point. Once the closest point and its corresponding grid cell is found, the lateral JI solutions are linearly interpolated onto the flare grid. The new solution on the flare grid is then obtained upto the base plane corresponding to J=26. Again the freestream values are frozen at K_{max} with no-slip and adabatic wall conditions imposed at the wall K=1. The pressure contours are given in Fig. 13 which assumes that there are no attitude jets. The body shock is seen to be contained within the computational domain. The bow shock incurred due to the lataral thruster has spread over a few grid cells, represented by contour numbers 15 through 21 in Fig. 13.

Updated Attitude JI Solutions

After the lateral thruster is turned on, the incoming flow to the attitude JI domain has changed considerably. The lee side of the lateral JI solution is now contained in the flare grid. Using this updated solutions, the boundary values at the bounding attitude JI domain is now revised at J_{max}, while the attitude JI inflow solutions at J=1 are frozen. The solution domain remains the same as in Figs. 4 and 5. It is noted that the symmetry plane is in $\phi = 0°$ for the lateral JI, while that of the attitude jet is in ϕ=45°.

First, Mach contours are presented in Fig. 14 which exhibits the effect of the upstream flow when compared to Fig. 6. Although the basic flow structure has not been altered, the size and

location of the bow and barrel shocks show a quite a contrast between the two figures. The relatively low-speed incoming flow makes the attitude jet appear stronger in Fig. 14. The velocity vector plots in Fig. 15 also demonstrate the existence of counter-rotating vortices on the upstream side of the jet. However, it is interesting to note that the separation zone has reduced its length in Fig. 14 than that in Fig. 7, but it has become thicker in Fig. 14. It seems that the stronger jet scoops up the nearing flow as it expands from the jet exit plane. The wall pressure distribution in Fig. 16 does not seem to show a much difference on the upstream side of the jet when compared to Fig. 8. But the pressure drops sharply to below the freestream value on the immediate downstream of jet, sucking a mass of flow from the HEDI base region.

A more vivid contrast between the two attitude JI fields with and without the lateral thruster can be observed from Fig. 17. Due to the low-speed flow over the upper symmetry region ($\phi = 0$), the barrel shock has now bulged upward into the low pressure area. The bow shock has been pushed further upward, and the lateral extent of it on the lower side has also been shrunken, compared to those in Fig. 9.

Figure 18 then collects all the grid and solution files in a three-dimensional picture, originally obtained in color from a Stellar workstation. It shows relative positions and physical extents of each solution domain represented in terms of Mach numbers. The Mach solutions seem to have blended smoothly from one computational domain to another.

Concluding Remarks

Current study deals with the combined effects of both lateral and attitude jets on the complete JI flow field. In the process it is hoped to demonstrate the methodology and the efficiency of the multiple grid strategy in resolving the disparate scales of JI structures. When a portion of a computational domain has been subjected to changes, the current approach can take only those affected area and re-do the computation in the influenced zone. This also saves in computation time without sacrificing the quality of flow structure in the complex flow such as the jet interaction flow field presented here.

In the course of the computation, it is always neceary to interact between the grid and the solution since the grid may have to be revised depending on the nature of the solution. For example, the bow shock in Fig. 17 is observed to have touched the grid boundary on the left-side of the computational domain. Although it was not a critical factor in that case, often times we found it necessary to expaned the grid to accomodate the important flow structures.

One drawback is a lack of experimental data to check the validity of current solutions. However, a similar computation on the McDonnell Douglas biconic body with a lateral jet has shown a reasonable match between our computed and the experimental data. Depending on the availability of experimental data, one can further study the influence of the turbulence model on the CFD solutions as well. This may bring an additional opportunity in improving the currently employed

mixing-length type turbulence model to account for the complex effects of shock-jet-boundary layer interactions and separated boundary layers.

References

[1] Sterrett, J. R. and Barber, J. B., "A Theoretical and Experimental Investigation of Secondary Jets in a Mach 6 Free Stream with Emphasis on the Structure of the Jet and Separation ahead of the Jet," AGARD Conference Proceedings IV, May 1966, pp. 667-699.

[2] Billig, F. S., Orth, R. C. and Lasky, M., "A Unified Analysis of Gaseous Jet Penetration," AIAA Journal, Vol. 9, No. 6, June 1971, pp. 1048-1058.

[3] Rogers, R. C., "A Model of Transverse Fuel Injection Applied to the Computation of Combustor Flow," 17th Aerospace Sciences Meeting, AIAA-79-0359, Jan. 15-17, 1979.

[4] Weidner, E. H. and Drummond, J. P., "Numerical Study of Staged Fuel Injection for Supersonic Combustion," AIAA Journal, Vol. 20, No. 10, October 1982, pp. 1426-1431.

[5] Rawlinson, E. G. et. al, "Computation of Two-Dimensional Jet Interaction Flow Field," AIAA 18th Thermophysics Conference, AIAA-83-1546, June 1983.

[6] Uenishi, K. and Rogers, R. C., "Three-Dimensional Computation of Mixing of Transverse Injector in a Ducted Supersonic Airstream," AIAA/ASME/SAE/ASEE 22nd Joint Propulsion Conference, AIAA-86-1423, June 1986.

[7] Shang, J. S., McMaster, D. L., Scaggs, N. and Buck, M., "Interaction of Jet in Hypersonic Cross Stream," AIAA 25th Aerospace Sciences Meeting, AIAA-87-0055, Jan. 12-15, Reno, 1987.

[8] McMaster, D. L., Shang, J. S., and Golbitz, W. C., "Supersonic, Transverse Jet from a Rotating Ogive Cylinder in a Hypersonic Flow," AIAA 19th Fluid Dynamics, Plasma and Lasers Conference, June 8-10, Honolulu, 1987.

[9] Hong, S. K., Nystrom, G. A., Wang, D., Bardina, J., and Lombard, C. K., "Simulation of 3-D Jet-Interaction Flowfields with CSCM on Multiple Grids," 25th Joint Propulsion Meeting, AIAA-89-2552, Monterey, California, July 1989.

[10] Lombard, C. K., Hong, S. K., Bardina, J., Codding, W. H. and Wang, D., "CSCM in Multiple Meshes with Application to High Resolution Flow Structure Capture in the Multiple Jet Interaction Problem," 26th Joint Propulsion Meeting, AIAA-90-2102, Orlando, FL, July 16-18, 1990.

[11] "Jet Interaction Test Case for Computational Fluid Dynamics (Single Lateral Thruster)", McDonnell Douglas Space Systems Company – Huntington Beach, 1988.

[12] Bardina, J. and Lombard, C. K., "Three Dimensional Hypersonic Flow Simulations with the CSCM Implicit Upwind Navier-Stokes Method," AIAA-87-1114-CP, AIAA 8th CFD Conference, June 1987, Honolulu, Hawaii.

[13] Lombard, C. K., Hong, S. K., Nystrom, G. A., and Bardina, J., "Asynchronous Concurrent Implicit CFD Algorithms", Fourth Conference on Hypercubes, Concurrent Computers, and Applications, March 6-8, 1989, Monterey, CA.

[14] Luh, R. C.-C. and Lombard, C. K., "FASTWO – A 2-D Interactive Algebraic Grid Generator," AIAA 88-0516, AIAA 26th Aerospace Sciences Meeting, January 11-14, 1988, Reno, Nevada.

[15] Baldwin, B. S. and Lomax, H., "Thin Layer Approximation and Algebraic Model for Separated Turbulent Flows," AIAA-73-257, 1973.

[16] Bardina, J. and Lombard, C. K., "Efficient Design Analysis for a New Launch Vehicle using the 3-D CSCM Navier-Stokes Method," AIAA-89-0336, January 1989, Reno, Nevada.

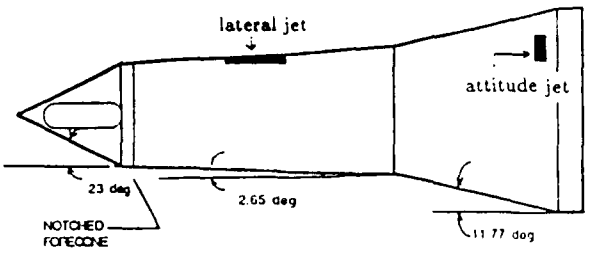


Fig. 1 Sketch of HEDI vehicle.

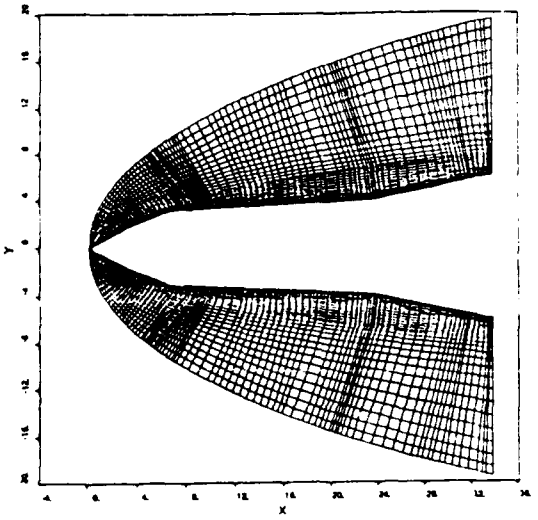


Fig. 2 Computational grid for HEDI external flow in symmetry plane.

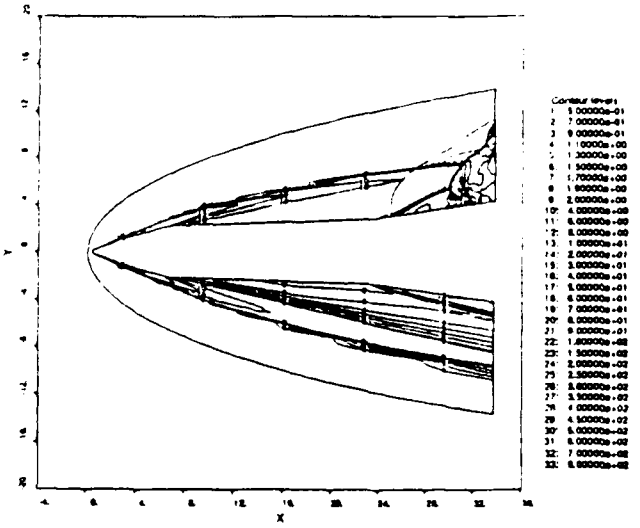


Fig. 3 Pressure contours of external HEDI through attitude JI symmetry plane: $M = 8.0, \alpha = -8.0°$.

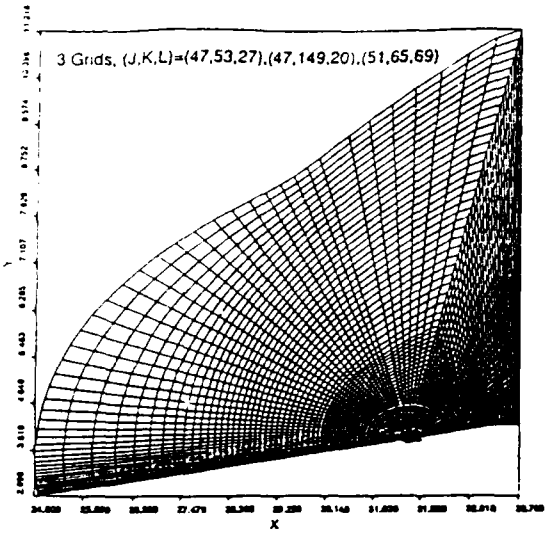


Fig. 4 Computational grid for attitude JI in symmetry plane ($\phi = 45°$).

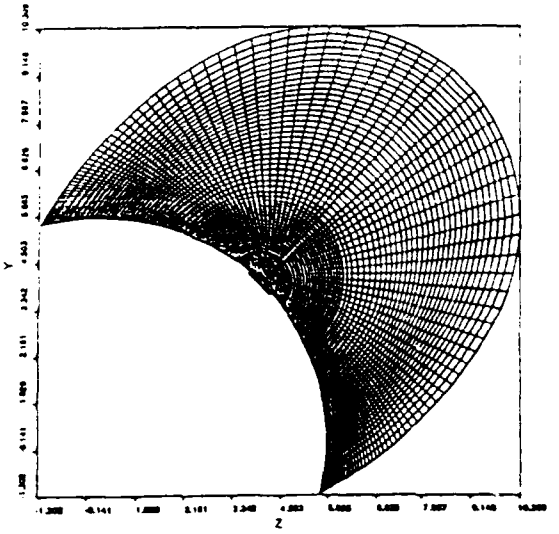


Fig. 5 Computational grid for attitude JI through jet cross plane.

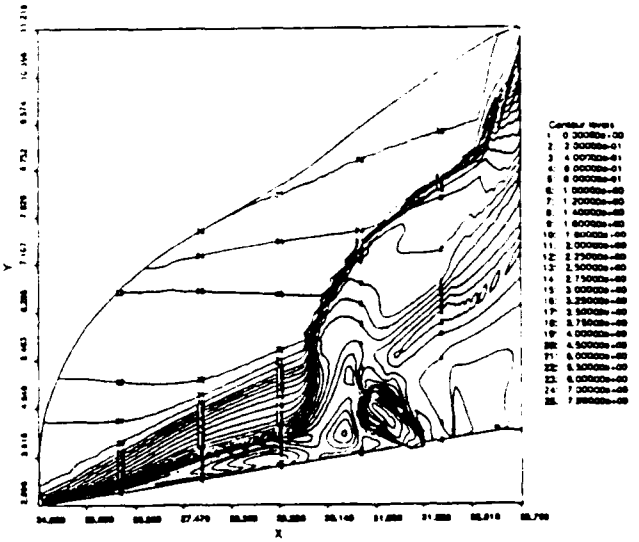


Fig. 6 Mach number contours in attitude JI symmetry plane without lateral thruster.

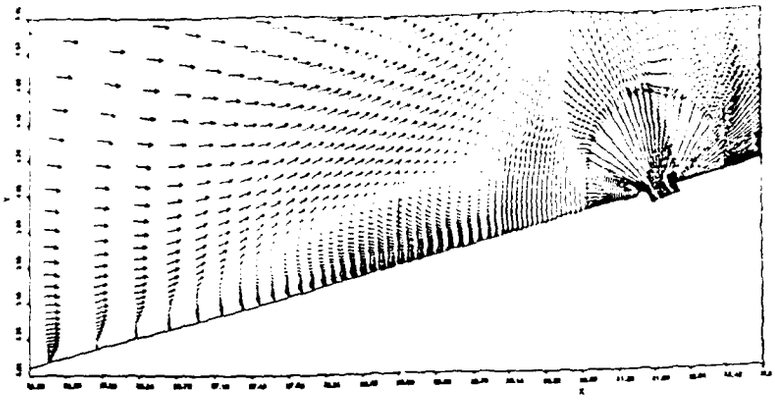


Fig. 7 Velocity vectors in attitude JI symmetry plane on the jet upstream side.

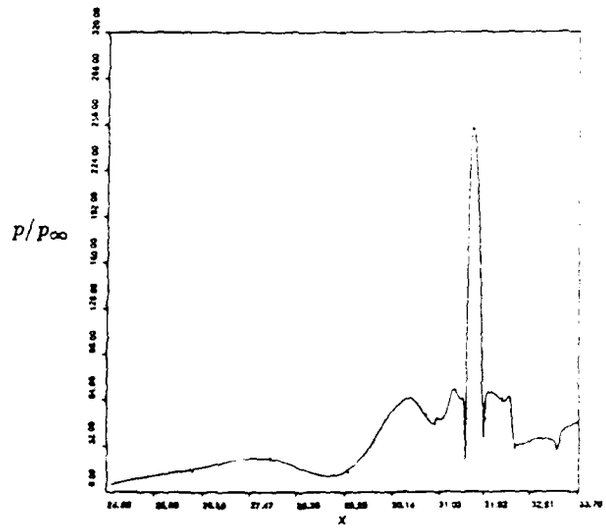


Fig. 8 Wall pressure distribution along attitude JI symmetry plane.

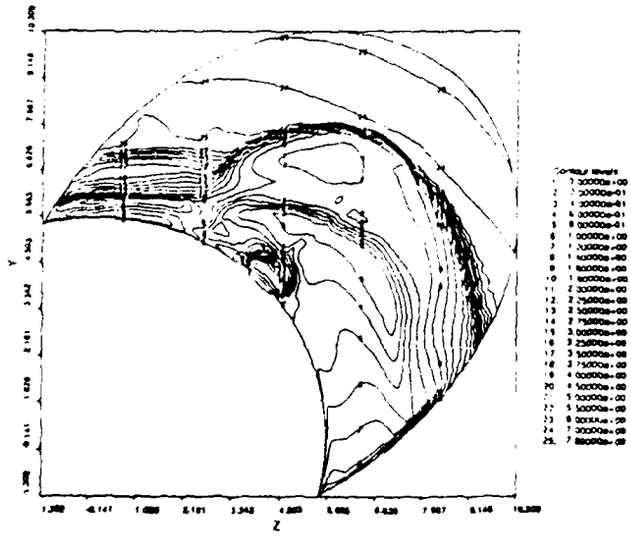


Fig. 9 Mach number contours of attitude JI field in a cross plane through jet without lateral thruster.

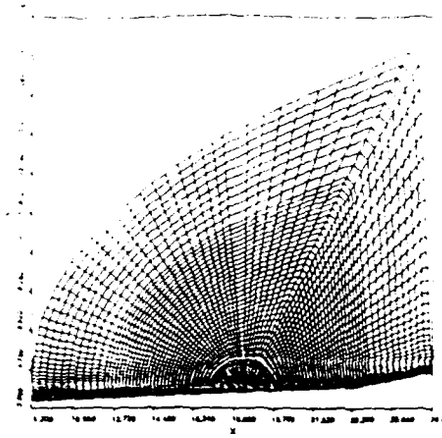


Fig. 10 Computational grid for lateral JI in symmetry plane.

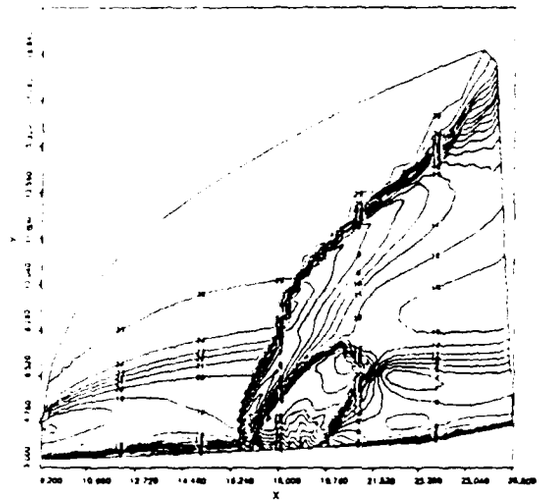


Fig. 11 Mach number contours in lateral JI symmetry plane: $M = 8.0, \alpha = -8.0°, p_{oj} = 225$.

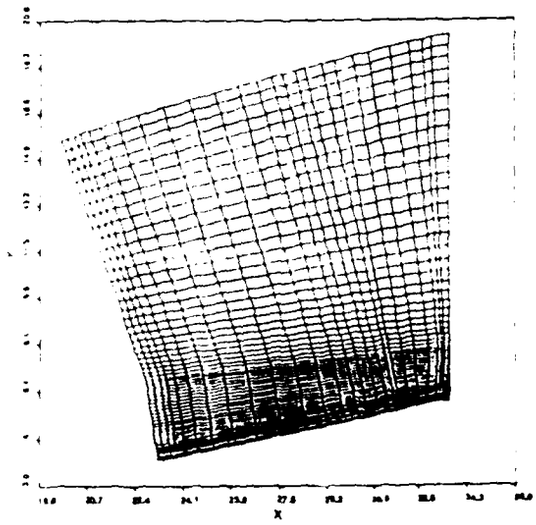


Fig. 12 Refined computational grid for flare section in symmetry plane.

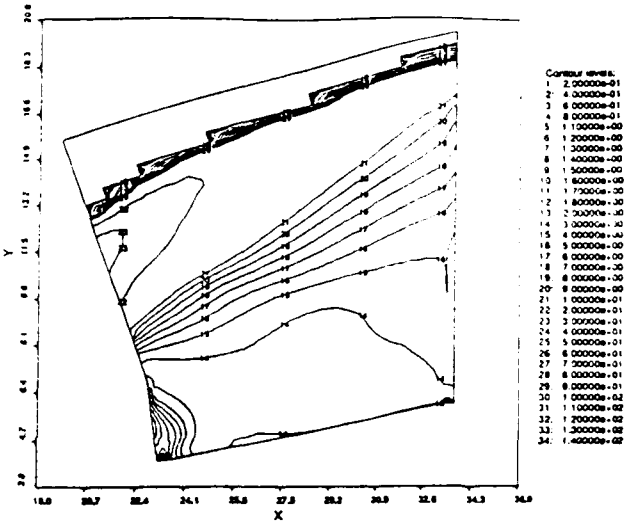


Fig. 13 Pressure contours over flare section in symmetry plane when lateral thruster is on.

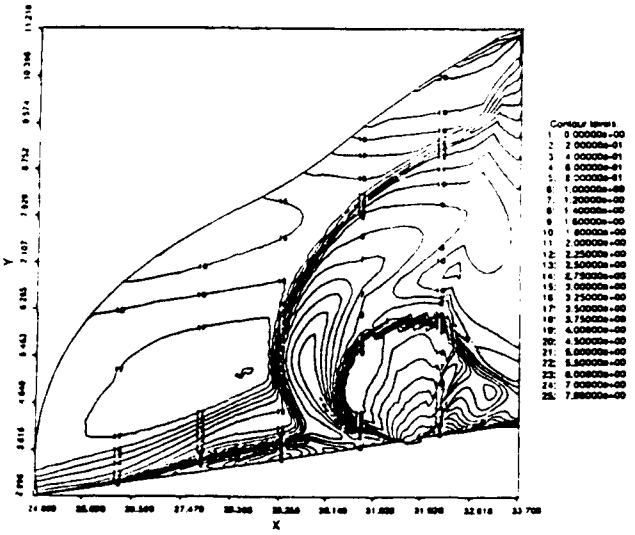


Fig. 14 Mach number contours in attitude JI symmetry plane when lateral thruster on.

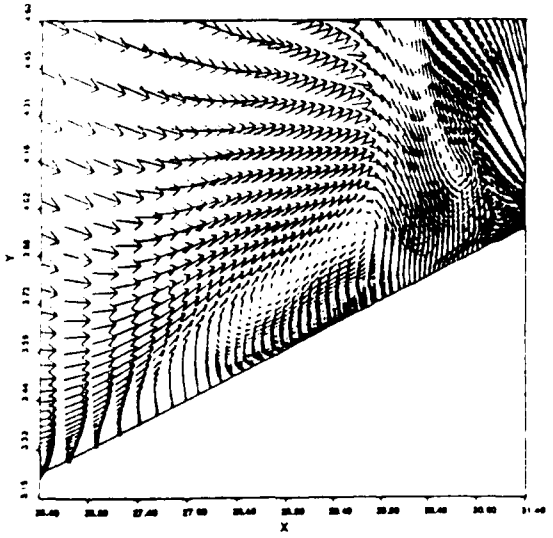


Fig. 15 Velocity vectors in attitude JI symmetry plane on the jet upstream side when lateral thruster is on.

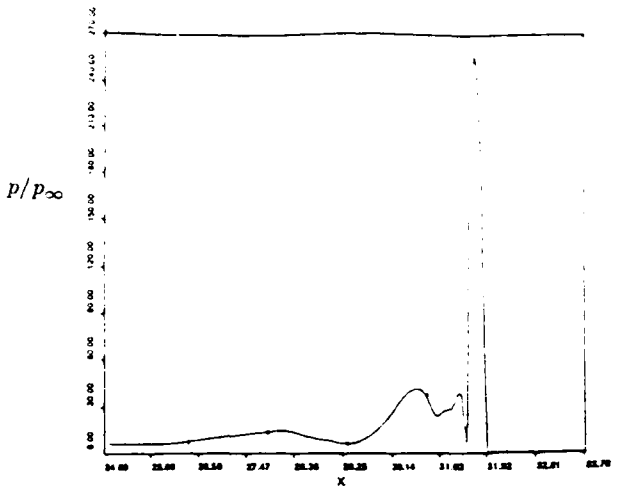


Fig. 16 Wall pressure distribution along attitude JI symmetry plane when lateral thruster is on.

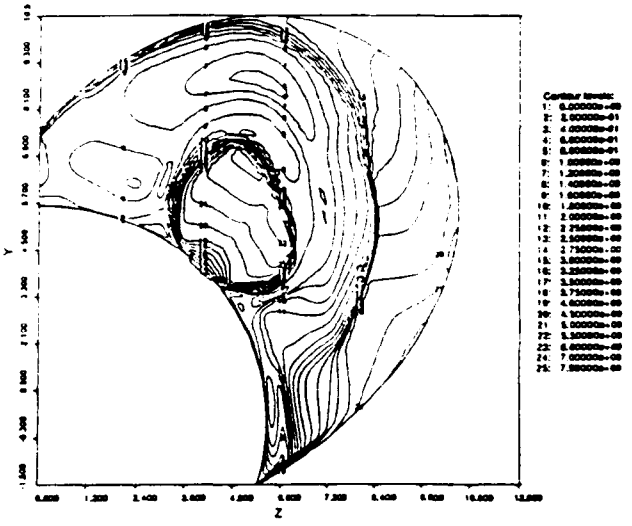


Fig. 17 Mach number contours of attitude JI field in a cross plane through jet when lateral thruster is on.

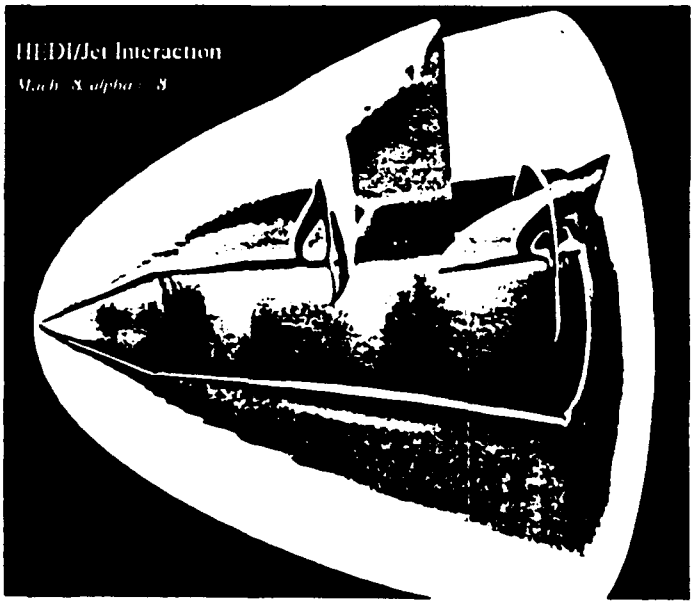


Fig. 18 Mach number distribution on 3-D view of multiple JIs over HEDI body.

B. Three-Dimensional Grid Generation for Blocked Mesh Systems

The Five Algebraic Step algorithm has been extended to generate three-dimensional blocked grids of possible topologically singular connectivity for arbitrary geometric domains. The main functional element which serves to define the grids on bounding surfaces as well as to determine the connecting interior grid lines is the transfinite interpolation of Hermite blending functions. Associated numerical procedures based on the algebraic method have the potential to be computationally fast and simple to conceptualize and use. Central to creating a smooth grid around an arbitrary complex geometry is the construction of a blocked mesh system over whose blocks the mesh geometry varies slowly and is readily represented by low order polynomials. The stepwise algorithm is appropriate for expression in a modern interactive graphics display environment.

I. Introduction

The algebraic method is becoming more and more competitive in the field of finite difference grid generation *vis-a-vis* more established differential equations methods[1,2]. Its simplicity and flexibility integrated in a framework for developing patched mesh system has resulted in a general purpose 2-D grid generation algorithm[3,4,5]. The successful graphics based implementation – that supports primitive user interaction between steps of the algebraic method – to generate grids for arbitrarily complex geometrical domains in 2-D has given impetus to further research development in three-dimensions. The current approach is similar to that taken for the two-dimensional problem in which the construction of a grid in a patched mesh system is performed in several functional steps.

In the patched mesh system, a grid domain is subdivided into regions or patches such that any single patch has only locally slowly varying geometric features. This serves to isolate prominent geometric variations from global influence which might otherwise introduce undesirable features into the grid. The patch boundaries make up a large scale visualizable wire frame of the desired grid. Interior grid lines are then filled in by transfinite interpolation of data at the boundaries. With coordinate transformation and use of cubic polynomials in an effective two boundary approach[1], local stretching or clustering can be applied easily and continuity across patch boundaries can be maintained. The number of patches naturally depends on the geometric complexities involved, but experiences in 2-D has shown[5] that the approach can be successfully applied to very complex geometries such as composite mesh around a four-element airfoil.

The 3-D problem is more complicated than in 2-D in a number of ways. First, and very importantly, there is the need for an appropriate systematic functional definition of three-dimensional surfaces. The present work successfully employs parametric bicubic technique which uses a set of surface points for bivariate interpolation on the surface. Second, the relative simplicity of the planar patch construction in

2-D is now replaced by three-dimensional blocks resulting in a blocked mesh system. The difficulty in visualizing such a block construction is much greater than the mathematics that are involved. The present work alleviates this difficulty by integrating the systematic procedure with graphic display at each step to create an interactive environment.

The main numerical elements that are either developed for 3-D or extended from 2-D are first identified and discussed. The way in which these elements are integrated in the proposed grid generation procedure is then explained with the help of an example case.

II. Numerical Elements

A. Surface Definition

It is essential that given a surface geometry in terms of data points, there be a way to redistribute points accurately and smoothly. Interpolations by the blending function methods have been widely used in connection with problems of computer-aided design and numerical control production of free-form surfaces such as ship hulls, airplane fuselages and automobile exteriors.[6] In the Coon's patch method[7], a rectangular surface is interpolated from prescribed curves on its boundaries. The method used by Ferguson[8] is a special case of Coon's in which only the corner points and their normal derivatives are required. In all blending function methods, the key is in obtaining realistic values for the derivatives. But since these are usually not available, they must be calculated from the given data. Derivative calculations are not unique and the resulting inaccuracies, especially for cross derivatives, can cause oscillatory problems.[9]

The present work accomplishes surface definition by parametric bicubic interpolation, which is a special case of Coon's patch method much like Ferguson's approach but includes twist terms. Consider a surface patch as shown in Figure 1 parametrized by u and v from 0 to 1. A point $P(u,v)$ on the surface may be interpolated as follows:

$$P(u,v) = \left[F(u)\right]\left[B\right]\left[F(v)\right]^T \tag{1}$$

where the blending functions $F(u)$ (and similarly for $F(v)$) are the cubic Hermite basis functions defined as

$$\begin{aligned} F_1(u) &= 1 - u^2(3 - 2u) \\ F_2(u) &= u^2(3 - 2u) \\ F_3(u) &= u(u - 1)^2 \\ F_4(u) &= u^2(u - 1) \end{aligned} \tag{2}$$

and B is the boundary condition matrix given below

$$[B] = \begin{pmatrix} P_{00} & P_{01} & P_{v,00} & P_{v,01} \\ P_{10} & P_{11} & P_{v,10} & P_{v,11} \\ P_{u,00} & P_{u,01} & P_{uv,00} & P_{uv,01} \\ P_{u,10} & P_{u,11} & P_{uv,10} & P_{uv,11} \end{pmatrix} \tag{3}$$

where $P_{u,00}$ stands for the derivative of P with respect to u evaluated at $u = 0, v = 0$ etc. In the above formulations, P stands for x, y, or z so that three sets of boundary conditon matrix B are required.

Partial derivative terms are evaluated by a procedure adapted from Akima's[10,11] method of weighted differences. Two examples are given to demonstrate the ability of the present method to interpolate accurately and smoothly over the entire surface. Both examples use analytical surface defined by a varying number of grid points. The task is to interpolate to a set of 25 by 25 points which are then compared to the exact analytical values. In the first example, a cylindrical surface section of unit radius is used, as shown in Figure 2. The carpet plot shows how errors in interpolation decreases steadily from a 20 by 20 grid to a 50 by 50 grid. The magnitude of the errors is never greater than $10^{-}4$ so that starting with a three-place accurary at a 20 by 20 grid, a four-place accuracy is possible at a 40 by 40 grid. The next example features a spherical surface section of unit radius with one collapsed edge. This example shown in Figure 3 demonstrates that the present method has been adapted to accomodate the special degenerate case of triangular shaped surface patches. Note that the accuracy towards the point is always good because the grid points are relatively dense. Again, a four-place accuracy can be obtained for a 40 by 40 grid except at two points.

B. Parametric Cubic

Fundamental to the present method is the formation of a block system. Each block consists of two geometrical surfaces as represented above and four corner boundary curves C_1 to C_4 as shown in Figure 4. These curves are constructed by the use of parametric cubic which can be generalized from the 2-D case. Consider a parameter s which varies from 0 and 1 between the two ends. A point $Q(s)$ is then defined as

$$Q(s) = [F(s)][b]^T \tag{5}$$

where the blending functions are the same as Equation 2 and the boundary condition matrix $[b]$ is given as

$$[b] = (\, Q_0 \;\; Q_1 \;\; k_0 Q_{s,0} \;\; k_1 Q_{s,1} \,) \tag{6}$$

The shape factors k_0 and k_1 applied at the lower and upper ends of the curve respectively permit iterative design for optimum shape. The curves thus obtained are each defined by four coefficients. The block can then be filled with cubic curves obtained from bilinear interpolation of these coefficients.

C. Orthogonalization Scheme

The transfinite interpolation as described does not insure orthogonality. With proper care, near orthogonality condition may be obtained in the majority of grid domain but usually not on solid boundaries where viscous flow calculations require orthogonality. A scheme is therefore developed as for the 2-D case[3] to replace the portion of each curve close to the surface. The problem is that of finding the surface point that is nearest the space point from which another cubic can be fit to intersect the surface orthogonally. Consider Figure 5 in which a surface patch is shown with point $\vec{P}$ and its unit normal $\vec{N}_1$. A point in space $\vec{Q}$ is nearest to $\vec{P}$ if the unit normal $\vec{N}_2$ resulting from the straight line $\vec{Q}\vec{P}$ is the same as $\vec{N}_1$. In 2-D, vector algebra leads to a single equation which can be solved by the Newton's method. In 3-D, there are actually three equations but only two unknowns. The scheme here checks on the components of the normal vector and discards the weakest component. The resulting non-linear equations are solved iteratively with the Newton-Raphson method.

D. Miscellaneous Items

The grid points are distributed on the surface and along cubic curves by a two-sided stretching function developed by Vinokur[12]. Experiences in 2-D have shown that the use of shape factors in Equation 6 can distort the point distribution. The piecewise continuous hyperbolic functions used in 2-D if extended to 3-D would be very time consuming so a more efficient method based on Lagrange polynomial interpolation is developed to regain the desired distribution. Another item which might be needed in case of a wraparound grid is a corner smoothing scheme. Composite clustering technique described in Ref. 3 can be extended to 3-D but has not been done.

III. Grid Generation Procedure

For illustrative purposes, consider the geometry shown in Figure 6. The projectile in question has flat surfaces tangent to conical sections and can be imaged from a single 45° patch cut along symmetry lines. The grid domain for this problem is bounded by the surface patch and a conical outer boundary with rounded nose. The first step is to simplify the relatively complex domain by breaking it up into a number of blocks. The underlying logic in performing this task is simply that block boundaries should emanate from the geometry surface where slope discontinuities or high curvature are present. The size of individual block is usually determined by how slow or fast the surface varies. For planar surfaces for example, the block can be infinitely large. In addition, when curvatures are present, attention should be given to local and global scales. The care in constructing the block system will be rewarded by a near-orthogonal grid. For the present problem, in order to preserve tangencies between flat and conical sections, seven blocks are required as shown in Figure 7. Each surface patch is represented by a set of 30 by 30 points. These points are parametrized and the partial derivatives are calculated and saved.

The block boundary curves are then generated. Each block as well as block boundary curve is indexed in a data structure that establishes the neighborhood information. The block structure thus constructed forms a wire frame of the desired grid and would be very helpful in visualization. This is especially true in more complex geometries because the

wire frame would bring out all distinguishing features such as wing-body intersections, cockpit outlines etc. in a full aircraft for example. The advantage in the blocked mesh system is further evidenced by the fact that when the grid lines are filled within each block, other blocks are not affected. Any problem that has to be corrected locally within a block would not mess up beyond its boundaries. Figure 8 shows a wire frame which does not include all the block boundaries for clarity but the skeletal structure is clearly illustrated.

After the grid lines have been generated, apply the orthogonalizatin step where needed in an interactive mode. One would choose where to begin turning the existing curves depending on the point distribution and local curvatures. The effect of this step can be observed in Figures 9 and 10 where two different views before and after are shown. In the surface view, shifts in the grid points are fairly subtle and barely discernible. It is more pronounced in the axial view as the orthogonality is clearly displayed.

IV. Summary

A numerical procedure for generating finite difference grids in three-dimension in arbitrary blocked mesh systems by the algebraic method has been described. The systematic approach presented has several advantages. It is fast because there are no differential equations to solve. Its usage is simple and provides flexibility in control where needed with a set of input parameters. The stepwise procedure with graphic display enhances user capability to interact efficiently during the course of grid generation. Finally, and perhaps most importantly, the difficulty in creating the grid for a problem is only as hard as constructing the block system or the wire frame for that grid. The potential for automating this block construction exists because it follows a fairly simple logic. Other tasks such as filling the meshes within each block and completing the orthogonalization step are also amenable to automation. The author envisions an eventual grid generation package with much of the work done in the background in an expert system environment.

References

[1] Thompson, J.F.: "Grid Generation Techniques in Computational Fluid Dynamics," *AIAA Journal*, Vol. 22, 1984, pp. 1505-1523.

[2] Eiseman, P.R.: "Grid Generation for Fluid Mechanics Computations," *Ann. Rev. Fluid Mech.*, Vol. 17, 1985, pp. 487-522.

[3] Luh, R. C.-C., Nagaraj, N., Lombard, C.K.: "Simplified Algebraic Grid Generation in Patched Mesh Systems," AIAA-87-0200, Reno, Nevada, Jan. 1987.

[4] Luh, R. C.-C., Lombard, C.K.: "Algebraic Grid Generation in Patched Mesh Systems," presented at SIAM Conference on Applied Geometry, Albany, N.Y., Jul. 1987.

[5] Luh, R. C.-C., Lombrad, C.K.: "FASTWO - A 2-D Interactive Algebraic Grid Generator," AIAA-88-0516, Reno, Nevada, Jan. 1988.

[6] Gordon, W.J.: "Blending-Function Methods of Bivariate and Multivariate Interpolation and Approximation," *SIAM J. Numer. Anal.*, Vol. 8, No. 1, 1971, pp. 158-177.

[7] Coons, S.A.: "Surfaces for Computer Aided Design of Space Forms," MAC-TR-41, 1967.

[8] Ferguson, J.: "Multivariable Curve Interpolation," *J. of the ACM*, Vol. 11, No. 2, 1964, pp. 221-228.

[9] Barnhill, R.E.: "Surfaces in Computer Aided Geometric Design: A Survey with New Results," *CAGD*, Vol. 2, 1985, pp. 1-17.

[10] Akima, H.: "A New Method of Interpolation and Smooth Curve Fitting Based on Local Procedures," *J. of the ACM*, Vol. 17, No. 4, 1970, pp. 589-602.

[11] Akima, H.: "A Method of Bivariate Interpolation and Smooth Surface Fitting Based on Local Procedures," *Comm. of the ACM*, Vol. 17, No. 1, 1974, pp. 18-20.

[12] Vinokur, M.: "On One-Dimensional Stretching Functions for Finite-Difference Calculations," NASA CR-3313, 1980.

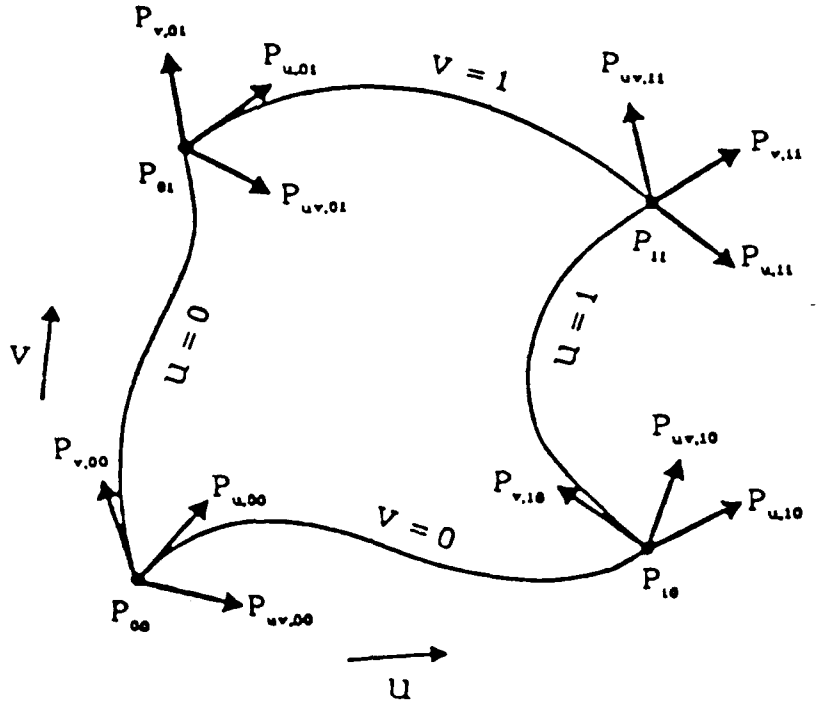


Figure 1 Surface patch defined by 16 scalar values at the corners for parametric bicubic interpolation.

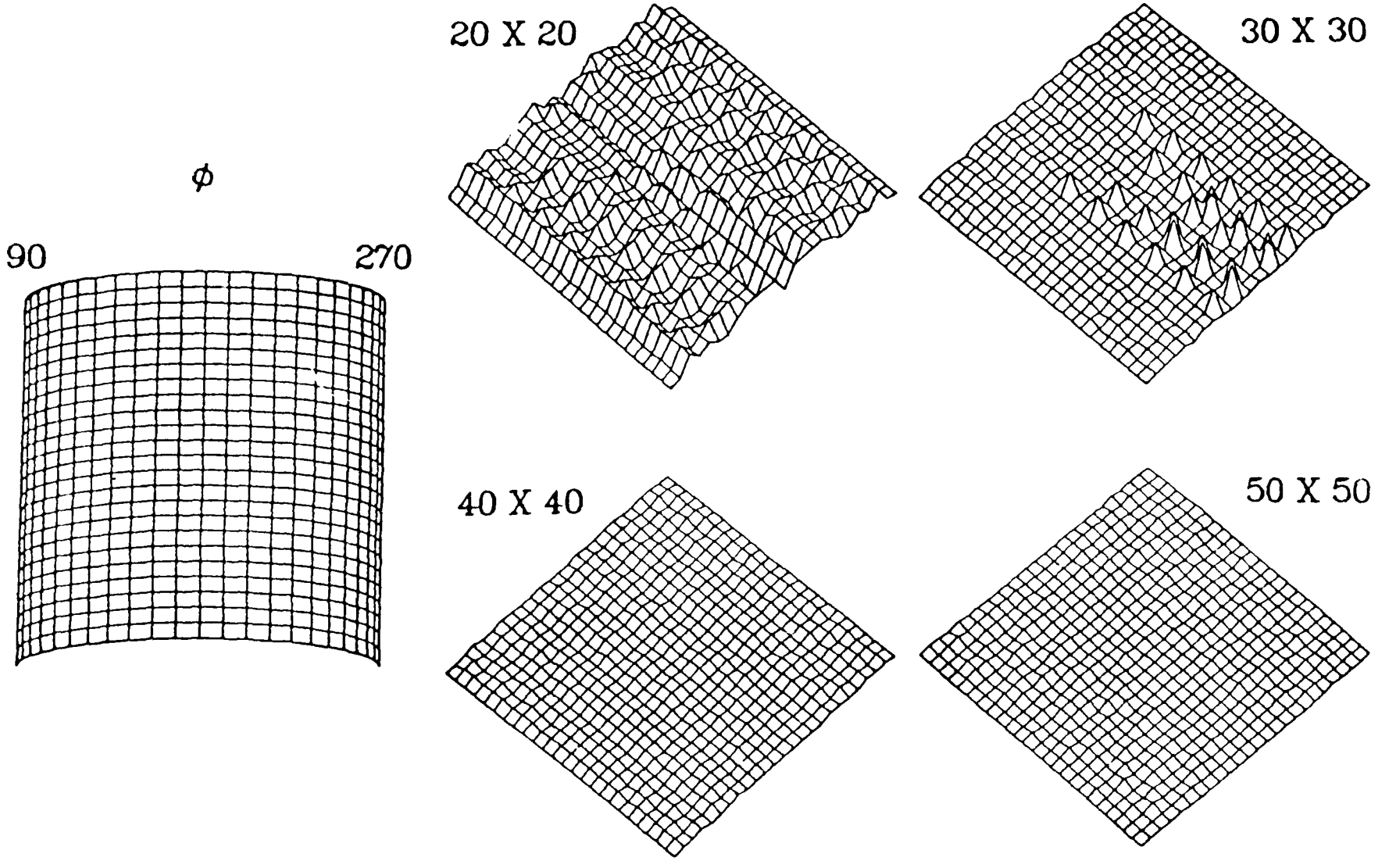


Figure 2 Carpet plot for error in surface interpolation for a half cylinder. Surfaces defined by various sizes shown are interpolated to a 25 by 25 grid.

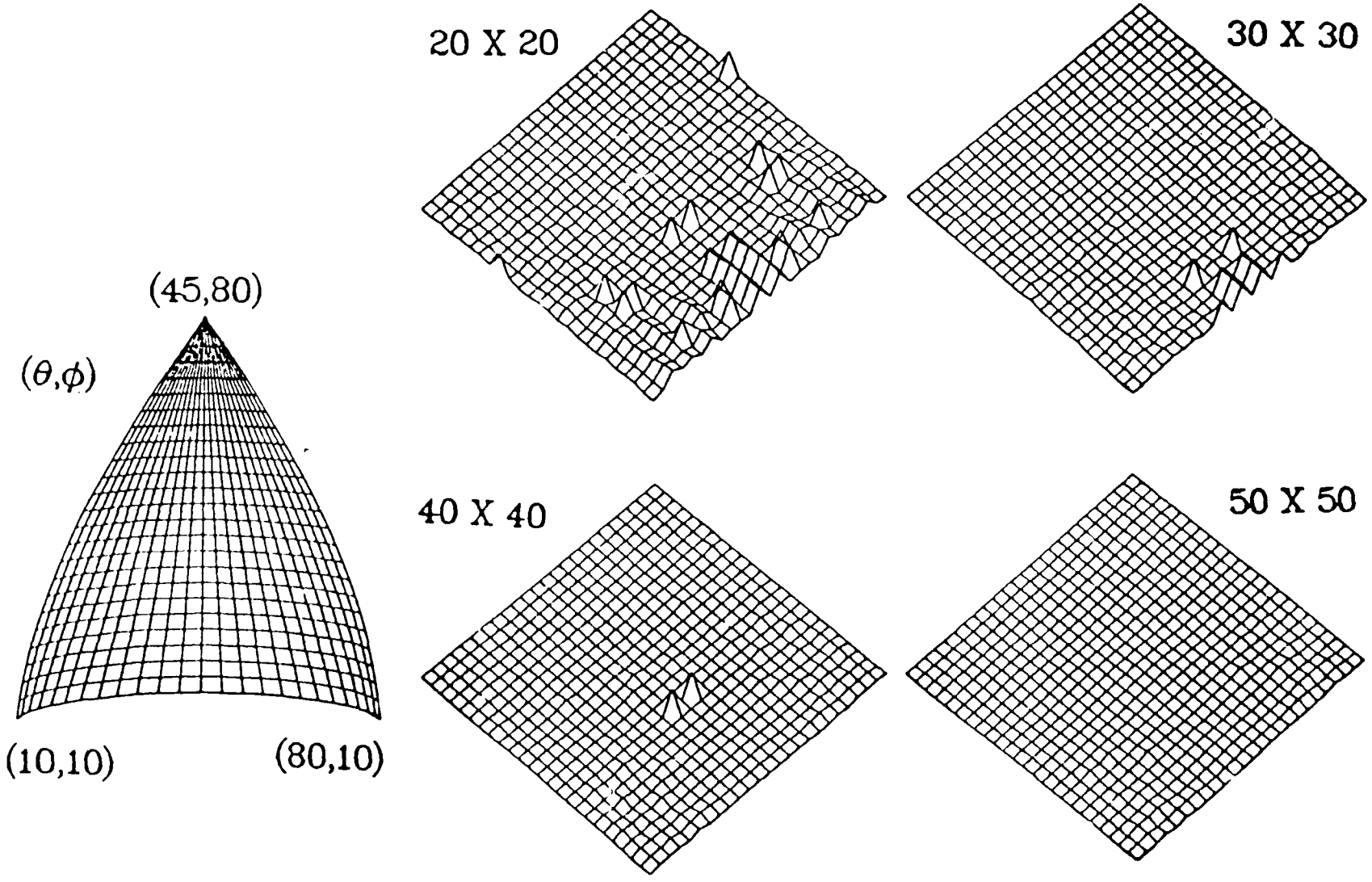


Figure 3 Same as Figure 2 but for a spherical patch with a collapsed edge.

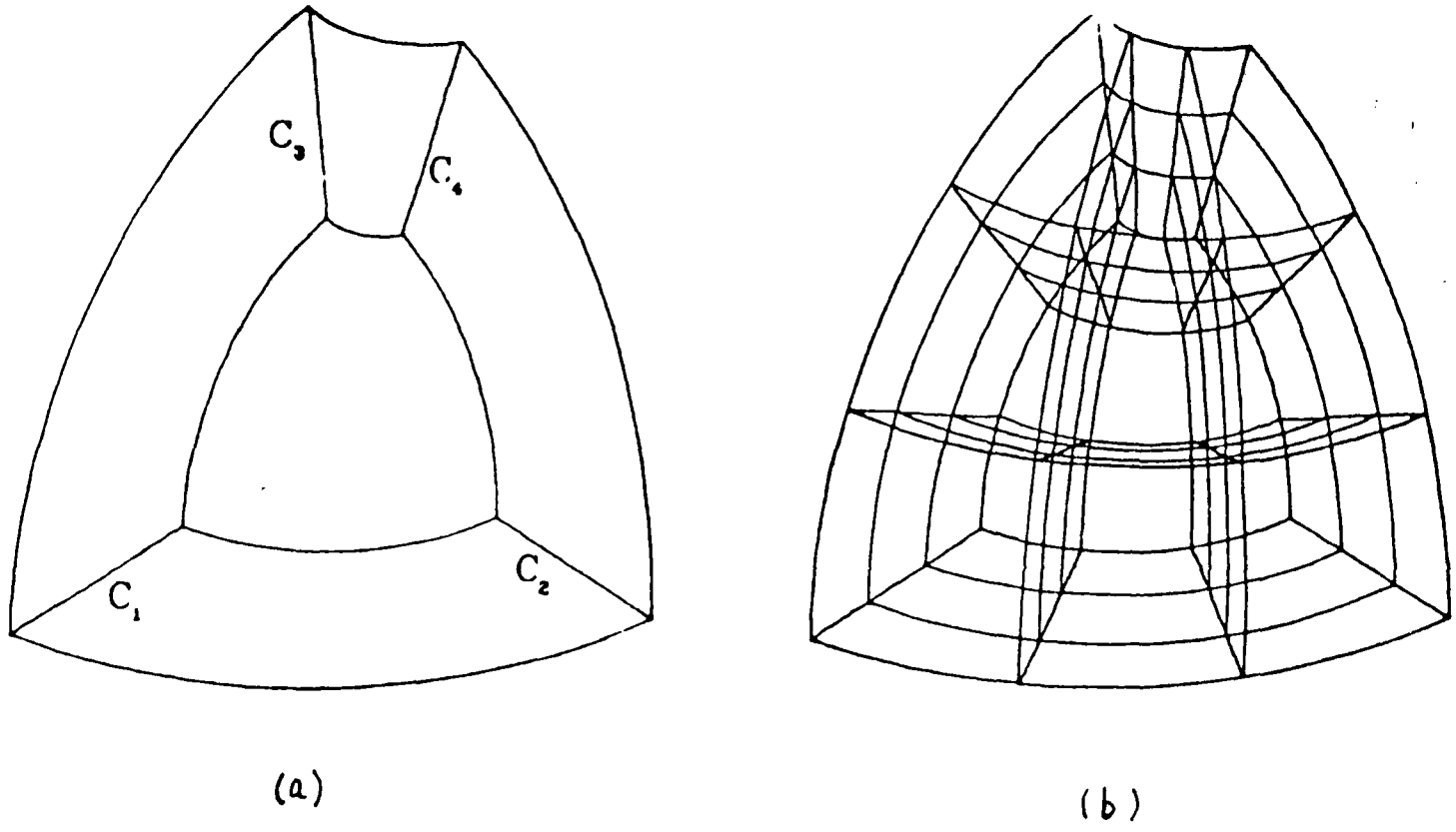


Figure 4 (a) A basic block showing two surfaces and four corner block boundary curves.

(b) Shows the block with skeletal grid lines resulting from bilinear blending of boundary curves.

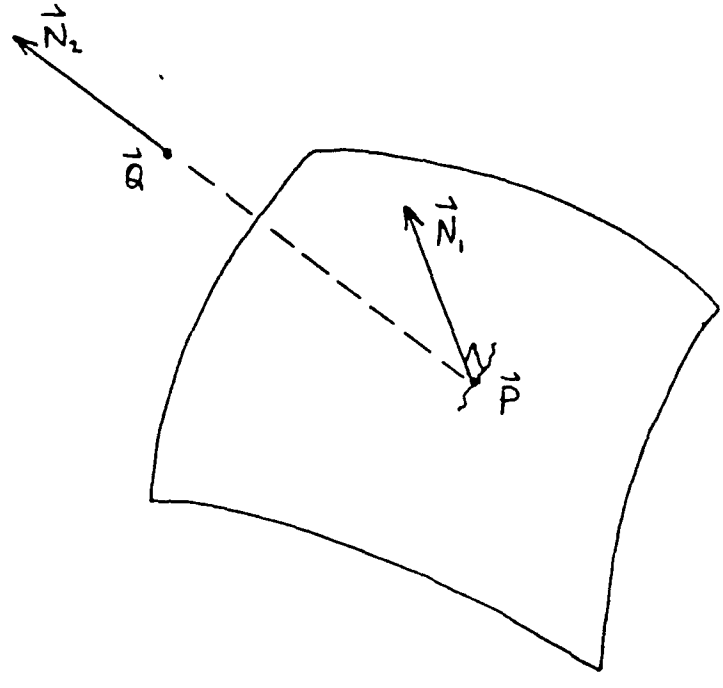


Figure 5 Sketch showing surface normals for derivation of nonlinear equations in the orthogonalizing scheme.

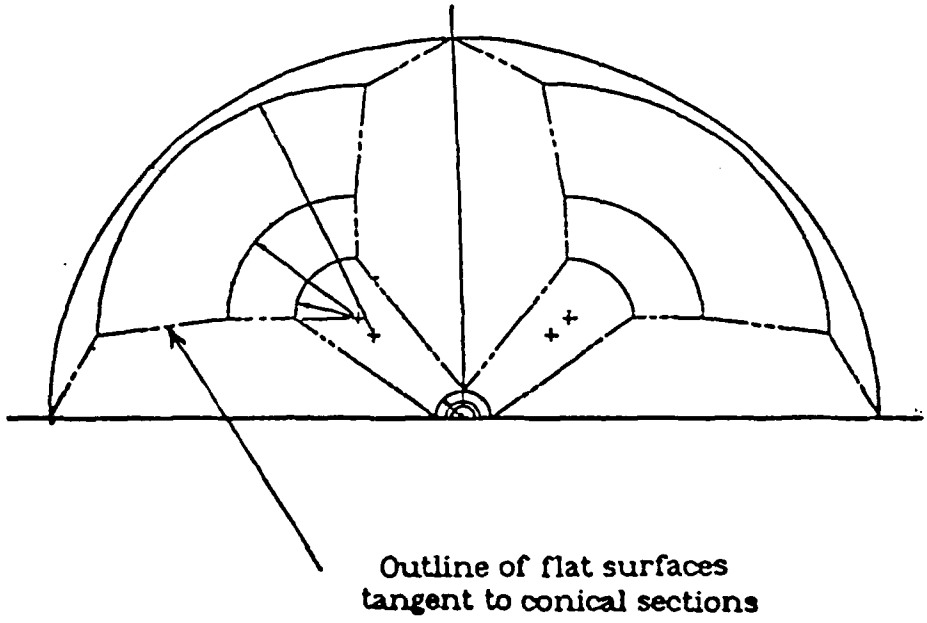


Figure 6 Half of a projectile viewed from top.

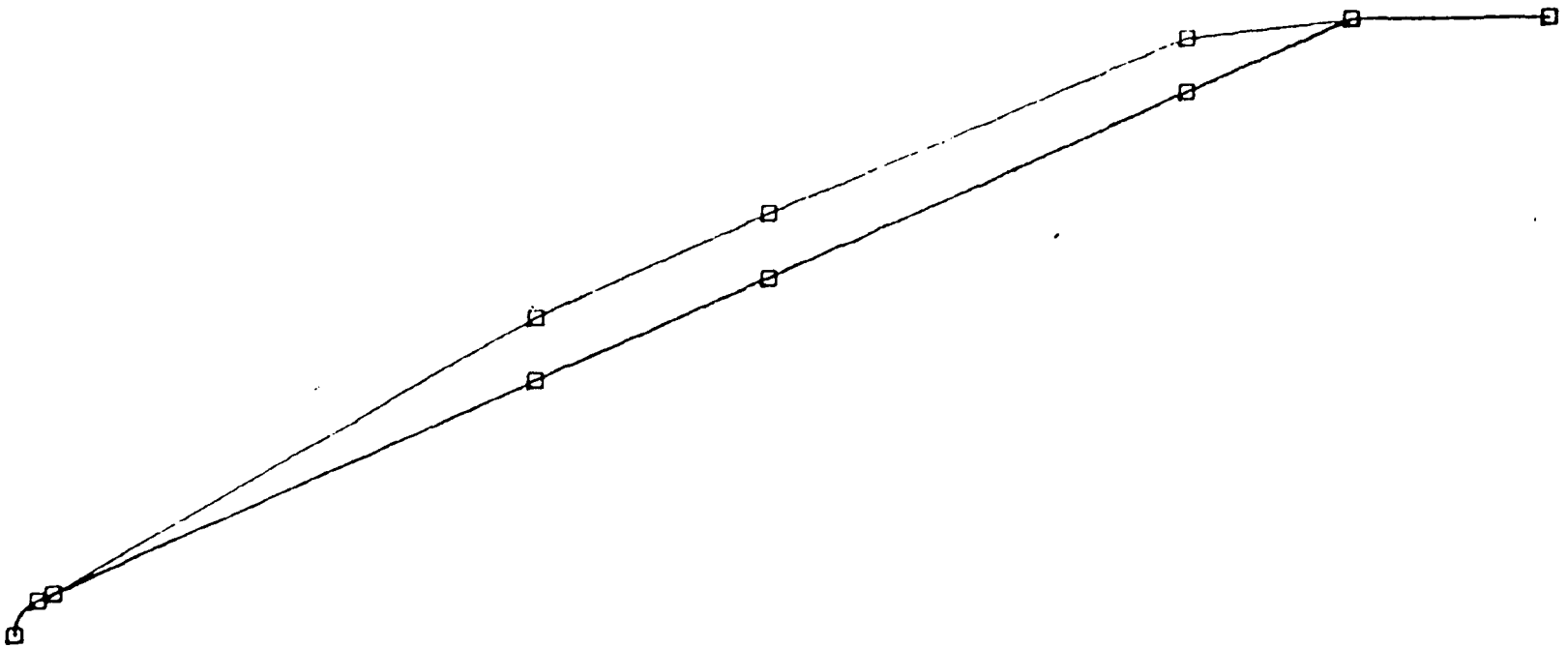


Figure 7 Side view of projectile surface at two ends showing where block boundaries are placed.

16 X 69 X 35

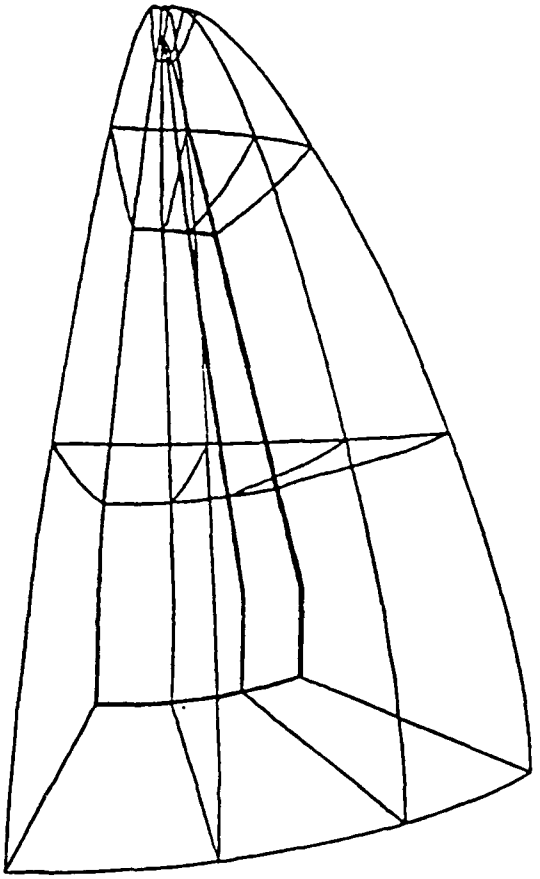


Figure 8 A wire frame display of the blocked mesh system.
The grid dimension is 16 by 69 by 35.

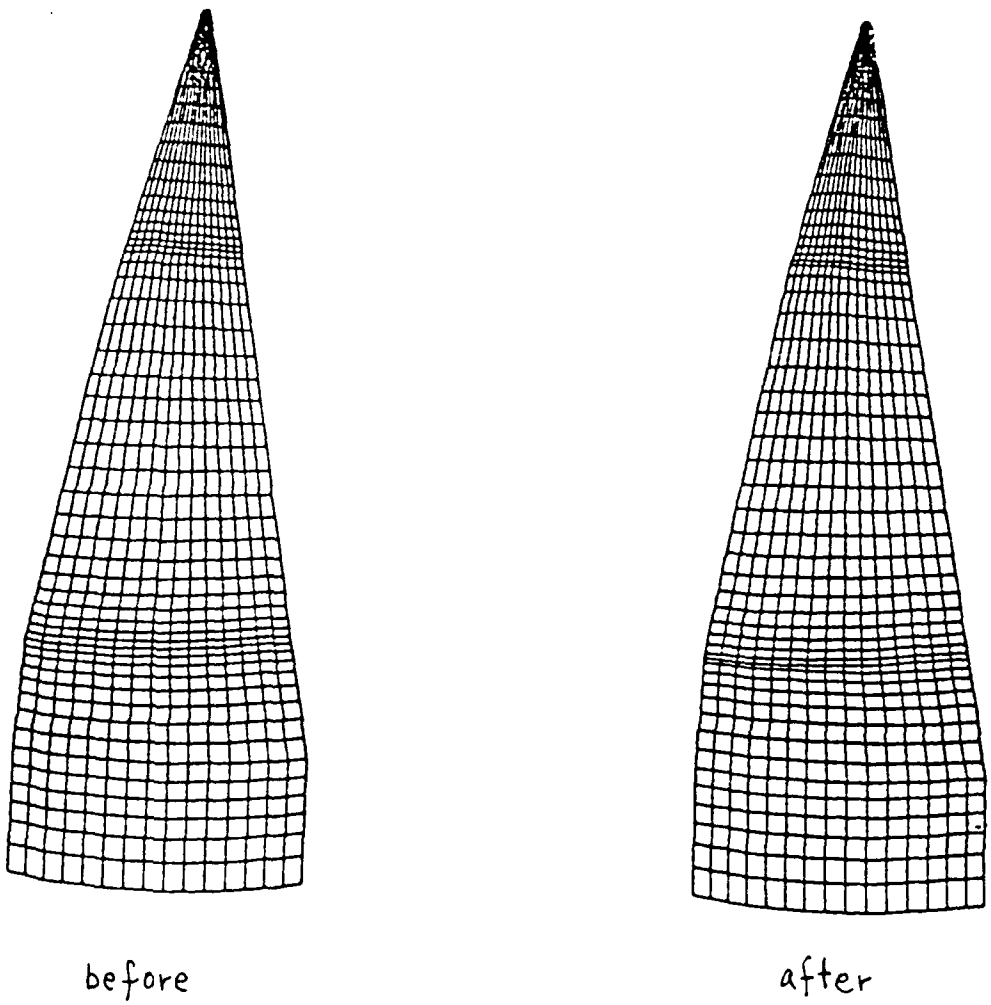


Figure 9 Effect of orgonalization on the surface as seen from the side.

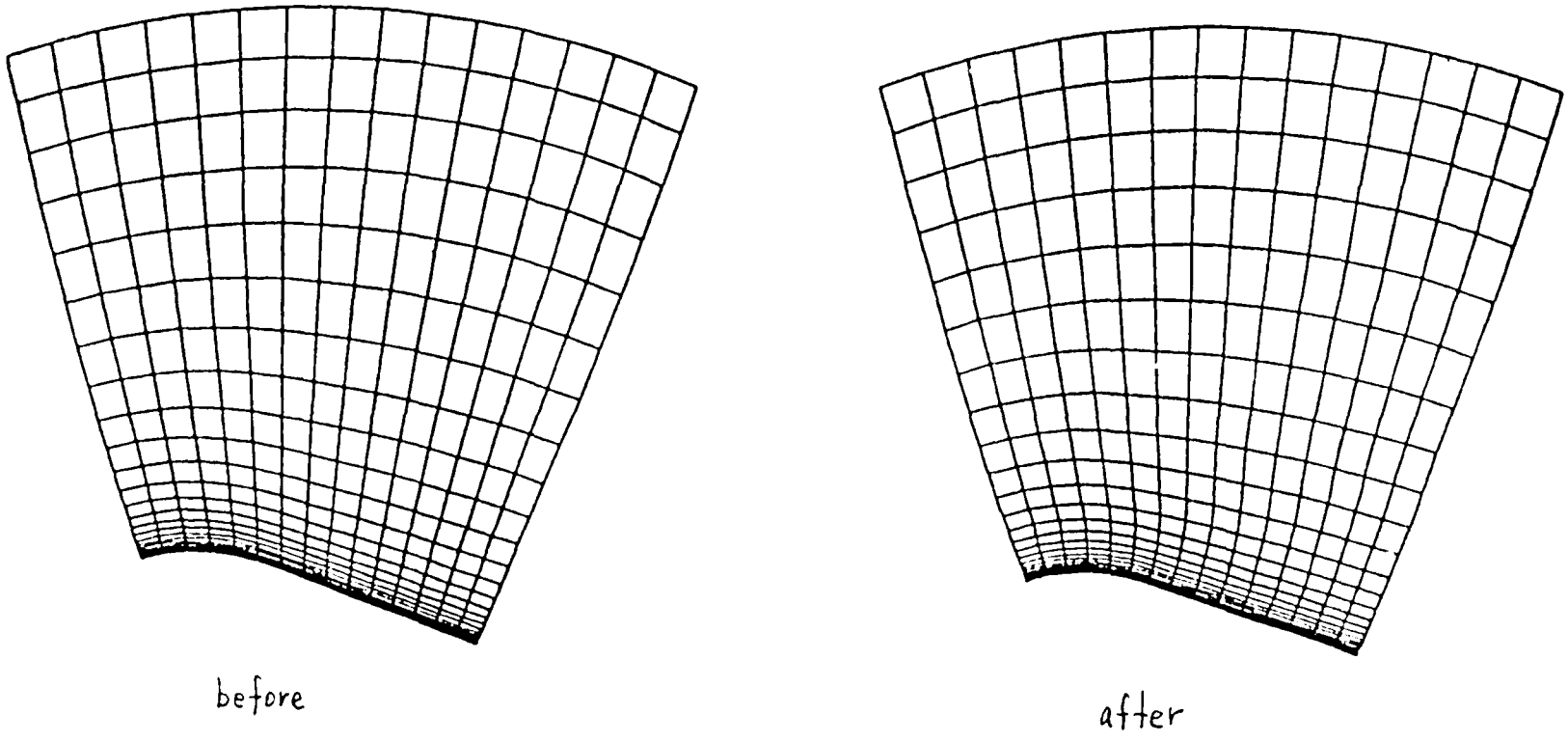


Figure 10 Effect of orgonalization as seen at an axial cut.

4. Professional Personnel

Professional researchers who were supported under and contributed to this project are:

Dr. Charles K. Lombard, Principal Investigator;
Professor Joseph Oliger, Consultant;
Dr. Jorge Bardina, Staff Scientist;
Dr. S.K. Hong, Staff Scientist;
Dr. Gustav A. Nystrom, Staff Scientist;
Dr. Ethiraj Venkatapathy, Staff Scientist;
Dr. J. Y. Yang, Senior Research Scientist;
Dr. R.C.C. Luh, Research Scientist II;
Mr. William H. Codding, Research Engineer II.

5. Interactions

Associated with the research of the AFOSR sponsored program are contributions to many U.S. meetings and workshops and seven international interactions abroad.

Solution adaptive grid research on multiple mesh systems has been given in a paper "Accurate Numerical Simulation of Supersonic Jet Exhaust Flow with CSCM on Adaptive Overlapping Grids," AIAA – 87 – 0465, AIAA 25th Aerospace Sciences Meeting, Reno, NV, January 12 – 15, 1987. The work has also been presented in a talk by C.K. Lombard at the Second Nobeyama Workshop on Supercomputing in Fluid Dynamics, Nobeyama, Japan, September 7 – 10, 1987.

Work on the CSCM upwind method expressed as a finite volume method on node bounded cells in staggered grids has been partially presented in a paper "Accurate, Efficient and Productive Methodology for Solving Turbulent Viscous Flows in Complex Geometry" for the 10th International Conference on Numerical Methods in Fluid Dynamics, Beijing, China, June 23 – 27, 1986.

Work on nonlinearly stable high order biased upwind methods has been partially presented in a paper "Uniformly Second Order Accurate ENO Schemes for the Euler Equations of Gas Dynamics," AIAA – 87 – 1166 – CP, AIAA 8th Computational Fluid Dynamics Conference, Honolulu, Hawaii, June 9 – 11, 1987.

The CSCM shock capturing scheme with improved diagonal dominance through frozen eigen functions for zero eigenvalue crossings has been presented with application of 3 – D hypersonic blunt body flow in the paper "Three Dimensional Hypersonic Flow Simulations with the CSCM Implicit Upwind Navier-Stokes Method," AIAA – 87 – 1114 – CP, AIAA 8th Computational Fluid Dynamics Conference, Honolulu, Hawaii, June 9 – 11, 1987.

The combined algebraic grid generation and surface geometry definition tools with the segmented patched 2-D/axisymmetric CSCM Navier-Stokes solver (developed under AFOSR support) has been applied to the design modification of the throat region of the Mach 14 nozzle of the NASA-Ames 3.5 ft. hypersonic tunnel, and reported in the paper, "Aerodynamic Design Modification of a Hypersonic Wind Tunnel Nozzle by CSCM

with High Order Accuracy," AIAA - 87 - 1896, AIAA/SAE/ASME/ASEE 23rd Joint Propulsion Conference, San Diego, CA, June 29 - July 2, 1987.

The new multiple and segmented mesh data structure for the 2 - D/axisymmetric and 3 - D CSCM algorithms, work described in this report, has been published as AIAA - 89 - 2552. "Simulations of 3 - D Jet - Interaction Flowfields with CSCM on Multiple Grids," also as AIAA - 89 - 2672, "Complex Wedge - Cavity Solutions Using a Multiple - Grid, Multiple - Patch Approach," and in the paper "Asynchronous Concurrent Implicit CFD Algorithms," given with other features of the method at the Fourth Conference on Hypercubes, Concurrent Computers, and Applications (co - sponsored by AFOSR), Monterey, June, 1989.

The methodology is to be further reported (in the context of a complex missile geometry for which we are computing a matrix of initially 15 Navier-Stokes solutions for aerodynamic coefficients) along with 3 - D algebraic grid generation and adaptive gridding in the papers: AIAA - 90 - 2102, "CSCM in Multiple Meshes with Application to High Resolution Flow structure Capture in the Multiple Jet Interaction Problem," and as AIAA - 91 - 2099, "A Matrix of 3-D Turbulent CFD Solutions for JI Control with Interacting Lateral and Altitude Thrusters." Lastly we add the new graph - object - based programming in "Simple, Efficient Parallel Computing with Asynchronous Implicit CFD Algorithms on Multiple Grid Domain Decompositions," a paper presented at the 12th International Conference on Numerical Methods in Fluid Dynamics (ICNMIFD), Oxford, England, July 8 - 11, 1990.

The 2 - D and 3 - D CSCM algorithms developed under AFOSR support have been applied in topical internal and external aerodynamic problems solved on patched composite grids and results presented in two papers and a workshop talk as given below.

The CSCM - S symmetric Gauss - Seidel 3 - D method of planes relaxation algorithm has been demonstrated to have markedly improved computational efficiency with respect to older technology in the paper AIAA - 88 - 2585, "CSCM Three Dimensional Navier Stokes Computational Aerodynamics for a projectile Configuration at Transonic Velocities," AIAA 6th Applied Aerodynamics Conference, Williamsburg, VA, June 6 - 8, 1988.

That work in the transonic area has been further explored showing similar effectiveness in a landmark effort to compute the nonlinear aerodynamic coefficients over a matrix of conditions for a long slender launch vehicle, work presented in the paper "Efficient Design Analysis for a New Launch Vehicle Using the 3 - D CSCM Navier - Stokes Method," AIAA 27th Aerospace Sciences Meeting, Reno, NV, January, 1989.

Combined algebraic grid generation and surface geometry definition tools have been further applied with the 2 - D/axisymmetric Navier - Stokes code for flow thermal analysis of the recently redesigned throat region of the Mach 14 nozzle in the NASA - Ames 3.5 ft. hypersonic tunnel. Results are reported in the paper AIAA - 88 - 2587, "Navier - Stokes Thermal/Aerodynamic Analysis of Hypersonic Nozzle Flows with Slot Injection and Wall Cooling," AIAA 6th Applied Aerodynamics Conference Williamsburg, VA, June 6 - 8,

1988.

Aspects of the latter work were also given in a talk at the Workshop on Aerodynamic Design of Hypersonic Facility Nozzles, AEDC, Arnold AFS, TN, November 3 - 4, 1987. This talk, suggesting greater potential for Navier - Stokes analysis in nozzle design evoked much interest among several participants involved in tunnel modification or new design activities.

The CSCM - S 3 - D implicit upwind Navier - Stokes algorithm modified for variable gas properties has been coupled with a free energy minimization procedure for equilibrium chemically reacting flow. Computational experiments were performed with underrelaxation schemes for coupling the procedures in an application to hypersonic entry vehicle flow. The work has been presented in AIAA - 88 - 2695, "Navier - Stokes Simulation of 3 - D Hypersonic Equilibrium Air Flow," AIAA Thermophysics, plasmadynamics and Lasers Conference, San Antonio, TX, June 27 - 29, 1988. Advances toward nonequilibrium chemically reacting flow have been given in the paper "Extensions of the CSCM Upwind Methodology for Nonequilibrium Reacting Gas Flows," International Symposium on Computational Fluid Dynamics (ISCFD), Sydney, Australia, July, 1987.

The utility of the CSCM upwind methodology developed with AFOSR support has been further explored in the hypersonic propulsion area in validations against turbulent shock boundary layer interaction, type IV shock interaction on cowl lips, and unsteady shock propagating inlet unstart flows. This work will be reported along with presently investigated validations against a sheared compound ramp inlet with sharpened growing sidewalls. The latter problem is being solved with the flow structure adaptive mesh point redistribution algorithm and the novel grid topology described herein.

Finally, in the international area the principal investigator attended the International Symposium on Computational Fluid Dynamics held in Nagoya, Japan in July, 1989, to assess achievement emphasis and directions in this rapidly evolving one - world discipline. PEDA Corporation research posture in addressing the many issues of computations in complex geometry, hypersonics, turbulence modeling, and complementary physics continues to maintain us very much abreast of the times and in a leadership position. Our attendance of that meeting in support of the very small U.S. presence was greatly appreciated particularly since NASA had withdrawn its sizable delegation shortly beforehand.

In another interaction, the principal investigator at the invitation of the Computer Center , Academia Sinica traveled to Beijing, China to deliver a series of lectures before the academy and also in a seminar of the Mathematics Department at Beijing University. There I found great interest and enthusiasm for our work, even to the meaningful extent that graduate students of a leading researcher had successfully implemented our CSCM algorithm in both 2-D and 3-D.

6. New Discoveries

Exponentially declining underrelaxation is a new way of stabilizing and solidly converging implicit ADI methods for gas dynamics coupled with procedures determining vari-

able gas properties (on other nonlinear modeled parameters) based on local estimate of state, as for equilibrium or nonequilibrium reacting gas.

The General Two Equation and Compressible Large Eddy Interaction are two new reduced computation Reynolds stress models for anisotropic turbulent flow.

Graph – Object Based Programming is a new cleanly extensible style of programming at high level with support for multiple mesh data structures, zonal methods/physics, concurrent processing, and graphical interaction.