NASA Conference Publication 10119

Unstructured Grid Generation Techniques and Software

Mary-Anne Posenau, *Editor*
Langley Research Center
Hampton, Virginia

Proceedings of a workshop sponsored by the
National Aeronautics and Space Administration,
Washington, D.C., and held at
Langley Research Center
Hampton, Virginia
April 27–28, 1993

SEPTEMBER 1993

NASA
National Aeronautics and
Space Administration

Langley Research Center
Hampton, Virginia 23681-0001

ORGANIZATION COMMITTEE

Ted G. Benjamin
Phone: 205 544-9402
e-mail: tedb@tyrell.msfc.nasa.gov
NASA Marshall Space Flight Center

Yung K. Choo
Phone: 216 433-5868
e-mail: sachoo@avelon.lerc.nasa.gov
NASA Lewis Research Center

Michael George
Phone: 415 604-5881
e-mail: Mike_George@qmgate.arc.nasa.gov
NASA Ames Research Center

Marie S. Noland
Phone: 804 864-5779
e-mail: m.s. noland@larc.nasa.gov
NASA Langley Research Center

Pamela F. Richardson
Phone: 202 358-4631
quick m: PF_Richardson@aeromail.hq.nasa.gov
e-mail: pam@picard.aero.hq.nasa.gov
NASA Headquarters

William R. Van Dalsem
Phone: 415 604-4469
e-mail: vandal@nas.nasa.gov
NASA Ames Research Center

Robert Williams
Phone: 205 544-3998
e-mail: bobw@tyrell.msfc.nasa.gov
NASA Marshall Space Flight Center

Matthew W. Blake
Phone: 415 604-4978
e-mail: blake@nas.nasa.gov
NASA Ames Research Center

Austin L. Evans
Phone: 216 433-5868
e-mail: thevans@lims01.lerc.nasa.gov
NASA Lewis Research Center

Fred W. Martin
Phone: 713 483-4698
e-mail: martin@euler.jsc.nasa.gov
NASA Johnson Space Center

Mark G. Potapczuk
Phone: 216 433-3919
e-mail: tons@frosty.lerc.nasa.gov
NASA Lewis Research Center

Robert E. Smith
Phone: 804 864-5774
e-mail: bobs@geolab1.larc.nasa.gov
NASA Langley Research Center

Robert P. Weston
Phone: 804 864-2149
e-mail: weston@cfd13.larc.nasa.gov
NASA Langley Research Center

Thomas A. Zang
Phone: 804 864-4082
e-mail: t.a.zang@larc.nasa.gov
NASA Langley Research Center

Workshop Chairman
Robert E. Smith

Workshop Administrator
Marie S. Noland
Phone: 804 864-5779
e-mail: m.s.noland@larc.nasa.gov

Conference Proceedings Editor
Mary-Anne K. Posenau
Phone: 804 864-6717
e-mail: m.a.k.posenau@larc.nasa.gov

WORKSHOP ON UNSTRUCTURED GRID GENERATION TECHNIQUES AND SOFTWARE

April 27-28, 1993

CONTENTS

Summary vii

Workshop Motivation and Objectives 1
Pam Richardson, NASA Headquarters

SESSION I
Robert Smith, Chair

Langley Unstructured Grid Generation Review 5
Michael Bockelie

Ames Unstructured Grid Generation Review 19
William Van Dalsem

Lewis Unstructured Grid Generation Review 43
Mark Potapczuk

Marshall Grid Generation Requirements 57
Larry Kiefling

Johnson Grid Generation Requirements 69
Fred Martin

SESSION 2
Yung Choo, Chair

3-D Unstructured Grid Generation Using Local Transformations 79
Tim Barth, NASA Ames Research Center

Status of VGRID/USM3D Aero Analysis System 87
Neal Frink, NASA Langley Research Center

Unstructured Low-Mach Number Viscous Flow Solver 107
Phillip Jorgenson, NASA Lewis Research Center

Robust Unstructured Grid Generation with VGRID 121
Shahyar Pirzadeh, NASA Langley Research Center

Three-Dimensional Unstructured Grid Method Applied to Turbomachinery 137
Oh Kwon, NASA Lewis Research Center

Development of a Gridless CFD Method 151
Jack Batina, NASA Langley Research Center

SESSION 3
Clyde Gumbert, Chair

An Advancing Front Delaunay Triangulation Algorithm Designed for Robustness 163
Dimitri Mavriplis, NASA Langley Research Center

Dynamic Mesh Adaption for 3D Unstructured Grids 181
Rupak Biswas, NASA Ames Research Center

Cartesian-Cell Based Grid Generation/Adaptive Mesh Refinement 193
William J. Coirier, NASA Lewis Research Center

2-D and 3-D Hypersonic Flow with Unstructured Meshes 203
Rajiv Thareja, NASA Langley Research Center

Unstructured Surface Grid Generation 223
Jamshid Abolhassani, NASA Langley Research Center

SESSION 4
Tom Zang

3-D Euler Solution Using Cartesian Grids 251
John Melton, NASA Ames Research Center

Accuracy Assessment for Grid Adaptation 261
Gary Warren, NASA Langley Research Center

Time-Dependent Adaption of Triangular and Tetrahedral Meshes 277
Russ Rausch, NASA Langley Research Center

Computational Geometry Issues 291
Mary-Anne Posenau, NASA Langley Research Center

Standards 311
Matt Blake, NASA Ames Research Center

Surface Acquisition through Virtual Milling 323
Marshall Merriam, NASA Ames Research Center

Attendee List 339

SUMMARY

NASA WORKSHOP ON UNSTRUCTURED GRID GENERATION TECHNIQUES AND SOFTWARE

APRIL 27-28, 1993

1 Introduction

Unstructured grid generation in Computational Fluid Dynamics (CFD) is the discrete representation of flow domains by irregular arrangements of points and cells. Because of the irregularity, explicit connectivity tables are required to define the relationship between neighboring points. Unstructured grids can consist of a wide spectrum of geometric cell shapes, but a large amount of attention has been devoted to the use of triangular cells in two dimensions and tetrahedral cells in three dimensions. Structured grids can be related to a curvilinear coordinate system, and neighboring points and cells are identified by a regular index system; therefore, a connectivity table is not required. In two dimensions, cells are quadrilateral and in three dimensions they are hexahedral. In the event that cells are embedded in a structured grid in an irregular manner, or when there is a large number of irregularly shaped structured-grid blocks covering a domain and a connectivity table is required, then the overall grid is also considered to be unstructured. In this sense, Cartesian grids that intersect an arbitrary body are unstructured grids. The dividing of cells near a curved surface results in an irregular arrangement of points. Also, the dividing of cells to capture flow phenomena, such as a shock or vortex, results in an irregular arrangement of grid points. Triangular/tetrahedral and Cartesian grids are the two primary types of unstructured grids discussed at the workshop.

NASA is also sponsoring the development of multiple-hexahedral-block grid generation software through Small Business Innovative Research (SBIR) contracts. Because of the large number of blocks and possible irregular arrangement of blocks, the grids can be considered to be unstructured. However, these approaches were not discussed at the workshop.

In December 1989, the NASA workshop on Future Directions in Surface Modeling and Grid Generation (NASA CP 10092) was held to assess

U.S. capabilities and to take a first step in improving the focus and pace of NASA surface-modeling and grid-generation efforts. Aerospace industries, universities, Department of Defense, software companies, and NASA centers participated in the workshop. It was recognized that surface modeling and grid generation were the most labor-intensive and time-consuming part of the computational aerospace design. It was noted that virtually all project-oriented CFD at the time utilized patched or overset structured-grid schemes, that surface modeling through CAD was quite advanced but not well coordinated with grid generation, that an aerospace-geometry data exchange standard was needed to improve coordination of U.S. activities, and that unstructured schemes began to produce promising results. Structured grid generation with interactive 3D domain decomposition received special attention at the workshop.

In April 1992, NASA organized and hosted the conference on Software Systems for Surface Modeling and Grid Generation (NASA CP-3143). It was evident that further progress had been made since the 1989 workshop in many areas including the CAD/grid interface, geometry data exchange standard (NASA introduced NASA-IGES for industry feedback), interactive blocked structured grid generation, and surface grid techniques.

The '92 workshop emphasized software systems. A large majority of the presentations and live demonstrations were on structured grids. Noting the considerable efforts were being expended within NASA to develop unstructured grid generation technology, the NASA Surface Modeling and Grid Generation Steering Committee (SMGGSC) organized the current workshop to assess it's unstructured grid activities, improve the coordination among NASA Centers, and promote the technology transfer to industry. The objectives established by the committee for the workshop were:

- Identify unstructured grid generation technology that can be transferred to customers in the short term (two years)

- Identify technical issues on which to focus research

- Communicate between NASA researchers progress being made

- Insure that duplicate R&D is not being performed

The format of the two-day workshop consisted of oral presentations on the first day and discussions on the second day. The oral presentations were center overviews and individual R&D reports. The presenters represented Ames, Langley, and Lewis Research Centers and the Johnson and Marshall Space Flight Centers. During the discussions on the second day, each research paper was critiqued, and the overall unstructured grid generation activity within NASA was evaluated.

During the presentations and discussions, it became evident that inviscid three-dimensional unstructured grid generation and solver technology has progressed rapidly and is beginning to see use within the U.S. aerospace industry. A variety of government and commercial software packages are under development. The NASA/BOEING TRANAIR package appears to be among the most mature packages and sees heavy use in early design. However, limitations in the TRANAIR approach are motivating continued research in adaptive Cartesian Euler schemes. Among the NASA software efforts, the VGRID/USM3D effort is progressing rapidly and is being used within NASA and industry.

At present viscous applications of unstructured grid generation appear to be limited to two-dimensions, with particular progress noted in the high-lift area. In response to this, research at the Ames, Langley, and Lewis centers is focused on improving grid generation and solver technology for viscous applications. Given a two-year time frame, it is highly likely that many of the results of this research will be transferable to NASA customers.

There was no unstructured grid generation research or development activity reported from the space centers; however, the complexity of the configurations under study and the requirement for rapid analysis merits a requirement for robust, user-friendly unstructured grid generation software and corresponding flow-solver software. The space flight centers and the aerospace industry are customers targeted for technology described herein.

The organization of this summary is to first discuss the center overviews followed by the near term development opportunities and the research underway. The summary closes with conclusions and recommendations complimenting the stated workshop objectives.

2 NASA OVERVIEWS

The NASA center overviews are discussed in the order that they were presented.

2.1 Langley Research Center

The Langley overview focused on the unstructured grid generation software available for application at the center. The systems are VGRID, FELISA, TETRA, NGP, and TGRID. The overview pointed out the strengths and weaknesses of their systems and devoted the most attention to the VGRID system under intense development at Langley.

VGRID is a software system for unstructured grid generation based on the advancing front method. It was initiated under an SBIR contract with VIGYAN Inc., Hampton, Virginia in 1988. VGRID is suitable for the generation of Euler grids about complex aerodynamic configuration. It is closely coupled with the USM3D unstructured-grid Euler solver developed at the Langley Research Center. At the present time, there are approximately 40 users of the VGRID/USM3D system throughout the United States. Because of its high level of development, support, and exiting customer base, VGRID/USM3D is highly suitable for aggressive transfer to aerospace customers.. Also, VGRID/USM3D is a potential customer for other NASA unstructured grid generation research results.

The unstructured grid generation research at Langley is concentrated on techniques suitable for viscous flows, adaptive solutions and effective user interfaces with unstructured grid generation software. All of the research at the center, including Gridless CFD, was presented in research papers.

2.2 Ames Research Center

Activities at Ames are focused on the utilization of state-of-the-art unstructured software to solve current aerodynamic design problems and the development of technology required for the next generation of unstructured tools. Engineers at Ames are making extensive use of TRANAIR, FELISA, and AIRPLANE in assisting U.S. aerospace industry with short-lead time design issues. In particular, the automated grid-generation capabilities of the TRANAIR code have allowed its use on many geometrically complex industrial problems. Building on the success of the TRANAIR approach, the TIGER development effort is extending the adaptive Cartesian approach to the solution of the Euler equations. Development activities at Ames are focused on the production of a high Reynolds number viscous capability for use on both vector and parallel supercomputers and utilizing heterogenous computer environments. A mid-term goal is to use hybrid schemes consisting of either prismatic or structured hexahedral elements in the viscous regions and tetrahedra or adaptive Cartesian systems in the inviscid regions. Two hybrid schemes have been prototyped (prismatic/Cartesian and hexahedral/tetrahedra) and the viability of these hybrid approaches and general integration issues is being studied. Viscous tetrahedra technology is being aggressively pursued, with particular emphasis on developing:

- Direct CAD link via NASA-IGES based solids-model interface

- Fully automated viscous surface/volume grid generation

- Adaptive grid generation

- Implicit viscous solvers, including turbulence models

- Efficient parallel implementations

Rapid progress is being made in the development of a solid model CAD data exchange standard (a NASA-wide activity) and the related software for surface interrogation. These CAD-model interface capabilities are being integrated with point insertion/local optimization grid generation technology to allow automated adaptive viscous-grid generation. Progress has also been made in flow adaptive grid modification, and results show the advantage of the unstructured approach for resolution of unsteady off-body flow structures. Extensive efforts in the development of load-balancing schemes for parallel environments have been demonstrated and will allow efficient use of these new systems. Finally, rapid progress is being made in the development of high-order, fully-implicit viscous solvers with efficient implementations on the latest parallel computers. Results were presented for both two- and three-dimensional high-lift applications.

A selection of the research in unstructured grid generation at the Ames Research Center was presented in individual papers.

2.3 Lewis Research Center

Internal flow about complex shapes is the driving force to using unstructured grids at the Lewis Research Center. The Lewis overview covered requirements for unstructured grids at the center, software in use, and research under way. Research relative to Euler and Navier-Stokes solutions using Cartesian based grids and to viscous low Mach number flows on triangular unstructured grids is underway. Also, Lewis is using VGRID/USM3D, which they have modified for turbomachinery flow computations. All of the research at the Lewis Research Center on unstructured grid generation was presented in individual papers.

2.4 Marshall Space Flight Center

The Marshall overview was devoted to the complex domains that the center must analyze. However, no research or development of unstructured grid generation techniques or software suitable for flow computations were reported. Marshall is a potential customer for unstructured grid generation software.

2.5 Johnson Space Flight Center

The Johnson Space Flight Center, like the Marshall Space Flight Center, has tremendous needs for rapid flow analysis about very complex launch configurations. The example of a high Reynolds Navier-Stokes simulation of flow about a very complex Shuttle configurations was used to highlight the strengths and weaknesses of current structured technologies. Specifically, structured techniques require extensive engineering time to generate the surface grids and define the volume topologies for the geometrically complex configurations of interest to the space centers. However, these techniques appear to be capable of accurately predicting very complex flow structures. Also, like Marshall, Johnson is not currently conducting research and development on unstructured grid technology but is a potential customer of the research centers.

3 NEAR-TERM DEVELOPMENT STATUS AND OPPORTUNITIES

The VGRID system has reached an advanced level of development for NASA CFD customers. Since its completion as a SBIR project in 1992, VGRID has been continually upgraded by NASA and VIGYAN Inc. The elapsed time required to prepare initial data and generate unstructured grids about airplane configurations has been reduced from several weeks to several days. Upgrades that are currently in progress will create surface descriptions compatible with NASA/IGES standards. A rapid and robust projection algorithm is being applied to compute surface grids, and a graphical user interface for preparation of the input is near completion. It is anticipated that the elapsed time required to prepare data and acquire satisfactory Euler unstructured grids will be less than two hours. Coupled with the USM3D Euler solver, the combined software system offers end users a complete package for unstructured grid generation and inviscid-compressible flow solution. Training for customers is performed periodically, and a high level of maintenance is provided.

Another area of development is the establishment of standard data formats for unstructured grids and the SUPERPATCH development. Standards will allow the rapid transfer of data between systems and organizations. A standard called NASA/IGES has been established for surface geometries by a subcommittee to the SMGGSC. This standard has been communicated to NASA's industrial customers. A standards subcommittee for unstructured grid data, lead by the Ames Research Center, is underway. This development will be transferred to customers within the two-year time frame.

Many of the research projects at the centers have associated software developments. For instance at Ames, efforts are underway to expand the development TIGER, an adaptive Cartesian Euler grid generation and solver package. There is also a concerted effort at Ames and Langley to develop viscous tetrahedra grid generation/adaptation and solver tools. At Lewis, an effort to extend existing unstructured tools (such as VGRID/USM3D) to the rotating turbomachinery area is being pursued. Within a two year time frame, it is likely that software tools will be incorporated into freestanding programs or incorporated into existing end-user software.

4 RESEARCH STATUS AND OPPORTUNITIES

NASA has a significant ongoing research effort in unstructured grid generation technology. For triangular or tetrahedral grids, this research is mainly directed at two fronts: (1) suitable unstructured grids for viscous flow computation and (2) adaptation methodologies for increased accuracy and reduced computational times. Cartesian grid techniques are also being researched,

however, to a lesser extent than triangular/tetrahedral grids.
For viscous grids, considerable progress is being made with local transformation techniques at Ames and advancing front Delaunay algorithms at Langley. For grid adaptation, a procedure for dynamic grid adaptation combined with an innovative data structure has been developed at Ames. Time dependent grid enrichment and grid coursing is providing interesting insights in research conducted at the Langley Research Center.

Research relative to Cartesian Grids is being conducted at the Ames and Lewis centers. Both endeavors utilize the strong advantage of adaptive Cartesian schemes where neither surface gridding nor volume grid topology definition is required - two of the most difficult challenges facing other approaches. Basic techniques that will allow Cartesian grids to deal with Euler and low Reynolds number Navier-Stokes solutions are a primary objective.

5 CONCLUSIONS AND RECOMMENDATIONS

NASA has a viable program in unstructured grid generation research and development. In addition to the U. S. aerospace industry, NASA is its own customer for user-friendly, robust and efficient unstructured grid generation software. Research results enhance NASA software and software products created in the private sector. The following recommendations are made based on the objectives of the workshop.

- Concentrate on the rapid development, dissemination, and support of emerging NASA unstructured software tools, such as VGRID/USM3D and TIGER

- Concentrate on viscous and adaptive unstructured tetrahedral grid generation research. Target customers and convey research results to them

- Evaluate potential advantages of hybrid, non-tetrahedra, adaptive Cartesian and "gridless" schemes, and within two years downselect to the development of the most promising of these approaches

- Initiate twice-a-year video conferences to increase communications between NASA unstructured grid generation research and development groups

- Develop standards for unstructured grid data

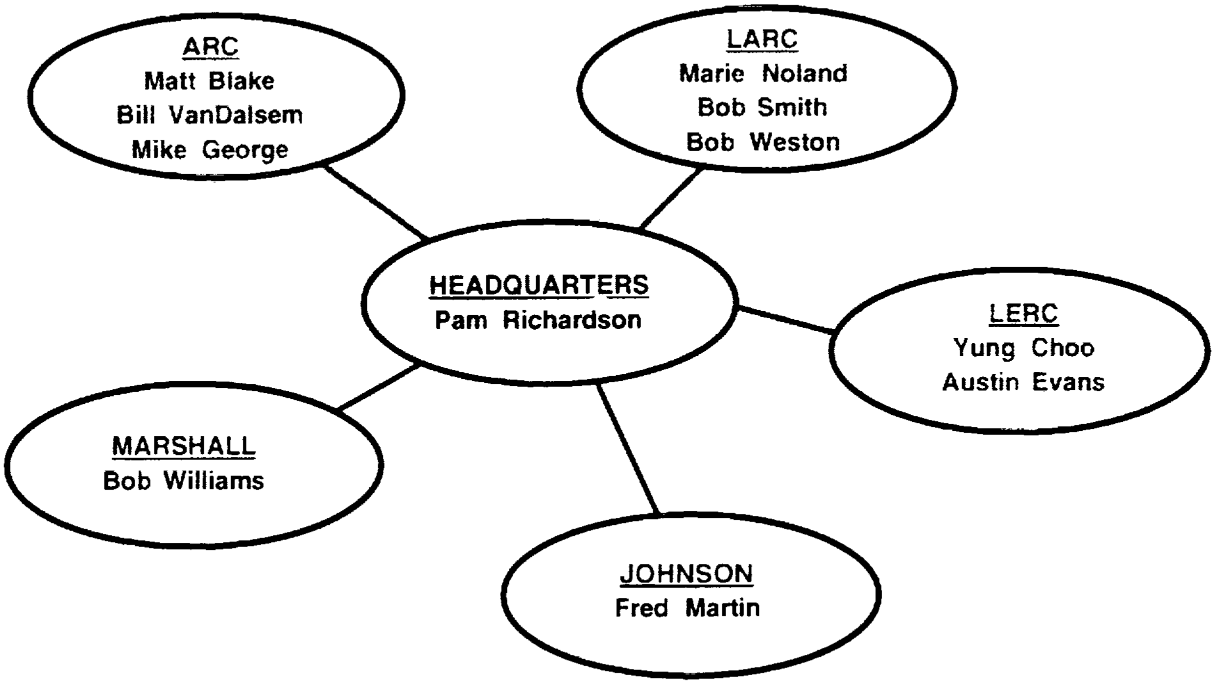
The workshop did not find that there is significant overlap of research and development at the NASA centers. Unstructured viscous grid techniques, adaptation techniques, and Cartesian grid techniques are under study at multiple centers, but different approaches are being pursued. A high level of communications between centers will establish needs and directions as well as insure that duplicate research does not occur.

There is a need to standardize unstructured grid data so that it can be rapidly transferred to different groups and systems. This has proven successful with the NASA-IGES formats for surface descriptions, and there should be similar results for unstructured grid and solution data.

MOTIVATION AND OBJECTIVE

PAMELA F. RICHARDSON
NASA HEADQUARTERS

Steering Committee Members



HEADQUARTERS
Pam Richardson

ARC
Matt Blake
Bill VanDalsem
Mike George

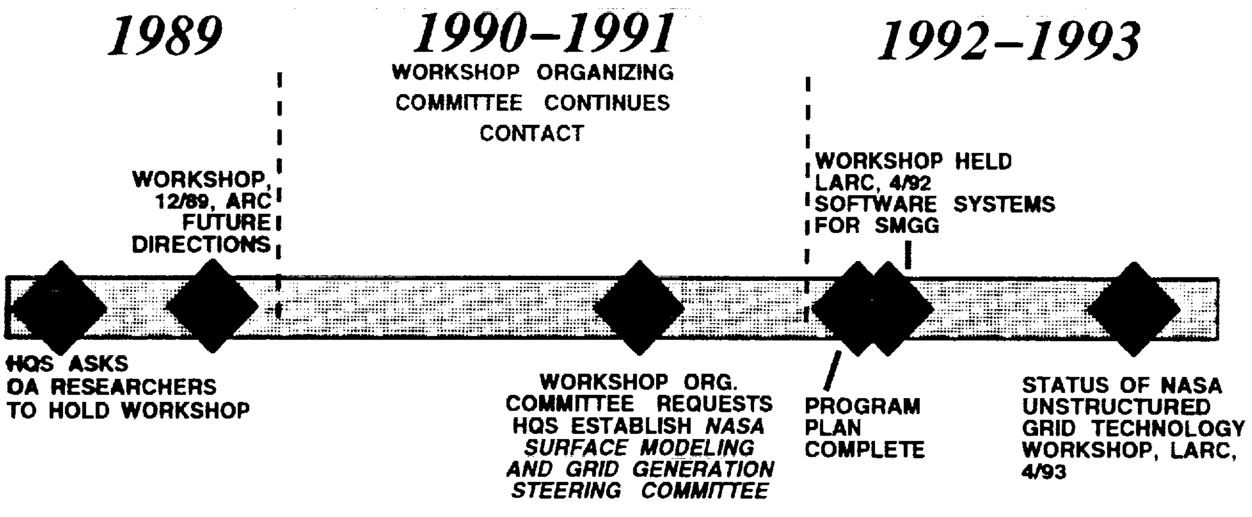
LARC
Marie Noland
Bob Smith
Bob Weston

LERC
Yung Choo
Austin Evans

MARSHALL
Bob Williams

JOHNSON
Fred Martin

Committee activities



1989

1990–1991

1992–1993

WORKSHOP ORGANIZING COMMITTEE CONTINUES CONTACT

WORKSHOP, 12/89, ARC FUTURE DIRECTIONS

WORKSHOP HELD LARC, 4/92 SOFTWARE SYSTEMS FOR SMGG

HQS ASKS OA RESEARCHERS TO HOLD WORKSHOP

WORKSHOP ORG. COMMITTEE REQUESTS HQS ESTABLISH *NASA SURFACE MODELING AND GRID GENERATION STEERING COMMITTEE*

PROGRAM PLAN COMPLETE

STATUS OF NASA UNSTRUCTURED GRID TECHNOLOGY WORKSHOP, LARC, 4/93

Overview

- NASA Surface Modeling and Grid Generation Steering Committee
  - Committee organized to develop and implement a coordinated, customer-focused NASA surface modeling and grid generation program
- Motivation of this meeting:
  - Steering Committee felt importance of understanding the NASA unstructured grid efforts
- Objective of this meeting:
  - Bring together as many NASA unstructured grid researchers as possible to assure understanding among all of the work underway
  - Review among ourselves the work for possible coordinated alignment changes, reduction in any identified overlap work

Wednesday morning's open session

- Purpose is to encourage an open forum where all involved research is reviewed and assessed by those doing the work
  - Every paper/research topic presented on Tuesday will be reviewed on Wednesday
  - Recommendations from colleagues should be considered seriously
- Questions to think about (for each paper):
  - Is the technical approach sound, reasonable, and showing promise?
  - Can the method/code/research shortly (less than 2 years) be used by NASA customers?
  - Is there any overlap with other work underway at NASA, if so, can the work be coordinated, aligned, reduced, stopped?
  - Are there any recommendations to your colleague for modifications to this research?

SOFTWARE SYSTEMS USED FOR UNSTRUCTURED GRID GENERATION AT NASA LANGLEY

MICHAEL J. BOCKELIE
COMPUTER SCIENCES CORPORATION

OVERVIEW

- **Grid Generation Systems For 3D Configurations (Euler Grids)**
  - **VGRID** (NASA / LaRC)
  - **FELISA** (Swansea College, UK)
  - **TETRA** (CDC / ICEM)
  - **NGP** (National Grid Project / Mississippi State University)
  - **TGRID** (Creare / RAMPANT)

- **Special Purpose (Research) Grid Generators**
  - Viscous and Inviscid
  - Solution Adaptive For Steady and Unsteady Flows

CRITERIA

- **User Orientation**
- **Type of Software System**
- **Surface Defintion**
- **Grid Generation Method**
- **User Interface**
- **" Computational Time " to generate 100K Cell Grid**
  - SGI IRIS / 4D with 50 MHz R4000 64 Bit CPU + 128 MB

NEW VGRID

- o **Most Widely Used System For 3D Configurations**
  - User Support / Training + Expert Users Available Locally
  - Tested On Many Configurations

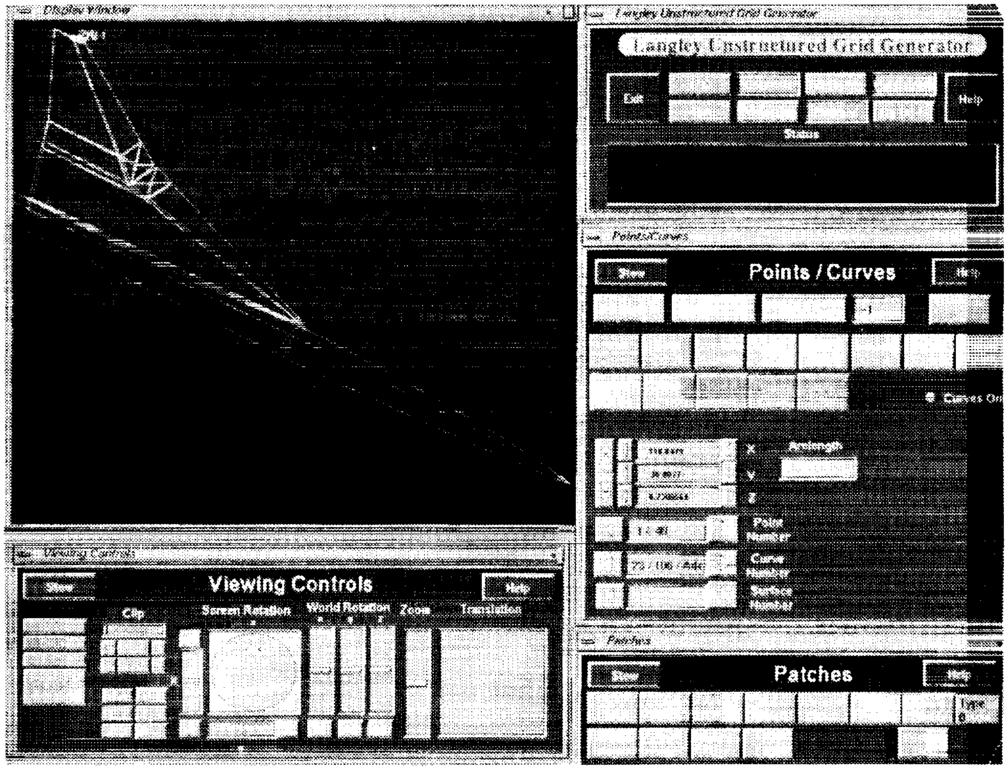
- o **NOT an Integrated System ==> Collection of Individual Codes**
  - Requires User with CFD Training (Engineer)

- o **Surface Definition : NURBS** **! ! *NEW* ! !**
  - INPUT : Point or NURBS Surface Data

- o **Grid Generation Method : Advancing Front** (Lohner, Parikh, Pirzadeh)
  - Node Spacing Data : Point / Line Sources
  - Surface Grid : Generated on Bi – Linear Surface Patch Approximation of Object and then Projected to NURBS Surface.

- o **Graphical User Interface ==> ! ! ! *NEW* ! ! !**
  - Create Surface Patches, Source Terms, Flow Solver BC's
  - " T " Connections for Patches

- o **100K Cell Grid => 12 CPUM**



FELISA

o **Small User Base**
- Limited User Support

o **NOT an Integrated System ==> Collection of Individual Codes**
- Requires CFD Engineer

o **Surface Definition : Networks of Bi-Cubic Hermite Patches**
- INPUT : Point Data

o **Grid Generation Method : Advancing Front** (Morgan & Peraire)
- Node Spacing Data : Point / Line / Triangle Sources
- Surface Grid : Generated on Bi-Cubic Surface in Uniform Parameter Space
  # best looking (prettiest) surface grids in open literature

o **No Graphical User Interface ==> Difficult To Set Up Problems**
- modify VGRID Interface To Output Required Data ?

o **100K Cell Grid => 25 CPUM**

TETRA

o **Very Small User Base for ICEM / TETRA Module**
- Expert Users + Strong Support Locally for other ICEM Modules

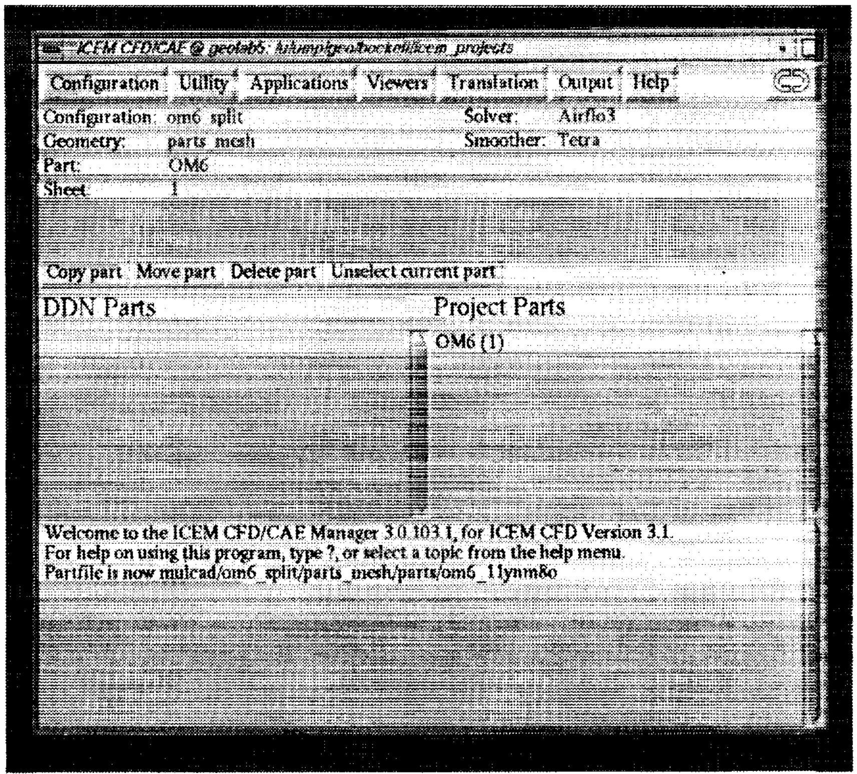
o **Grid Generator Fully Integrated Into CAD / CAE Environment**
- ***Grid Generator Sits On Top Of Full CAD***
- Commercial Grade Software System With Good Customer Support
- Grid Topologies : Unstructured / Structured / Cartesian / Body Fitted Cartesian
- Grid Smoothing, Visualization and Flow Solver Output Modules
- *Oriented For Engineering Technician* (CFD training useful – NOT required)

o **Surface Definition : NURBS**
- INPUT : Point / CAD (IGES) / NURBS Data

o **Grid Generation Method : Octree**
- Node Spacing Data : specify values for surfaces / curves
- Surface Grid : must be **cut** out of volume grid => " noisy " surface grids
  # need to asses if grid quality is adequate for Aerospace CFD

o **User Interface =>** easy to use but can be confusing for non – CAD user

o **100K Cell Grid => 17 CPUM**



ICEM CFD/CAE @ geolab6: h/ump/geo/bockelli/icem_projects

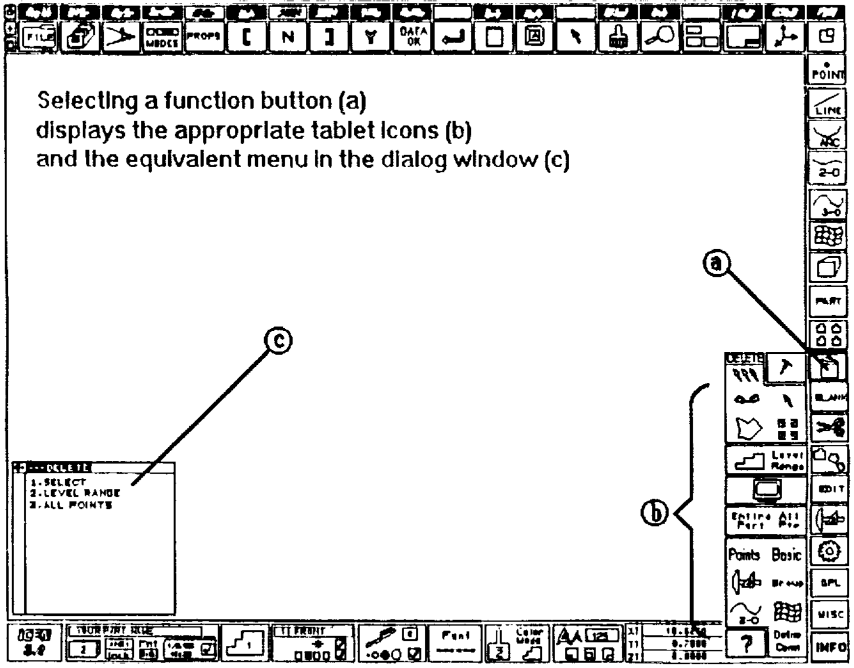
Configuration: om6 split   Solver: Airflo3
Geometry: parts mesh   Smoother: Tetra
Part: OM6
Sheet: 1

DDN Parts   Project Parts

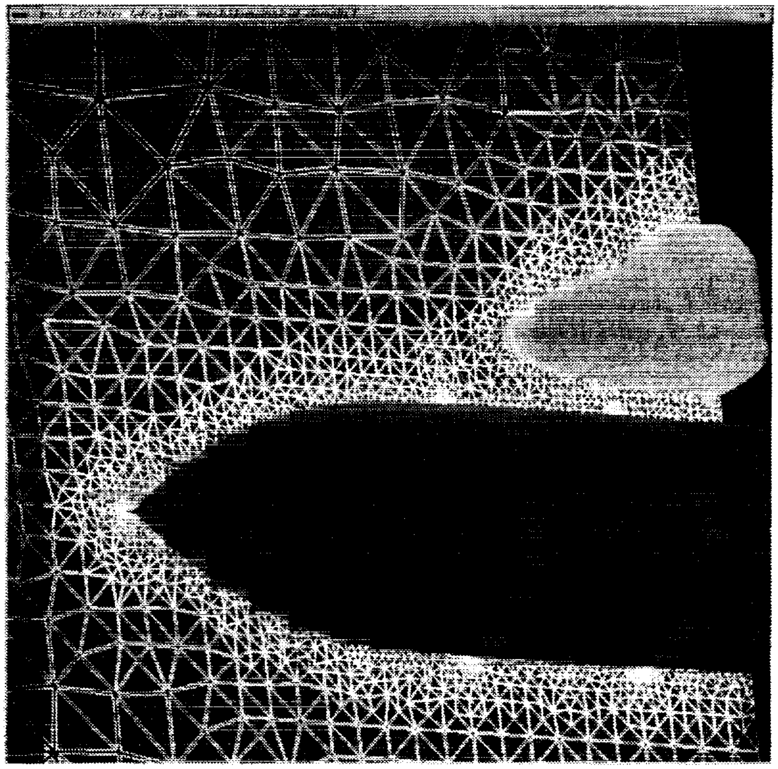
OM6 (1)

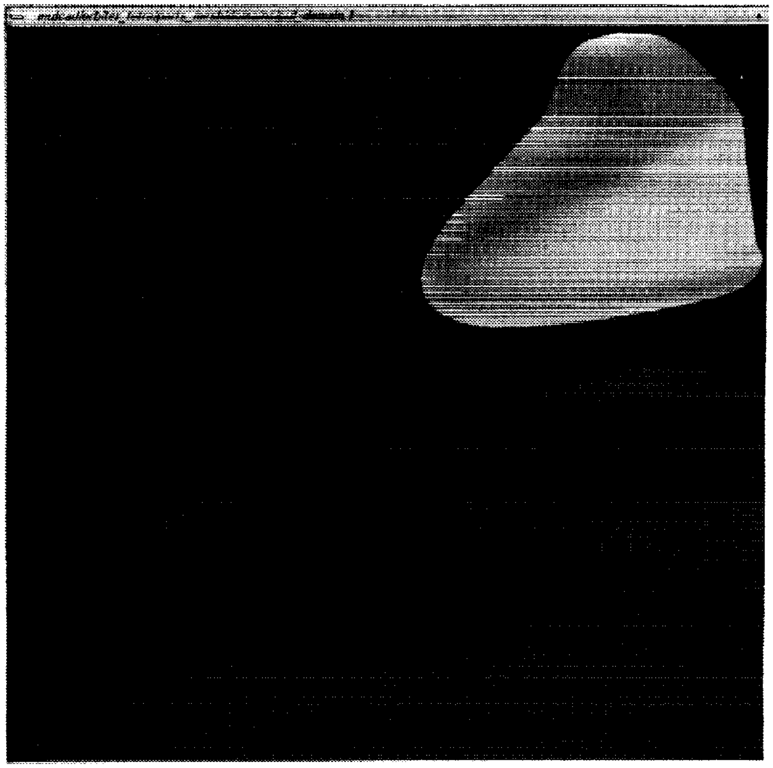
Welcome to the ICEM CFD/CAE Manager 3.0.103.1, for ICEM CFD Version 3.1.
For help on using this program, type ?, or select a topic from the help menu.
Partfile is now mulcad/om6_split/parts_mesh/parts/om6_1lynm8o

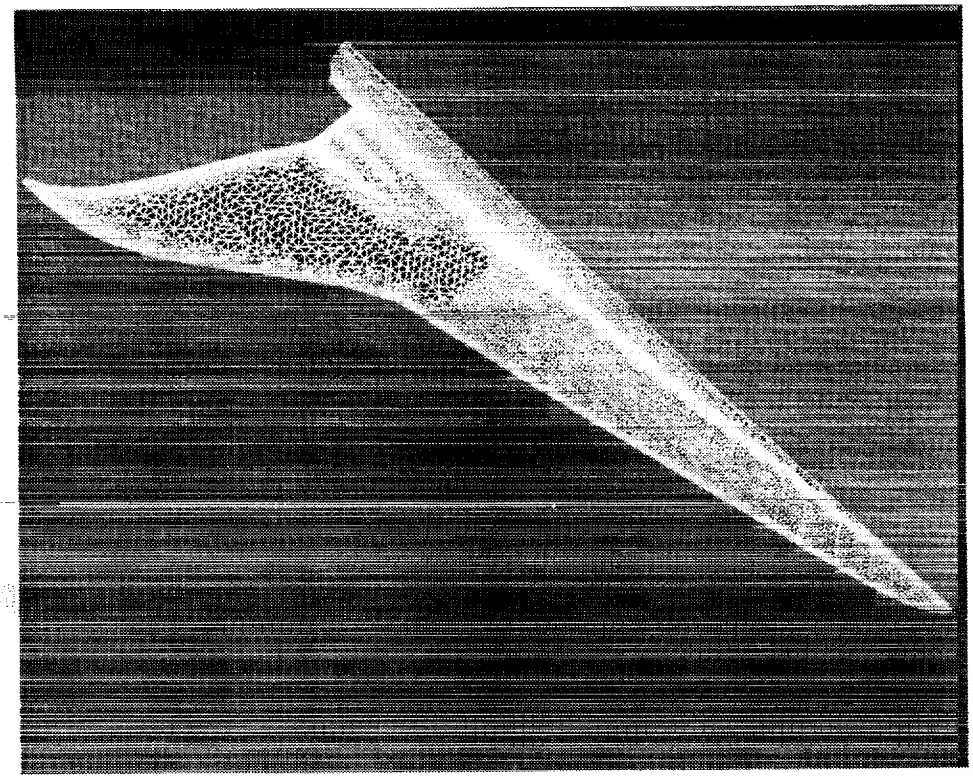
DISPLAYING MENUS

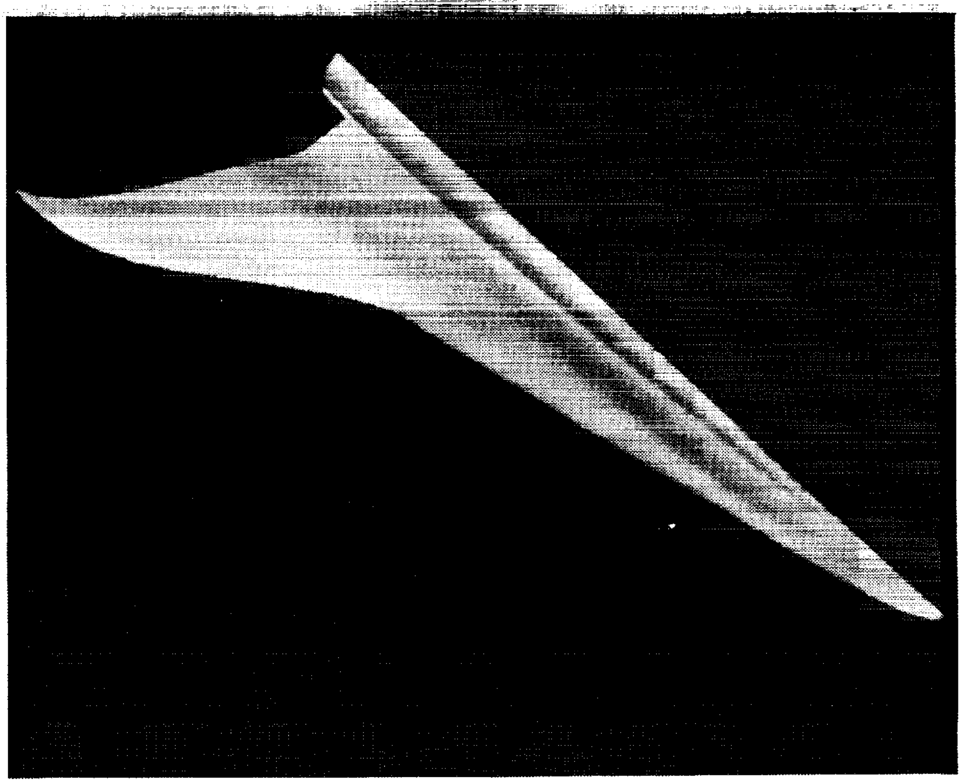


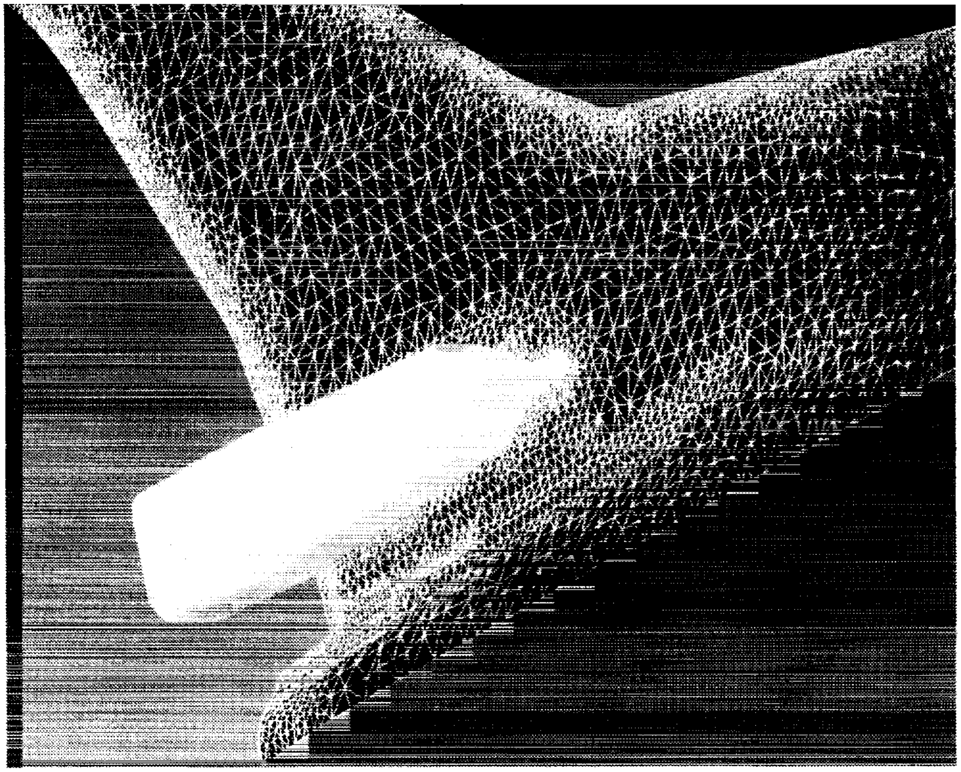
Selecting a function button (a)
displays the appropriate tablet icons (b)
and the equivalent menu in the dialog window (c)

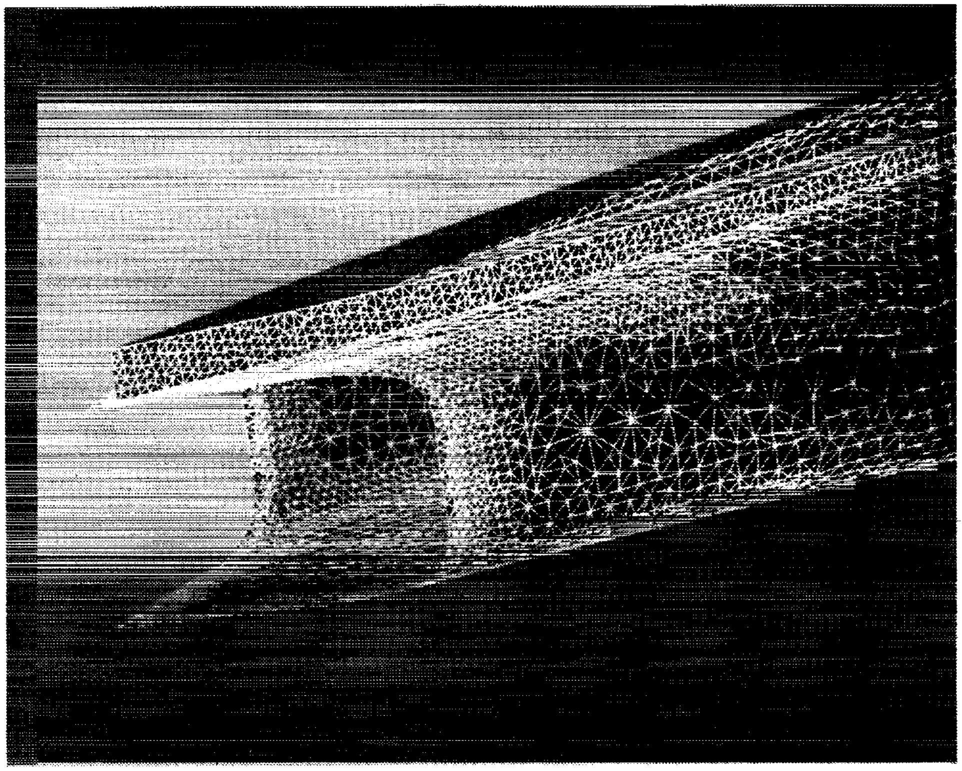


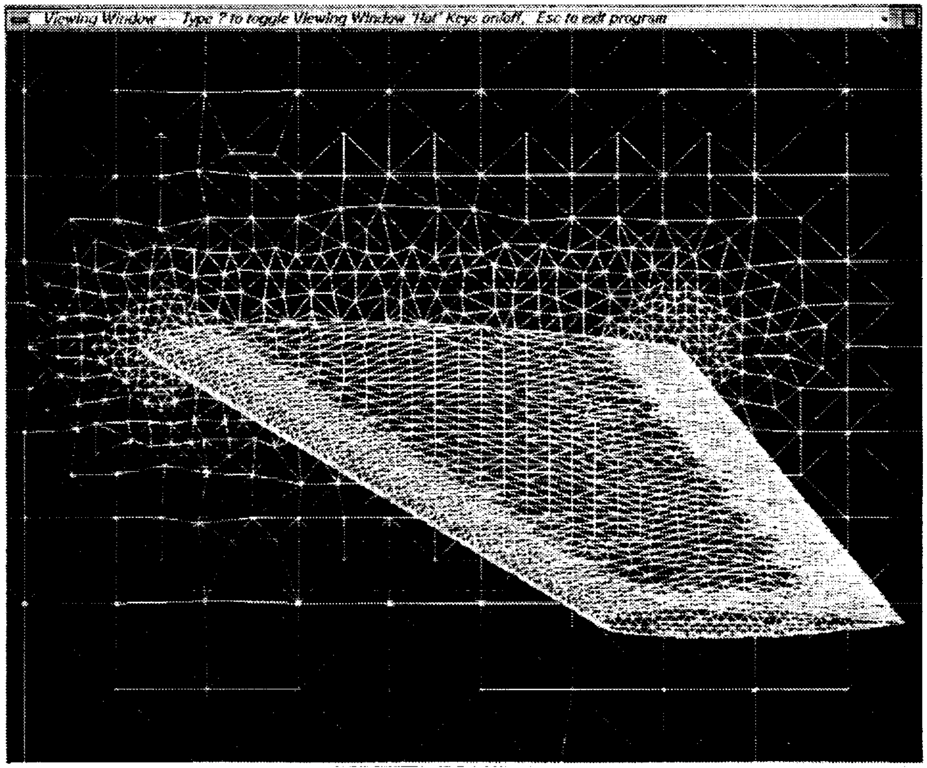


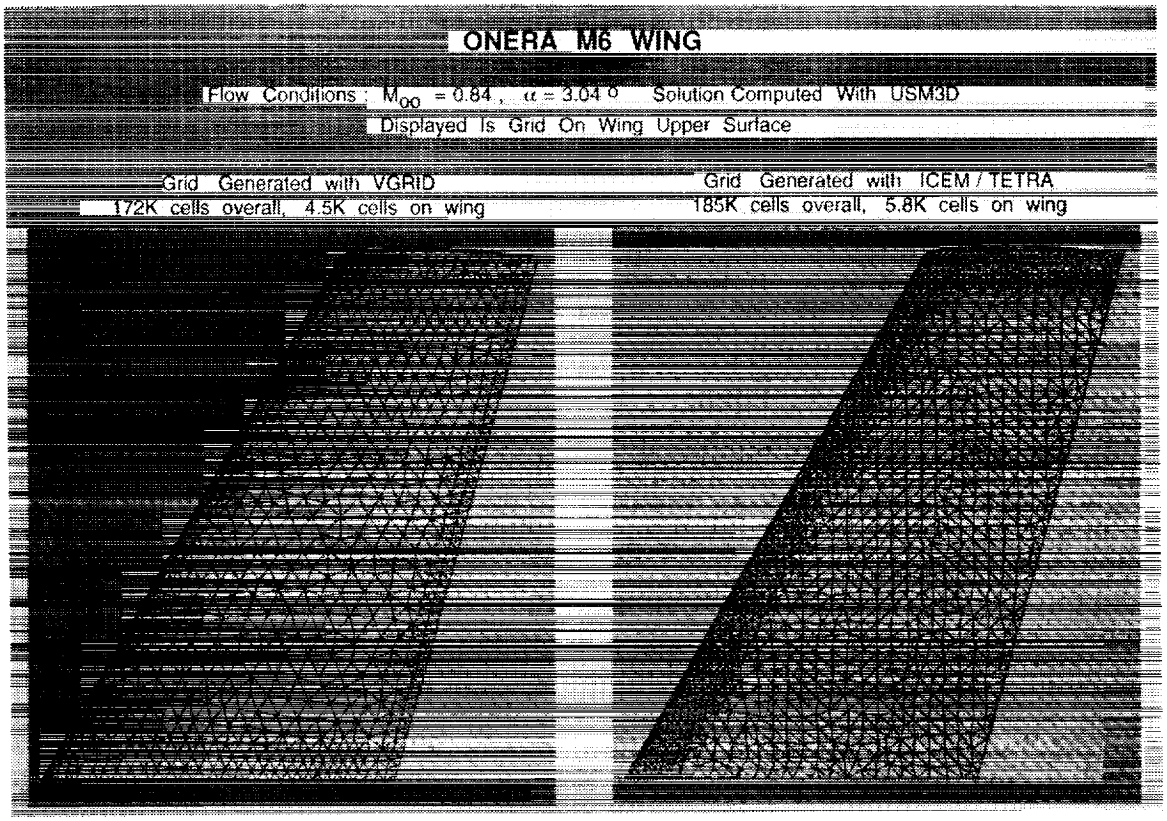









Viewing Window -- Type ? to toggle Viewing Window 'Hot' Keys on/off, Esc to exit program


ONERA M6 WING

Flow Conditions : $M_{oo} = 0.84$, $\alpha = 3.04°$ Solution Computed With USM3D

Displayed Is Grid On Wing Upper Surface

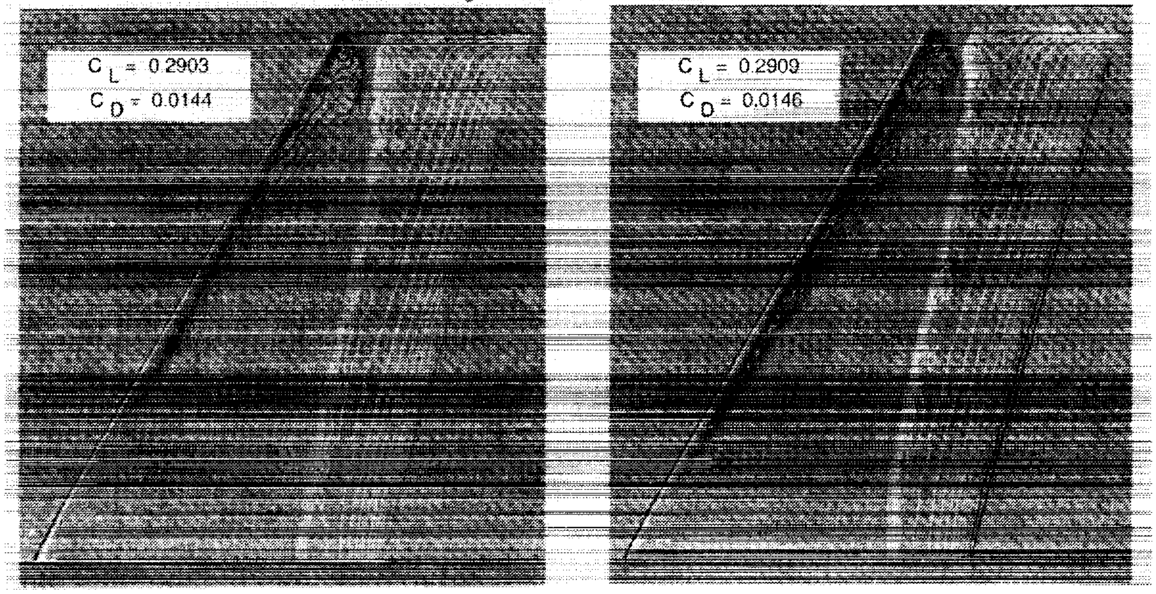
Grid Generated with VGRID
172K cells overall, 4.5K cells on wing

Grid Generated with ICEM / TETRA
185K cells overall, 5.8K cells on wing

ONERA M6 WING

Flow Conditions : $M_{oo} = 0.84$, $\alpha = 3.04^{\circ}$ Solution Computed With USM3D

Displayed Is Normalized Pressure (P / P_{oo}) On Wing Upper Surface (contours: $\Delta P / P_{oo} = 0.02$)

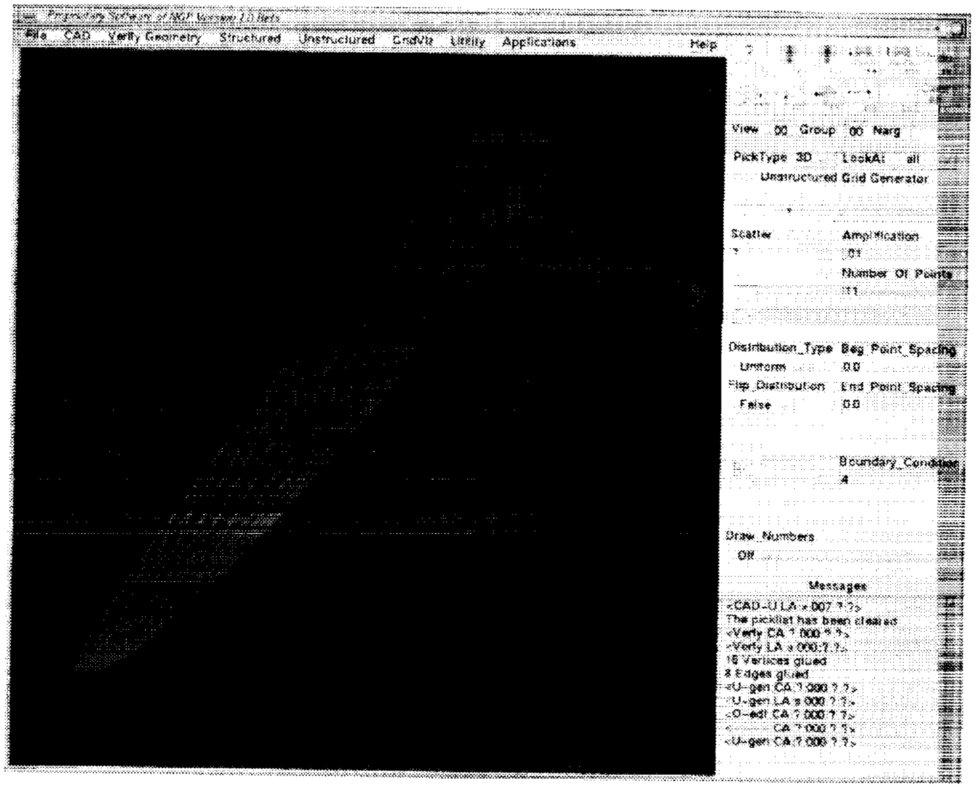


Grid Generated with VGRID
172K cells overall, 4.5K cells on wing

Grid Generated with ICEM / TETRA
185K cells overall, 5.8K cells on wing

NGP

- **Very, Very Small User Base**
  - Code Still In Development => next release in August 1993
- **Fully Integrated Into CAD / CAE Environment**
  - ***Sits On Top of "mini" CAD System***
  - Grid Topologies : Unstructured / Structured (automatic blocking)
  - Grid Visualization and Flow Solver Output Modules
  - *Oriented For Engineering Technician* (CFD training useful – NOT required)
- **Surface Definition : NURBS**
  - INPUT : Point / CAD (IGES) / NURBS Data
- **Grid Generation Method : Delaunay** (Weatherill)
  - Node Spacing : Now => specify distributions on curves, Future => sources (?)
  - Surface Grid : a) generate on NURBS surface using combination of data in physical and uniform parameter space
    b) surface grid must be recovered in final volume grid
- **User Interface => very clean and easy to use**
- **100K Problem => 2 CPUM** (estimated from values reported in literature)



View 00 Group 00 Narg

PickType 3D LookAt all

Unstructured Grid Generator

Scatter ?

Amplification .01

Number Of Points 11

Distribution_Type Uniform

Beg Point_Spacing 0.0

Flip_Distribution False

End Point_Spacing 0.0

Boundary_Condition 4

Draw_Numbers Off

Messages

```
<CAD-U LA ? 007 ? ?>
The picklist has been cleared
<Verify CA ? 000 ? ?>
<Verify LA ? 000 ? ?>
16 Vertices glued
8 Edges glued
<U-gen CA ? 000 ? ?>
<U-gen LA ? 000 ? ?>
<O-edi CA ? 000 ? ?>
<       CA ? 000 ? ?>
<U-gen CA ? 000 ? ?>
```

CONCLUSIONS

- o Wide Variety Of Unstructured Grid Generation Tools Available and In Use At NASA / LaRC

- o VGRID Is Clearly The Most Widely Used Code For 3D Applications

  WHY?
  - customer oriented user support available on site
  - can generate CFD quality grids in " reasonable " time
  - graphical interface available
    - => new interface and improved surface definition will increase use

- o FUTURE

  Tool Requirements :
  - integrated into NURBS based CAD / CAE environment
  - customer oriented and have local support
  - designed for use by non - CFD expert (e.g., engineering tech)
  - simple to use and have user friendly graphical interface
  - provide fast turnaround :
    - => reduce / automate data required for grid generation module
    - => improve grid generation algorithms

VISUALIZATION

- o **General Purpose Grid and Solution Visualization Tools**
  - **FAST**
  - **VPLOT3D**
  - **VISUAL3**
  - **TECPLOT** (surface grids only)
  - **SURFACE** (surface grids only)
  - **DEMAC** (surface grids and advancing front)

  note :
  FAST, VPLOT3D & SURFACE contain visualization tools for grid quality

- o **Special Purpose Grid and Solution Visualization Tools**

SPECIAL PURPOSE GRID GENERATORS

- o **Inviscid**
  - 2D => several codes in use
  - 3D => research codes in development

- o **Viscous**
  - 2D => couple research codes in use
  - 3D => " in development "
    # prismatic element grids being investigated

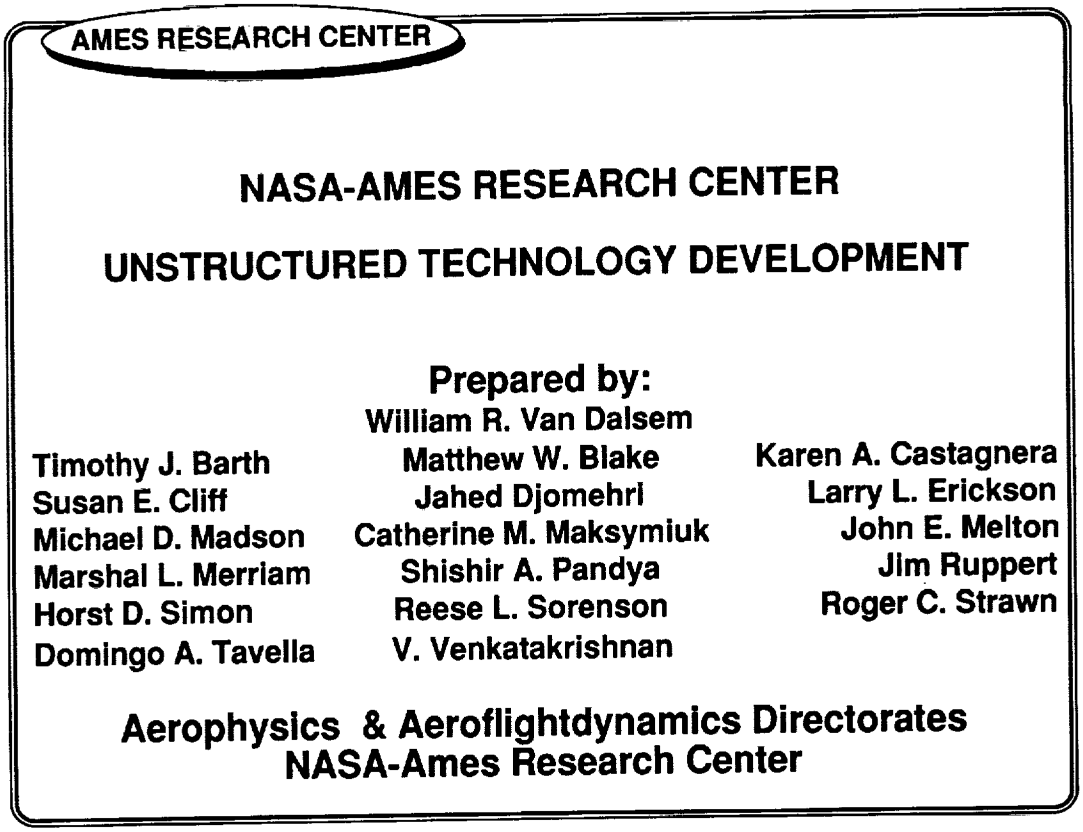
- o **Solution Adaptive**
  - several research codes available for 2D / 3D steady and unsteady flow
    # primarily h refinement and redistribution methods
  - general purpose (production) codes not yet available

TGRID

- **Small (?) User Base**
- **Not A Fully Integrated System**
  - Module Within Creare / RAMPANT Flow Solver System
- **Surface Definition : N / A**
  - ONLY Generates Volume Grid
- **Grid Generation Method : Delaunay** (Blake & Spragle)
  - Node Spacing : computed from given surface grid
  - Surface Grid : a) must be computed in another software package
    b) surface grid must be recovered from final volume grid
    c) volume grid highly dependent on quality of surface grid
- **User Interface => ?**
- **100K Cell Grid => 4 (?) CPUM** (estimated from values reported in literature)

NASA-AMES RESEARCH CENTER UNSTRUCTURED TECHNOLOGY DEVELOPMENT

WILLIAM R. VAN DALSEM
NASA AMES RESEARCH CENTER



AMES RESEARCH CENTER

NASA-AMES RESEARCH CENTER

UNSTRUCTURED TECHNOLOGY DEVELOPMENT

Prepared by:

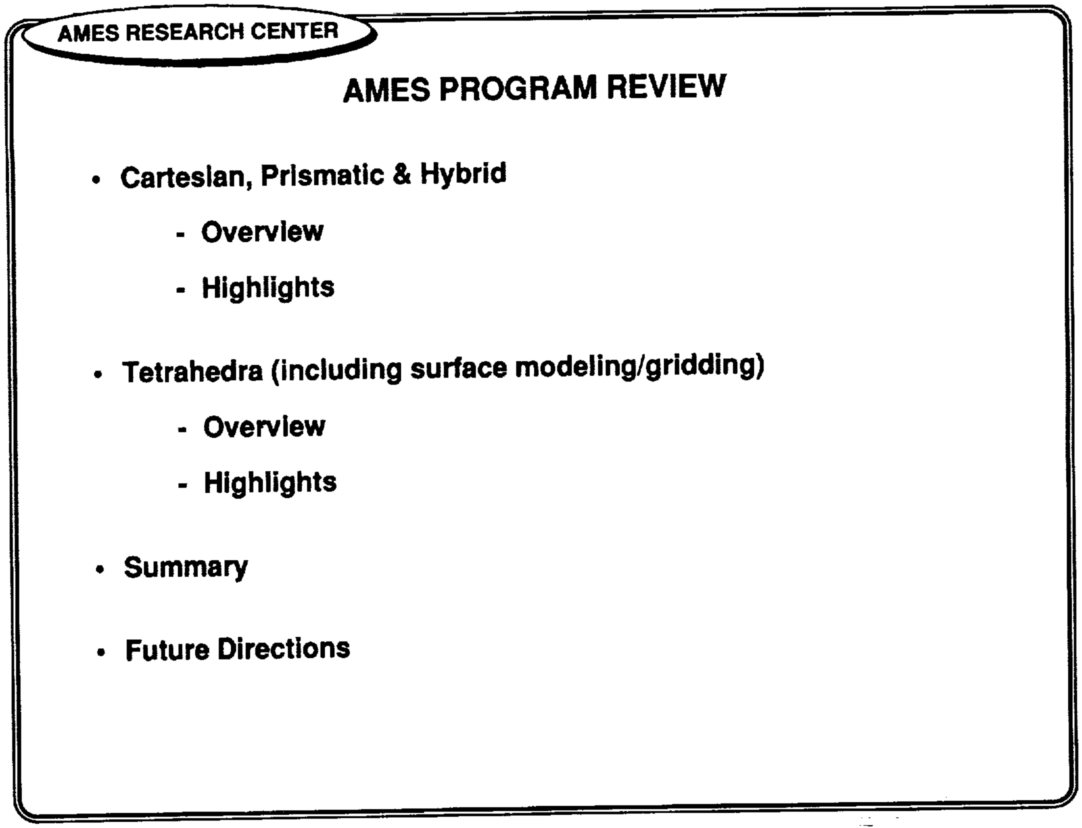
William R. Van Dalsem

Timothy J. Barth
Susan E. Cliff
Michael D. Madson
Marshal L. Merriam
Horst D. Simon
Domingo A. Tavella

Matthew W. Blake
Jahed Djomehri
Catherine M. Maksymiuk
Shishir A. Pandya
Reese L. Sorenson
V. Venkatakrishnan

Karen A. Castagnera
Larry L. Erickson
John E. Melton
Jim Ruppert
Roger C. Strawn

Aerophysics & Aeroflightdynamics Directorates
NASA-Ames Research Center



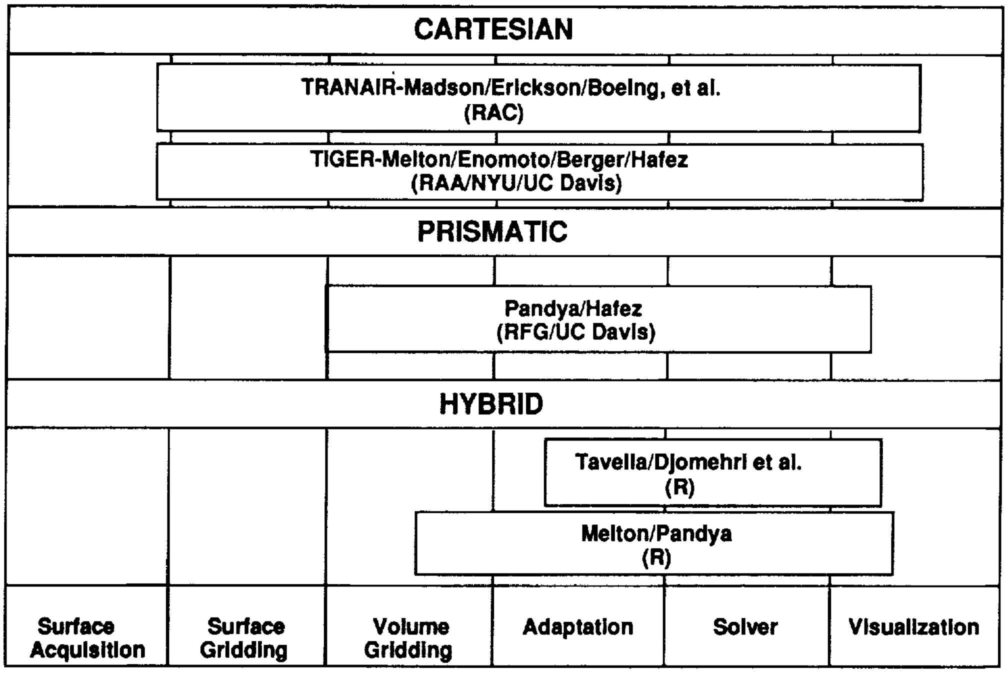
AMES RESEARCH CENTER

AMES PROGRAM REVIEW

- Cartesian, Prismatic & Hybrid
  - Overview
  - Highlights
- Tetrahedra (including surface modeling/gridding)
  - Overview
  - Highlights
- Summary
- Future Directions

AMES RESEARCH CENTER

OVERVIEW OF CARTESIAN, PRISMATIC, & HYBRID ACTIVITIES



CARTESIAN

TRANAIR-Madson/Erickson/Boeing, et al.
(RAC)

TIGER-Melton/Enomoto/Berger/Hafez
(RAA/NYU/UC Davis)

PRISMATIC

Pandya/Hafez
(RFG/UC Davis)

HYBRID

Tavella/Djomehri et al.
(R)

Melton/Pandya
(R)

Surface Acquisition | Surface Gridding | Volume Gridding | Adaptation | Solver | Visualization

AMES RESEARCH CENTER

TRANAIR
TRANSONIC ANALYSIS CODE FOR ARBITRARY CONFIGURATIONS

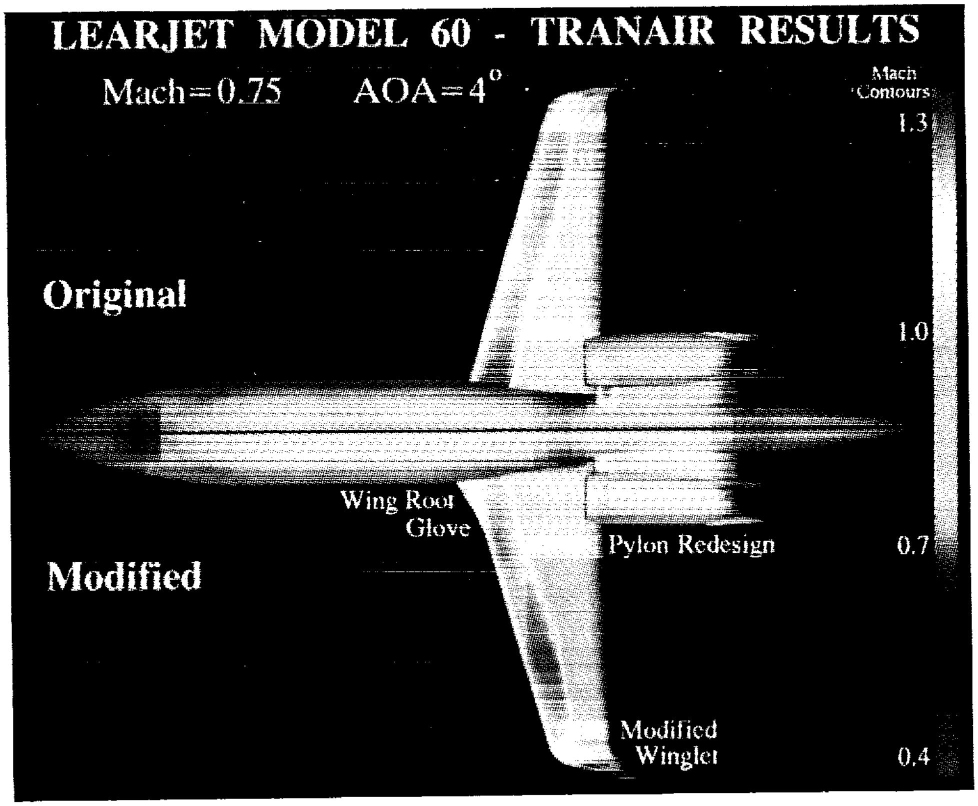
MADSON, ERICKSON, BOEING (JOHNSON), et al.

OBJECTIVE
- Develop and validate an aerodynamic analysis and design capability which eliminates the use of surface-conforming grids

TECHNICAL APPROACH
- Embed surface panel model in a uniform Cartesian gird
- Local grid refinement based on surface model, flow gradients, or user input
- Finite-element non-linear full-potential operators applied and solved iteratively
- Coupled three-dimensional finite-difference boundary-layer code

STATUS
- Extensive NASA and U.S. Aerospace Industry user base: Boeing, Grumman, Learjet, Beech, Gulfstream, etc...

FUTURE DIRECTIONS
- Complete validation of viscous capability



LEARJET MODEL 60 - TRANAIR RESULTS

Mach=0.75 AOA=4°

AMES RESEARCH CENTER

TIGER
AUTOMATED 3D CARTESIAN GRID GENERATION AND EULER FLOW SOLUTIONS

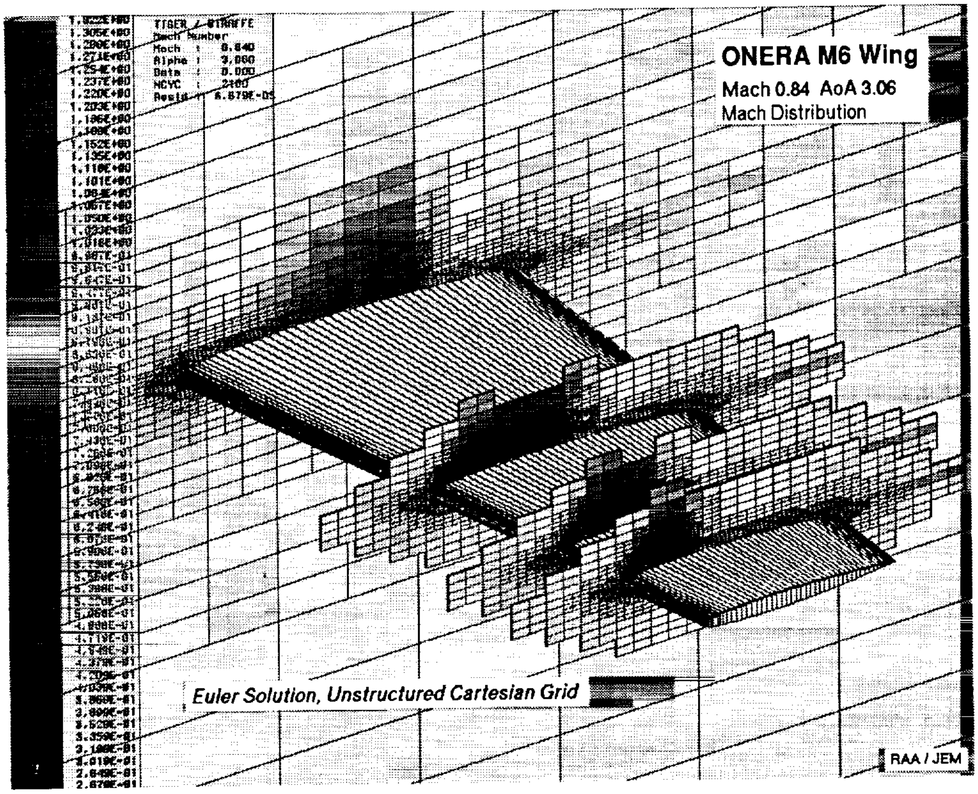
MELTON, ENOMOTO, BERGER, & HAFEZ

OBJECTIVE
- Complete automation of Cartesian Euler grid generation and flow simulation for arbitrary 3D NURBS geometries

TECHNICAL APPROACH
- Automated Cartesian 3D body-intersecting grid generation using NURBS CAD/CAM database and DTNURBS evaluation routines
- Modified Jameson finite-volume Euler flow solver

STATUS
- Developing complete NURBS/IGES input capability
- Improving flux/dissipation calculations
- Integrating "intelligent" feature-based and automated refinement grid generation capabilities

FUTURE DIRECTIONS
- Continued development towards a completely automated adaptive Euler flow simulation capability

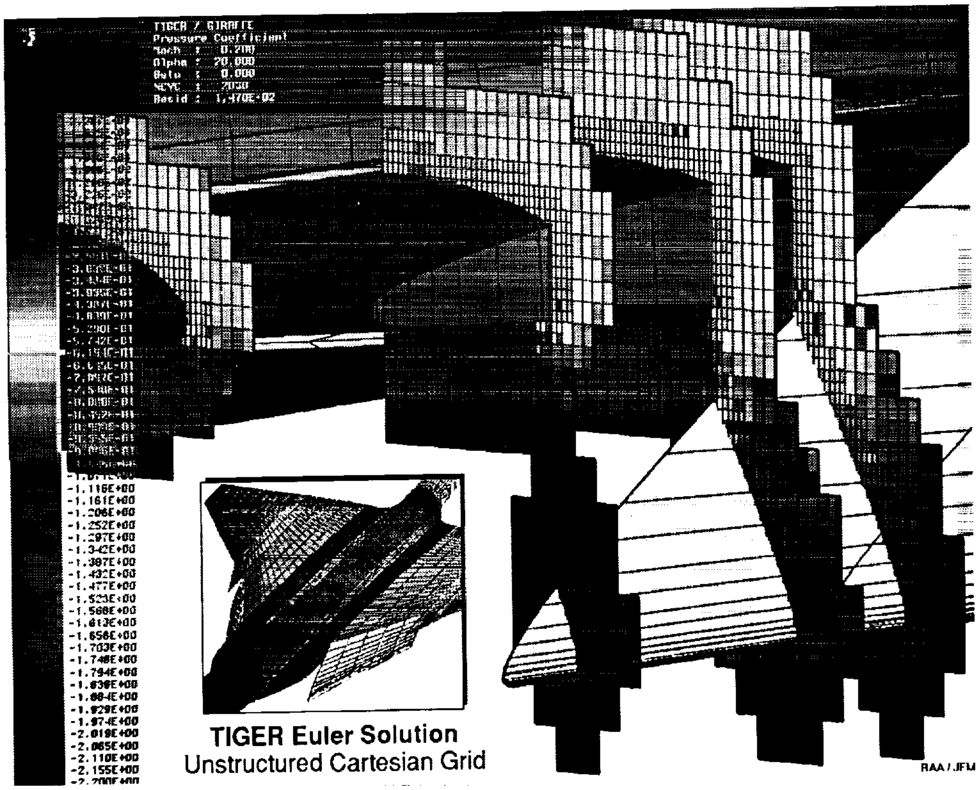


ONERA M6 Wing

Mach 0.84 AoA 3.06

Mach Distribution

Euler Solution, Unstructured Cartesian Grid



TIGER Euler Solution

Unstructured Cartesian Grid

AMES RESEARCH CENTER

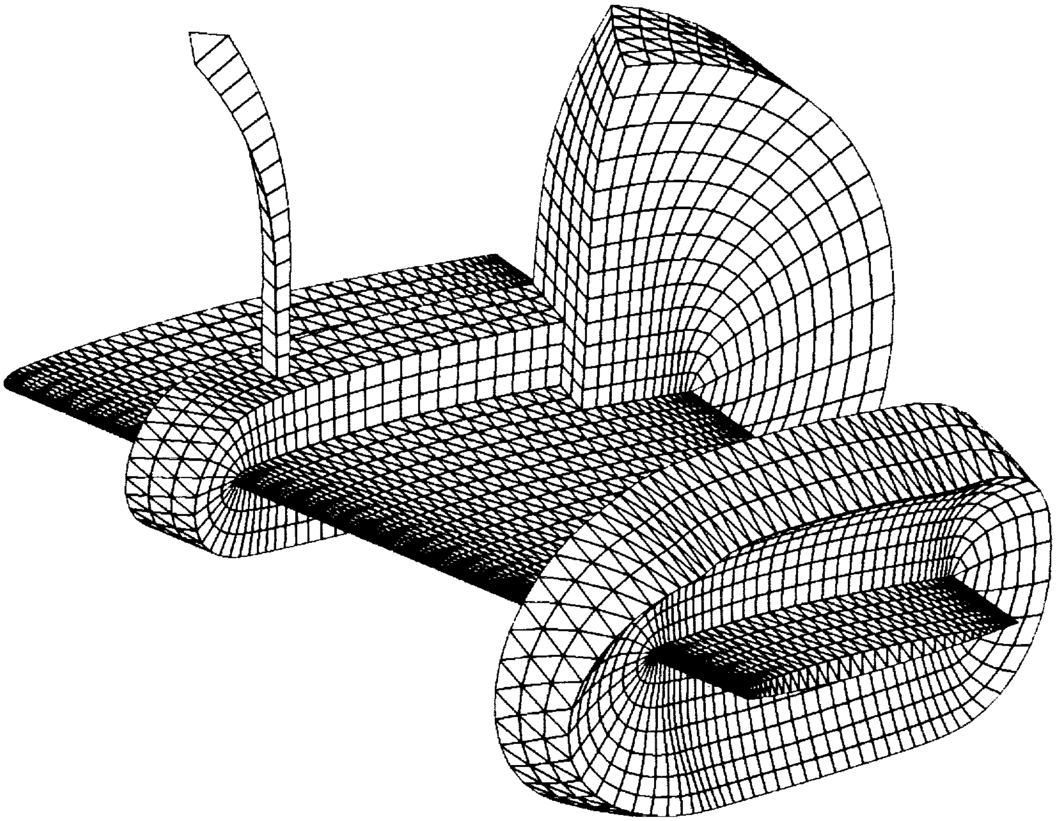
PRISMATIC GRID GENERATION/FLOW SOLVER
PANDYA & HAFEZ

OBJECTIVE
- Explore feasibility of prismatic grid/solver technology for use in hybrid schemes (combine with overset structured, tetrahedra, or Cartesian)

TECHNICAL APPROACH
- Use hyperbolic structured grid technology (Steger et al.) to "grow" volume grids from surface triangularization
- Developing semi-implicit solvers

STATUS
- Explicit hyperbolic volume grid generator complete
- Hybrid grid scheme (prismatic/Cartesian) prototyped
  - Simplified grid generation and low memory requirements
- Semi-implicit inviscid solver in development

FUTURE DIRECTIONS
- Develop implicit hyperbolic volume grid generator
- Develop semi-implicit viscous solver



AMES RESEARCH CENTER

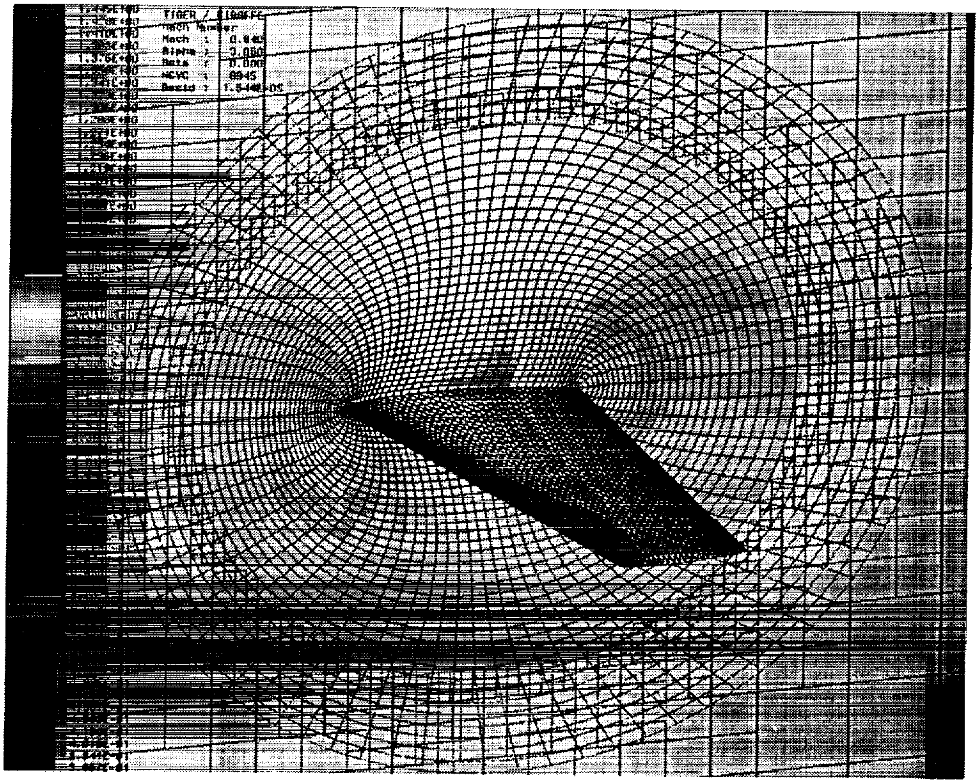
HYBRID PRISMATIC/CARTESIAN GRID GENERATION /SOLVER
MELTON, PANDYA & STEGER

OBJECTIVE
- Explore hybrid prismatic/Cartesian grid/solver technology

TECHNICAL APPROACH
- Combine prismatic near-body grid with outer Cartesian grid using a hybrid Chimera technique
- Solve Euler equations via modified Jameson finite-volume solver

STATUS
- Demonstrated Euler solutions about ellipsoid and ONERA M6 wing

FUTURE DIRECTIONS
- Continued development of prismatic and Cartesian grid generation/solver technology before further hybrid work pursued
  - Semi-implicit prismatic Navier-Stokes solver
  - Improved Cartesian grid adaptation



AMES RESEARCH CENTER

HYBRID STRUCTURED/UNSTRUCTURED NAVIER-STOKES
TAVELLA, DJOMEHRI, KISLITZIN, BLAKE, & ERICKSON

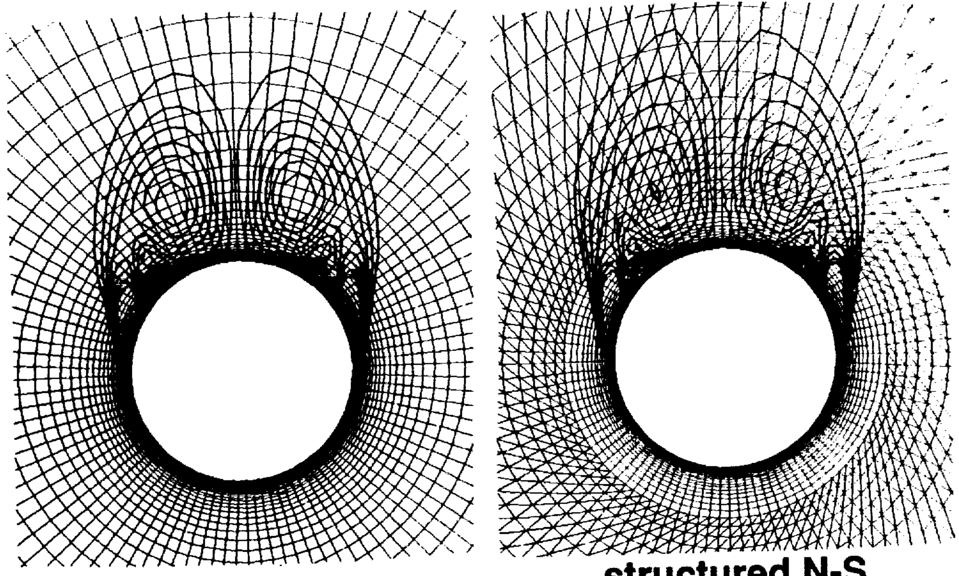
OBJECTIVE
- Explore hybrid structured/unstructured grid/solver technology

TECHNICAL APPROACH
- Combine structured near-body grid/solver with outer unstructured grid/solver
- Couple highly-developed structured/unstructured solvers with minimum modification using sockets programming
- Each solver execute separately as a UNIX process

STATUS
- Demonstrated hybrid Euler-unstructured/Navier-Stokes-structured simulation of high-angle-of-attack flow

FUTURE DIRECTIONS
- Upgrade solvers
- Explore heterogeneous environments

VORTICITY DENSITY
MISSILE AT 30 DEGREES, MACH 0.2

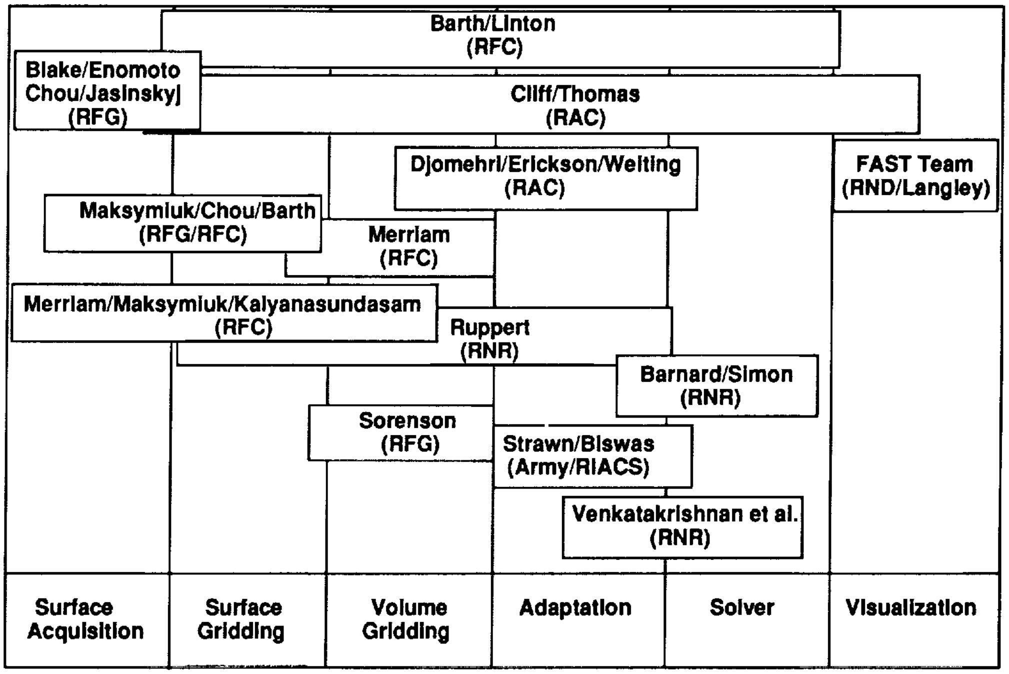


structured N-S

structured N-S
+
unstructured Euler

AMES RESEARCH CENTER

OVERVIEW OF TETRAHEDRA ACTIVITIES



Barth/Linton (RFC)

Blake/Enomoto Chou/Jasinskyj (RFG)

Cliff/Thomas (RAC)

Djomehri/Erickson/Welting (RAC)

FAST Team (RND/Langley)

Maksymiuk/Chou/Barth (RFG/RFC)

Merriam (RFC)

Merriam/Maksymiuk/Kalyanasundasam (RFC)

Ruppert (RNR)

Barnard/Simon (RNR)

Sorenson (RFG)

Strawn/Biswas (Army/RIACS)

Venkatakrishnan et al. (RNR)

Surface Acquisition | Surface Gridding | Volume Gridding | Adaptation | Solver | Visualization

AMES RESEARCH CENTER

SURFACE DEFINITION THROUGH VIRTUAL MILLING
MERRIAM, MAKSYMIUK, & KALYANASUNDARAM

OBJECTIVE

- Develop an automated 3-D laser digitizer capability to obtain an accurate surface representation of an aircraft model for use in CFD simulations

TECHNICAL APPROACH

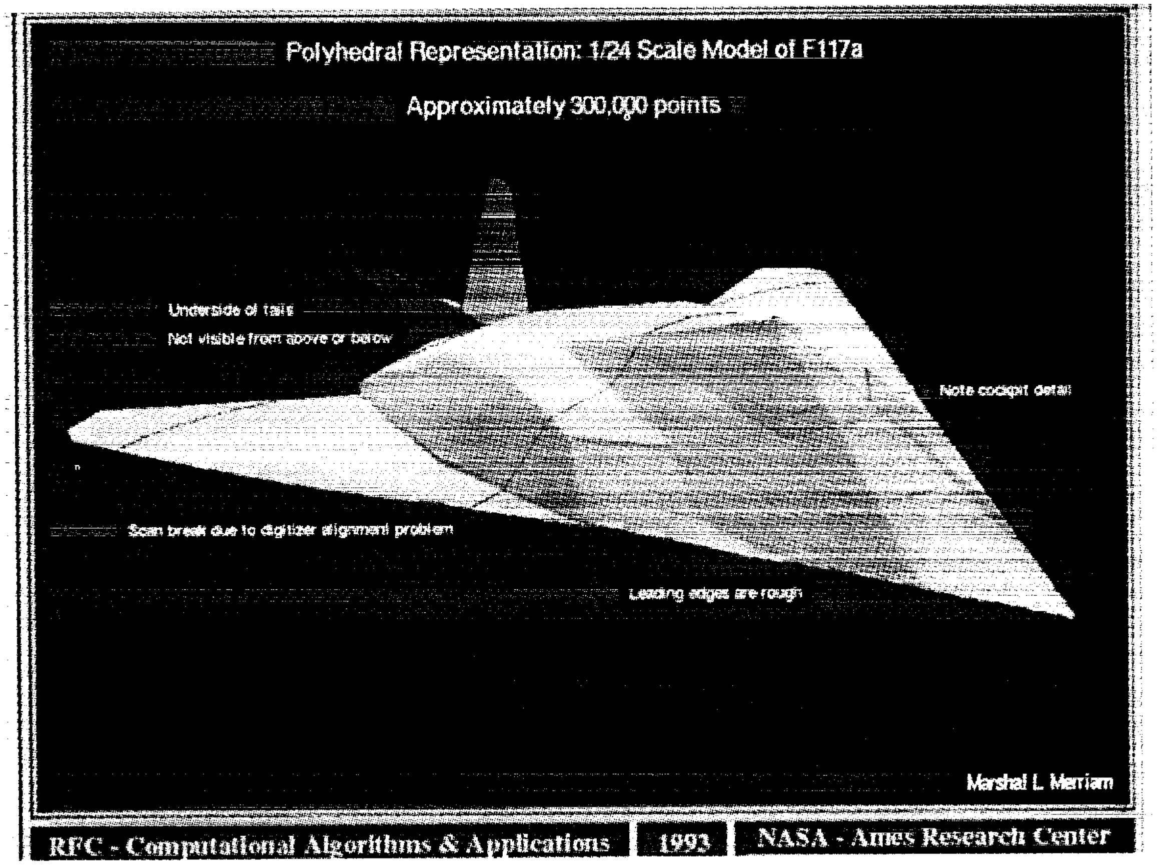
- A 3-D laser digitizer system is used to acquire a rich (~300,000 pts.) and accurate definition of the model surface
- Surface measurements are converted to a polyhedral representation of the model using a virtual milling algorithm
- Unstructured surface grid is generated from acquired polyhedral surface model

STATUS

- An arbitrary number of scans can be combined to produce a polyhedral surface model

FUTURE DIRECTIONS

- Developing a geometry adaptive algorithm for development of optimal surface model and grid
- Integrate with volume gridding/solver technology (Barth, et al.)



Polyhedral Representation: 1/24 Scale Model of F117a

Approximately 300,000 points

Underside of tails
Not visible from above or below
Note cockpit detail
Scan break due to digitizer alignment problem
Leading edges are rough

Marshal L Merriam

RFC - Computational Algorithms & Applications | 1993 | NASA - Ames Research Center

AMES RESEARCH CENTER

SUPERPATCH
BLAKE, ENOMOTO, CHOU, & JASINSKYJ

OBJECTIVE

- Allow acquisition of surface models from diverse sources and:
  - Modest repair and editing of surfaces
  - Addition of patch-topology information (automated and interactive)
- Output B-Rep/SUPERPATCH solid model which contains all surface information for automated surface gridding

TECHNICAL APPROACH

- Define NASA-IGES and SUPERPATCH (IGES B-Rep) standards
- Develop automated software library for:
  - I/O and interrogation of all NASA-IGES entities
  - Convert all NASA-IGES entities to NURBS
  - Add patch-topology information (with interactive back-up)

STATUS:

- NASA-IGES/SUPERPATCH standards proposed (NASA-wide activity)
- NASA-IGES I/O and interrogation library near completion

FUTURE DIRECTIONS:

- Develop automated patch-topology definition techniques

AMES RESEARCH CENTER

AUTOMATED SURFACE GRIDDING FROM CAD MODEL
MAKSYMIUK, CHOU, & BARTH

OBJECTIVE

- Develop automated unstructured surface grid generation technology:
  - Surface/solution adaptive clustering
  - NASA-IGES and SUPERPATCH I/O

TECHNICAL APPROACH

- Combine:
  - NIGES/SUPERPATCH I/O and interrogation functions
  - Barth's surface grid generation (incremental insertion with local optimization, geometric error minimization, and quality repair)

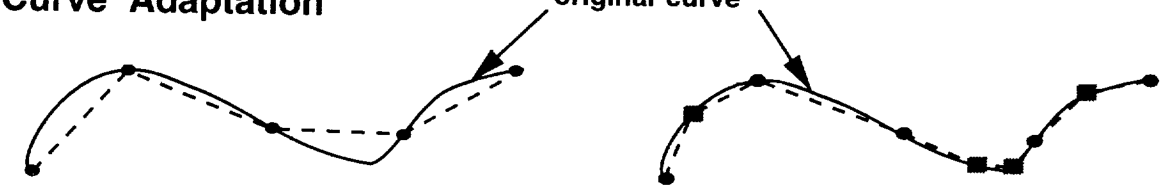
STATUS

- SUPERPATCH integrated with surface grid generator
- Surface gridding with geometric/quality adaptation off IGES B-Rep models accomplished, awaiting additional B-Rep models

FUTURE DIRECTIONS

- Integrate NIGES to allow gridding off NASA-IGES data
- Add solution adaptation capabilities
- Develop completely patch-independent gridding

Curve Adaptation



original curve

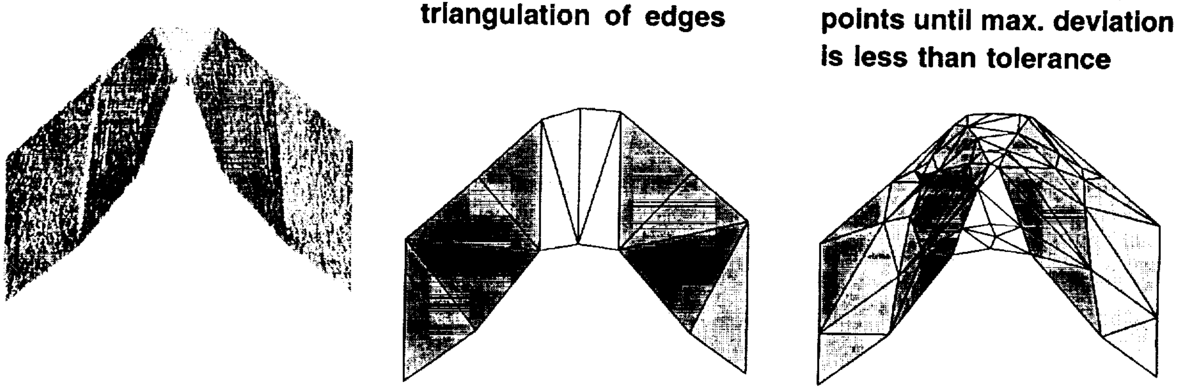
initial distibution

equal arc–length spacing

final distribution

■ marks points added to reduce deviation below tolerance

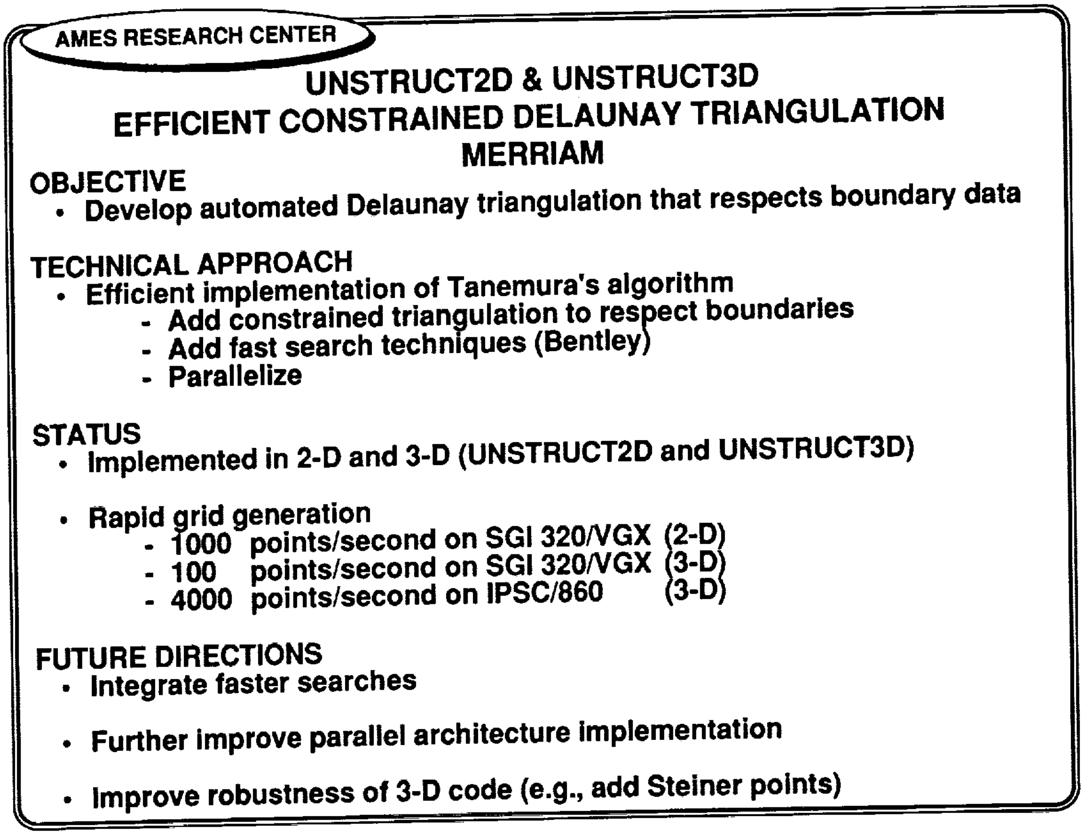
Surface Adaptation



original surface

initial grid, obtained from triangulation of edges

final grid, after adding points until max. deviation is less than tolerance



AMES RESEARCH CENTER

UNSTRUCT2D & UNSTRUCT3D
EFFICIENT CONSTRAINED DELAUNAY TRIANGULATION
MERRIAM

OBJECTIVE

- Develop automated Delaunay triangulation that respects boundary data

TECHNICAL APPROACH

- Efficient implementation of Tanemura's algorithm
    - Add constrained triangulation to respect boundaries
    - Add fast search techniques (Bentley)
    - Parallelize

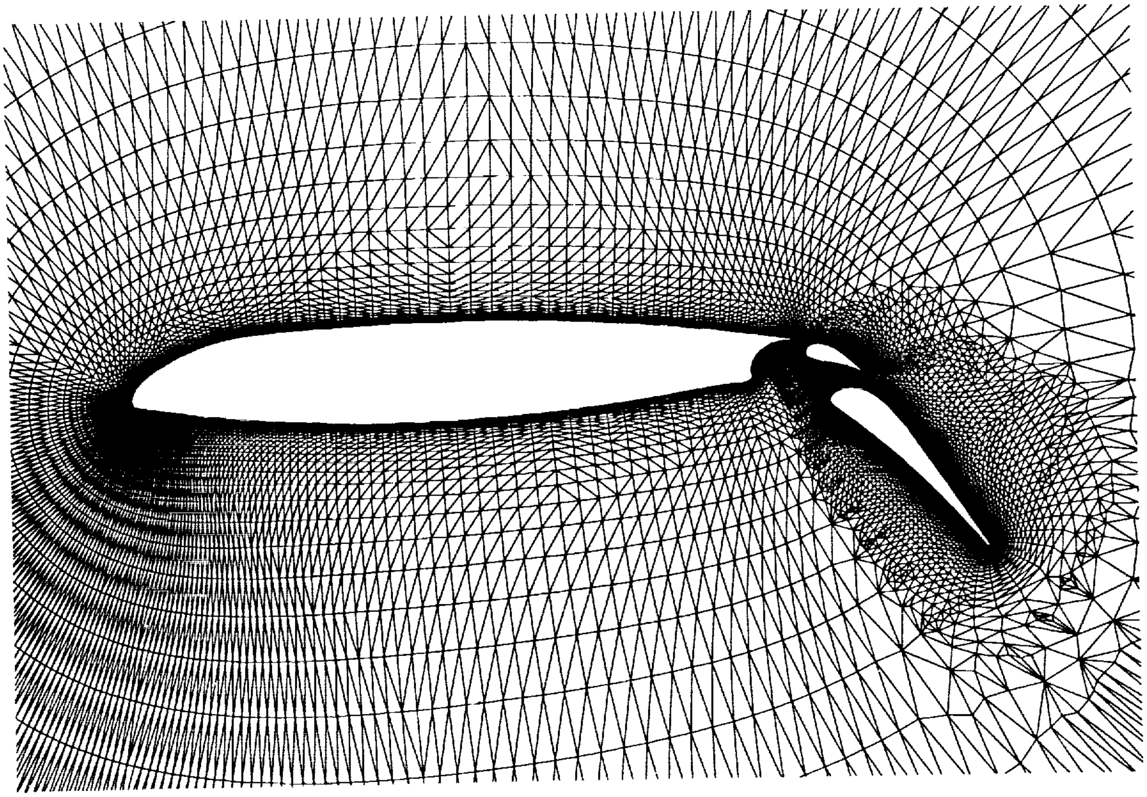
STATUS

- Implemented in 2-D and 3-D (UNSTRUCT2D and UNSTRUCT3D)
- Rapid grid generation
    - 1000 points/second on SGI 320/VGX (2-D)
    - 100 points/second on SGI 320/VGX (3-D)
    - 4000 points/second on IPSC/860 (3-D)

FUTURE DIRECTIONS

- Integrate faster searches
- Further improve parallel architecture implementation
- Improve robustness of 3-D code (e.g., add Steiner points)

One View Of The Completed Triangulation



AMES RESEARCH CENTER

VISCOUS SURFACE & VOLUME MESHING
BARTH & LINTON

OBJECTIVE
- Develop an unstructured mesh generation capability suitable for high Reynold's number viscous computations

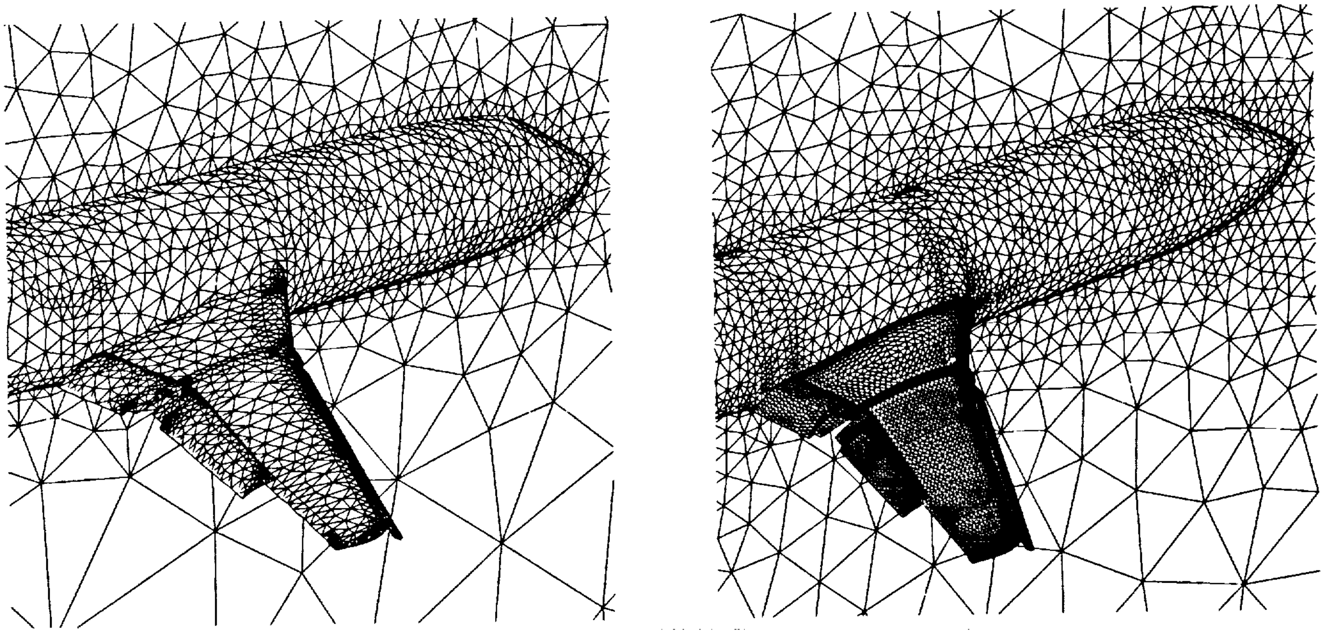
TECHNICAL APPROACH
- Incremental point insertion and local optimization
  - Local optimization allows the generation of high-quality stretched meshes
  - Amenable to solution adaptation

- Surface mesh capability on spline tensor product patches
  - Geometric error minimization
  - Quality repair

- Volume mesh capability includes the construction of conformed and constrained triangulations

STATUS
- Software complete and under evaluation for 3-D high-lift applications

FUTURE DIRECTIONS
- Complete development in cooperation with RFG



Sample surface triangulations contrasting isotropic and stretched capability

AMES RESEARCH CENTER

DYNAMIC MESH ADAPTION
STRAWN & BISWAS

OBJECTIVE

- Develop a fast anisotropic mesh adaptation scheme for large 3-D problems

TECHNICAL APPROACH

- Anisotropic adaptation based on directional error indicators
- Parent element storage allows rapid and scalable grid coarsening
- Edge-based data structure with linked-lists
- Implemented in "C" with dynamic memory allocation

STATUS

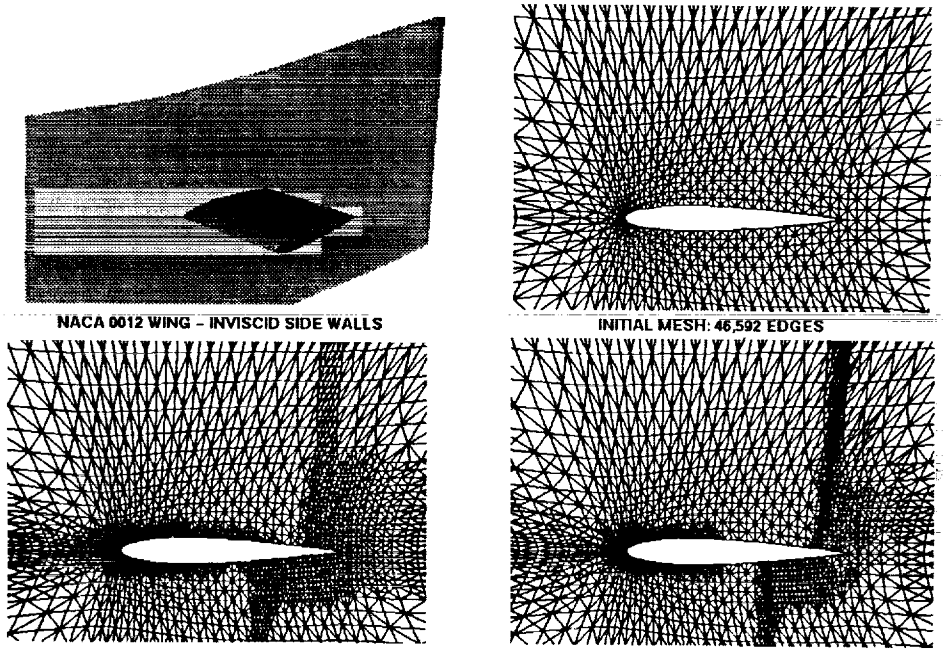
- Refinement/coarsening schemes have been implemented and applied in 2-D and 3-D

FUTURE DIRECTIONS

- Integration of mesh adaptation and flow solver (Barth et al.)
- Arbitrary levels of adaptation with assurance of high mesh quality
- Implement on CM-5

EXAMPLE: 3-D ADAPTIVE GRID REFINEMENT AND COARSENING

FSMACH = 0.85, ALPHA = 1.0 DEG



NACA 0012 WING - INVISCID SIDE WALLS

INITIAL MESH: 46,592 EDGES

FIRST REFINEMENT: 75,656 EDGES

3 REFINEMENT LEVELS, 2 COARSENING LEVELS
85,869 EDGES

Rupak Biswas - RIACS
Roger Strawn - US Army AFDD

AMES RESEARCH CENTER

PARALLEL UNSTRUCTURED GRID GENERATION
RUPPERT

OBJECTIVE

- Develop efficient adaptive parallel-computer unstructured surface and volume grid generation capability

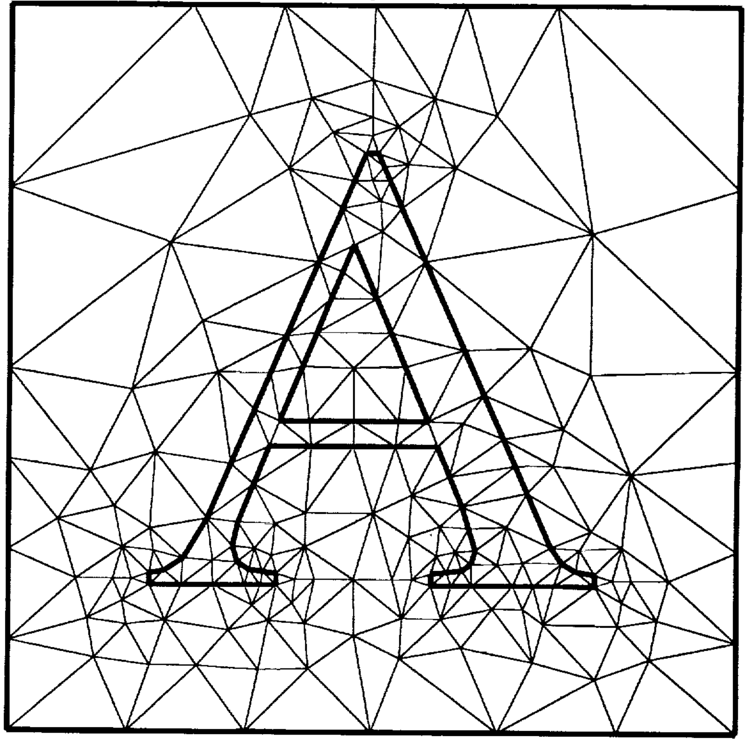
TECHNICAL APPROACH

- Begin with advanced sequential grid generators:
  - Delaunay Refinement algorithm
    Triangles guaranteed to have specified range of aspect ratios
    Number of triangles within a constant factor of optimal
- Research grid quality criteria
- Interface with solver
- Generalize for moving objects
- Parallelize on CM-5

STATUS

- Delaunay Refinement algorithm developed

High-Quality 2D Grid



188 points, 96 segments, min angle=25.2 degrees

AMES RESEARCH CENTER

FAST PARTITIONING & LOAD BALANCING FOR UNSTRUCTURED SOLVERS

BARNARD & SIMON

OBJECTIVE

- Develop partitioning and load balancing technology which allows optimal use of a parallel computer

TECHNICAL APPROACH

- Recursive spectral bisection (RSB) has proven effective, but costly
- A multilevel implementation of RSB which retains favorable features of RSB partitions and reduced cost was developed

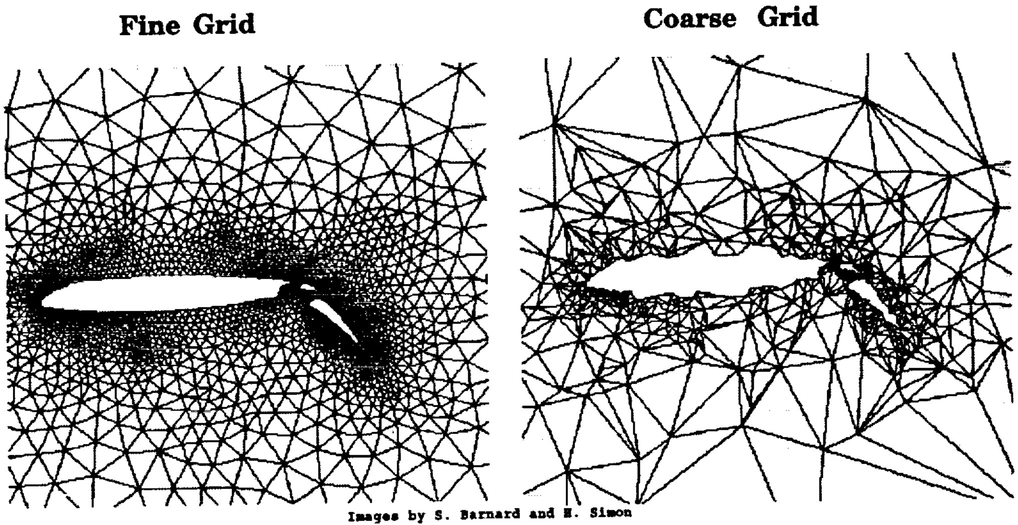
STATUS

- Implemented on workstations, savings up to a factor of 20 verified

FUTURE DIRECTIONS

- Extension to dynamic partitioning of an adapting grid
- Implement in heterogeneous computer network

A Fast Multilevel Implementation of RSB for Partitioning Unstructured Problems



Fine Grid

Coarse Grid

Images by S. Barnard and H. Simon
NASA Ames Research Center

- The coarse grid gives qualitatively the same partitioning.
- Multilevel is an order of magnitude faster than single level for large grids.

AMES RESEARCH CENTER

DYNAMIC LOAD BALANCING FOR UNSTRUCTURED SOLVERS
VENKATAKRISHNAN, VIDWANS, & KALLINDERIS

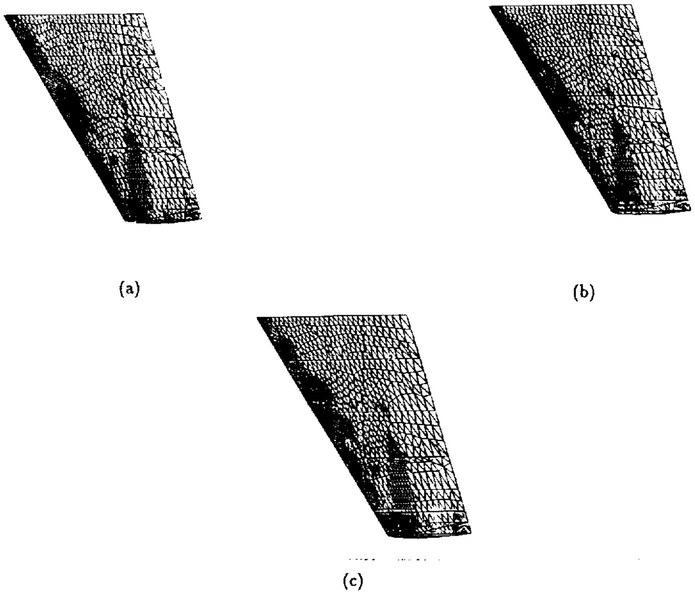
OBJECTIVE
- Develop dynamic load balancing technology which allows optimal use of a parallel computer with dynamic unstructured grid adaptation

TECHNICAL APPROACH
- Divide-and-Conquer strategy used to balance load between each processor
- Local Migration strategy used to actually move points between processors

STATUS
- Implemented on iPSC/860 with application to a variety of grid systems
- Efficient dynamic load-balancing achieved, confirming advantage of using load balancing approaches (e.g., Divide-and-Conquer) with inherent parallel structure

FUTURE DIRECTIONS
- Integrate load balancing technology with complete adaptive unstructured grid generation/flow solver technology, to allow effective use of parallel computer systems in large scale applications

LOAD BALANCING STEPS FOR AN ADAPTED M6 WING


(a)

(b)

(c)

Surface plots for an adapted M6 wing. The thick lines denote partition boundaries. (a) Initial grid. (b) After the first step of load balancing. Processor groups 0,1 and 2,3 are balanced. (c) At the completion of the load balancing. All the processors now have the same load.

AMES RESEARCH CENTER

FELISA

(Finite Element, Langley, Imperial Swansea, Ames)

DJOMEHRI, ERICKSON, WEITING, & IMPERIAL COLLEGE

OBJECTIVE
- Develop a robust solution-adaptive, unstructured Euler grid-generation/solver tool for complex configurations

TECHNICAL APPROACH
- Splined surface definition
- Advancing front grid generation
- Runge-Kutta and Taylor-Galerkin solvers
- Remeshing based on solution gradients

STATUS
- Code evaluation (usability and capabilities)

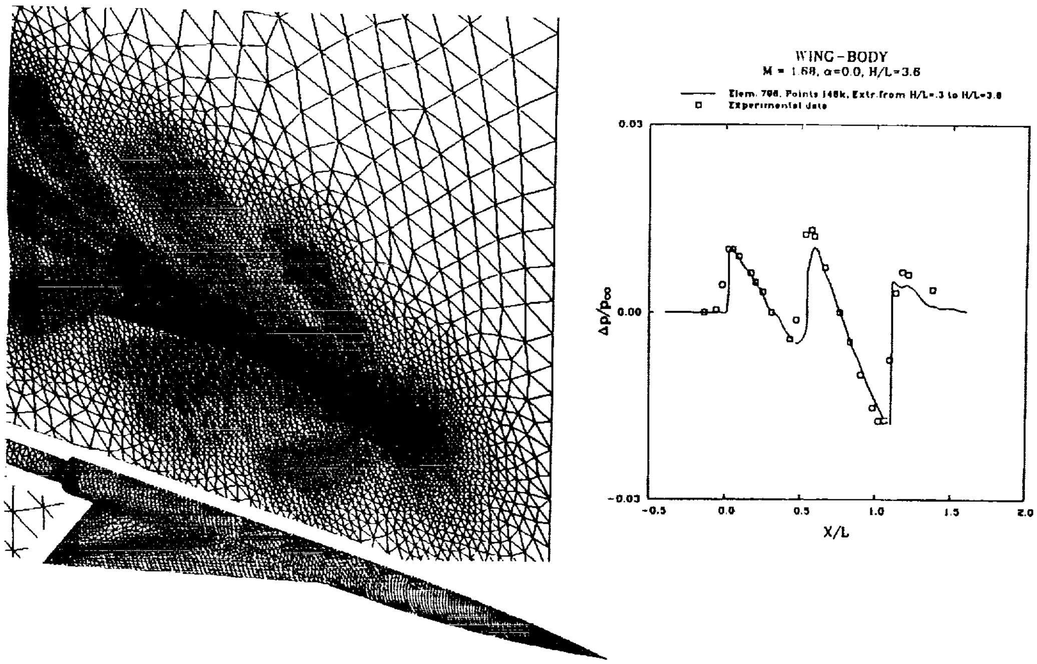
APPLICATIONS
- Generic sonic boom configurations

FUTURE DIRECTIONS
- Learjet applications

- Allow user-specification of surface grid

Wing-Body

Adaptive-Grid Solution



AMES RESEARCH CENTER

AIRPLANE CODE APPLIED TO HSCT CONFIGURATIONS
CLIFF & THOMAS

OBJECTIVE

- Evaluate sonic-boom pressure signatures and aerodynamic performance of High Speed Civil Transport (HSCT) configurations using the AIRPLANE unstructured tetrahedra grid generation/solver package

APPROACH

- Compute near-field off-body pressure signatures and aerodynamic quantities for complete HSCT configurations

- Integrate analysis capability into optimization process

STATUS

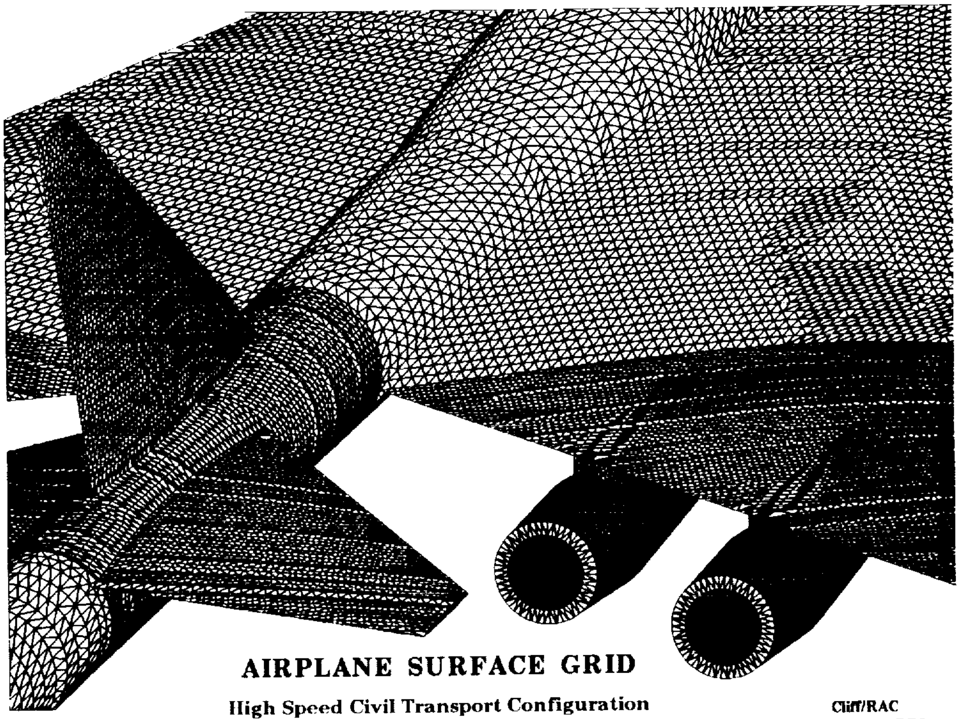
- Accurate prediction of sonic-boom signatures and aerodynamic quantities

- Useful tool for evaluation of complete configurations during the design process

FUTURE DIRECTIONS

- Surface gridding from triangulated surface definition and SUPERPATCH

- Solution adaptation



AIRPLANE SURFACE GRID

High Speed Civil Transport Configuration

Cliff/RAC
Thomas/RFG

AMES RESEARCH CENTER

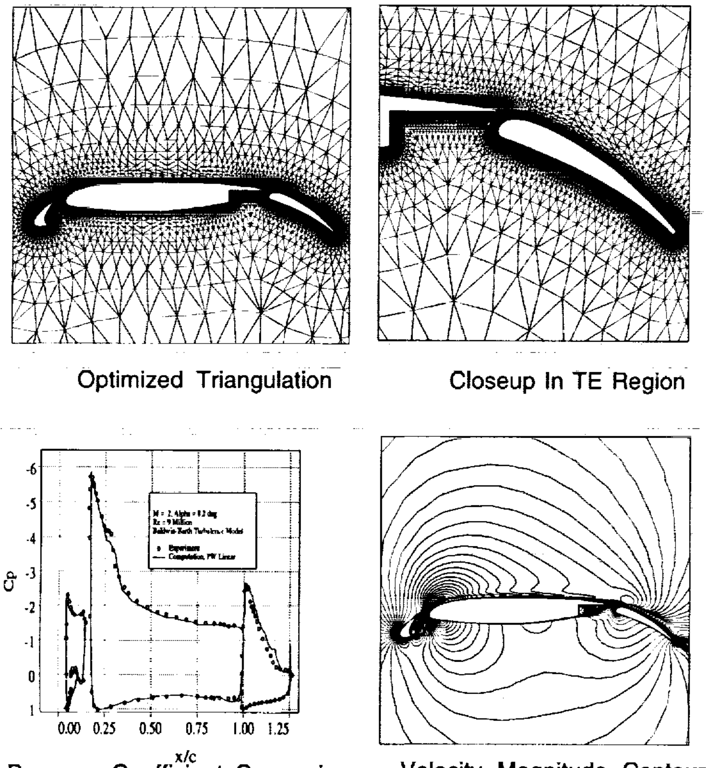
PARALLEL UNSTRUCTURED MESH FLOW SOLVERS
BARTH & LINTON

OBJECTIVE
- **Develop a *fully-implicit* solver for the Euler & Navier-Stokes equations on tetrahedral meshes**

TECHNICAL APPROACH
- **Upwind finite-volume scheme with second-order spatial accuracy**
- **Fully Implicit solver:**
  - **Utilizes a preconditioned minimum residual solver**
  - **Domain decomposed preconditioning using modified incomplete LU decomposition**
  - **Optimized for parallel computer (e.g., CM-5, Intel Paragon)**
- **One-equation turbulence transport model**
- **On-line mesh adaptation**

STATUS
- **Implicit 2-D Navier-Stokes solver capability**
- **Implicit Euler solver is currently in testing on CM-5**

FUTURE DIRECTIONS
- **Investigate alternative preconditioners and higher-order spatial discretizations**
- **Complete implicit 3-D Navier-Stokes solver**



Optimized Triangulation

Closeup In TE Region

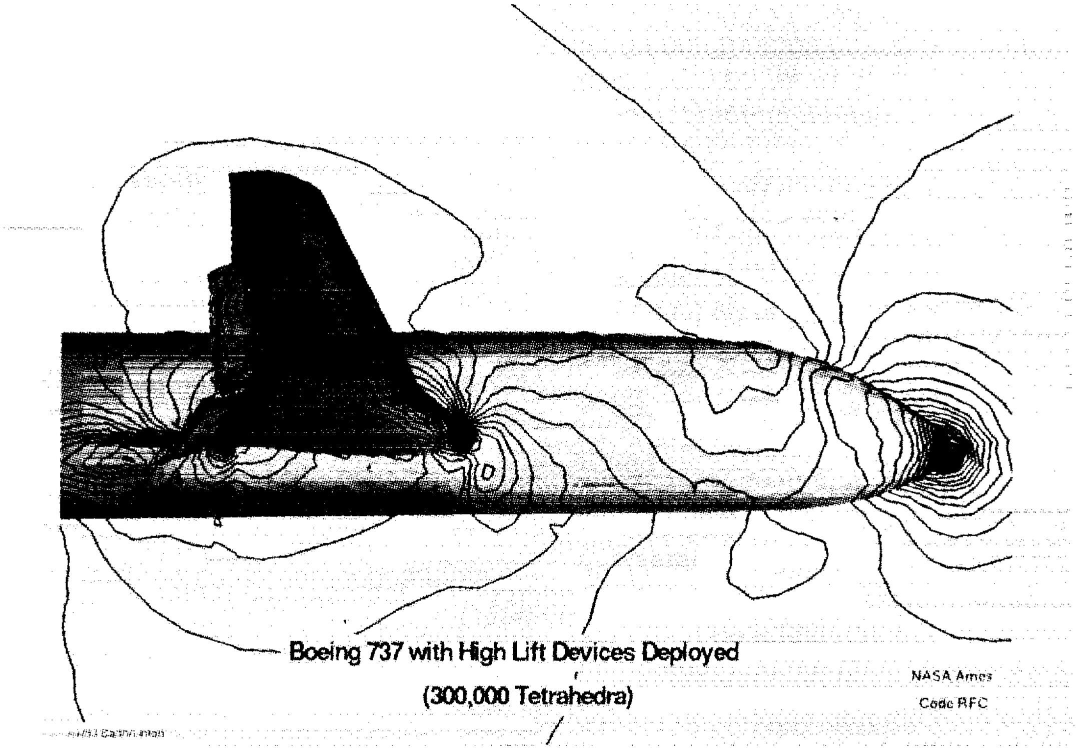
Pressure Coefficient Comparison

Velocity Magnitude Contours

Viscous Flow Past Multi-Element Airfoil

NASA Ames
Code RFC

11/15/92 Barth



Boeing 737 with High Lift Devices Deployed

(300,000 Tetrahedra)

AMES RESEARCH CENTER

FAST DEVELOPMENT
MERRITT, McCABE, SANDSTROM, WEST, BARONIA, SCHMITZ, CASTAGNERA, NEELY & GUMBERT

OBJECTIVE
- A consistent environment for CFD visualization

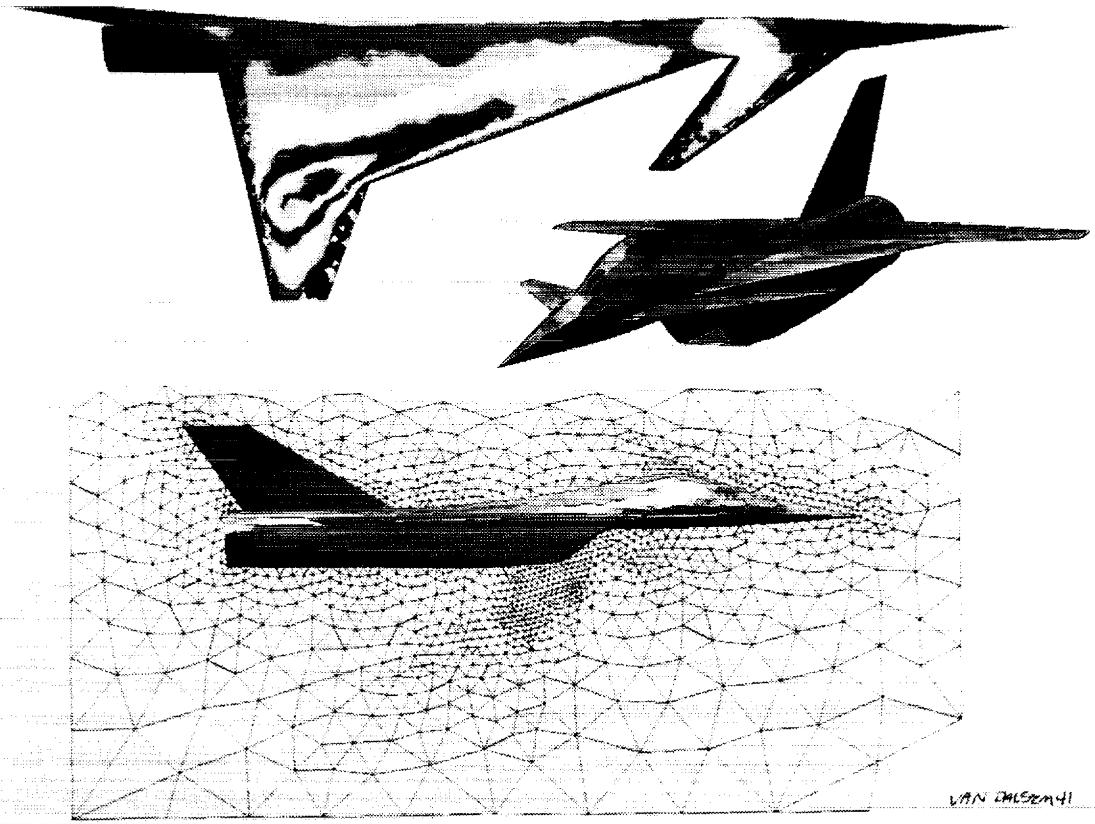
APPROACH
- In cooperation with NASA-Langley (Neely and Gumbert), integrate unstructured visualization modules into the FAST environment

STATUS
- Following modules have been developed, integrated, and tested:

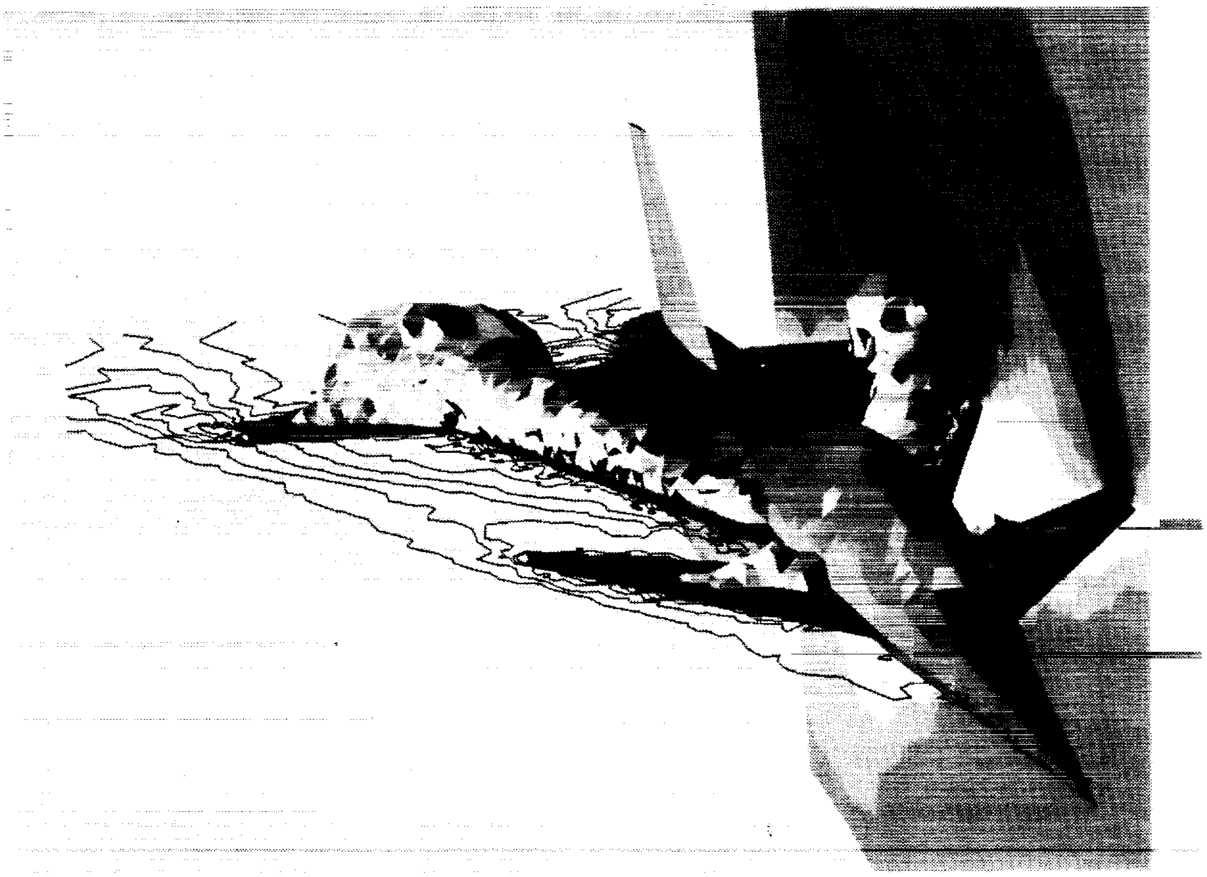
- SURFERU	renders surfaces
- ISOLEVU	displays isosurfaces, cutting planes, etc...
- SHOTET	analyze tetrahedral cells

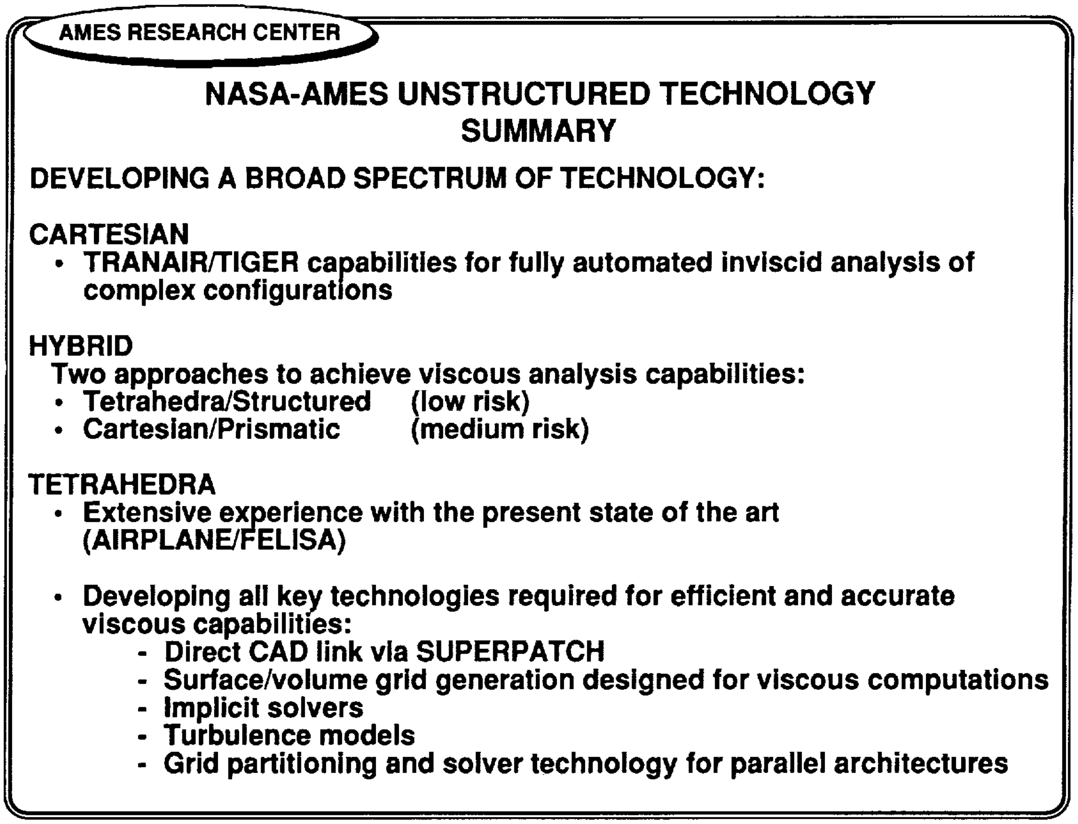
FUTURE DIRECTIONS
- Integrate TRACERU which is used to compute and display particle paths

- Allow visualization of hybrid grid results



VAN DALSEM41 60





AMES RESEARCH CENTER

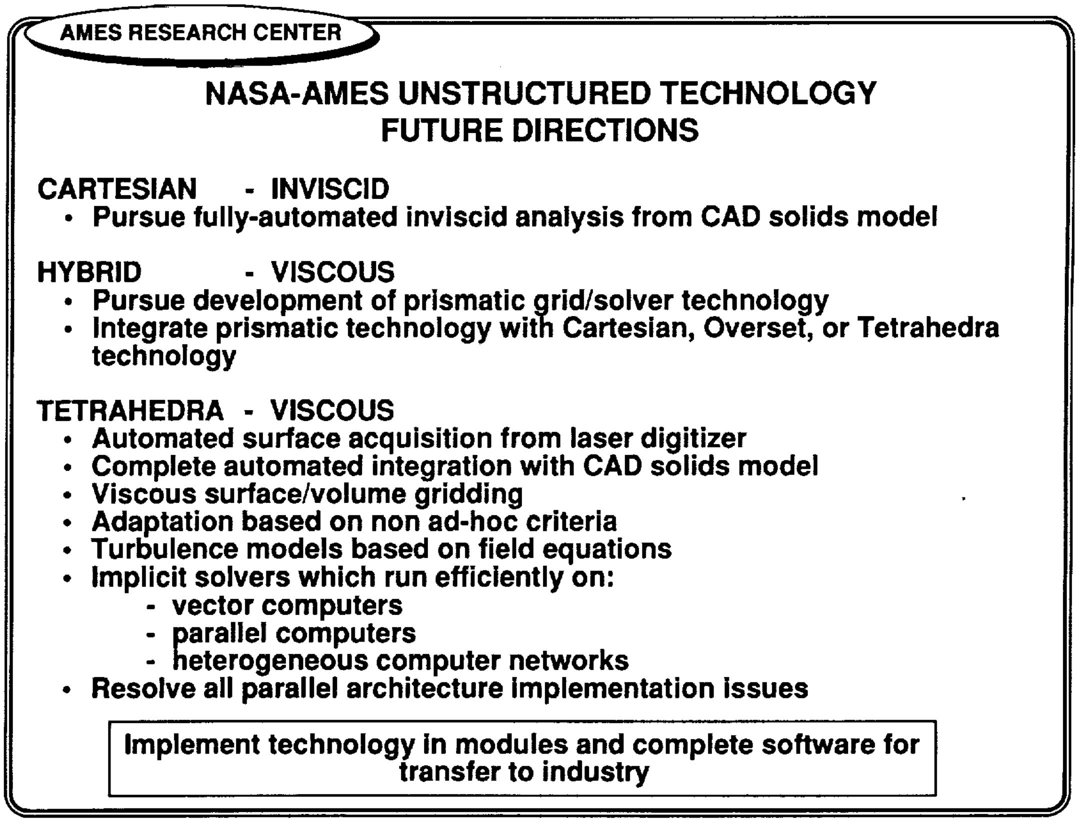
NASA-AMES UNSTRUCTURED TECHNOLOGY SUMMARY

DEVELOPING A BROAD SPECTRUM OF TECHNOLOGY:

CARTESIAN
- TRANAIR/TIGER capabilities for fully automated inviscid analysis of complex configurations

HYBRID
Two approaches to achieve viscous analysis capabilities:
- Tetrahedra/Structured (low risk)
- Cartesian/Prismatic (medium risk)

TETRAHEDRA
- Extensive experience with the present state of the art (AIRPLANE/FELISA)
- Developing all key technologies required for efficient and accurate viscous capabilities:
    - Direct CAD link via SUPERPATCH
    - Surface/volume grid generation designed for viscous computations
    - Implicit solvers
    - Turbulence models
    - Grid partitioning and solver technology for parallel architectures



AMES RESEARCH CENTER

NASA-AMES UNSTRUCTURED TECHNOLOGY FUTURE DIRECTIONS

CARTESIAN - INVISCID
- Pursue fully-automated inviscid analysis from CAD solids model

HYBRID - VISCOUS
- Pursue development of prismatic grid/solver technology
- Integrate prismatic technology with Cartesian, Overset, or Tetrahedra technology

TETRAHEDRA - VISCOUS
- Automated surface acquisition from laser digitizer
- Complete automated integration with CAD solids model
- Viscous surface/volume gridding
- Adaptation based on non ad-hoc criteria
- Turbulence models based on field equations
- Implicit solvers which run efficiently on:
    - vector computers
    - parallel computers
    - heterogeneous computer networks
- Resolve all parallel architecture implementation issues

Implement technology in modules and complete software for transfer to industry

UNSTRUCTURED GRID RESEARCH AND USE AT NASA LEWIS RESEARCH CENTER

MARK G. POTAPCZUK
NASA LEWIS RESEARCH CENTER

CFD Applications at Lewis Research Center

- Inlets, Nozzles, and Ducts
- Turbomachinery
- Propellors - Ducted and Unducted
- Aircraft Icing

Grid Generation Development and Use at Lewis Research Center

- Inlets and Nozzles
  - GRIDGEN
  - TURBO-I/SG
- Turbomachinery and Propellors
  - TIGER
  - TCGRID
  - TIGMIC
  - IGB
  - TIGGERC
  - HGRID
  - TRBGRD
- General
  - GENIE
  - RAMPANT
  - ICEM
- Aircraft Icing
  - HYPGRID
  - GRAPE
  - MINMESH

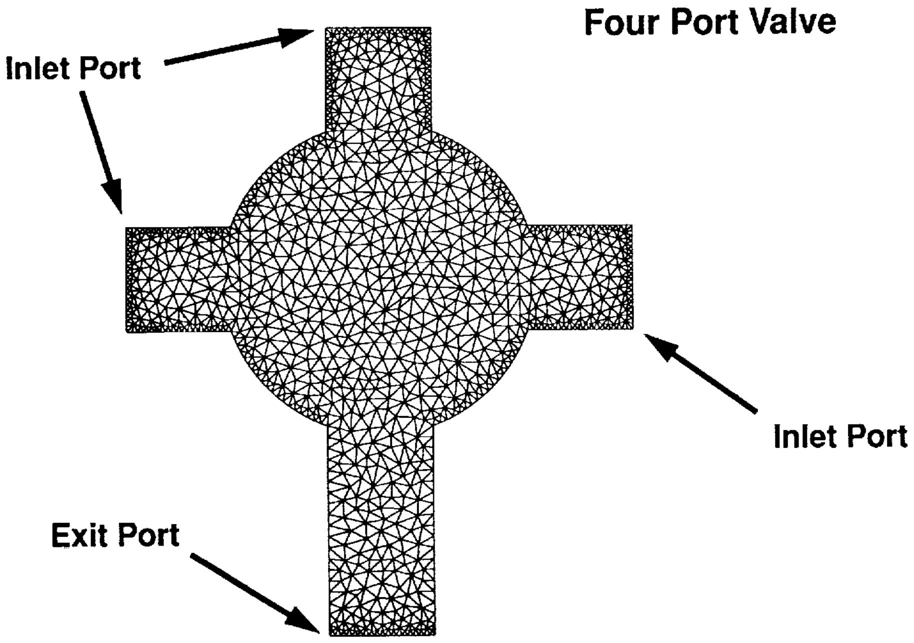
Some Issues related to Internal Flow Grid Generation

- Resolution requirements on several boundaries
- Shock resolution vs. grid periodicity
- Grid spacing at blade/shroud gap
- Grid generation in turbine blade passages
- Grid generation for Inlet/Nozzle geometries

Resolution Requirements on Several Boundaries

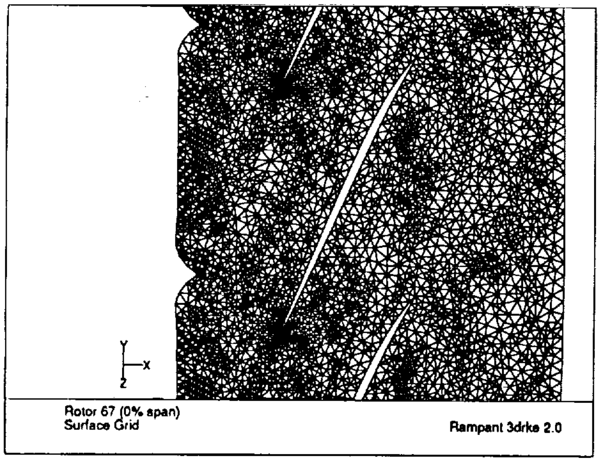
- Internal flow problems may have many intersecting surfaces
- Resolution requirements along surfaces may vary
- Structured grid generators can have great difficulty in meeting both requirements simultaneously

Resolution Requirements on Several Boundaries



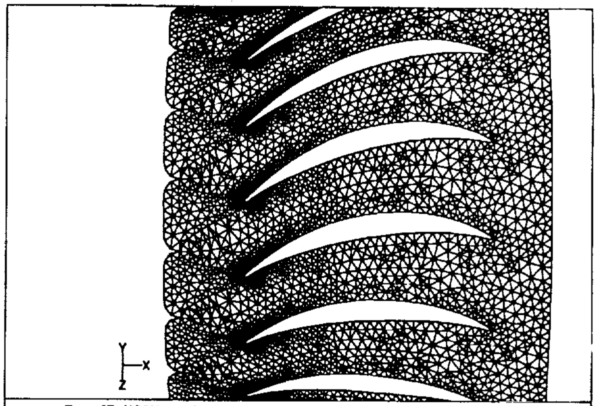
Shock Resolution vs. Grid Periodicity

- Shock locations on upper and lower blade surfaces of cascade occurr at different chordwise locations
- Geometry of shock does not correspond to direction of grid lines
- These two requirements result in highly skewed grids and in an excessive number of grid points



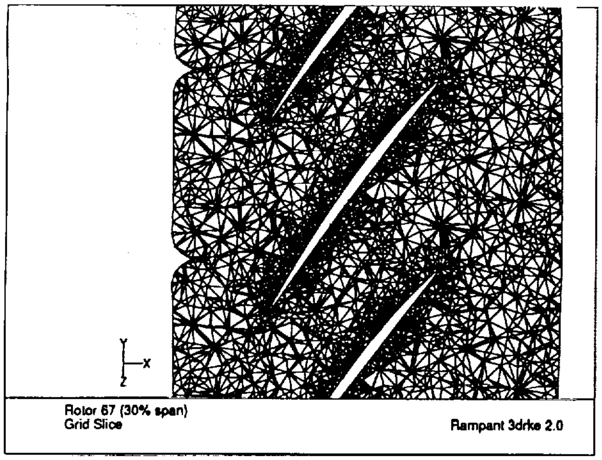
Rotor 67 (0% span)
Surface Grid

Rampant 3drke 2.0



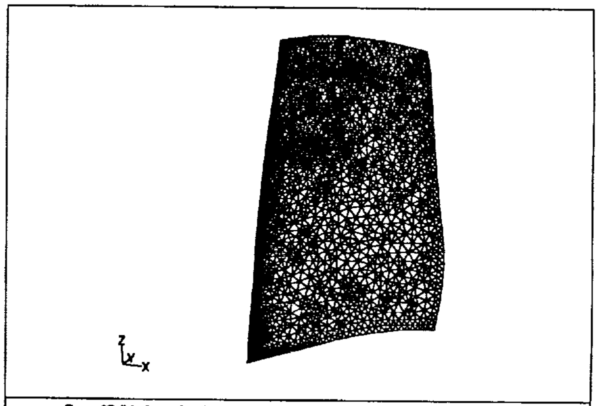
Rotor 67 (100% span)
Surface Grid

Rampant 3drke 2.0



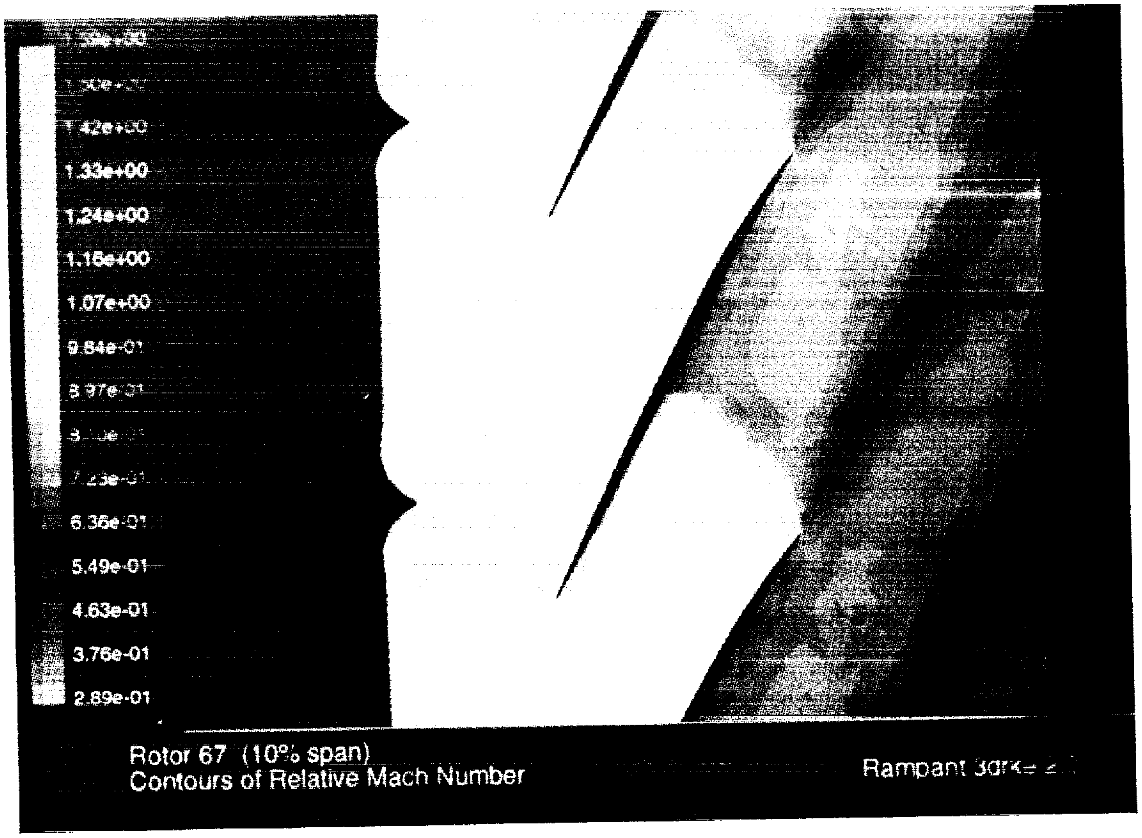
Rotor 67 (30% span)
Grid Slice

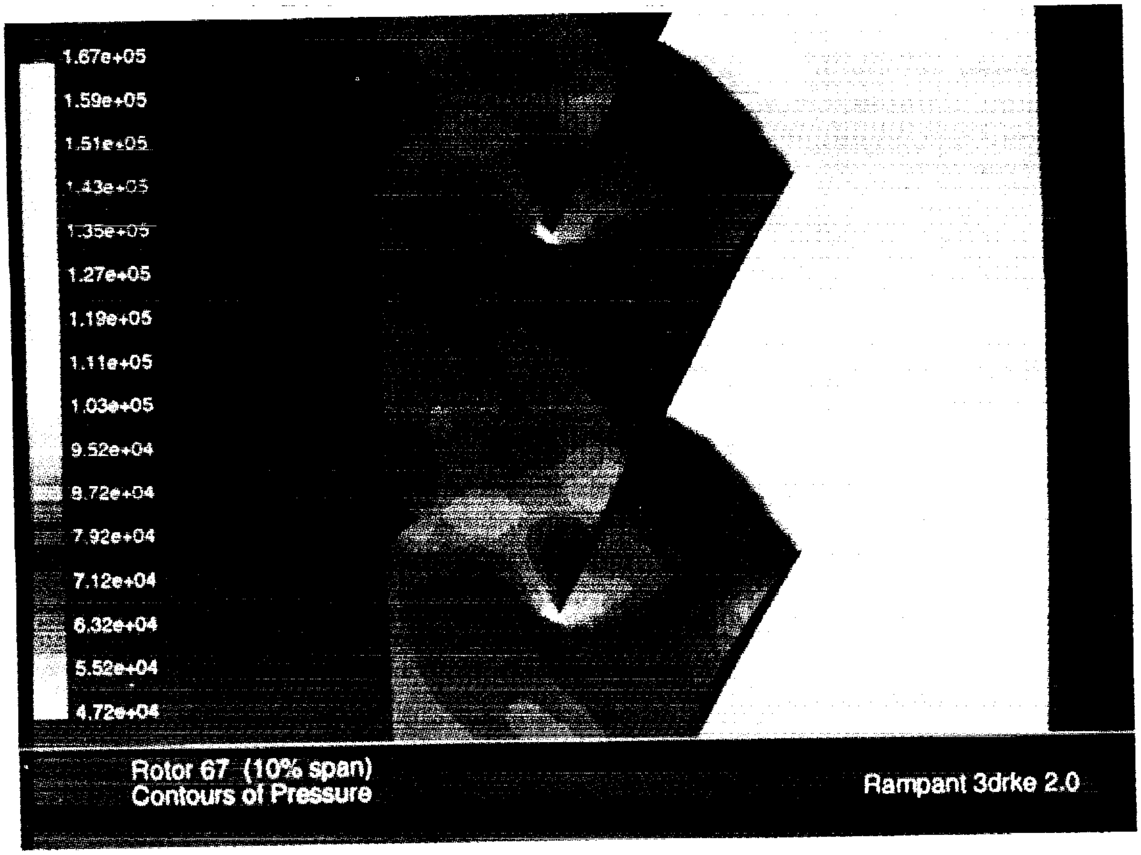
Rampant 3drke 2.0



Rotor 67 (blade surface)
Surface Grid

Rampant 3drke 2.0


1.59e+00
1.50e+00
1.42e+00
1.33e+00
1.24e+00
1.16e+00
1.07e+00
9.84e-01
8.97e-01
8.10e-01
7.23e-01
6.36e-01
5.49e-01
4.63e-01
3.76e-01
2.89e-01
Rotor 67 (10% span)
Contours of Relative Mach Number
Rampant 3drke 2.0

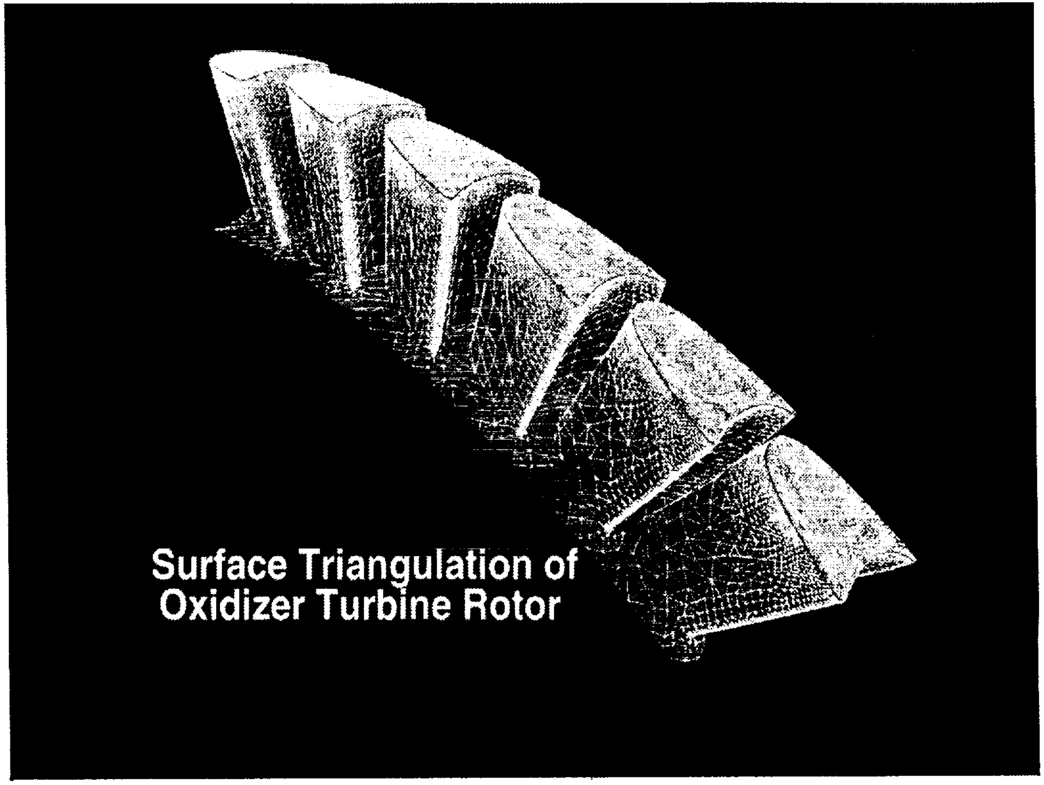

1.67e+05
1.59e+05
1.51e+05
1.43e+05
1.35e+05
1.27e+05
1.19e+05
1.11e+05
1.03e+05
9.52e+04
8.72e+04
7.92e+04
7.12e+04
6.32e+04
5.52e+04
4.72e+04
Rotor 67 (10% span)
Contours of Pressure
Rampant 3drke 2.0

Grid Spacing at Blade/Shroud Gap

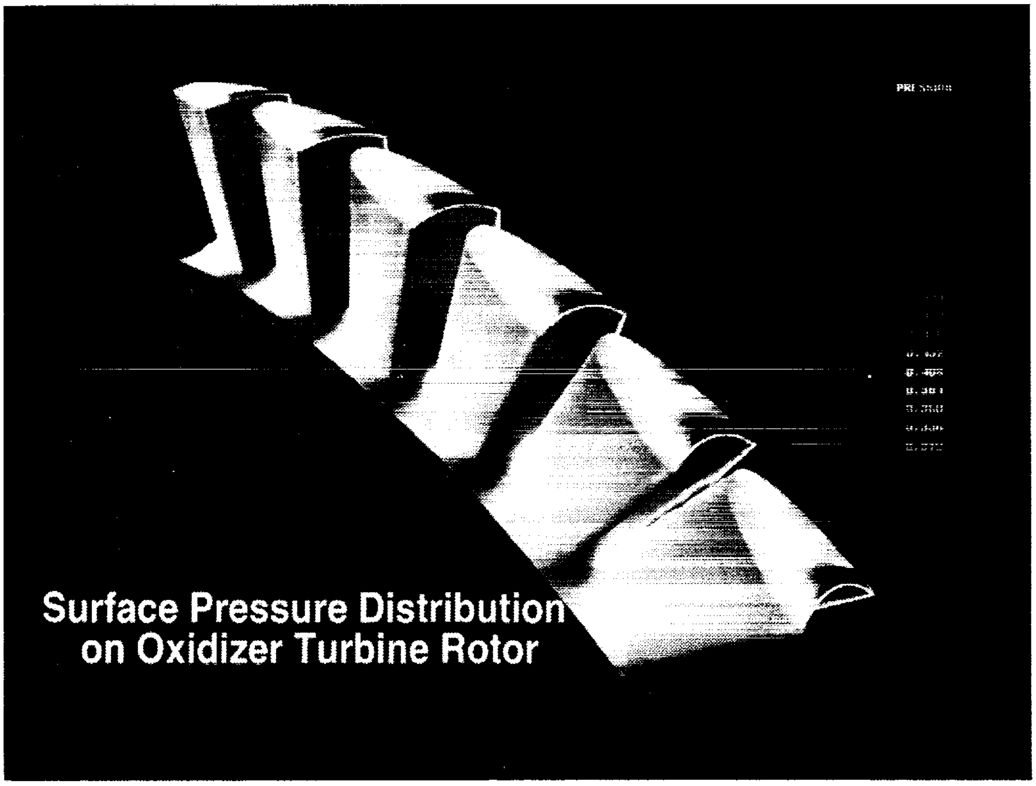
- Small gap (<.2% of blade span) exists between rotor blades and surrounding shroud

- Attempts at modeling gap result in high grid skewing and large number of grid points

- Many structured grid solutions neglect the gap region



Surface Triangulation of
Oxidizer Turbine Rotor



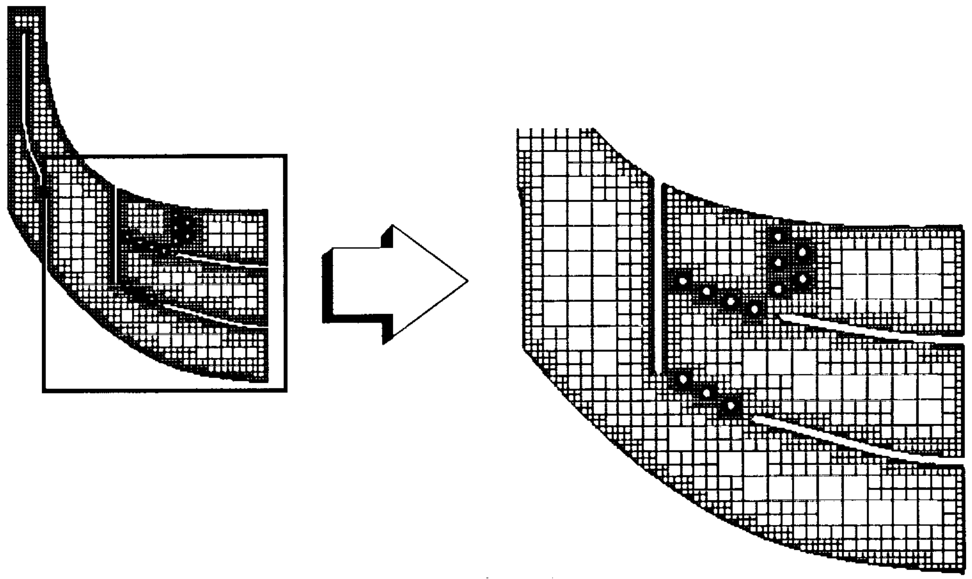
Surface Pressure Distribution
on Oxidizer Turbine Rotor

Grid Generation in Turbine Blade Passages

- Complex geometry and viscous flow modeling results in:
  - Multi-block grid
  - Large number of grid points
  - Labor-intensive grid generation effort

- Automatic generation of internal grid points is required

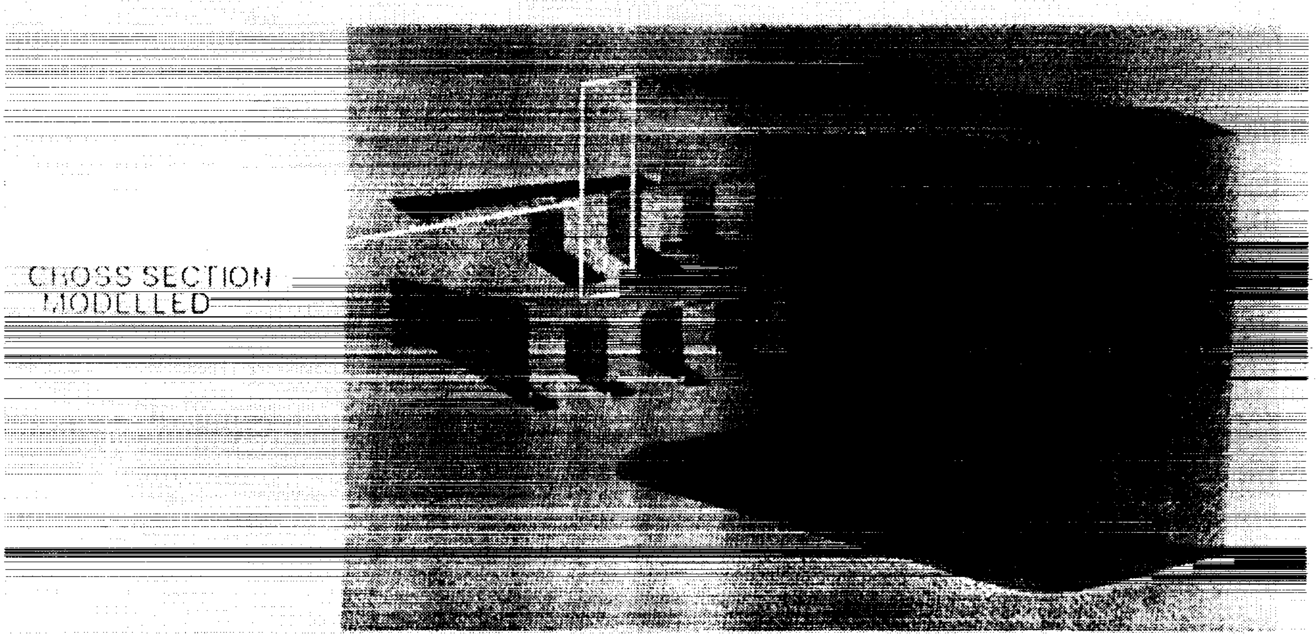
Grid Generation in Turbine Blade Passages



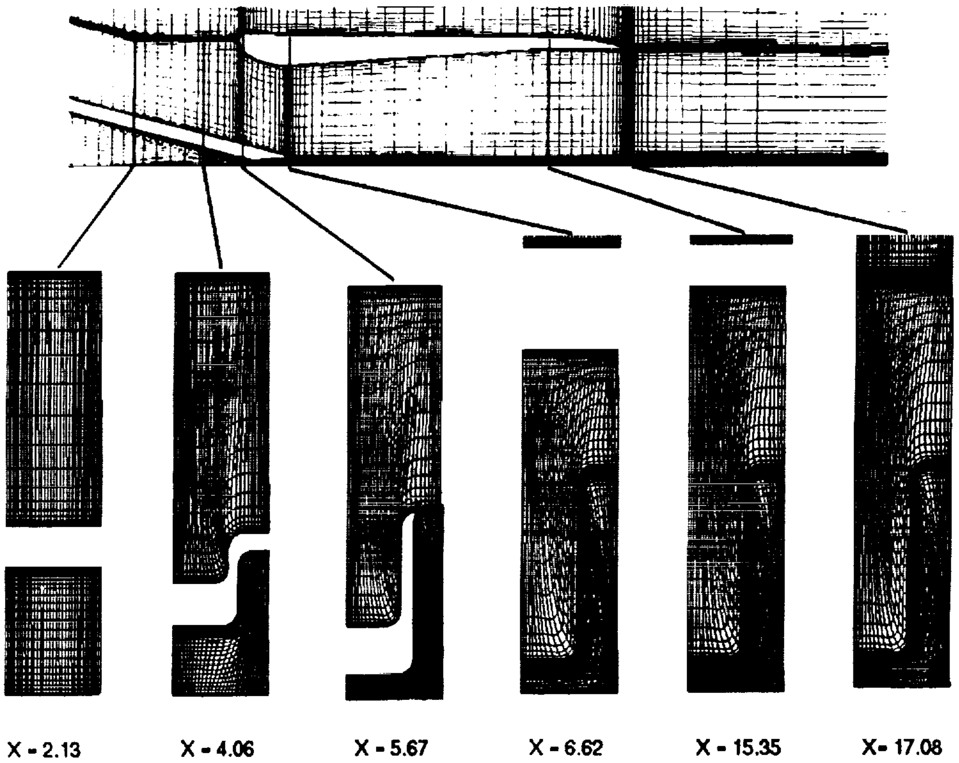
Grid Generation for Inlet/Nozzle Geometries

- Rapidly varying flow passage geometries can result in difficult blocking schemes
- Interfacing of blocks at regions of rapid geometry change can be difficult to achieve
- Geometry and flow phenomena resolution requirements can be conflicting and result in excessively large grids
- Grid development time can be extensive

PRATT & WHITNEY 2D MIXER-EJECTOR NOZZLE GEOMETRY



AXIAL CUTS THROUGH 3-D GRID

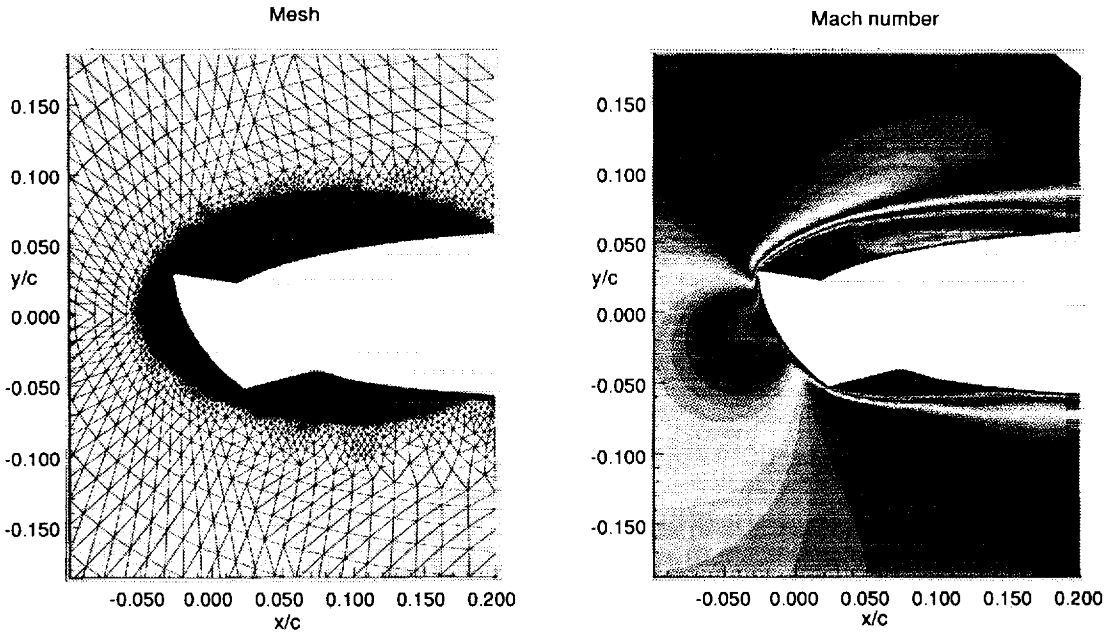


Aircraft Icing Grid Generation Issues

- Small structures relative to airfoil chord must be resolved
- Excessive number of grid points in far-field using structured grid
- Grid must be re-created as ice shape grows

NACA 0012 Airfoil with Simulated Glaze Ice

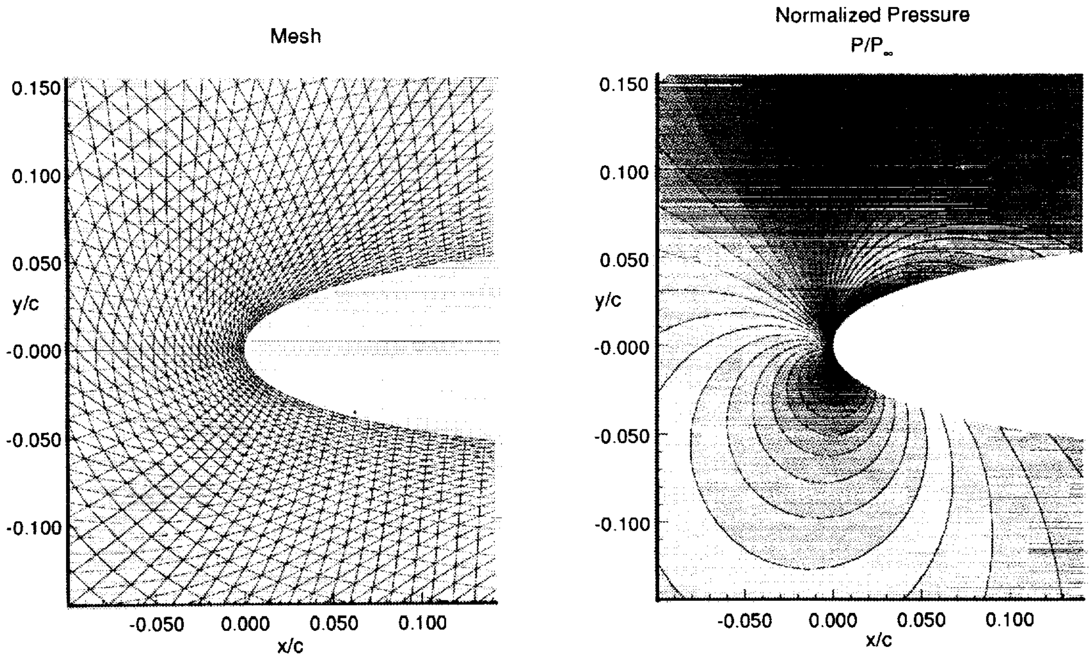
$M_\infty = 0.12, \alpha = 4°$



LEWICE/UE Ice Shape Prediction for Iced NACA 0012 Airfoil

Example 2, Clean Airfoil Calculation

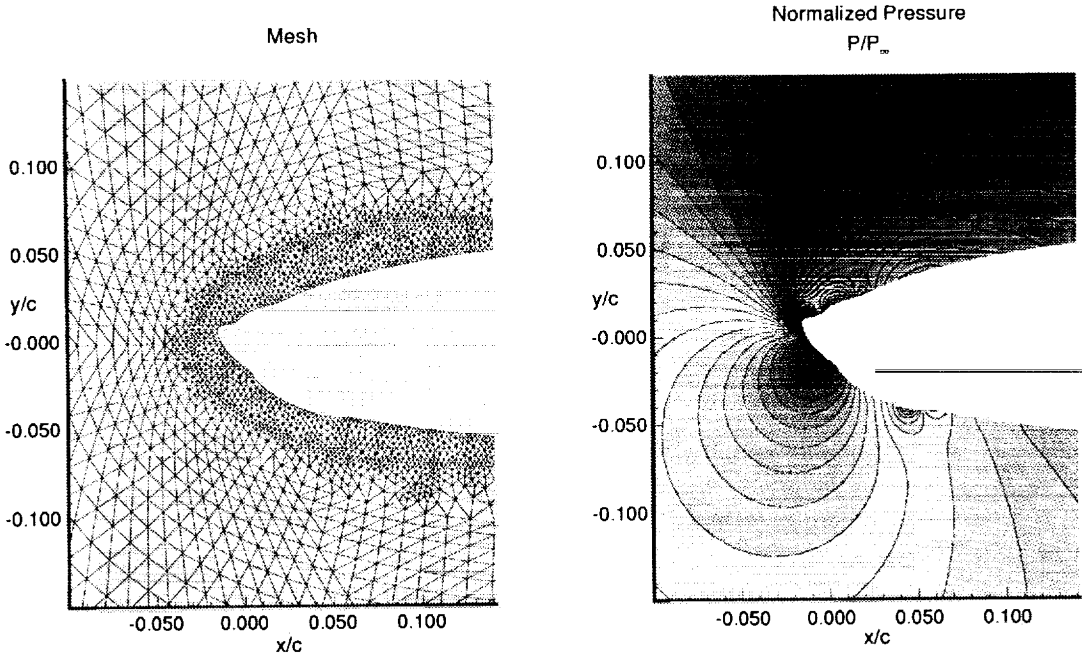
Mach = 0.4, α = 4°



LEWICE/UE Ice Shape Prediction for Iced NACA 0012 Airfoil

Example 2, Time = 60 sec.

Mach = 0.4, α = 4°



Concluding Remarks

- LeRC has several general-purpose and many application-specific grid generators for internal flow CFD analysis

- LeRC has some unstructured grid generation development activities in-house targeted at internal flow problems

- Unstructured grids can simplify and in some cases enable CFD analysis of internal flow geometries

- Unstructured grids are ideally suited for complex, changing geometries such as ice growth on aircraft surfaces

GRID GENERATION REQUIREMENTS AT MARSHALL SPACE FLIGHT CENTER

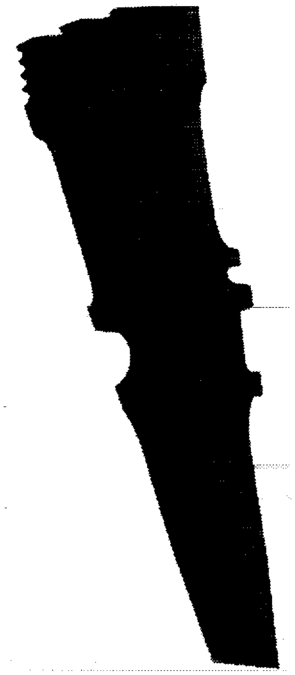
LARRY KIEFLING
MARSHALL SPACE FLIGHT CENTER

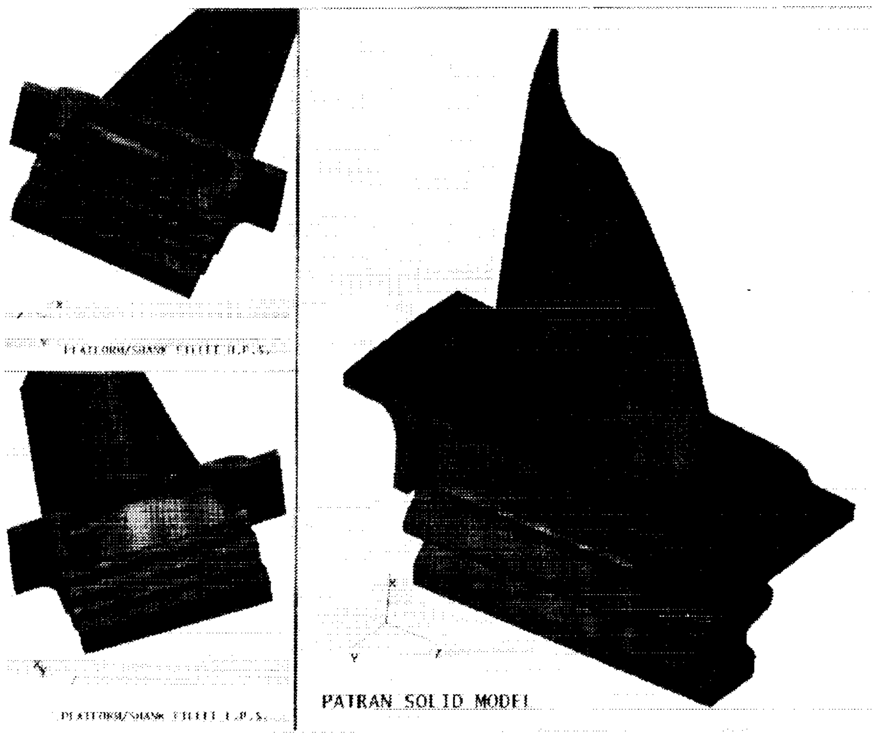
ORGANIZATION: MSFC	**MARSHALL SPACE FLIGHT CENTER**	NAME: LARRY KIEFLING
CHART NO:	SOME EXAMPLES OF STRUCTURAL GRID GENERATION	DATE: 4/27/93

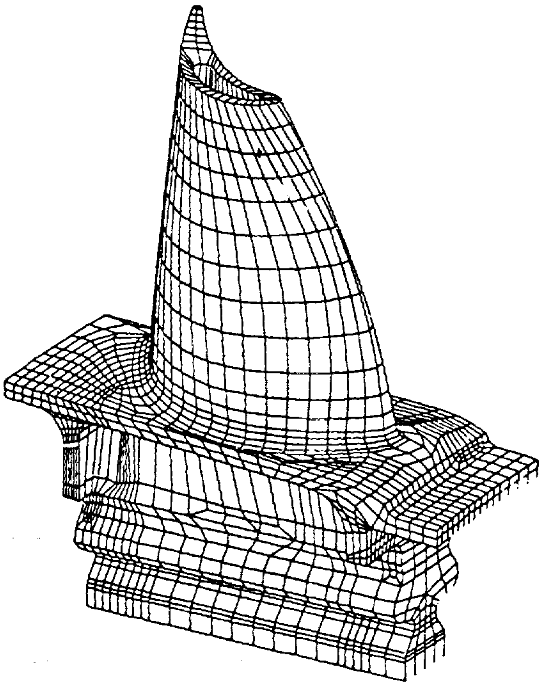
- THREE EXAMPLES FROM MSFC ANALYSTS

- PROPULSION SYSTEMS COMPONENTS
  - HIGH PERFORMANCE REQUIREMENTS
  - MULTI-CURVED SURFACES

- SUMMARY OF NEEDS



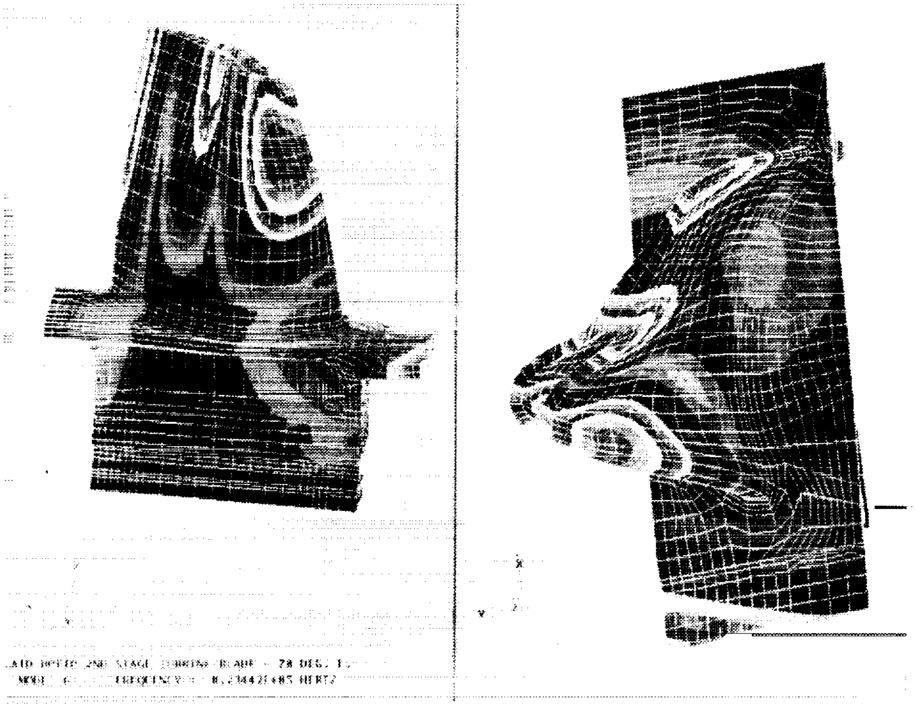




Sverdrup Technology, Inc. Blade Model

Attachment 1

Model Summary	
Quadrilateral Elements	none
Triangular Elements	none
Tetrahedral Elements	none
Pentahedral Elements	14
Hexahedral Elements	7123
Nodes	9478
Duplicate Nodes	24
Element Shape Summary	
Pentahedral Elements	
Aspect Ratio	1.4 - 4.2
Face Skew	32.0 - 61.0
Face Warp	2.3 - 14.0
Face Taper	0.53 - 0.91
Twist Angle	1.1 - 34.0
Edge Angle	21.0 - 75.0
Jacobian Ratio	1.1 - 2.3
Hexahedral Elements	
Aspect Ratio	1.0 - 27.0
Face Skew	0.90 - 83.0
Face Warp	0.0 - 59.0
Face Taper	0.16 - 1.0
Twist Angle	0.0 - 132.0
Edge Angle	1.8 - 90.0
Jacobian Ratio	1.0 - 19.0
General	
Duplicate Elements	None
Element Volume	All positive
Unconnected Nodes	None
Boundary Check	Good
Free Face Check	Good
Free Edge Check	Good
Conne.tivity	Complete
Optimization	PATRAN


2ND STAGE TURBINE BLADE - 78 DEG. F
MODE 6 FREQUENCY = 8.23442E+03 HERTZ
X
Y
Z

23480 Hz

MSFC GRID GENERATION EXAMPLE

Terry Prickett
Rockwell, International

GRID GENERATION OR FINITE ELEMENT MESHING FOR STRUCTURAL AND THERMAL ANALYSIS MODELS IS CURRENTLY ACCOMPLISHED USING INTERACTIVE GRAPHICS BASED SOFTWARE ON PERSONAL WORKSTATIONS. THE TWO SOFTWARE PACKAGES USED MOST OFTEN ARE INTERGRAPH'S I/FEM AND PDA ENGINEERING'S PATRAN. THE TWO PROGRAMS EACH HAVE THEIR STRONG POINTS AND WEAK POINTS, THEREFORE MANY USERS WILL USE BOTH PACKAGES DURING THEIR MODEL CONSTRUCTION.

I/FEM IS AN ADD ON PACKAGE THAT WORKS WITH INTERGRAPH'S I/EMS, WHICH IS A NURBS (NON-UNIFORM RATIONAL B-SPLINE) BASED CAD PACKAGE. MOST ENGINEERING DRAWINGS PREPARED ON-SITE AT MSFC ARE PRODUCED WITH I/EMS. THE MODELER USES BOTH I/EMS CAD COMMANDS AND I/FEM COMMANDS TO BUILD HIS MESH. THIS METHOD WORKS WELL FOR GENERATING MODELS WITH COMPLICATED GEOMETRY.

PATRAN IS A FINITE ELEMENT GENERATION PROGRAM THAT IS BASED ON PARAMETRIC CUBIC GEOMETRY. GENERATING COMPLICATED GEOMETRY IN PATRAN IS MORE TIME CONSUMING THAN I/FEM, BUT MODIFICATIONS TO THE MESH ARE MORE EASILY MADE THAN IN I/FEM. ONE OF THE MOST ATTRACTIVE FEATURES OF PATRAN IS THAT IT ALLOWS YOU TO MODIFY NODE AND ELEMENT ATTRIBUTES BY THEIR ASSOCIATION WITH ANOTHER ENTITY, THEIR INDIVIDUAL ID, PROPERTY ID, MATERIAL ID, OR LOCATION IN SPACE. PATRAN CAN BE CUSTOMIZED USING PCL(PATRAN COMMAND LANGUAGE). PCL IS A HIGH LEVEL BLOCK STRUCTURED PROGRAMMING LANGUAGE DESIGNED TO FIT AROUND THE USER INTERFACE OF PATRAN. IT CAN BE USED TO CREATE SPECIFIC COMMANDS, CREATE TRANSLATORS, PERFORM REPEATED STEPS, etc..

MSFC GRID GENERATION EXAMPLE

- THE HPOTP FIRST STAGE TURBINE DISC FEA MODEL IS A GOOD EXAMPLE OF OF A MODELING EFFORT WHICH USED BOTH SOFTWARE PACKAGES.

- A 2D DRAWING OF THE DISC WAS LOCATED ON THE INTERGRAPH SYSTEM (Fig 1).

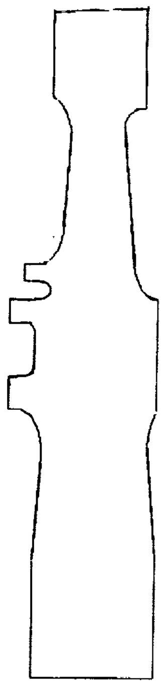
- A 3D SOLID CAD MODEL WEDGE SECTION WAS CREATED FROM THE 2D DRAWING (Fig 2).

- THE SOLID MODEL WAS NEXT BROKEN DOWN INTO LINES THAT COULD BE TRANSLATED TO PATRAN THROUGH IGES (Fig 3).

- THE LINES IN PATRAN WERE USED TO CREATE HYPER-PATCHES, A 3D PARAMETRIC CUBIC SOLID REGION TO WHICH A MESH CAN BE MAPPED (Fig 4).

- GENERATION OF THE FINITE ELEMENT MESH WAS NEXT PERFORMED USING THE HYPERPATCHES. MESH DENSITY WAS CHANGED SEVERAL TIMES, WITHOUT MUCH TIME OR EFFORT, UNTIL AN ACCEPTABLE MESH WAS CREATED (Fig 5).

THE MODEL WAS TRANSFERRED TO ANOTHER ANALYSIS PACKAGE WHERE LOADS AND BOUNDARY CONDITIONS WERE APPLIED AND THE MODEL WAS SOLVED.



ATD HPFTP 2ND STAGE TURBINE BLADE MODAL ANALYSIS
JOHN BERNOT, SVERDRUP CORPORATION

•A THREE DIMENSIONAL SOLID FINITE ELEMENT MODEL WAS GENERATED USING PATRAN. THE PATRAN SOLID MODEL, PRIOR TO GENERATION OF THE FINITE ELEMENT MESH, IS SHOWN IN FIGURE 5.

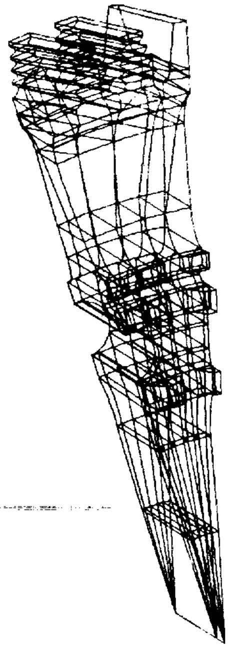
•ALL MAJOR FILLET RADII IN THE BLADE SYSTEM WERE MODELED. WHERE A TOLERANCE WAS SPECIFIED FOR THE BLADE SYSTEM FILLET RADII, MINIMUM VALUES WERE CHOSEN.

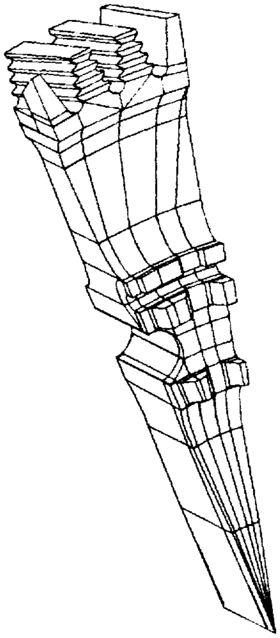
•THE MODEL IS ALMOST ENTIRELY COMPOSED OF HEXAHEDRAL BRICK ELEMENTS IN ORDER TO TAKE ADVANTAGE OF THE BRICK ELEMENT GENERALLY BETTER PERFORMANCE OVER PENTAHEDRAL AND TETRAHEDRAL ELEMENTS.

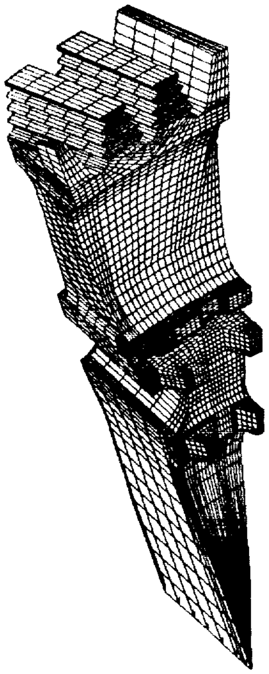
•HIGHER QUALITY ELEMENTS WERE USED IN AREAS OF ANTICIPATED INTEREST, SUCH AS THE BLADE ATTACHMENT RADIUS TO THE PLATFORM, FOR ANY SUBSEQUENT STRESS ANALYSIS WHICH MIGHT LATER BE PERFORMED.

•LOWER QUALITY ELEMENTS, WHOSE GEOMETRIC DISTORTION IS TOO SEVERE TO ACCURATELY PREDICT REALISTIC STRESSES, ETC., WERE RESTRICTED IN THE INTERIOR OF THE BLADE SYSTEM VOLUME AND/OR WHERE RESULTS WERE NOT ANTICIPATED TO BE OF ANY SIGNIFICANT INTEREST.

•THE FINITE ELEMENT MODEL WAS CHECKED BY SVT PERSONNEL AS PART OF ROUTINE QUALITY CONTROL PROCEDURES. MODEL GEOMETRY, CONSTRAINTS, ETC., WERE INDEPENDENTLY EVALUATED AGAINST THE DRAWINGS, ETC. THESE CHECKS ARE SUMMARIZED IN ATTACHMENT 1.



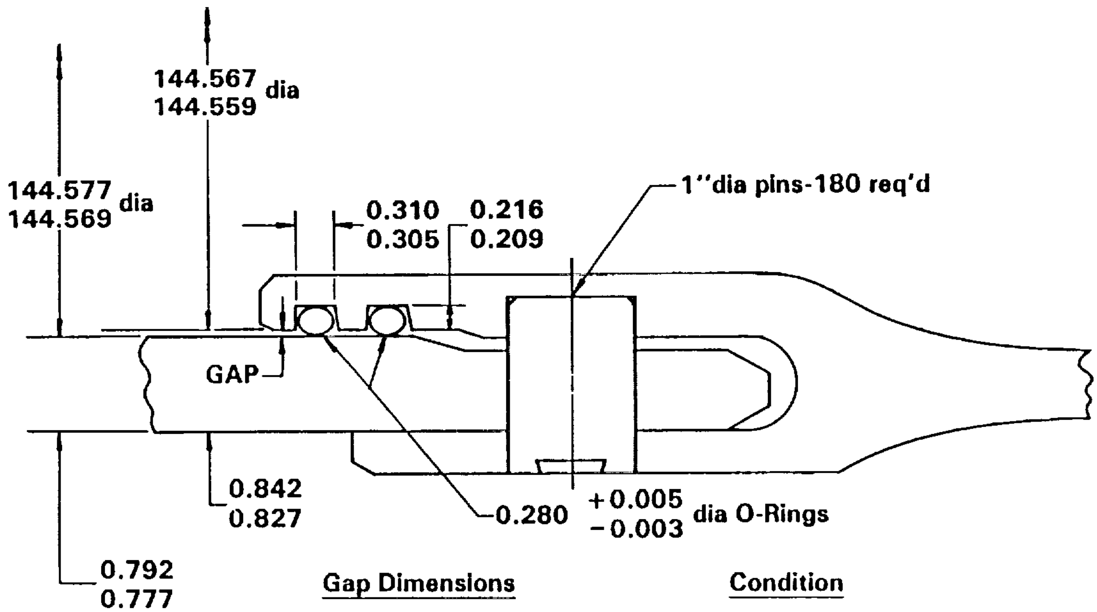




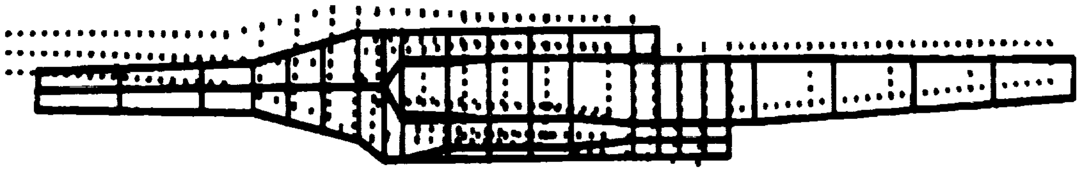
SPACE SHUTTLE SOLID ROCKET MOTOR FIELD JOINT (OLD)

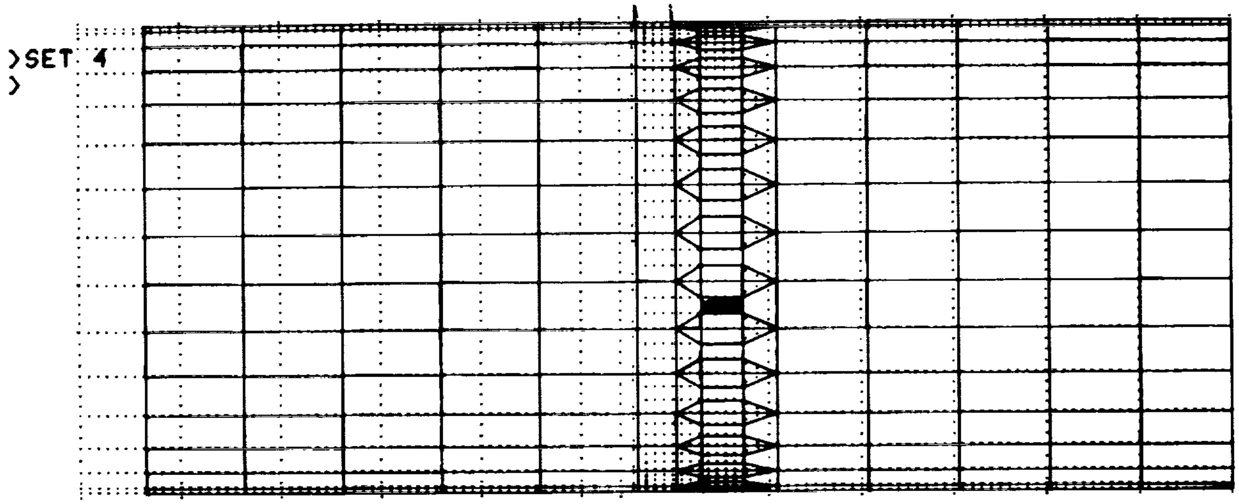
- USED TO USCCESSFULLY MODEL CHALLENGER FAILURE
- SELECTED TO DEMONSTRATE LARGE RANGE OF SCALING

Space Shuttle SRM Segment Joint



Gap Dimensions	Condition	
0.005 ± 0.004	Concentric	Diameter Basis
0.010 ± 008	Non-symmetric	
0.033 Max	Non-symmetric -	Gathering





```
>SET 4
>
```

SOME SPECIFIC NEEDS

•NEED TO HAVE JOINTS AND ELEMENTS IN A RATIONAL ORDER. NEED TO VISUALIZE LOCATION OF MAX STRESS, ETC.

•NEED TO USE QUADS FOR SHELLS AND HEXAHEDRAL SOLIDS AS MUCH AS PRACTICABLE, ESPECIALLY HIGH STRESS AREAS.

•NEED TO KEEP ELEMENT SURFACES FLAT. MOST IMPORTANT FOR SHELLS.

•NEED TO MATE SOLIDS WITH SHELLS AND BEAMS.

•ABILITY TO SELECT NODES AND ELEMENTS MANY WAYS.

•NEED TO DEVELOP MULTISCALE GRIDS.

•NEED TO VISUALIZE BEAM CROSS-SECTION ORIENTATIONS.

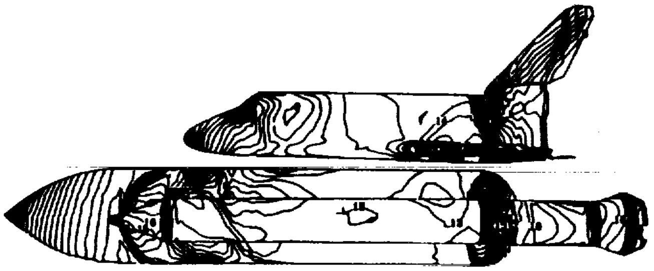
•NEED TO MODEL THEORETICAL POINTS SUCH AS A HINGE CENTERLINE (WITH COINCIDENT NODES).

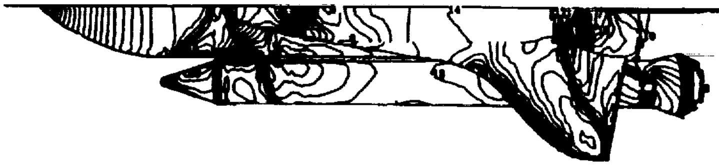
•NEED TO INPUT NONGEOMETRIC DATA.

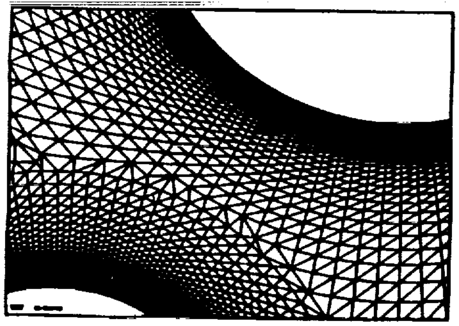
JOHNSON SPACE CENTER CFD GRID GENERATION REQUIREMENTS

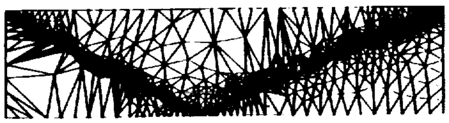
FRED MARTIN
JOHNSON SPACE CENTER

- Thomas Wey/ LESC
  - Grid Generation & Inviscid Solver
    - o THREE-DIMENSIONAL UNSTRUCTURED GRID GENERATION — ANGLE-BASED ADVANCING FRONT METHOD.
    - o THREE-DIMENSIONAL EULER SOLVER — POINT-JACOBIAN, UPWIND, GRID ADAPTATION.
    - o HIGH REYNOLDS NUMBER VISCOUS UNSTRUCTURED GRID GENERATION — CUT AND PASTE, ANGLE-BASED ADVANCING FRONT METHOD.
    - o TRIANGULATION OF OVERLAPPED SURFACE GRIDS — SURFACE PROPERTY INTEGRATION FOR CHIMERA SCHEME.
- Jay Lebeau/EG3
  - Studied Under Tayfun Tezduyar at the University of Minnesota





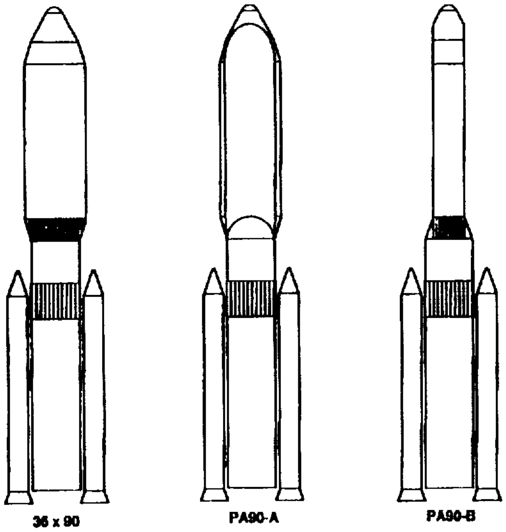




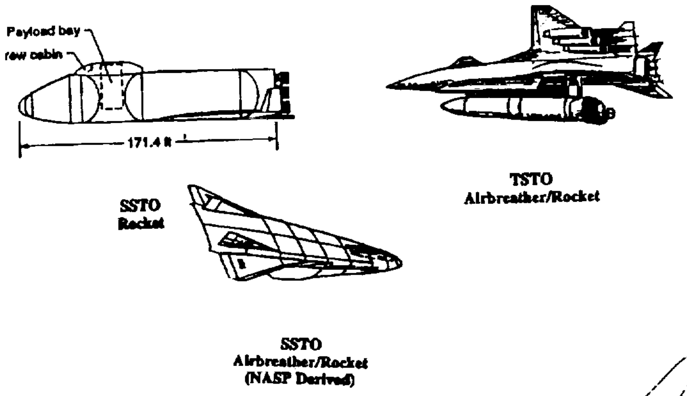
- Requirements Are Driven By :
  - JSC Structures Division's Need for **VERY** Accurate Aerodynamic Loads
  - Program Office Need For CFD Results That Meet **THEIR** Schedule

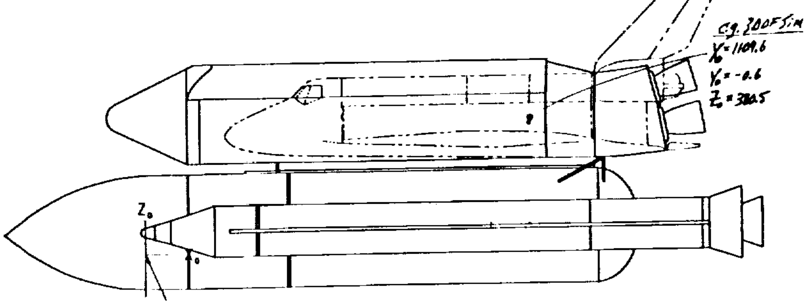
- Launch Vehicles
  - Very Complex Geometry
    - Parallel Configurations
    - Attach Hardware
    - Plumbing, Cable Trays, Structural Stiffeners, etc.
    - Engine Bells

- Entry Vehicles
  - Complex Geometry
    - Control Surfaces - Gaps
    - RCS Scarfed Nozzles

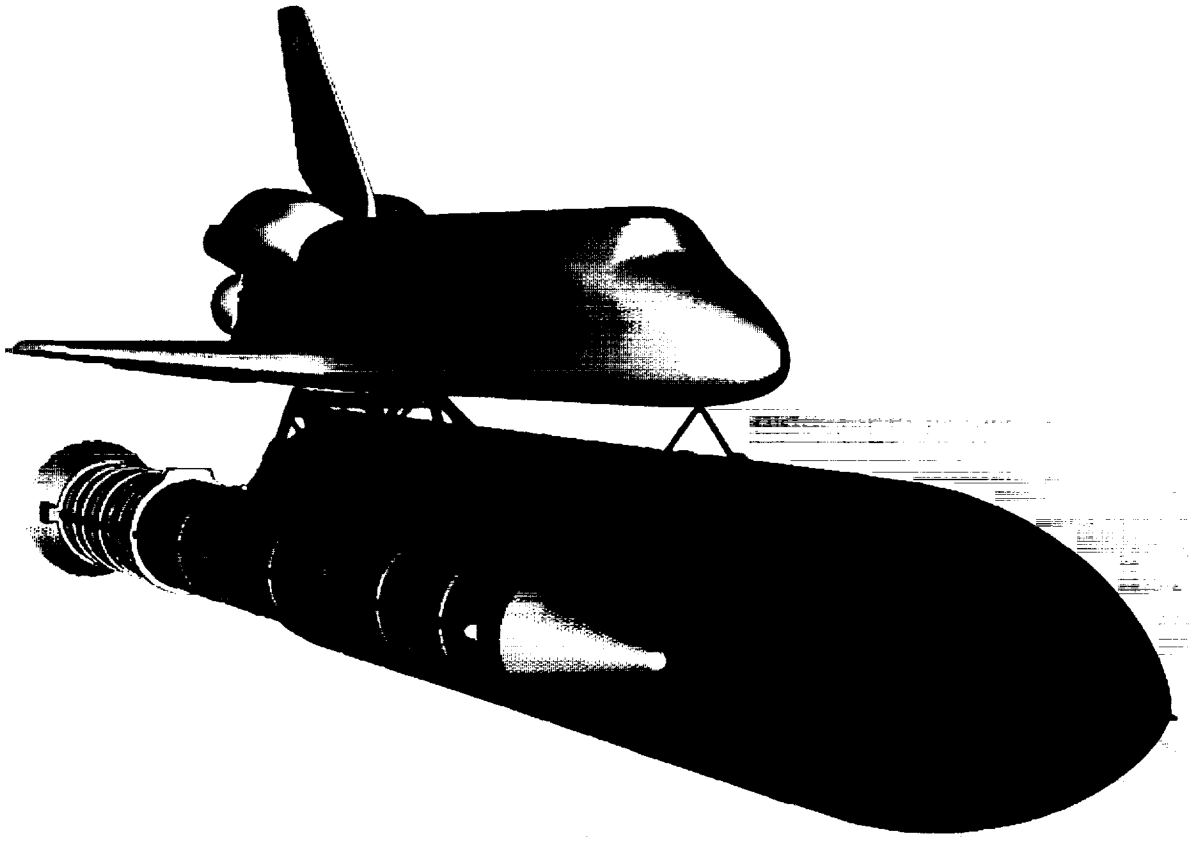


Early HLLV Configurations.





SINGLE LAUNCH STATION



GOAL: Create a High Fidelity Grid/Flow Field That Meets Accuracy Req.

- **5% of Orbiter Wing Limit Load**

1st November, 1990

- Evaluate and Search for Tools (Rockwell, Space Division using ICEM)
- ICEM-CFD Demo Version Installed - Evaluated for 2 Months
- Initiated Purchase of ICEM-CFD
- Coordinated Transfer of External Tank CAD Definition from Martin Marietta

1st May, 1991

- IGES Transfer of Computer Vision, Wire Frame, CAD Models From Martin Marietta (4 months)

1st September, 1991

- Conversion of Wire Frame to Surface Model (4 months)

1st January, 1992

1st January, 1992

- Approximate Geometry, As Required (6 months)
- **CREATE SURFACE GRIDS in ICEM-CFD**

1st July, 1992

- **CREATE SURFACE GRIDS IN HYPGIN (ARC, Buning, Chan)** (1 month)

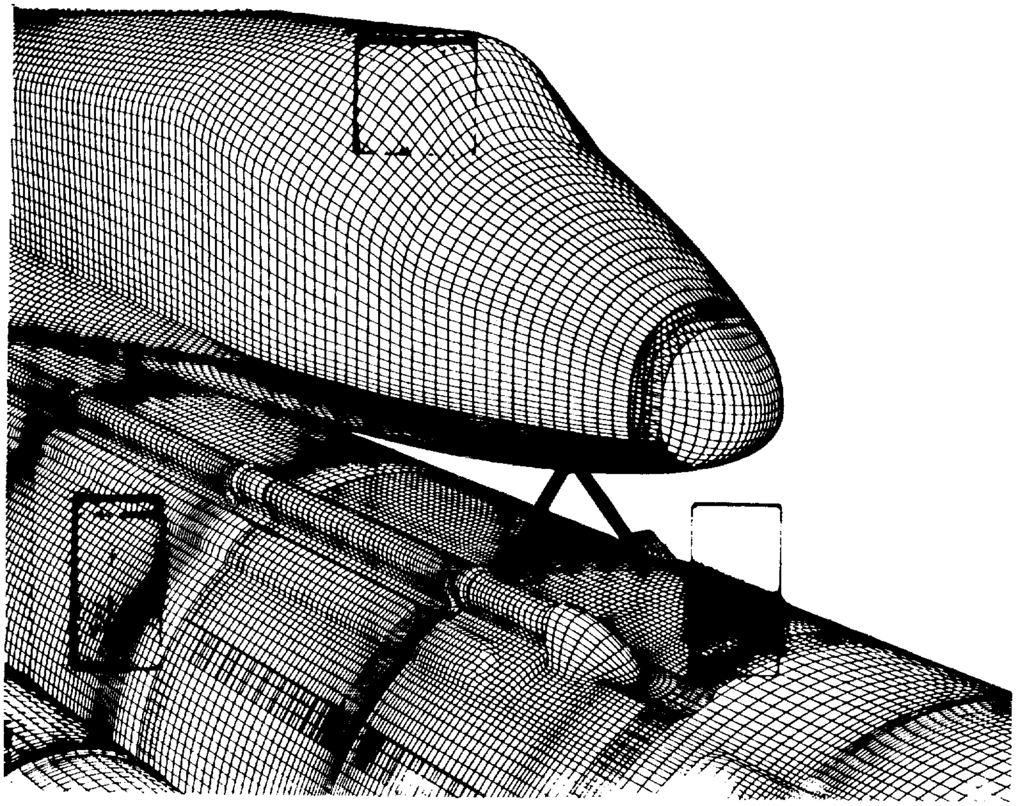
1st August, 1992

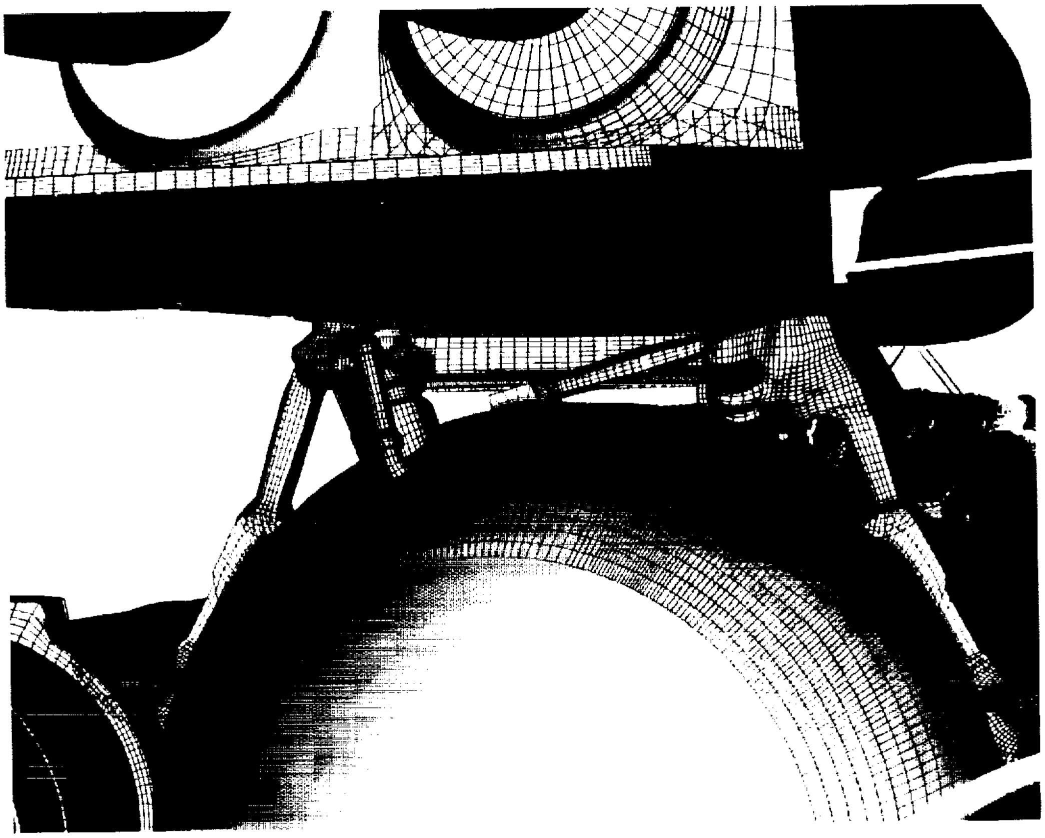
- **CHIMERA GRID to GRID COMMUNICATIONS with PEGSUS (ARC, AEDC)** (6 months)

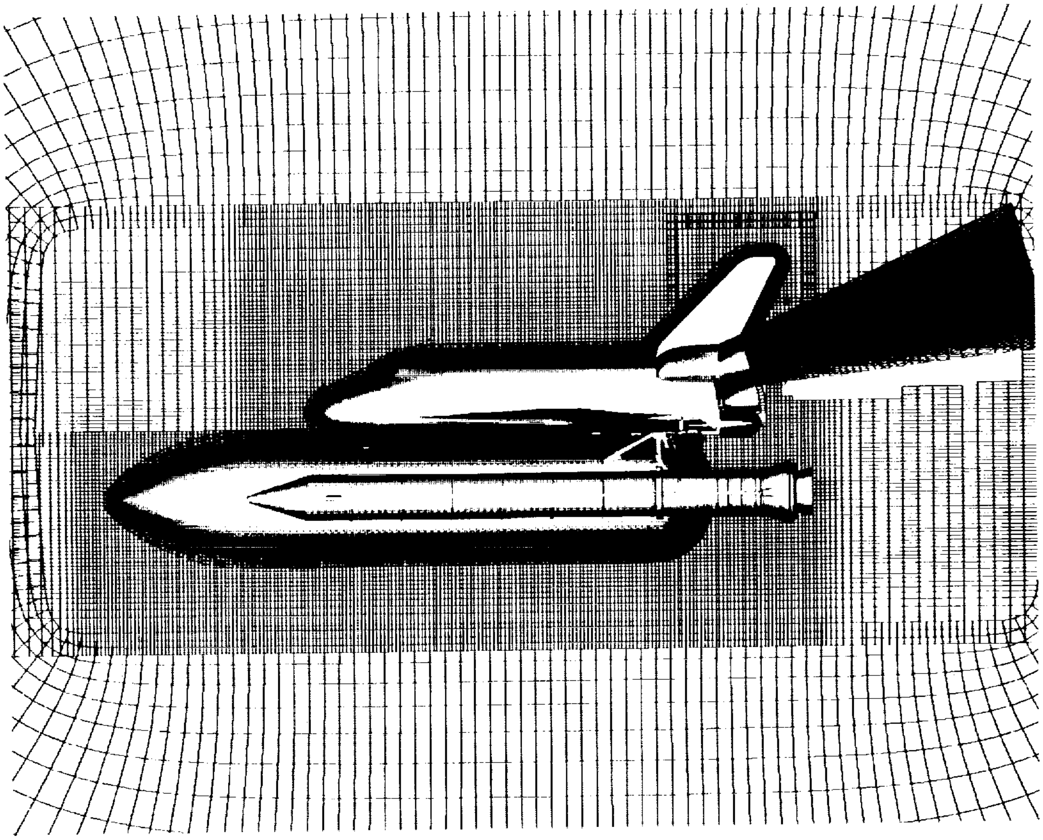
1st January, 1993

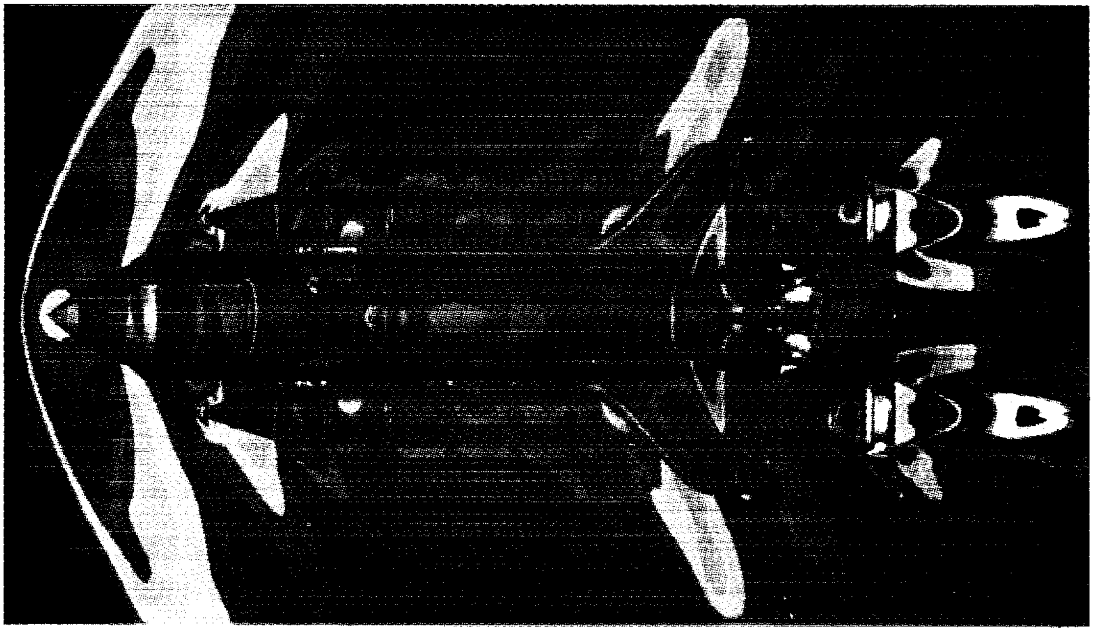
- Started Running The Flow Solver - OVERFLOW (ARC, Buning)
- Minor Corrections to the Grid System
- 16.5 Million Grid Points in 113 Grids, 64 bit Words - Flight Reynolds #

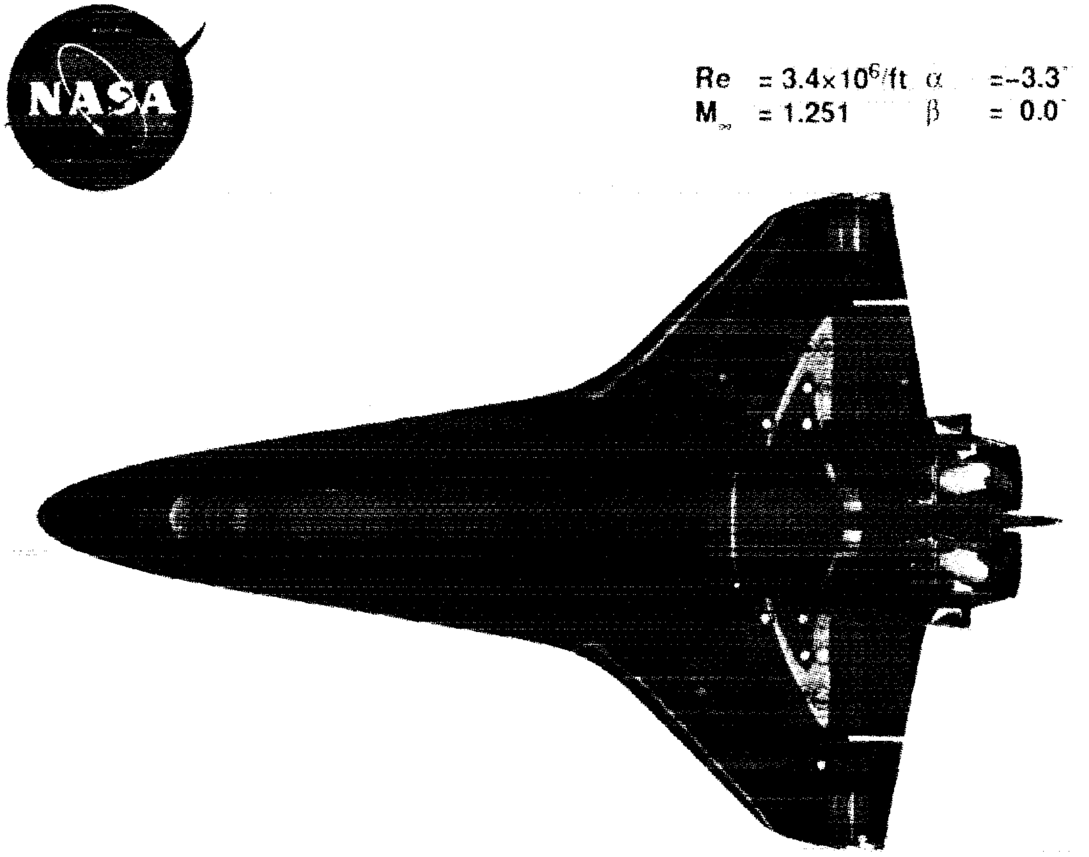
"ALL STEPS LOOP BACK TO ALL PREVIOUS STEPS"

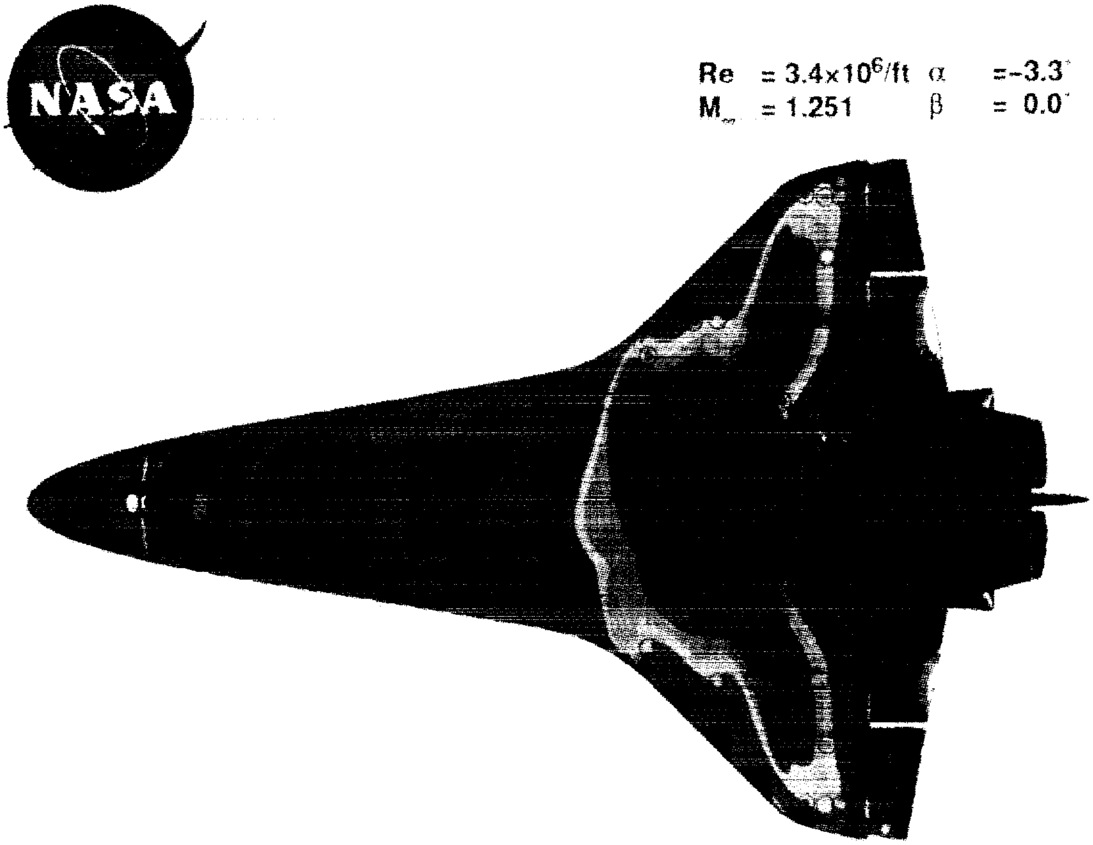









NASA
Re = 3.4×10^6/ft α =−3.3°
M∞ = 1.251 β = 0.0°



$Re = 3.4\times10^{6}/ft$ $\alpha = -3.3^{\circ}$
$M_{\infty} = 1.251$ $\beta = 0.0^{\circ}$

- Replace Orbiter with Space Station Core

6th April, 1993

- Dan Pearce is asked to Grid SLSS

16th April

- CAD model is Avaliable From JSC Structures
    - MCAUTO, Surface Model, IGES transfers
- Rebuild Surfaces!

20th April

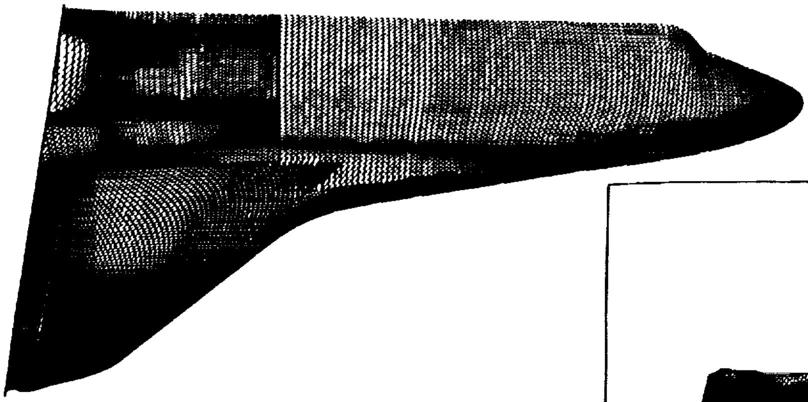
- Surface Gridding in ICEM-CFD

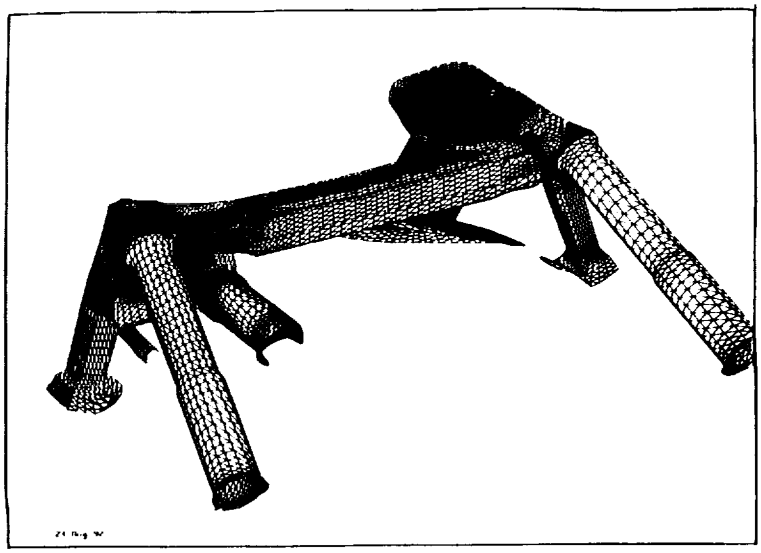
21st April

- Volume Gridding with HYPGIN

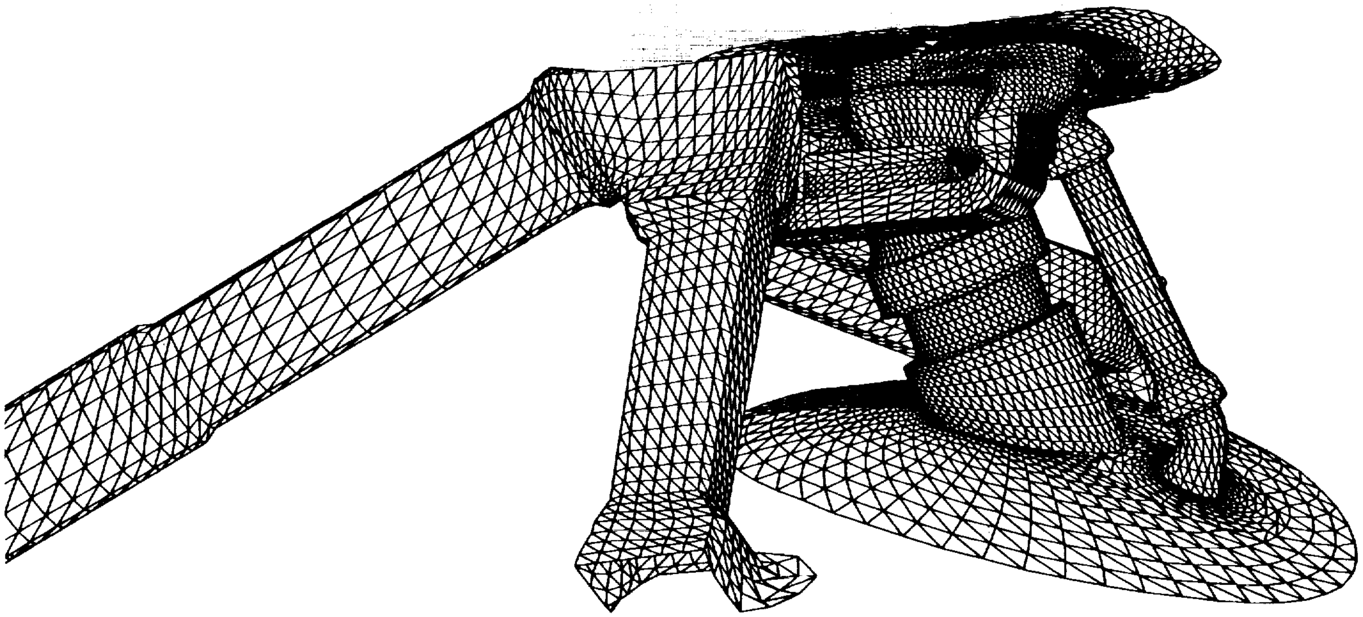
23rd April

- Ready to start developing the Grid to Grid Communications





24 Aug 90



- Complex Geometry - You Get The Picture

- Complex Physics
  - Must Be Viscous Solutions
  - Multiple Species Reacting Flows
    - Ascent Plumes - After Burning, Heating, Ingestion
    - Hypersonic Entry Flows
    - Reaction Control System Flow Field Interactions
  - Unsteady Flows
    - Booster Separation

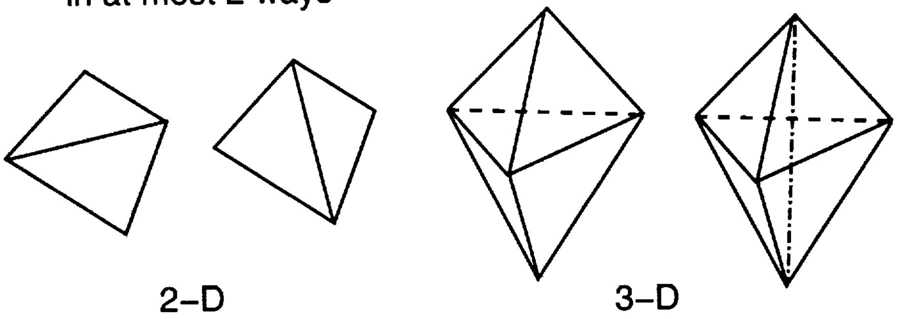
- Computer Issues
  - Out of Core Grid Generation ? (1 large grid will probably not fit in memory)
  - Out of Core Flow Field Solver

3-D UNSTRUCTURED MESH GENERATION USING LOCAL TRANSFORMATIONS

TIMOTHY J. BARTH
NASA AMES RESEARCH CENTER

3-D Combinatorial Edge Swapping

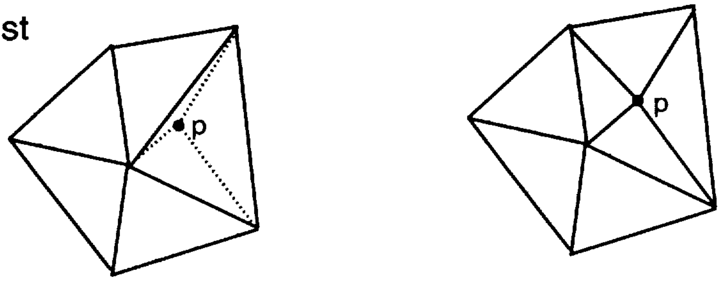
- Convex sets of n+2 sites in $\mathbf{R}^n$ can be configured in at most 2 ways



- This **local transformation** based on a Boolean decision serves as mechanism for local optimization

3-D Incremental Triangulation via Local Transformations

- Joe (1989) and Rajan (1991) showed that 3-D Delaunay triangulations can be constructed using local transformations based on the Boolean circumsphere test



2-D Example of Incremental Insertion and Optimization

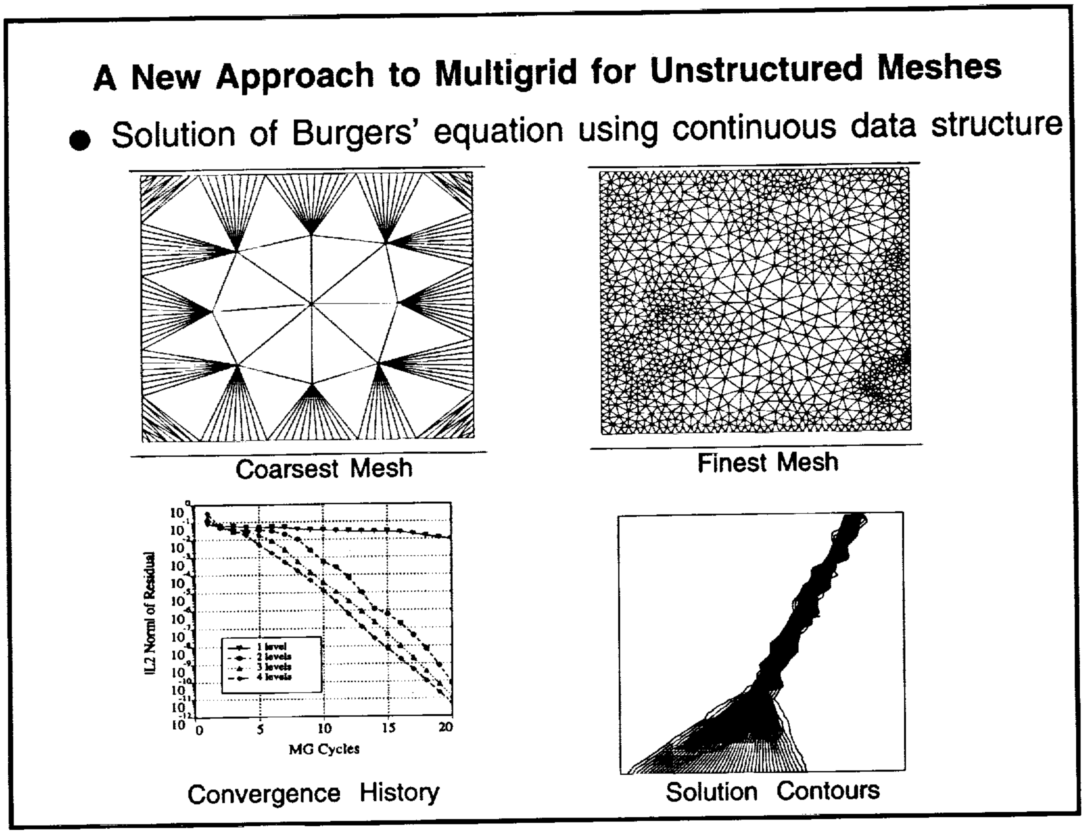
- We have constructed triangulation algorithms in 3-D which locally optimize other mesh qualities: max-min dihedral angles, min-max dihedral angles, etc.

Motivations

- Develop a mesh generation capability suitable for generating highly stretched meshes required for viscous flow computations at high Reynolds numbers
  - Experience has shown that existing triangulation methods such as Delaunay triangulation are not suitable for the generation of highly stretched meshes
- Investigate triangulation algorithms which accommodate mesh generation and adaptation while maintaining high robustness

Randomized △ Algorithms Based on Local Transformations

- Worst case optimal complexity can be achieved by *randomizing* the order in which sites are introduced into the triangulation (Guibas, Knuth, Sharir, 1992)
  - n log (n) expected performance in 2–D
  - n^2 expected worst case performance in 3–D
- Suggests a new "continuous" data structure which encodes a family of triangulations (coarsest to finest)
  - 2–D randomized theory predicts O(n) size of this structure
  - We have exploited this construction to produce a novel multigrid scheme and theory for solving differential eqns



A New Approach to Multigrid for Unstructured Meshes

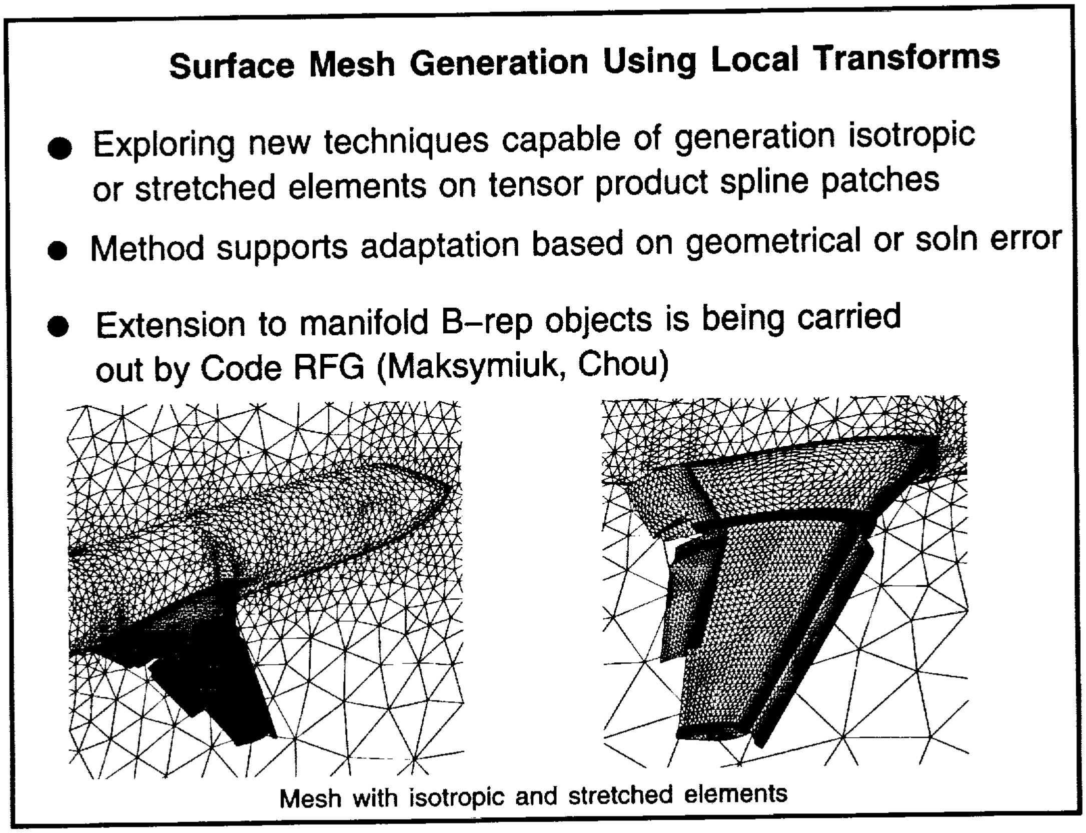
- Solution of Burgers' equation using continuous data structure

Coarsest Mesh

Finest Mesh

Convergence History

Solution Contours



Surface Mesh Generation Using Local Transforms

- Exploring new techniques capable of generation isotropic or stretched elements on tensor product spline patches
- Method supports adaptation based on geometrical or soln error
- Extension to manifold B-rep objects is being carried out by Code RFG (Maksymiuk, Chou)

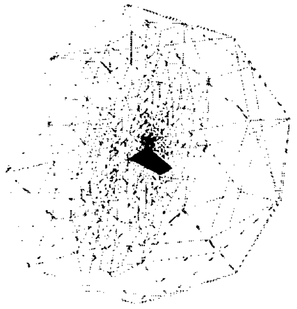
Mesh with isotropic and stretched elements

Volume Triangulations

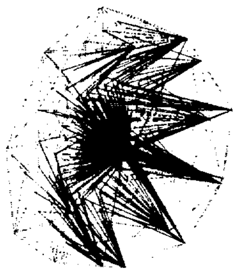
(1) Initial Triangulation of Surface Data

(2) Constrained/Conforming Triangulation to Preserve Body Integrity

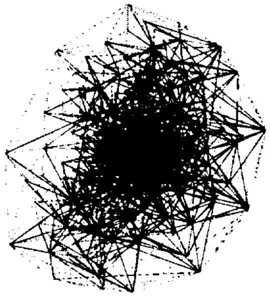
(3) Incremental Insertion and Optimization of Specified Sites



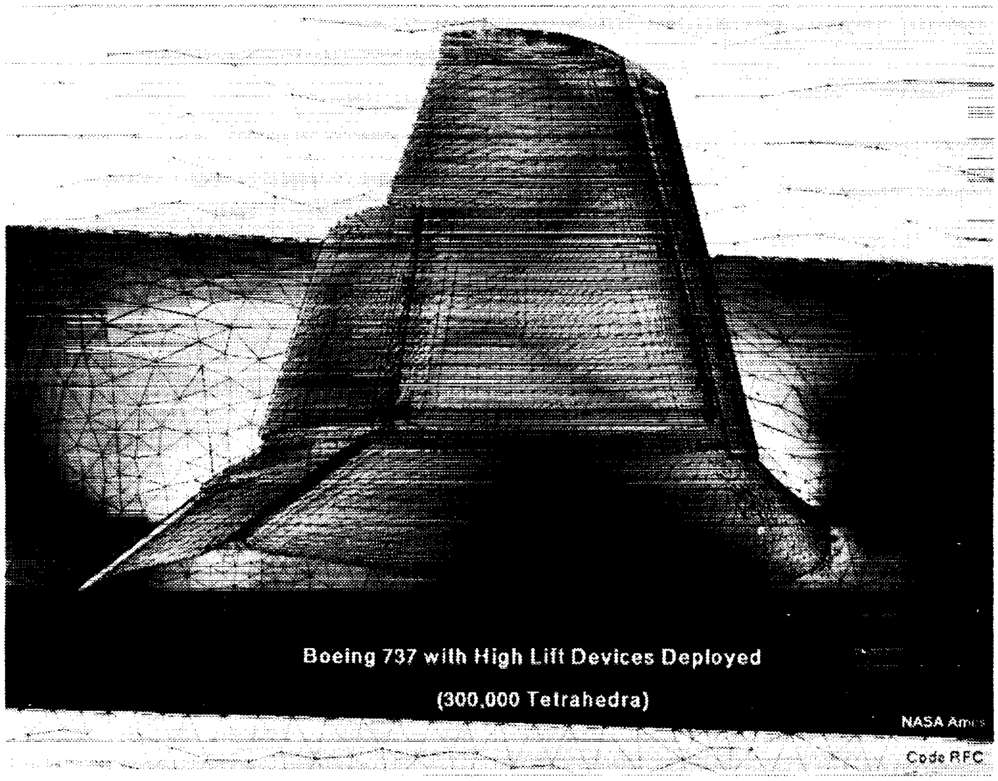
Surface Triangulation



Constrained/Conforming Triangulation of Boundary



Final Volume Triangulation

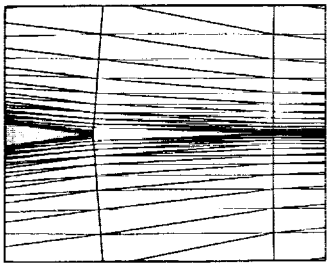


Boeing 737 with High Lift Devices Deployed

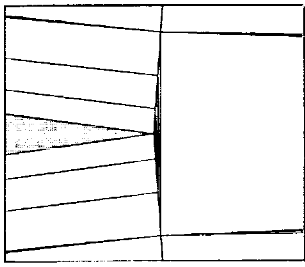
(300,000 Tetrahedra)

Why Some Standard Triangulation Methods Fail

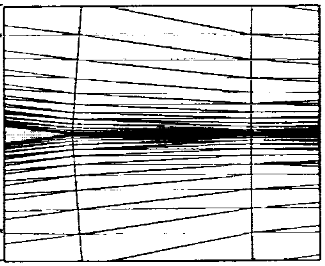
- Delaunay triangulation has a well known characterization that it maximizes the minimum angle for triangle pairs
- Theoretical and practical considerations indicate that it may be more beneficial to minimize the maximum angle for triangle pairs
- Incremental insertion and local optimization can be used to produce locally optimal Min–Max triangulations



Delaunay triangulation near an airfoil trailing edge



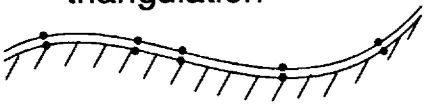
Extreme closeup of DT in trailing edge region



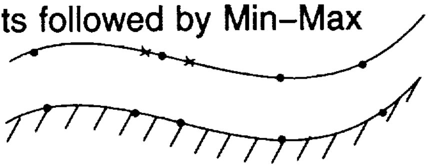
Min–Max triangulation obtained by local optimization

Viscous Mesh Generation

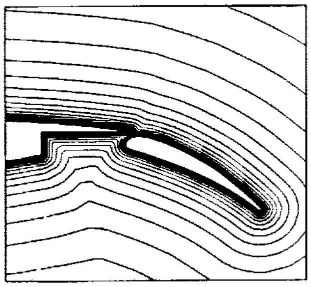
- Automatic generation of viscous meshes by adaptive placement of sites on level sets followed by Min–Max triangulation



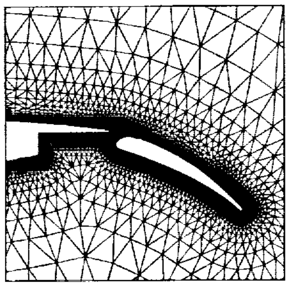
Point Selection (AR >>1)



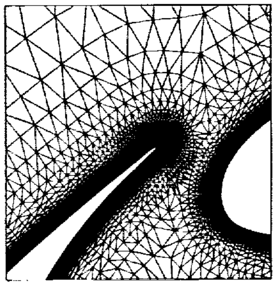
Point Selection and Adaptation (AR ≅ 1)



Distance Function



Min–Max Triangulation



Closeup in Flap Region

Future Directions

- Continue investigating optimization criteria for tetrahedral meshes
- Develop new strategies for site placement
    - Level set strategies
    - Steiner point strategies
- Solution adaptation based on *a priori* error estimates

STATUS OF VGRID/USM3D AERO ANALYSIS SYSTEM

NEAL T. FRINK
NASA LANGLEY RESEARCH CENTER

PARESH PARIKH
ViGYAN

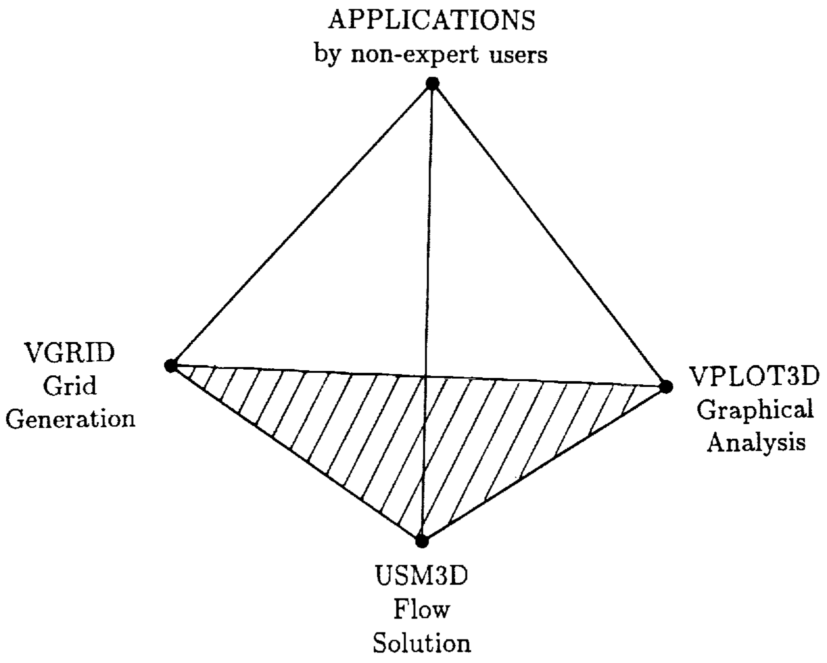
SHAHYAR PIRZADEH
ViGYAN

Outline

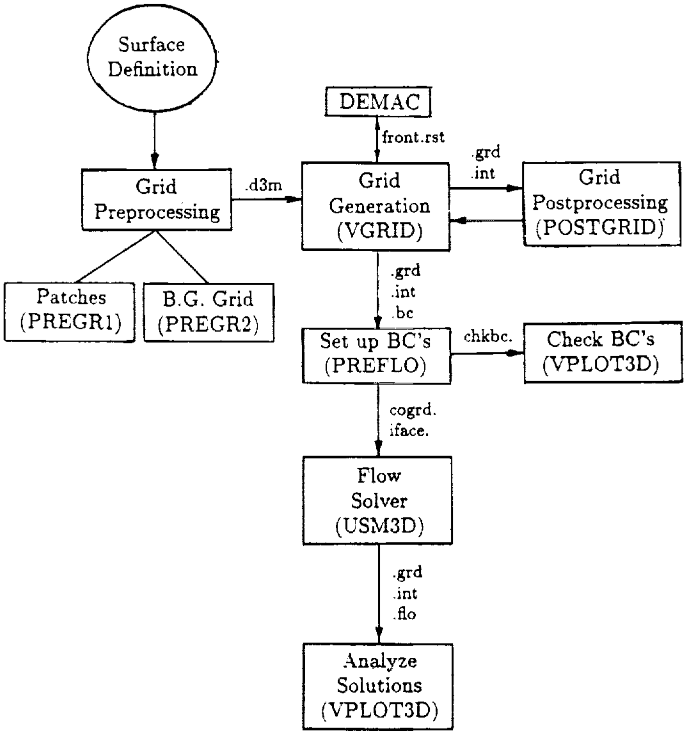
- Introductory Remarks
- General Capabilities
  - Grid generation
  - Flow solver
  - Graphic Postprocessing
- Dissemination
- Customer Applications
- Plans
- Closing Remarks

The Structure Behind Our Unstructured Work

- An Application-Oriented Development Program -



Flowchart for Unstructured Codes



Unstructured Grid Generation, VGRID

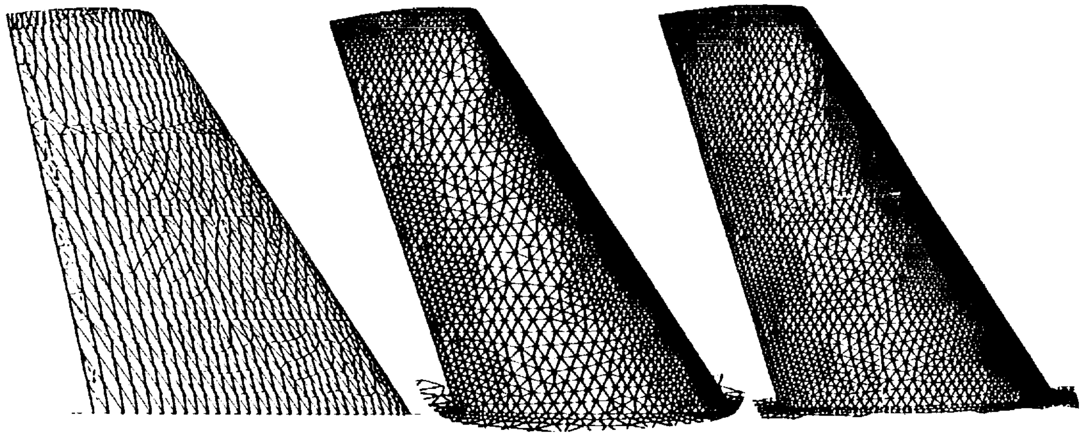
- A program for generation of unstructured tetrahedral grids around complex configurations using the Advancing Front Method.
  - o Base code developed under SBIR with ViGYAN
  - o Considerable extentions made in TAB to improve:
    - robustness
    - grid quality
    - reduced grid generation time
  - o Viscous grid generation effort well underway

- Additional enhancements made by GEOLAB/CSC
  - o Surface projection/correction
  - o New graphic interface tool under development
    - Enhanced surface patches
    - Improved surface grid generation

Unstructured Euler Solver, USM3D

- Finite-volume approach with cell-centered, tetrahedral elements
- Upwind-biased, flux-difference splitting (Roe's Scheme)
- Fast higher-order differencing formula
- Three-stage Runge-Kutta time stepping to advance to steady state
- Acceleration techniques:
  - ○ Local time stepping
  - ○ Implicit residual smoothing
- Efficient data structure:
  - ○ CPU time: 17.5 μ-sec/cell/cycle on Cray Y-MP
  - ○ Memory usage: 45 words/cell

Upper Surface Grid
OM6 Wing

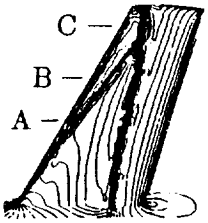


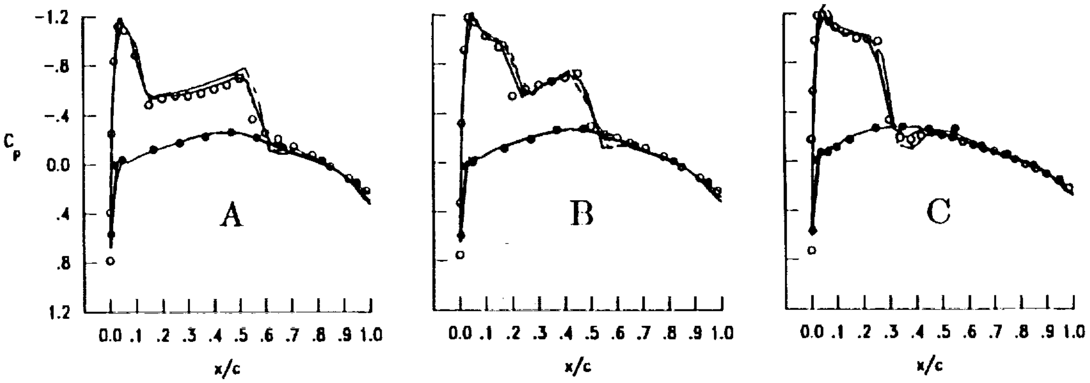
	Stretch	Coarse	"Workshop" Fine
No. Cells	= 35008	= 108755	= 231507
No. Nodes	= 6910	= 20412	= 42410

Effect of Grid on Chordwise Surface Pressure Distributions

USM3D, $M_\infty = 0.84$, $\alpha = 3.06°$

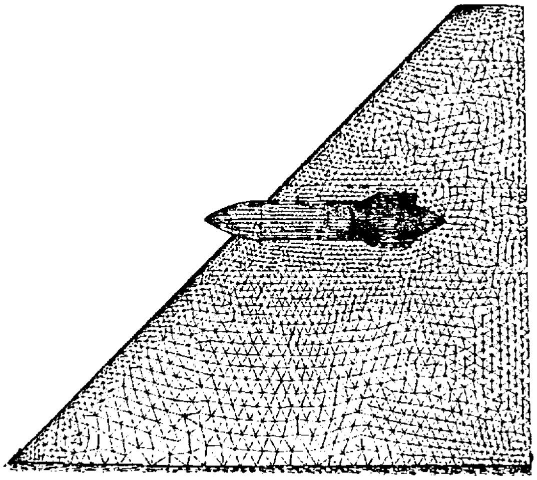
		Memory	CRAY2S Run time
o ●	Data		
———	Stretch,	2.3MW,	11.5 min
- - - - - - -	Coarse,	7.0MW,	1 hr 29 min
—— - ——	Fine,	14.9MW,	3 hr 23 min



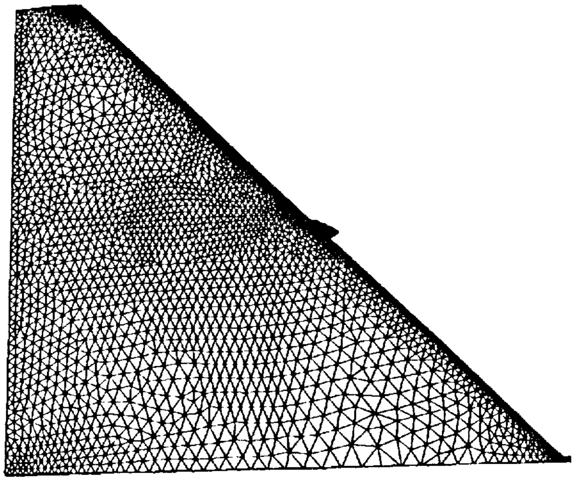


SURFACE GRID ON THE CONFIGURATION

13,256 Points
27,044 Faces

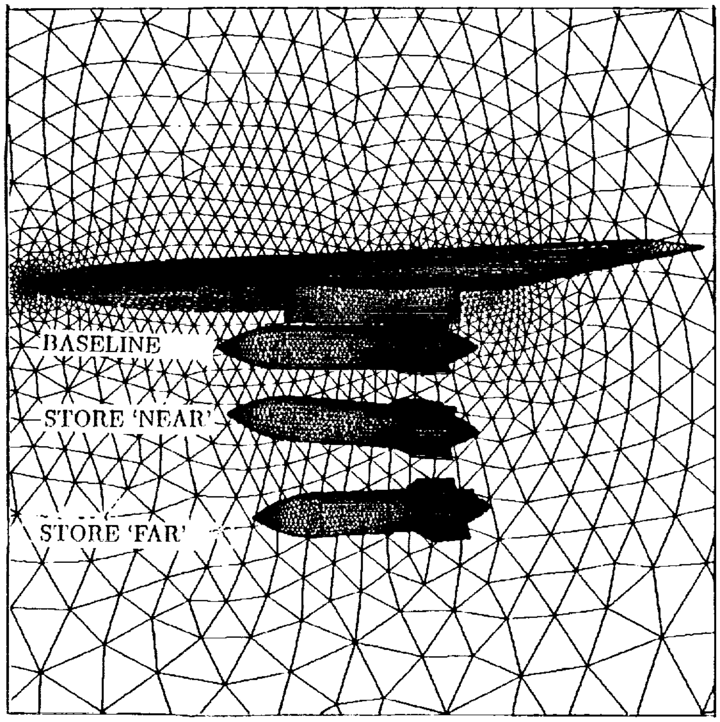


Lower Surface

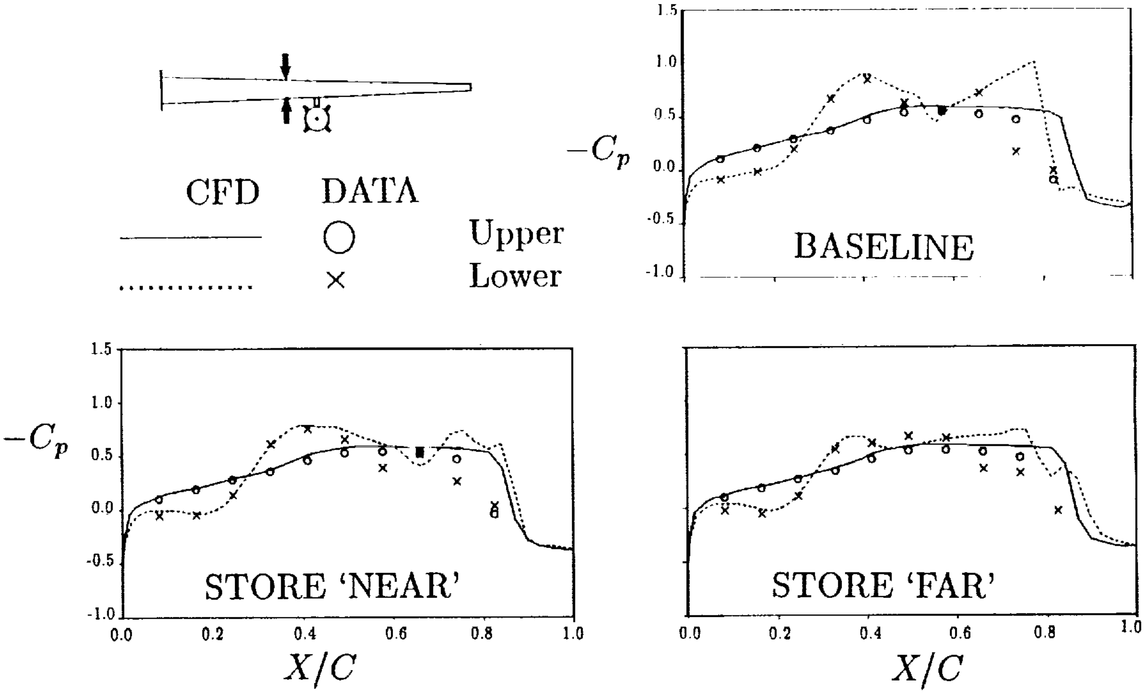


Upper Surface

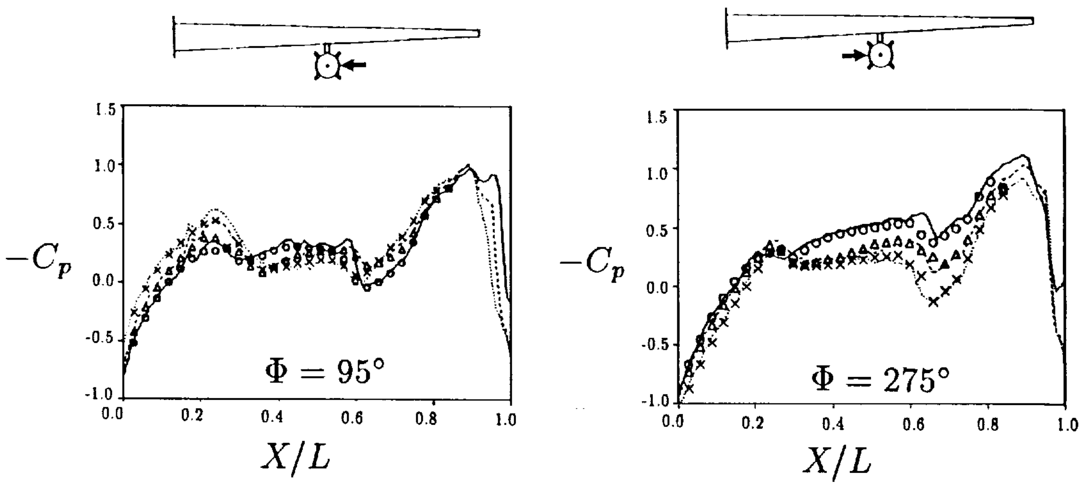
REPRESENTATIVE STORE LOCATIONS



PRESSURE COMPARISON ON THE WING, $M_\infty = 0.95$
Location: 1.2 Store Diameter Inboard



SURFACE PRESSURE COMPARISON ON THE STORE
$M_\infty = 0.95$

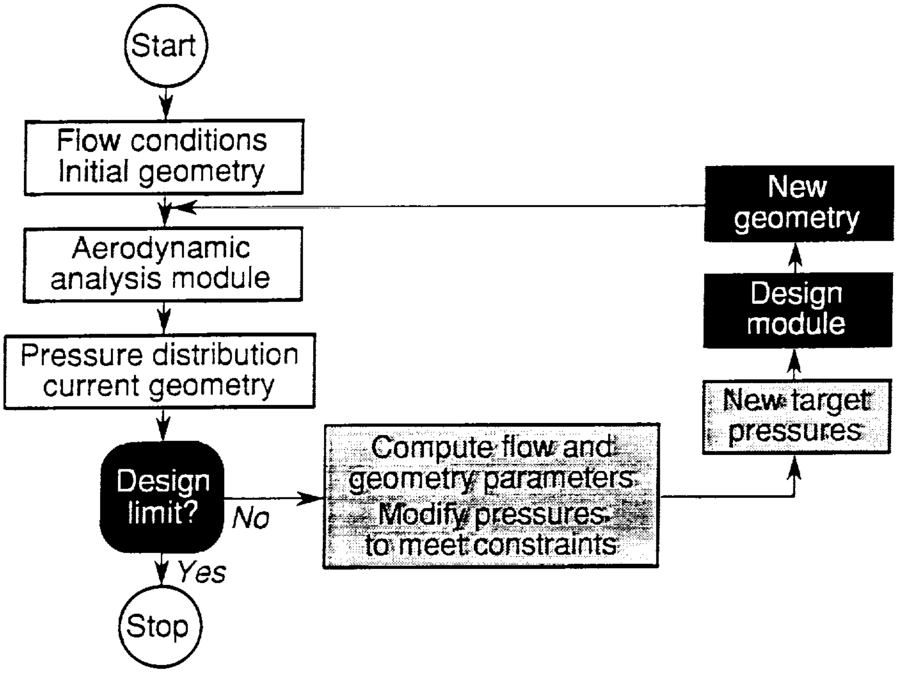


CFD	DATA	
————	○	BASELINE
- - - - - -	△	STORE 'NEAR'
....	×	STORE 'FAR'

Recent Improvements to USM3D

- Implemented 2nd-order nodal averaging technique
  - ○ higher-order boundary condtions
- Improved data structure through face coloring
- Teamed with Dr. Kyle Anderson, CAB/FlMD, to install his implicit time integration algorithm and FVS
- Iterative design capability installed by L. A. Smith, TAB/AAD

CONSTRAINED DIRECT ITERATIVE SURFACE CURVATURE (CDISC) DESIGN METHOD



Start

Flow conditions
Initial geometry

Aerodynamic analysis module

Pressure distribution current geometry

Design limit?

No

Yes

Stop

Compute flow and geometry parameters
Modify pressures to meet constraints

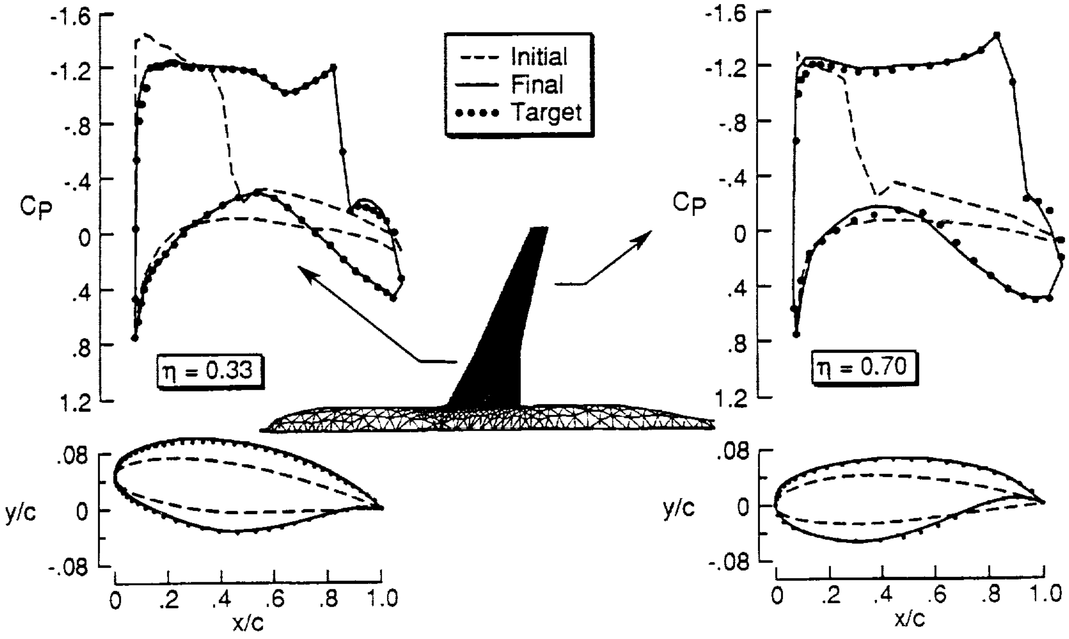
New target pressures

Design module

New geometry

TRANSONIC WING DESIGN USING THE DISC DESIGN METHOD AND USM3D

M = .77



Dissemination of VGRID/USM3D Developmental Codes

- Academia - 4 universities

- Government
  - 3 NASA research centers
  - 3 Air Force research laboratories
  - 2 Naval air research/development centers
  - National Institute of Standards and Technology

- Industry - 11 companies, including 4 major aircraft companies

- Total of 30 outside requests

- Provided hands-on training to 48 users

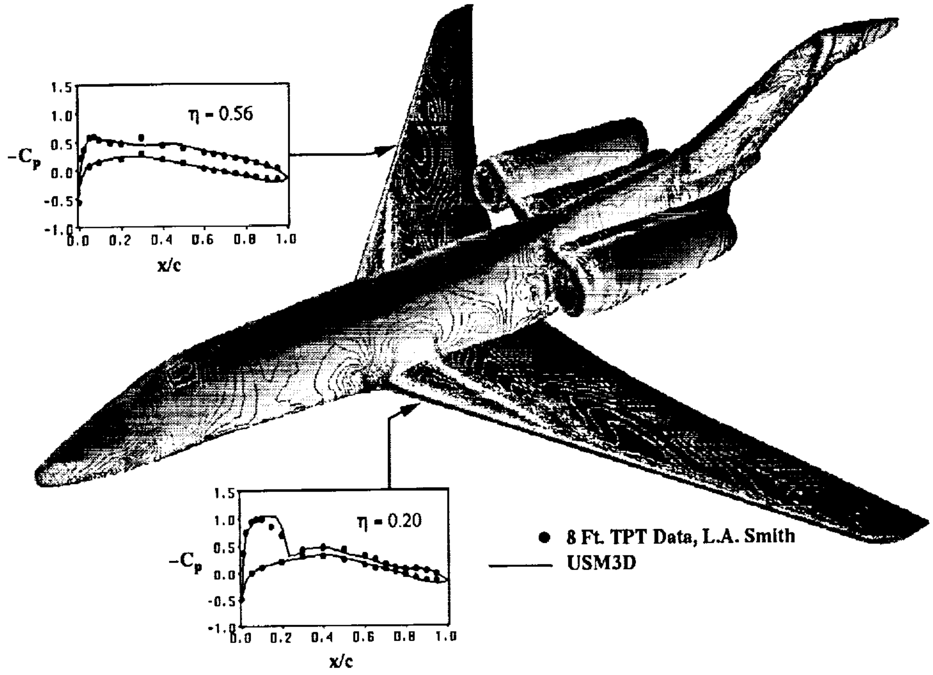
Selected Customer Applications

- Subsonic Aircraft
  - Cessna Citation - (Cessna/Parikh)
  - MD-11 - (Douglas/NASA)
  - B737 - (SAB, S. Dodbele)
  - C-17 - (HRNAB, J.Alsaadi)
  - T-39 - (WPAFB, J. Slavey)
- High-Speed Civil Transport
  - Generic HSR Configuration - (SAB K. Kjerstad)
  - Cranked wing LEVF - (SAB, K. Kjerstad)
  - HSCT - (Boeing, J. Wai)
  - Sonic Boom research - (VIB, K. Fouladi)
- High-Performance Military Aircraft
  - Fighter - (Boeing, J. Wai)
  - Joined wing - (Boeing, J. Wai)
  - MTVI - (TAB, F. Ghaffari)
- Other
  - Cavities - (TAB Cavity Flow Team)
  - Internal flow - (NASA LeRC, O.J. Kwon)

Comparison of C_p Distributions on Cessna Citation 10

Mach = 0.82, $\alpha = 1.11^{\circ}$

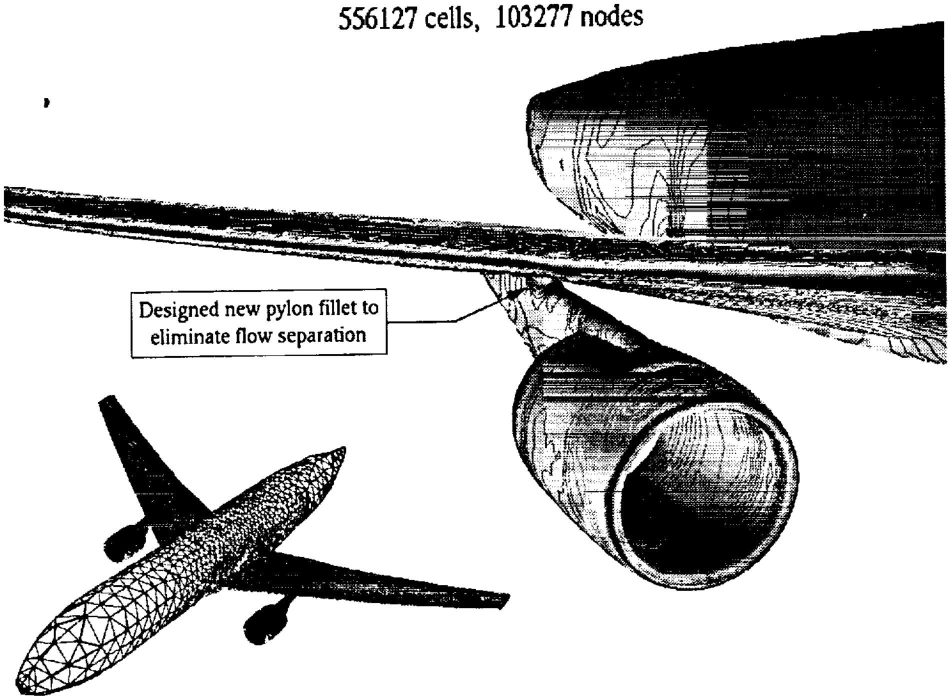
762553 cells, 137742 nodes

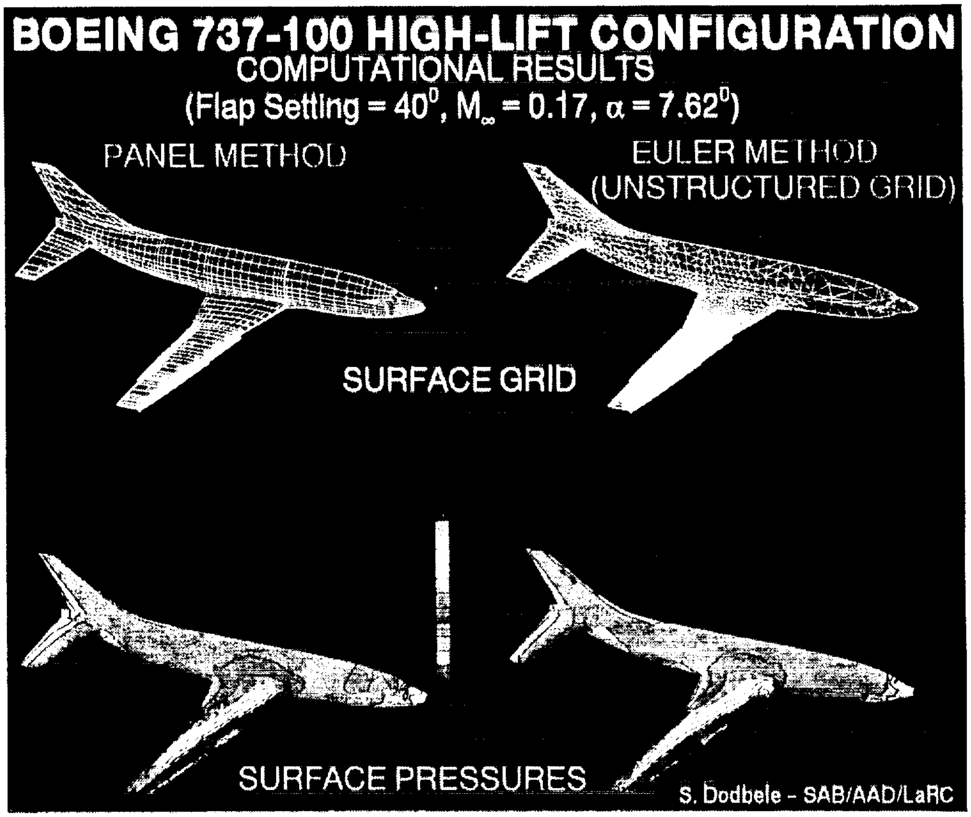


Wing-Pylon Fillet Design Using USG Methodology

MD-11 Configuration, Mach=0.83, $\alpha = 2.35^{\circ}$

556127 cells, 103277 nodes





BOEING 737-100 HIGH-LIFT CONFIGURATION

COMPUTATIONAL RESULTS

(Flap Setting = 40^0, $M_\infty = 0.17$, $\alpha = 7.62^0$)

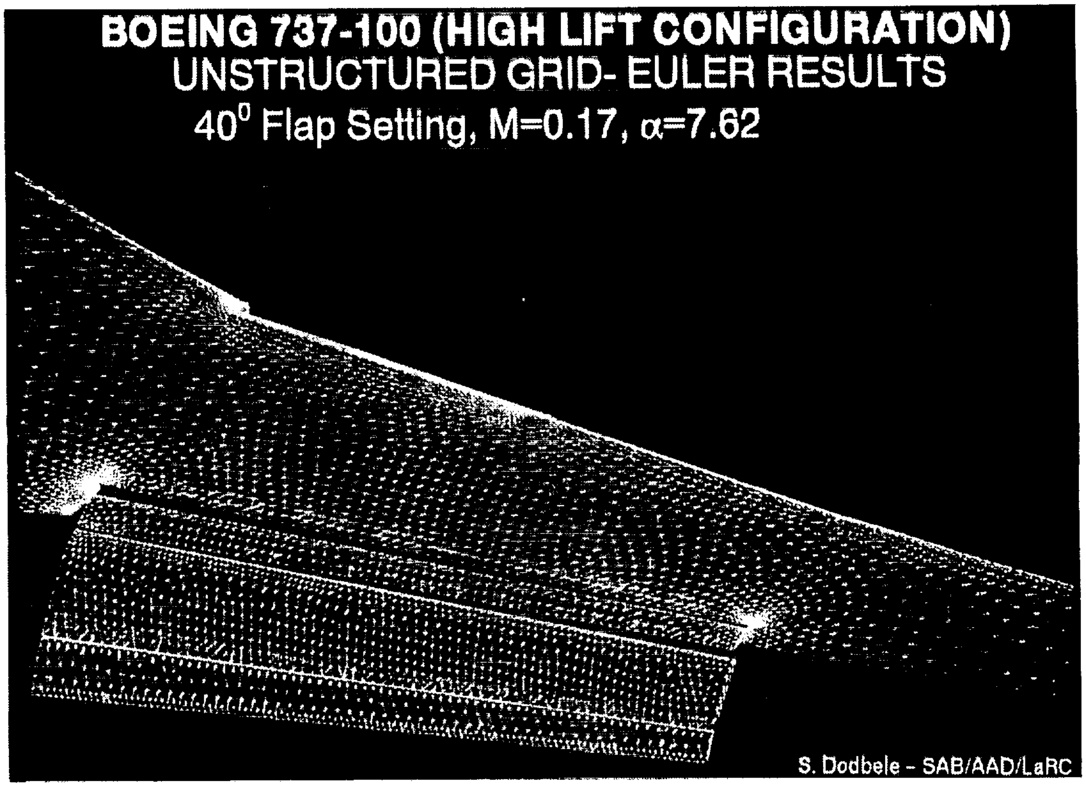
PANEL METHOD

EULER METHOD
(UNSTRUCTURED GRID)

SURFACE GRID

SURFACE PRESSURES

S. Dodbele - SAB/AAD/LaRC



BOEING 737-100 (HIGH LIFT CONFIGURATION)

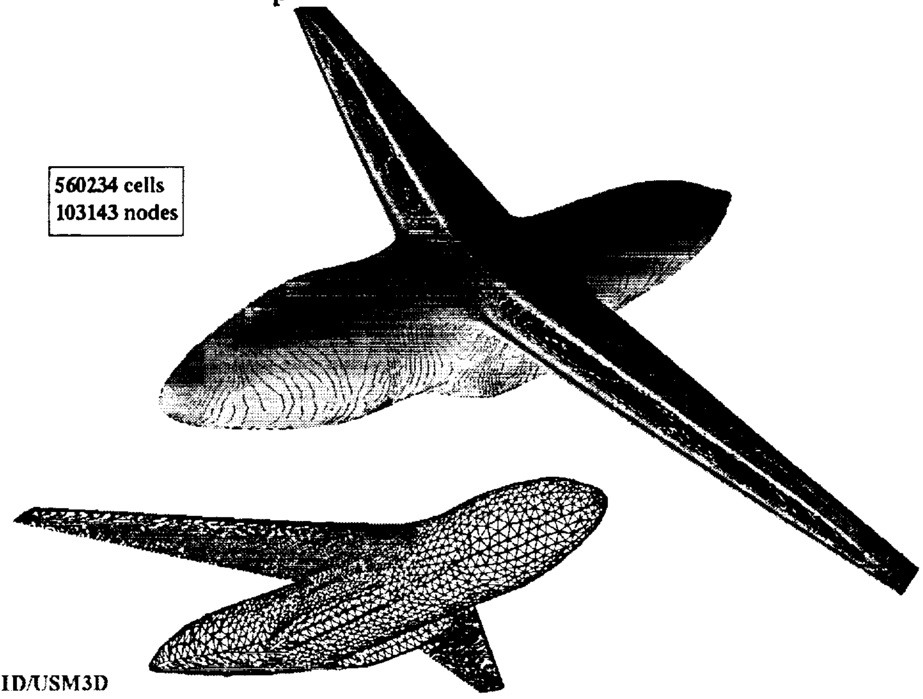
UNSTRUCTURED GRID- EULER RESULTS

40^0 Flap Setting, M=0.17, α=7.62

S. Dodbele - SAB/AAD/LaRC

High-Wing Transport Configuration

C_p Contours, Mach=0.77, α - 1.6°

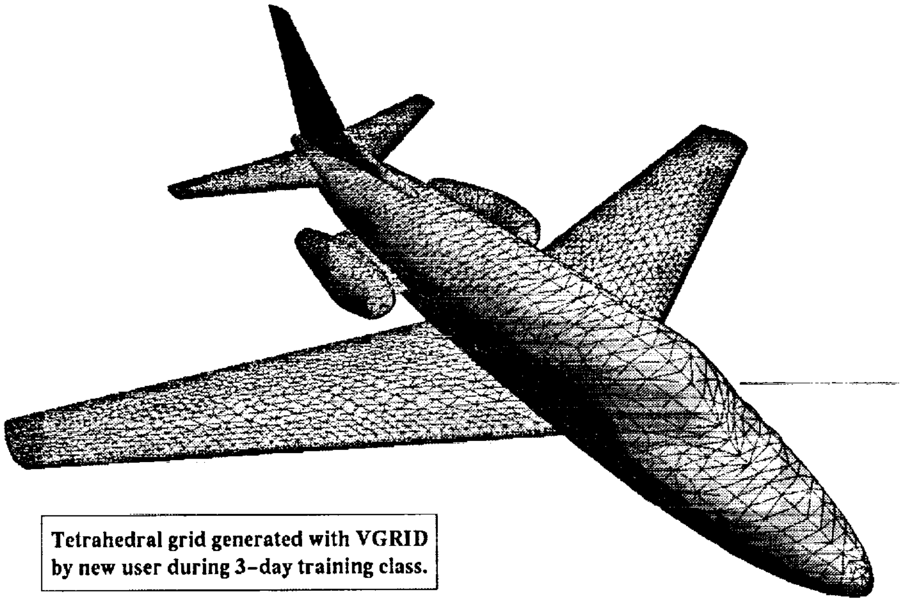


560234 cells
103143 nodes

VGRID/USM3D

Unstructured Grid for T-39 Aircraft

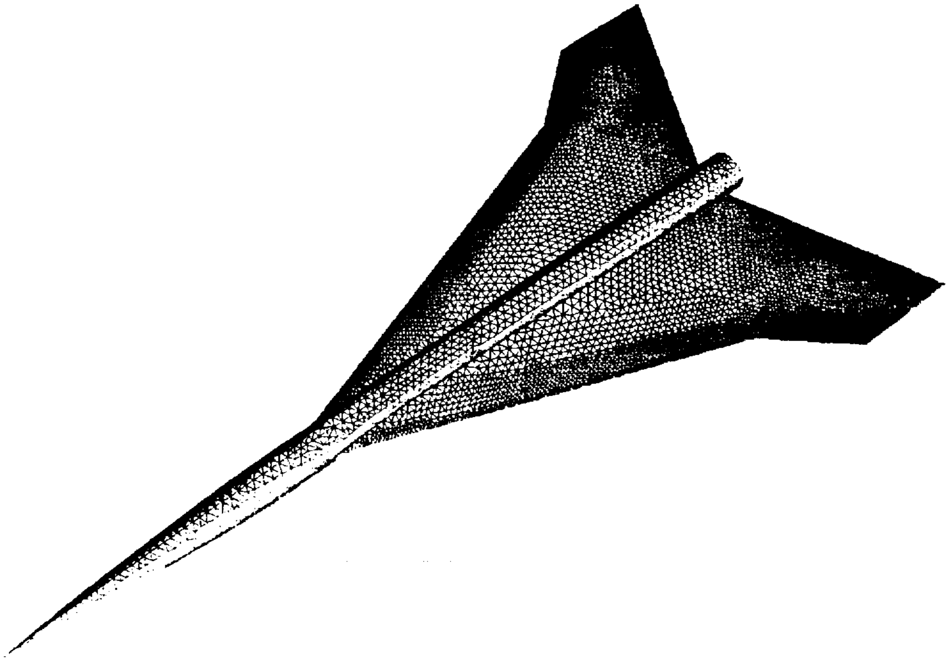
244156 cells, 46050 nodes

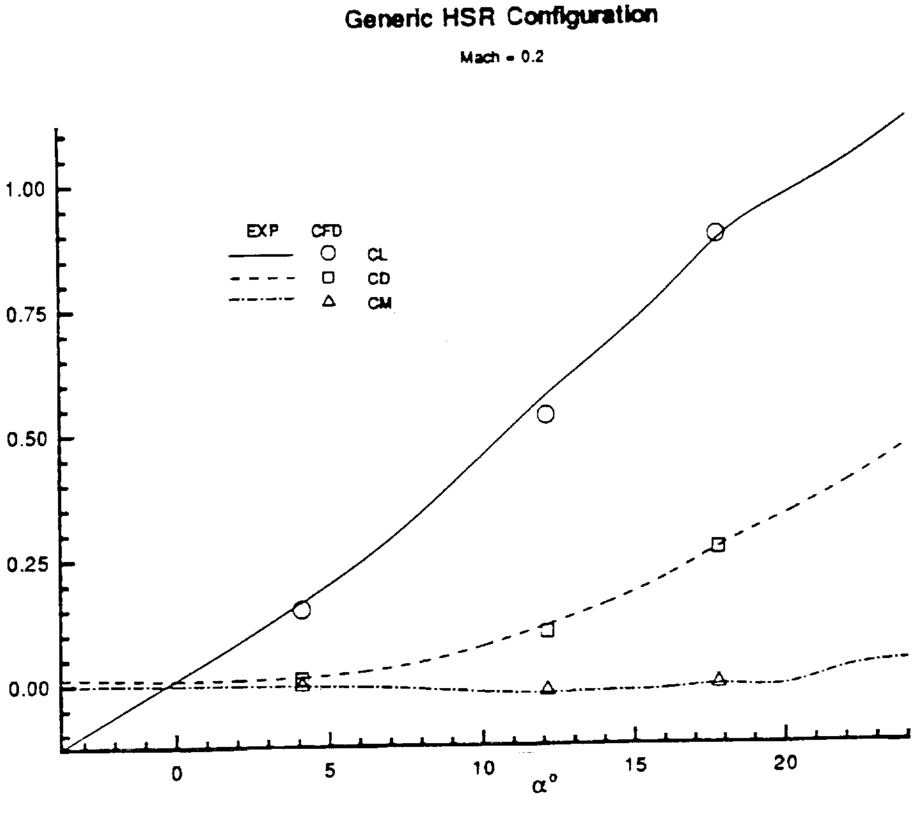


Tetrahedral grid generated with VGRID
by new user during 3-day training class.

Generic HSR Configuration

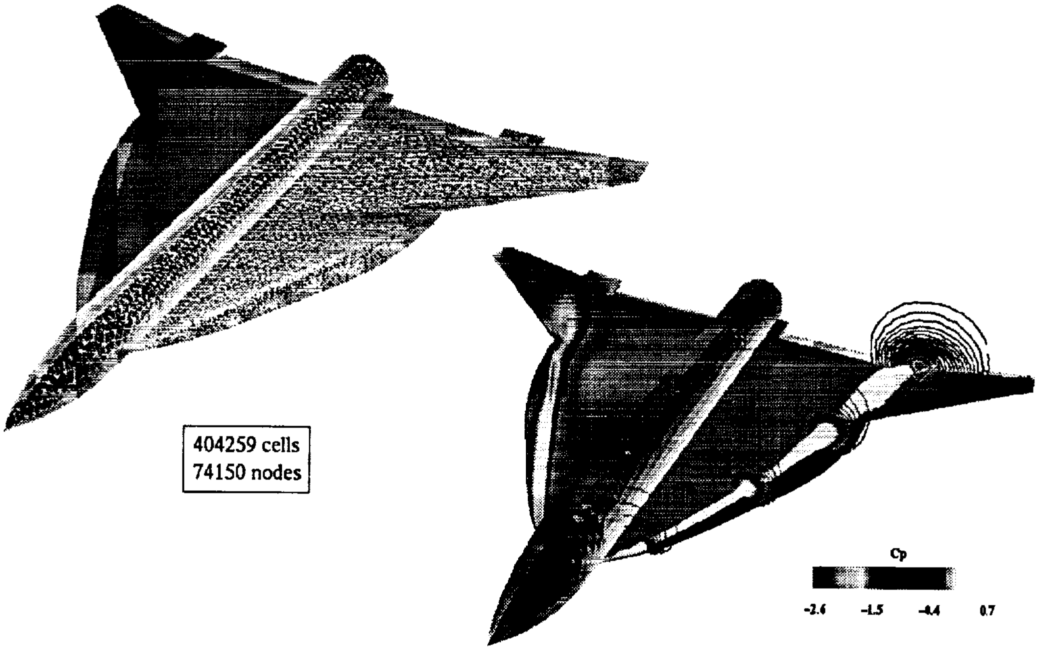
Unstructured Grid





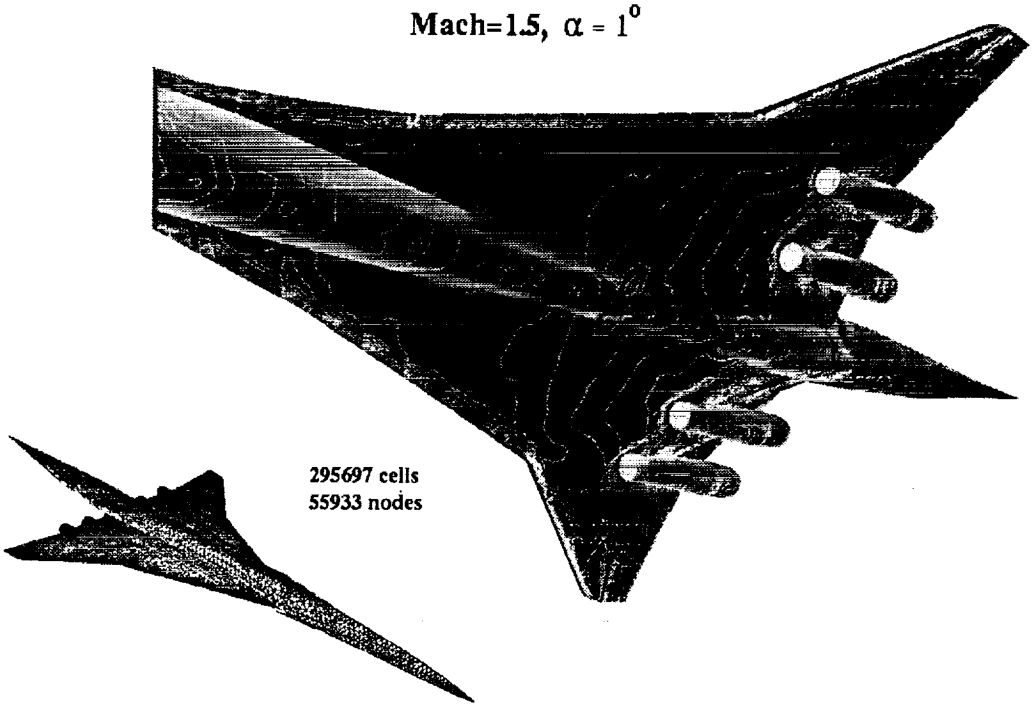
HSR Planform Study (VGRID/USM3D)

68/48 planform with $\delta_{vf} = 30^\circ$, $\delta_{te} = 15^\circ$, Mach=0.22, $\alpha = 12^\circ$

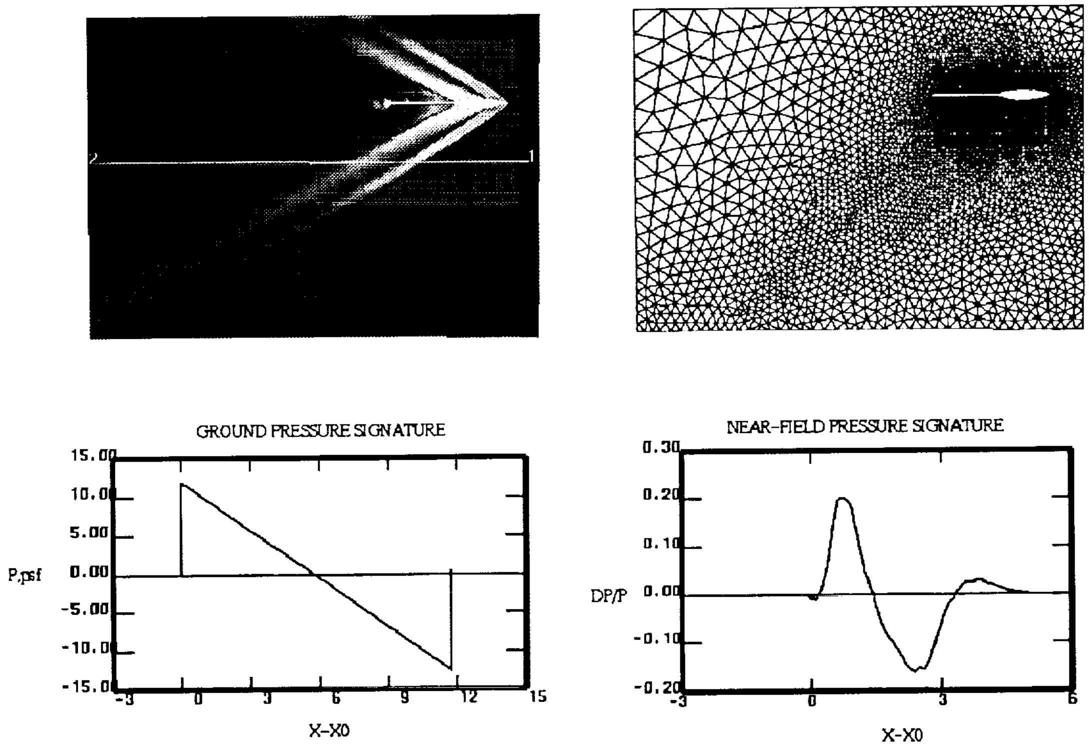


HSR C_p Distribution Using USM3D

Mach=1.5, $\alpha = 1^\circ$



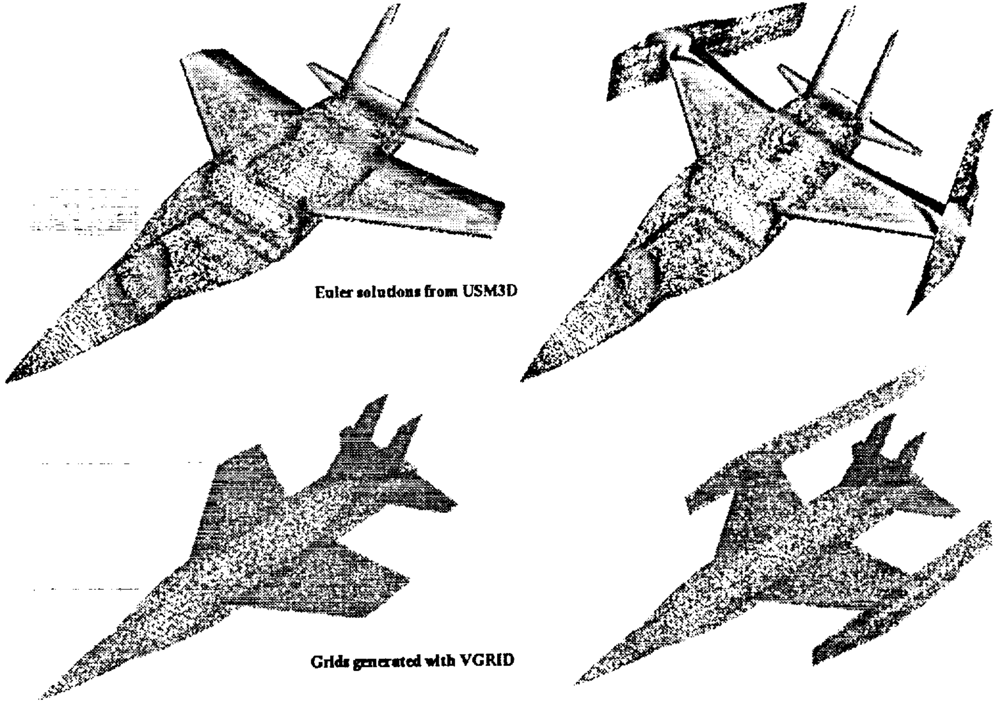
SONIC BOOM ANALYSIS OF A BODY OF REVOLUTION



Boeing Multirole Fighter Configuration

Assessment of Tunnel Installation Interference

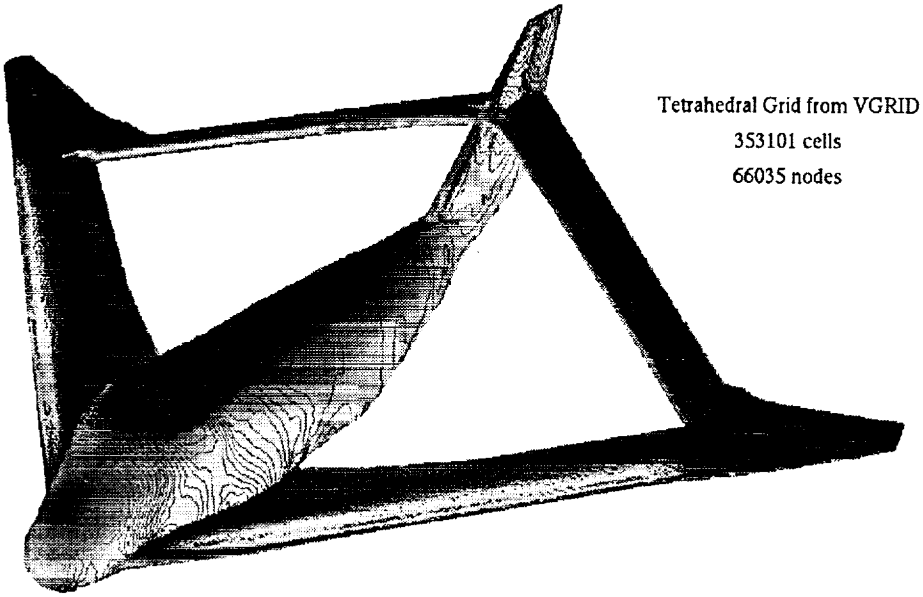
Mach=0.9, $\alpha = 3°$



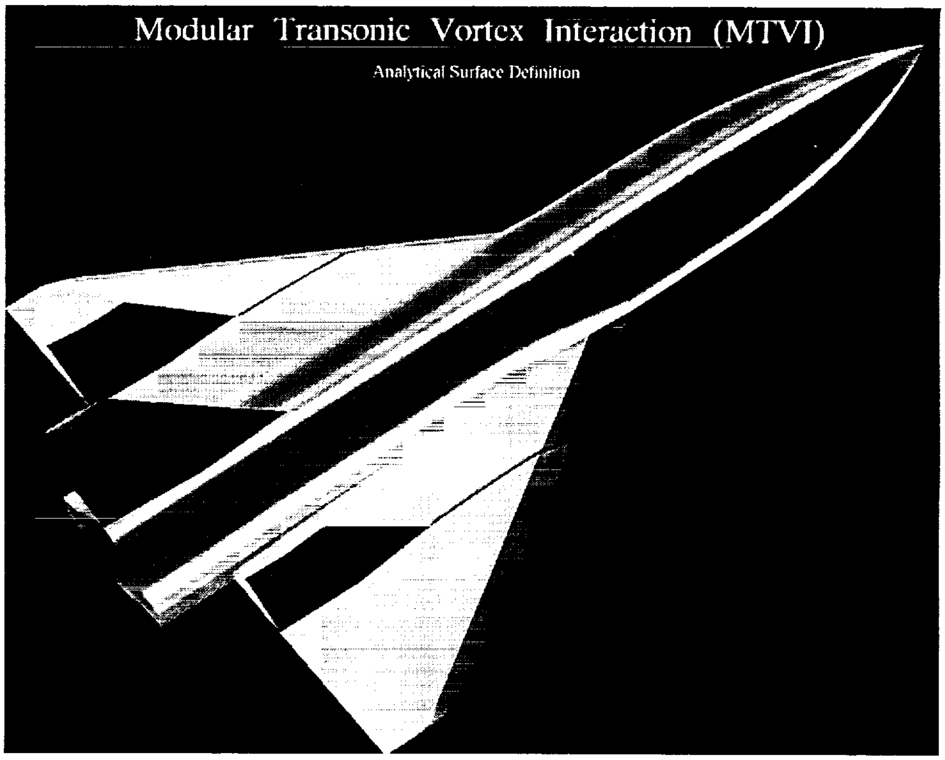
Boeing Joined-Wing Configuration

Cp Distribution from USM3D

Mach=.38, $\alpha = 4^{\circ}$



Tetrahedral Grid from VGRID
353101 cells
66035 nodes

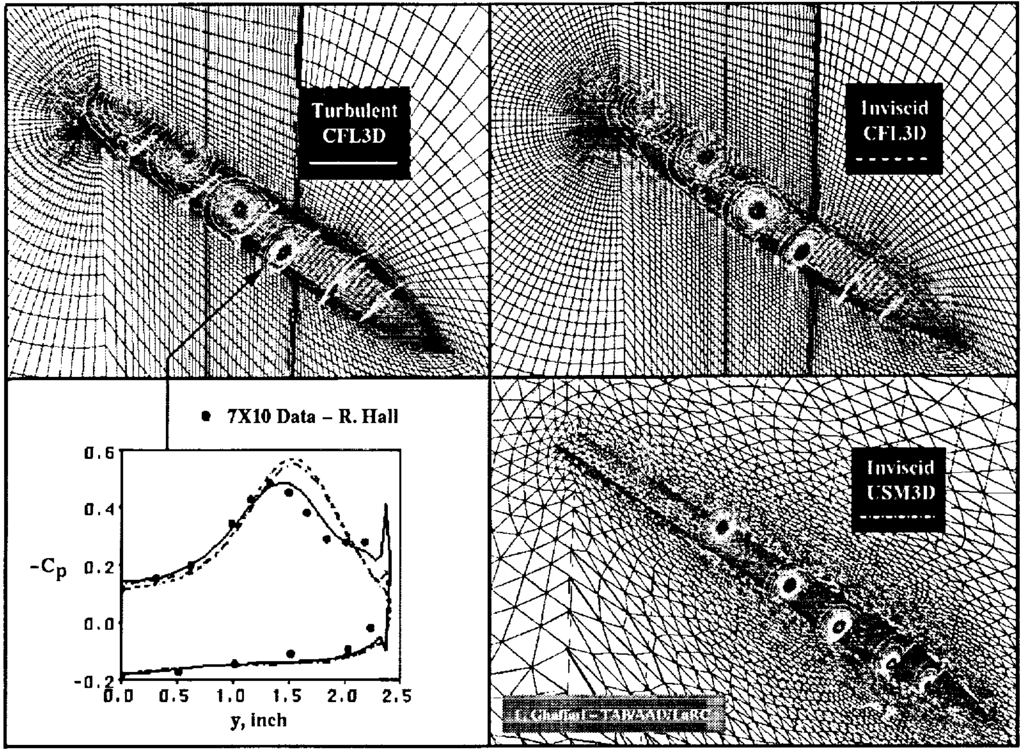


Modular Transonic Vortex Interaction (MTVI)

Analytical Surface Definition

Structured/Unstructured Code Validation Study

Isolated MTVI Fuselage Configuration, Mach=0.4, $\alpha = 20^{\circ}$



Planned Capabilities
(work underway)

- One-day turnaround for inviscid problems
- Viscous grid generation (2D and 3D)
- 3-D viscous flow solver
- Solution adaptive grids
- Dynamic moving grids (ODU contribution)

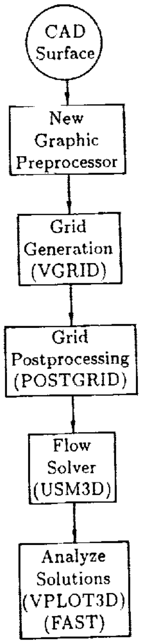
User Related Plans

- Establishment of VGRID/USM3D local user's group

- Release/training for VGRID Version 2.5 on June 1, 1993
  - o New graphic interface with consolidated preprocessing functions
  - o More generalized surface patches with T-intersection feature

- VGRID Version 3.0 to be released later in Summer 1993
  - o Direct surface triangulation with n-sided patches
  - o More consolidation entire flow analysis process
  - o Use of more standardized file formats

Flowchart for Version 3.0 USG System
Release in late Summer 1993



Note: All codes to be interfaced with common file formats

Closing Remarks

- Assembled an integrated aerodynamic analysis and design capability using state-of-the-art three-dimensional USG technology

- Ongoing application-oriented development program dependent on feedback from wide user base

- Grid generation time for complex geometries now measured in days for experienced users

- Made significant advances in overall technology through teaming

UNSTRUCTURED LOW-MACH NUMBER VISCOUS FLOW SOLVER

PHILIP C. E. JORGENSON
NASA LEWIS RESEARCH CENTER

Outline

- Governing Equations
- Grid Generation
- Numerical Approach
  - Discretization Technique
  - Preconditioning
  - Artificial Dissipation
  - Boundary Conditions
  - Sparse Matrix Solvers
- Results
- Conclusions

Unstructured Low-Mach Number Viscous Flow Solver

- Navier-Stokes equations(2-D)
  - Conservation law form in terms of primative variables substitute $\rho = \frac{P}{RT}$
  - Cell centered finite volume discretization
  - Implicit delta formulation written as: $A\vec{x} = \vec{b}$

Navier-Stokes equations nondimensionalized

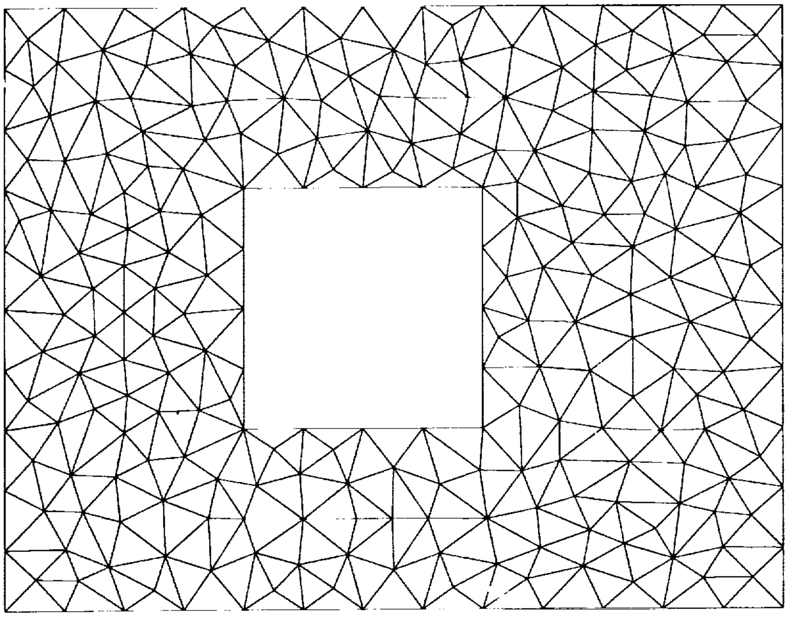
$$\frac{\partial Q(\tilde{w})}{\partial \tilde{t}} + \frac{\partial G(\tilde{w})}{\partial \tilde{x}} + \frac{\partial H(\tilde{w})}{\partial \tilde{y}} = 0$$

$$\tilde{w} = \begin{bmatrix} \tilde{P} \\ \tilde{u} \\ \tilde{v} \\ \tilde{T} \end{bmatrix}, \quad Q = \begin{bmatrix} \dfrac{\tilde{P}}{\tilde{T}} \\ \dfrac{\tilde{P}\tilde{u}}{\tilde{T}} \\ \dfrac{\tilde{P}\tilde{v}}{\tilde{T}} \\ \dfrac{\tilde{P}}{\tilde{T}}\left[\left(\tilde{C}_p - \tilde{R}\right)\tilde{T} + \dfrac{\tilde{u}^2}{2} + \dfrac{\tilde{v}^2}{2}\right] \end{bmatrix}$$

Unstructured Grid Generation

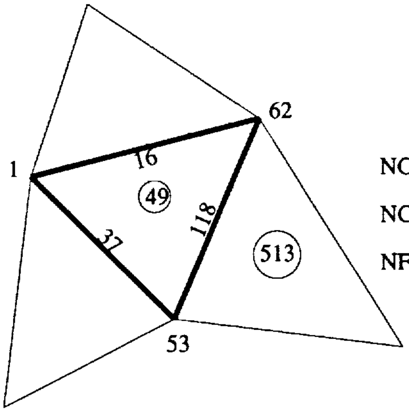
- Delaunay Triangulation
    - Bowyer's Algorithm
      Grid refinement based on aspect ratio, area, circumcircle radius

- Connectivity



Grid Code Output: Geometry, Connectivity

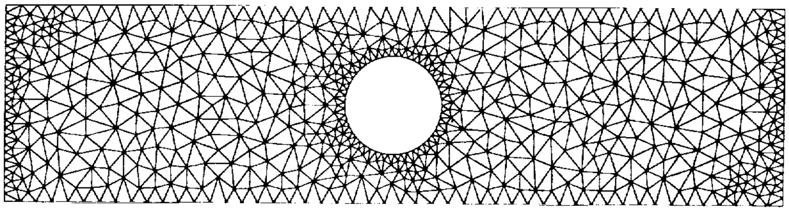
- Node point x, y coordinates
- Cell nodes, cell faces, face cells



NCELL(1:3,49)=37,118,16

NCELL(4:6,49)=1,53,62

NFACE(1:2,118)=49,513



Preconditioning(N-S eq. 1-d)

$$\frac{\partial Q(w)}{\partial t} + \frac{\partial G(w)}{\partial x} - \frac{\partial G_v(w)}{\partial x} = 0$$

where

$$w = \begin{bmatrix} P \\ u \\ T \end{bmatrix}, \quad Q = \begin{bmatrix} \frac{P}{RT} \\ \frac{Pu}{RT} \\ \frac{P}{\gamma R} + \frac{M^2(\gamma - 1)}{2}\frac{Pu^2}{RT} \end{bmatrix}$$

$$\mathbf{A}_t \frac{\partial w}{\partial t} + \mathbf{A}_x \frac{\partial w}{\partial x} = \frac{\partial G_v(w)}{\partial x}; \quad R = (\gamma M^2)^{-1}$$

where

$$\mathbf{A}_t = \begin{bmatrix} \dfrac{\gamma M^2}{T} & 0 & -\dfrac{P}{RT^2} \\ \dfrac{\gamma M^2 u}{T} & \dfrac{P}{RT} & -\dfrac{Pu}{RT^2} \\ M^2 + \dfrac{\gamma M^4(\gamma-1)u^2}{2T} & \dfrac{\gamma M^4(\gamma-1)Pu}{T} & -\dfrac{\gamma M^4(\gamma-1)Pu^2}{2T^2} \end{bmatrix},$$

$$\mathbf{A}_x = \begin{bmatrix} \dfrac{\gamma M^2 u}{T} & \dfrac{P}{RT} & -\dfrac{Pu}{RT^2} \\ \dfrac{\gamma M^2 u^2}{T} + 1 & \dfrac{2Pu}{RT} & -\dfrac{Pu^2}{RT^2} \\ \gamma M^2 u + \dfrac{\gamma M^4(\gamma-1)u^3}{2T} & \dfrac{P}{R} + \dfrac{3M^4(\gamma-1)Pu^2}{2T} & -\dfrac{\gamma M^4(\gamma-1)Pu^3}{2T^2} \end{bmatrix}$$

$$\frac{\partial w}{\partial t} + \mathbf{A}_t^{-1}\mathbf{A}_x \frac{\partial w}{\partial x} = \mathbf{A}_t^{-1} \frac{\partial G_v(w)}{\partial x}$$

instead

$$\mathbf{A}_p \frac{\partial w}{\partial \tau} + \mathbf{A}_t \frac{\partial w}{\partial t} + \mathbf{A}_x \frac{\partial w}{\partial x} = \frac{\partial G_v(w)}{\partial x}$$

$$\mathbf{A}_p = \begin{bmatrix} \dfrac{1}{T} & 0 & -\dfrac{P}{RT^2} \\ \dfrac{u}{T} & \dfrac{P}{RT} & -\dfrac{Pu}{RT^2} \\ \dfrac{1}{\gamma} + \dfrac{M^2(\gamma-1)u^2}{2\gamma T} & \dfrac{\gamma M^4(\gamma-1)Pu}{T} & -\dfrac{\gamma M^4(\gamma-1)Pu^2}{2T^2} \end{bmatrix}$$

Boundary Conditions

- Implicit treatment
- Solid wall specified as viscous no-slip or inviscid tangency.
- Symmetry and periodic boundaries are treated through connectivity

Characteristics of Preconditioned System(N-S eq. 1-d)

Find characteristics of $\mathbf{A}_p^{-1}\mathbf{A}_x$ instead of $\mathbf{A}_x$.

$$\mathbf{A}_p^{-1} = \begin{bmatrix} \frac{\gamma M^2(\gamma-1)u^2}{2} & -\gamma M^2(\gamma-1)u & \gamma \\ -\frac{uRT}{P} & -\frac{RT}{P} & 0 \\ -\frac{((\gamma-1)u^2-2RT)T}{2P} & -\frac{(\gamma-1)Tu}{P} & \frac{\gamma RT}{P} \end{bmatrix}$$

$$\lambda_1 = U\,,\ \text{and}\ \lambda_{2,3} = \frac{U \pm \sqrt{U^2+4\gamma T}}{2}$$

when letting $M \rightarrow 0$. Preconditioning gives a finite instead of infinite value of λ.

Sparse Matrix Solvers

- Point Gauss-Seidel scheme
- Point block Gauss-Seidel scheme
- Conjugate gradient like method(SITRSOL)

used to solve

$$A\vec{x} = \vec{b}$$

Point Gauss-Seidel Scheme

- Every element of matrix except diagonal of block moved to RHS
- Prone to divergence with poor initial conditions
- Very sensitive to lack of diagonal dominance

Point Block Gauss-Seidel Scheme

- All blocks except diagonal block moved to RHS
- Uses LU decomposition to the remaining matrix equation
- More robust than the point G-S scheme

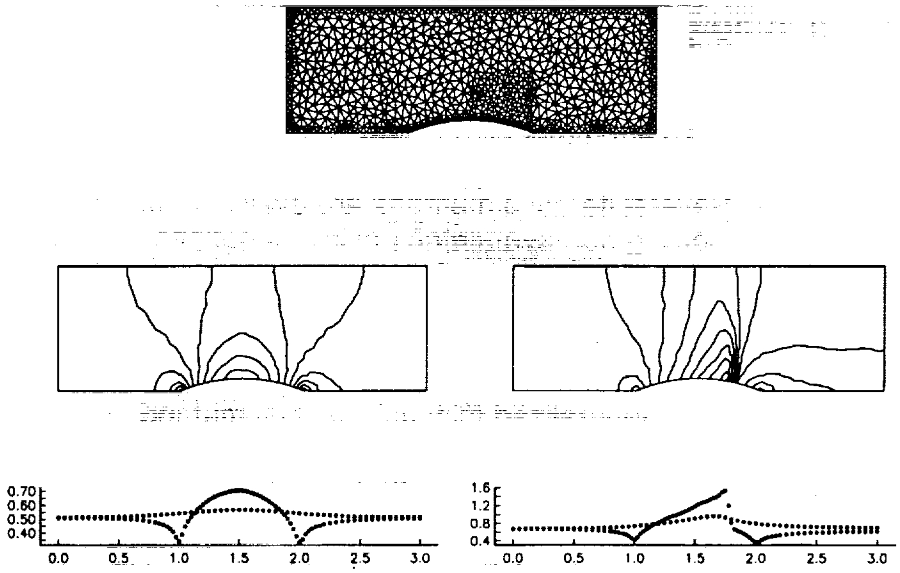
A grid coloring scheme was used to vectorize the Gauss-Seidel method since it suffers from recurrence. The four color theorem was used to remove the recurrence from the convective terms. Recurrence remains in the viscous terms but doesn't seem to affect the convergence rate. The coloring scheme was done by sweeping the computational cells twice.

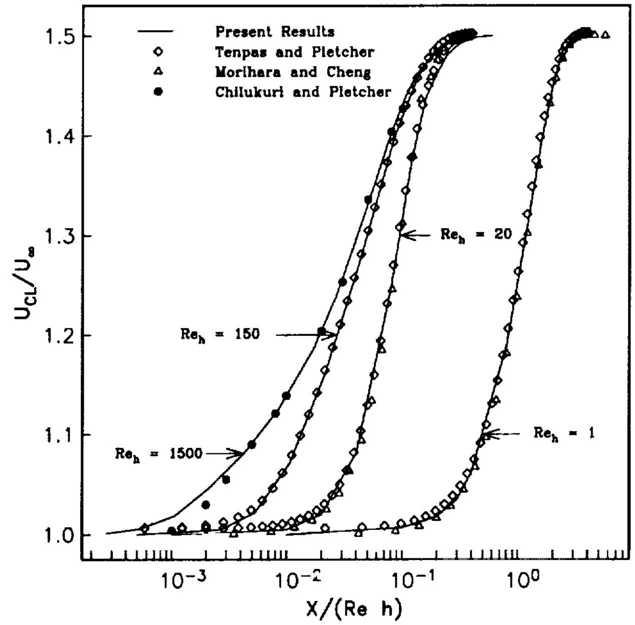
Conjugate gradient like solver(SITRSOL)

- Iterative solver based loosely on the conjugate gradient method
- Several iterative methods are available for solving non-symmetric positive indefinite sparse linear systems
  - Bi-conjugate gradient method
  - Generalized minimal residual method
  - Generalized conjugate residual method
- The incomplete LU preconditioner was used

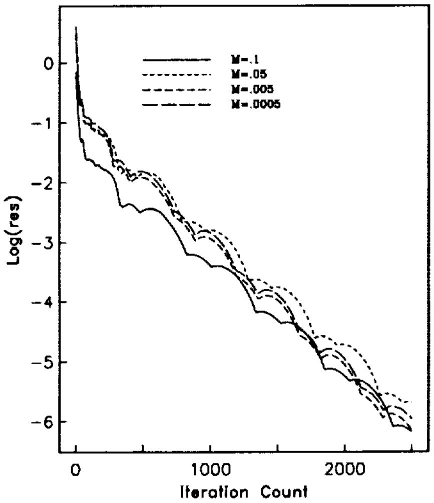
Results

- Bump on Wall
- Developing Channel Flow
- Sudden Expansion
- Periodic Tandem Circular Cylinders in Cross Flow
- Four Port Valve

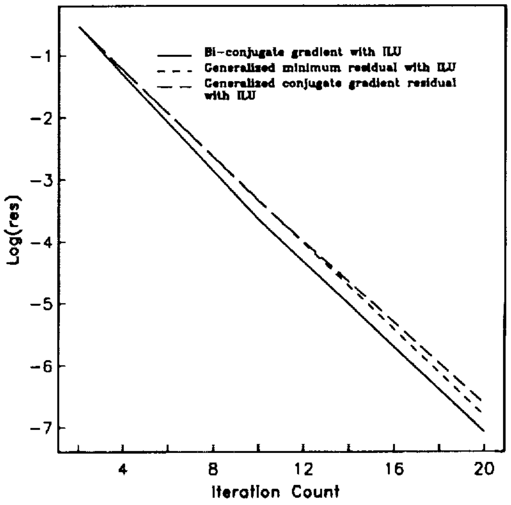




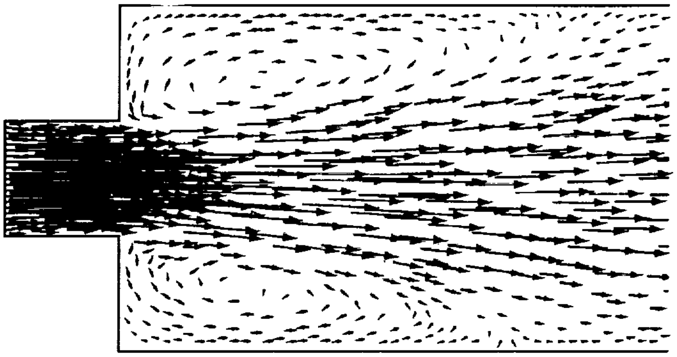
Present Results
Tenpas and Pletcher
Morihara and Cheng
Chilukuri and Pletcher
U_{CL}/U_s
1.5
1.4
1.3
1.2
1.1
1.0
Re_h = 20
Re_h = 150
Re_h = 1500
Re_h = 1
10^{-3}
10^{-2}
10^{-1}
10^{0}
X/(Re h)

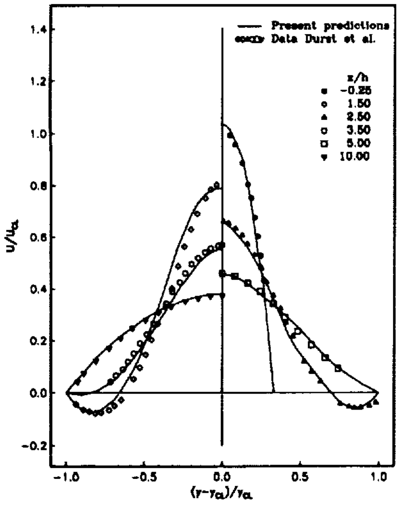


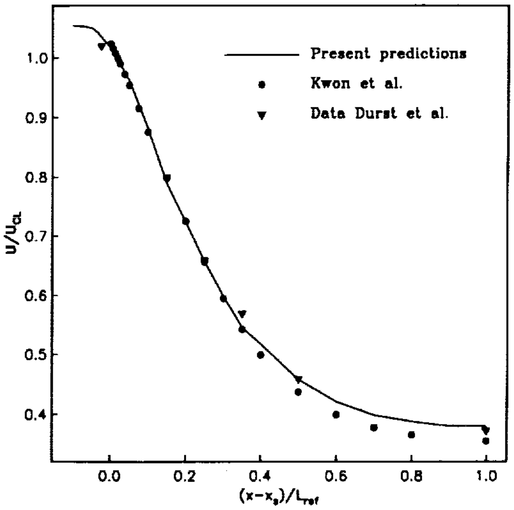
M=.1
M=.05
M=.005
M=.0005
Log(res)
0
-1
-2
-3
-4
-5
-6
0
1000
2000
Iteration Count

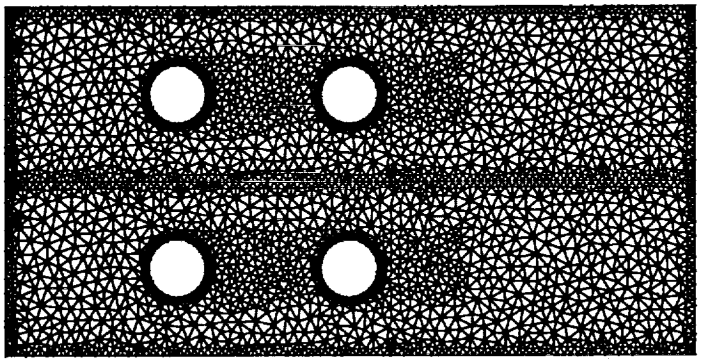


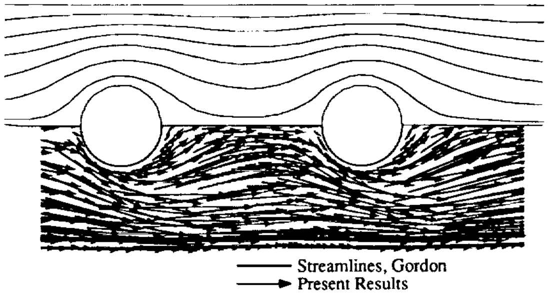
Bi-conjugate gradient with ILU
Generalized minimum residual with ILU
Generalized conjugate gradient residual with ILU
Log(res)
-1
-2
-3
-4
-5
-6
-7
4
8
12
16
20
Iteration Count

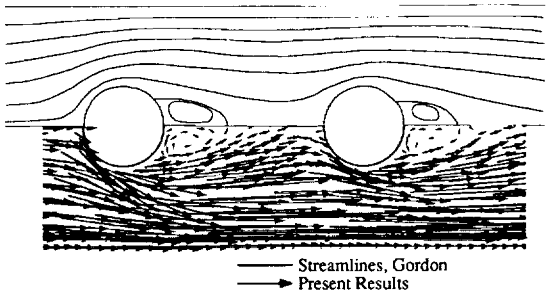


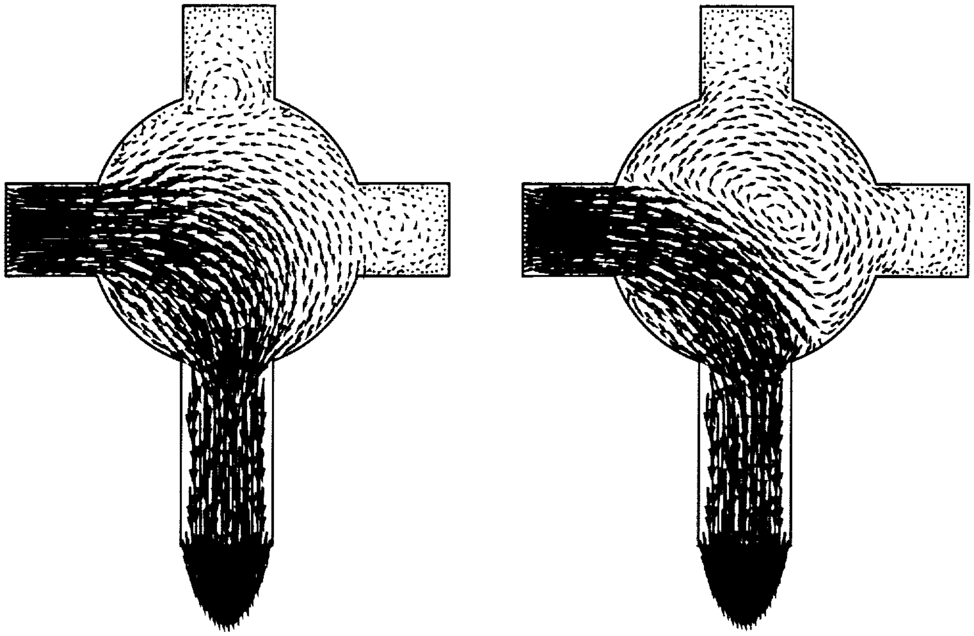

Present predictions
Data Durst et al.
x/h
■ -0.25
○ 1.50
▲ 2.50
○ 3.50
□ 5.00
▼ 10.00
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
-0.2
U/U$_{CL}$
-1.0
-0.5
0.0
0.5
1.0
$(y-y_{CL})/y_{CL}$


Present predictions
● Kwon et al.
▼ Data Durst et al.
1.0
0.9
0.8
0.7
0.6
0.5
0.4
U/U$_{CL}$
0.0
0.2
0.4
0.6
0.8
1.0
$(x-x_s)/L_{ref}$




Streamlines, Gordon
Present Results


Streamlines, Gordon
Present Results



Conclusions

- Grids can be generated about complex geometries
- Diagonal block Gauss-Seidel solver more robust than point diagonal Gauss-Seidel version of solver
- Coloring scheme allowed the vectorization of the implicit Gauss-Seidel solver
- Sparse iterative solver(SITRSOL) allowed a much larger time step than Gauss-Seidel(ran 2 to 2.5 times faster)
- Temporal preconditioning allowed the compressible code to run at very low Mach numbers

N94-22359

ROBUST UNSTRUCTURED GRID GENERATION WITH VGRID

SHAHYAR PIRZADEH
ViGYAN, INC.

Outline

- Objective and scope of present work
- Methodology
- Applications
- Concluding remarks

Scope of Present Work

- Objective:

  to develop a **robust**, **user oriented** unstructured grid-generation technique for **fast** generation of Euler/viscous grids around 2D/3D **complex** configurations

- Approach:
  - **Advancing-Front** method for generation of Euler grids (established technique)
  - **Advancing-Layers** method for generation of viscous grids (work in progress)

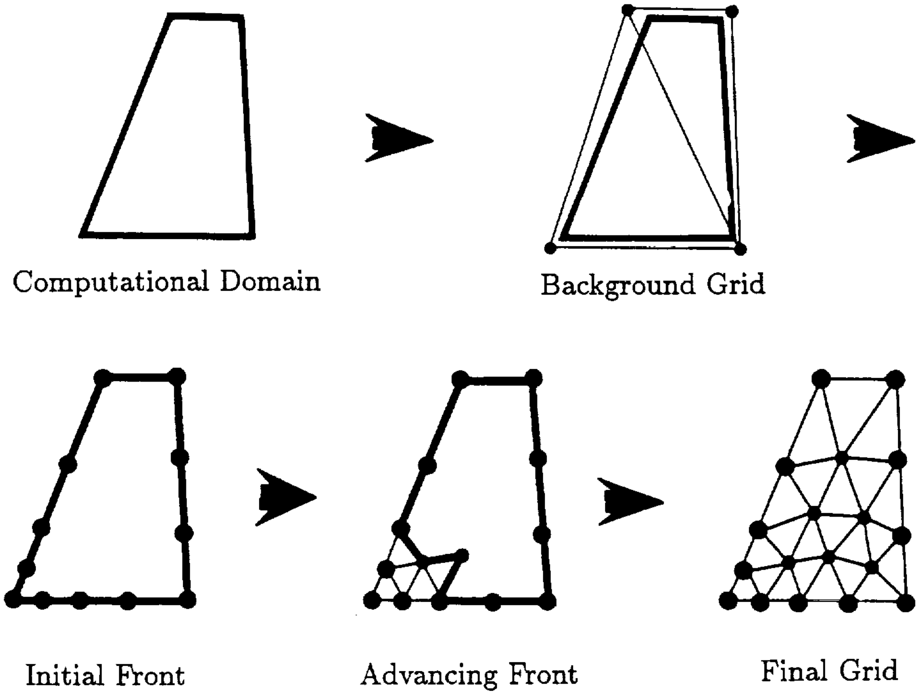
Advancing-Front Method

- Salient features:
  - grid quality
  - robustness
  - self-sufficiency for grid point distribution
  - established methodology (especially in 3D)

- Recent developments resulting in substantial enhancement of AFM :
  - structured background grids with source elements (AIAA Journal, Feb. 1993)
  - grid restart capability
  - local remeshing

  } grid post-processing (AIAA paper 92-0445)

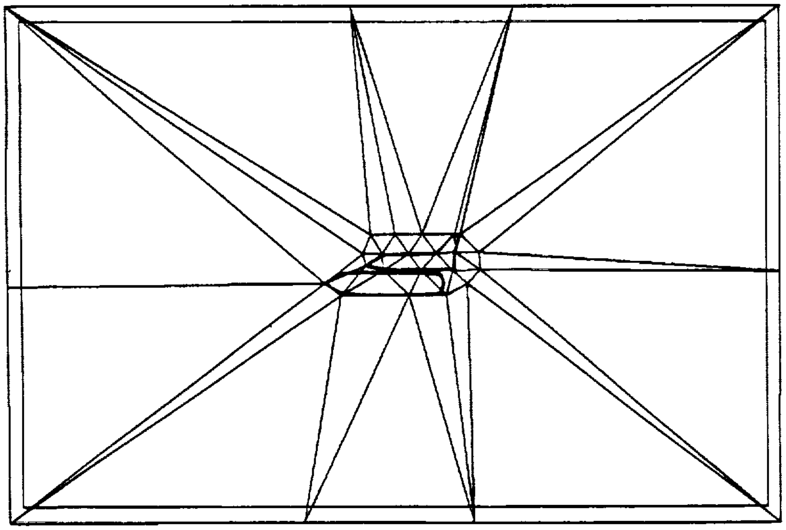
Advancing Front Method



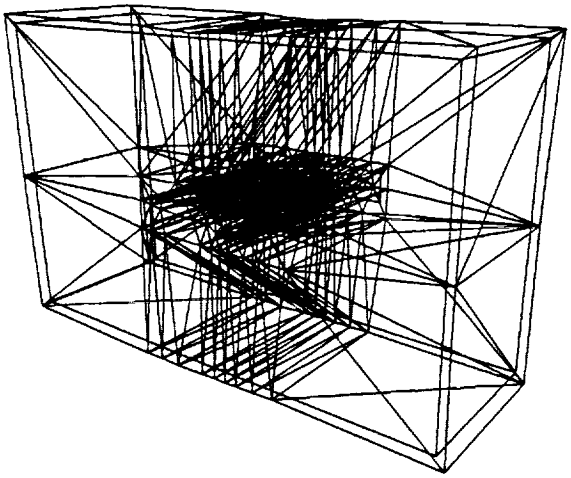
Background Grids

- A secondary mesh containing grid characteristic information
  - need not conform to the domain boundaries
  - integral to the AFM
- Background grids should
  - be simple to construct
  - provide smooth and controlled variation of grid spacings in the field
  - be flexible to modifications

Unstructured Background Grids



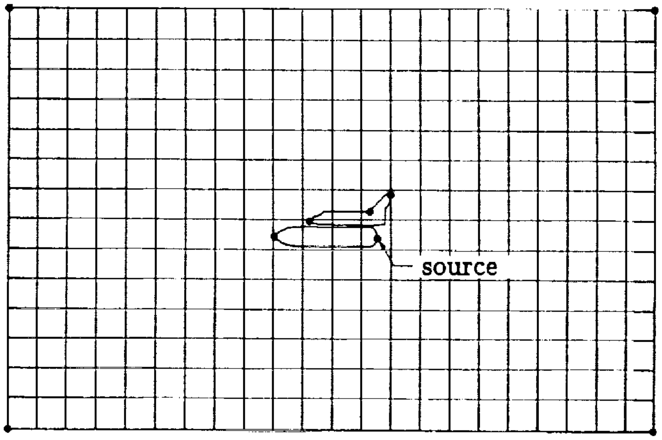
2-D



3-D

Structured Background Grids

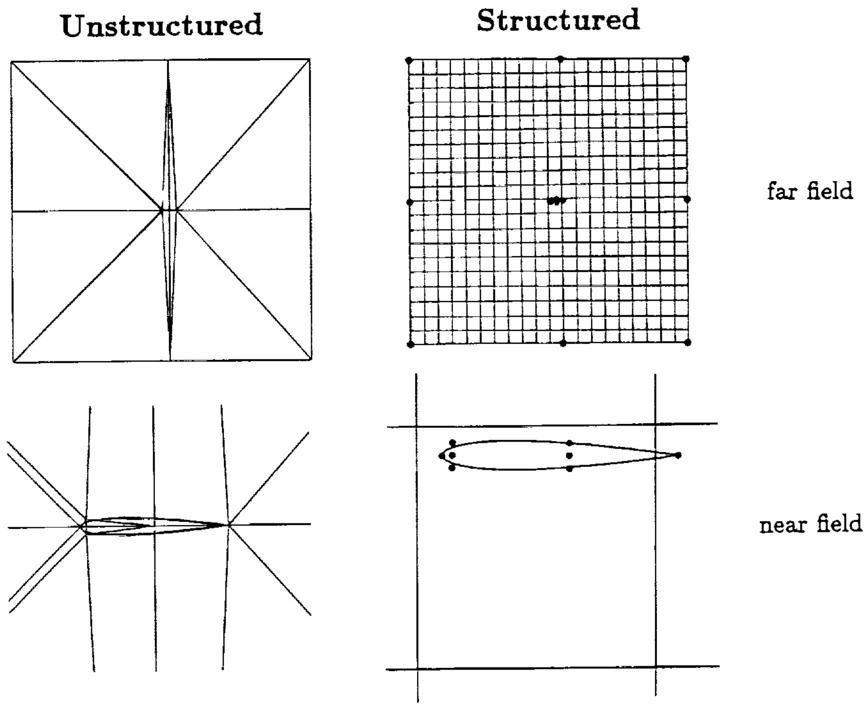
- Simple uniform Cartesian grids; easy to construct
- Source elements with prescribed spacing parameters: nodal and linear elements
- Provides smooth grid distribution, flexible control, and ease of grid modification



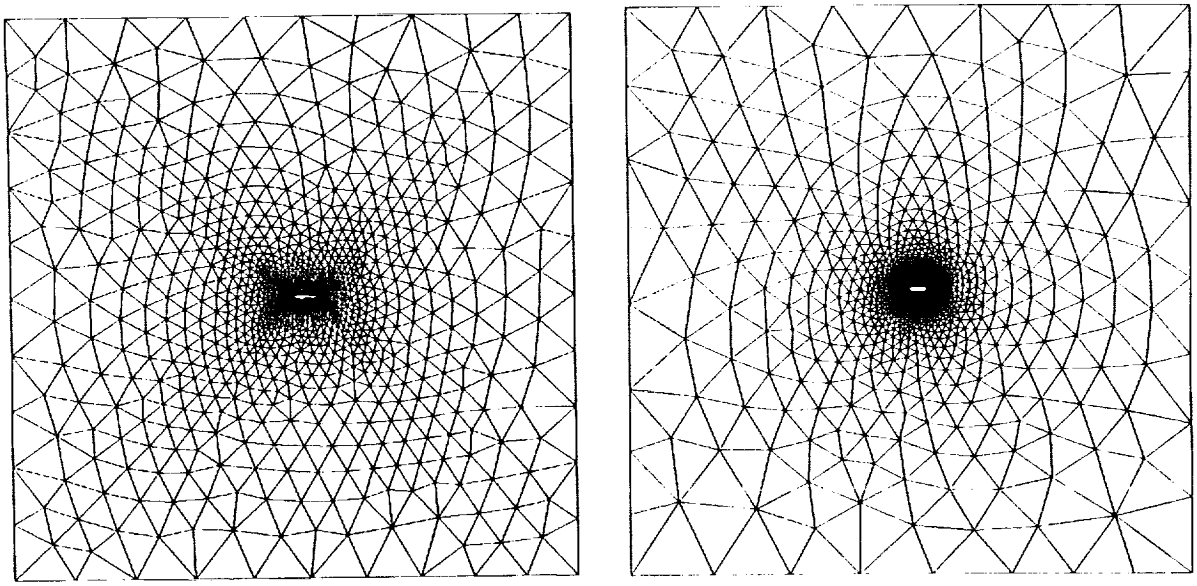
Distribution of Spacing Parameters

- Determined by a process similar to diffusion of 'heat' from discrete heat sources in a conducting medium
- Modeled by solving a Poisson equation, $\nabla^2 S = G$
- Resulting discretized algebraic equations solved with an iterative method
- The solution provides 'pseudo-isotherms' varying smoothly from high- to low-potential regions

Background Grid for a NACA 0012 Airfoil



Unstructured Grid around a NACA 0012 Airfoil

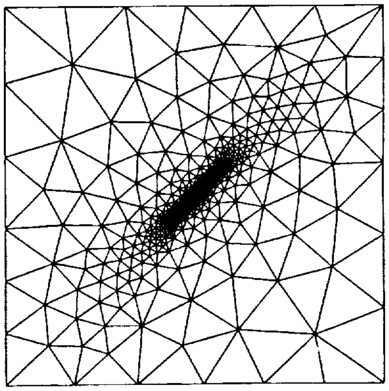


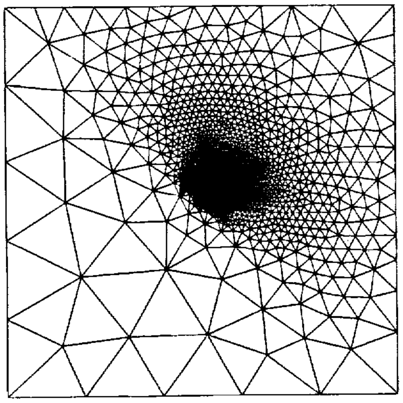
using unstructured background grid

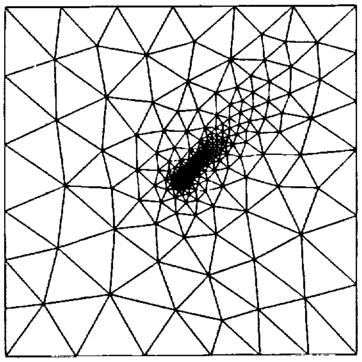
using structured background grid

Directional Control of Source Intensity

(Nodal Elements)

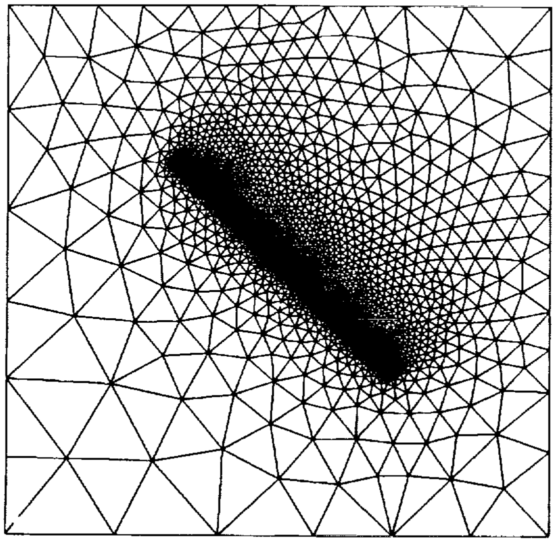


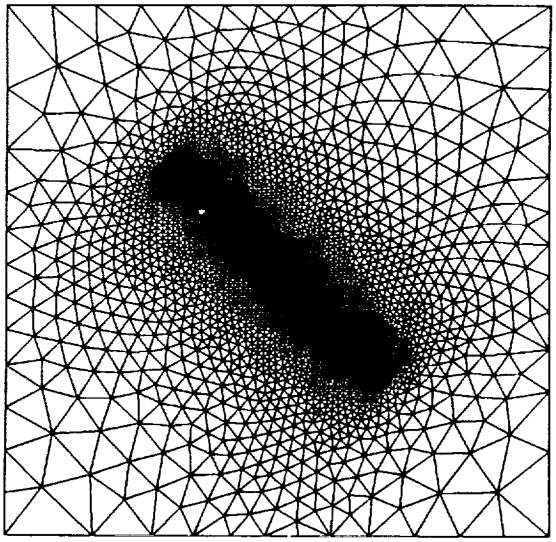




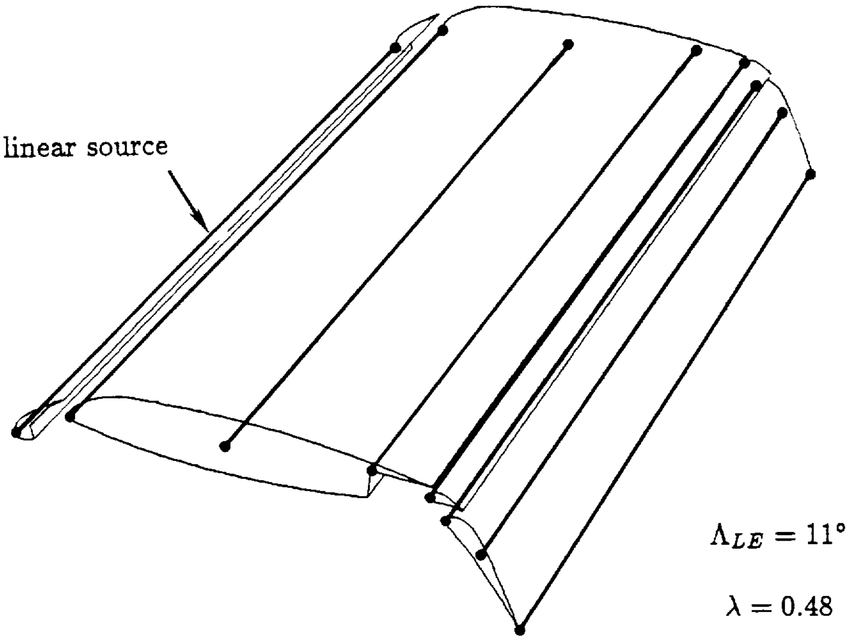
Directional Control of Source Intensity

(Linear Elements)



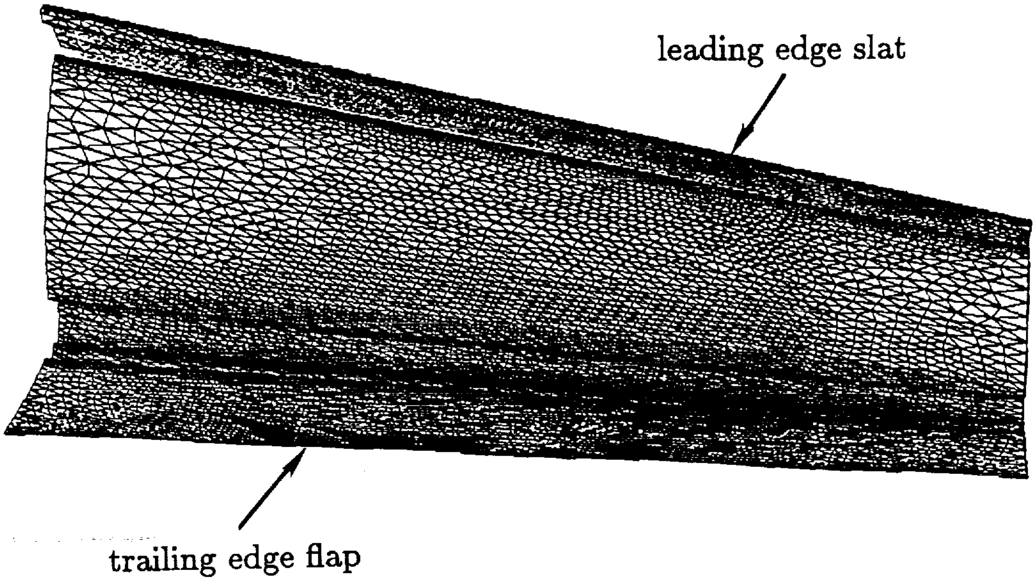


Source Elements on a Generic Multi-Element Wing

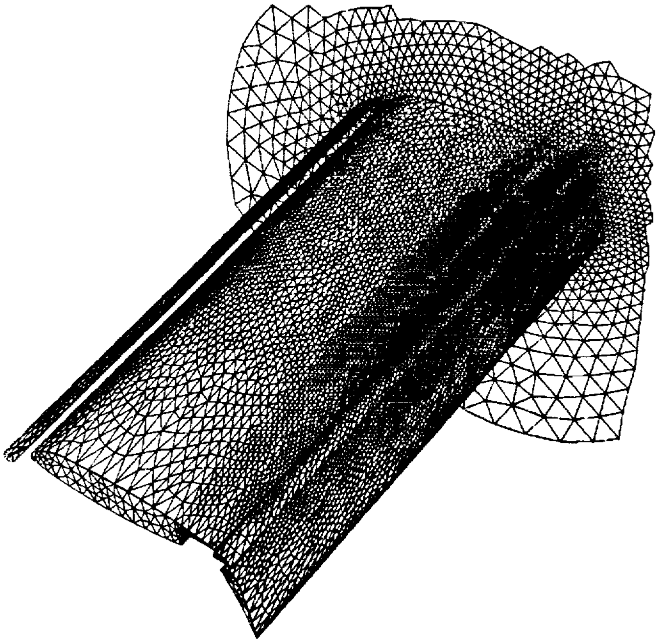


Surface Triangulation on a Generic Multi-Element Wing

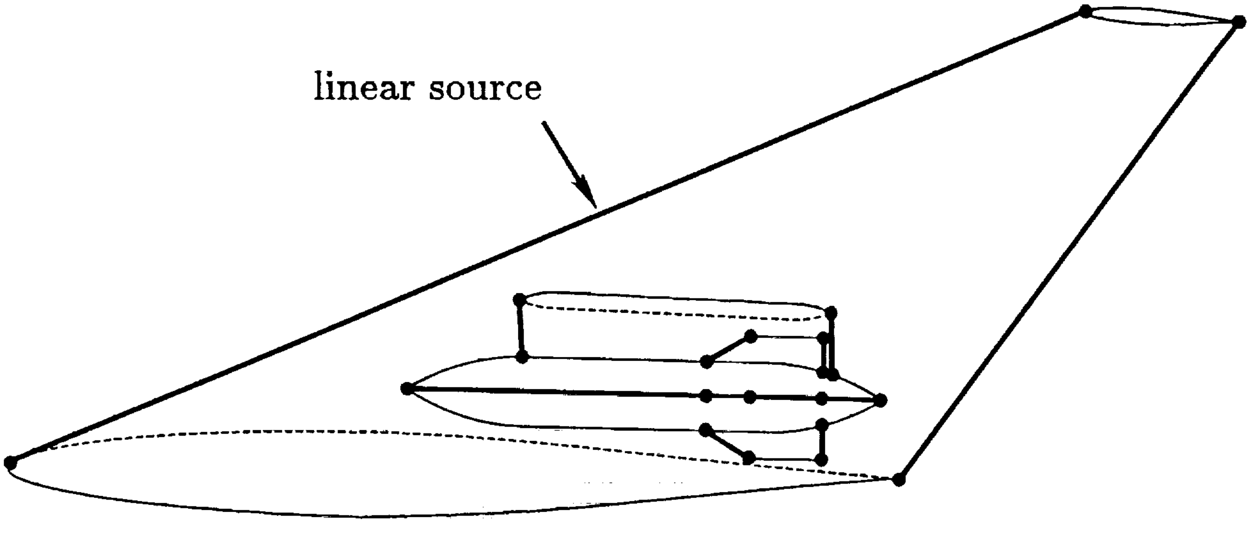
(wing lower surface)



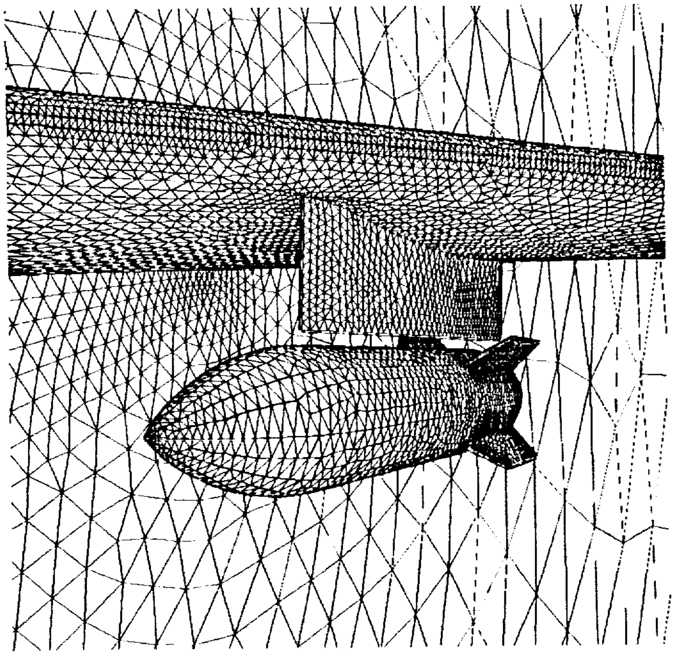
Surface Triangulation on a Generic Multi-Element Wing



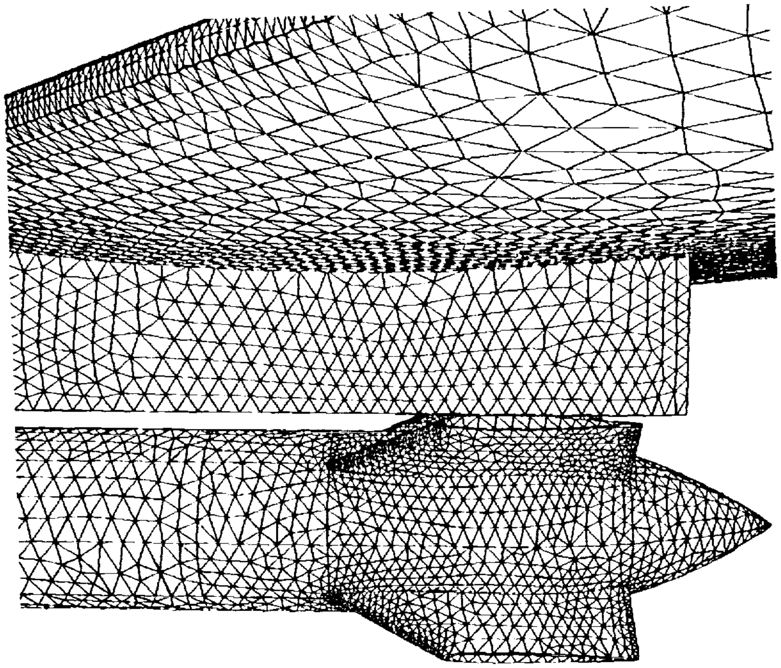
A Wing/Pylon/Store Configuration



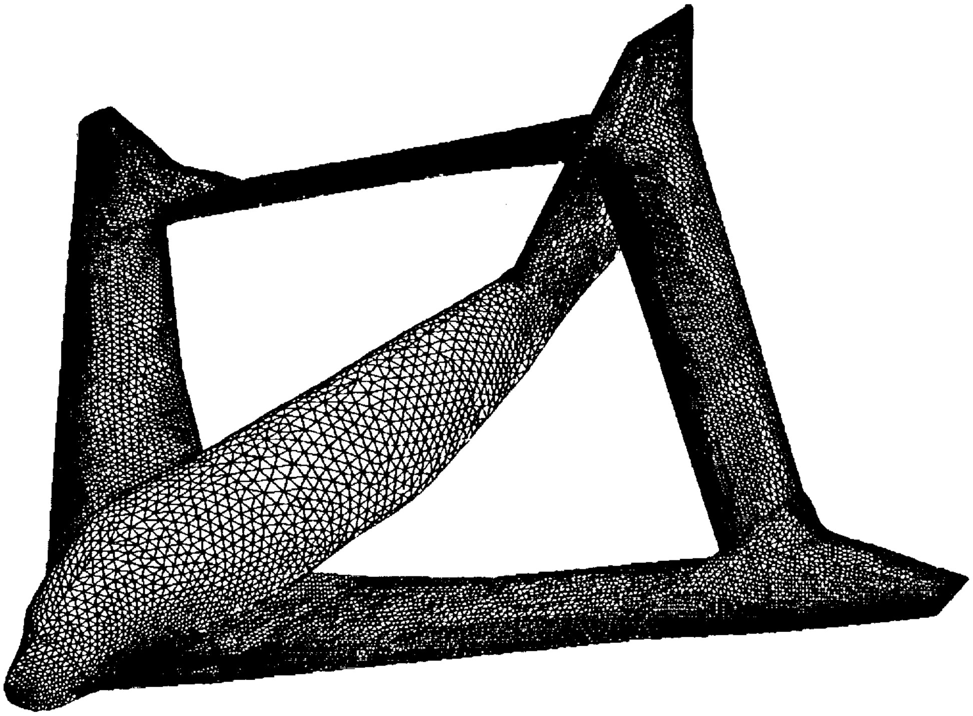
Details of Surface Grid on a
Wing/Pylon/Store Configuration



Details of Surface Grid on a
Wing/Pylon/Store Configuration



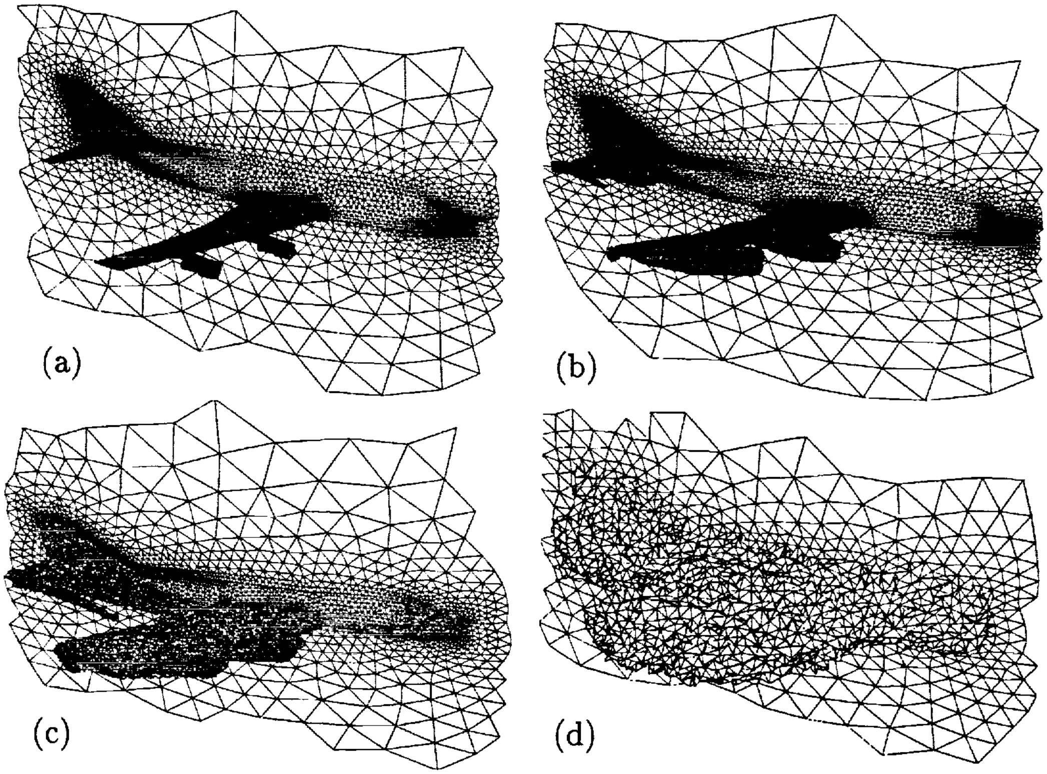
A Boeing Joined-Wing Aircraft Configuration



Grid Restarting

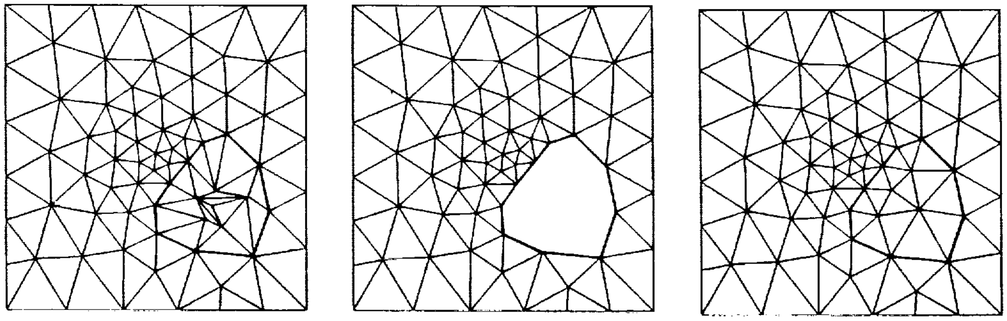
- Grid generated in a marching fashion in AFM
  - o only information on the current front needed for further advancement
  - o process may be stopped and restarted without carrying previously generated grid
- Procedure based on a recurrent local/global renumbering resulting in:
  - o substantial reduction in memory requirement
  - o capability of generating large grids on small machines
  - o substantial increase in productivity of the method

Partial Restarted Grids Around a B747 Configuration



Local Remeshing

- Irregularity of unstructured grids ⟹ arbitrary cell groupings
- A cell grouping, being independent of surrounding mesh, may be
  - removed, creating pockets and new fronts in the grid
  - remeshed with no effect on rest of the grid
- Local remeshing and restart capability have resulted in a useful 3D grid post-processing tool ⟹ program **Postgrid**



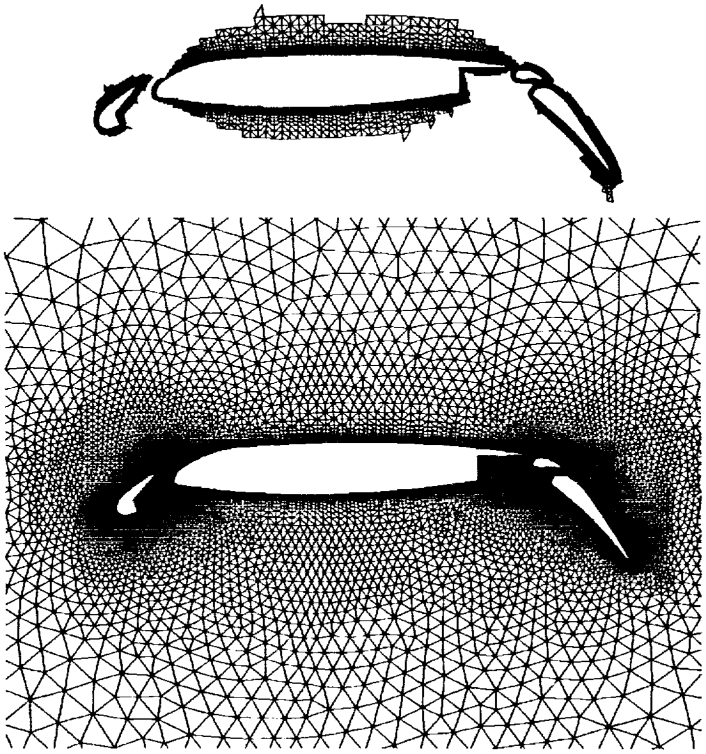
Unstructured Viscous Grid Generation

- Problem still unresolved, especially in 3D
- Generation of highly stretched cells proven to be non-trivial
- Issues to be considered:
  - automation
  - self-sufficiency for grid point distribution
  - grid quality
  - flexibility and ease of grid control
  - capability of handling difficult regions such as sharp corners, singular points, wakes, gaps between close surfaces, etc. without users' interaction

Advancing Layers for Generation of Viscous Grids

- An extension of Advancing-Front method to generate highly stretched cells
  - grid advances in the field one layer at a time
  - benefits from generality and flexibility of AFM
  - method is automatic, fast, self-sufficient, and robust
  - provides smooth and structured-looking viscous grids
  - practically, no limit to the extent of cell aspect ratio
  - minimal user's input data (uses same surface mesh and B.G.)
  - resolves many of shortcomings of the semi-structured methods
- Has been shown in 2D with good results (NASA CR 191449, 1993)
- Work in progress in 3D

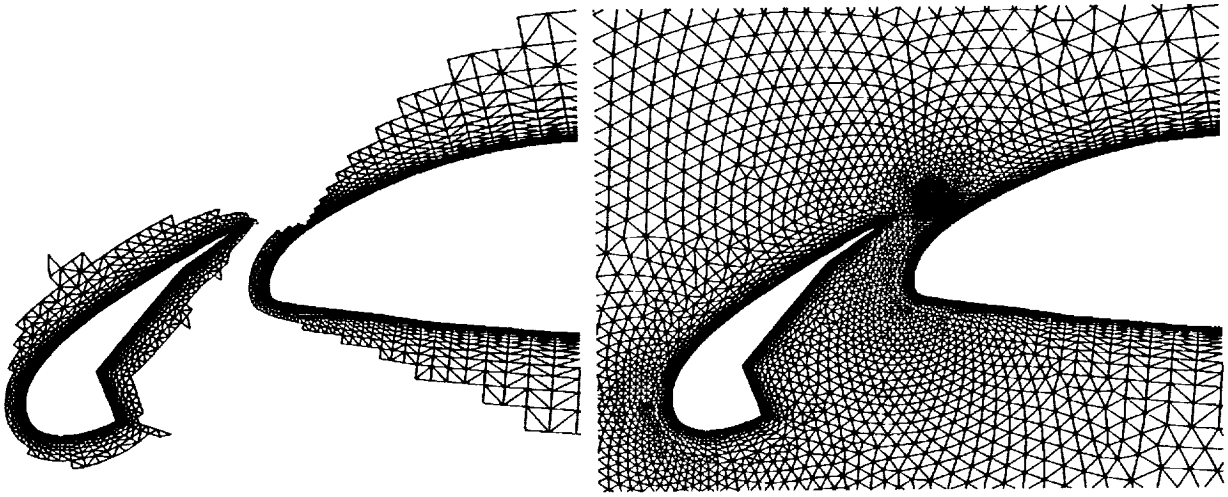
Viscous Grid around a Multi-element Airfoil



Partial grid

complete grid

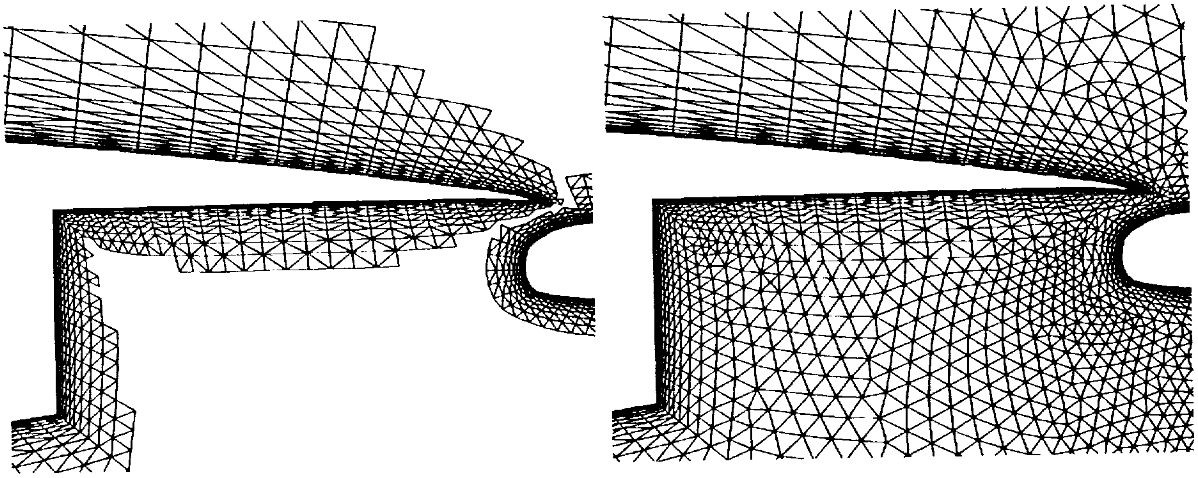
Viscous Grid around a Multi-element Airfoil
(by Advancing Layers / Advancing Front Methods)



Partial grid

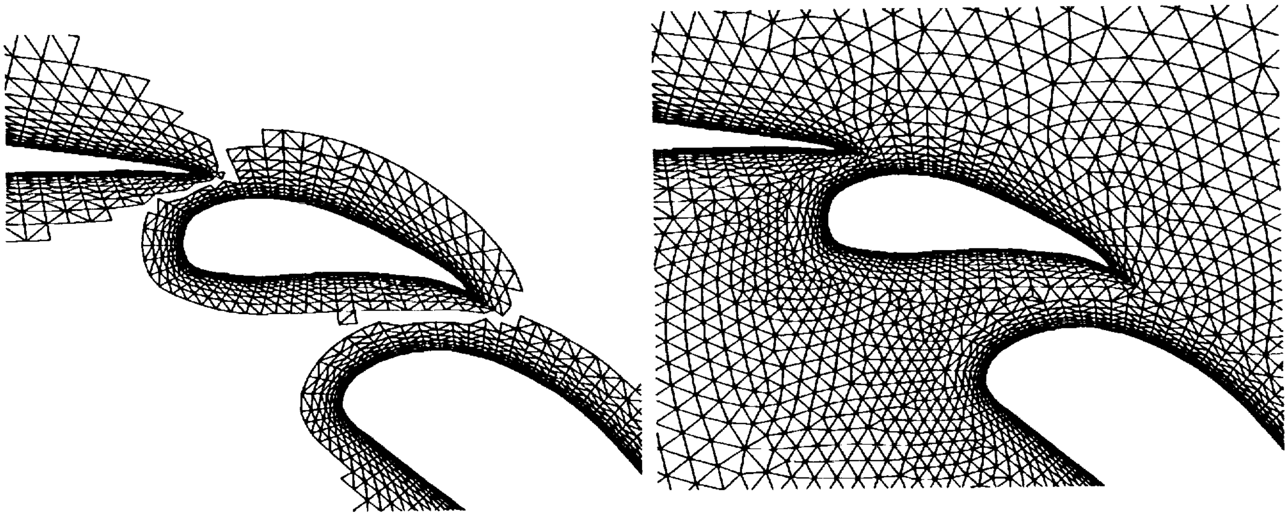
complete grid

Viscous Grid around a Multi-element Airfoil
(by Advancing Layers / Advancing Front Methods)



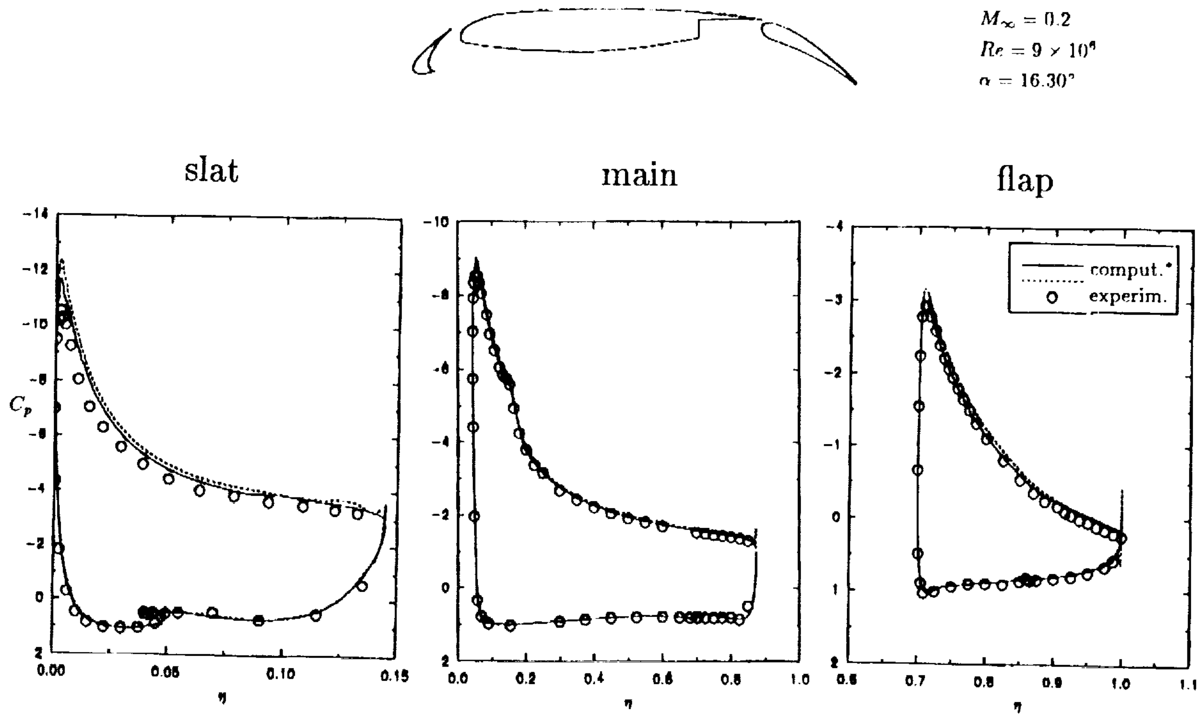
Partial grid complete grid

Viscous Grid around a Multi-element Airfoil
(by Advancing Layers / Advancing Front Methods)



Partial grid complete grid

Surface Pressure on a Douglas Multi-element Airfoil



Concluding Remarks

- Routine generation of Euler grids around complex configurations now possible with VGRID as currently used by many users from NASA and industry

- Continuous enhancement of the technique is performed in response to the users' requirements and feedback

- The new method of 'Advancing Layers' has produced good unstructured viscous grids in 2D (extension to 3D in progress)

- Plan: a single robust code for generation of both Euler and viscous unstructured tetrahedral grids

N94-22360

THREE-DIMENSIONAL UNSTRUCTURED GRID METHOD APPLIED TO TURBOMACHINERY

OH JOON KWON
SVERDRUP TECHNOLOGY, INC.

CHUNILL HAH
NASA LEWIS RESEARCH CENTER

OBJECTIVES

- To develop a three-dimensional flow solver based on unstructured tetrahedral meshes for turbomachinery flows.

- To validate the solver through comparisons with experimental data.

- To apply the solver for better understanding of the flow through turbomachinery geometries and design improvement.

APPROACH

- Existing external flow solver/grid generator *(USM3D/VGRID)* has been extensively modified for internal flows.

- Three–dimensional, finite-volume solver based on Roe's flux-difference splitting and explicit Runge-Kutta time stepping.

- Three–dimensional unstructured tetrahedral mesh generation using an advancing-front technique.

GOVERNING EQUATIONS

The governing equations are cast in body-fixed coordinate system which may rotate with an angular velocity Ω about the x-axis :

$$\frac{\partial}{\partial t}\iiint_{\Omega} Q\,dV + \iint_{\partial\Omega} F(Q)\cdot\hat{n}\,dS = R$$

$$Q = \begin{Bmatrix} \rho \\ \rho u^* \\ \rho v^* \\ \rho w^* \\ e_\circ \end{Bmatrix}, F(Q)\cdot\hat{n} = \begin{Bmatrix} \rho\bar{u} \\ \rho u^*\bar{u} + p\hat{n}_x \\ \rho v^*\bar{u} + p\hat{n}_y \\ \rho w^*\bar{u} + p\hat{n}_z \\ e_\circ\bar{u} + pu_n \end{Bmatrix}, R = V\begin{Bmatrix} 0 \\ 0 \\ \Omega\rho w^* \\ -\Omega\rho v^* \\ 0 \end{Bmatrix}$$

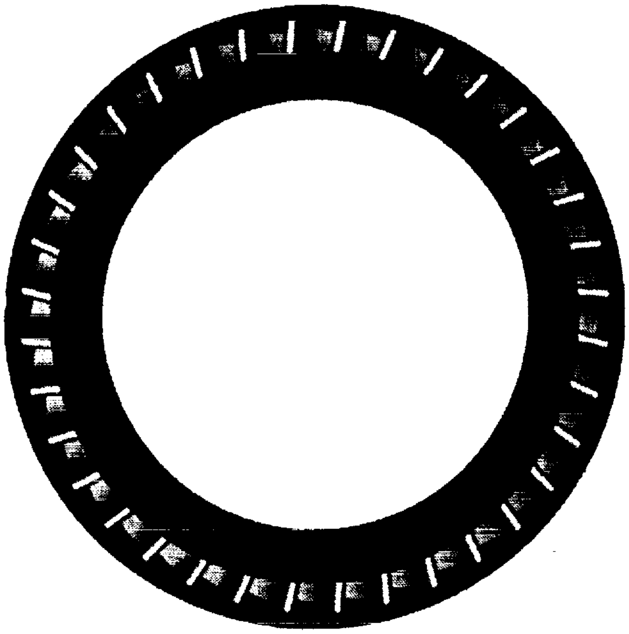
BOUNDARY CONDITIONS

- Flow tangency condition is imposed on solid surfaces.
- Periodic flow condition is imposed between the blades.
- At the inflow boundary, total pressure, total temperature, and the flow angle are specified.
- At the exit plane, the static pressure is prescribed.

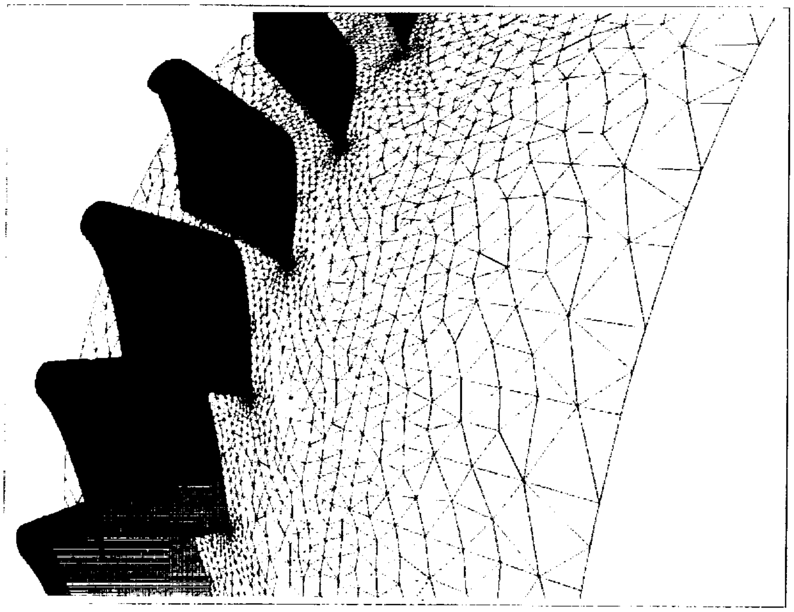
MESH GENERATION

VGRID has been modified to enforce grid periodicity of the surface mesh on the periodic boundaries.

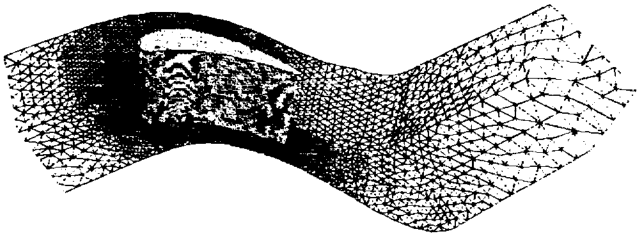
- The same surface patches are defined on the periodic boundaries from the definition of computational domain.
- The corresponding boundary lines on the periodic surfaces are divided into same segments.
- One periodic boundary surface is meshed and the surface triangles are replaced on the other surface with proper connectivity.



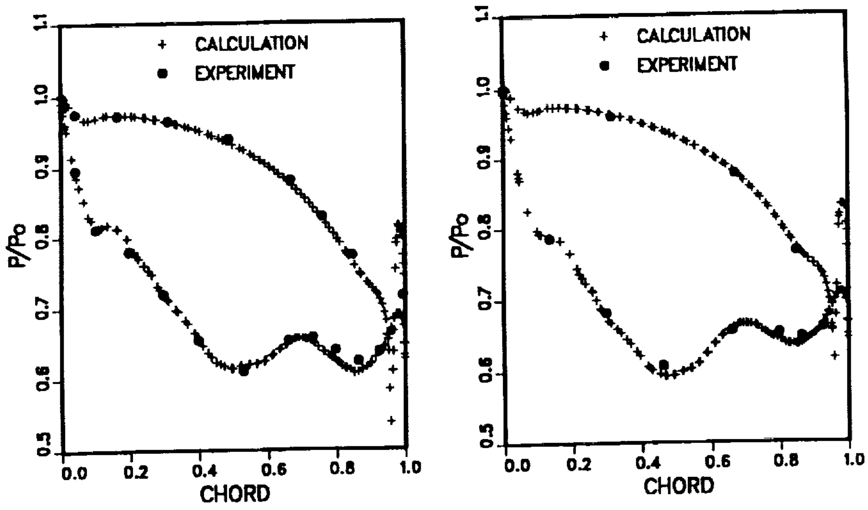
Turbine Stator Annular Cascade



Close–Up View of Blade and Hub
Surface Triangulation



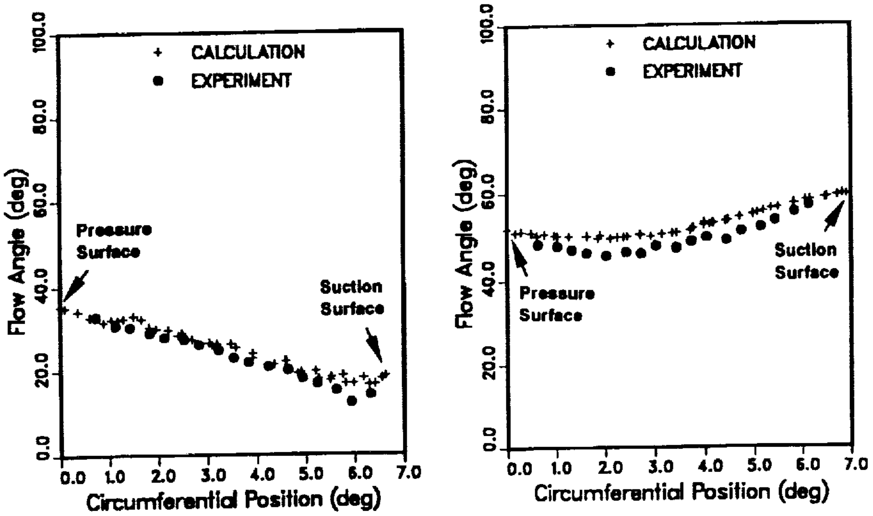
Surface Triangulation of Computational Domain



50.0% Span

86.7% Span

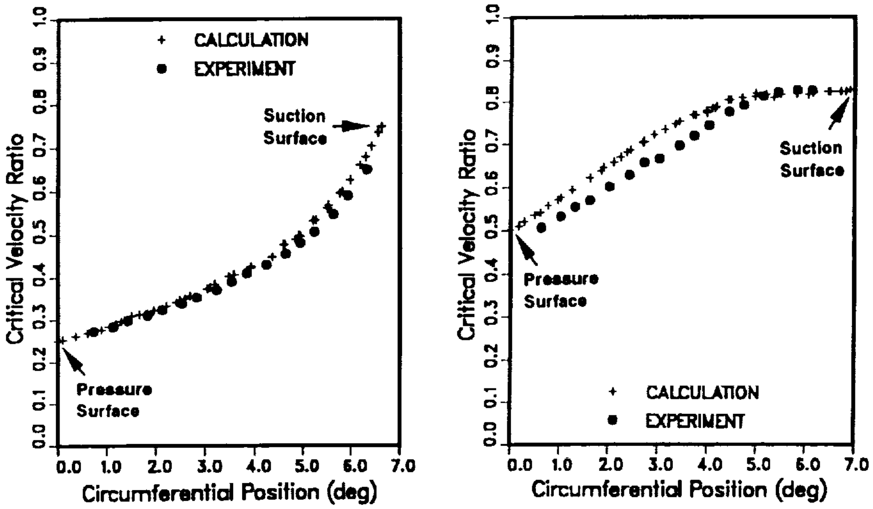
Static Pressure Distribution on the Blade



30% Axial Chord

70% Axial Chord

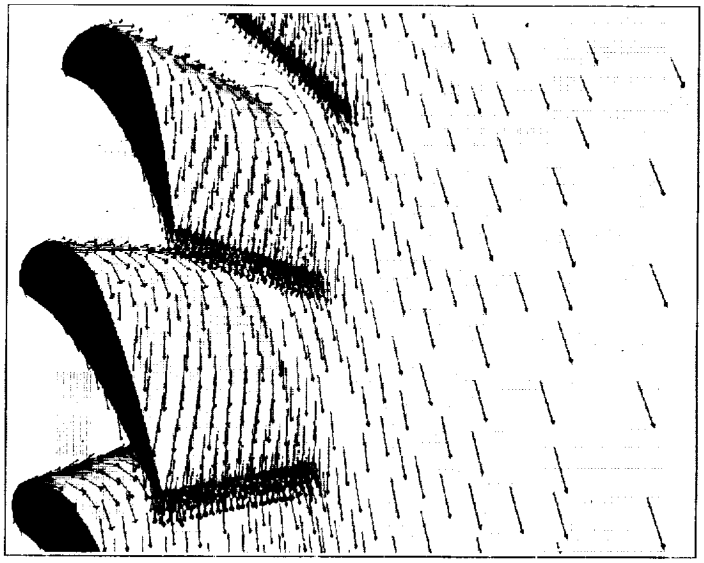
Flow Angle at 50% Span



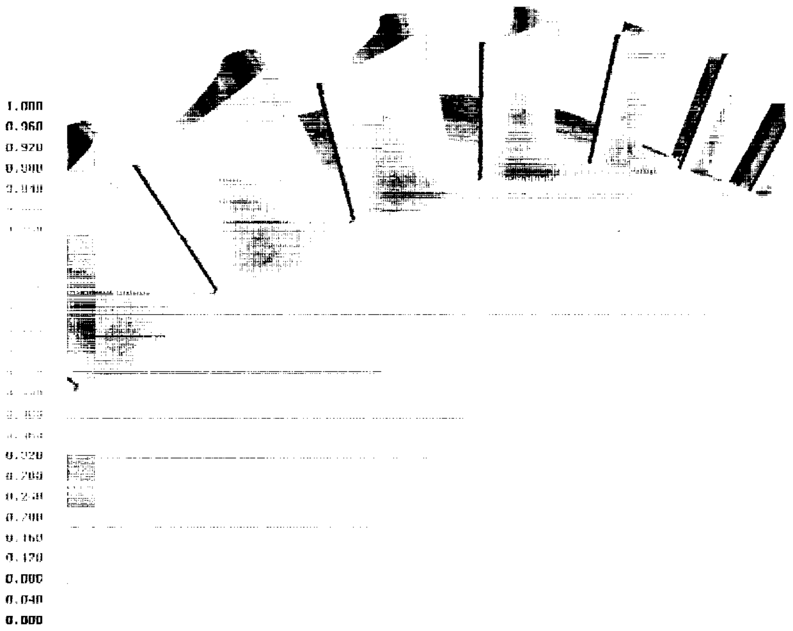
30% Axial Chord

70% Axial Chord

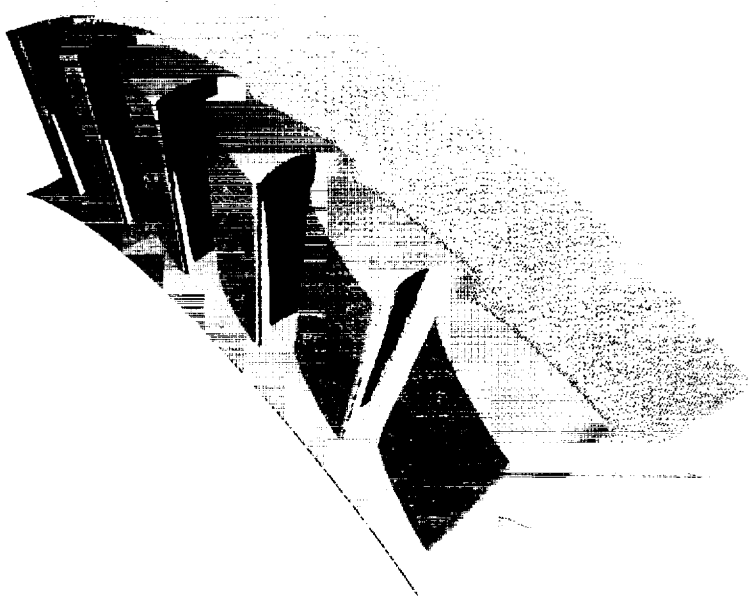
Critical Velocity Ratio at 50% Span



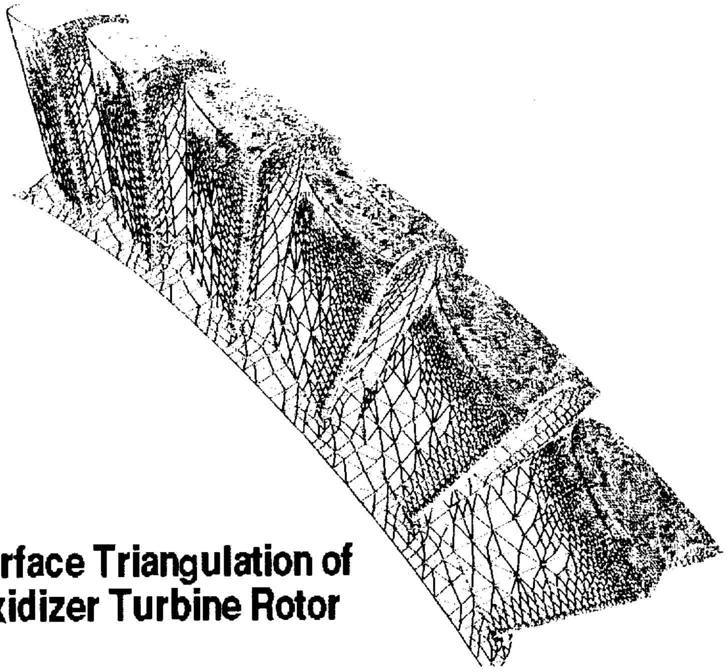
Velocity Vectors on the Blade and Hub Surfaces



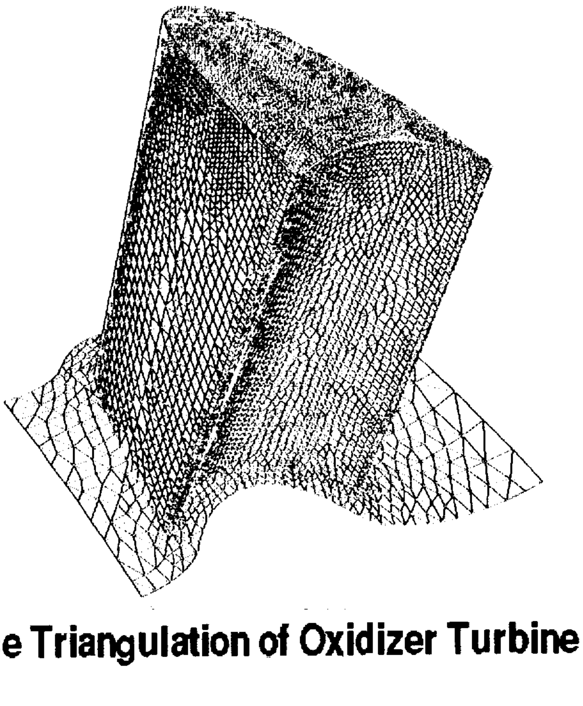
Mach Number Contour on the Blade and Hub Surfaces



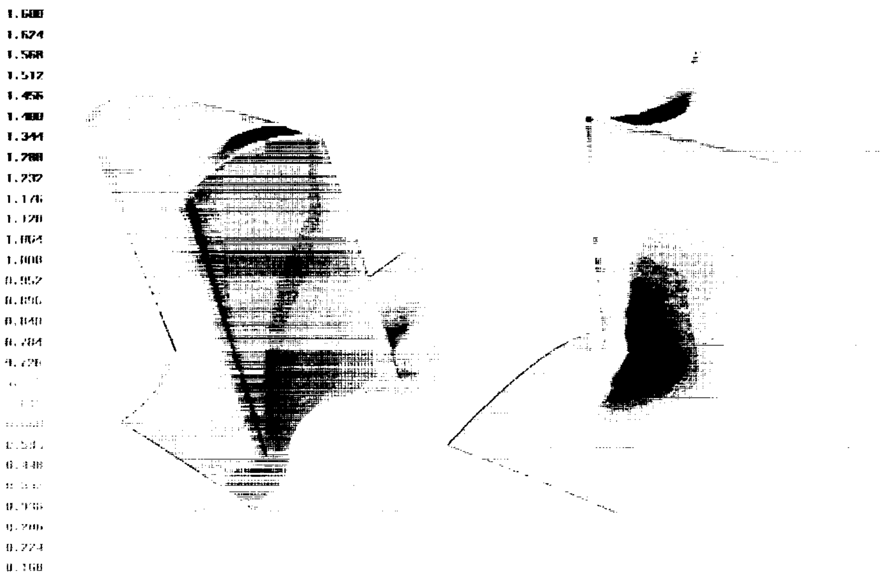
Advanced Gas Generator Oxidizer Turbine Rotor



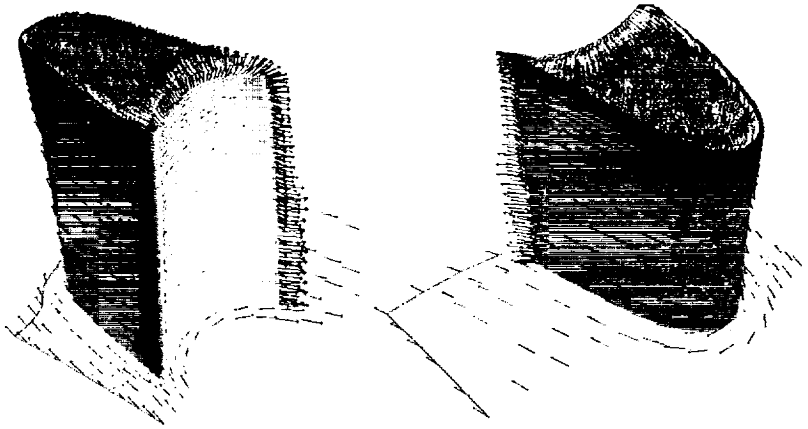
Surface Triangulation of Oxidizer Turbine Rotor



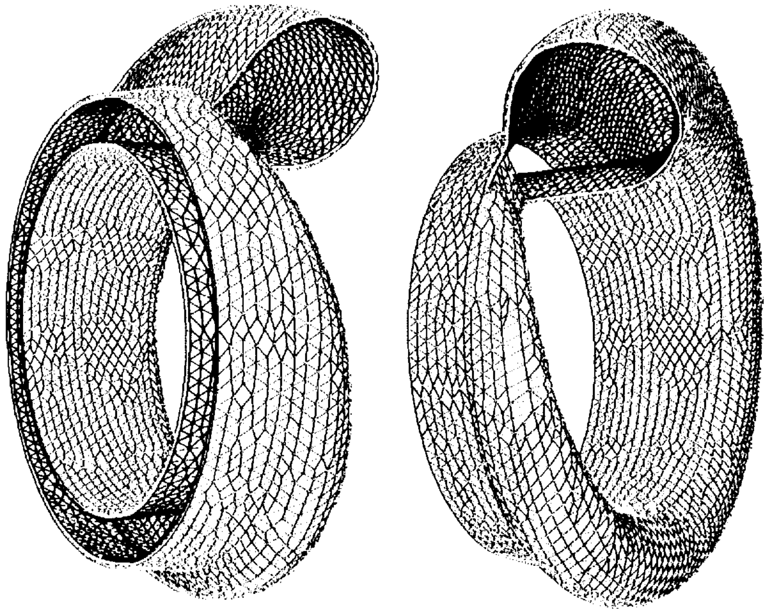
Surface Triangulation of Oxidizer Turbine Rotor



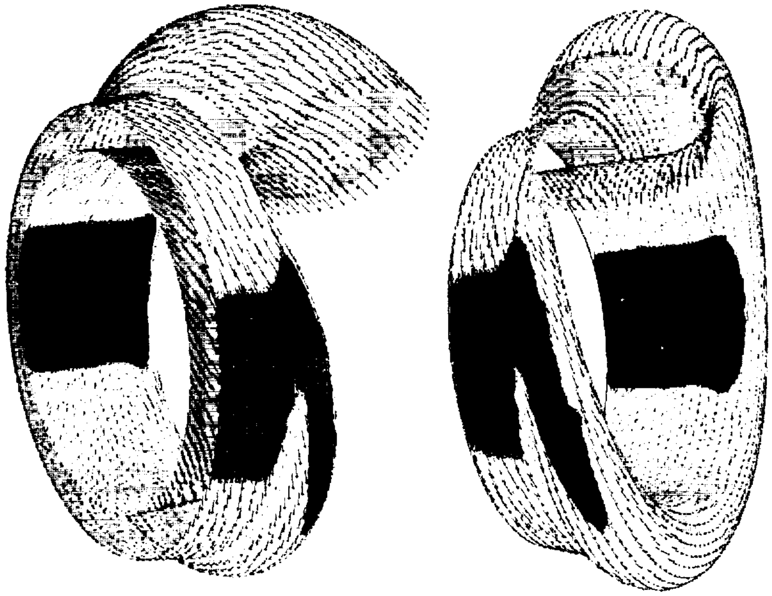
Mach Number Contour on Oxidizer Turbine Rotor



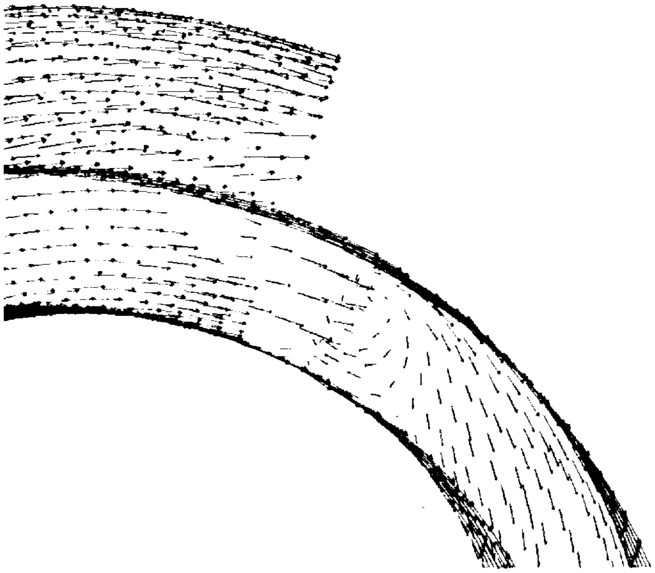
Velocity Vectors on Oxidizer Turbine Rotor



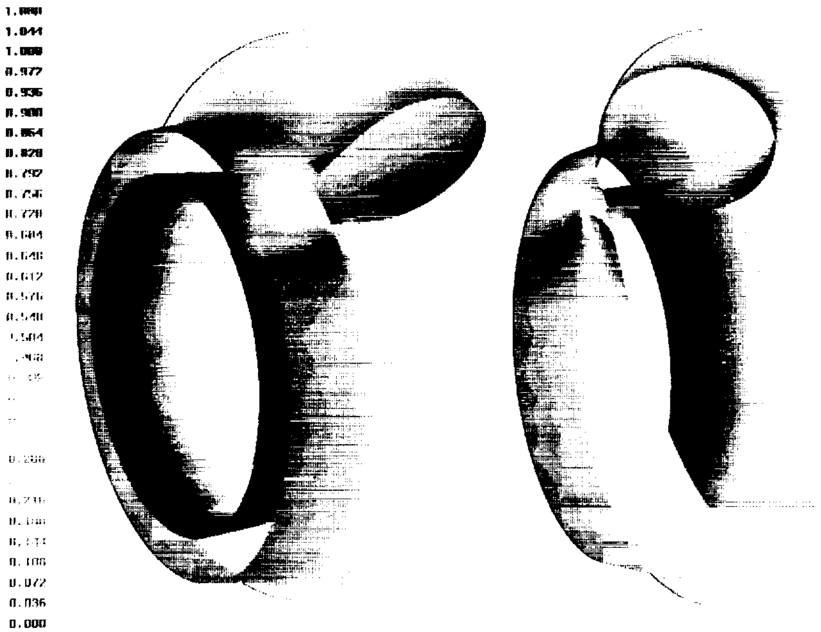
Surface Triangulation of Oxidizer Turbine Volute



Velocity Vectors on Oxidizer Turbine Volute



Velocity Vectors on Oxidizer Turbine Volute



Mach Number Contour on Oxidizer Turbine Volute

ORIGINAL PAGE IS
OF POOR QUALITY

CONCLUDING REMARKS

- A three-dimensional unstructured grid Euler solver has been developed for turbomachinery flows based on an existing external flow solver *USM3D*.

- Good correlation with experimental data has been observed both on the blade surface and in the flow passage between the blades.

- Applications are successfully made to calculate flows through various turbomachinery geometries.

FUTURE WORKS

- Solution-adaptive grid generation.

- Add viscous terms for the solver.

- Add adequate turbulence model.

DEVELOPMENT OF A GRIDLESS CFD METHOD

JOHN T. BATINA
NASA LANGLEY RESEARCH CENTER

PRESENTATION OBJECTIVE

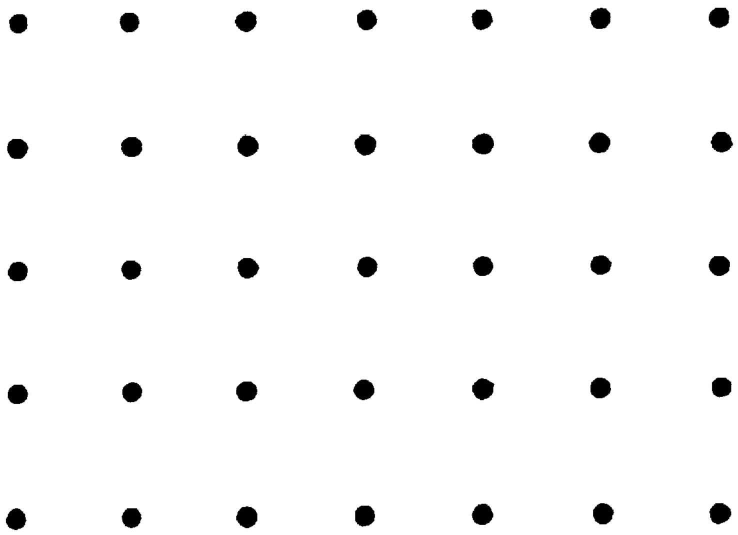
- Leave you with some thoughts or ideas on an alternative approach to discretizing fluid flow problems (namely the so-called gridless approach)

- Ask you today to:
  - Expand your thinking
  - Be unconventional

- Why? Because if you expand the possibilities for generating grids or developing solution algorithms you might actually discover techniques that are superior to conventional procedures!

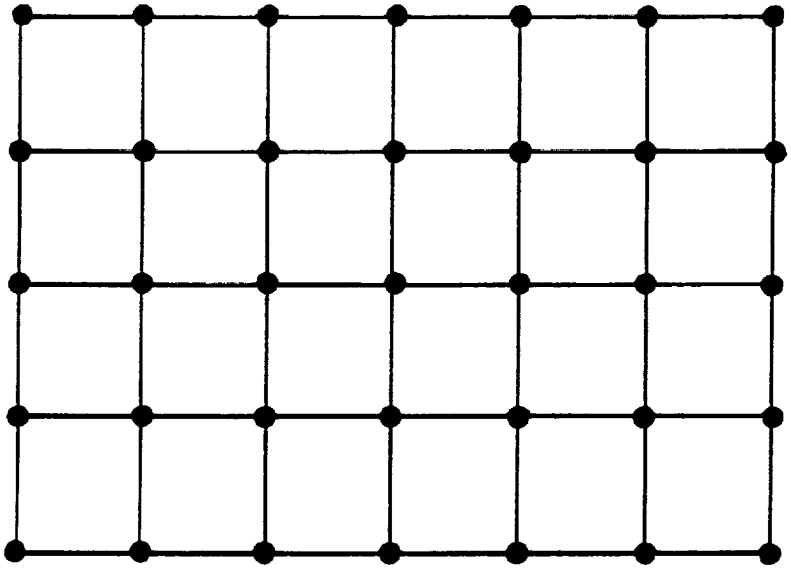
CONSIDER A SET OF POINTS IN A TWO-DIMENSIONAL DOMAIN

- How do you connect the points?



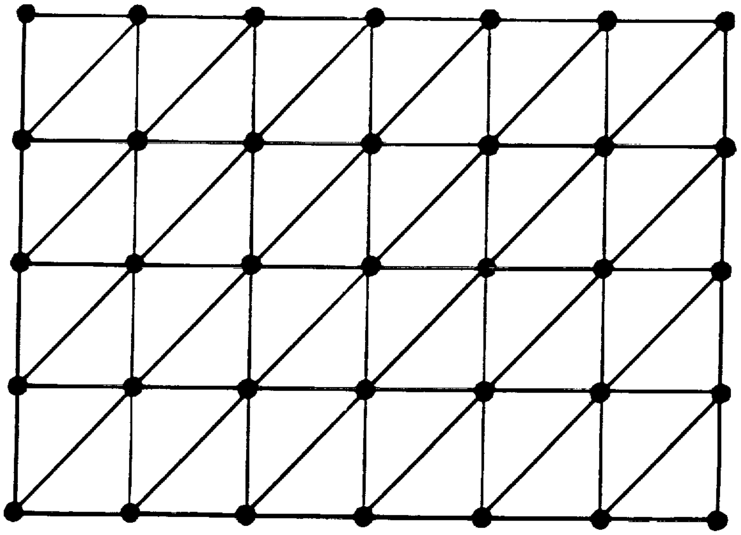
STRUCTURED GRID

- Should the points be connected in a structured fashion?



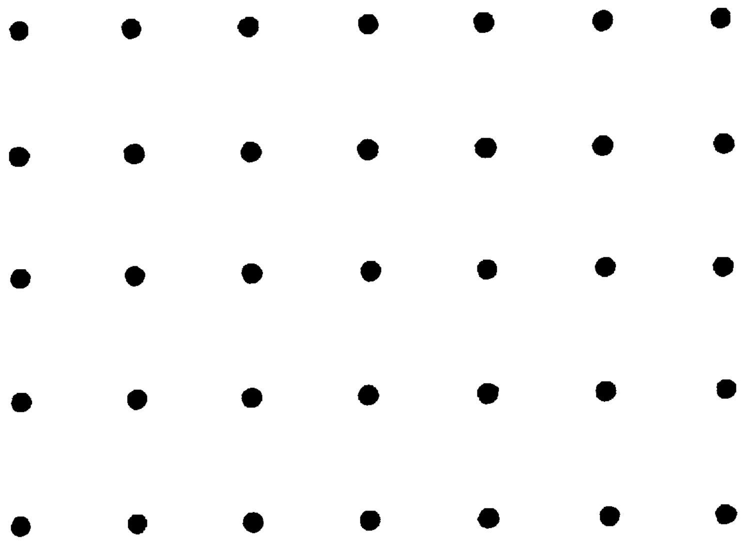
UNSTRUCTURED GRID

- Or should they be connected as an unstructured grid of triangles?



FIELD OF POINTS

- Maybe the points didn't need to be connected in the first place!



MOTIVATION FOR ALTERNATIVE APPROACH

- Tetrahedral meshes have an excessively large number of cells than structured grids
- These meshes, while reasonably adequate in the streamwise direction, tend to be much finer in the spanwise direction than is necessary for accurate flow computation
- Furthermore, for viscous applications, the additional requirement that the mesh be fine near the body, exacerbates the inefficiency
- The basic problem is that the tetrahedron is an inefficient geometrical shape

INTRODUCTION OF GRIDLESS APPROACH

- To alleviate the problem, some researchers have put structure back into the mesh in one coordinate direction
- This helps, but rather than take a step back toward grid structure, can we take a step forward and develop algorithms that do not require that the points be connected at all?
- This type of approach, referred to as "gridless," uses only clouds of points and does not require that the points be connected to form a grid as is necessary in conventional CFD algorithms
- The governing equations are solved directly, by performing local least-squares curve fits in each cloud of points, and then analytically differentiating the resulting curve fits to approximate the derivatives

SPATIAL DISCRETIZATION – DERIVATIVES

- Fluxes assumed to vary locally as

$$f(x,y,z) = a_0 + a_1x + a_2y + a_3z$$

- a_0, a_1, a_2, and a_3 determined from a least-squares curve fit resulting in

$$\begin{bmatrix} n & \Sigma x_i & \Sigma y_i & \Sigma z_i \\ \Sigma x_i & \Sigma x_i^2 & \Sigma x_i y_i & \Sigma x_i z_i \\ \Sigma y_i & \Sigma x_i y_i & \Sigma y_i^2 & \Sigma y_i z_i \\ \Sigma z_i & \Sigma x_i z_i & \Sigma y_i z_i & \Sigma z_i^2 \end{bmatrix} \begin{Bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{Bmatrix} = \begin{Bmatrix} \Sigma f_i \\ \Sigma x_i f_i \\ \Sigma y_i f_i \\ \Sigma z_i f_i \end{Bmatrix}$$

where n is the number of points in the cloud and the summations are taken over the n points

- The spatial derivatives are now known since

$$\frac{\partial f}{\partial x} = a_1 \qquad \frac{\partial f}{\partial y} = a_2 \qquad \frac{\partial f}{\partial z} = a_3$$

SOLUTION BY QR – DECOMPOSITION

- Least-squares equations are of the form

$$\left(A^T A\right) a = A^T f$$

but $\left(A^T A\right)$ may be ill conditioned

- Instead the equations

$$Aa = f$$

are solved using a decomposition where $A = QR$ such that $Q^T Q = I$ and R is a square upper triangular matrix

- Solution given by

$$Ra = Q^T f$$

SPATIAL DISCRETIZATION – ARTIFICIAL DISSIPATION

- Artificial dissipation is added to the solution procedure since the method is conceptually analogous to a central-difference type approach

- Harmonic and biharmonic terms are added to the governing equations defined by

$$D = \nabla\left(\epsilon^{(2)}\lambda\right)\nabla Q - \nabla^2\left(\epsilon^{(4)}\lambda\right)\nabla^2 Q$$

where λ is the local maximum eigenvalue and $\epsilon^{(2)}$ and $\epsilon^{(4)}$ are local dissipation coefficients

- For the Navier-Stokes equations, an anisotropic model is used in part to account for the close spacing of points normal to the surface relative to the tangential distribution

BOUNDARY CONDITIONS

- Ghost points are used inside or outside of boundaries to impose the boundary conditions
- Along solid surfaces
  - velocity components determined by slip (Euler) or no-slip (Navier-Stokes) condition
  - pressure and density determined by extrapolation
- In the farfield
  - inviscid flow variables determined by a characteristic analysis based on Riemann invariants
  - viscous flow variables determined by extrapolation

TEMPORAL DISCRETIZATION – TIME INTEGRATION

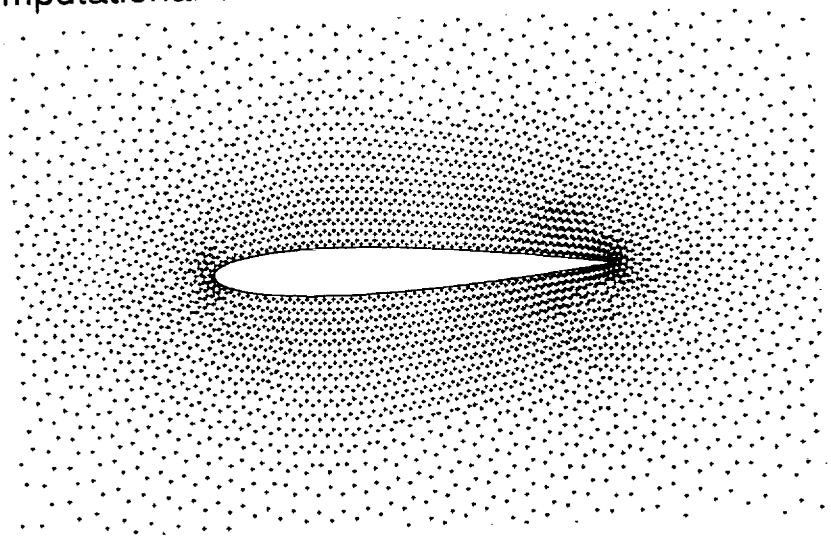
- Governing flow equations are integrated numerically in time using an explicit Runge-Kutta scheme
  - To solve the Euler equations, a four-stage scheme is used with the artificial dissipation evaluated only during the first stage
  - To solve the Navier-Stokes equations, a five-stage scheme is used with the artificial dissipation evaluated during the odd stages

OVERVIEW OF EULER RESULTS

- NACA 0012 airfoil
  - $M_\infty = 0.8$ and $\alpha = 0^\circ$
  - $M_\infty = 0.85$ and $\alpha = 1^\circ$
  - $M_\infty = 0.8$ and $\alpha = 1.25^\circ$
  - $M_\infty = 1.2$ and $\alpha = 7^\circ$
- ONERA M6 wing at $M_\infty = 0.84^\circ$ and $\alpha = 3.06^\circ$

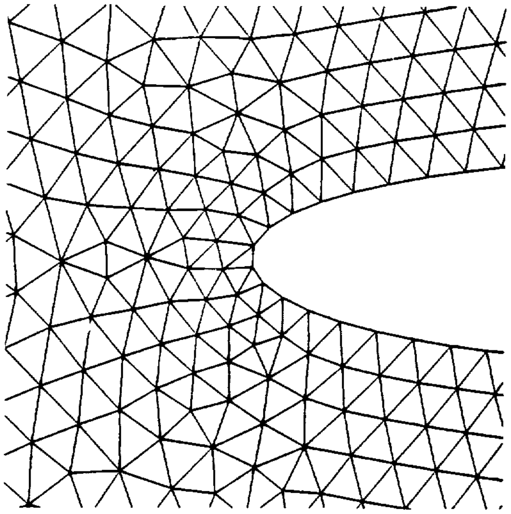
FIELD OF POINTS ABOUT NACA 0012 AIRFOIL

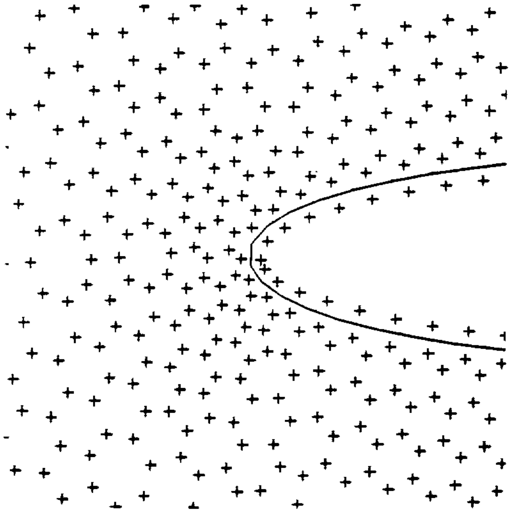
- Locations of points determined using the cell centers of an unstructured grid for convenience
- Computational domain has a total of 6500 points



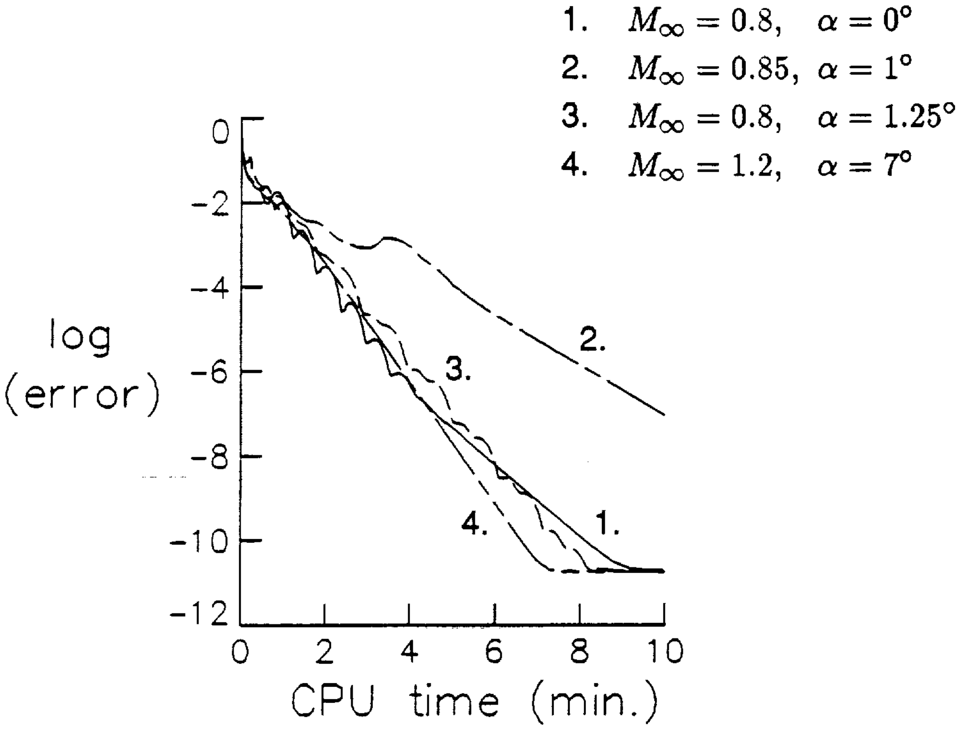
NEAR NOSE OF NACA 0012 AIRFOIL

- Unstructured mesh of triangles
- Corresponding field of points including ghost points

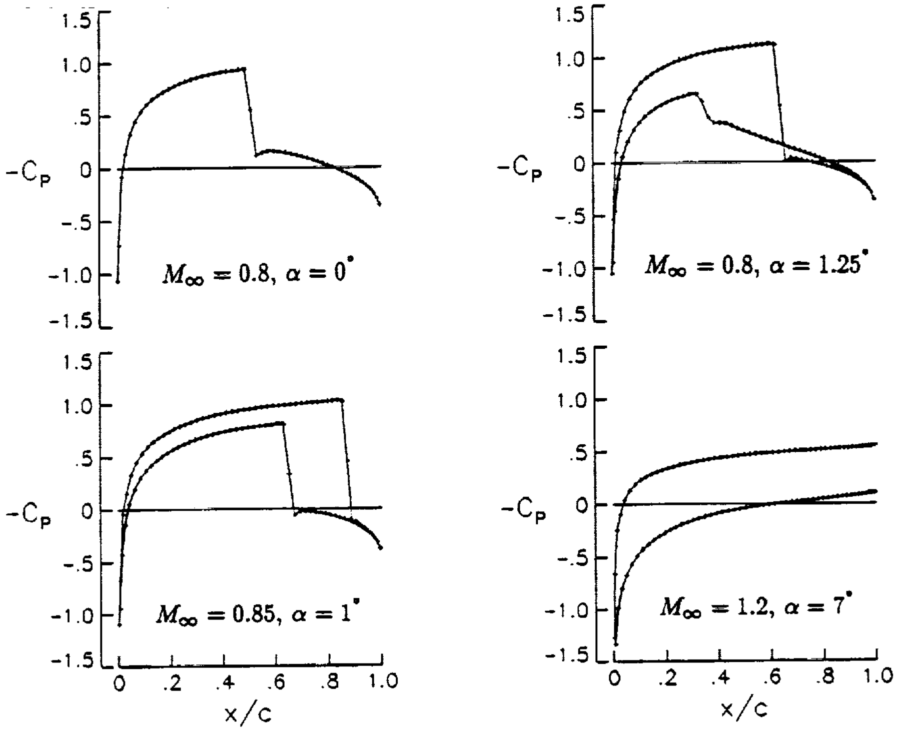




CONVERGENCE HISTORIES FOR NACA 0012 AIRFOIL

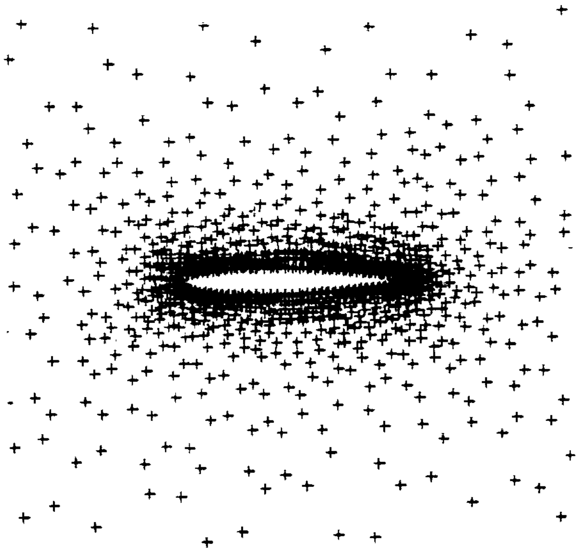


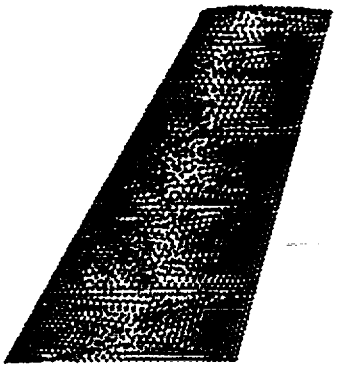
PRESSURE DISTRIBUTIONS FOR NACA 0012 AIRFOIL



GHOST POINTS FOR ONERA M6 WING

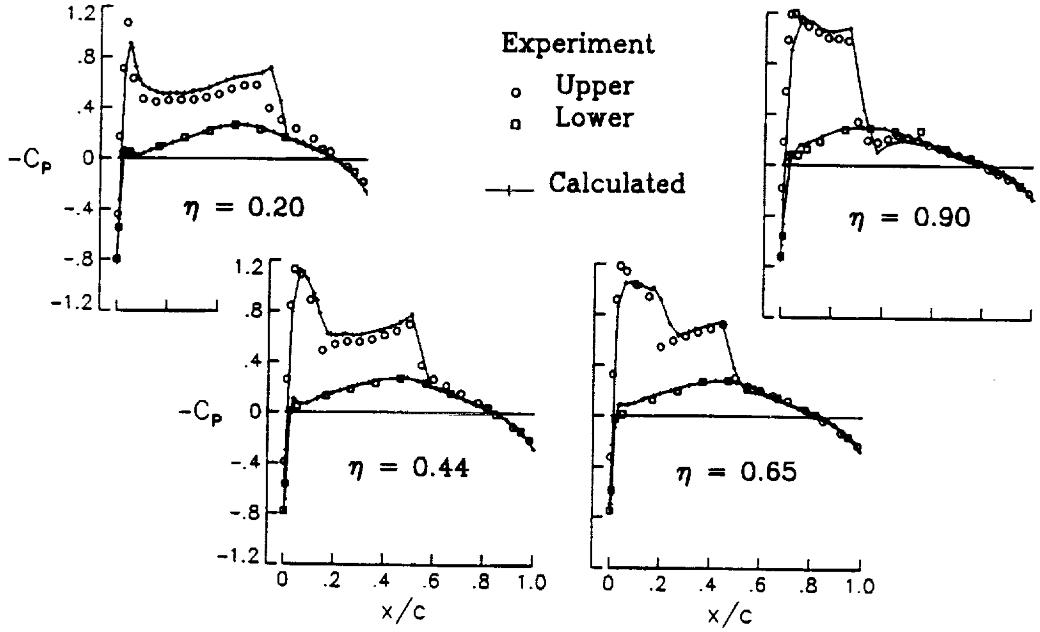
- Computational domain has a total of 108,705 points
- Symmetry plane
- Planform





EULER SOLUTION FOR ONERA M6 WING AT $M_{\infty} = 0.84$ AND $\alpha = 3.06^{\circ}$

- Pressure coefficient distribution



ADVANTAGES/DISADVANTAGES OF GRIDLESS METHOD

- Gridless method is not faster on a per point basis in comparison with methods developed for structured or unstructured grids
- Advantage is that it allows the use of fields of points where the points are more appropriately located and clustered, leading to far fewer points to solve a given problem
- Method retains the advantages of the unstructured grid methods
  - general geometry treatment
  - spatial adaptation
- Disadvantage is that it requires indirect addressing to store cloud to point information

SUMMARY

- Development of a gridless method for the solution of the 2D and 3D Euler and Navier-Stokes equations was described
- Method uses only clouds of points and does not require that the points be connected to form a grid as is necessary in conventional CFD algorithms
- Calculations for standard Euler and Navier-Stokes cases were found to be reasonably accurate and efficient in comparison with alternative methods and experimental data

FINAL THOUGHTS

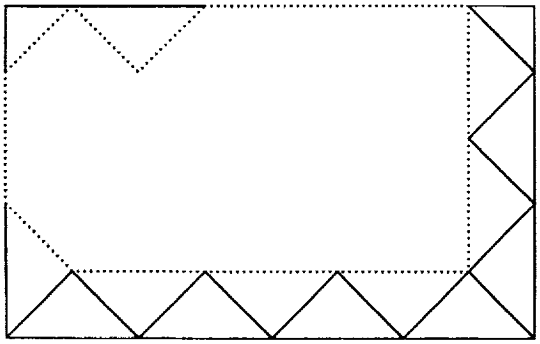
- The advent of gridless CFD does not obviate the need for "grid" generation — just the opposite
- Gridless CFD still requires surface definition and opens up the need to develop techniques for generating fields of points (in place of grids of points)

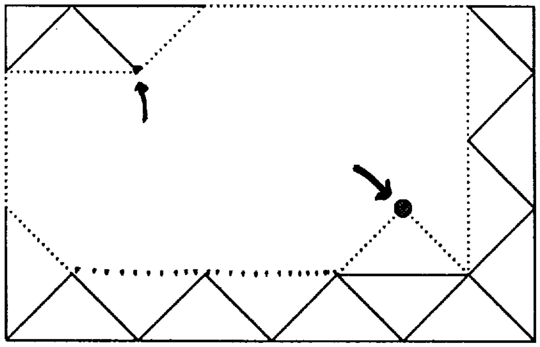
AN ADVANCING-FRONT DELAUNAY-TRIANGULATION ALGORITHM DESIGNED FOR ROBUSTNESS

D. J. MAVRIPLIS
I.C.A.S.E.- NASA LANGLEY RESEARCH CENTER

UNSTRUCTURED MESH GENERATION

- Advancing Front Method
- Delaunay Triangulation Techniques
- Combinations of Both
  - — Merriam
  - — Rebay, Muller and Roe
- Others (Computational Geometry)
  - — Edulsbrunner, Bern, Eppstein





ADVANCING FRONT

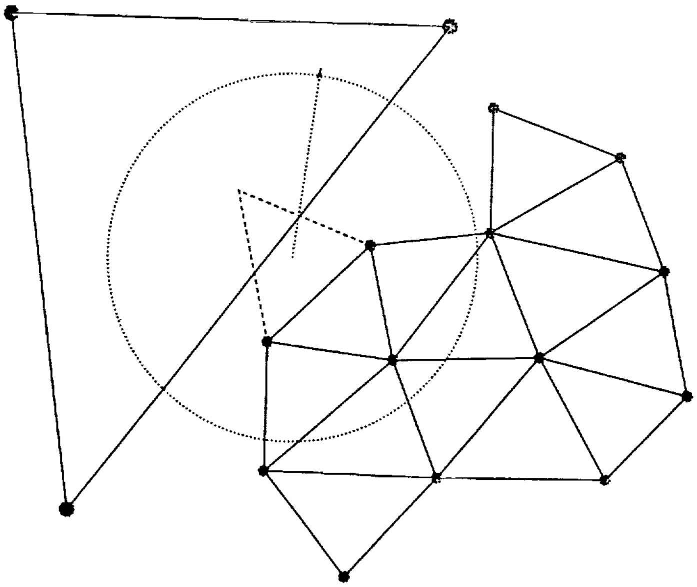
- Always Pick Smallest Front Edge
  - Front edges form heap-list
  - Dynamic data structure (insert-delete)

- Join Edge to New Point or Existing Front Point
  - Intersection checking

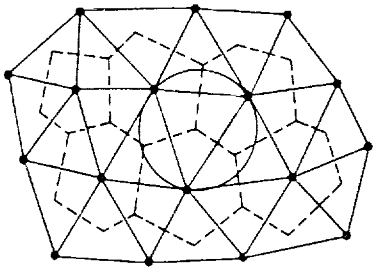
- Requires Location of "Close" Front Points
  - Quadtree Data Structures
  - Dynamic (insert-delete)

FAILURE OF ADVANCING FRONT

- Merging Two Fronts of Dissimilar Length Scales
- Usually Result of Rapid Variation in Field Function f(x,y)

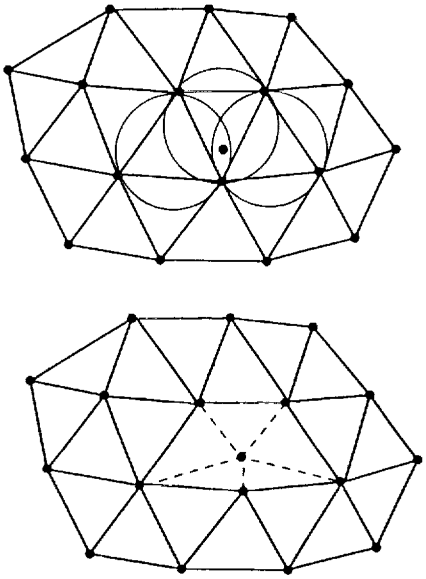


DELAUNAY TRIANGULATION



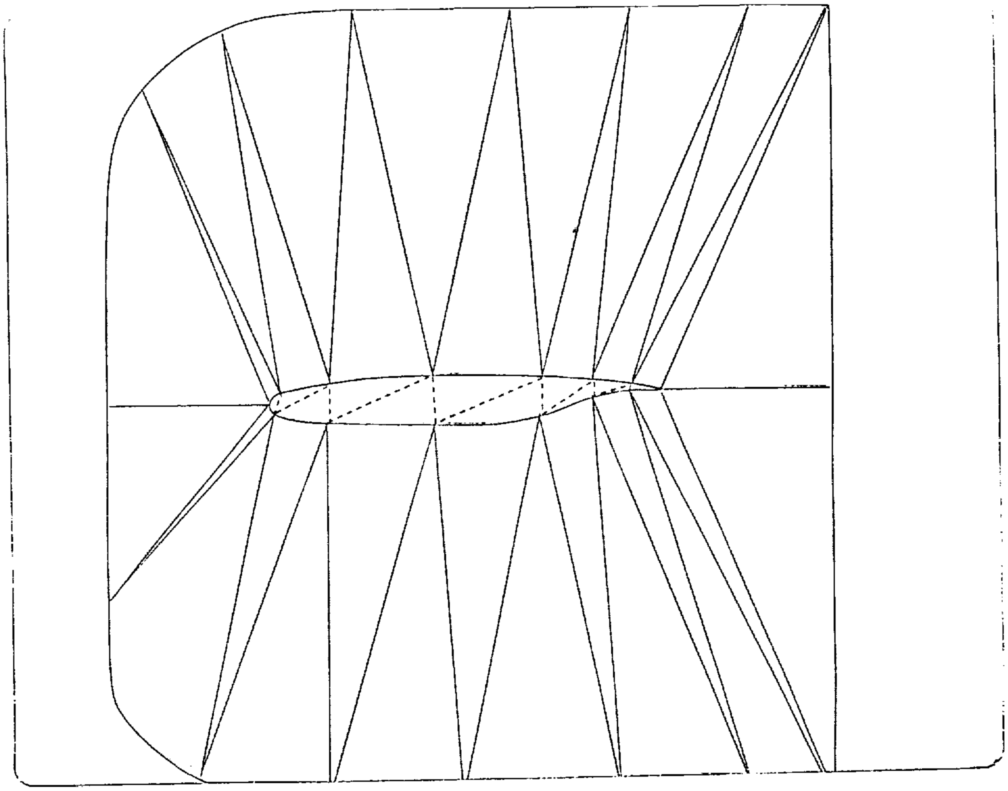
- Decouples Grid Points from Triangulation Procedure
- Produces Most Equiangular Triangles
- Purely Local Construction

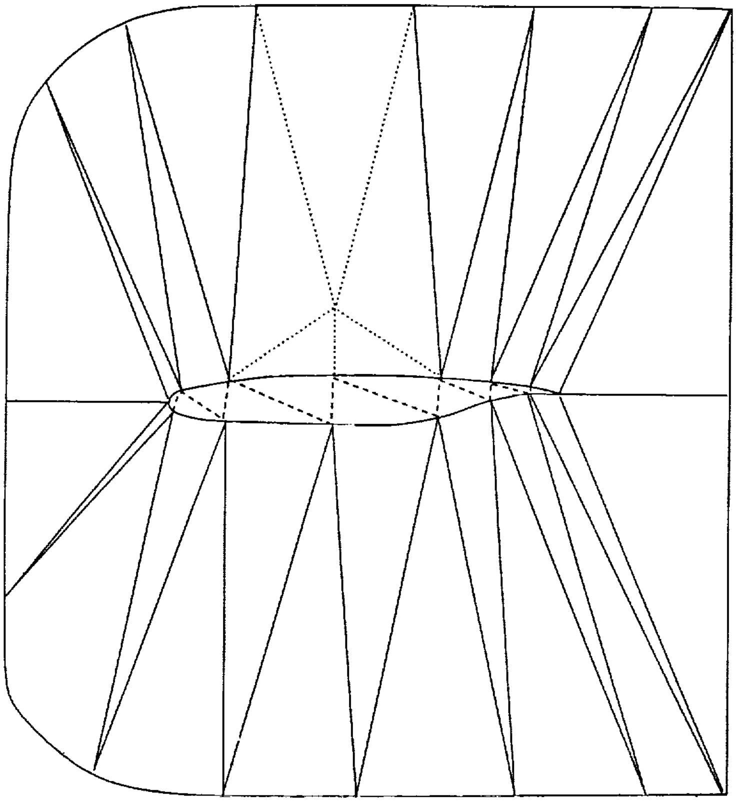
BOWYER'S ALGORITHM FOR DELAUNAY TRIANGULATION

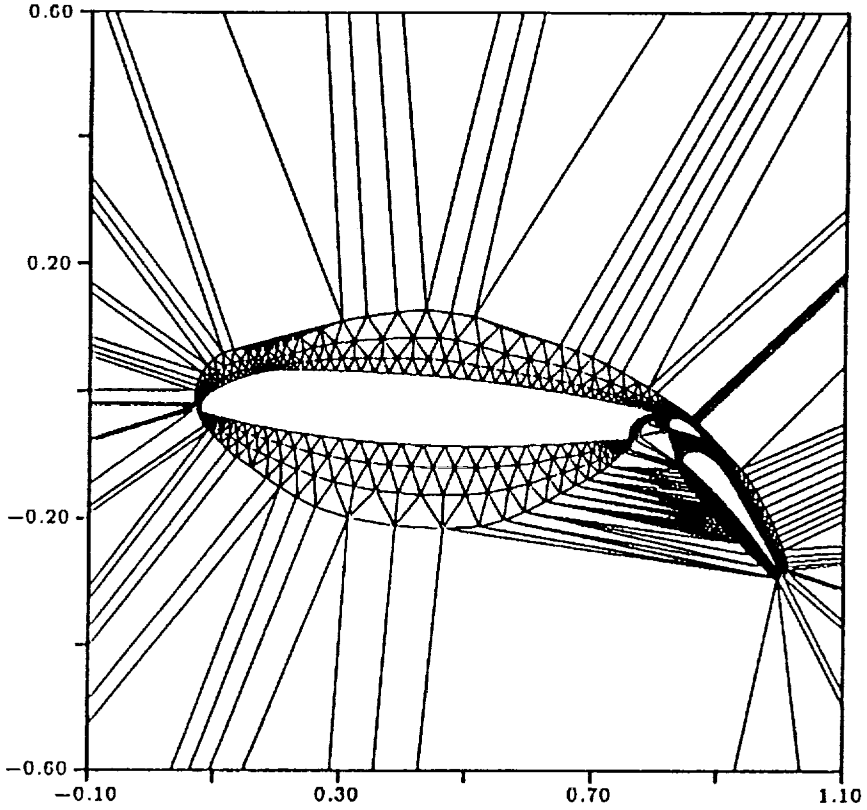


DELAUNAY TRIANGULATIONS

- Fundamental Data Structure in Computational Geometry
- Essentially a Reconnection Strategy
- Rigorous CG Construct
- Must be Modified for Non-Convex Domains
- Heuristic Point Placement Strategies
- Very Simple and Efficient Algorithms

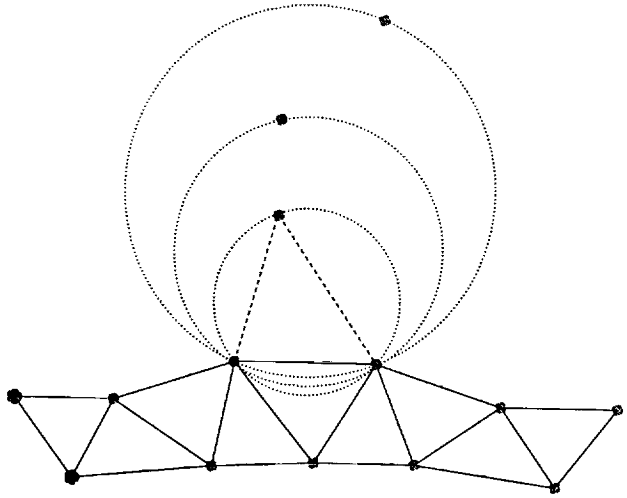





0.60
0.20
−0.20
−0.60
−0.10
0.30
0.70
1.10

TANAMURA-MERRIAM ALGORITHM

- Delaunay Triangulation of a Given Set of Points
- Advancing Front Generating Each Triangle Sequentially
- Never Modify Already Generated Triangles



Y A G G

Yet Another Grid Generator

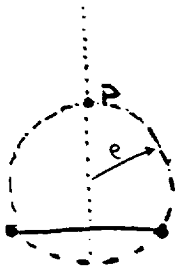
- 2-D Non-Stretched Grid Generation Fairly Easy
- Existing Methods Still Unsatisfactory
  - — Advancing Front
    - . Efficiency
    - . Robustness
      (Counter Examples for Merging Fronts)
  - — Delaunay Triangulation
    - . Boundary Integrity
    - . Round-off Error Failures
- Objectives:
  - — High Quality Mesh
  - — Efficient Strategy
  - — Theoretically Guaranteed Robustness
  - — Extendible to 3-D and Stretched Meshes

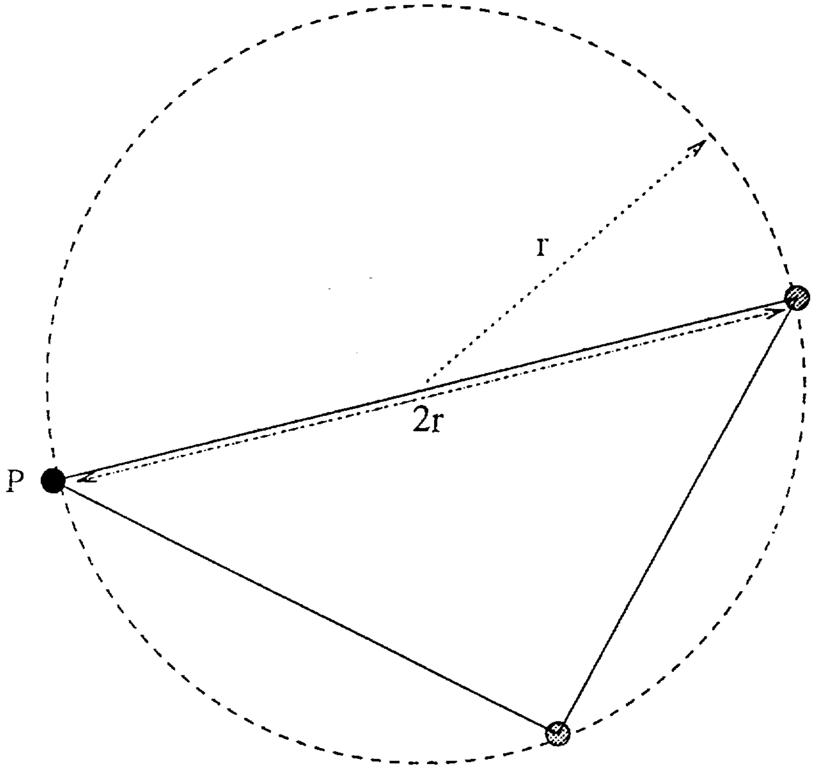
ADVANCING-FRONT DELAUNAY-TRIANGULATION

- Advancing Front Point Placement
- Delaunay Triangulation Reconnection
- Combines Advantages of Both Methods
  - Boundary Integrity Guaranteed
  - Rigorous CG Construction (Constrained Delaunay Triangulation)
  - Local Operations Only

ADVANCING-FRONT DELAUNAY-TRIANGULATION

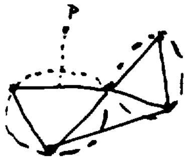
- Define Field Function for Circumcircles $\rho = f(x,y)$
- Choose Front Edge (Heap List)
- Place New Point (determined by $\rho = f(x,y)$)
- Construct All Triangles with New Point such that $\rho_{new} < \rho$
  - Join New Point to All Point Pairs of Grid and Retain only Valid Triangles
  - Only Test Subset of Grid Points Less than 2ρ away from. New Point



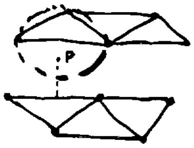


3 POSSIBILITIES FOR NEW POINT

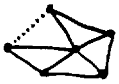
- New Point Does Not Lie in Any Existing Circumcircles
  - All Existing Triangles Remain Valid
  - New Triangles Formed with Front Points Only



- New Point Lies in Existing Circumcircle(s)
  - These Triangles Must Be Deleted Before Generation of New Elements
  - Requires Search for Intersected Circumcircles

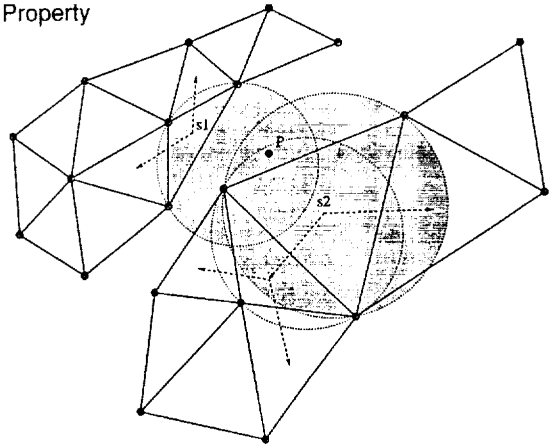


- New Point Not Needed
  - Valid Triangle by Joining Current Edge to a Front Point
  - Due to Variation in $\rho = f(x,y)$
  - Determined by Tanamura-Merriam Algorithm



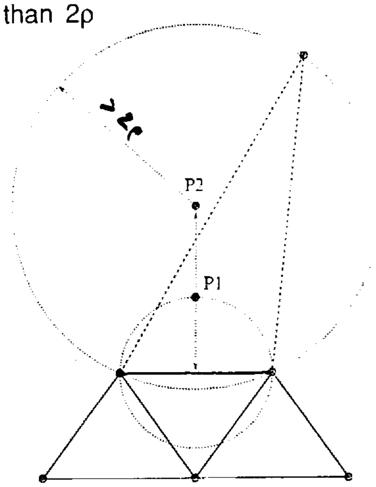
INTERSECTED CIRCUMCIRCLE SEARCH

- Grid Not Fully Connected (Neighbor Search Not Valid)
- Search All Front Triangles for Intersections
- Search Through Neighbors from Each Intersected Front Triangle
- Correctness Guaranteed by Delaunay Visibility Property



NEW POINT PLACEMENT

- Positioned Along Median to Yield Triangle of Radius $\rho = f(x,y)$
- Lower Limit P1 (Smallest Circumcircle)
- Upper Limit P2 (Equidistant from Other Points)
  — Only Relevent if There Exists a Point Closer than 4ρ which yields a Delaunay Triangle Smaller than 2ρ



AF-DT ALGORITHM

1.) Construct Original Front (Boundary Edges)

2.) Choose Edge of Front (Heap List)

3.) Determine Max Circumradius as $\rho = f(x,y)$

4.) Locate All Front Points Less Than 4ρ from Edge

5.) Use TM Algorithm to Determine The Triangle Formed Between Edge and "Close" Points

— If Triangle Exists and is Acceptable Go To 9

— If Triangle is Too Large:
Create New Point, Limit Position by Center

— If Triangle Does Not Exist:
Create New Point

AF-DT ALGORITHM

6.) Determine All Front Triangle Circles Intersected by New Point

7.) Determine All Interior Intersected Triangles (Neighbor Search)

8.) Remove All Intersected Triangles and Update Front

9.) Form All Acceptable Triangles With New Point and "Close" Points (which do not intersect boundary edges)

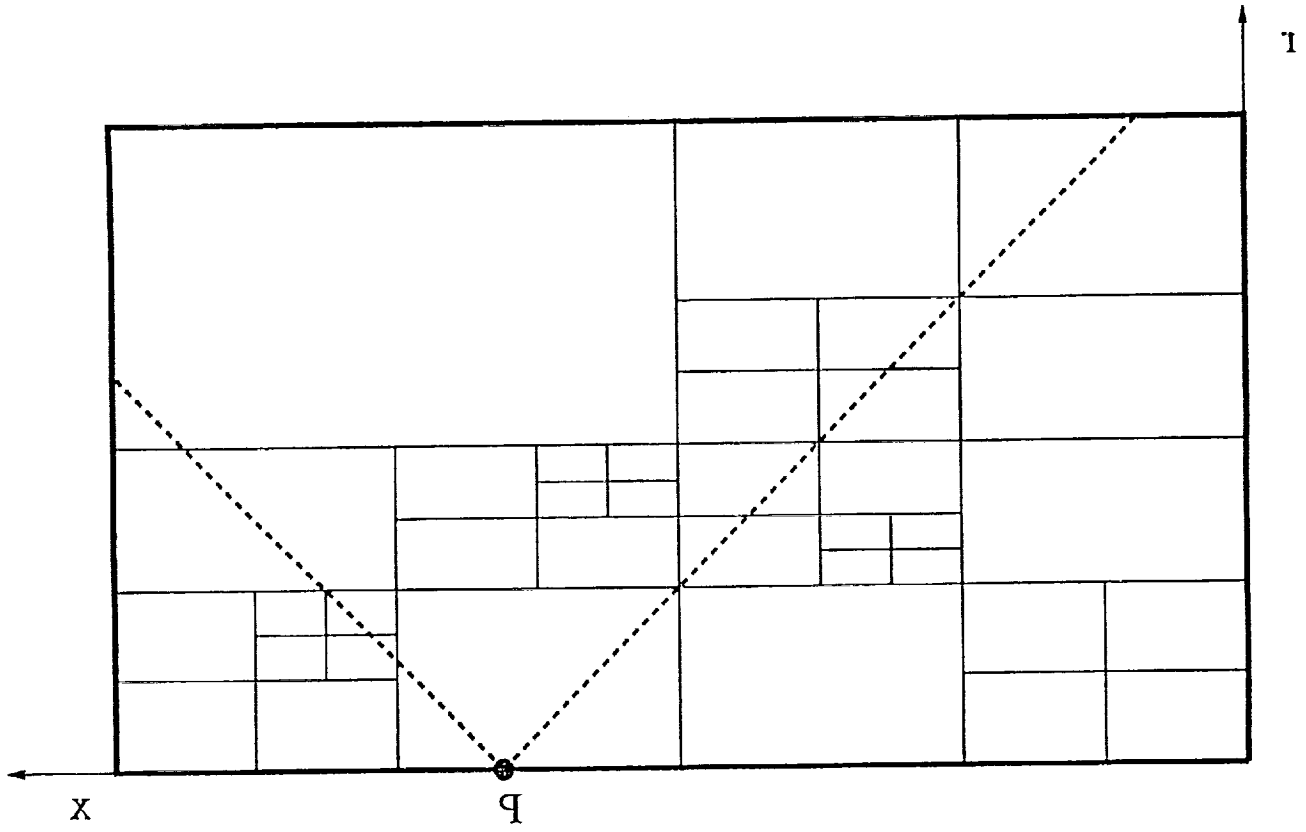
10.) Add Triangles to Mesh, Update Front

11.) If Front Queue Empty: Stop
Else: Go to 2

REQUIRED SEARCHES

- All Searches Based on Front, $O(\sqrt{N})$:
  - Dynamic
  - $O(N\log N)$

- Heap List for Choosing Front Edge
- Quadtree for Locating "Close" Points
- Octree for Intersected Front Triangle Circles
  - Point (x,y,r) in 3D
  - Generates Additional Length Scale



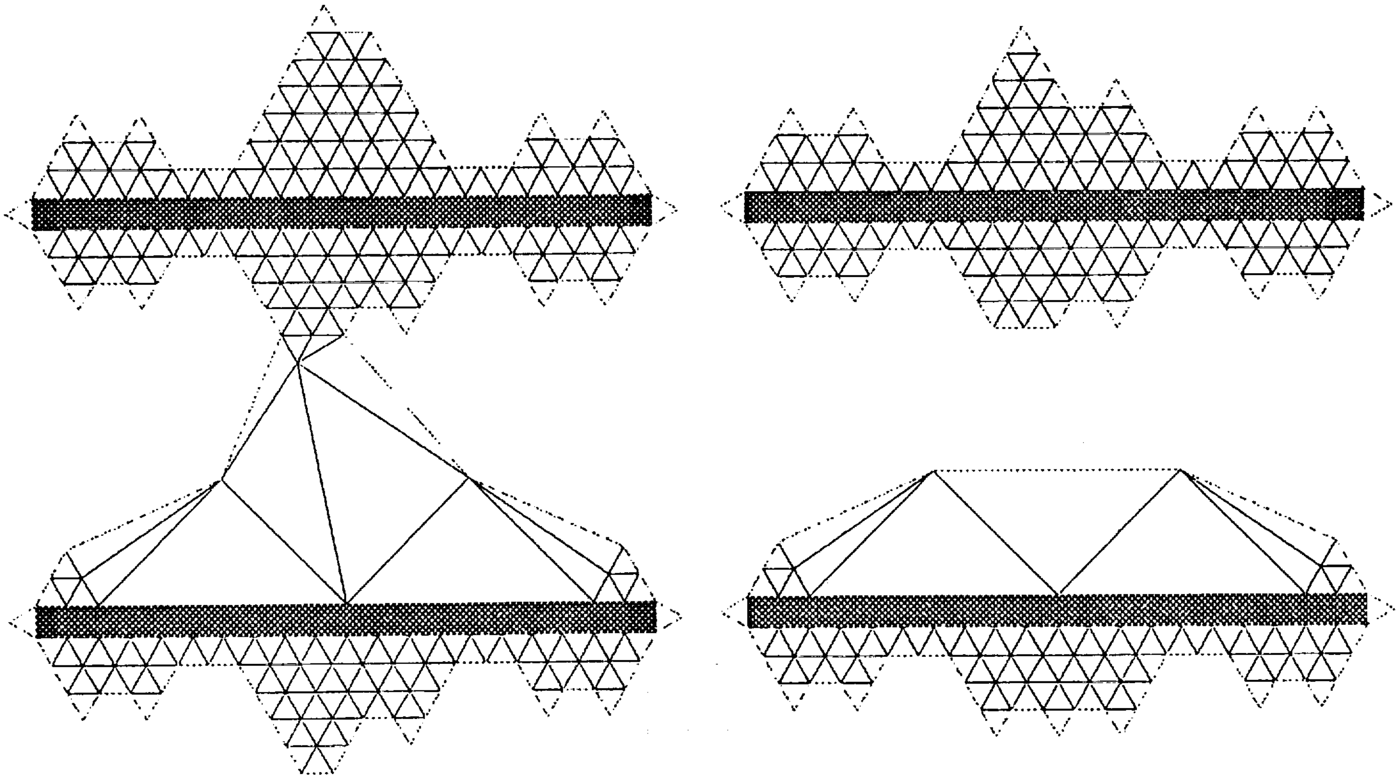
INTERSECTED CIRCUMCIRCLES

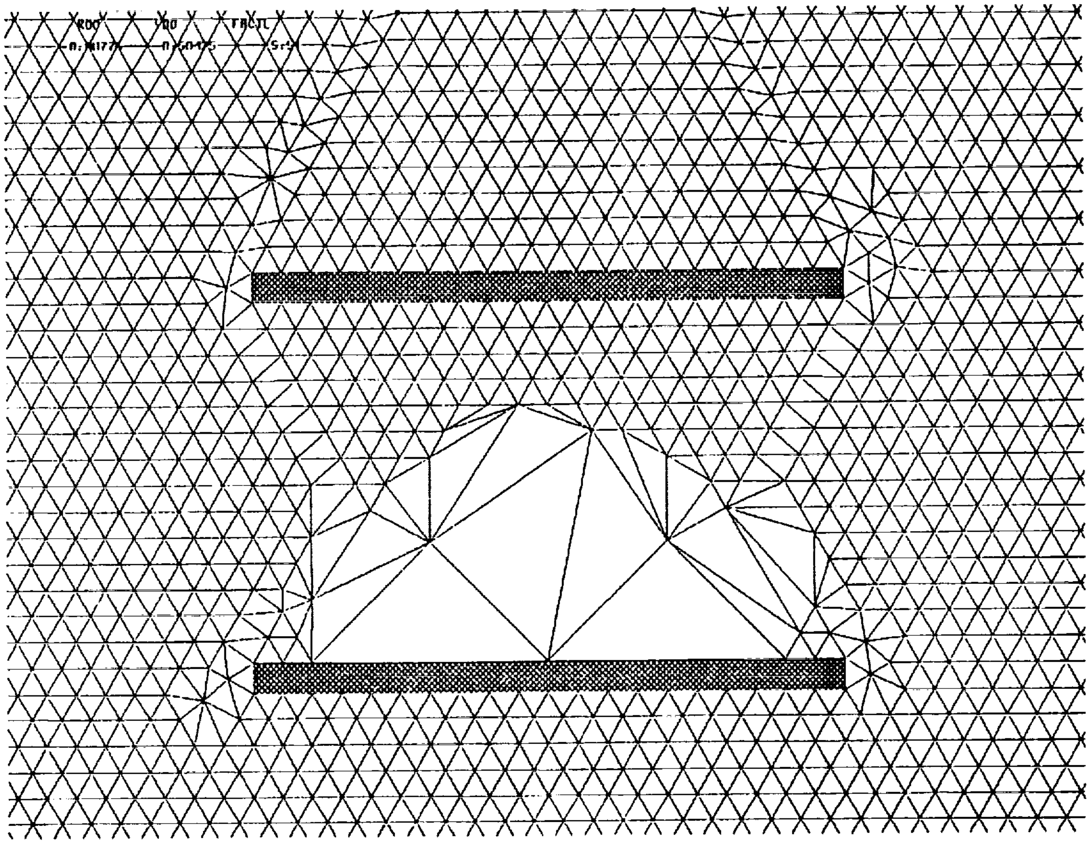
- Radius of Intersected Circles Provides Additional Length Scale

- Corresponds to f(x,y) on that (opposing) Front

- Useful in Regions of Rapid Variations in f(x,y)
  - if f(x,y) = constant circumcircles never intersected

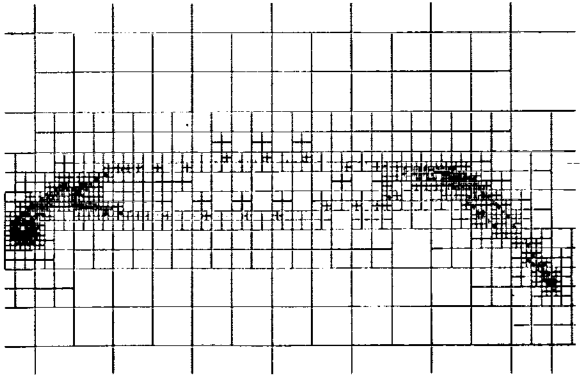
- Additional Length Scale is Missing in Traditional Advancing Front Method

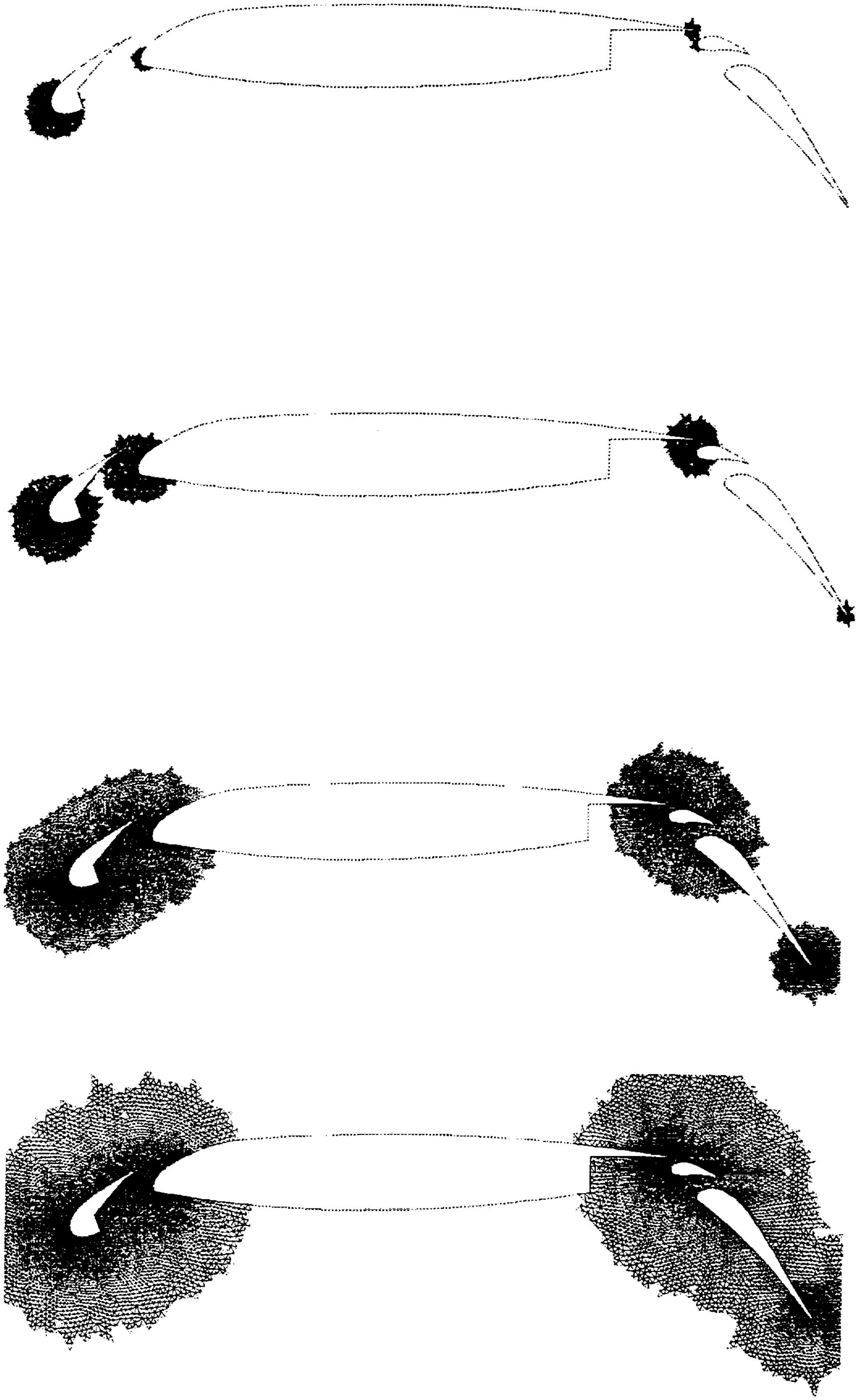


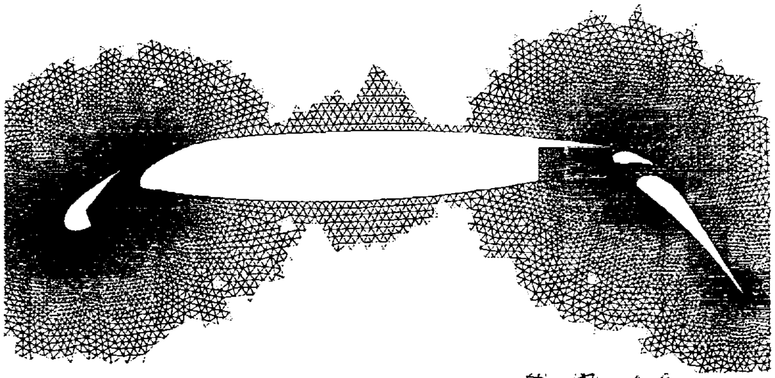


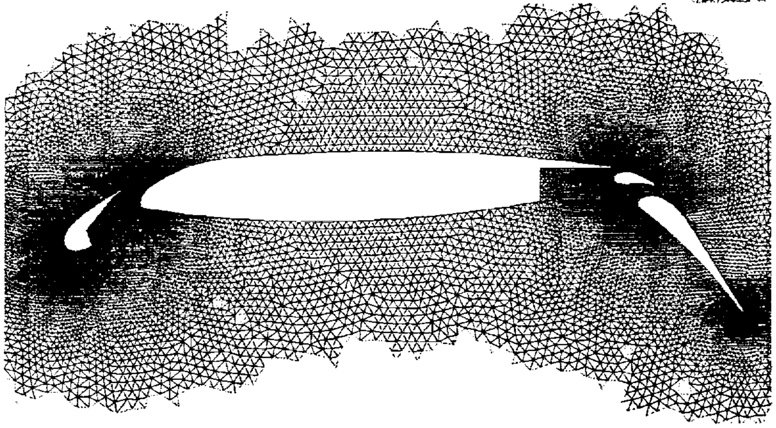
FIELD FUNCTION: F(X,Y)

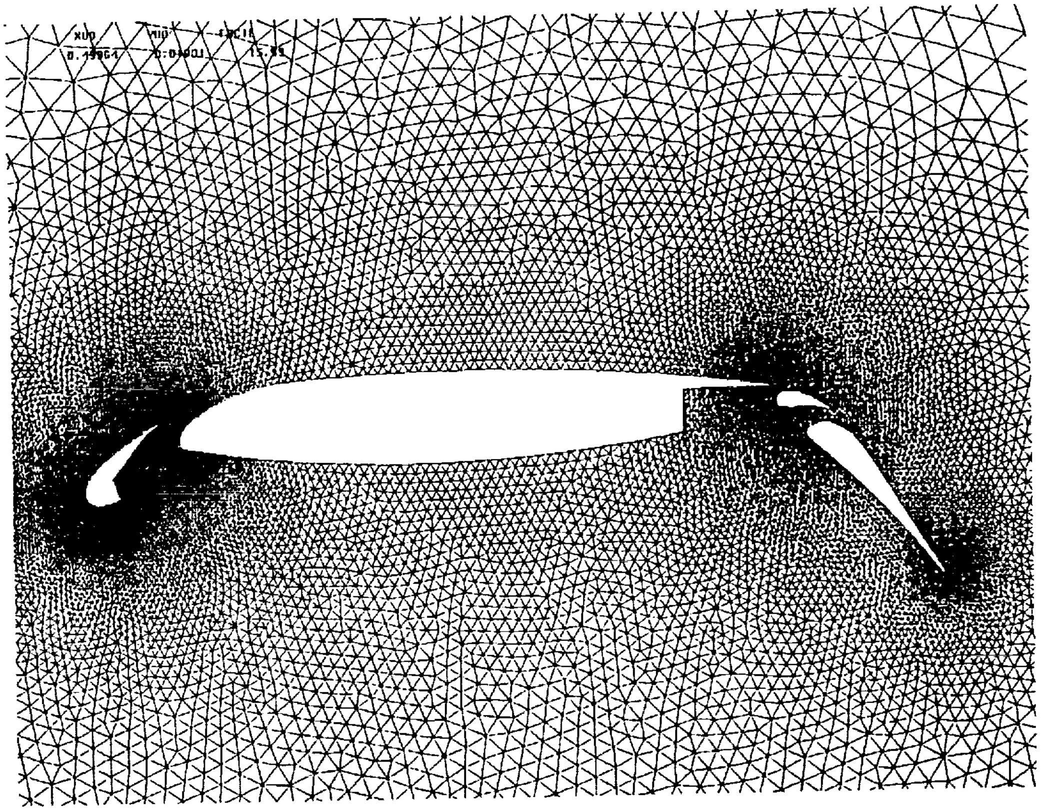
- Create Point Sources in Field and Solve Poisson Equation on Background Grid (Pirzadeh)
- Supporting Grid Taken as Initial Quadtree of Boundary Points
- To Determine f(x,y):
  - Traverse Quadtree to Locate Quad Containing (x,y)
  - $\rho(x,y)$ = Bilinear Interpolation of 4 Corners of Quad

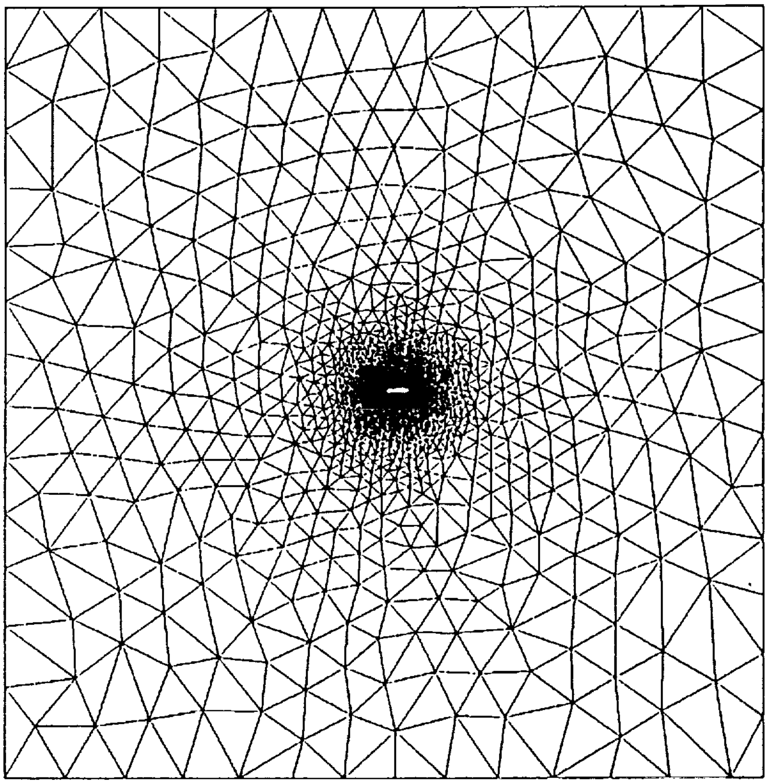









XUO
YUO



AF-DT ALGORITHM

- Boundary Integrity Guaranteed (Initial Condition)

- Robustness:
  - — All Local Operations
    (Never Create Unacceptable Triangles)
  - — Validity Guaranteed by Constrained DT
    (Two Length Scales Required)

- Efficiency:
  - — Generates Grid 1 Point at a Time
    (vs 1 Triangle at a Time)
  - — Complexity: $O(N\log N)$
  - — Storage: $O(\sqrt{N})$

- Counterpoint: Increased Coding Complexity

CONCLUSIONS

- Generates 500 Triangles/secd on SGI 4D35 Workstation

- 35% - 40% of Time Spent in Front Circle Test

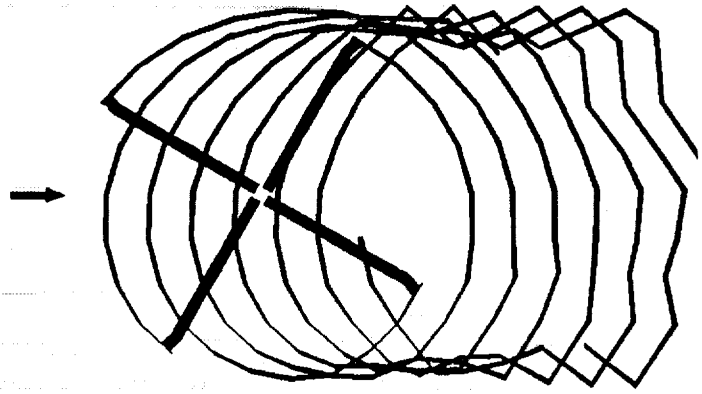
- Extensions to 3D

DYNAMIC MESH ADAPTION FOR TRIANGULAR AND TETRAHEDRAL GRIDS

RUPAK BISWAS
RIACS-NASA AMES RESEARCH CENTER

ROGER STRAWN
US ARMY AFDD-NASA AMES RESEARCH CENTER

ROTOR WAKE CAPTURING WITH A CFD METHOD

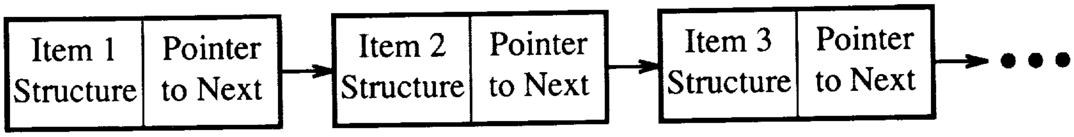


Requirements for Dynamic Mesh Adaption

- Anisotropic refinement capability in order to efficiently resolve directional flow features
- Coarsening required for both steady and unsteady applications
- Algorithm scaling important
- Low memory overhead using dynamic memory allocation
- CPU time comparable to a time step of the flow solver

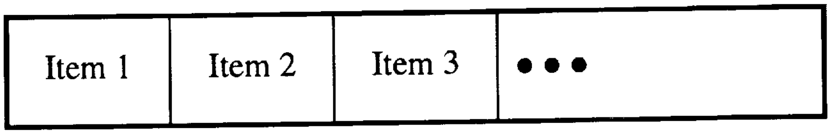
Linked-List Data Structure

Linked List



Item 1 Structure | Pointer to Next → Item 2 Structure | Pointer to Next → Item 3 Structure | Pointer to Next → • • •

Static Array

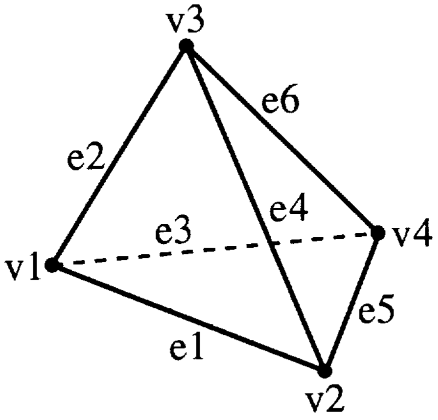


Item 1 | Item 2 | Item 3 | • • •

- Facilitates quick insertion and deletion of items
- Dynamically allocates and frees memory
- No need for compaction and garbage collection

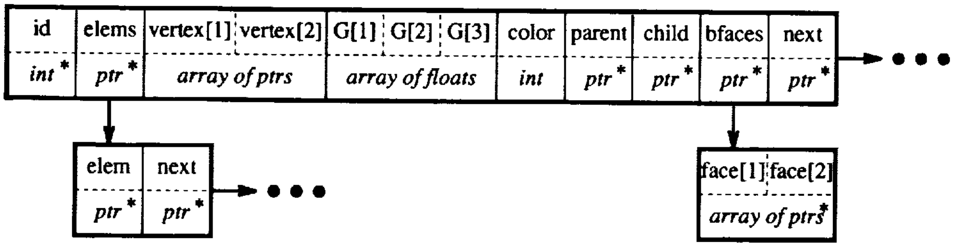
Edge-Based Data Structure

- An edge is a line segment that connects two vertices
- A tetrahedron can be uniquely defined by its six edges: e1, e2, e3, e4, e5, e6



Adaptive-Grid Data Structure

Edge List

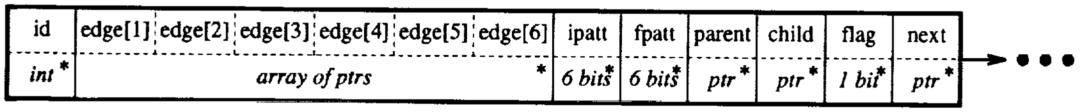


id	elems	vertex[1]	vertex[2]	G[1]	G[2]	G[3]	color	parent	child	bfaces	next
int *	*ptr* *	*array of ptrs*		*array of floats*			*int*	*ptr* *	*ptr* *	*ptr* *	*ptr* *

elem	next
ptr *	*ptr* *

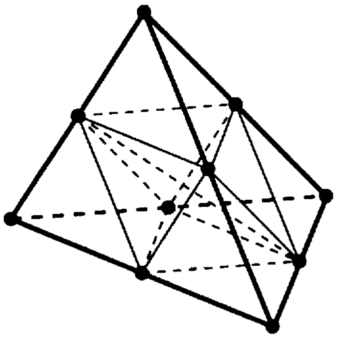
face[1]	face[2]
array of ptrs *	

Element List

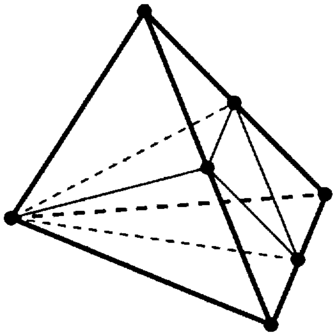


id	edge[1]	edge[2]	edge[3]	edge[4]	edge[5]	edge[6]	ipatt	fpatt	parent	child	flag	next
int *	*array of ptrs* *						*6 bits* *	*6 bits* *	*ptr* *	*ptr* *	*1 bit* *	*ptr* *

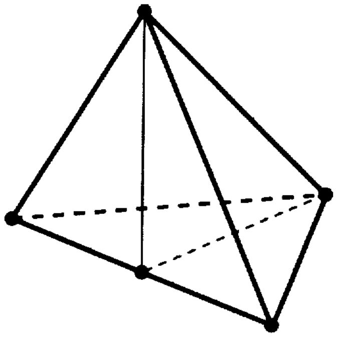
Three Types of Element Subdivision



1:8



1:4

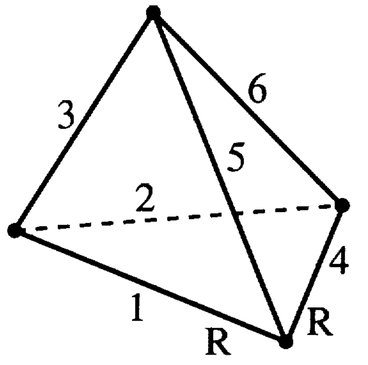


1:2

- The 1:4 and 1:2 elements are the result of anisotropic refinement or act as buffers between the 1:8 elements and the surrounding unrefined mesh

Mesh Refinement

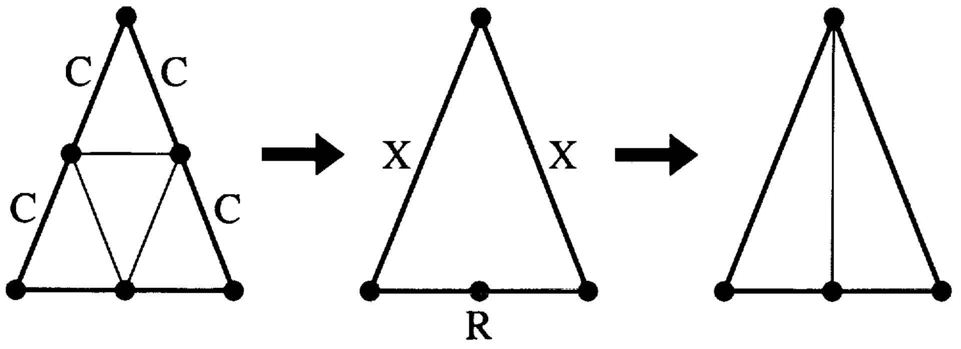
- Individual edges marked for refinement
- Marked edges combined to form binary pattern (ipatt) for each element
- Element patterns upgraded to form valid 1:8, 1:4, or 1:2 subdivisions (fpatt)



6	5	4	3	2	1	Edge #
0	0	1	0	0	1	ipatt = 9
0	0	1	0	1	1	fpatt = 11

Mesh Coarsening

- Elements with edges to be coarsened immediately revert back to their parents
- Parent elements have their ipatt values modified to reflect the fact that some edges have coarsened
- Parent elements then appropriately refined

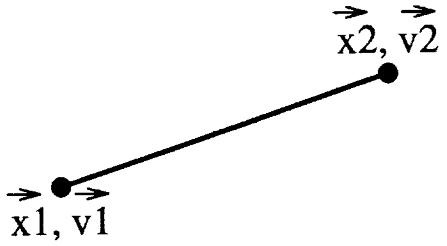


Additional Constraints for Coarsening

- In general, edges and elements must be coarsened in an order reversed from the one by which they were refined
- An edge can coarsen if and only if its sibling also marked for coarsening
- Edges of non-leaf elements or of their siblings cannot be coarsened

Anisotropic Error Indicator for Edges

- Adaption based on an error indicator computed for every edge of the mesh
- Flow gradients must be aligned with the edges for them to be marked for refinement
- Relative number of edges marked for coarsening and refinement adjusted to maintain a user-specified upper limit on problem size



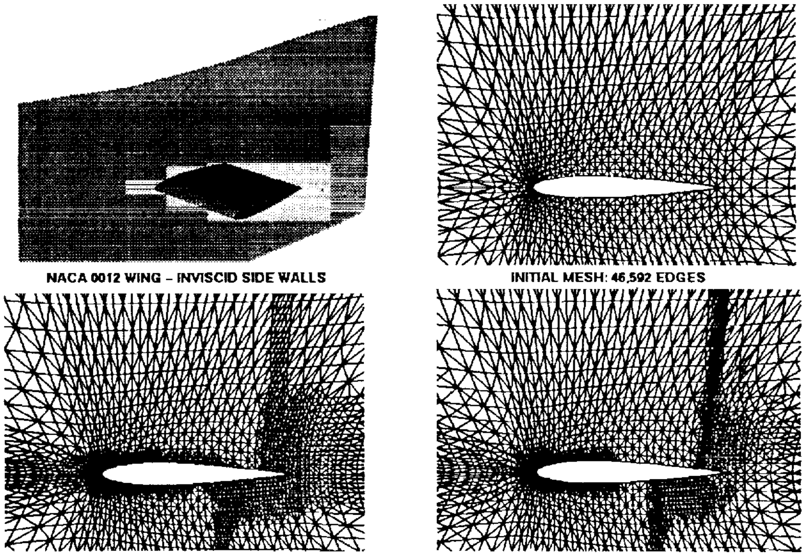
$$|E_e| = \|\vec{\Delta x} \cdot \vec{\Delta v}\|$$

Unstructured-Grid Euler Solver

- Basic code written by Barth; rotary-wing version developed by Strawn and Barth
- Finite-volume method with upwind differencing
- Computational control volumes centered at cell vertices
- Edge data structure allows arbitrary polyhedra
- Solution advanced in time using conventional explicit procedures

EXAMPLE: 3-D ADAPTIVE GRID REFINEMENT AND COARSENING

FSMACH = 0.85, ALPHA = 1.0 DEG



NACA 0012 WING – INVISCID SIDE WALLS

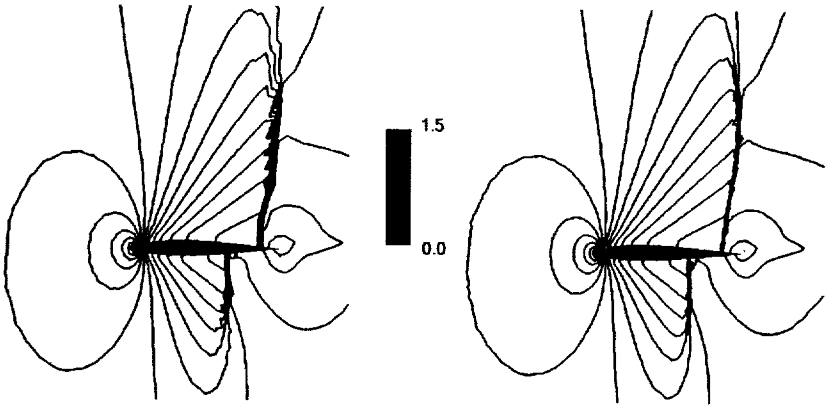
INITIAL MESH: 46,592 EDGES

FIRST REFINEMENT: 75,656 EDGES

3 REFINEMENT LEVELS, 2 COARSENING LEVELS
85,869 EDGES

MACH NUMBER CONTOURS

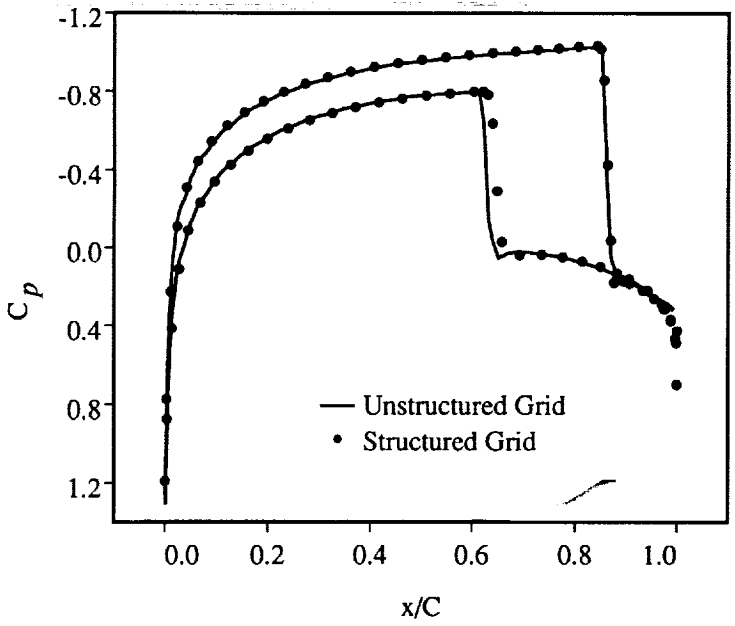
FSMACH - 0.85, ALPHA - 1.0 DEG.



ORIGINAL MESH: 46,592 EDGES

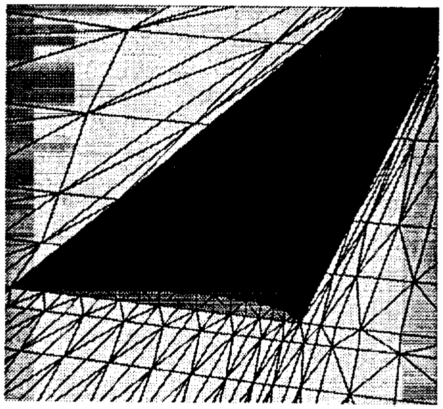
FINAL MESH: 85,869 EDGES

Example: Inviscid 3-D Wing

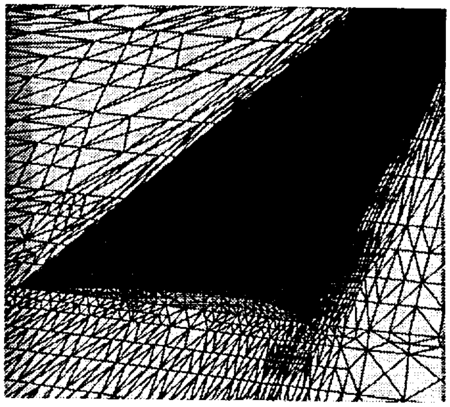


SOLUTION -ADAPTED MESH FOR A HOVERING ROTOR

Mtip - 0.90, AR - 13.7, NONLIFTING BLADE



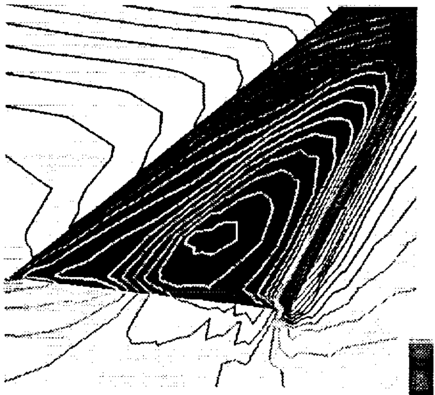
INITIAL MESH: 5,267 POINTS, 28,841 EDGES



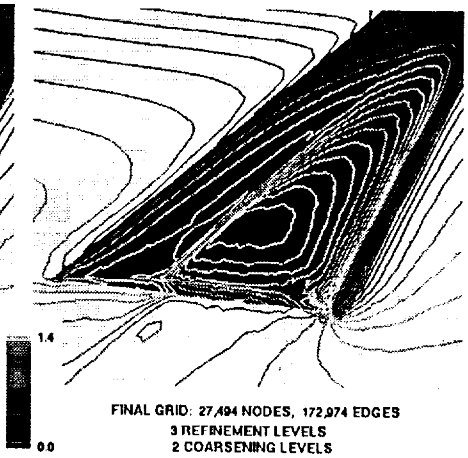
FINAL GRID: 27,494 NODES, 172,974 EDGES
3 REFINEMENT LEVELS
2 COARSENING LEVELS

MACH CONTOURS FOR THE ROTOR BLADE

Mtip - 0.90, AR - 13.7, NONLIFTING BLADE



INITIAL MESH: 5,267 POINTS, 28,841 EDGES



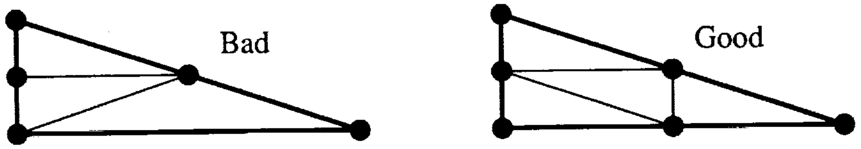
FINAL GRID: 27,494 NODES, 172,974 EDGES
3 REFINEMENT LEVELS
2 COARSENING LEVELS

Current Projects

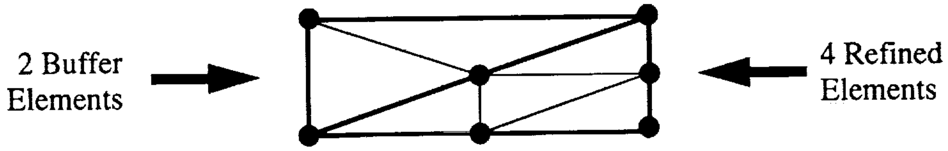
- Mesh quality for 2-D and 3-D adaptive schemes — Goal is to guarantee that mesh quality does not degrade
- Concurrent operation of flow solver and dynamic mesh adaption on CM-5
- Error estimates/indicators for unstructured-grid solutions

Mesh Quality for Solution-Adaptive Grids

- Elements are checked for quality before they are actually subdivided

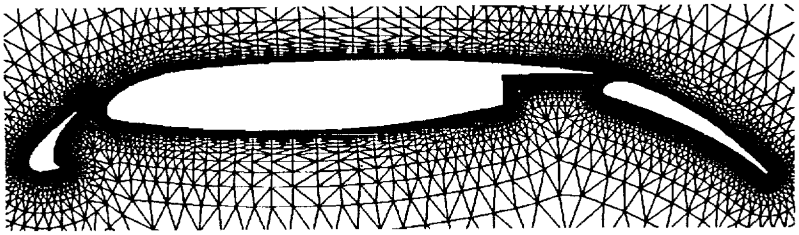


- Buffer elements with large angles that may result at boundaries between different refinement levels are "corrected" before they are further subdivided

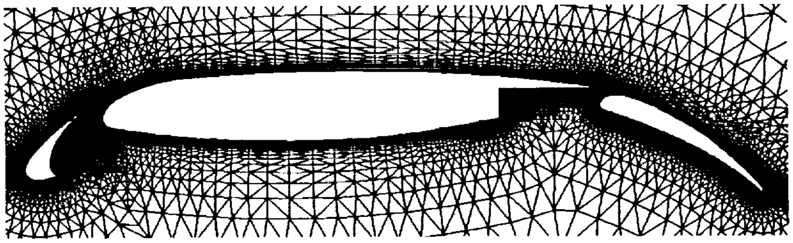


- Both techniques can be used in two and three dimensions

MESH ADAPTION FOR A 2-D VISCOUS GRID



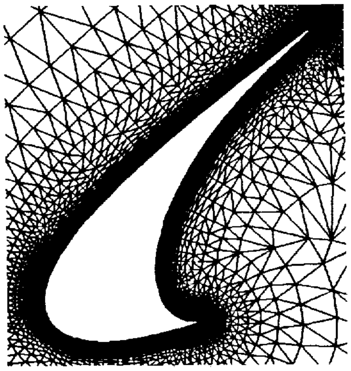
ORIGINAL GRID: 27,705 NODES, 54,725 TRIANGLES



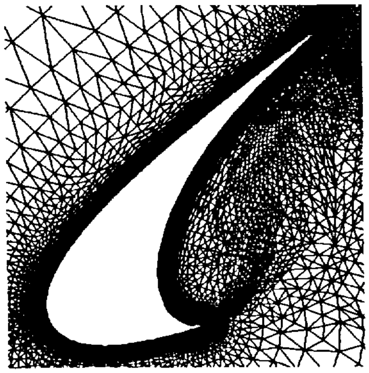
3 REFINEMENT LEVELS, 2 COARSENING LEVELS: 73,142 NODES, 144,270 TRIANGLES

MESH ADAPTION FOR A 2-D VISCOUS GRID

CLOSE-UP OF FIRST AIRFOIL ELEMENT



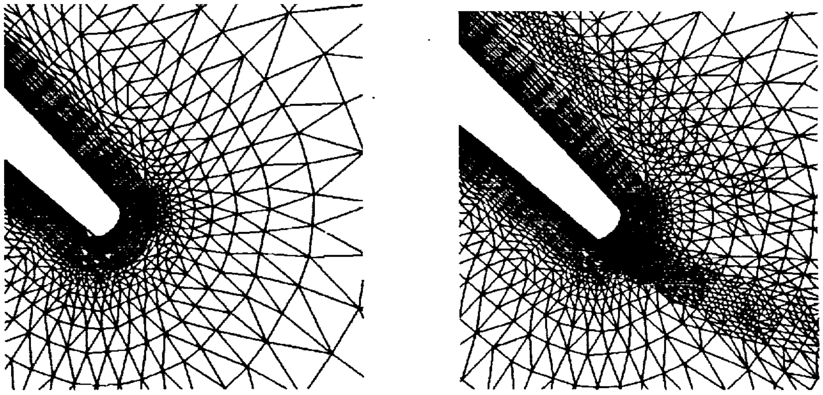
ORIGINAL GRID: 27,705 NODES, 54,725 TRIANGLES



3 REFINEMENT LEVELS, 2 COARSENING LEVELS:
73,142 NODES, 144,270 TRIANGLES

MESH ADAPTION FOR A 2-D VISCOUS GRID

TRAILING EDGE OF THIRD AIRFOIL ELEMENT



ORIGINAL GRID:
27,705 NODES, 54,725 TRIANGLES

3 REFINEMENT LEVELS, 2 COARSENING LEVELS:
73,142 NODES, 144,270 TRIANGLES

Summary and Conclusions

- A new procedure has been developed for dynamic adaption of two- and three-dimensional unstructured grids
- An innovative new data structure combined with dynamic memory allocation results in fast coarsening and refinement
- Mesh quality can be "controlled" for arbitrary refinement levels
- Computed results using the solution-adaptive algorithm show excellent agreement with results for conventional structured-grid solvers

CARTESIAN-CELL BASED GRID GENERATION AND ADAPTIVE MESH REFINEMENT

WILLIAM J. COIRIER
NASA LEWIS RESEARCH CENTER

KENNETH G. POWELL
UNIVERSITY OF MICHIGAN

MOTIVATION

Wouldn't it be nice to just define the geometry and the free-stream conditions, and let the grid generation/adaptive refinement do the rest?

Objectives

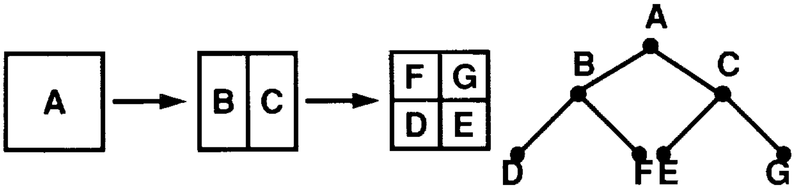
- Automated Grid Generation for Complex Bodies
- Automated Grid Refinement (Convergence?)
- Alternative to Triangular/Tetrahedral Meshes

A Cartesian-Mesh Approach

- Use Cartesian Cells of Unit Aspect Ratio to Create Background Mesh
- "Cut" Bodies Out of Background Mesh, Creating Irregularly Shaped Boundary Cells
- Arbitrary Numbers of Arbitrarily Shaped Bodies Are Allowed
- Geometry Defined With Sets of General Basis Functions Along Surfaces
- Background Mesh Created By Recursively Refining Cartesian Cell Into Four Cells

GRID GENERATION

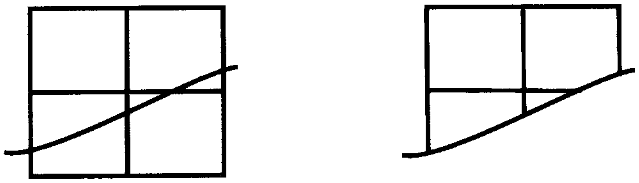
- Grid Generation Process Creates Binary Tree



- Binary Tree Allows Quad and Binary Refinement
- Connectivity/Tree Hierarchy Closely Related

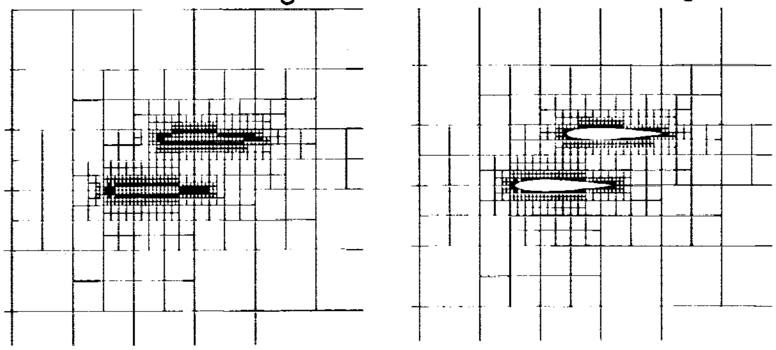
GRID GENERATION

- Recur to Leaves of Tree and Determine Intersections (if any) with Bodies
- Use Simple Set of Rules to Determine If It is Legal to Cut Leaf into Cell: Recursively Refine if Illegal
- Vertex Locality Used to Determine Cut Cell Geometry



CELL CUTTING

- Example: Staggered Biplane Configuration of Clarke, Salas and Hassan (AIAA J. 1986)



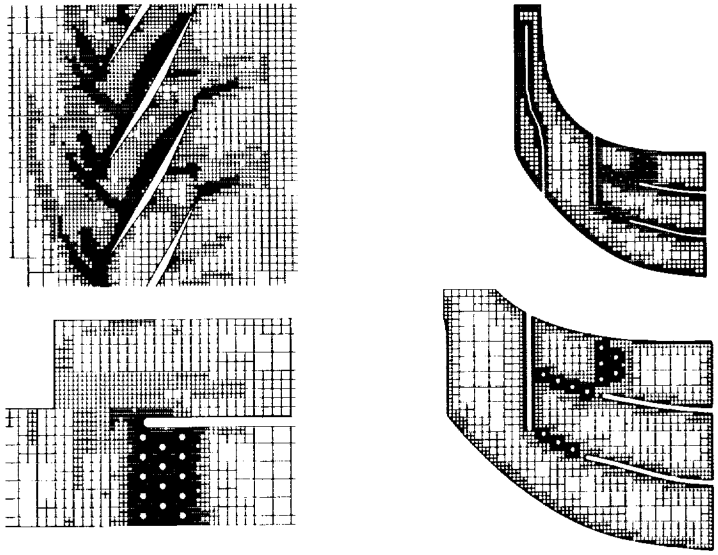
Prior to Cutting

After Cutting

DATA STRUCTURE(S)

- Cartesian Cell Geometric Data Inferred From Tree
- Cut Cell Geometric Data From (Local) Ordered List of Pointers to (Global) List of Vertices
- Connectivity Is Inferred Directly From Tree By Logical Tree Traversals (Centroid Compares, Face Matching)
- Code Written in ANSI C: Dynamic Memory Allocation/Deallocation, Self-Referential Data Structures

SAMPLE GRIDS



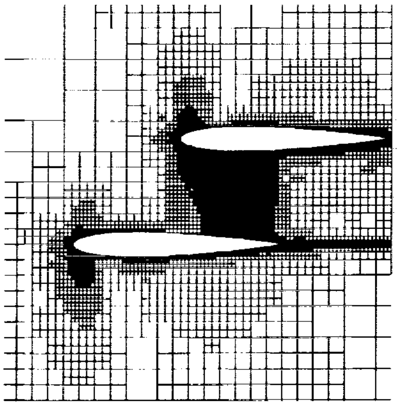
FLOW SOLVER FORMULATION

- Cell Centered, Finite Volume, Upwind Based Scheme
- Linear Reconstruction (Minimum-Energy) of Primitives Used to Compute Left/Right Interface States as Input to Approximate Riemann Solver
- Adaptive Mesh Refinement Using Cell Size Weighted Criterion Based on Velocity Divergence and Curl (Compressibility and Rotation)
- Perform Flow Solve/Adaptation Set Number of Times

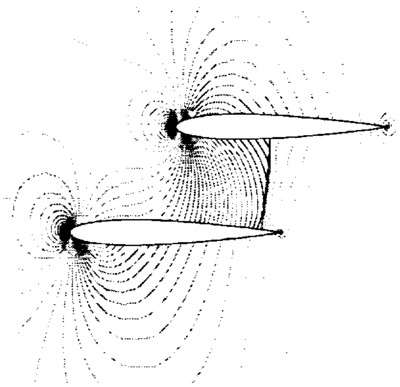
ADAPTIVE MESH REFINEMENT

- Staggered Biplane Case

Grid



Pressure Contours

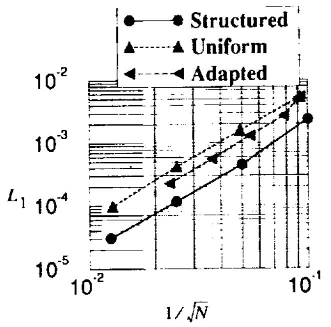


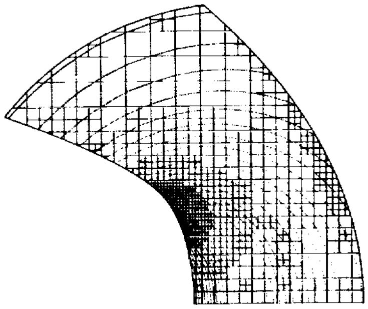
ACCURACY ASSESSMENT

- Use Exact, Analytic Solution (Ringleb's Flow)
- Infer Order of Error From Uniform and Adaptive Refinement
- Infer Magnitude of Error by Comparing to Structured Solver
- Asks Question:

Can Adaptive Mesh Refinement Beat Uniform Refinement and/or Structured Uniform Refinement?

ACCURACY ASSESSMENT

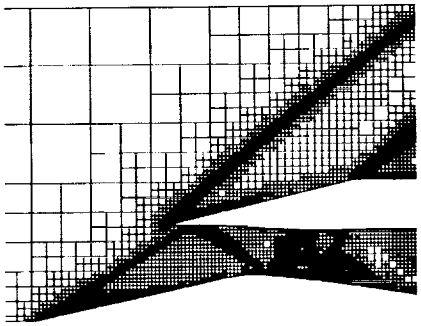


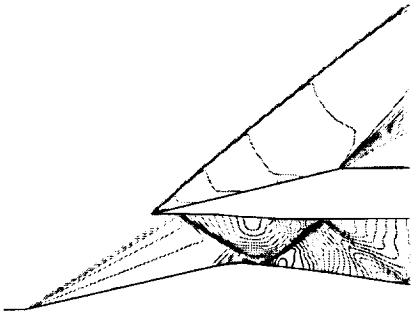


- Approach is 2nd Order (Global), Better than 1st (Local)
- **Smooth Flow:** Can't Beat Uniform Refinement or Structured

ADAPTIVE MESH REFINEMENT

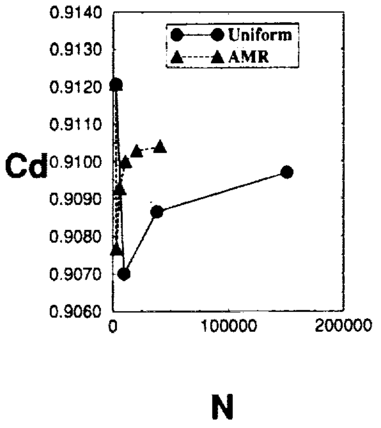
- What About Non-Smooth Flows?
- Grid Convergence Study on Supersonic, Axi-Symmetric, Mixed-Compression Inlet





ADAPTIVE MESH REFINEMENT

- Compare Uniform and Adapted Drag Coefficients

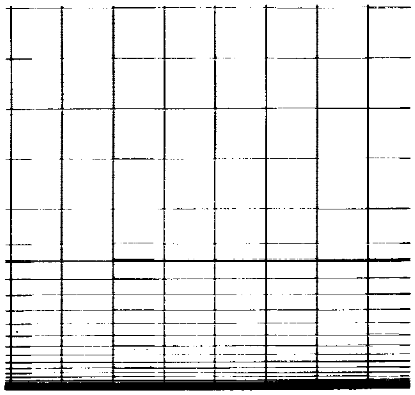


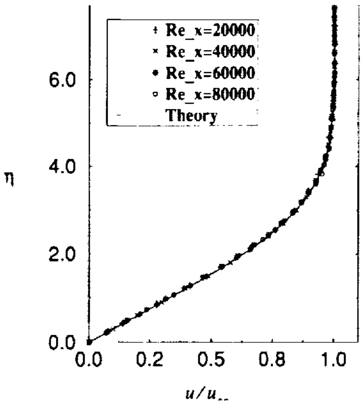
Conclusion

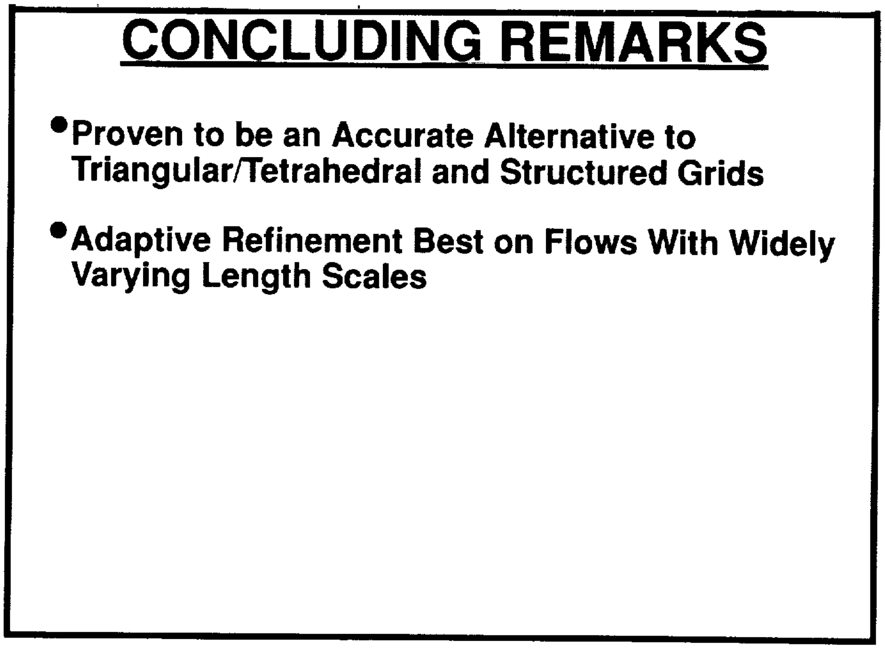
- AMR Grid Converged
- Uniform Not Converged (150,683 Cells!)
- Adaptive Mesh Refinement Best For Non-Smooth Flows With Multiple Length Scales

VISCOUS FLOWS

- Presently Extending to Viscous Flows
- "Cut" Level Distance Lines From Bodies

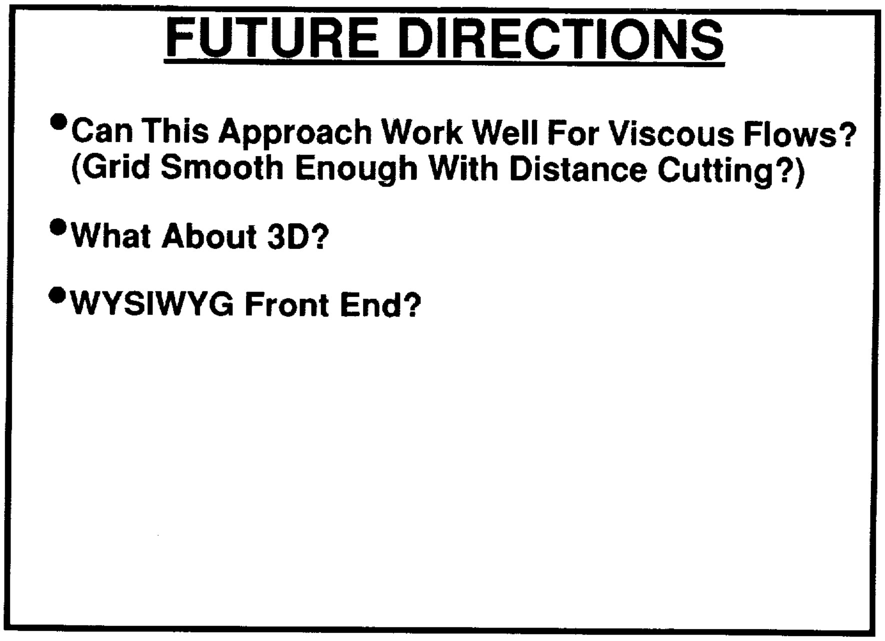






CONCLUDING REMARKS

- Proven to be an Accurate Alternative to Triangular/Tetrahedral and Structured Grids
- Adaptive Refinement Best on Flows With Widely Varying Length Scales



FUTURE DIRECTIONS

- Can This Approach Work Well For Viscous Flows? (Grid Smooth Enough With Distance Cutting?)
- What About 3D?
- WYSIWYG Front End?

2D & 3D HYPERSONIC FLOWS WITH UNSTRUCTURED MESHES

RAJIV THAREJA
LOCKHEED ENGINEERING & SCIENCES COMPANY

OUTLINE

Introduction

2D Viscous Shock-Shock Interaction

3D Inviscid NASP-Like (Unadapted)

3D Inviscid NASP-Like (Adapted)

INTRODUCTION

Funded by Aerothermal Loads Branch (NASA LaRC)

Development of finite elements in fluids and unstructured grid generation (began 1983-1984)

In-house research

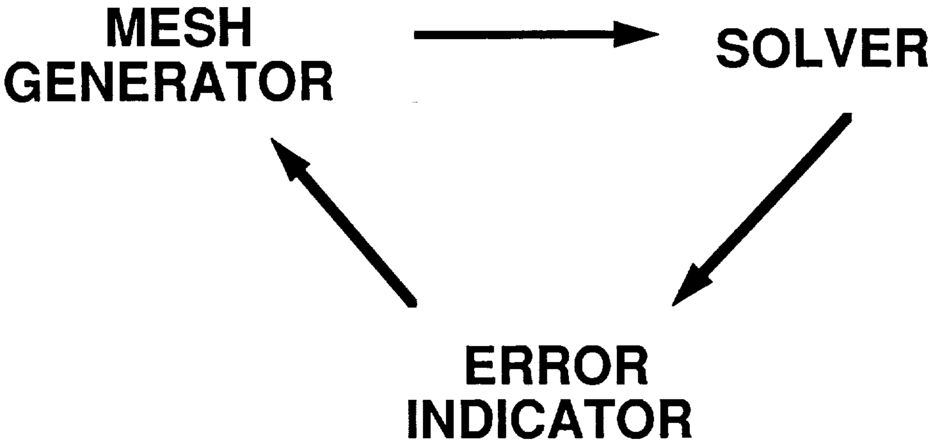
- Civil servants and contractors

Grantees' research

- Morgan, Lohner, Peraire (Swansea)
- Hughes (Stanford)
- Oden (Austin)
- Thornton (ODU)

Current status

COUPLED MODULES



MESH GENERATOR

SOLVER

ERROR INDICATOR

MESH GENERATION

Advancing Front Method

Generation Parameters

- Spacing
- Orientation
- Stretching

Sources

- Point
- Line
- Triangles

Background Mesh

2D CAPABILITIES
(LARCNESS)

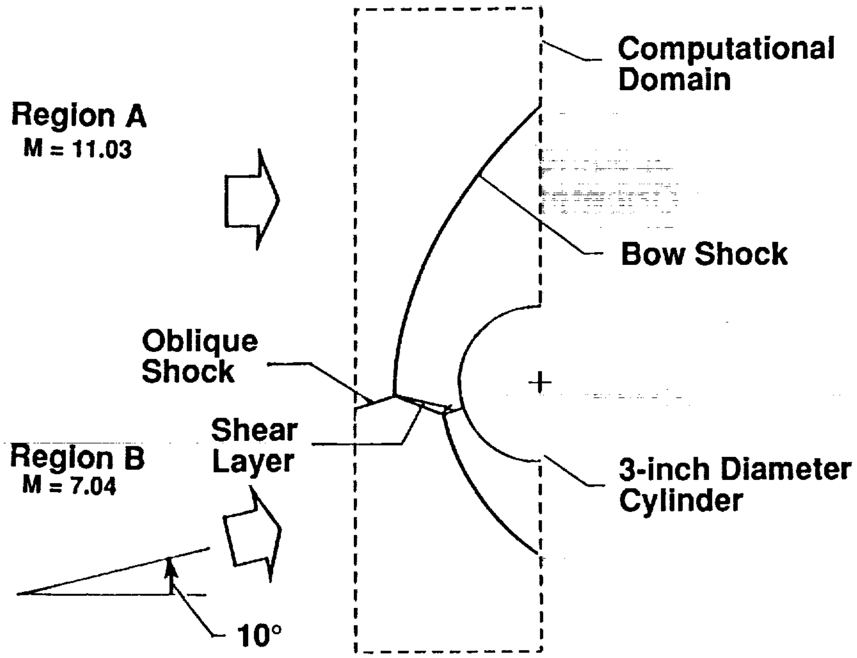
Generation of initial meshes
- Structured near walls
- Unstructured elsewhere

Generation of adapted meshes
- (Remeshing) from previous solution

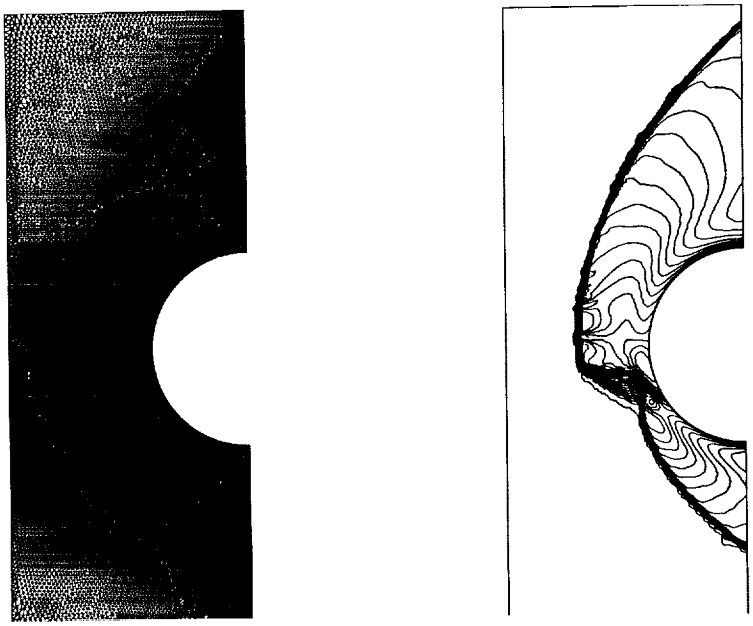
Mesh refinement
- Solution adaptive
- Geometry-based

Mesh movement

2D SHOCK-SHOCK INTERACTION
Schematic



INITIAL MESH

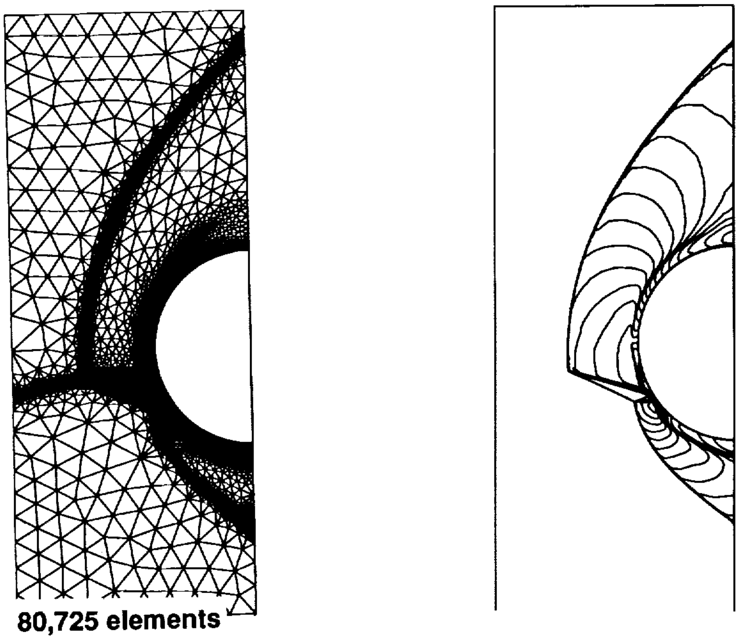


Mesh

U-Velocity Contours

29,499 elements

ADAPTED MESH



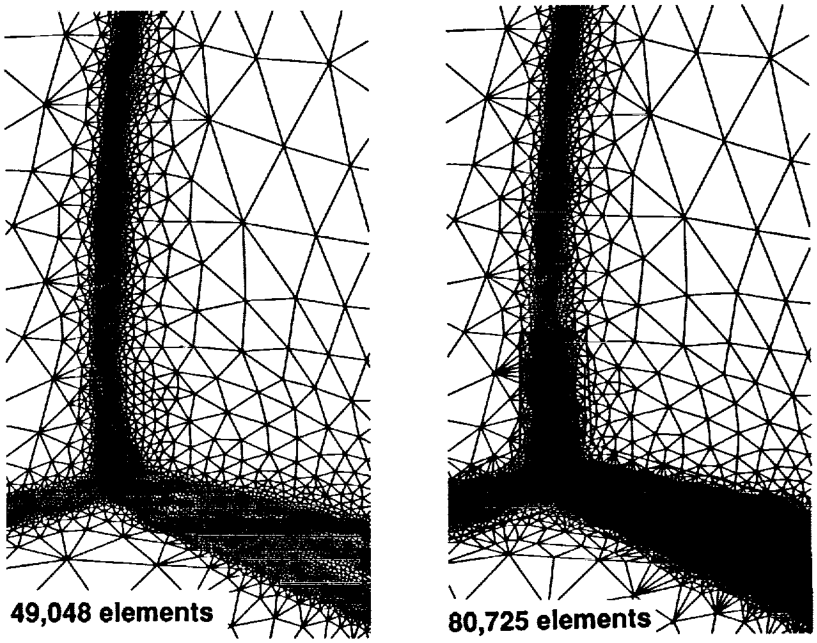
Mesh

U-Velocity Contours

80,725 elements

MESH REFINEMENT

Meshes

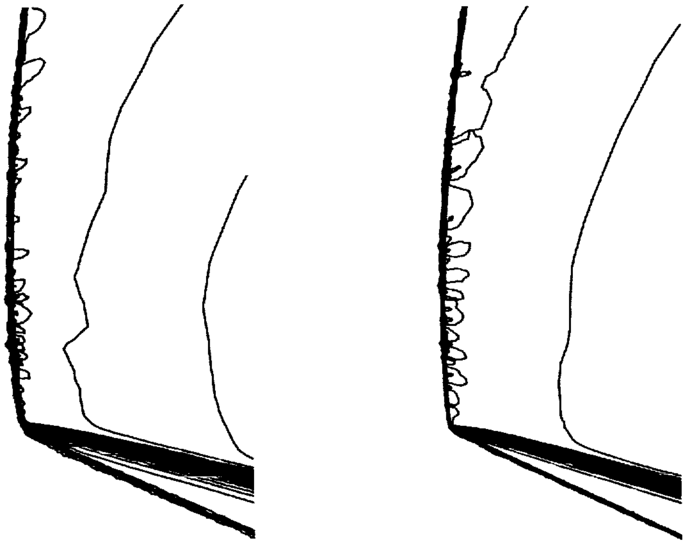


Original

Refined

MESH REFINEMENT

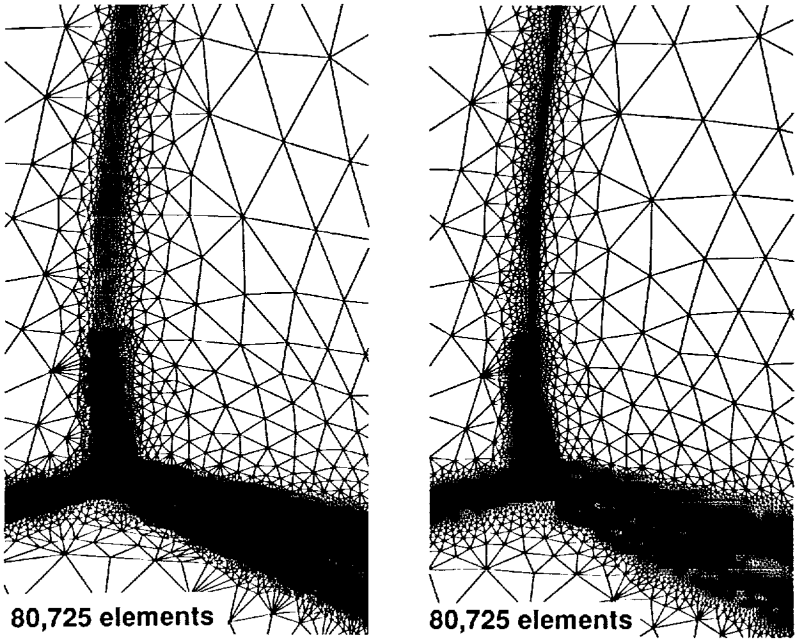
U-Velocity Contours



Original

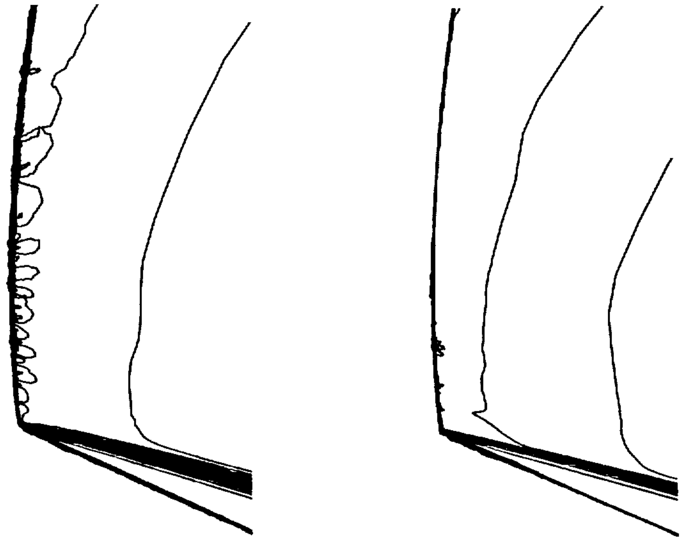
Refined

MESH MOVEMENT
Meshes


Original

Moved

MESH MOVEMENT
U-Velocity Contours


Original

Moved

3D CAPABILITIES
(FELISA)

Developed by Peraire, Morgan, Peiro

3D Unstructured Mesh Generator

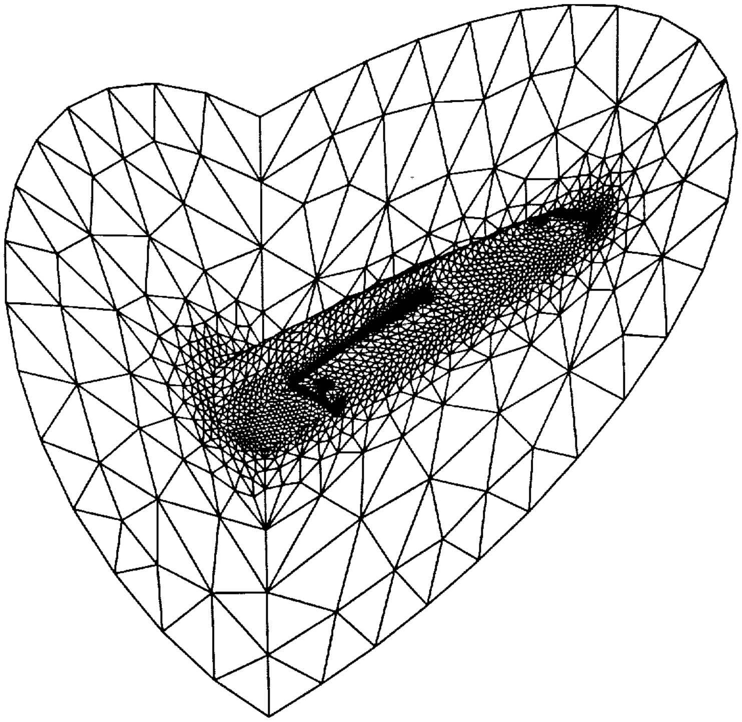
Solver
- Hypersonic Flows
- Unstructured Multigrid
- Matrix Dissipation

Adaption
- Remeshing
- Refinement

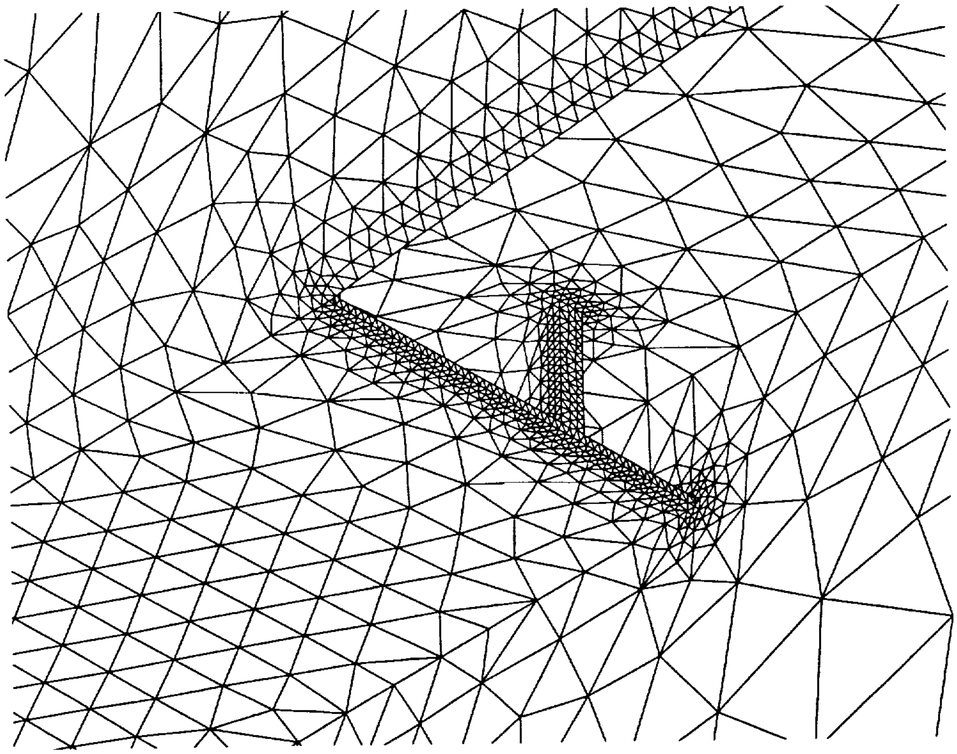
SUMMARY OF MESHES GENERATED BY VARYING SOURCE STRENGTHS

MESH	SURFACE TRIANGLES	VOLUME TETRAHEDRA
1	6,348	39,004
2	24,402	255,853
3	76,254	1,303,666

MESH 1

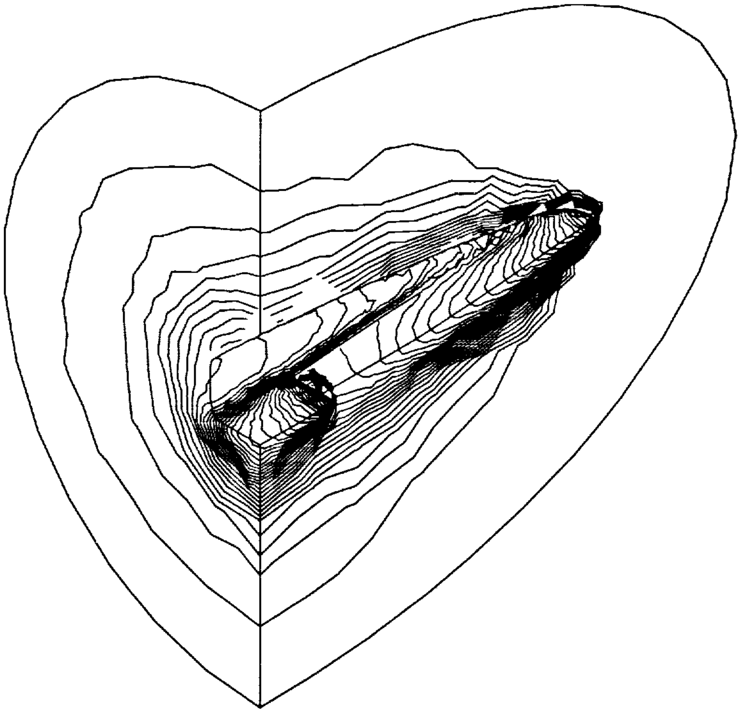


CLOSE-UP OF MESH 1



MACH NUMBER CONTOURS

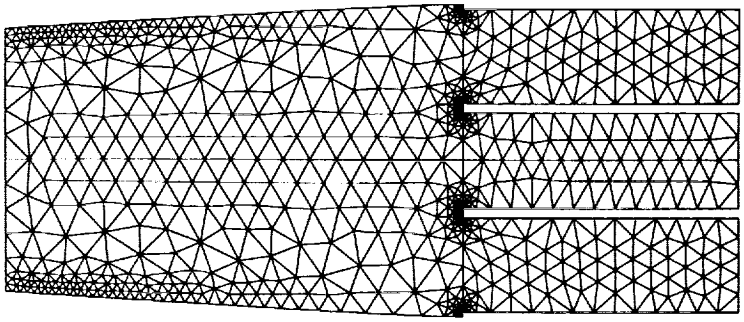
Mesh 1



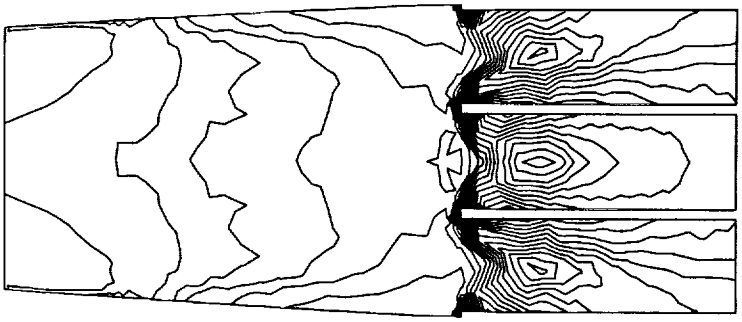
VEHICLE BOTTOM SURFACE

Mesh 1

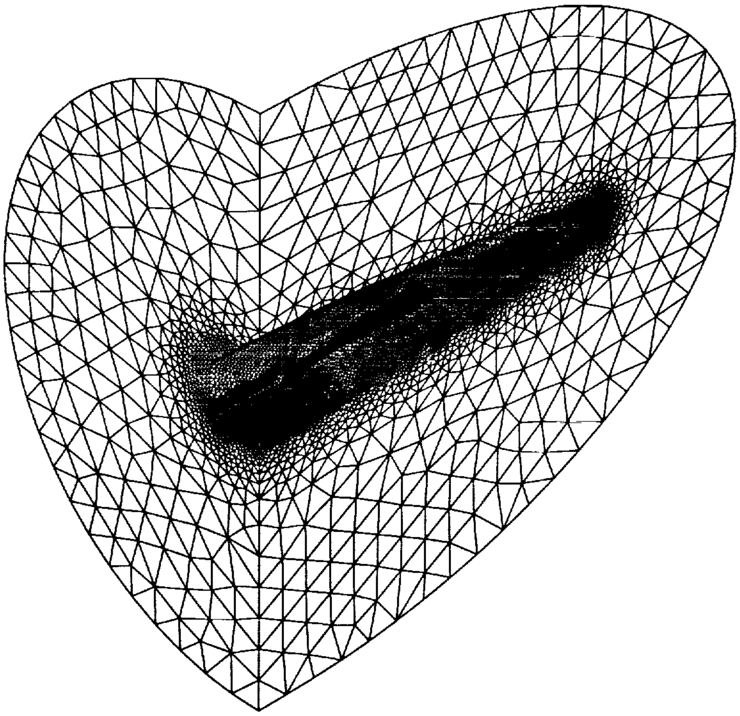
Mesh



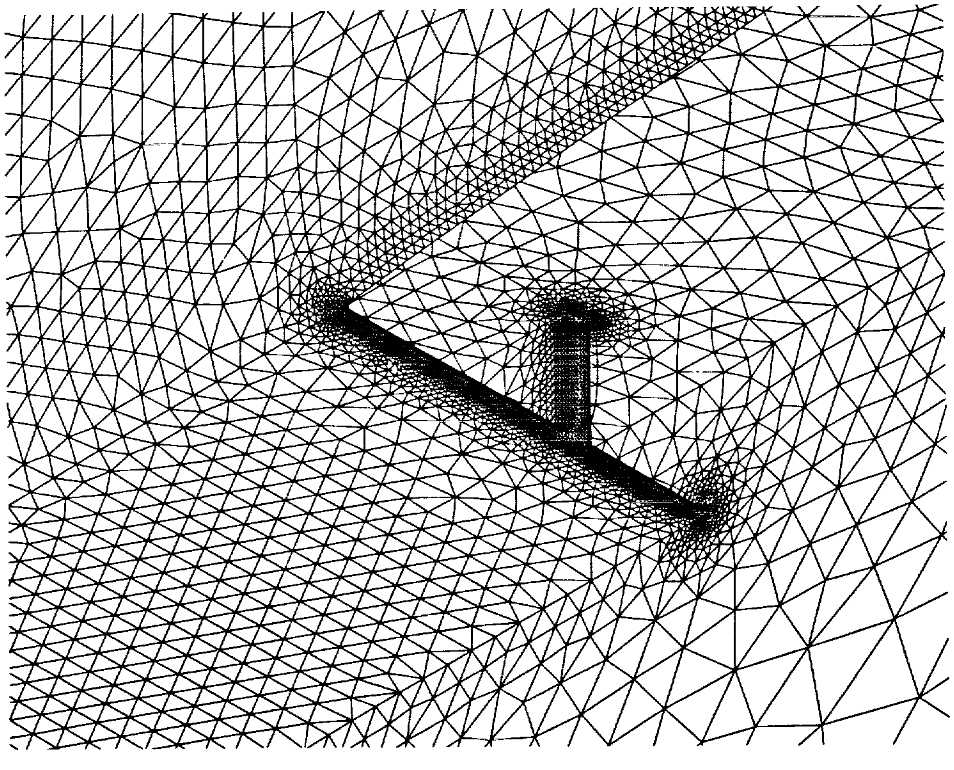
Density Contours



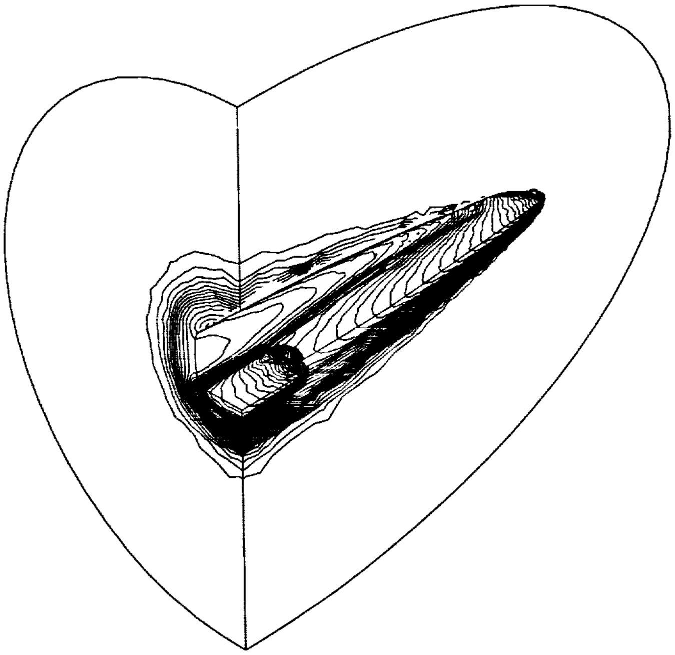
MESH 2



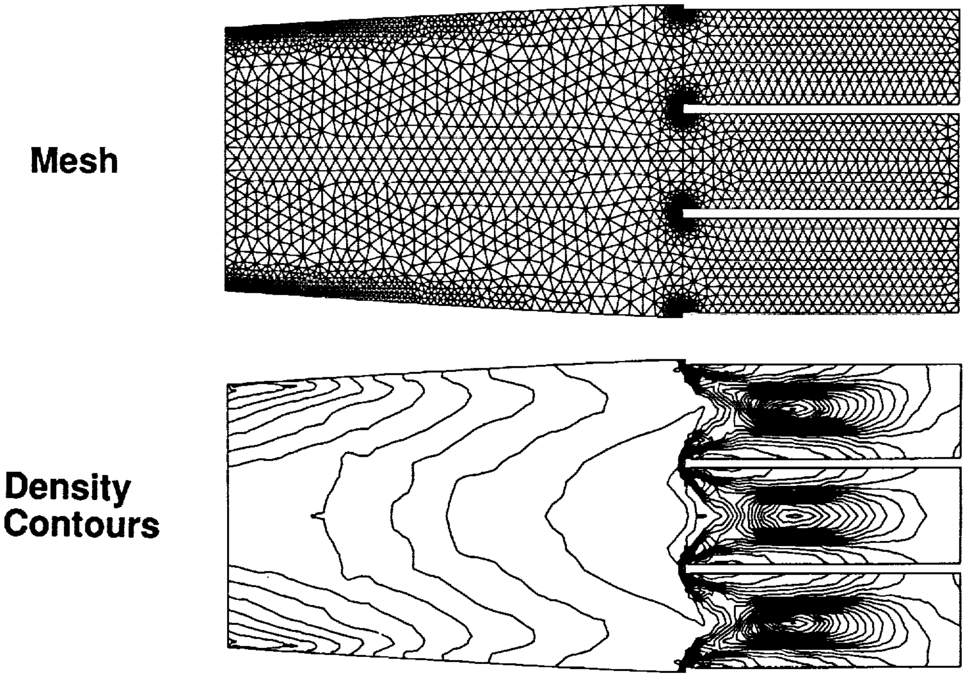
CLOSE-UP OF MESH 2



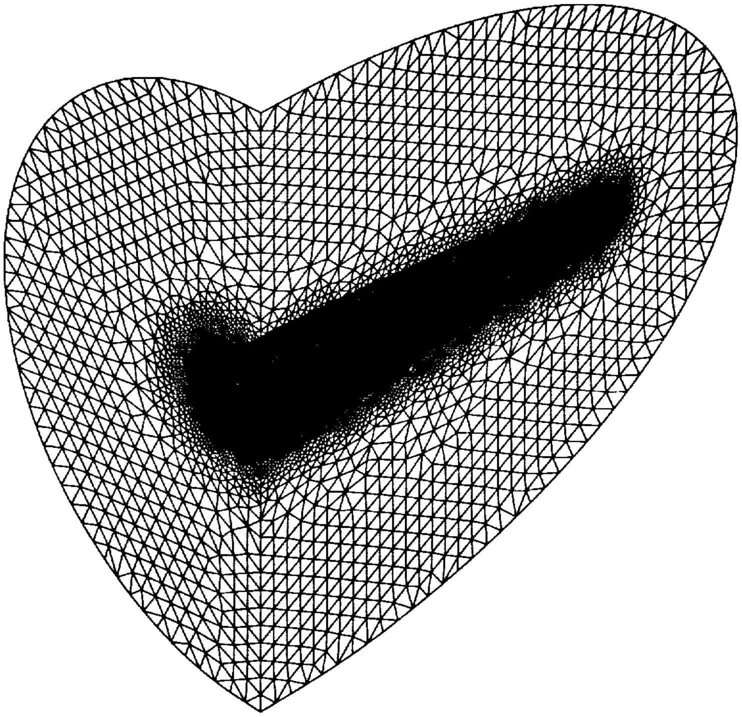
MACH NUMBER CONTOURS
Mesh 2



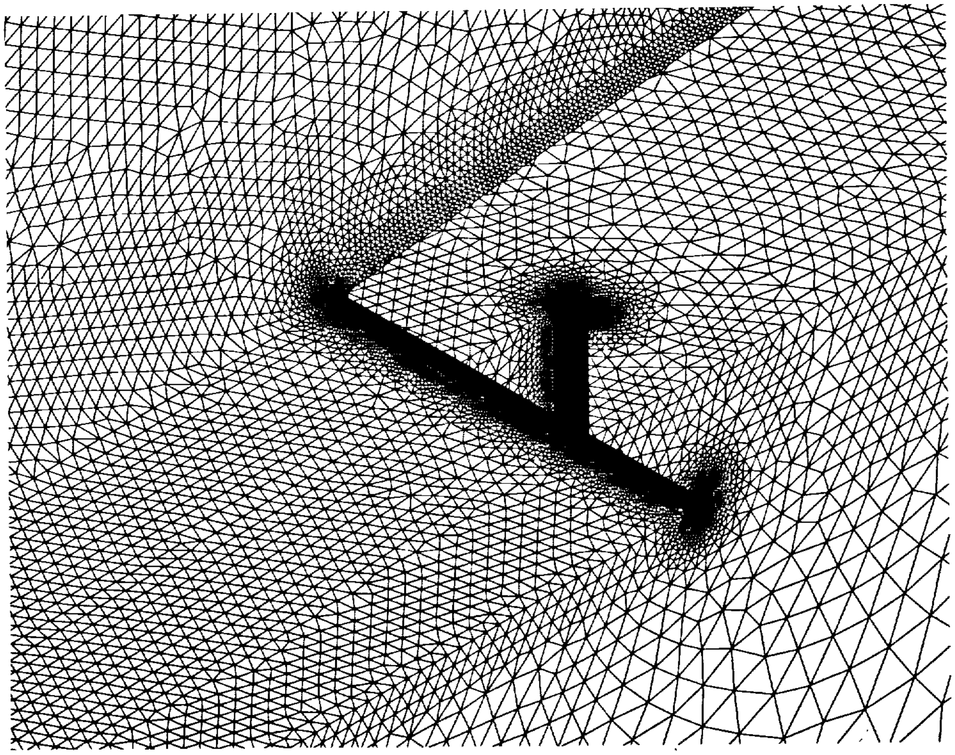
VEHICLE BOTTOM SURFACE
Mesh 2



MESH 3

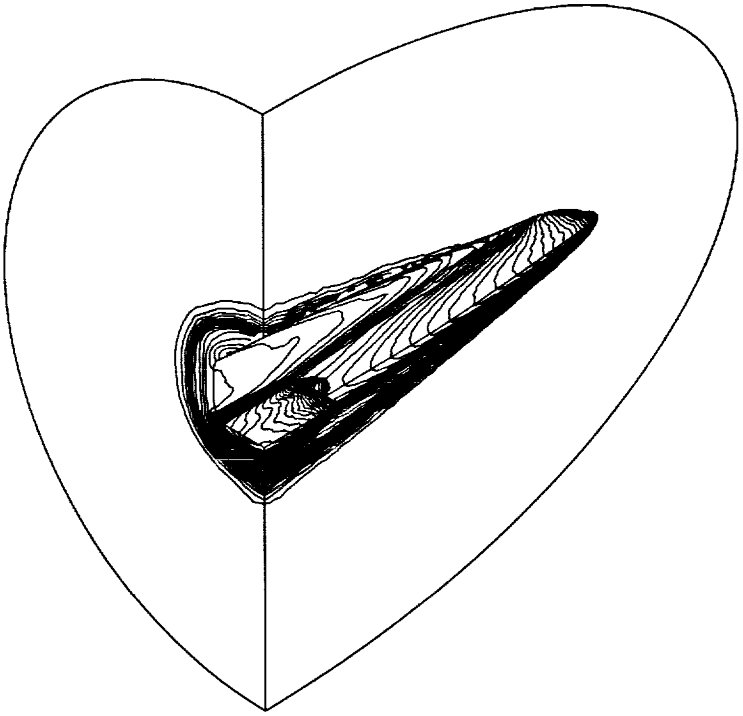


CLOSE-UP OF MESH 3



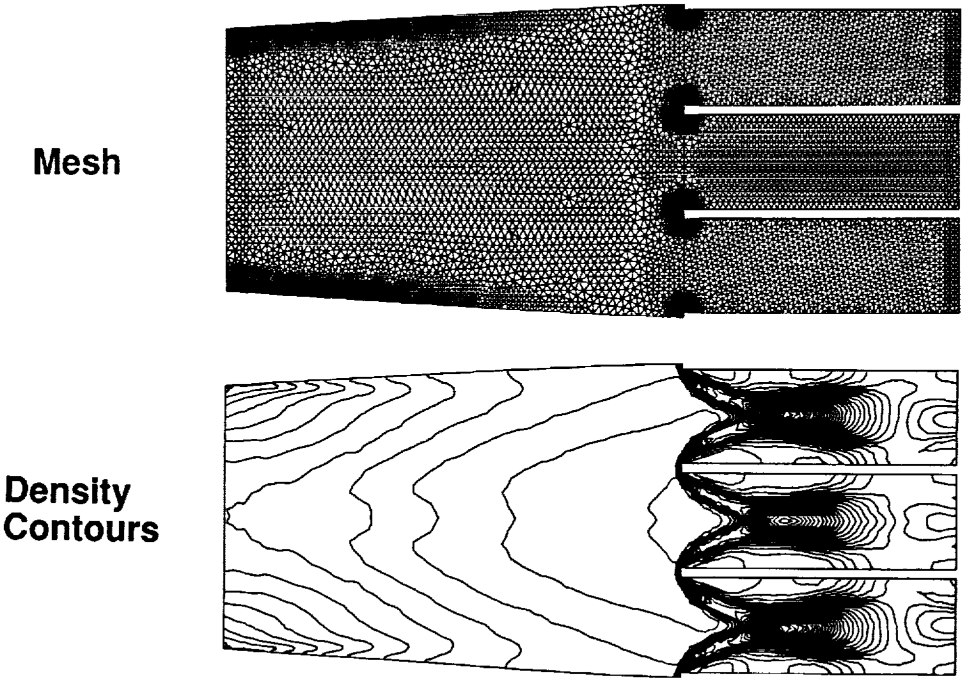
MACH NUMBER CONTOURS

Mesh 3



VEHICLE BOTTOM SURFACE

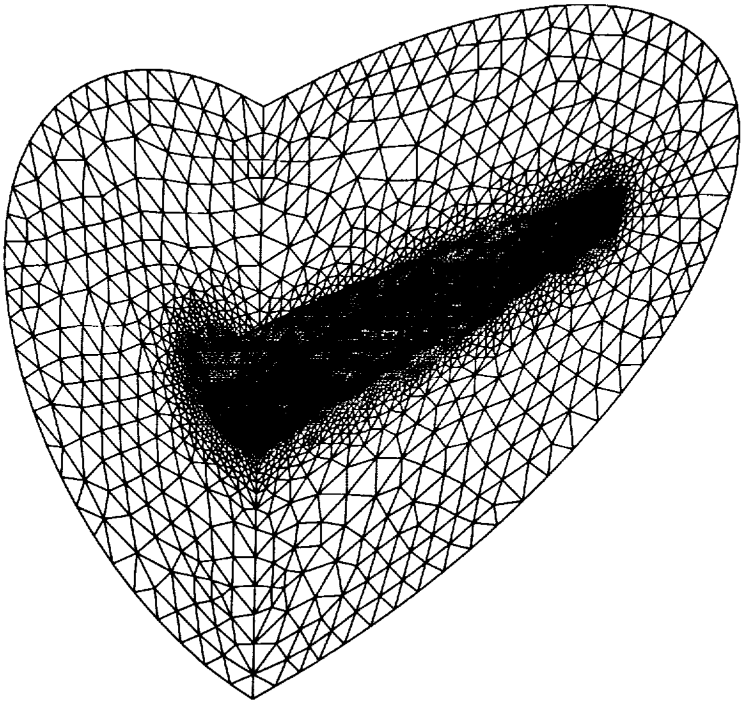
Mesh 3



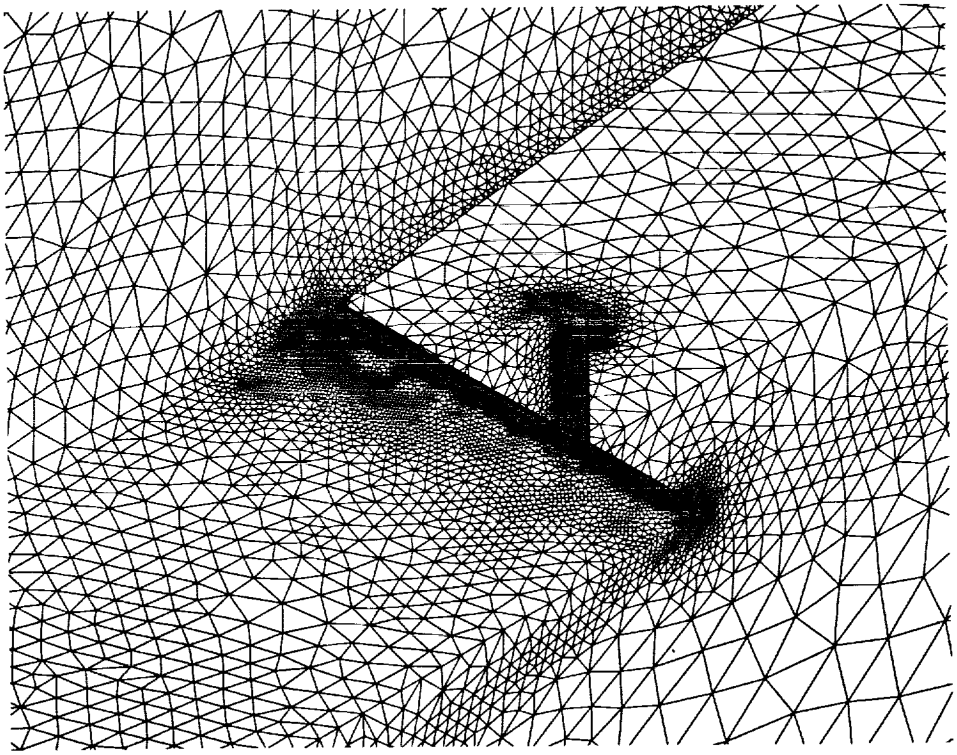
SUMMARY OF ADAPTED MESHES

MESH	SURFACE TRIANGLES	VOLUME TETRAHEDRA
1	41,736	531,610
2	73,930	1,469,105

ADAPTED MESH 1

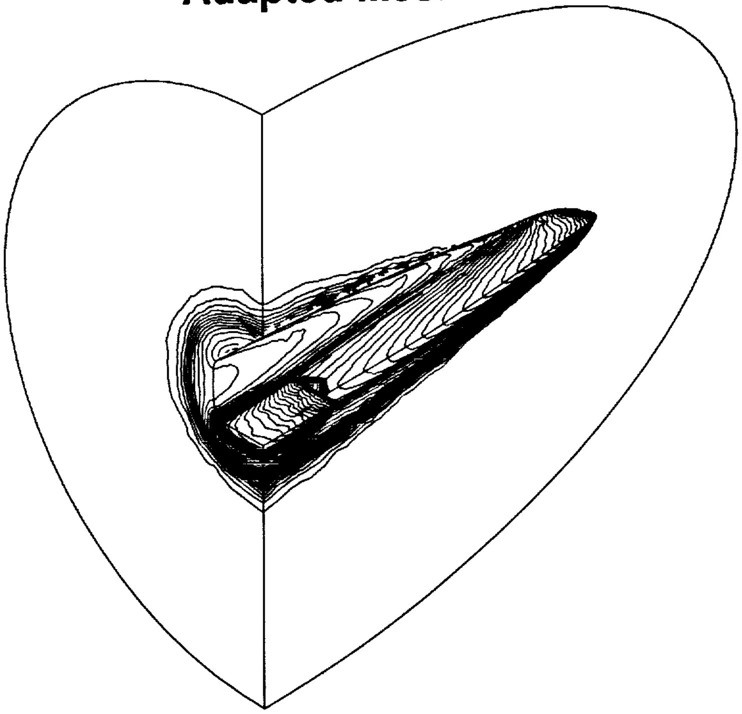


CLOSE-UP OF ADAPTED MESH 1



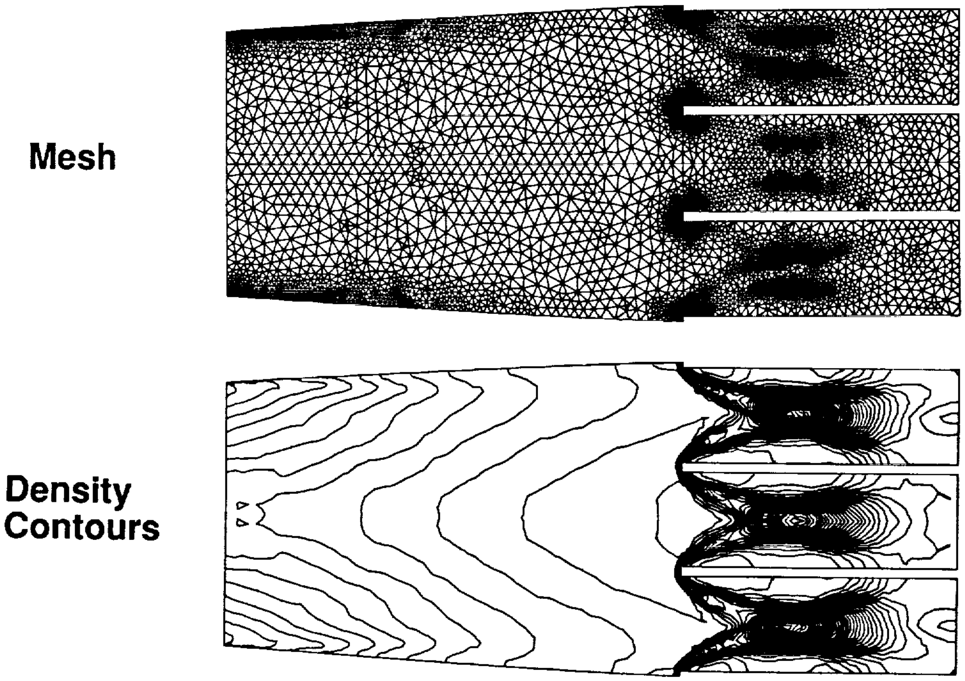
MACH NUMBER CONTOURS

Adapted Mesh 1

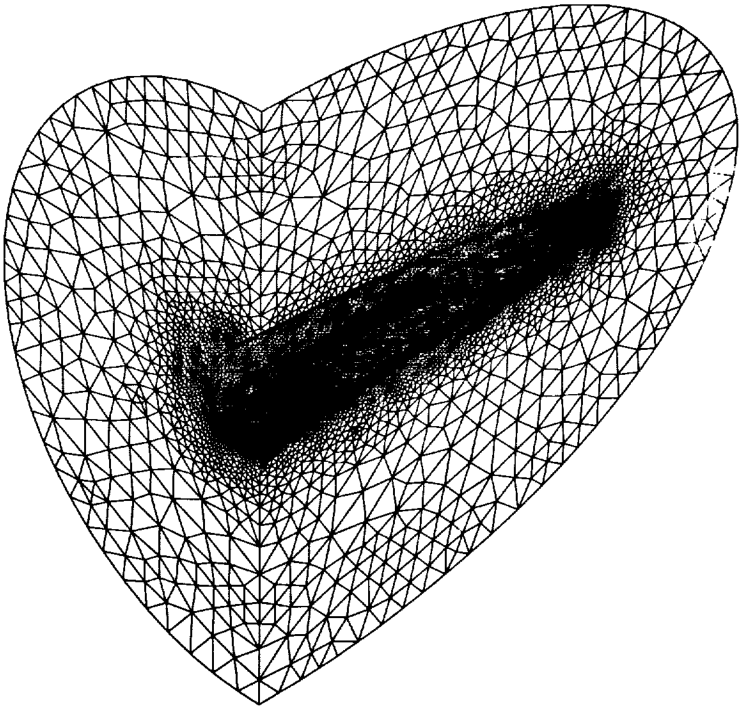


VEHICLE BOTTOM SURFACE

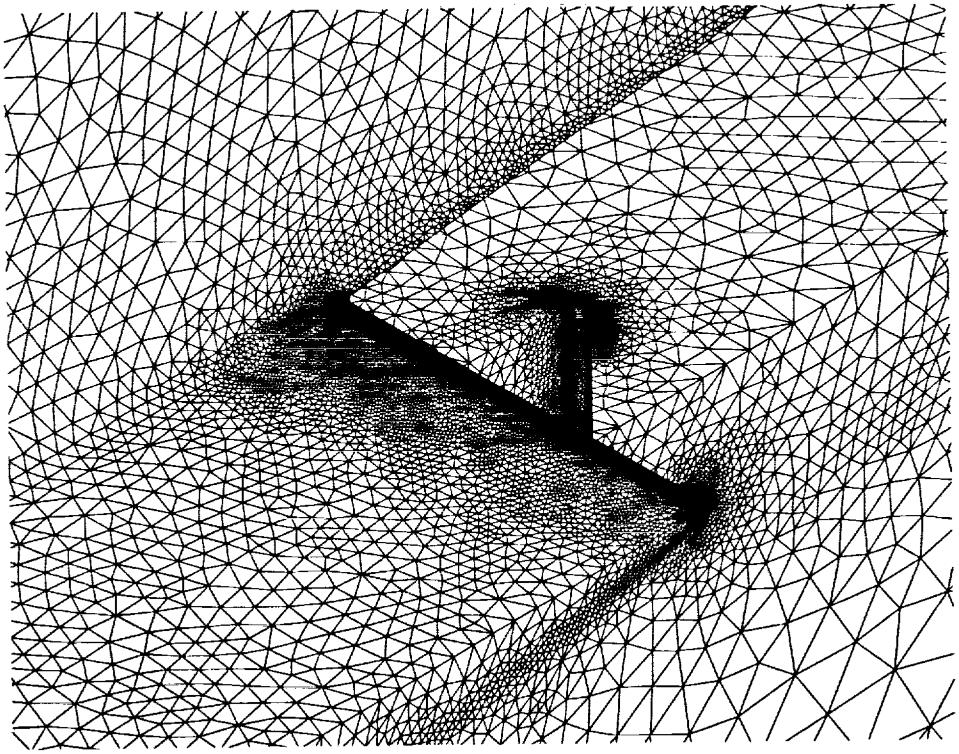
Adapted Mesh 1



ADAPTED MESH 2



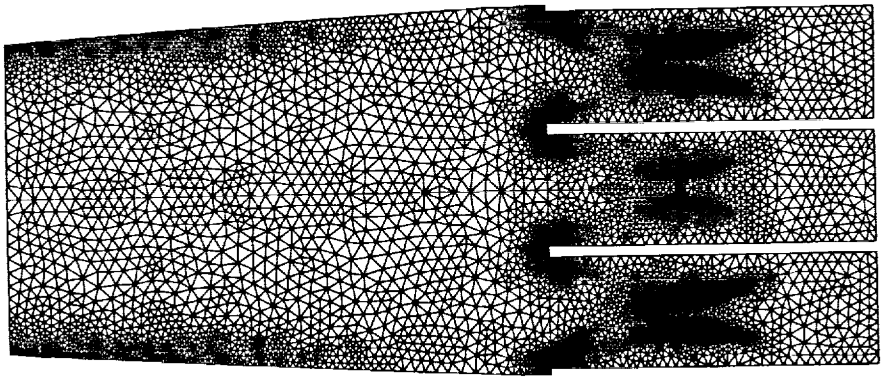
CLOSE-UP OF ADAPTED MESH 2



VEHICLE BOTTOM SURFACE

Adapted Mesh 2

Mesh



Density
Contours

CONCLUSIONS

Adaptive remeshing demonstrated for problems with large number of elements

Though efficient, these schemes exhaust cpu-time, memory and disk-space on current computers

3D meshes with element sizes equivalent to those necessary in 2D would need more than 10 million elements

Current capability is significantly better than what was available only a few years ago

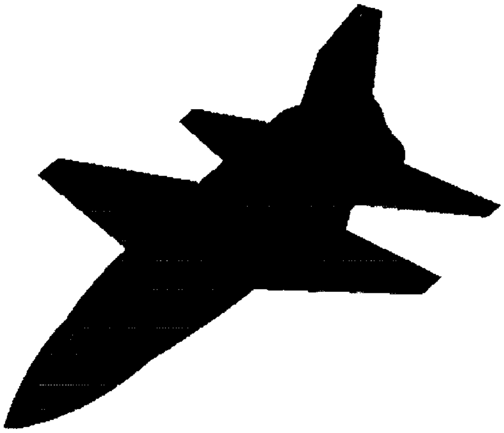
Further improvements in mesh generation, flow solvers and adaptivity still needed

UNSTRUCTURED SURFACE GRID GENERATION

JAMSHID SAMAREH-ABOLHASSANI
COMPUTER SCIENCES CORPORATION

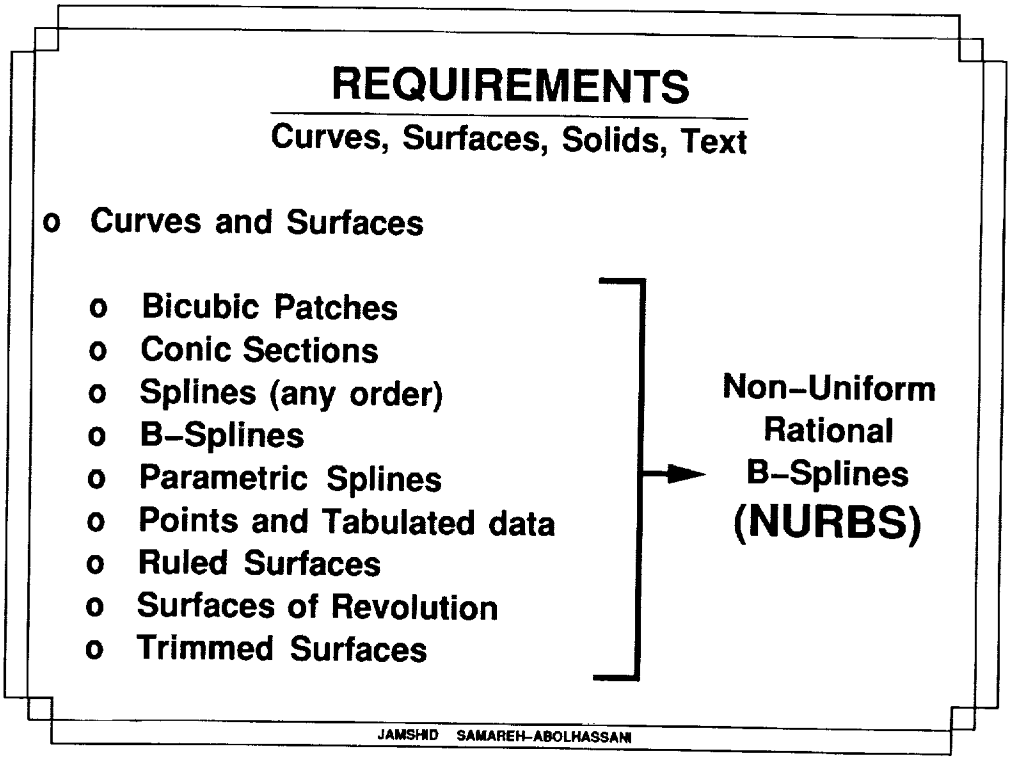
- o INTRODUCTION
- o REQUIREMENTS
- o SURFACE APPROXIMATIONS
- o METHODS
- o GEOLAB EFFORT

JAMSHID SAMAREH-ABOLHASSANI



- o Complex Shapes
- o Turn-Around Time
- o CPU Time
- o Applications
  - o Advancing Front
  - o Prismatic Elements
  - o Delaunay (Steiner Triangulation)

JAMSHID SAMAREH-ABOLHASSANI



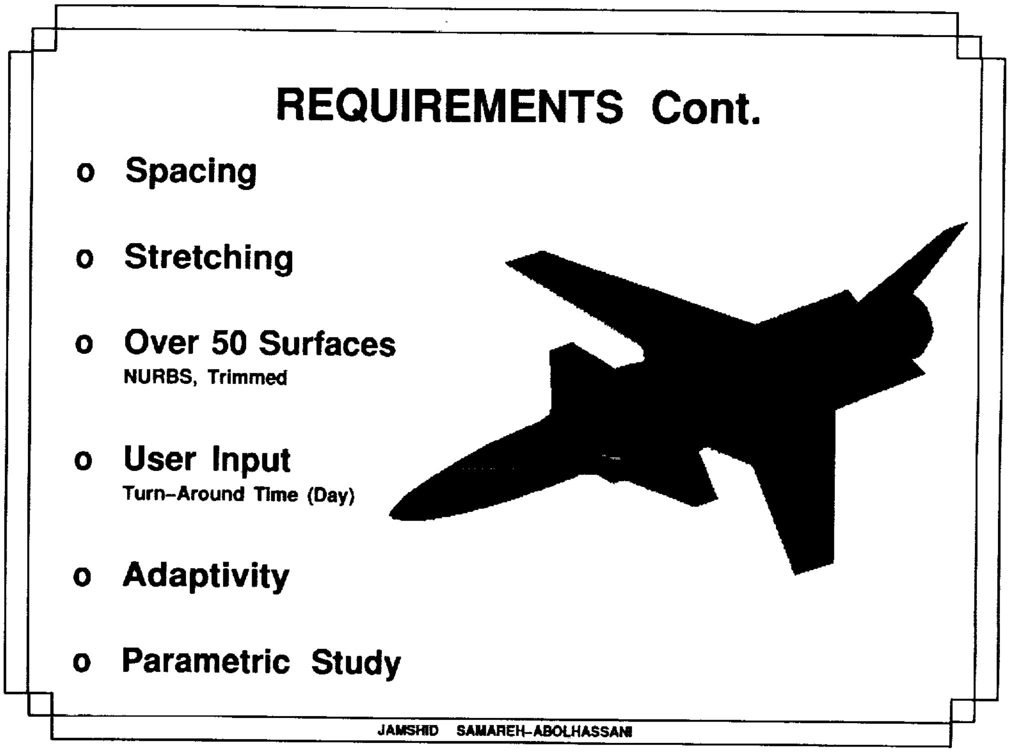
REQUIREMENTS

Curves, Surfaces, Solids, Text

- Curves and Surfaces
  - Bicubic Patches
  - Conic Sections
  - Splines (any order)
  - B-Splines
  - Parametric Splines
  - Points and Tabulated data
  - Ruled Surfaces
  - Surfaces of Revolution
  - Trimmed Surfaces

→ Non-Uniform Rational B-Splines **(NURBS)**

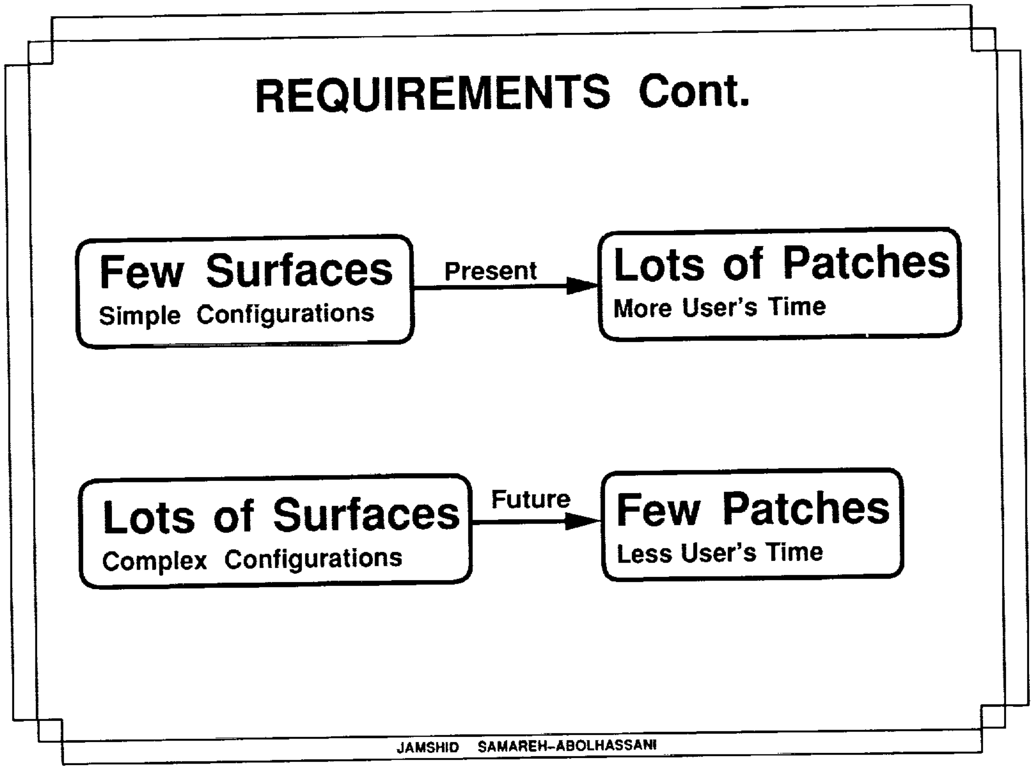
JAMSHID SAMAREH-ABOLHASSANI



REQUIREMENTS Cont.

- Spacing
- Stretching
- Over 50 Surfaces
  NURBS, Trimmed
- User Input
  Turn-Around Time (Day)
- Adaptivity
- Parametric Study

JAMSHID SAMAREH-ABOLHASSANI



REQUIREMENTS Cont.

Few Surfaces
Simple Configurations

Present →

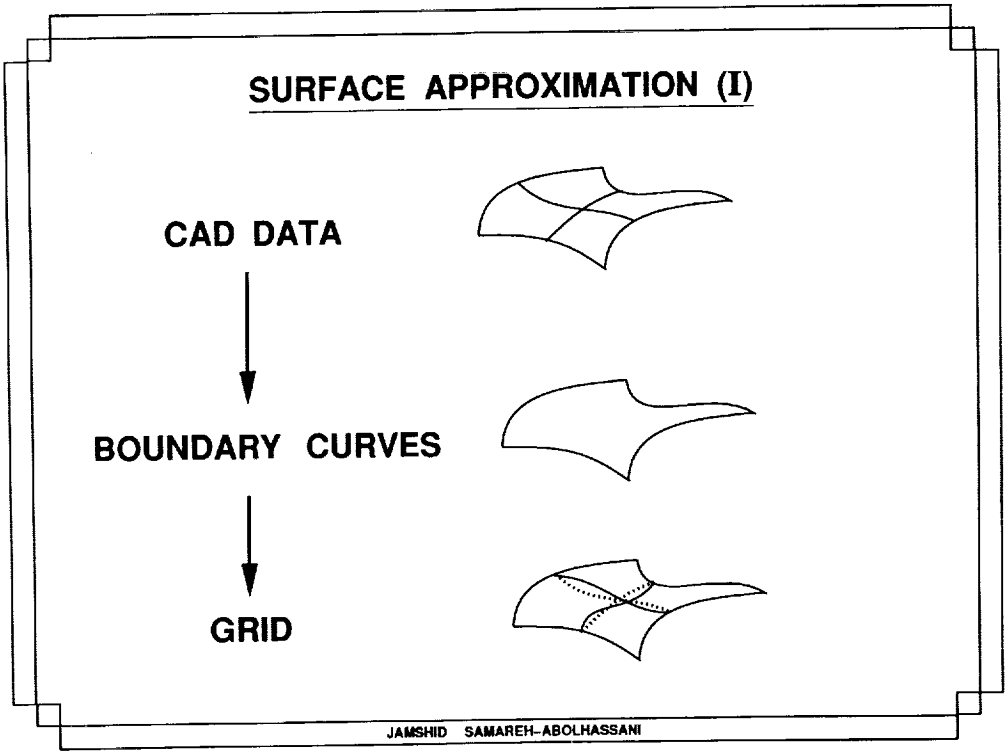
Lots of Patches
More User's Time

Lots of Surfaces
Complex Configurations

Future →

Few Patches
Less User's Time

JAMSHID SAMAREH-ABOLHASSANI



SURFACE APPROXIMATION (I)

CAD DATA

↓

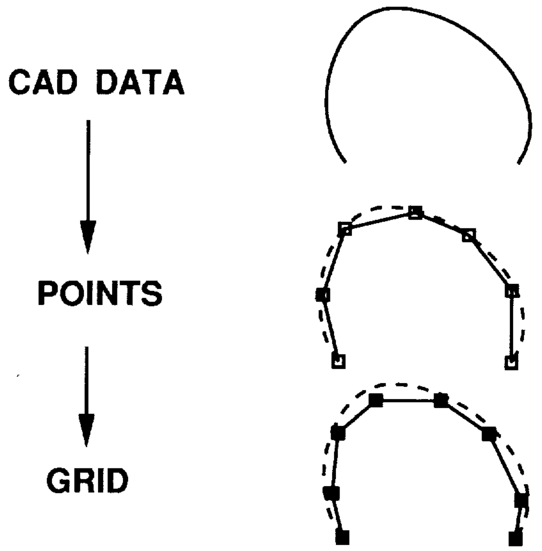
BOUNDARY CURVES

↓

GRID

JAMSHID SAMAREH-ABOLHASSANI

SURFACE APPROXIMATION (II)



CAD DATA

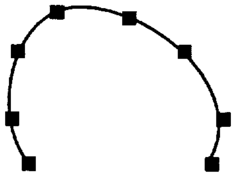
POINTS

GRID

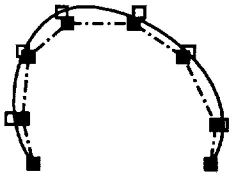
JAMSHID SAMAREH-ABOLHASSANI

EXACT SURFACE REPRESENTATION

Direct Surface Triangulation



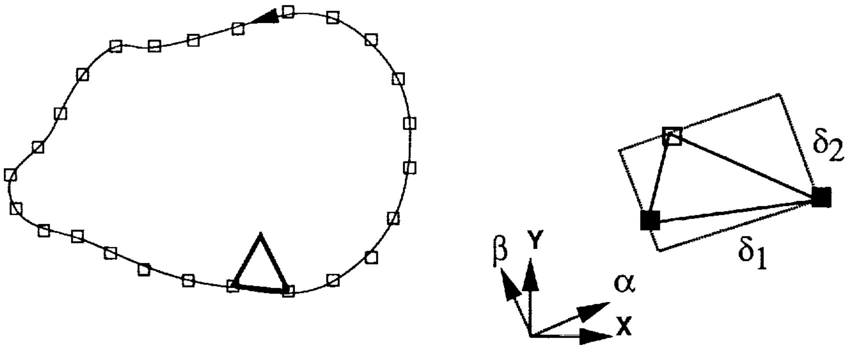
Type I and II + Projection



LANGLEY HAS TWO PROJECTION
CODES FOR STRUCTURED AND
UNSTRUCTURED GRIDS
AIAA 93-3454 (august 1993)
info: jamshid@geosun1.larc.nasa.gov
copy: pkerr@geolab2.larc.nasa.gov

JAMSHID SAMAREH-ABOLHASSANI

Advancing Front



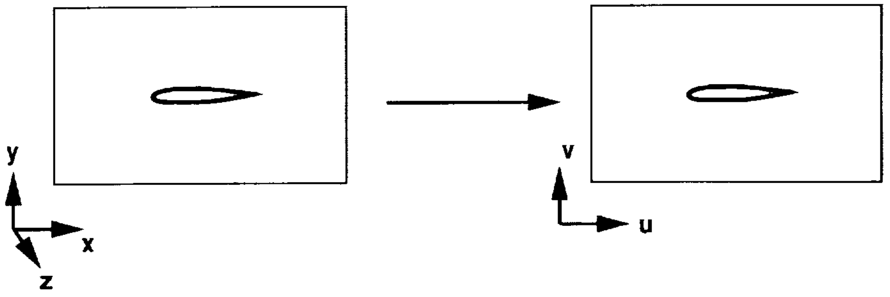
JAMSHID SAMAREH-ABOLHASSANI

METHODS

- o **2D** (Planes, Triangulation is performed in the parameter space)
- o **2 1/2 D** (Triangulation is performed in the Parameter Space)
- o **3D** (Triangulation is performed in the Physical and Parameter Spaces)

JAMSHID SAMAREH-ABOLHASSANI

2D (PLANES)

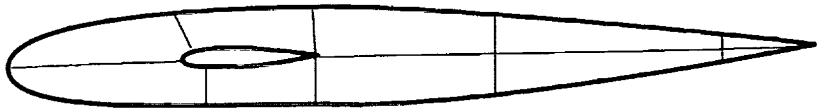


- o Exact
- o No Shearing (Exact shape and size)
- o Speed (0)

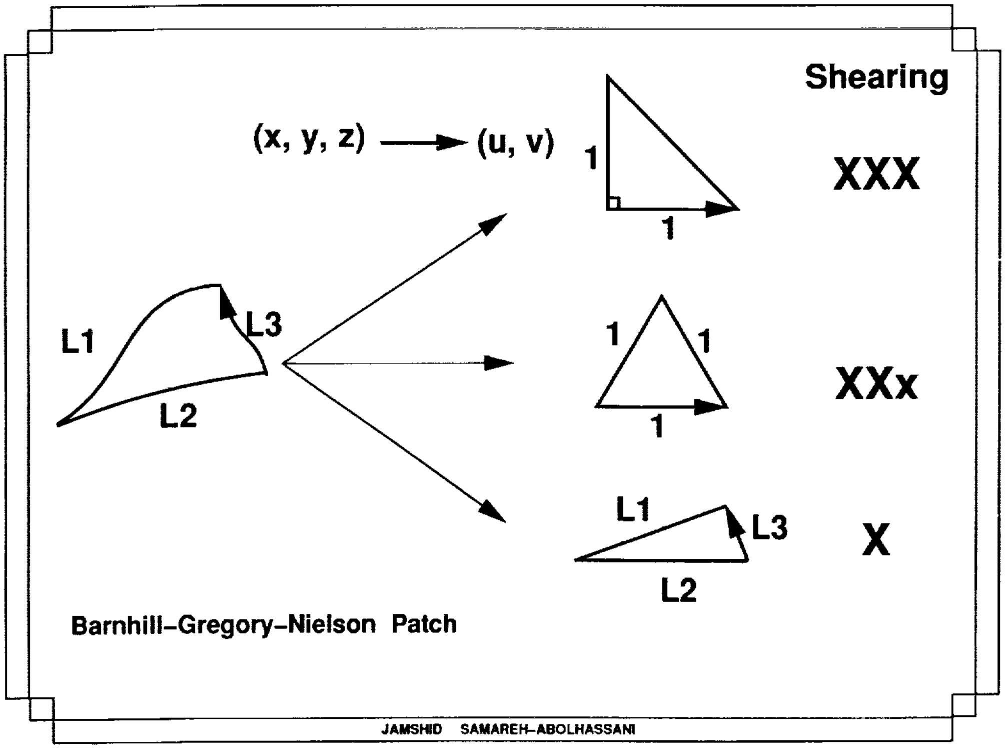
JAMSHID SAMAREH-ABOLHASSANI

2½D Advancing Front

Type I



JAMSHID SAMAREH-ABOLHASSANI



Shearing

$(x, y, z) \longrightarrow (u, v)$

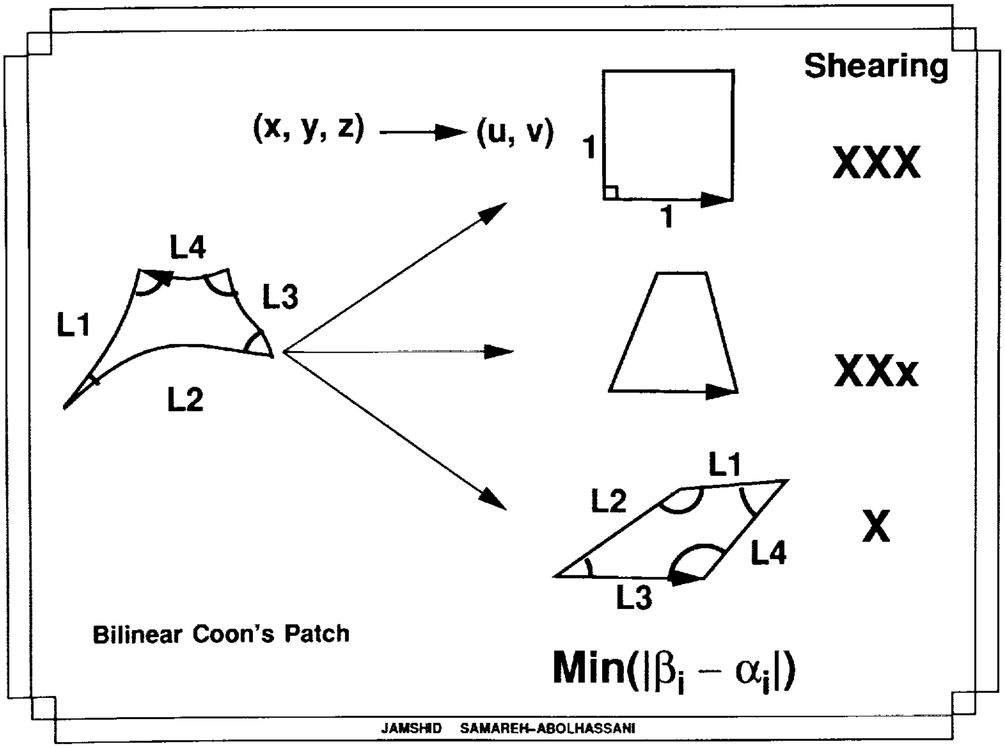
XXX

XXx

X

Barnhill-Gregory-Nielson Patch

JAMSHID SAMAREH-ABOLHASSANI



Shearing

$(x, y, z) \longrightarrow (u, v)$

XXX

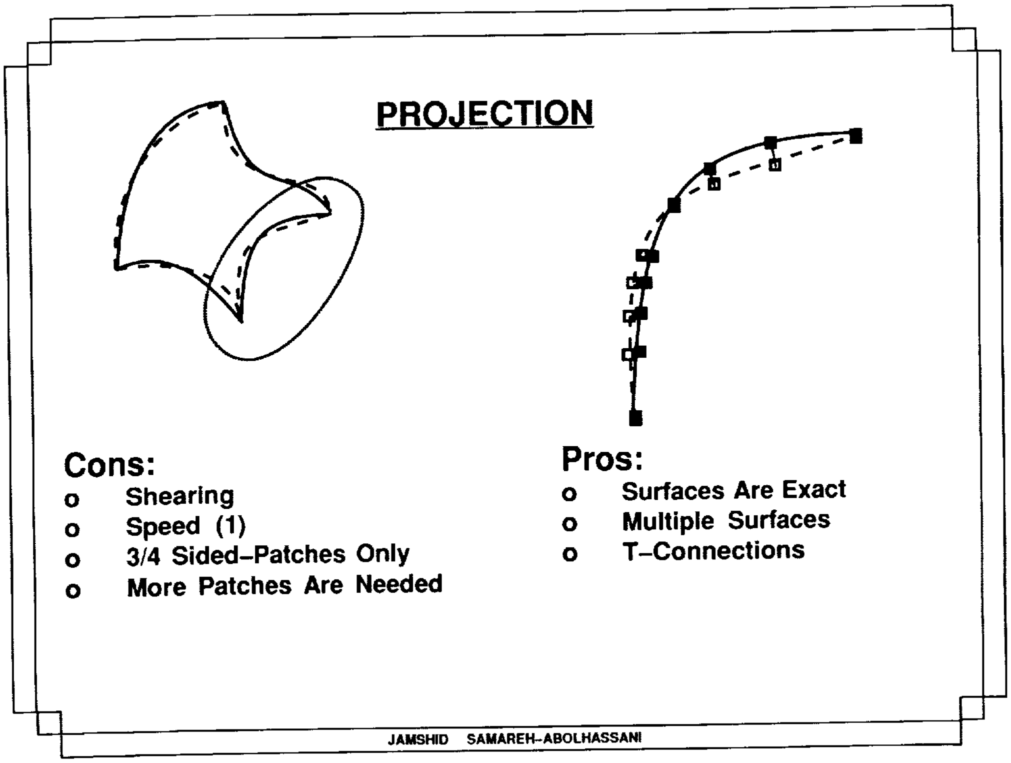
XXx

X

Bilinear Coon's Patch

$\mathrm{Min}(|\beta_i - \alpha_i|)$

JAMSHID SAMAREH-ABOLHASSANI



PROJECTION

Cons:

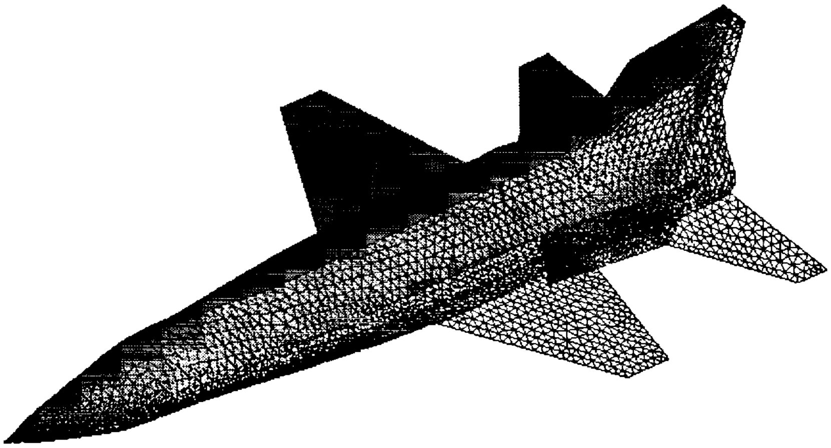
- o Shearing
- o Speed (1)
- o 3/4 Sided-Patches Only
- o More Patches Are Needed

Pros:

- o Surfaces Are Exact
- o Multiple Surfaces
- o T-Connections

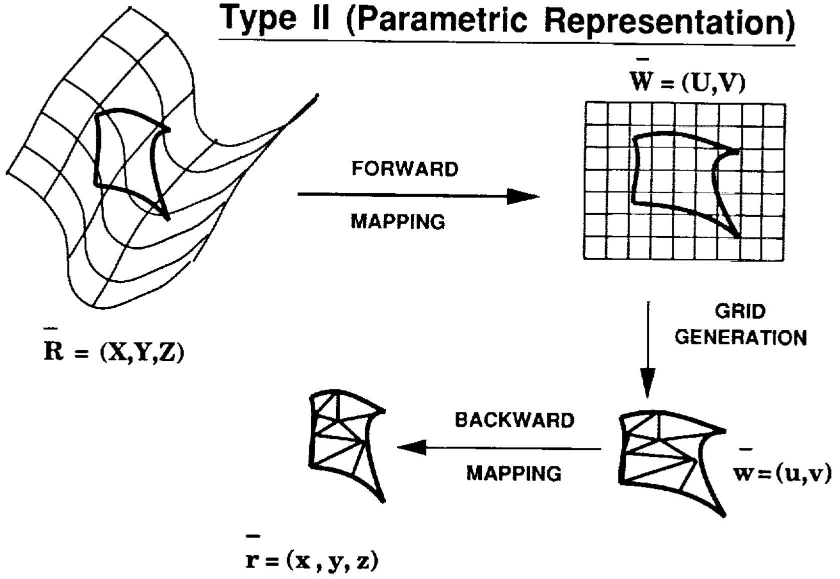
JAMSHID SAMAREH-ABOLHASSANI

X-15



2½D Advancing Front

Type II (Parametric Representation)



$\bar{W} = (U,V)$

FORWARD

MAPPING

$\bar{R} = (X,Y,Z)$

GRID

GENERATION

BACKWARD

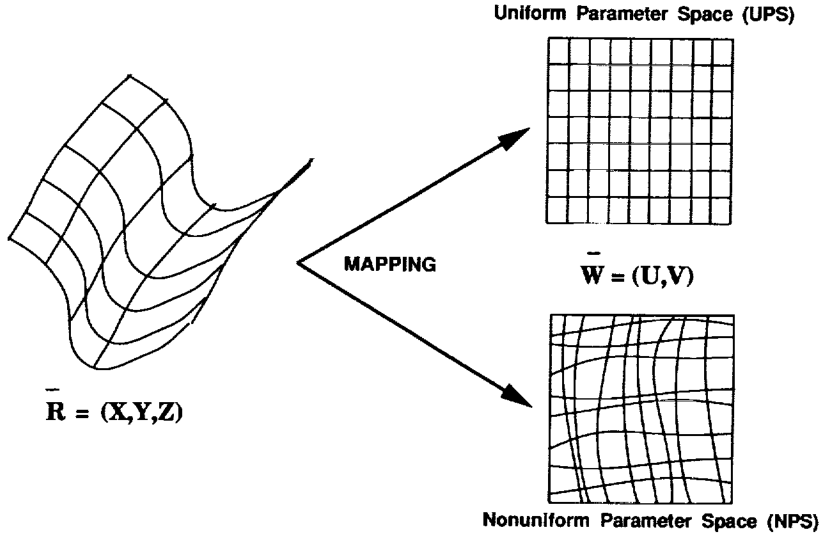
MAPPING

$\bar{w} = (u,v)$

$\bar{r} = (x, y, z)$

JAMSHID SAMAREH-ABOLHASSANI

2½D Advancing Front Cont.



Uniform Parameter Space (UPS)

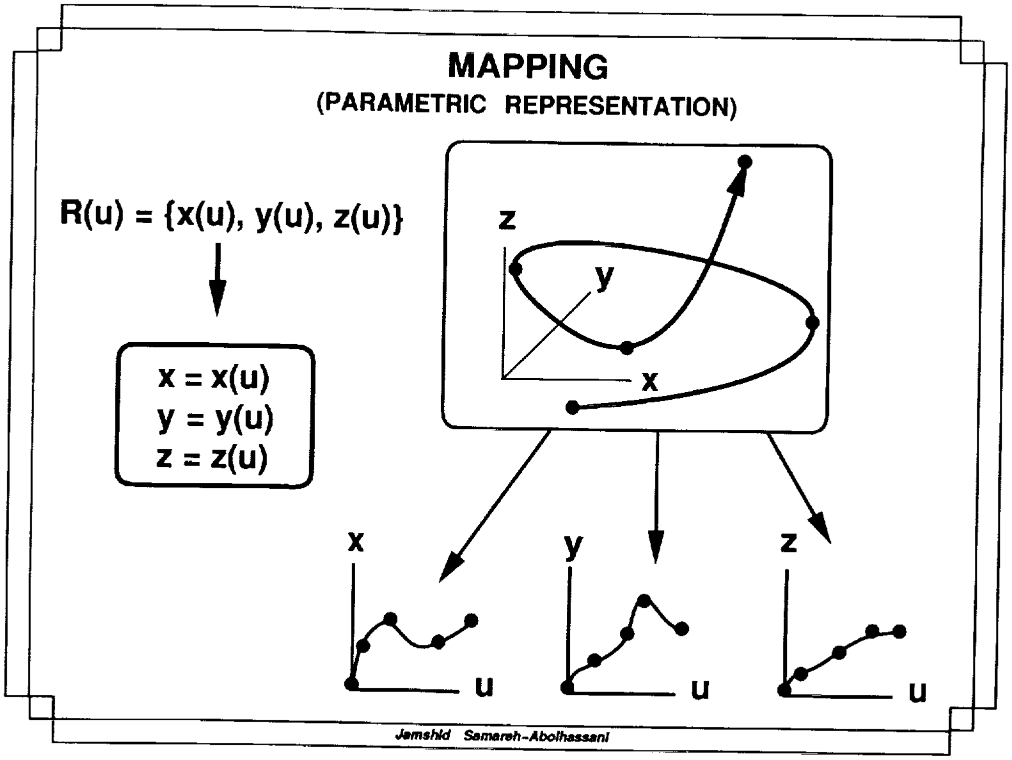
MAPPING

$\bar{W} = (U,V)$

$\bar{R} = (X,Y,Z)$

Nonuniform Parameter Space (NPS)

Jamshid Samareh-Abolhassani



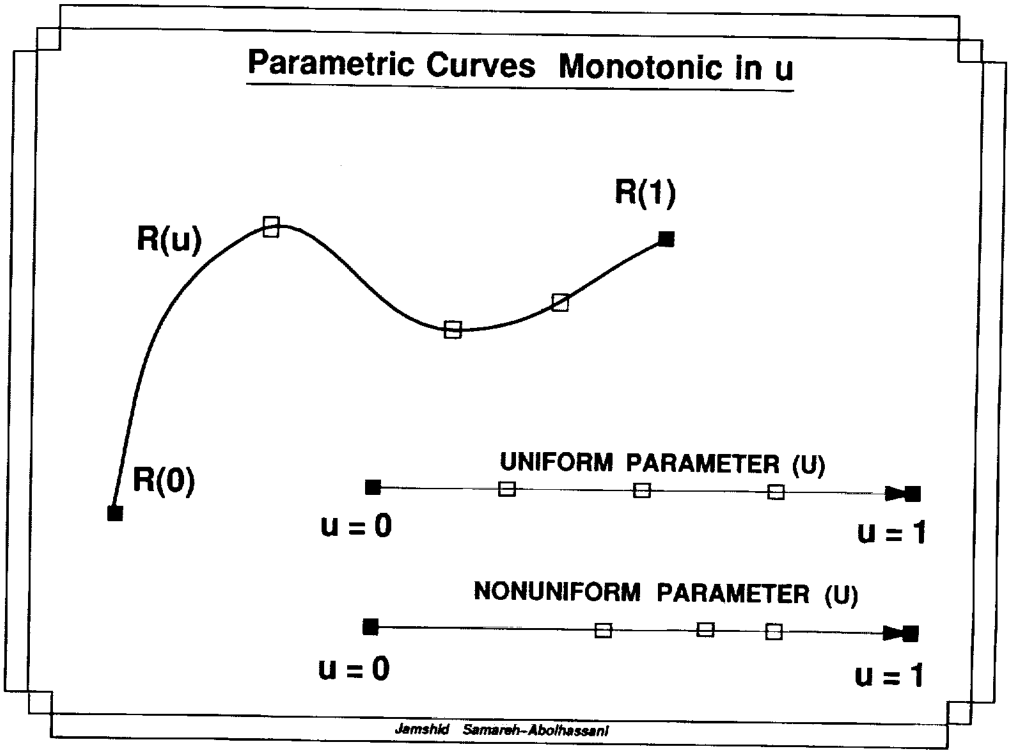
MAPPING

(PARAMETRIC REPRESENTATION)

$R(u) = \{x(u), y(u), z(u)\}$

$x = x(u)$
$y = y(u)$
$z = z(u)$

Jamshid Samareh-Abolhassani

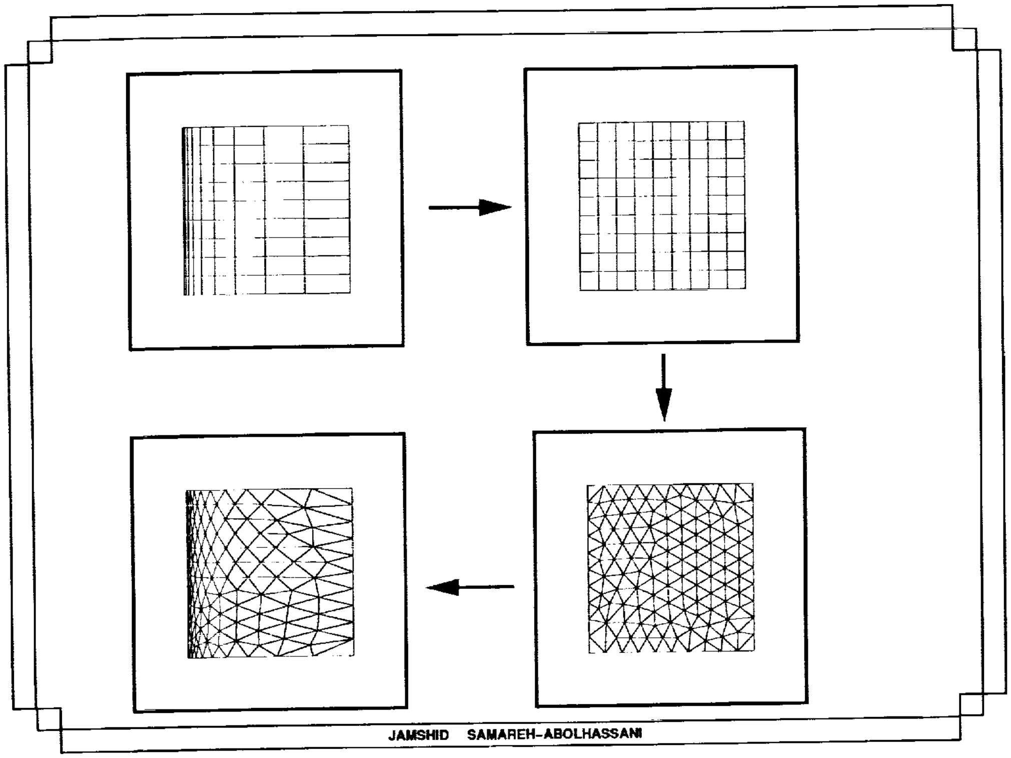


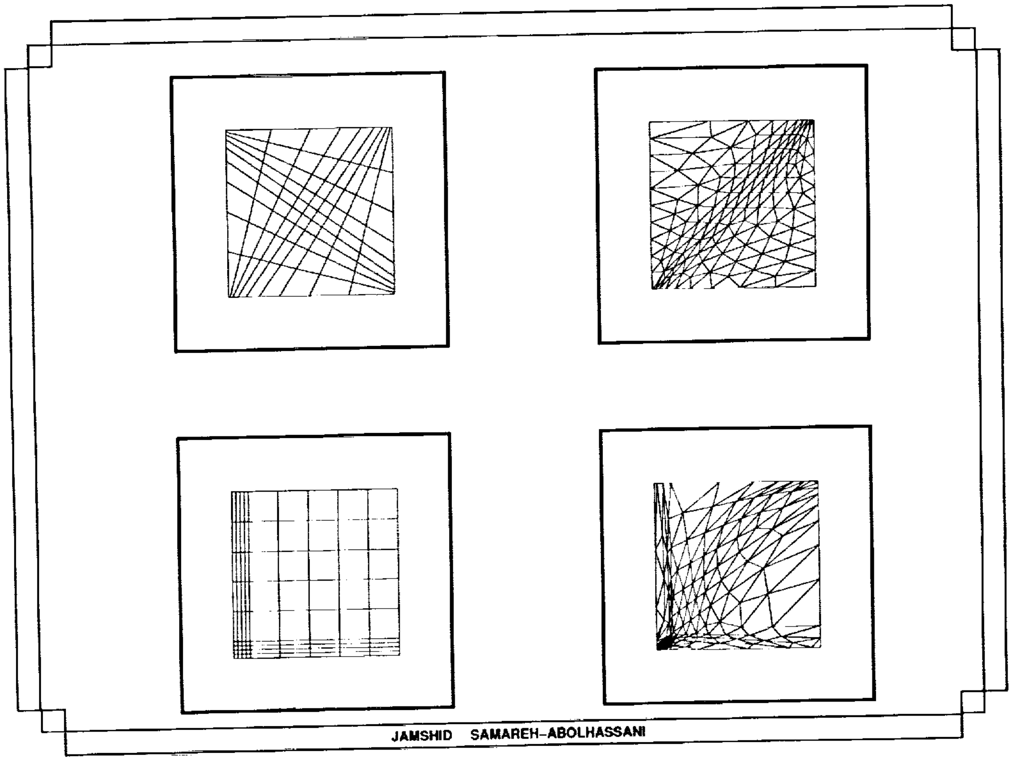
Parametric Curves Monotonic in u

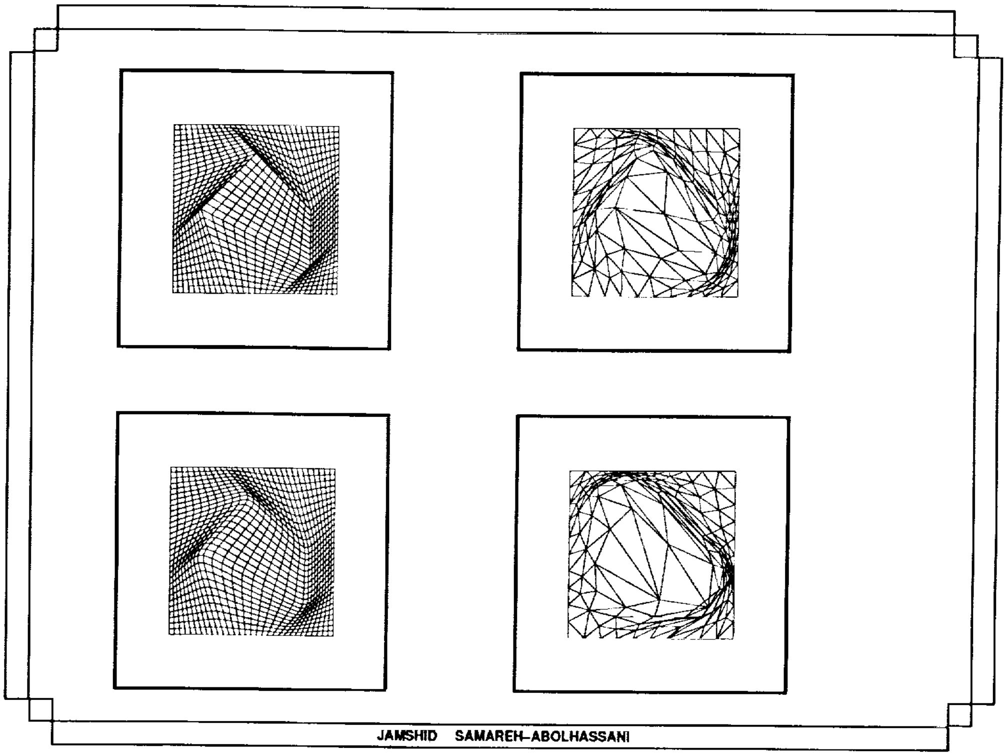
UNIFORM PARAMETER (U)

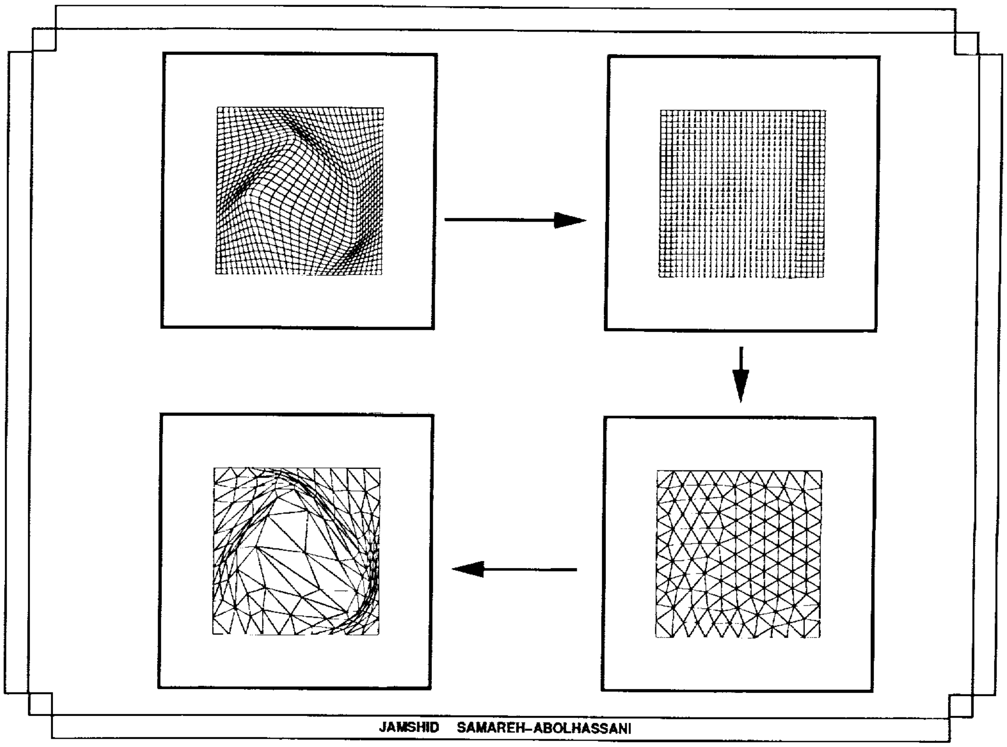
NONUNIFORM PARAMETER (U)

Jamshid Samareh-Abolhassani

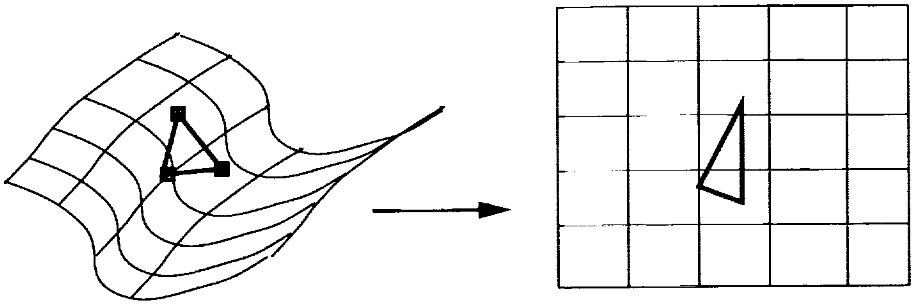

JAMSHID SAMAREH-ABOLHASSANI


JAMSHID SAMAREH-ABOLHASSANI

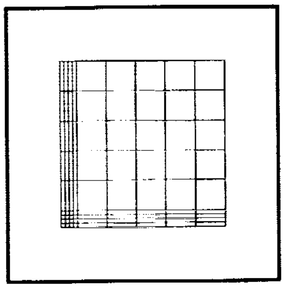

JAMSHID SAMAREH-ABOLHASSANI

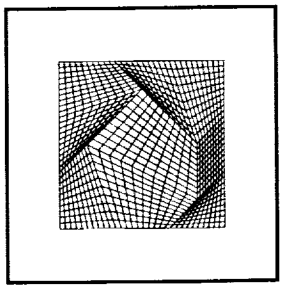

JAMSHID SAMAREH-ABOLHASSANI

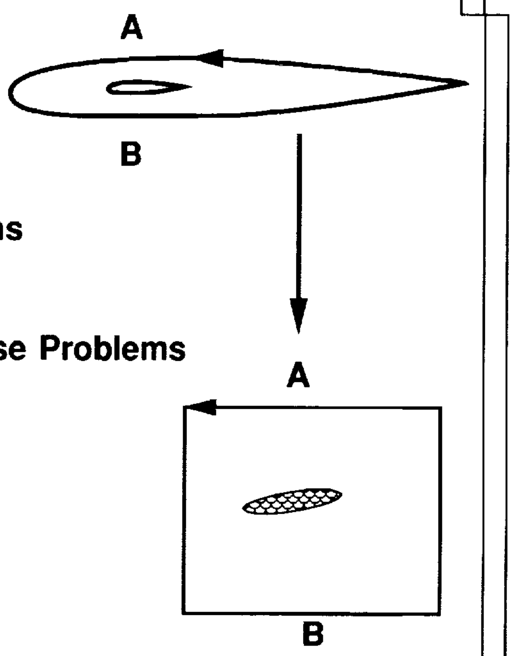
PARAMETER SPACE



$$\boxed{\delta_1, \delta_2, \alpha, \beta} \xrightarrow{\text{Transformation}} \boxed{\delta'_1, \delta'_2, \alpha', \beta'}$$







Cons

- o Metrics Transformations
- o Speed (2)
- o One Surface Only
- o Singularity Could Cause Problems

Pros

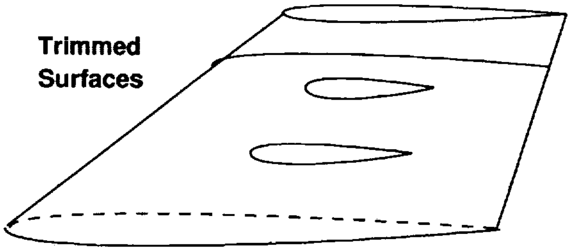
- o Exact Surface
- o N-Sided Patches
- o Trimmed Surfaces
- o Fewer Patches
- o No Shearing (?)

JAMSHID SAMAREH-ABOLHASSANI

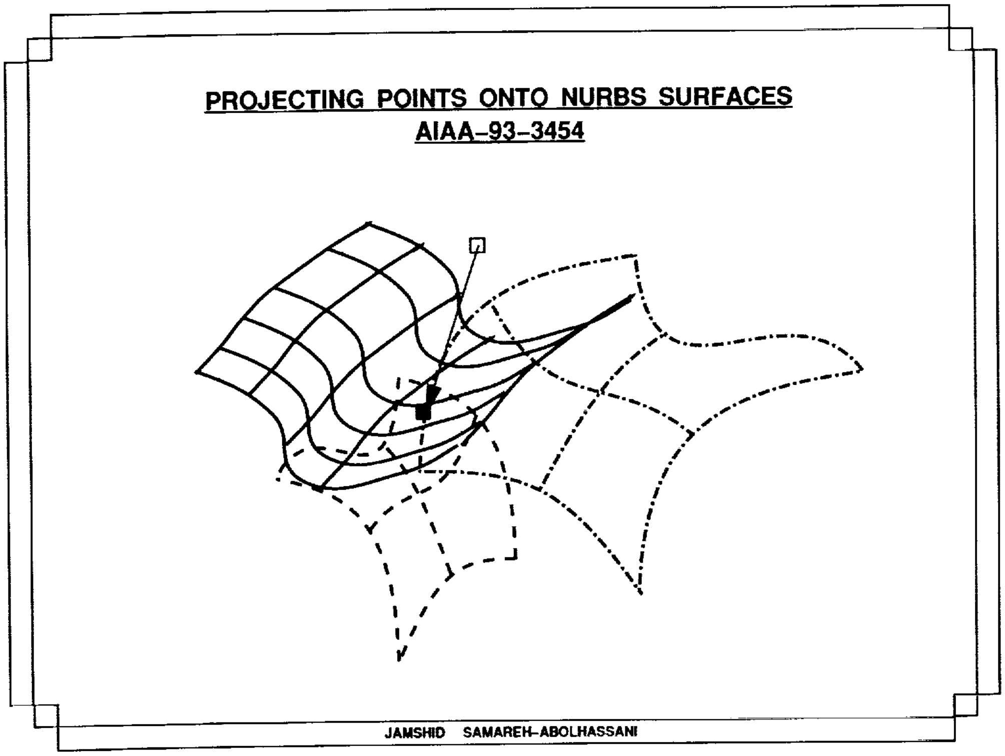
3D Advancing Front

Curved Surfaces

- o Surface Points
- o Surface Normals
- o Loops in 3D

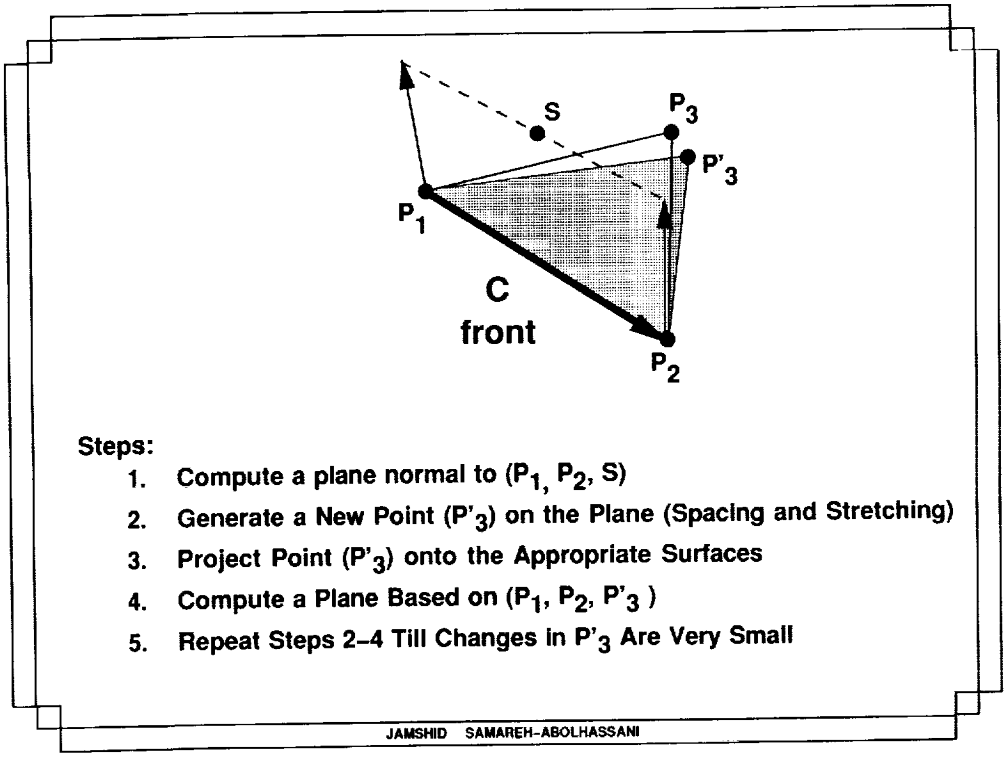


JAMSHID SAMAREH-ABOLHASSANI



PROJECTING POINTS ONTO NURBS SURFACES
AIAA-93-3454

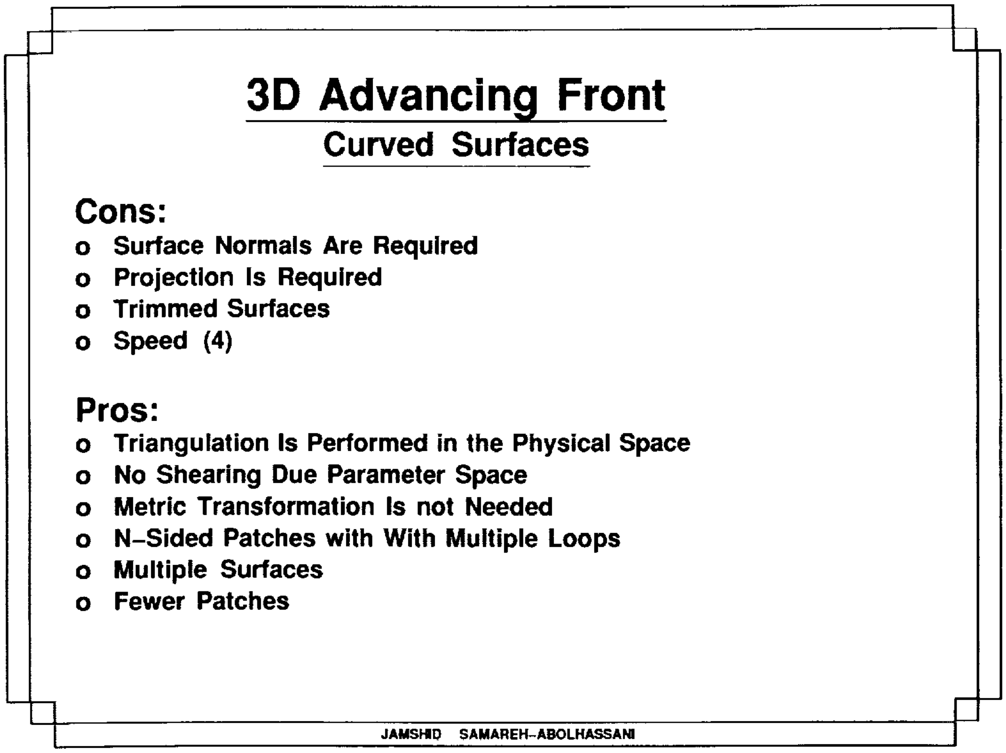
JAMSHID SAMAREH-ABOLHASSANI



Steps:

1. Compute a plane normal to (P_1, P_2, S)
2. Generate a New Point (P'_3) on the Plane (Spacing and Stretching)
3. Project Point (P'_3) onto the Appropriate Surfaces
4. Compute a Plane Based on (P_1, P_2, P'_3)
5. Repeat Steps 2-4 Till Changes in P'_3 Are Very Small

JAMSHID SAMAREH-ABOLHASSANI



3D Advancing Front

Curved Surfaces

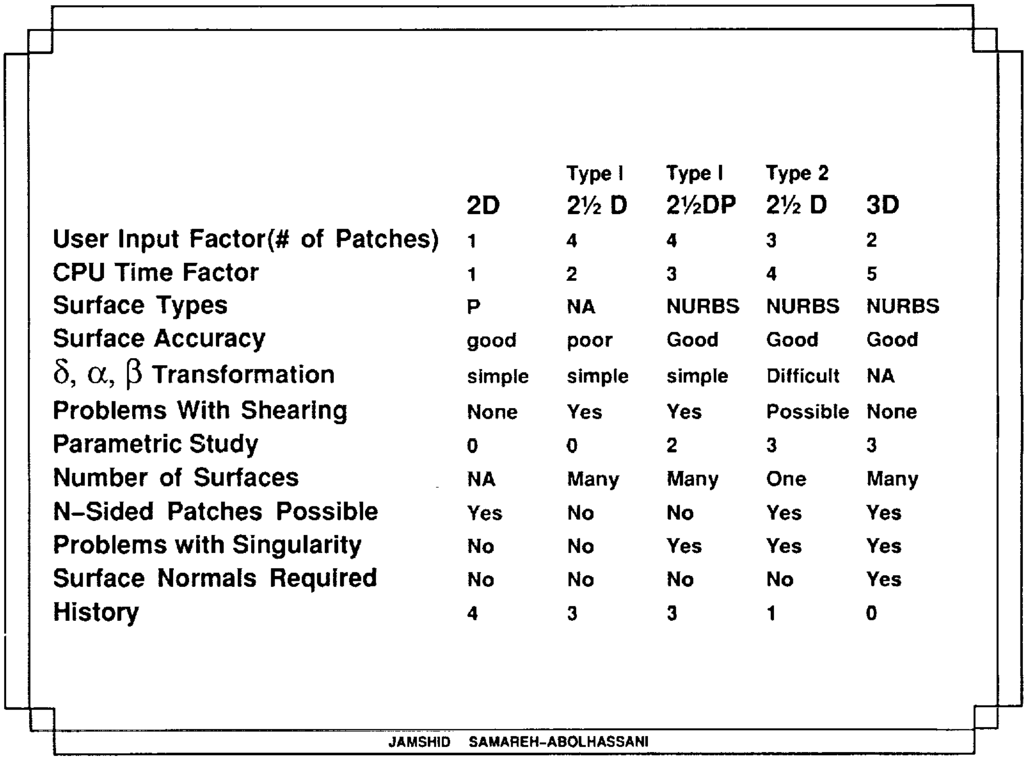
Cons:

- o Surface Normals Are Required
- o Projection Is Required
- o Trimmed Surfaces
- o Speed (4)

Pros:

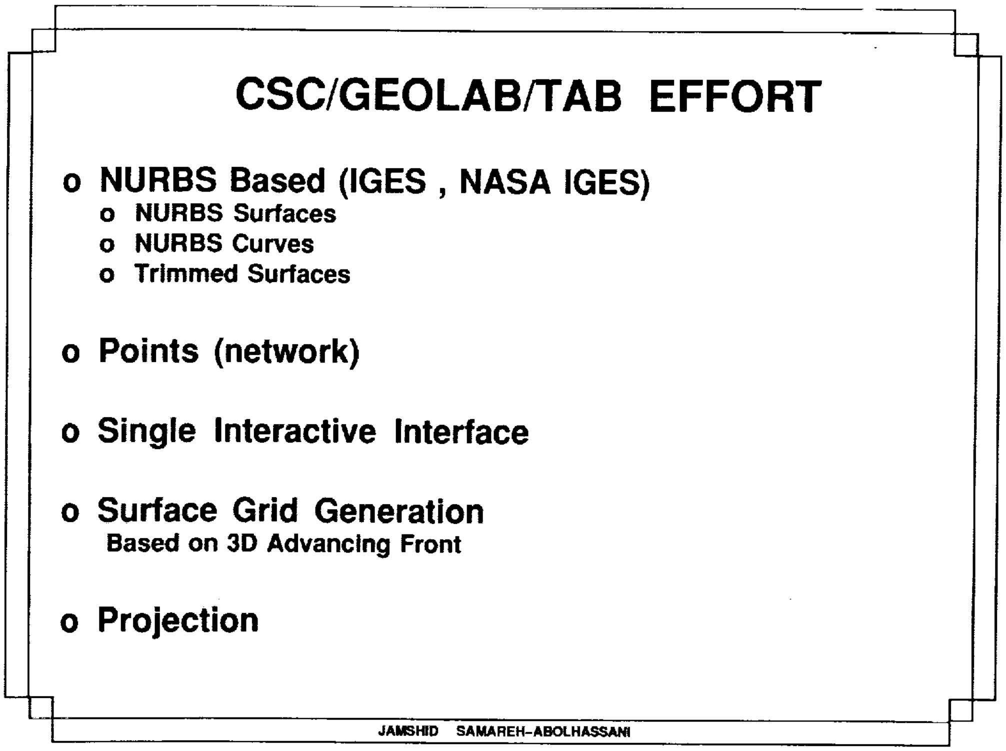
- o Triangulation Is Performed in the Physical Space
- o No Shearing Due Parameter Space
- o Metric Transformation Is not Needed
- o N-Sided Patches with With Multiple Loops
- o Multiple Surfaces
- o Fewer Patches

JAMSHID SAMAREH-ABOLHASSANI



		Type I	Type I	Type 2	
	2D	2½ D	2½DP	2½ D	3D
User Input Factor(# of Patches)	1	4	4	3	2
CPU Time Factor	1	2	3	4	5
Surface Types	P	NA	NURBS	NURBS	NURBS
Surface Accuracy	good	poor	Good	Good	Good
δ, α, β Transformation	simple	simple	simple	Difficult	NA
Problems With Shearing	None	Yes	Yes	Possible	None
Parametric Study	0	0	2	3	3
Number of Surfaces	NA	Many	Many	One	Many
N-Sided Patches Possible	Yes	No	No	Yes	Yes
Problems with Singularity	No	No	Yes	Yes	Yes
Surface Normals Required	No	No	No	No	Yes
History	4	3	3	1	0

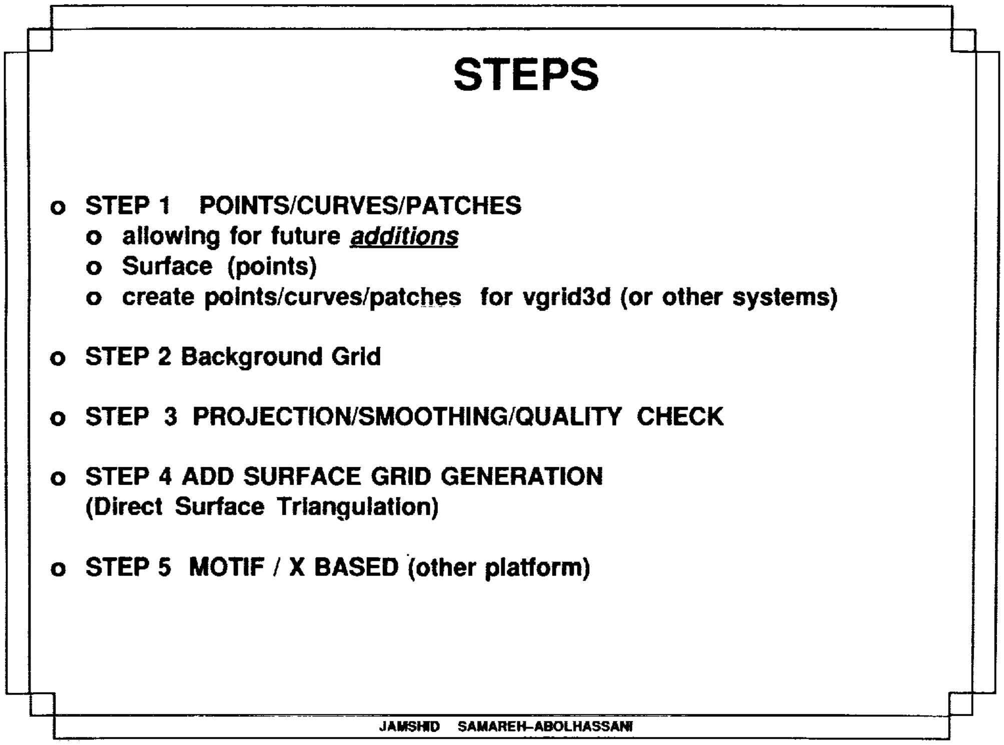
JAMSHID SAMAREH-ABOLHASSANI



CSC/GEOLAB/TAB EFFORT

- o NURBS Based (IGES , NASA IGES)
  - o NURBS Surfaces
  - o NURBS Curves
  - o Trimmed Surfaces
- o Points (network)
- o Single Interactive Interface
- o Surface Grid Generation
  Based on 3D Advancing Front
- o Projection

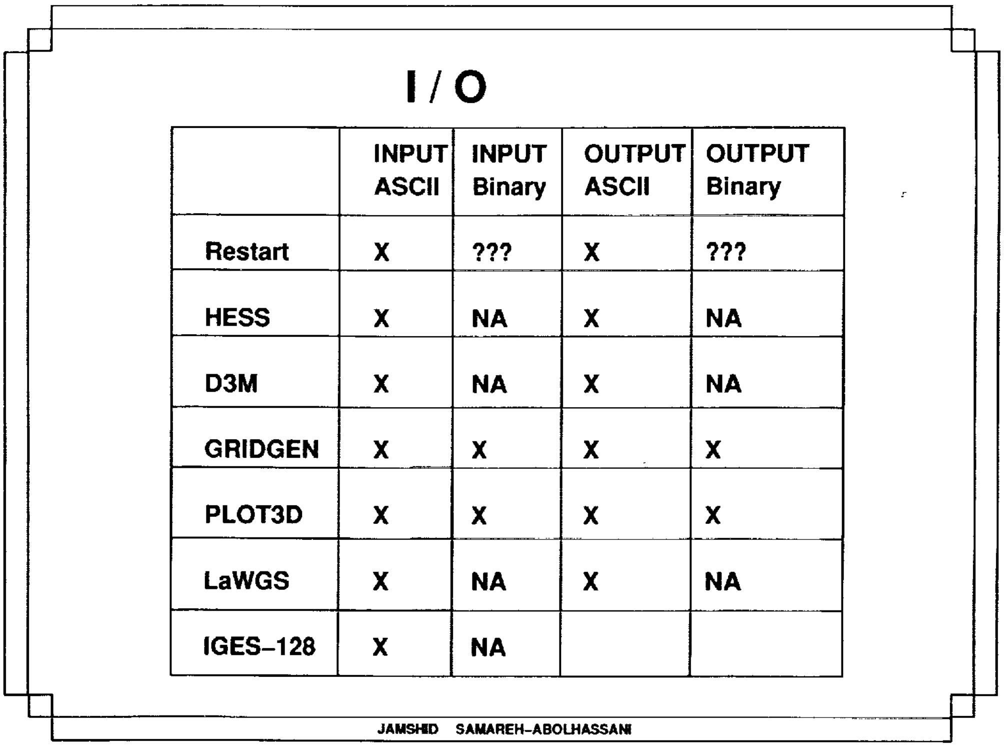
JAMSHID SAMAREH-ABOLHASSANI



STEPS

- o STEP 1 POINTS/CURVES/PATCHES
  - o allowing for future *additions*
  - o Surface (points)
  - o create points/curves/patches for vgrid3d (or other systems)
- o STEP 2 Background Grid
- o STEP 3 PROJECTION/SMOOTHING/QUALITY CHECK
- o STEP 4 ADD SURFACE GRID GENERATION
  (Direct Surface Triangulation)
- o STEP 5 MOTIF / X BASED (other platform)

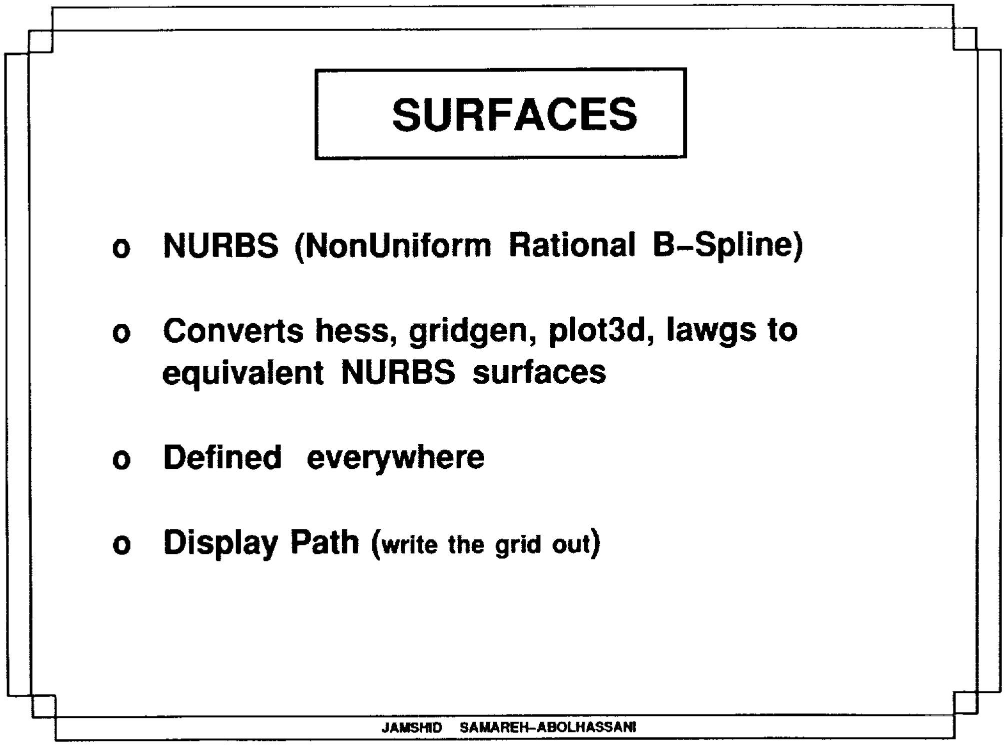
JAMSHID SAMAREH-ABOLHASSANI



I / O

	INPUT ASCII	INPUT Binary	OUTPUT ASCII	OUTPUT Binary
Restart	X	???	X	???
HESS	X	NA	X	NA
D3M	X	NA	X	NA
GRIDGEN	X	X	X	X
PLOT3D	X	X	X	X
LaWGS	X	NA	X	NA
IGES-128	X	NA		

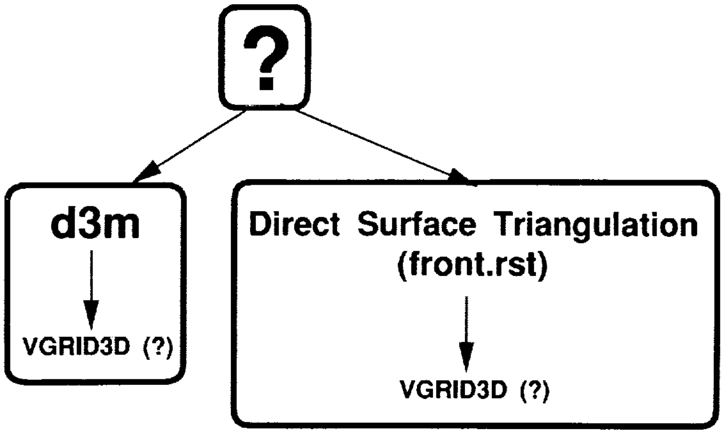
JAMSHID SAMAREH-ABOLHASSANI



SURFACES

- o NURBS (NonUniform Rational B-Spline)
- o Converts hess, gridgen, plot3d, lawgs to equivalent NURBS surfaces
- o Defined everywhere
- o Display Path (write the grid out)

JAMSHID SAMAREH-ABOLHASSANI



?

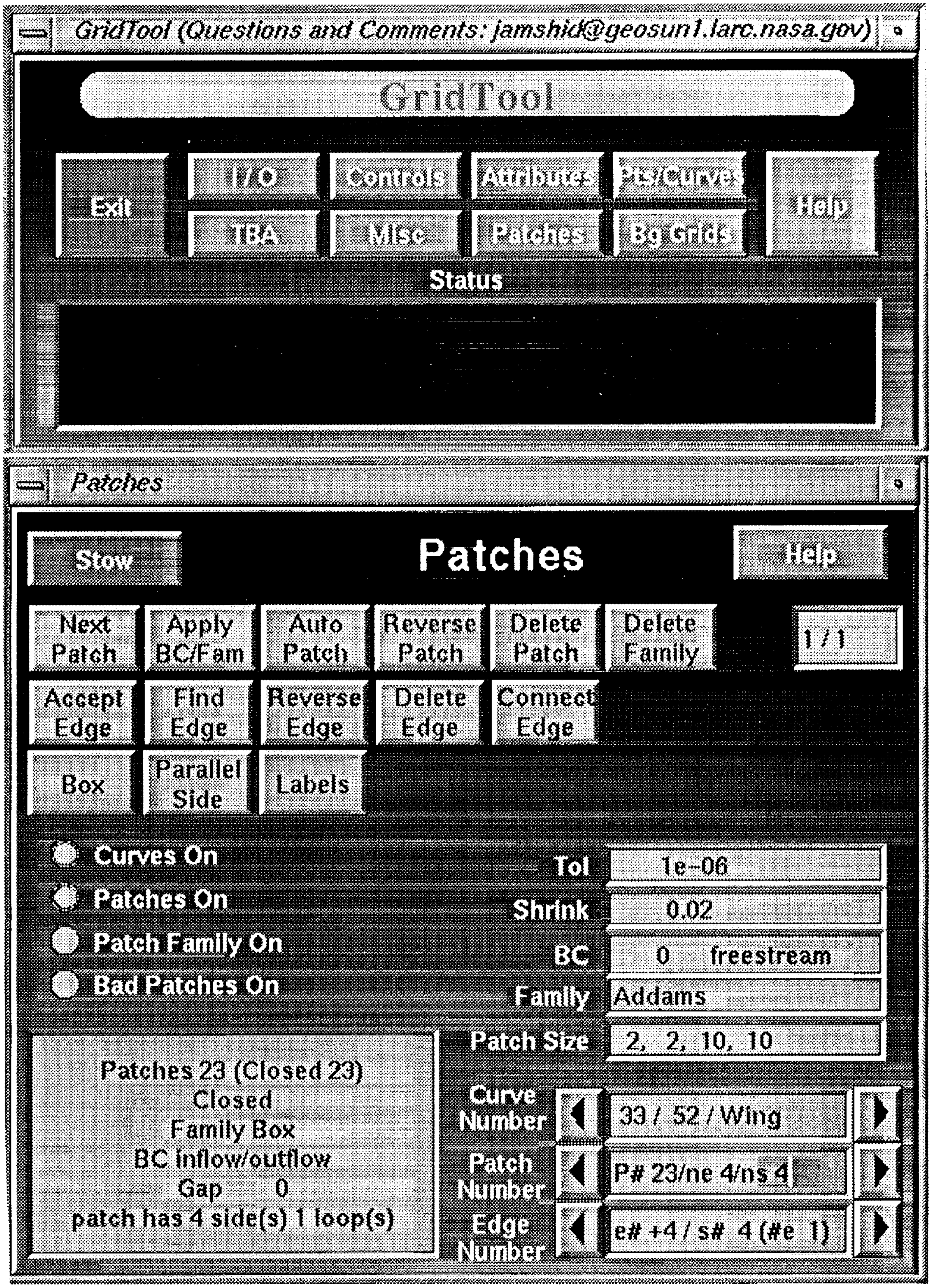
d3m

VGRID3D (?)

Direct Surface Triangulation (front.rst)

VGRID3D (?)

Beta Release 1.0 (Mid May)
Release 1.1 (End of Summer)
To Obtain a Copy, Contact:
pkerr@geolab2.larc.nasa.gov



○ Curves On

○ Patches On

○ Patch Family On

○ Bad Patches On

Tol 1e-06

Shrink 0.02

BC 0 freestream

Family Addams

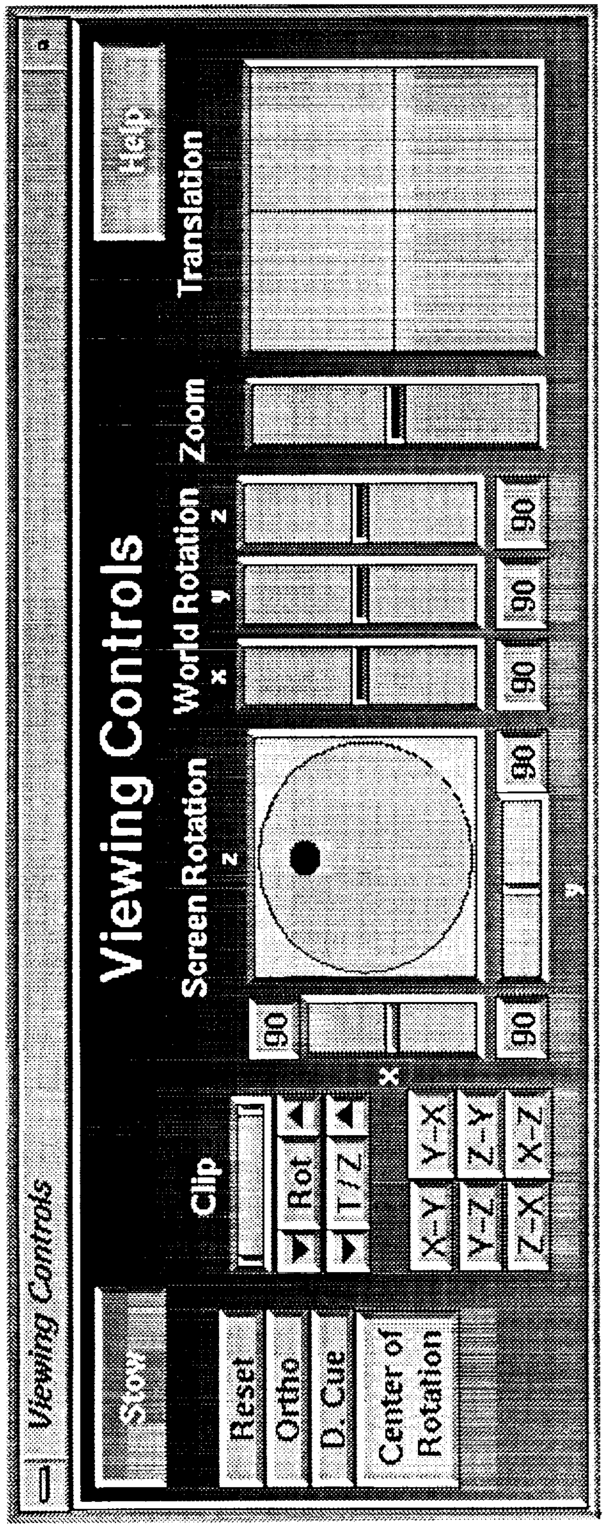
Patch Size 2, 2, 10, 10

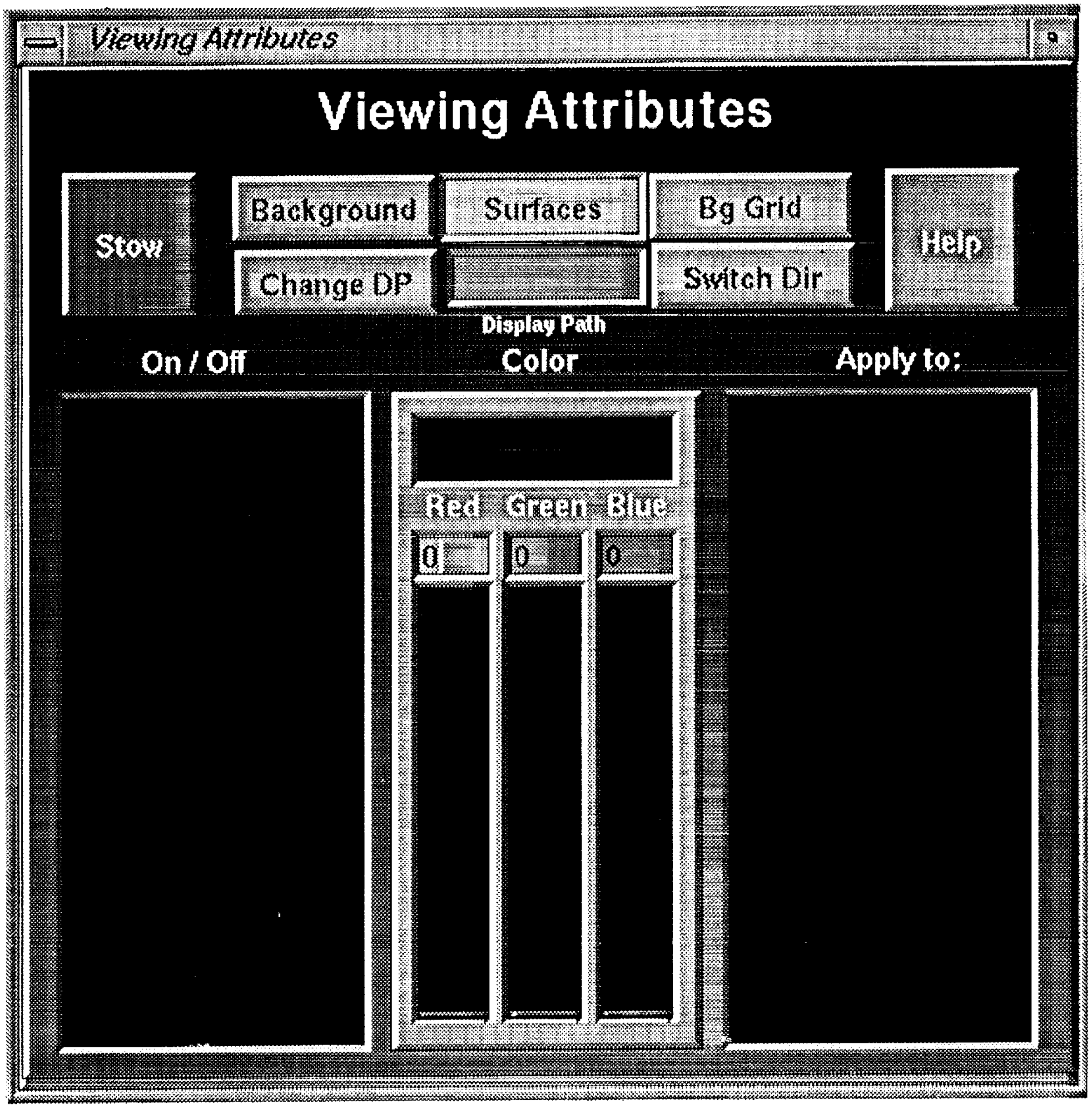
Curve Number 33 / 52 / Wing

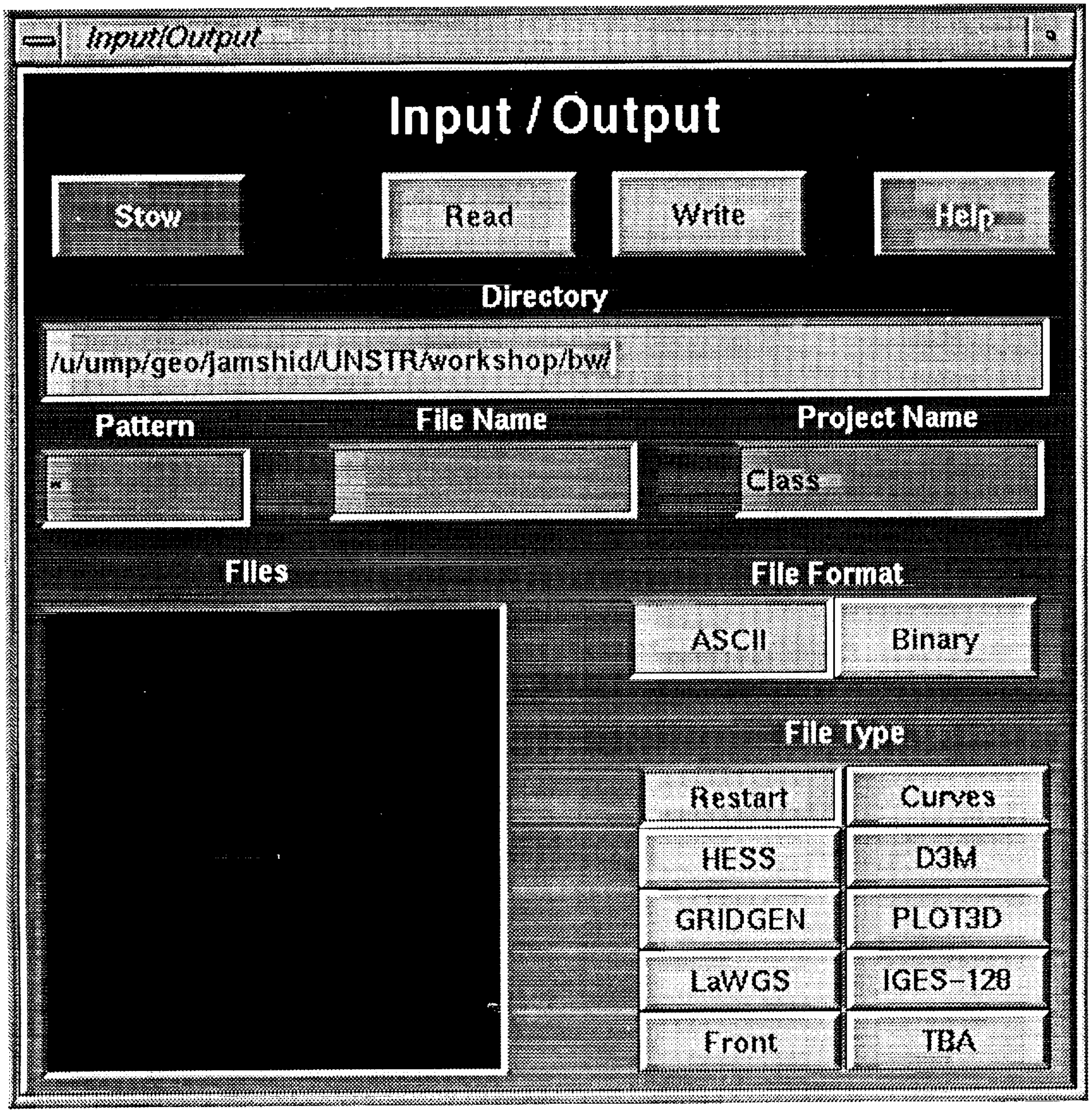
Patch Number P# 23/ne 4/ns 4

Edge Number e# +4 / s# 4 (#e 1)

Patches 23 (Closed 23)
Closed
Family Box
BC inflow/outflow
Gap 0
patch has 4 side(s) 1 loop(s)


Viewing Controls
Stow
Reset
Ortho
D. Cue
Center of Rotation
Clip
Rot
T/Z
X-Y
Y-X
Y-Z
Z-Y
Z-X
X-Z
Viewing Controls
Screen Rotation
z
x
y
90
90
90
World Rotation
x
y
z
90
90
90
Zoom
Translation
Help


Viewing Attributes
Viewing Attributes
Stow
Background
Surfaces
Bg Grid
Change DP
Switch Dir
Help
Display Path
On / Off
Color
Apply to:
Red
Green
Blue
0
0
0

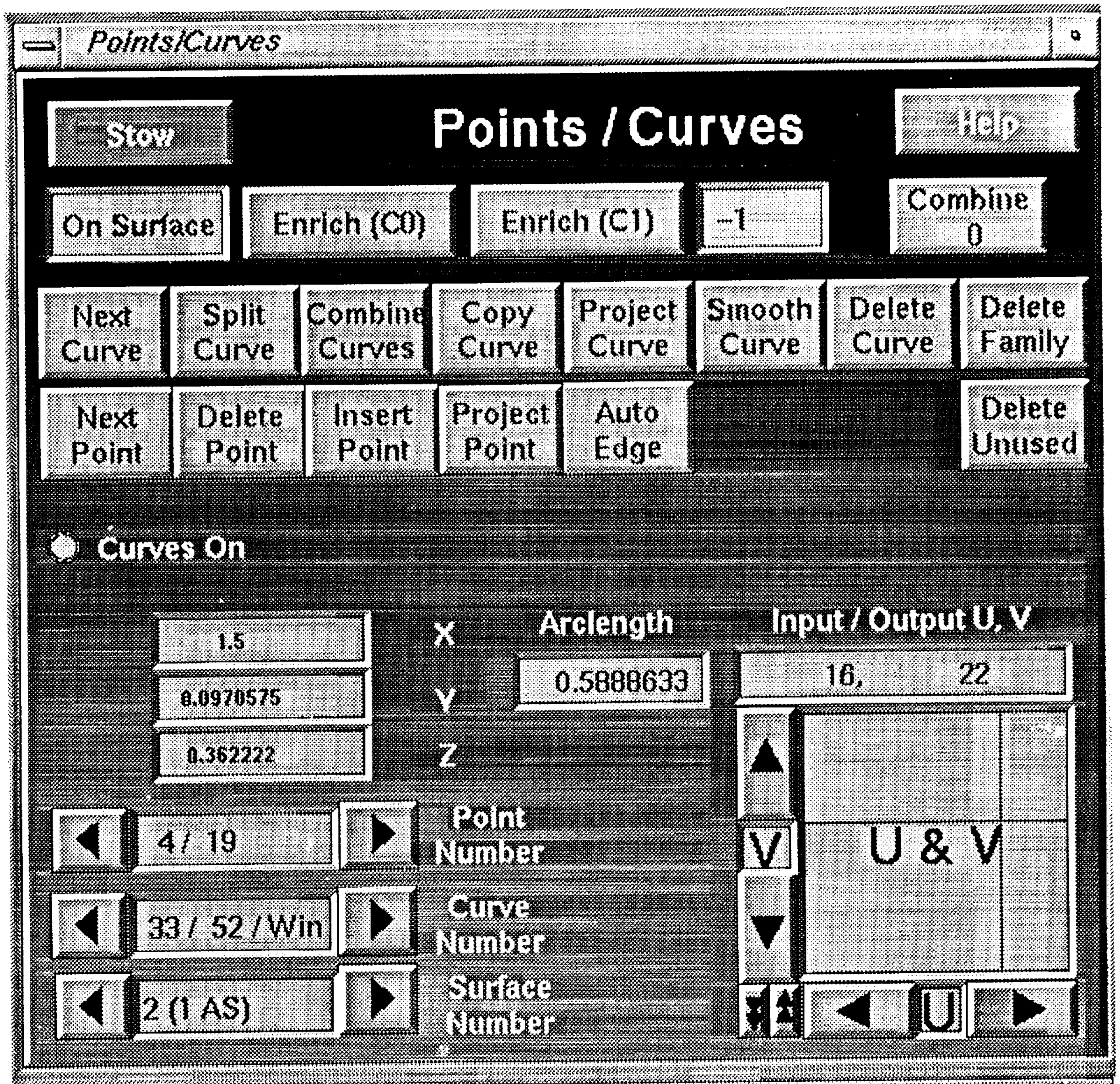


Directory
/u/ump/geo/jamshid/UNSTR/workshop/bw/

Pattern
*

File Name

Project Name
Class



X 1.5

Y 0.0970575

Z 0.362222

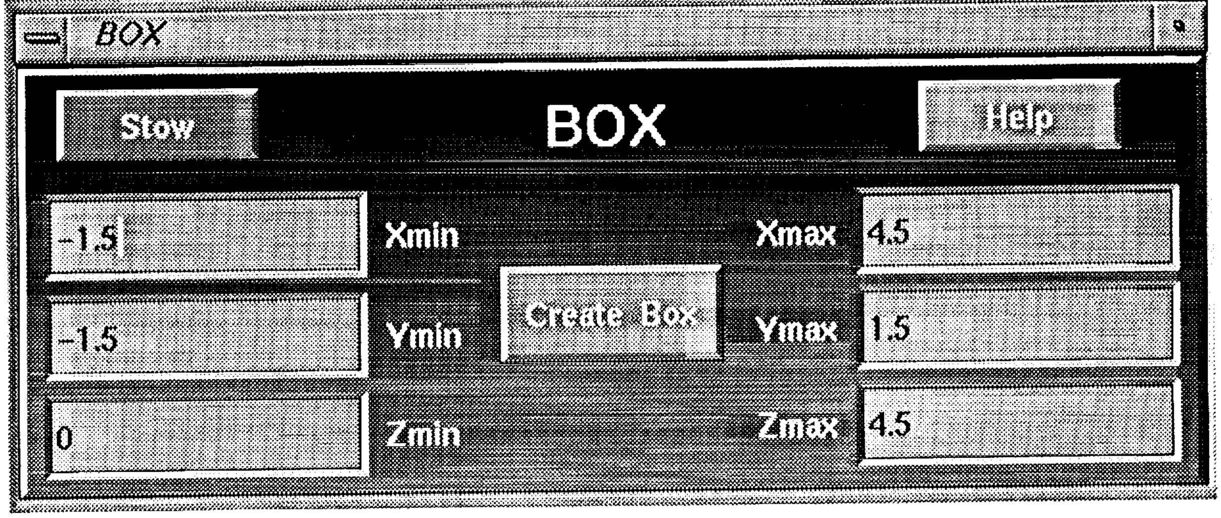
Arclength 0.5888633

Input / Output U, V 16, 22

Point Number 4/ 19

Curve Number 33 / 52 / Win

Surface Number 2 (1 AS)



Xmin -1.5

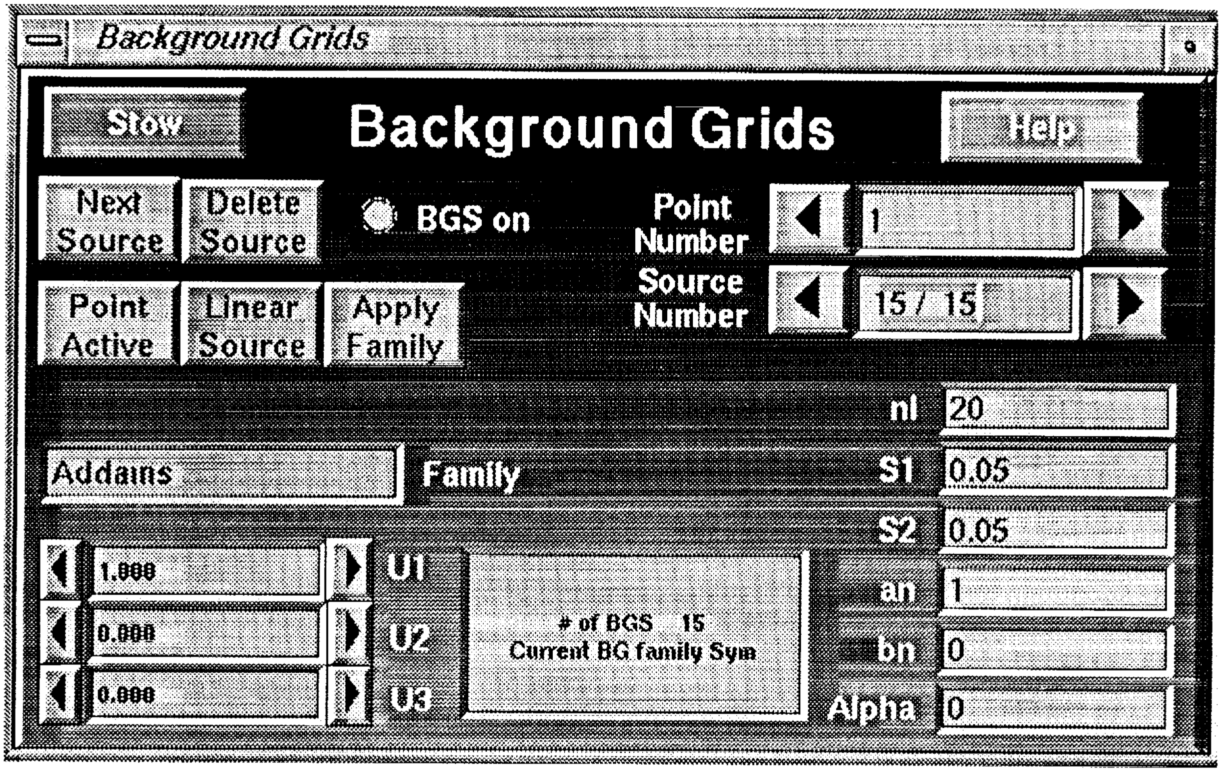
Ymin -1.5

Zmin 0

Xmax 4.5

Ymax 1.5

Zmax 4.5



Background Grids

● BGS on

Point Number: 1

Source Number: 15 / 15

nl: 20

Addains Family

S1: 0.05

S2: 0.05

U1: 1.000

U2: 0.000

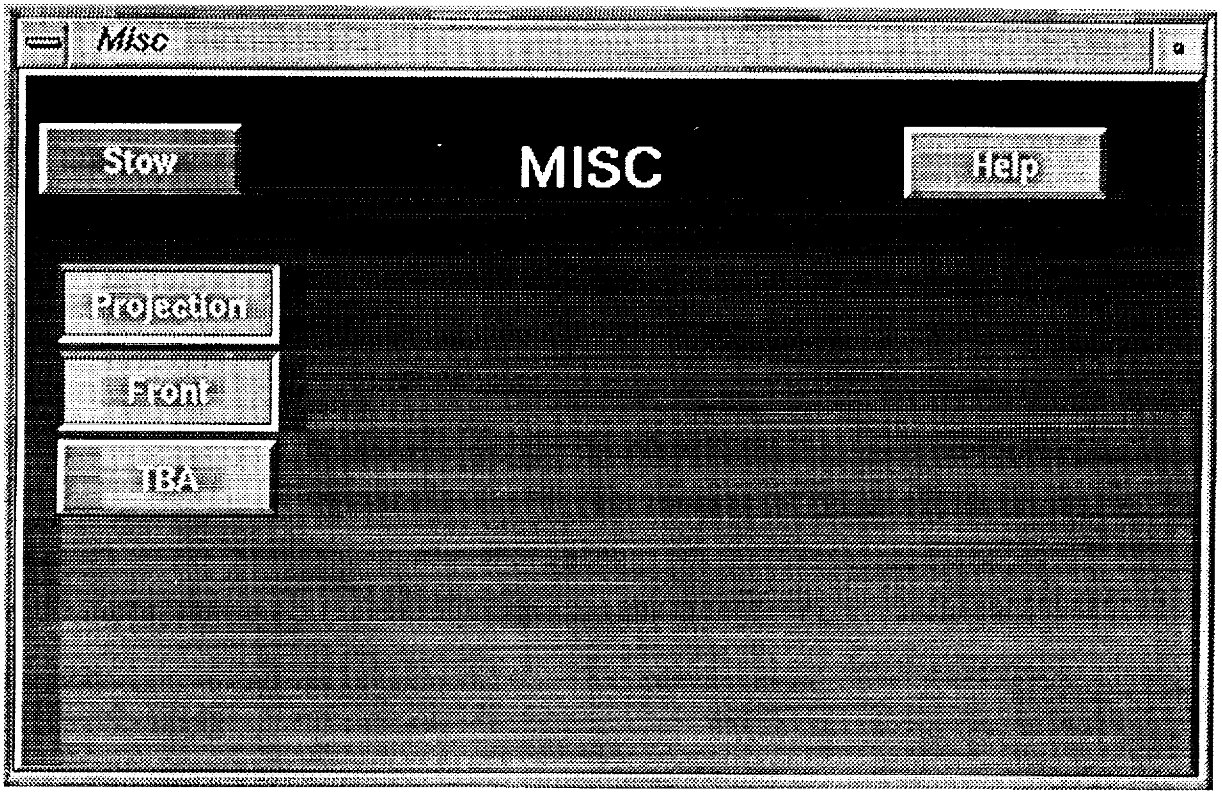
U3: 0.000

an: 1

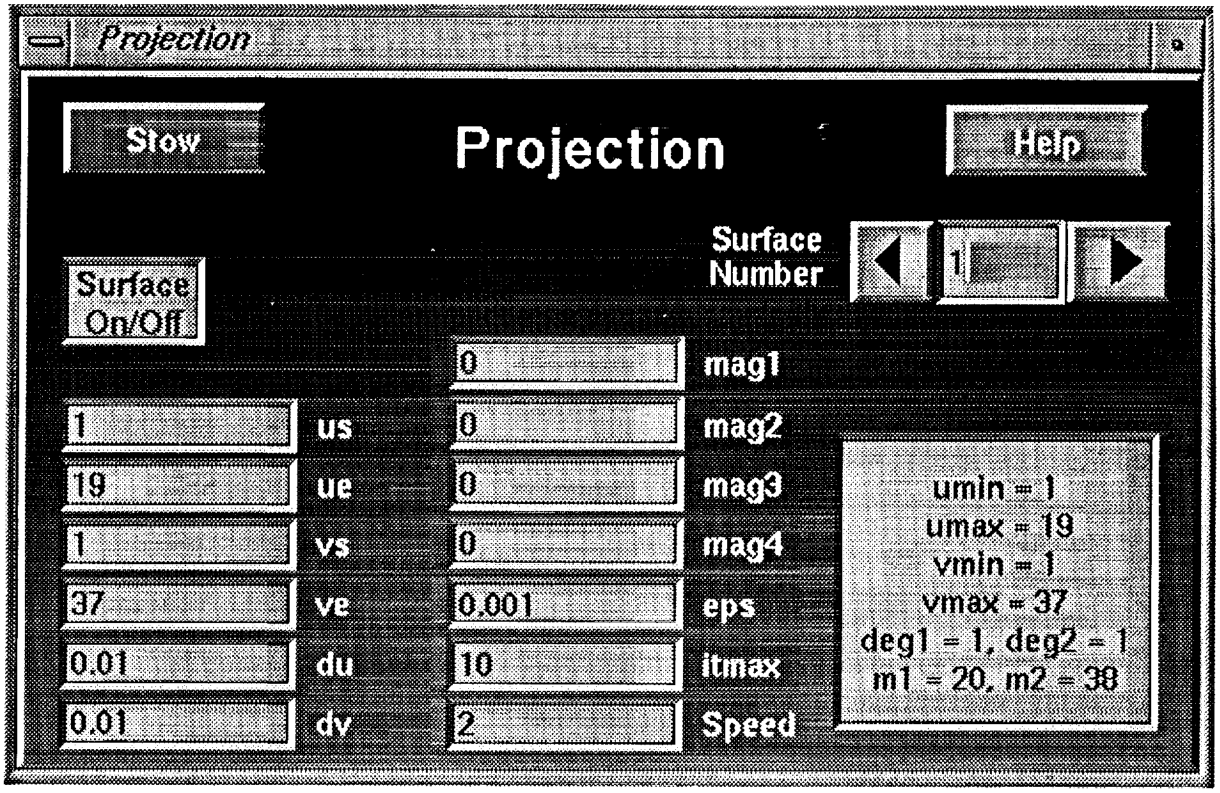
bn: 0

Alpha: 0

# of BGS 15
Current BG family Sym



MISC

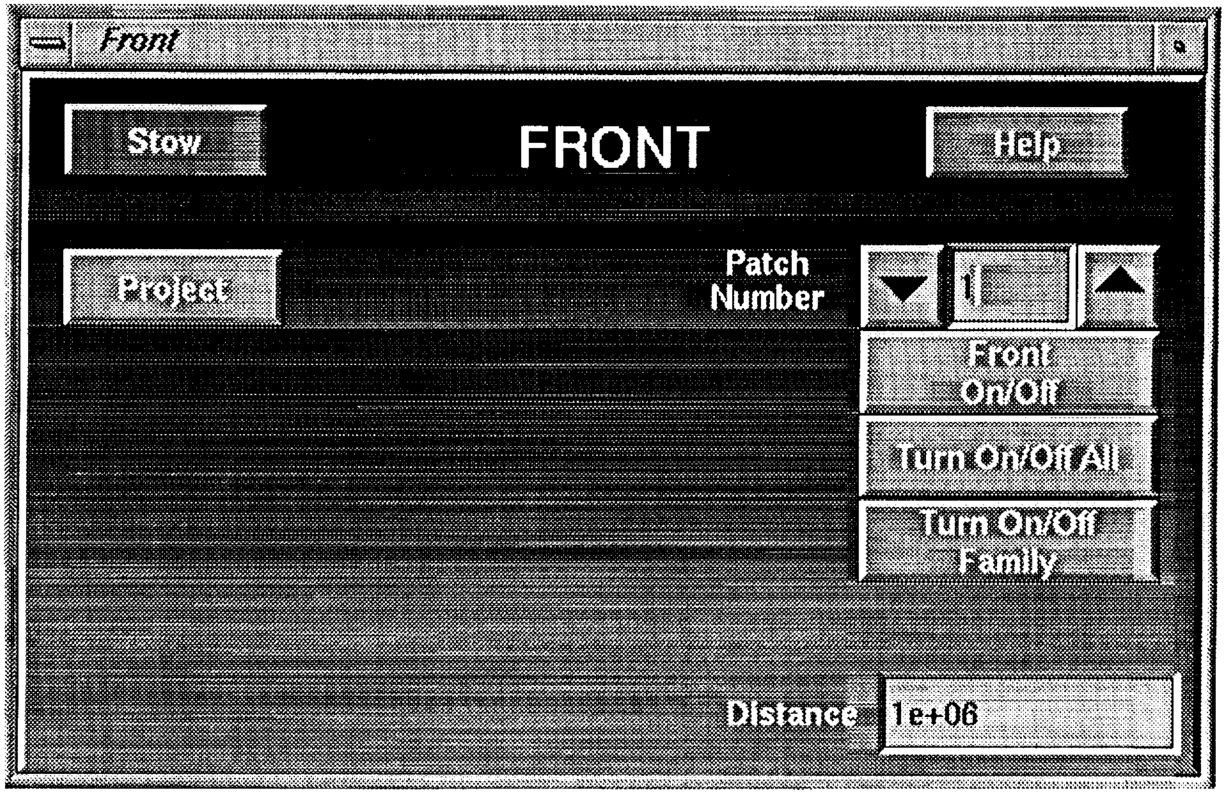


Projection

Surface Number 1

		0	mag1
1	us	0	mag2
19	ue	0	mag3
1	vs	0	mag4
37	ve	0.001	eps
0.01	du	10	itmax
0.01	dv	2	Speed

umin = 1
umax = 19
vmin = 1
vmax = 37
deg1 = 1, deg2 = 1
m1 = 20, m2 = 38



FRONT

Patch Number 1

Distance 1e+06

3D EULER SOLUTIONS USING AUTOMATED CARTESIAN GRID GENERATION

JOHN E. MELTON
NASA AMES RESEARCH CENTER

FRANCIS Y. ENOMOTO
NASA AMES RESEARCH CENTER

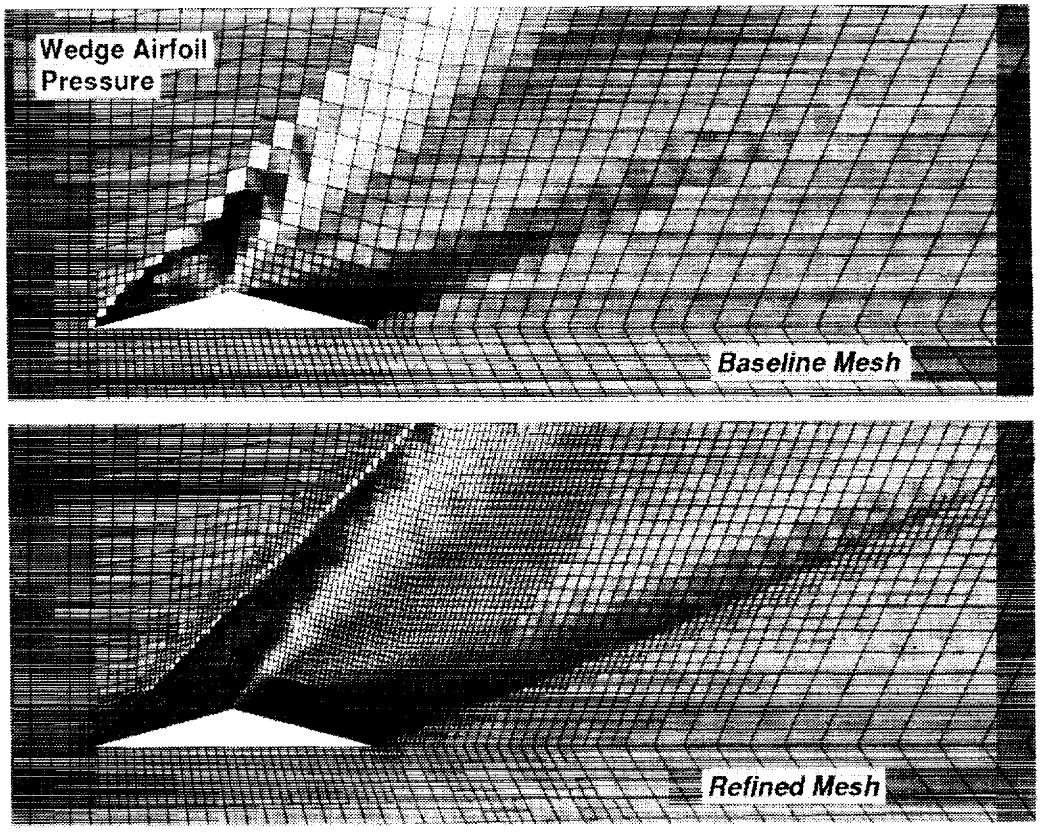
MARSHA J. BERGER
NEW YORK UNIVERSITY

Agenda

- History
- Cartesian Overview
- Technique Comparisons
- 3D Cartesian Grid Generation Strategy
- Survey of simple test cases
- Current research and future plans
- Summary

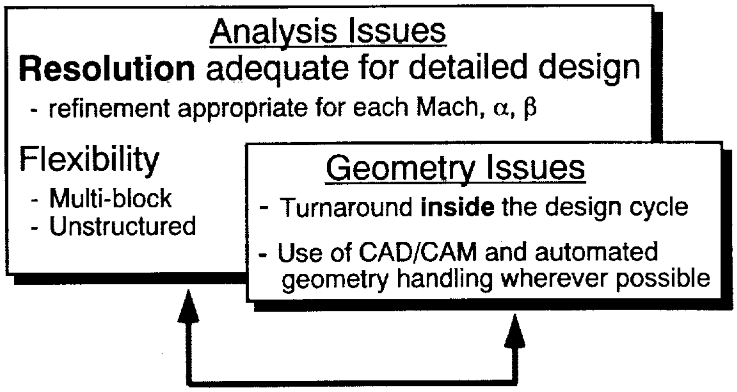
History

- Lessons from ATP grid generation
- AIAA 91-0637 with Thomas and Cappuccio
  - Unstructured, refined, hexahedral body-fitted grid
  - Euler FV RK4 Jameson flow solver algorithm (FLO57)
- TIGER = Topologically Independent, Euler Refinement
- GIRAFFE = Grid Interactive Refinement and Flow Field Examination



CFD and the Design Cycle

Compute ***better*** solutions ***faster*** and ***cheaper***



Analysis Issues

Resolution adequate for detailed design

- refinement appropriate for each Mach, α, β

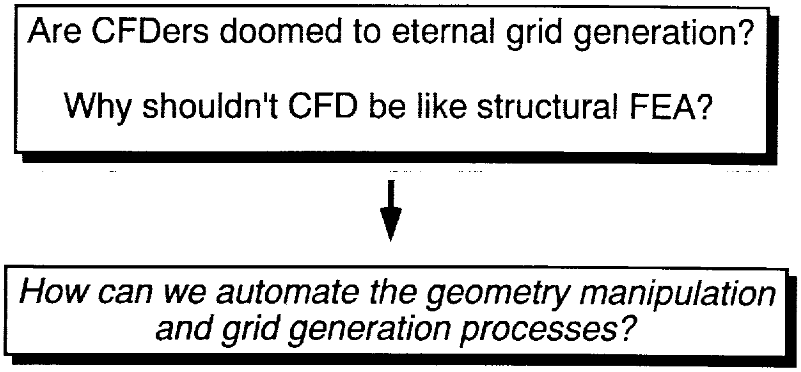
Flexibility

- Multi-block
- Unstructured

Geometry Issues

- Turnaround **inside** the design cycle
- Use of CAD/CAM and automated geometry handling wherever possible

Three Important Questions



Are CFDers doomed to eternal grid generation?

Why shouldn't CFD be like structural FEA?

How can we automate the geometry manipulation and grid generation processes?

Cartesian Grid Strategy

- South, Clarke, Salas, Hassan, Berger, LeVeque, Powell, Epstein, Morinishi, TRANAIR

- Make the computer do the work
  - ***Interactivity ≠ Automation***
  - Divorce surface grid from field grid
  - Use computational geometry algorithms to extract surface/cell intersection information
  - Use NURBs (Non-Uniform Rational B-Splines) to maintain a single, accurate, database

- Use grid refinement for "efficient" resolution
  - Unstructured grid (block or cell)
  - Flowfield and geometry-based refinement

Technique Comparisons

Task	Structured Body-fitted	Cartesian
Grid generation	tedious and boring time-consuming requires surface grid good tools are available	automated NURB accuracy no surface grid research software
Flux and BCs	"simple" and familiar	"complicated"
Connectivity overhead	minimal	~60 words/cell
Grid refinement/ adaptation	not automated difficult	automated for both geometry and flowfield
Flow solver	highly vectorizable	vectorizable

TIGER Surface Geometry

Entity	Advantages	Disadvantages
Triangles	"Simple" intersections	Poor refinement accuracy
	LaWGS / FEM / PANAIR	Creation
	Compute - inexpensive	Loss of surface information
NURBS	Direct from CAD	"Nonlinear" intersections
	Complete accuracy	- tolerance specifications
	Complete information	- polynomial root-finding
	NASA/IGES standard	Topology determination
		Unfamiliarity
		Compute - expensive

2-Step Cartesian Grid Generation Algorithm

1 - Create initial equi-spaced Cartesian grid

Flag cells that intersect with surface
Refine along with a number of neighbors
Repeat to create desired resolution

2 - Compute cell geometric information

- face areas
- body surface normals
- cell volumes
- face and volume centroids

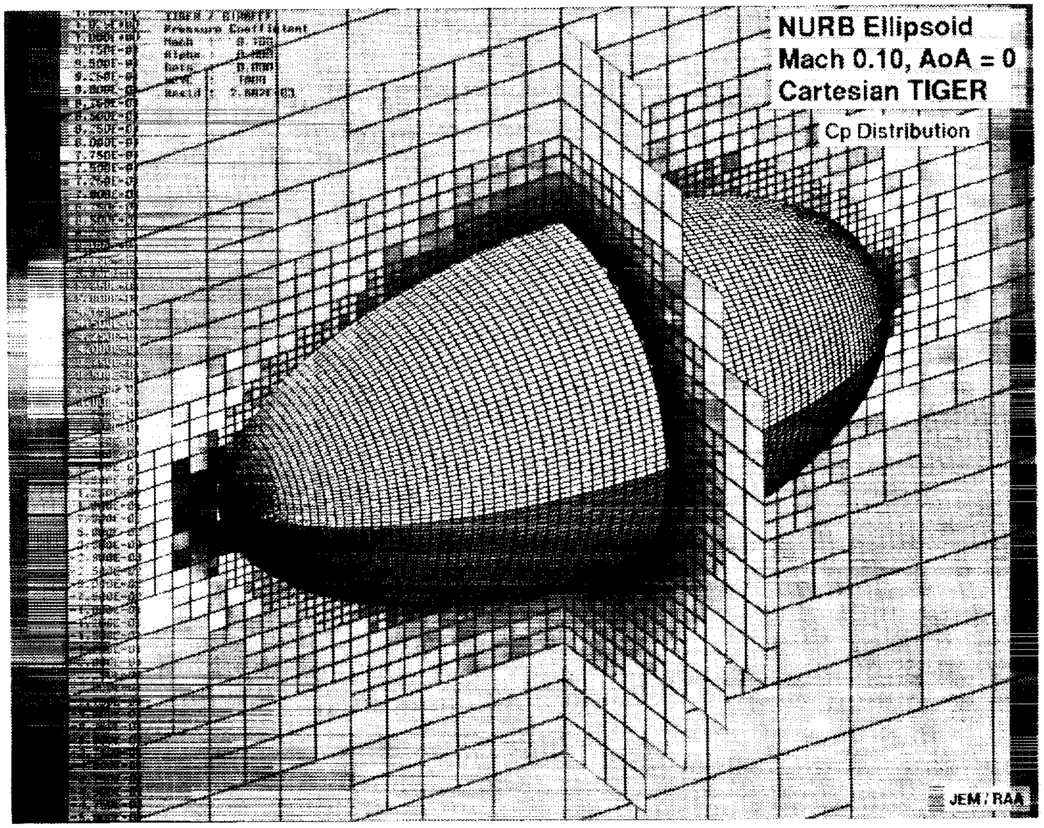
Current TIGER Connectivity Data Structure

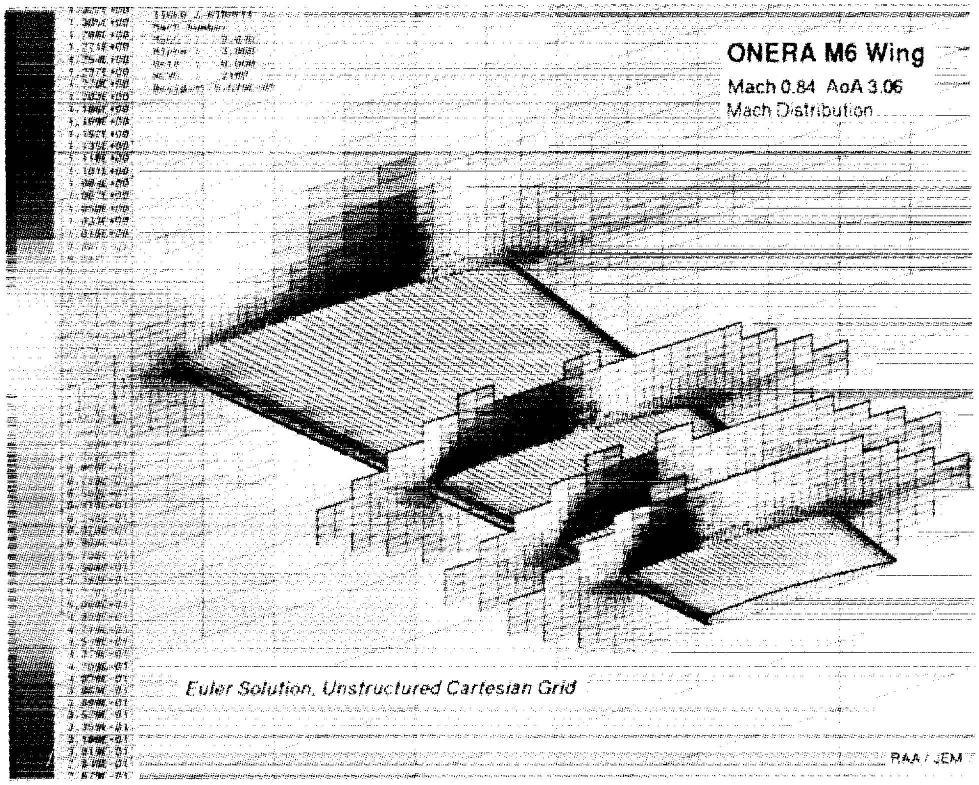
Item	Words per cell
Pointer to connecting cells	6 faces x 4 connections per face
Face BC flags	6 faces x 2 flags per face
Face area vectors	7 faces x 3 components per face
Cell Refinement Level	1
Cell BC flag	1
Cell volumes	1

Unstructured Cartesian Overhead ~ 60+ words per cell

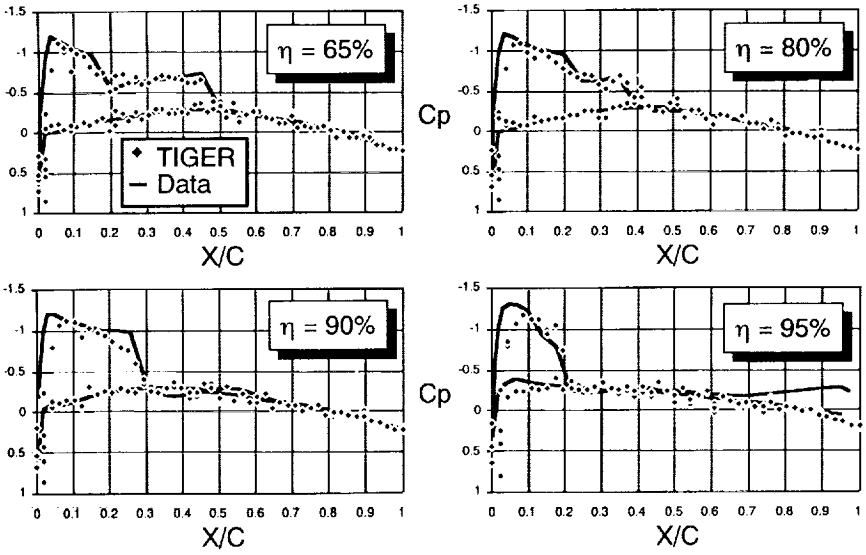
Survey of Test Cases

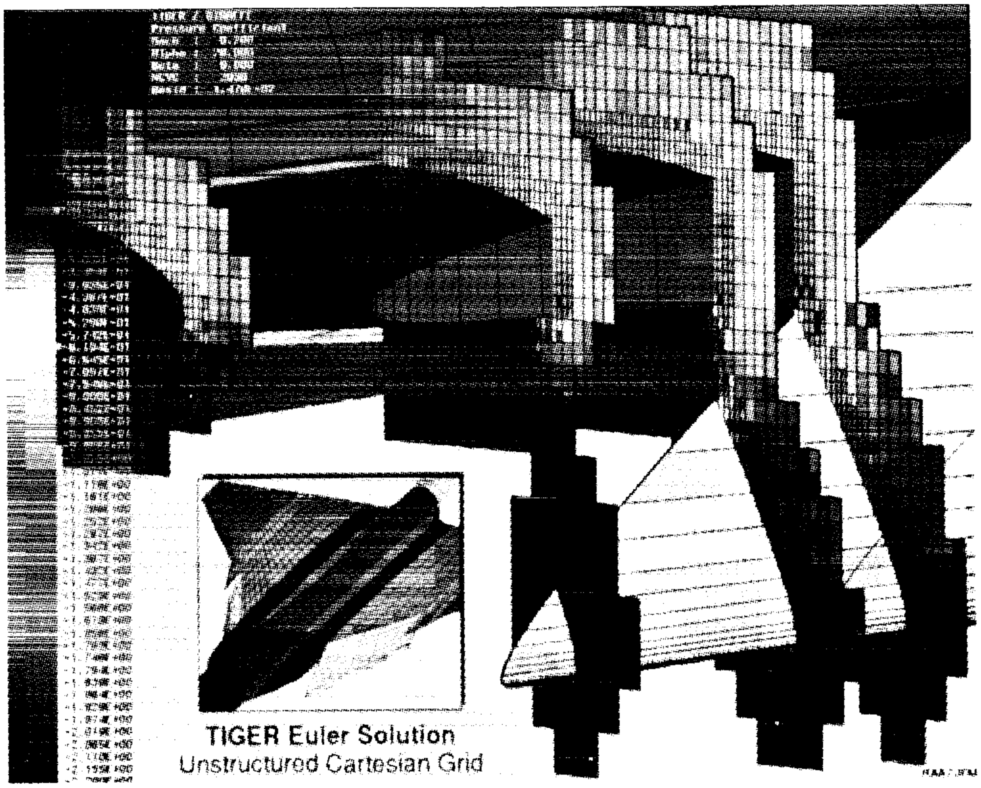
- Prolate Spheroid - NURB input
- ONERA M6 wing - Triangle input
- HSCT with LE flap - Triangle input





ONERA M6 Mach=0.84 α=3.06




TIGER Euler Solution
Unstructured Cartesian Grid

HSCT Grid Generation Command Files

Step 1

```
1           1: use tiger.net data   2: use tiger.tri
1           1: flip y-z  0: don't flip
1           1: make new base grid  2: restart
-1   4000        : x-range
-1300   1700   : y-range
0   1201         : z-range
17  15   9       : dims
1           1: split surface cells  0: stop
6             : number of splitting passes
2             : number of buffer layers
1           1: reset symmetry plane cells  0: skip
1           1: compress the files  0: skip
```

Step 2

```
1       1: read from tiger.net  2: tiger.tri
1       1: flip y-z  0: don't flip
1       1: reset symmetry plane cells  0: skip
1       1: compress files  0: skip
```

Current Research and Future Plans

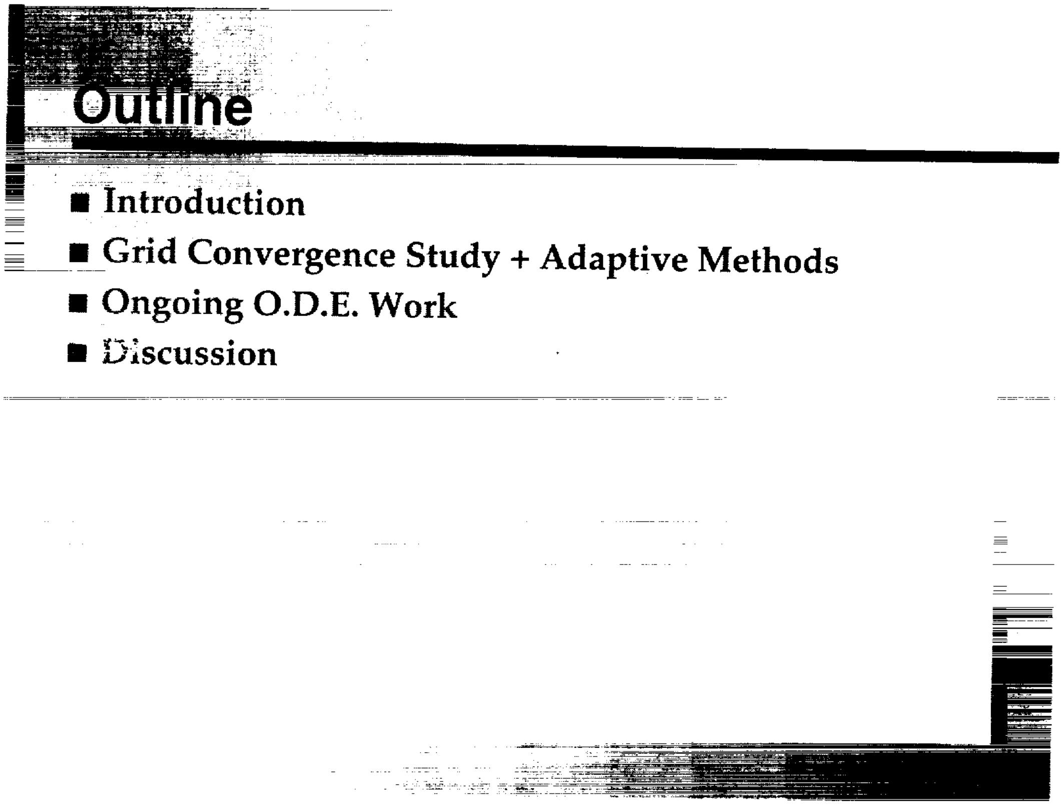
- Improved flux and dissipation modeling
- Improved boundary conditions
- "Intelligent" grid generation
- Flowfield refinements
- Validations

Summary

- Use of a single NURB geometry database for design and analysis has many advantages
  - allows for geometry manipulation with commercial CAD/CAM tools
  - provides analyst with complete and accurate surface information
  - provides consistent method for data transfer
- A mature unstructured Cartesian approach will have additional advantages
  - eliminate surface and volume gridding tasks via automation
  - provide local resolution appropriate for each flow condition
  - shrink CFD turnaround from months to hours
  - allow designers to concentrate on aerodynamic performance instead of computational geometry and numerical analysis
- Interactive techniques should be viewed as short term solutions, and not as long term CFD goals

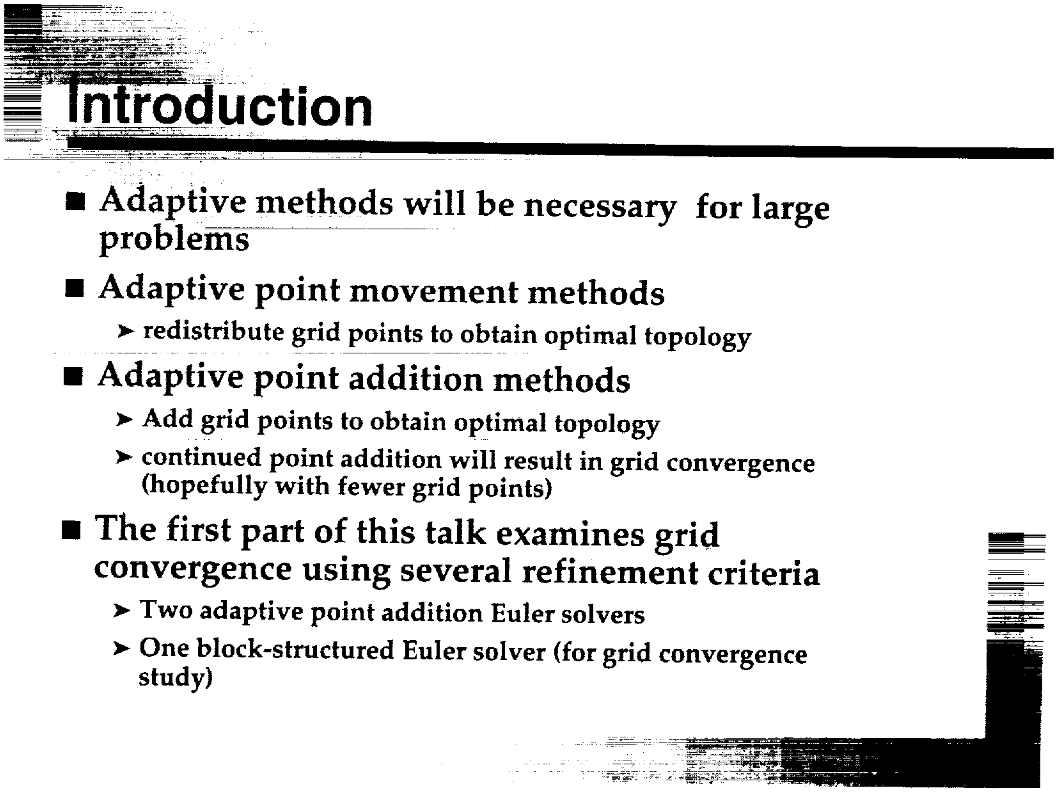
ACCURACY ASSESSMENT FOR GRID ADAPTATION

GARY P. WARREN
NASA LANGLEY RESEARCH CENTER



Outline

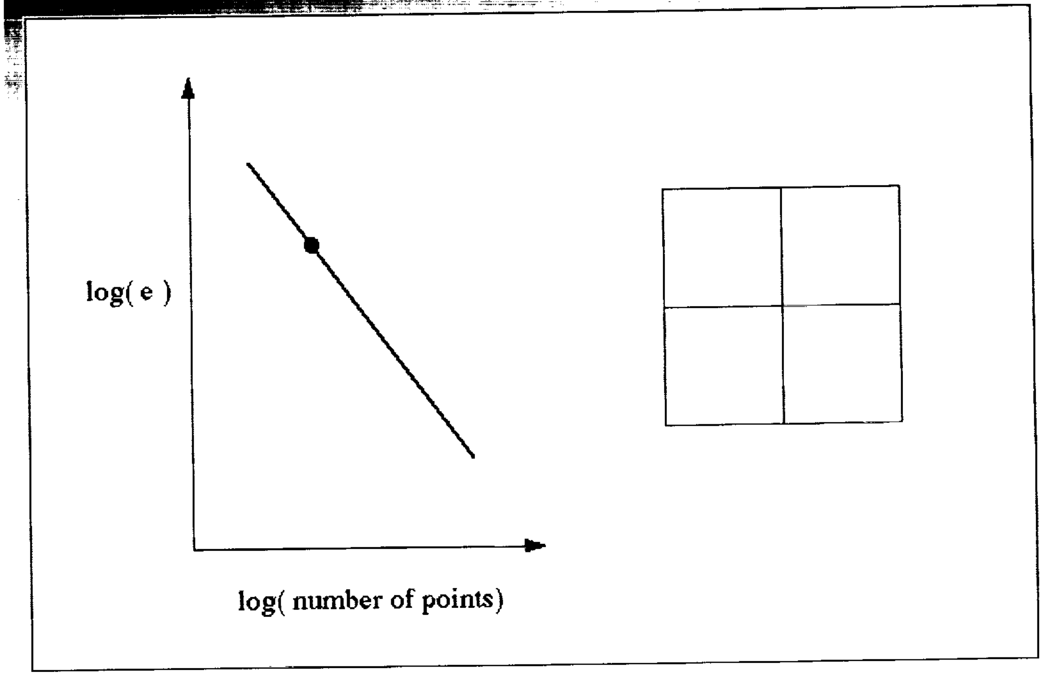
- Introduction
- Grid Convergence Study + Adaptive Methods
- Ongoing O.D.E. Work
- Discussion



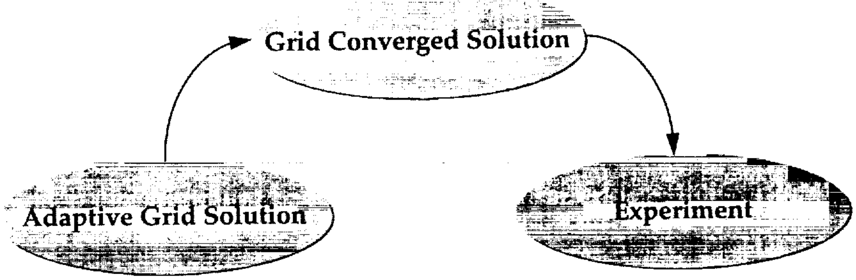
Introduction

- Adaptive methods will be necessary for large problems
- Adaptive point movement methods
  - ➤ redistribute grid points to obtain optimal topology
- Adaptive point addition methods
  - ➤ Add grid points to obtain optimal topology
  - ➤ continued point addition will result in grid convergence (hopefully with fewer grid points)
- The first part of this talk examines grid convergence using several refinement criteria
  - ➤ Two adaptive point addition Euler solvers
  - ➤ One block-structured Euler solver (for grid convergence study)

Uniform Refinement



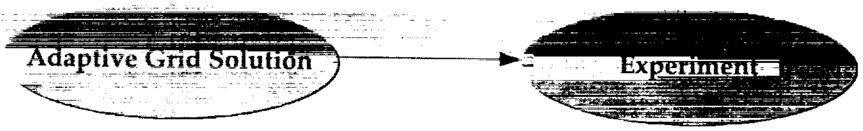
What we should be doing


Grid Converged Solution

Adaptive Grid Solution

Experiment

What we should not be doing (Until we have the bugs out)

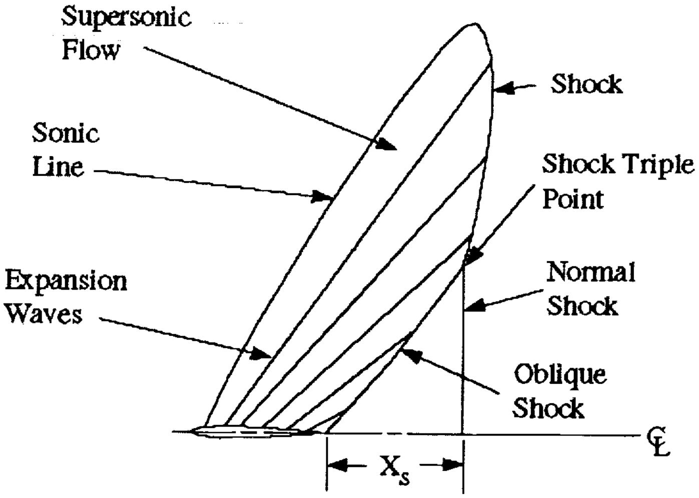


Adaptive Grid Solution

Experiment

AGARD 03 Test Case

NACA 0012
Mach = 0.95
$\alpha = 0$ degrees



Grid Convergence Study

- Conditions for grid convergence

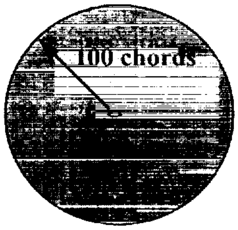
$$\lim_{n\to\infty} \| e \|_\infty = 0$$

This occurs if method is consistent and

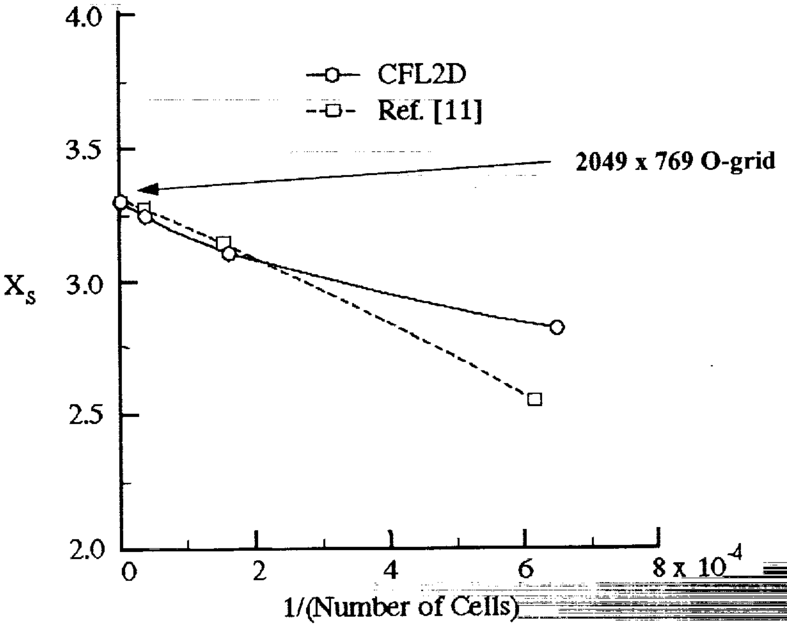
$$\lim_{n\to\infty} h_i = 0$$

Grid Convergence Study

- O-grid with fixed outer boundary of 100 chords
- Use sequence of finer grids
  - ➤ 65 x 25
  - ➤ 129 x 49
  - ➤ 257 x 97
  - ➤ 2049 x 769
- Extrapolate shock location to "infinitely refined grid"



Grid Convergence Study



Common Adaptive Methods

Divided differences

$$\tilde{e} = \frac{\partial p}{\partial x} \approx \frac{\Delta p}{\Delta x}$$

Undivided differences

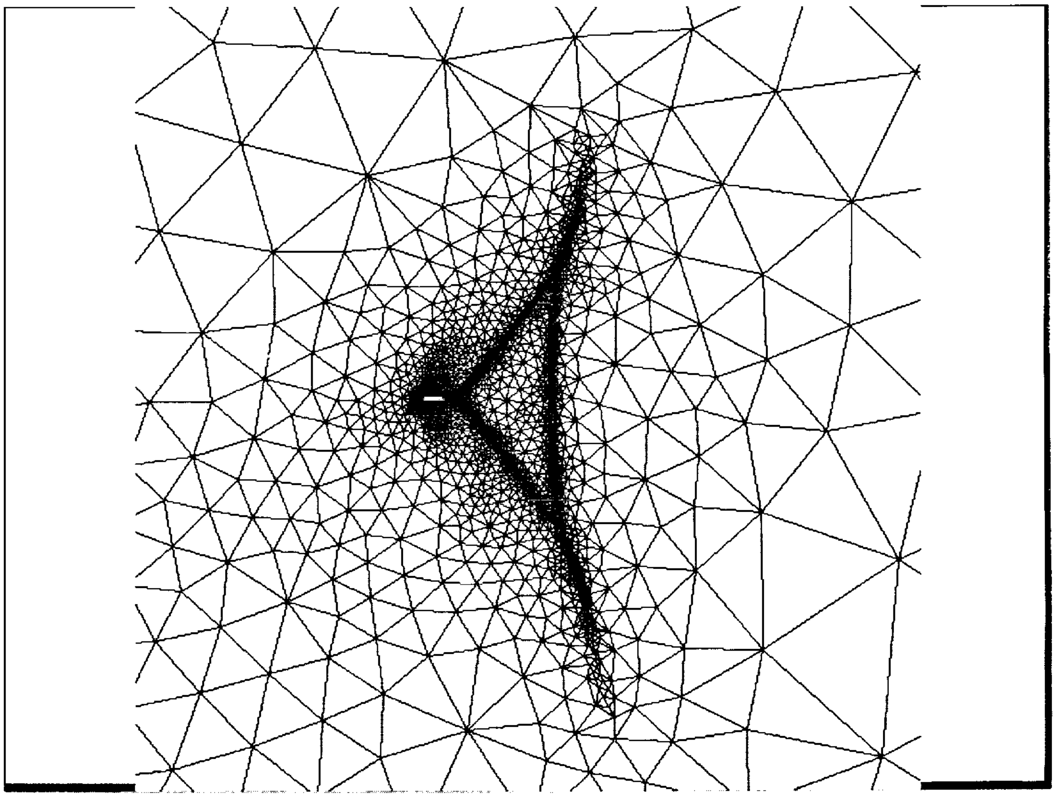
$$\tilde{e} = \tilde{\Delta} p$$

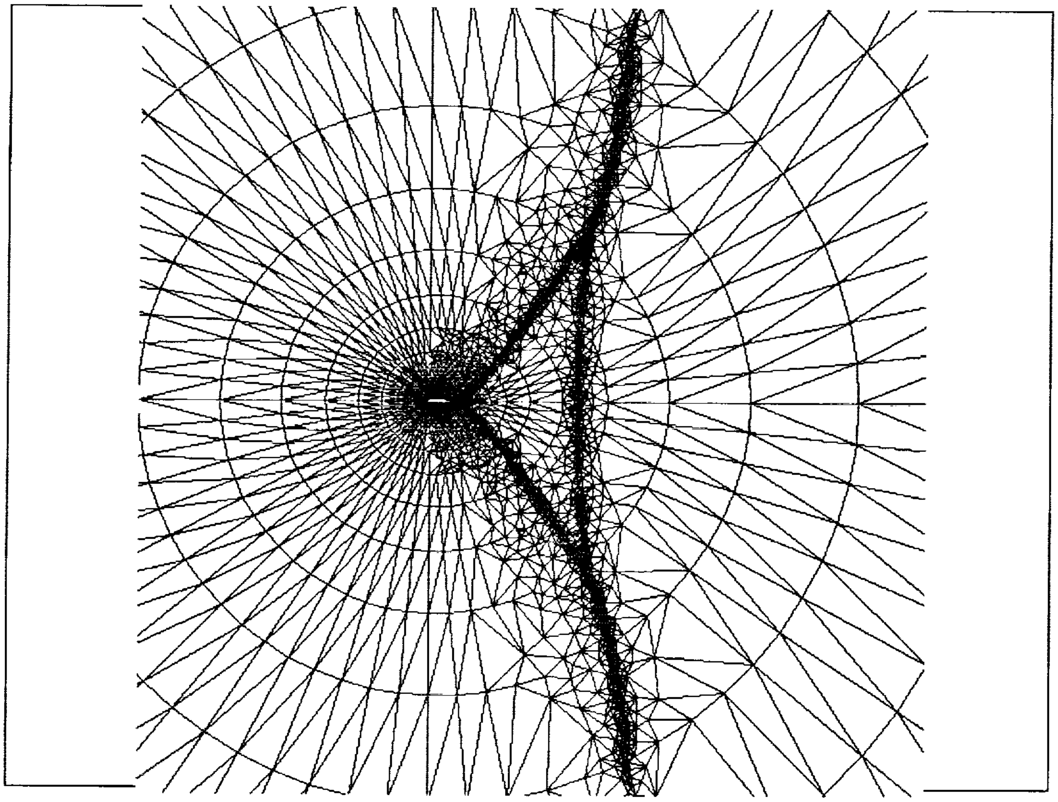
$$\tilde{e} = h^2 \frac{\partial^2 p}{\partial x^2} \approx \tilde{\Delta}^2 p$$

Truncation error estimates

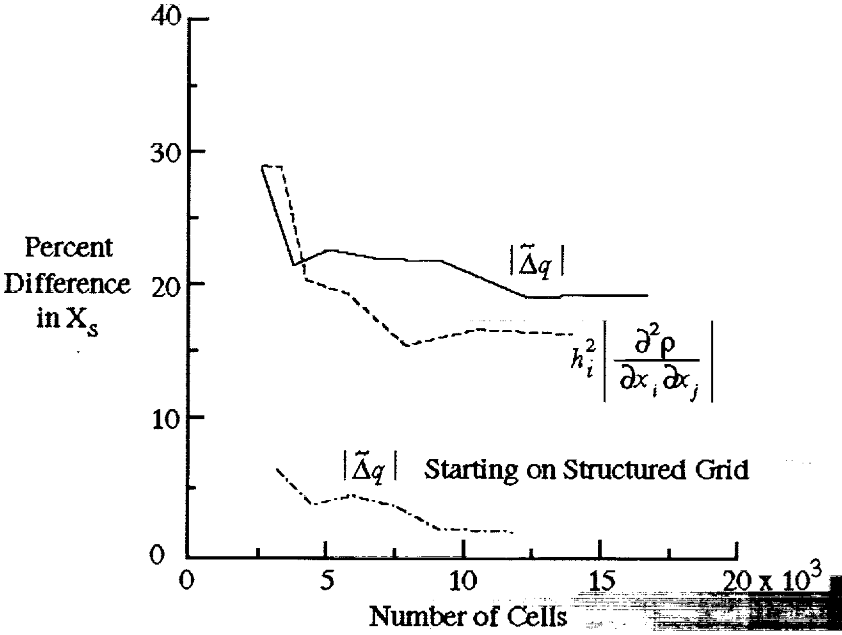
Threshold Determination

- Based on experience (?)
- Statistical approach
  - Threshold = average + standard deviation

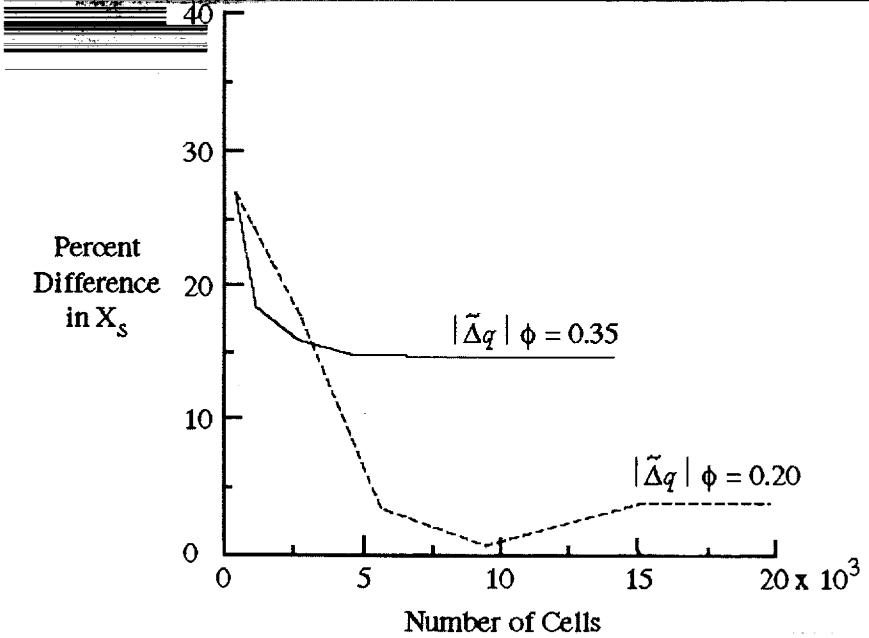




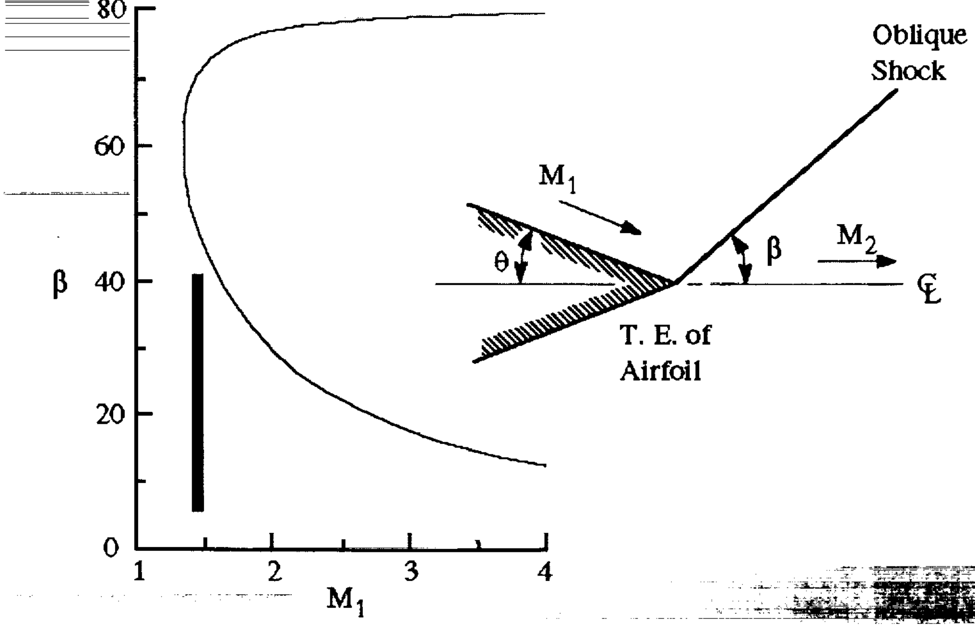
Adaptive Results (FUN2D)



Adaptive Results (SUN2D)



Shock Polar for θ = 7.99 degrees



C.4

Corrected Adaptation Method

Problem occurs when

$$\lim_{n\to\infty} \| e \|_{\infty} \neq 0$$

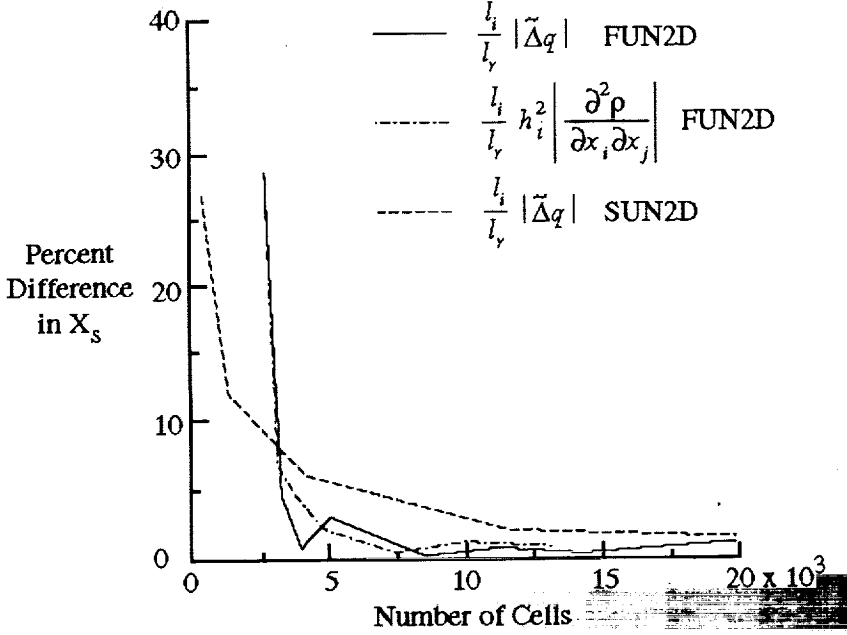
which causes

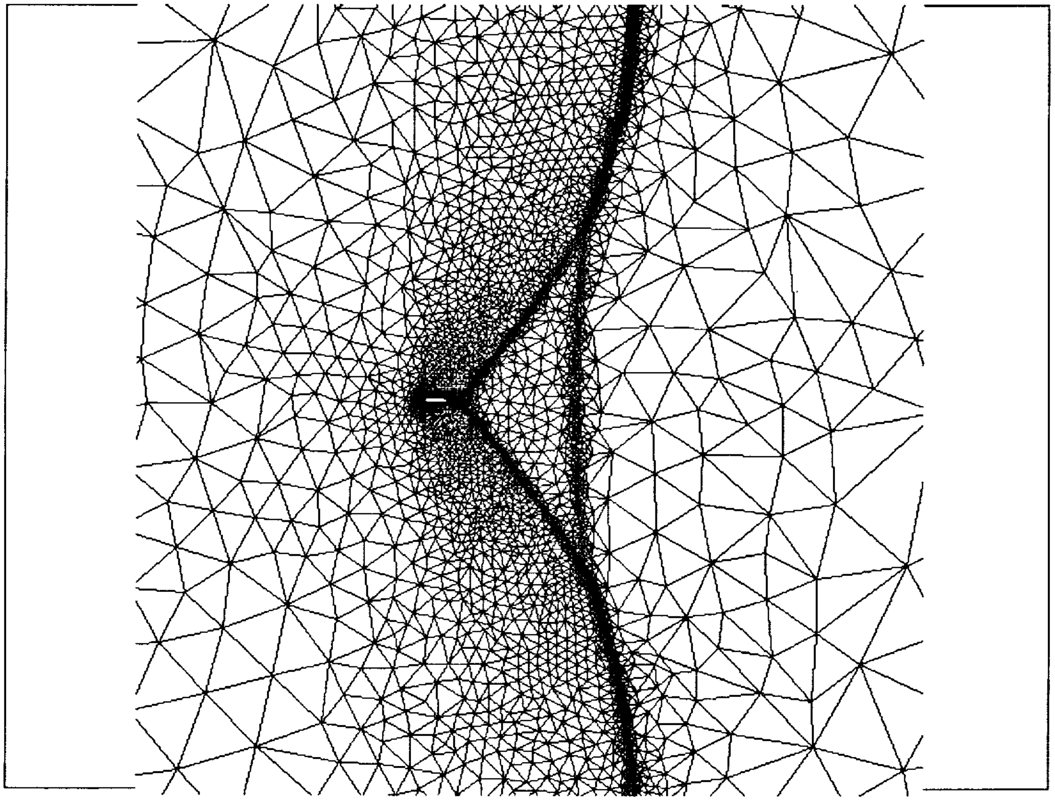
$$\lim_{n\to\infty} h_i \neq 0$$

Desirable limit properties can be enforced by multiplying by local length scale

$$\tilde{e} = \frac{l_i}{l_r} \tilde{\Delta} q \qquad \text{... etc}$$

Results From Corrected Adaptive Criteri





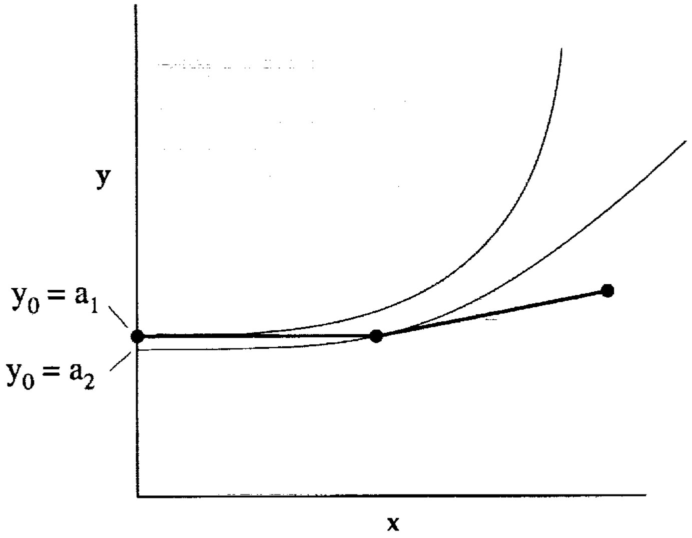
Lesson Learned

- Beware of adaptive criteria that refine "gradients" only and do not approach zero for all cells.

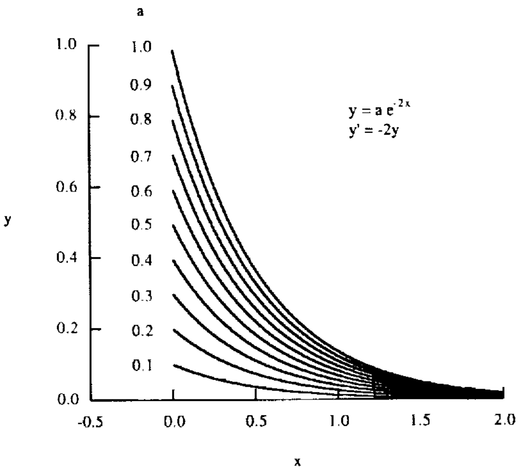
One-Dimensional O.D.E's

- Two-Point Boundary Value Problems
  - Babuska - Optimal grid spacing occurs when error is evenly distributed
  - Models elliptic and parabolic p.d.e. behavior
- Initial Value Problem
  - Models hyperbolic p.d.e. behavior
  - Must account for error propagation and accumulation

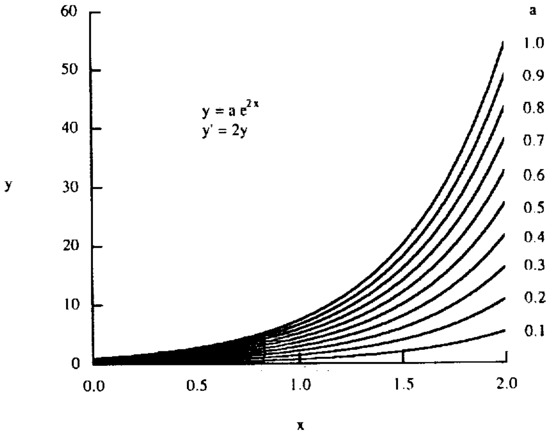
Error and I.V.P.'s



Model O.D.E.

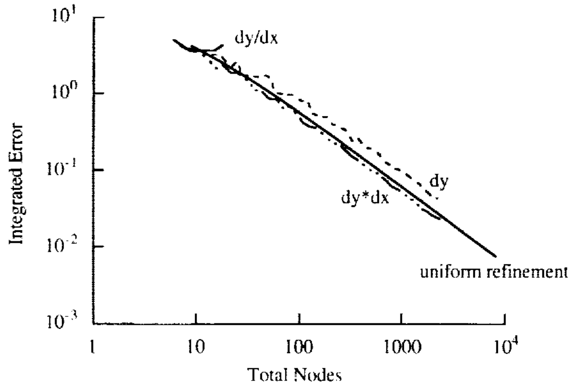


Model O.D.E.



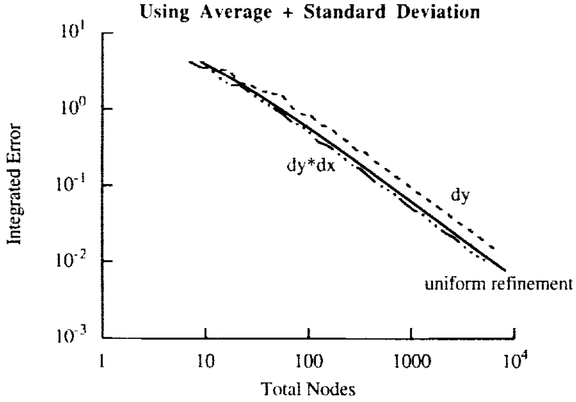
O.D.E. Adaptation

Point Removal and Addition
Using Average +- Standard Deviation

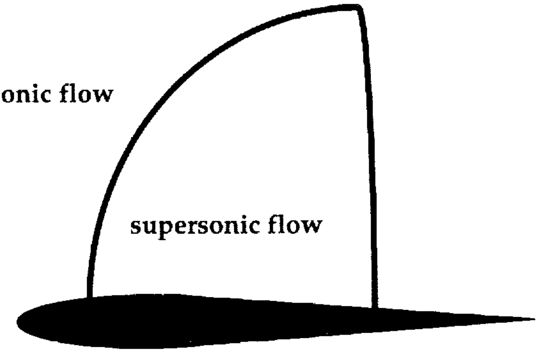


Model O.D.E.

Point Addition
Using Average + Standard Deviation



How do we adapt to transonic flow??



Discussion

- Adaptation criteria must approach zero in all cells as they are refined (like local error) to guarantee grid convergence
- Adapting to marching problems is not the same as for two-point boundary value problems
- Marching problems must take into account spatial stability + zone of influence

TIME-DEPENDENT GRID ADAPTATION FOR MESHES OF TRIANGLES AND TETRAHEDRA

RUSS D. RAUSCH
NATIONAL RESEARCH COUNCIL

MOTIVATION

- Unsteady CFD flow calculations are computationally expensive when compared to steady flow calculations
- Conflicting interests: We want adequate spatial & temporal accuracy but we don't want to pay the price (Excessive CPU time)
- The computational mesh drives the cost of CFD calculations and should be optimized for each flow condition. This suggests that solution algorithms should be closely tied with grid generation
- How do we optimize the mesh? Distribute the numerical error evenly throughout the mesh
- Use adaptive meshing to evenly distribute the spatial discretization errors
  - locally enrich in regions of relatively large errors
  - locally coarsen in regions of relatively small errors

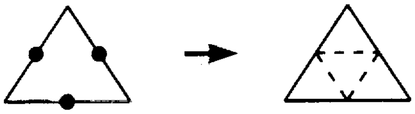
ENRICHMENT INDICATOR FOR THE SPATIAL ADAPTATION PROCEDURES

- Discretization errors generally occur where flow gradients are relatively large
  - shock waves
  - stagnation points
  - slip lines
  - expansion fans
- Magnitude of the gradient of density was used to detect relatively large flow gradients in 2D & 3D

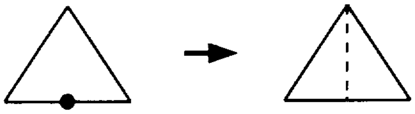
$$|\nabla\rho|$$

OVERVIEW OF 2D MESH ENRICHMENT STRATEGIES

- Type-4 enrichment element

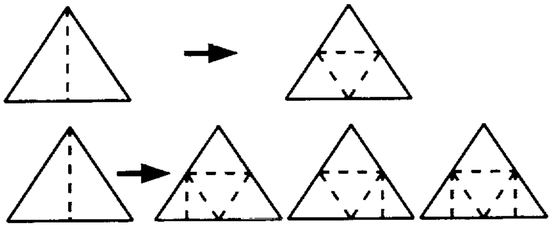


- Type-2 enrichment element

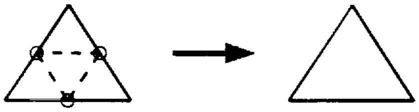


OVERVIEW OF 2D MESH ENRICHMENT STRATEGIES

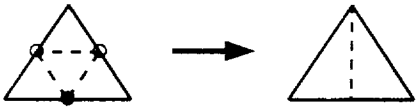
- Further enrichment of a type-2 enrichment element



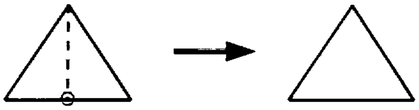
OVERVIEW OF 2D MESH COARSENING STRATEGIES



three nodes removed from a type-4 element



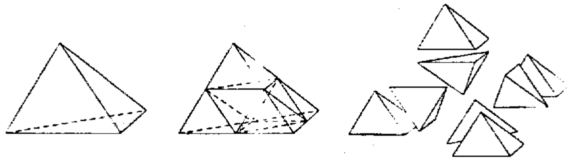
two nodes removed from a type-4 element



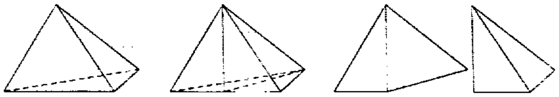
one node removed from a type-2 element

OVERVIEW OF 3D MESH ENRICHMENT STRATEGIES

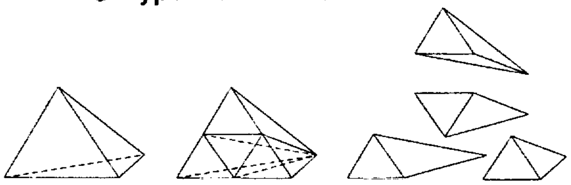
- Type-8 enrichment element



- Type-2 enrichment element

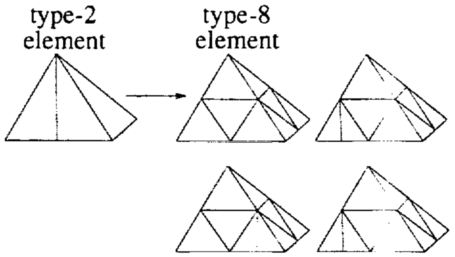


- Type-4 enrichment element



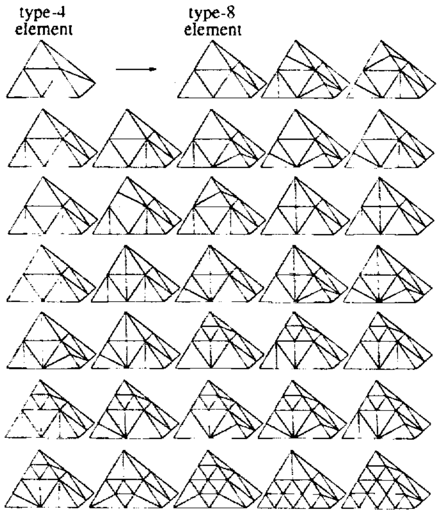
OVERVIEW OF 3D MESH ENRICHMENT STRATEGIES

- **Further enrichment of a type-2 enrichment element**



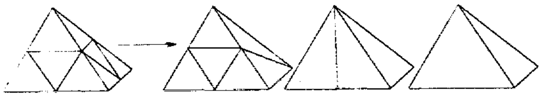
OVERVIEW OF 3D MESH ENRICHMENT STRATEGIES

- **Further enrichment of a type-4 enrichment element**

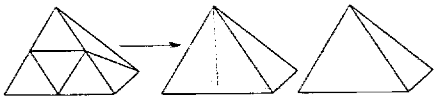


OVERVIEW OF 3D MESH COARSENING STRATEGIES

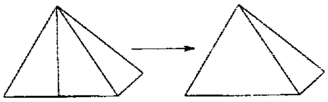
- Type-8 element coarsening



- Type-4 element coarsening



- Type-2 element coarsening



DESCRIPTION OF 2D & 3D UPWIND-TYPE EULER ALGORITHM OF BATINA

- Finite-volume spatial discretization on unstructured-grids

  - triangles in 2D
  - tetrahedra in 3D

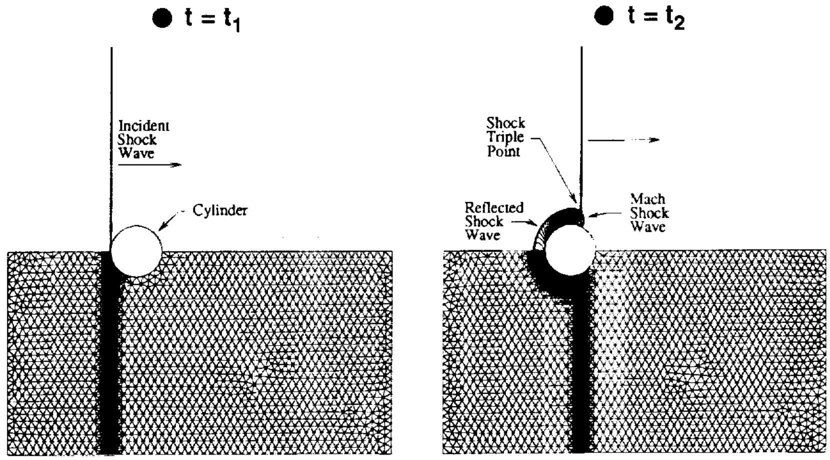
- flux vector splitting of van Leer
- Flux limiting to suppress oscillations near shock waves
- Time integration may be either explicit Runge-Kutta scheme or implicit Gauss-Seidel relaxation scheme
- Implicit scheme allows very large CFL numbers for rapid convergence to steady state
- Choose time step for unsteady calculations based on physics of problem rather than numerical stability

OVERVIEW OF SPATIAL ADAPTATION RESULTS

- Two dimensional case
  - Shock diffraction problem
- Three dimensional cases
  - ONERA M6 wing
  - Shock-tube problem

INSTANTANEOUS MESH AND DENSITY CONTOUR LINES FOR THE SHOCK DIFFRACTION PROBLEM

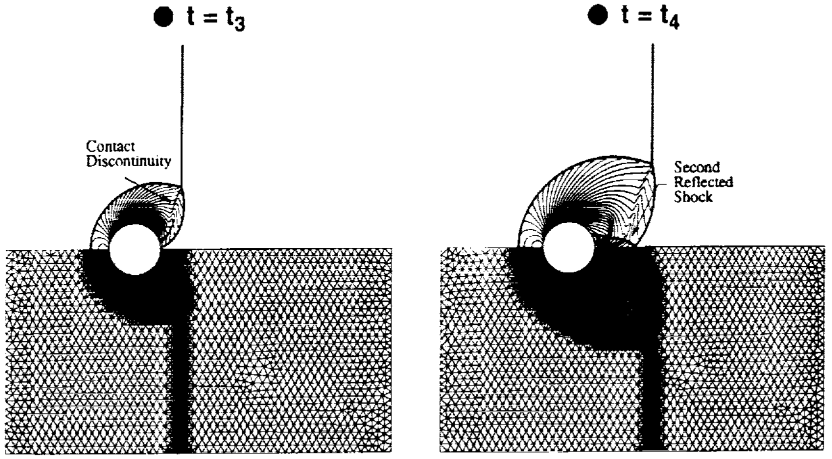
- $M_s = 2.81$
- $\Delta\rho = 0.2$



Rausch, 1992

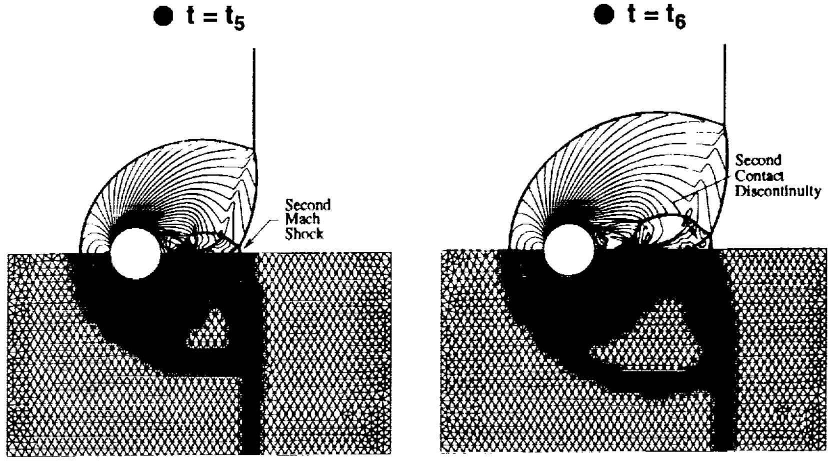
INSTANTANEOUS MESH AND DENSITY CONTOUR LINES FOR THE SHOCK DIFFRACTION PROBLEM

- ● $M_s = 2.81$
- ● $\Delta\rho = 0.2$



INSTANTANEOUS MESH AND DENSITY CONTOUR LINES FOR THE SHOCK DIFFRACTION PROBLEM

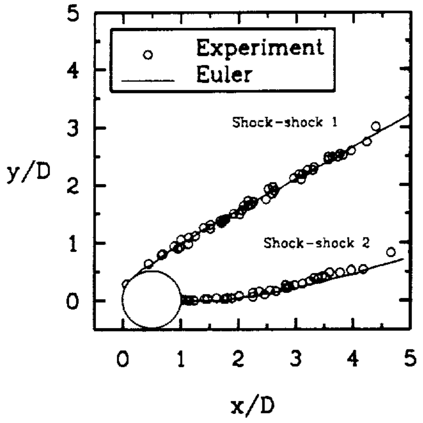
- ● $M_s = 2.81$
- ● $\Delta\rho = 0.2$



Rausch, 1992

COMPARISON OF SHOCK TRIPLE POINT LOCATIONS WITH EXPERIMENTAL DATA

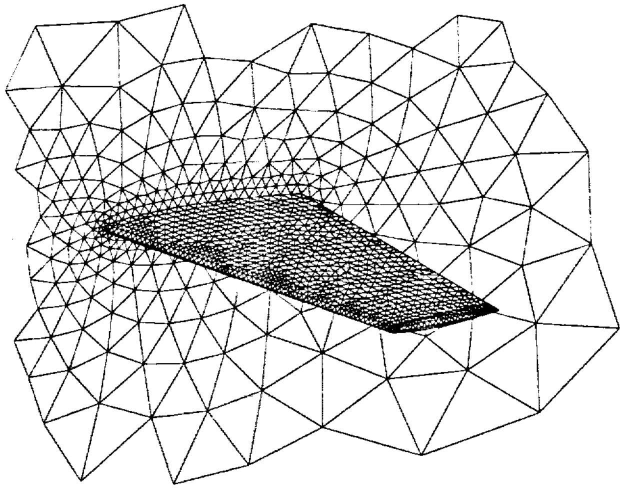
- **Experimental data by Bryson and Gross, *Journal of Fluid Mechanics*, vol. 10, pp. 1-16, 1961**



Rausch, 1992

PARTIAL VIEW OF THE SURFACE MESHE FOR THE SYMMETRY PLANE AND THE ONERA M6 WING

- **Total mesh has 46,516 tetrahedra and 8,824 nodes**

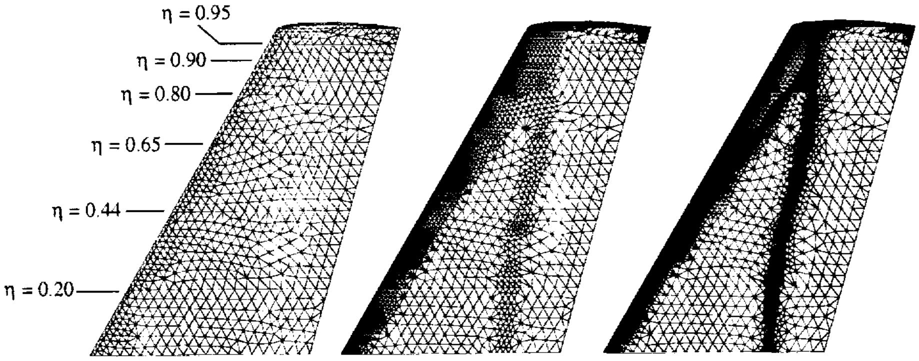


Rausch, 1992

COMPARISON OF UPPER SURFACE MESHES FOR THE ONERA M6 WING

- $M_\infty = 0.84$, $\alpha_o = 3.06°$

● Original mesh ● 1 level ● 2 levels

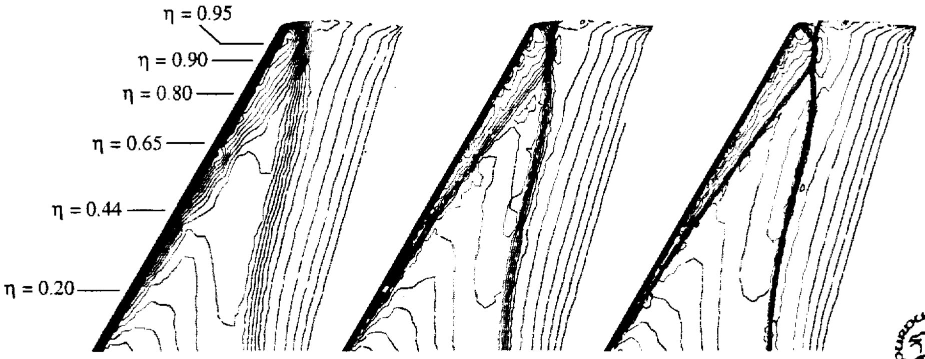


Rausch, 1992

COMPARISON OF UPPER SURFACE DENSITY CONTOUR LINES FOR THE ONERA M6 WING

- $M_\infty = 0.84$, $\alpha_o = 3.06°$
- $\Delta\rho = 0.025$

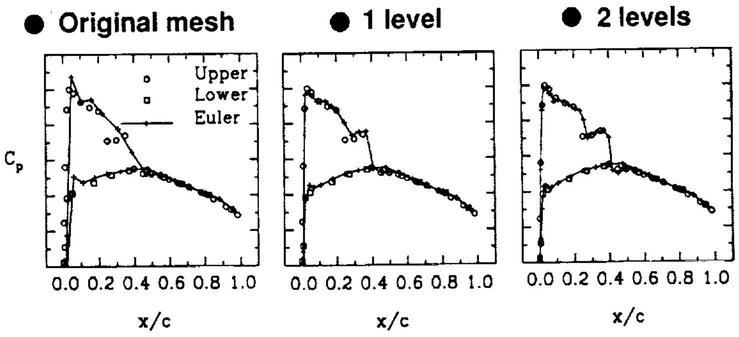
● Original mesh ● 1 level ● 2 levels



Rausch, 1992

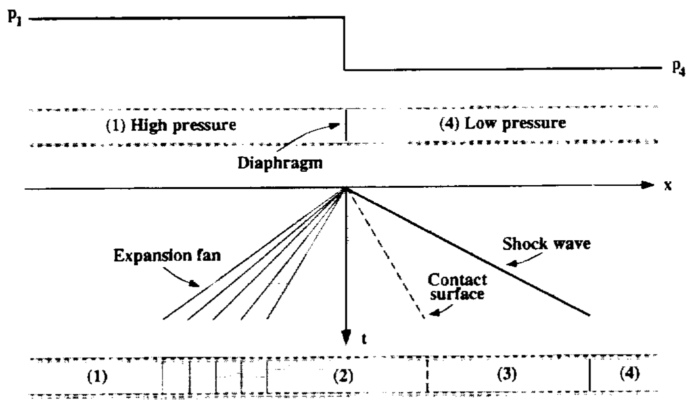
COMPARISON OF COEFFICIENT OF SURFACE PRESSURE FOR THE ONERA M6 WING

- ● $M_\infty = 0.84$, $\alpha_0 = 3.06^\circ$
- ● $\eta = 0.80$



Rausch, 1992

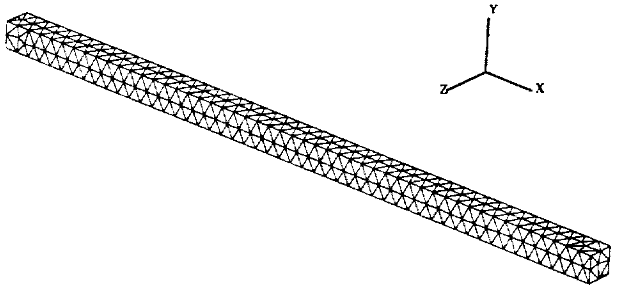
ILLUSTRATION OF THE SHOCK-TUBE PROBLEM



Rausch, 1992

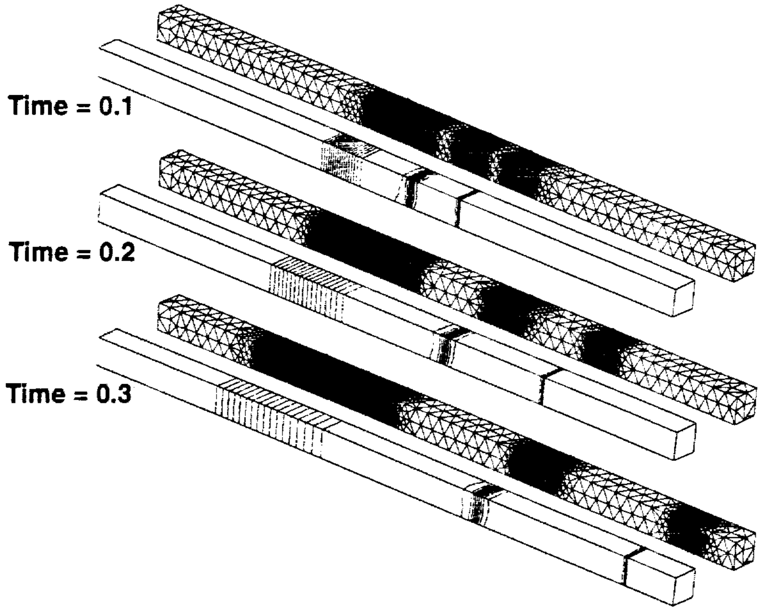
SURFACE MESH FOR THE SHOCK-TUBE PROBLEM

- Total mesh contains 562 nodes and 1,800 tetrahedra



Rausch, 1992

INSTANTANEOUS SURFACE MESH AND DENSITY CONTOUR LINES FOR THE SHOCK-TUBE PROBLEM

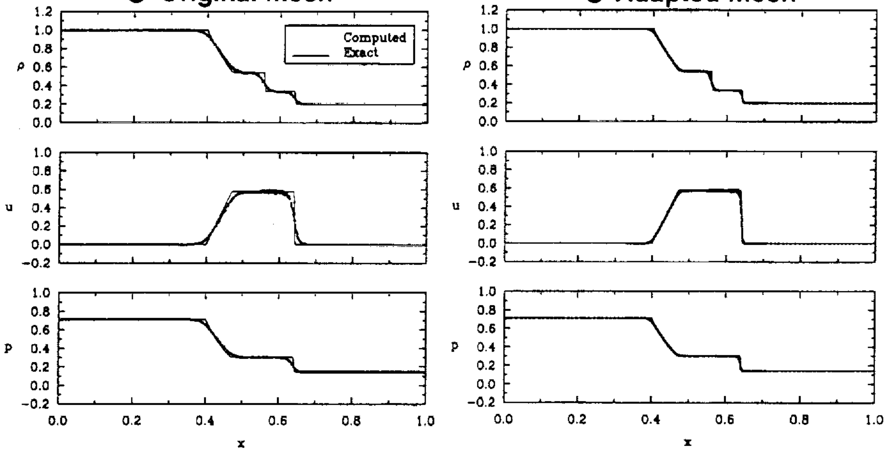


Rausch, 1992

COMPARISON OF THE VARIATION OF DENSITY, VELOCITY, AND PRESSURE THROUGHOUT THE SHOCK-TUBE

- Solution at time t = 0.1

- Original mesh
- Adapted mesh

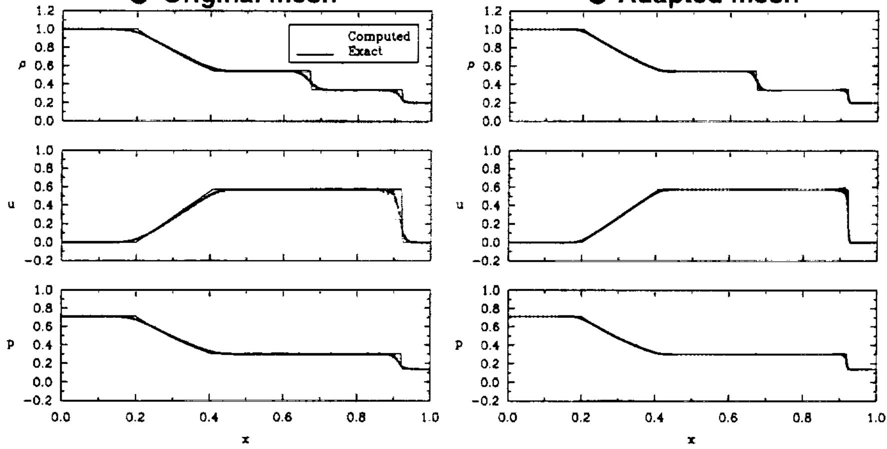


Rausch, 1992

COMPARISON OF THE VARIATION OF DENSITY, VELOCITY, AND PRESSURE THROUGHOUT THE SHOCK-TUBE

- Solution at time t = 0.3

- Original mesh
- Adapted mesh



Rausch, 1992

SUMMARY

- Final solution adapted mesh depends on the original mesh
    - adapted mesh cannot be coarser than the original mesh
- Enrichment/Coarsening procedures are robust for isotropic cells; however, enrichment of high aspect ratio cells may fail near boundary surfaces with relatively large curvature
- Enrichment indicator worked well for the cases shown, but in general requires user supervision for a more efficient solution

COMPUTATIONAL GEOMETRY ISSUES

MARY-ANNE POSENAU
NASA LANGLEY RESEARCH CENTER

OUTLINE

Computational Geometry - how it fits in

Survey - recent work

A Computational Geometry Approach - current work

COMPUTATIONAL GEOMETRY

The design and analysis of algorithms and data structures for the solution of geometric problems.

WHY COMPUTATIONAL GEOMETRY

Complexity

Bounds

Robustness

"This program takes 2 minutes to generate a grid for model X on workstation Y."

Questions:

Does the program always generate a grid?
How does the number of grid cells affect execution time?
What can be said about grid quality?

"O"-Notation

A function T(n) is O(f(n)) is there exist constants c and n_0 such that for all $n>n_0$, $T(n)\leq c\ f(n)$

Delaunay Triangulation - O(n log n)

- Shamos and Hoey - Divide and conquer
- Fortune - Sweepline
- Guibas, Knuth, Sharir - Randomized incremental

OPTIMALITY CRITERIA

The Constrained Delaunay Triangulation

- minimizes the largest circumcircle
- minimizes the largest min-containment circle
- maximizes minimum angle
- lexicographicaly maximizes list of angles, smallest to largest
- minimizes roughness as measured by Sobolev semi-norm
- guarantees a maximum principle
  for the discrete Laplacian approximation

OTHER OPTIMAL TRIANGULATIONS

Minimize max edge length - $O(n^2)$ Edelsbrunner, Tan

Greedy Triangulation - $O(n^2)$

Minimum weight triangulation

- not known to be NP-complete
- not known to be solvable in polynomial time
- variant is NP-complete
- approximations used

STEINER TRIANGULATION - RECENT RESULTS

Chew (89) - Range: [30°, 120°]

- size optimal among all uniform meshes

Baker, Grosse, Rafferty (88) - Range: [13°, 90°]

- aspect ratio < 4.6

Bern, Eppstein, Gilbert (90) - Range: [36°- 80°]

- aspect ratio < 5

Ruppert (93) - Range: [alpha, Pi-2 alpha]

- $\left|\frac{1}{\sin \text{alpha}}\right| <$ aspect ratio $< \left|\frac{1}{\sin 2\text{alpha}}\right|$
- size optimal within a constant C_{alpha}

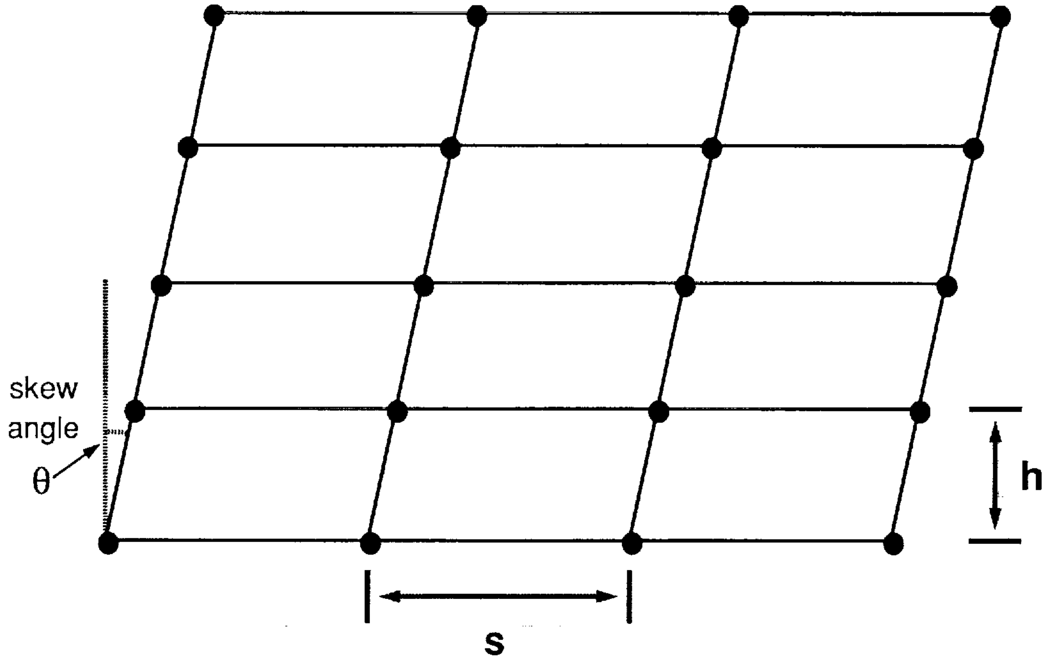
HIGH ASPECT RATIO TRIANGULATIONS

Delaunay triangulation can be unsuitable for high aspect ratio, body-conforming triangulations.

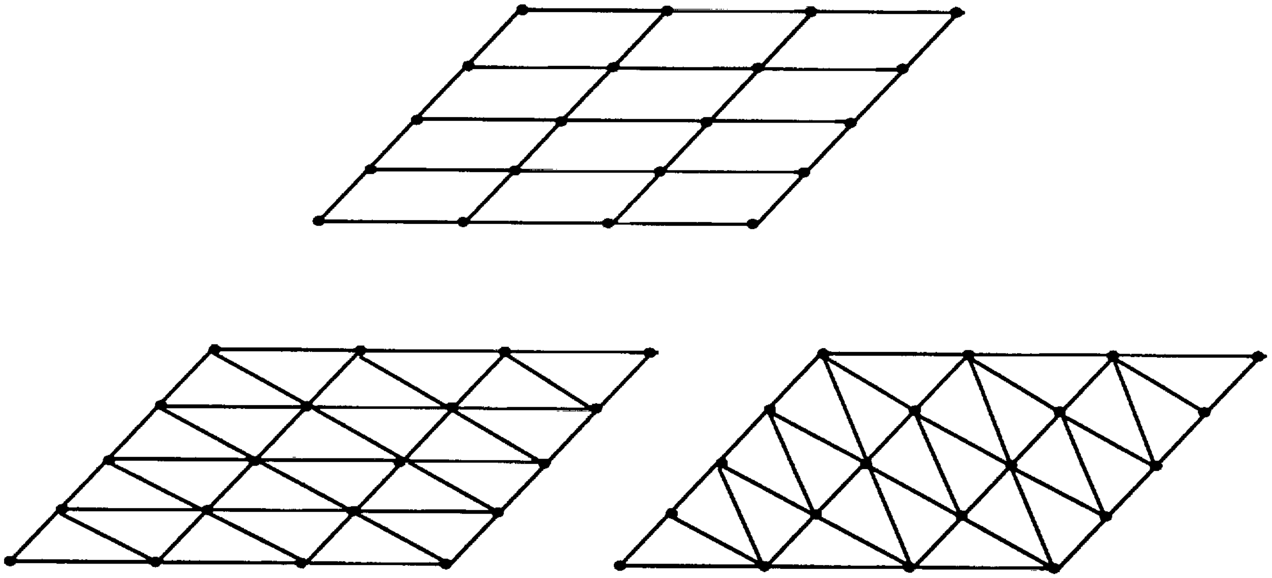
Robust, efficient, global algorithms are in need.

Computational geometers are not looking at this problem.

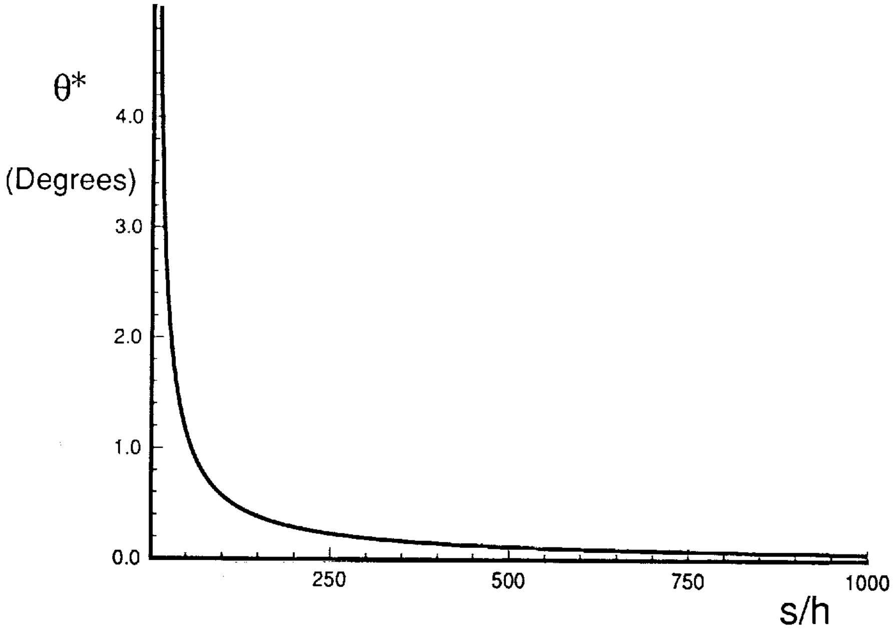
SKEWED STRUCTURED GRID



DELAUNAY REALIZABILITY

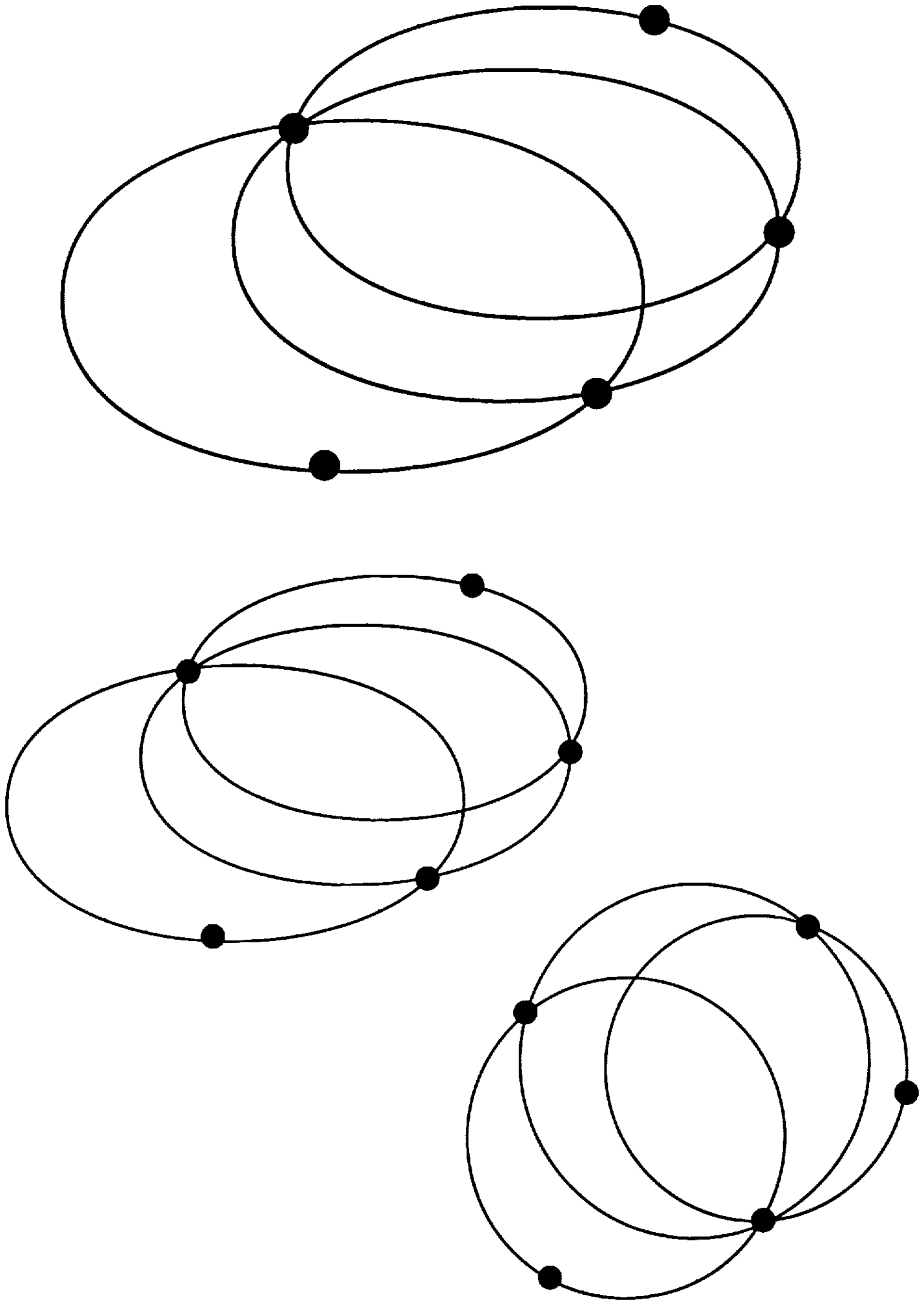


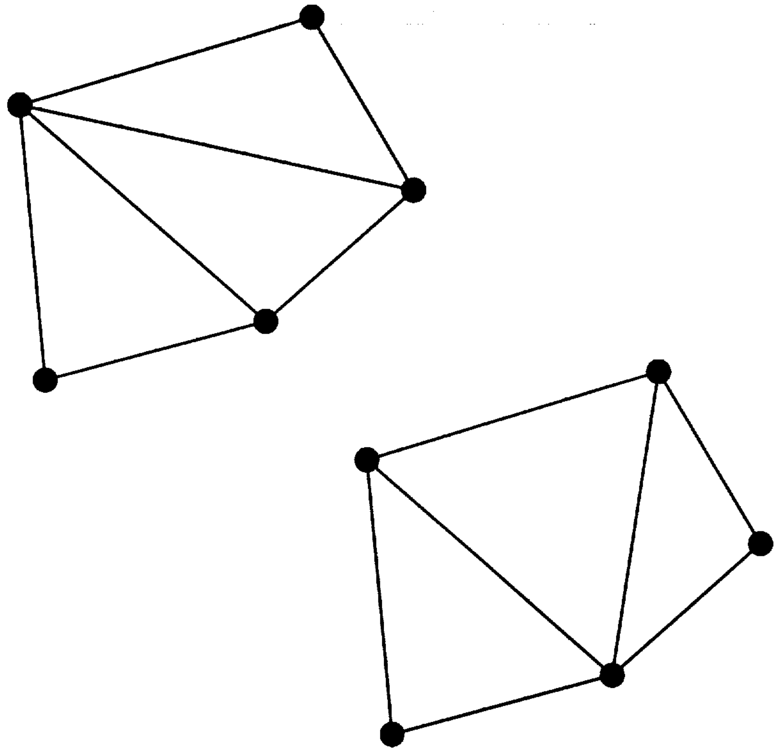
DELAUNAY ANGLE CUT-OFF vs. ASPECT RATIO



CONVEX DISTANCE FUNCTIONS
Chew, 1985

Change the concept of circumcircle to that of a convex distance function

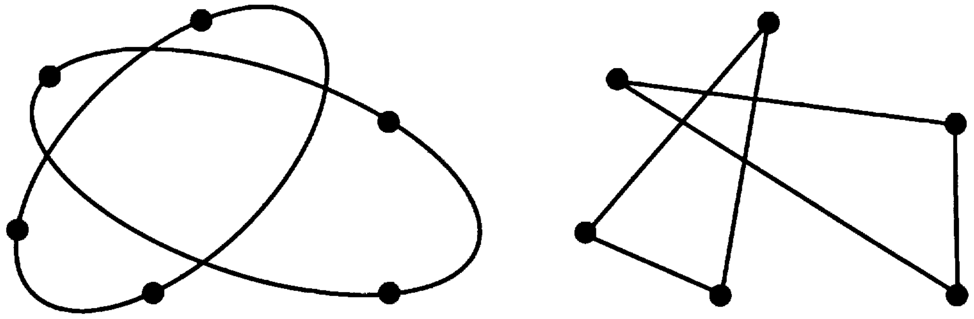




ISSUES

Generalize to a distance function which can vary throughout the plane.

Avoid ambiguous cases.



CONVEX BODY PROJECTION AND CONVEX HULL

Brown, 1979
Edelsbrunner, 1987

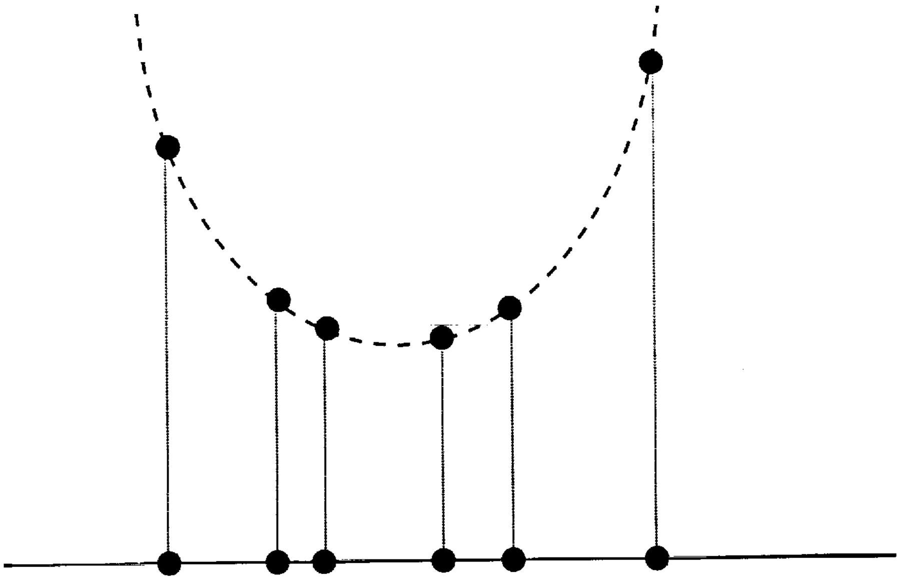
Project points from the plane to a paraboloid using parallel projection.

Find the convex hull of the 3D point set (all points will be on the convex hull).

The lower hull, projected back to the plane, will give the Delaunay triangulation of the point set in the plane.

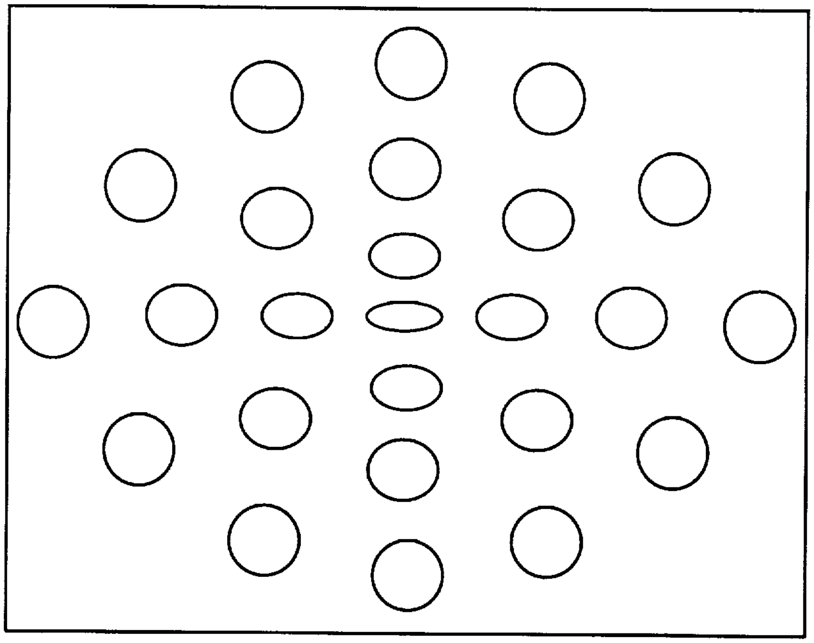
Notes: One convex body handles entire domain.
Shifting the body to a new location gives the same result.

CONVEX BODY PROJECTION AND CONVEX HULL
Brown, 1979
Edelsbrunner, 1987



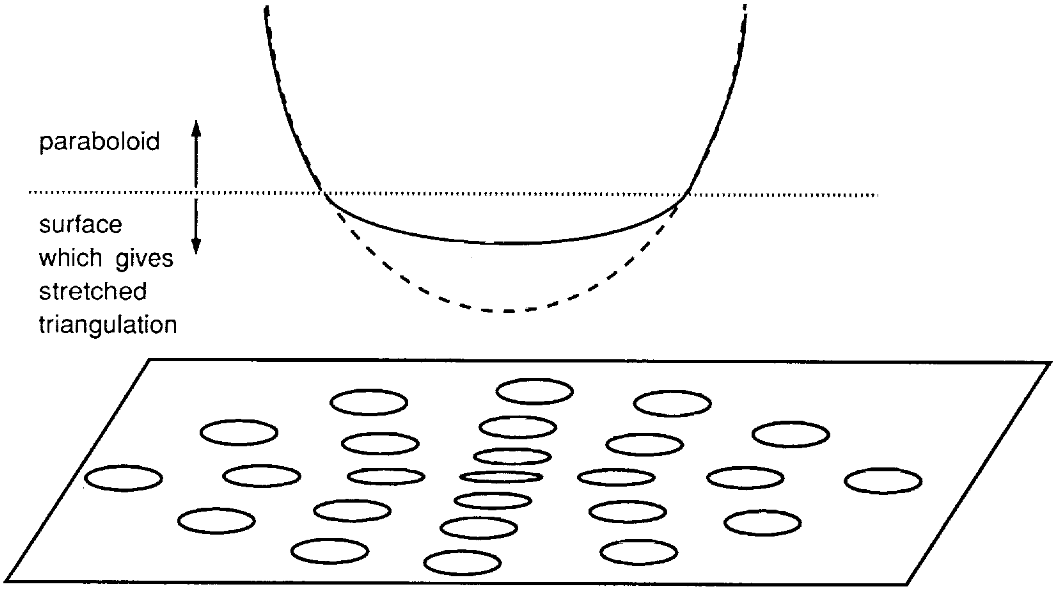
STRETCHED TRIANGULATIONS

Step 1a: Model simple stretching.



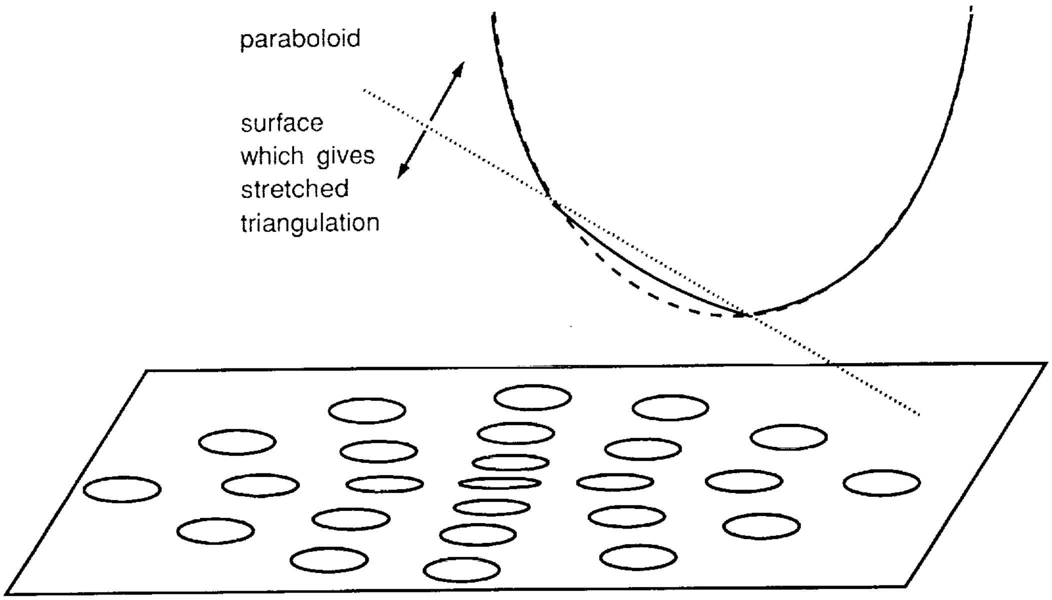
STRETCHED TRIANGULATIONS

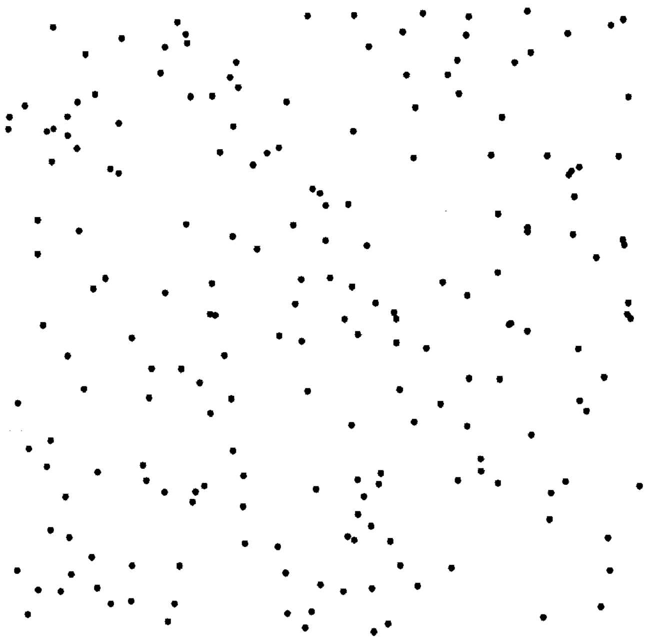
Step 1b: Design convex surface which will produce desired stretched triangulation.



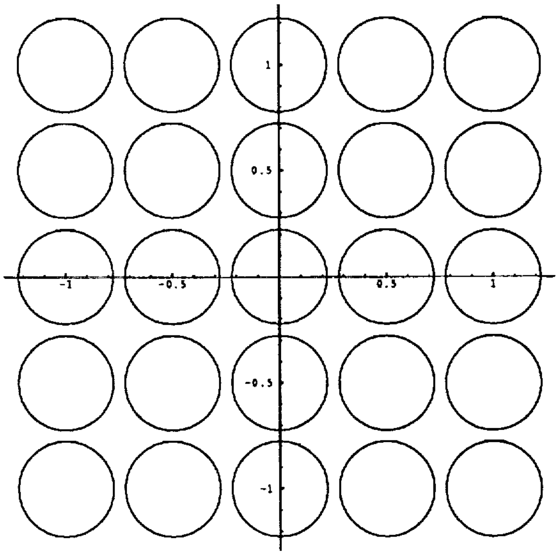
STRETCHED TRIANGULATIONS

Note: Body will not be "shift invariant".

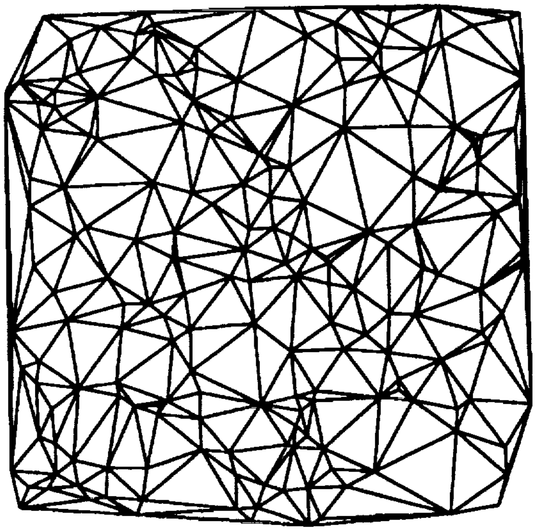




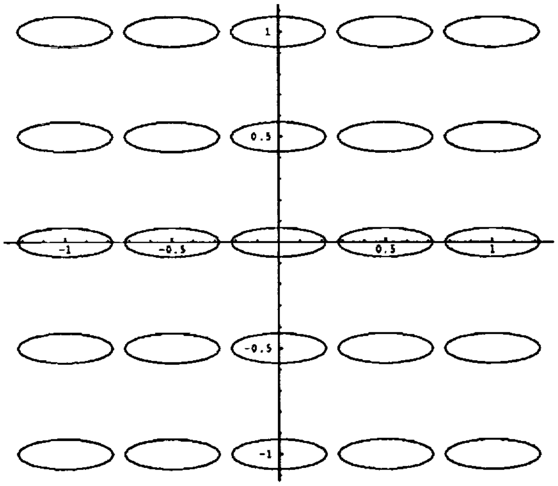
Test data used for all examples.



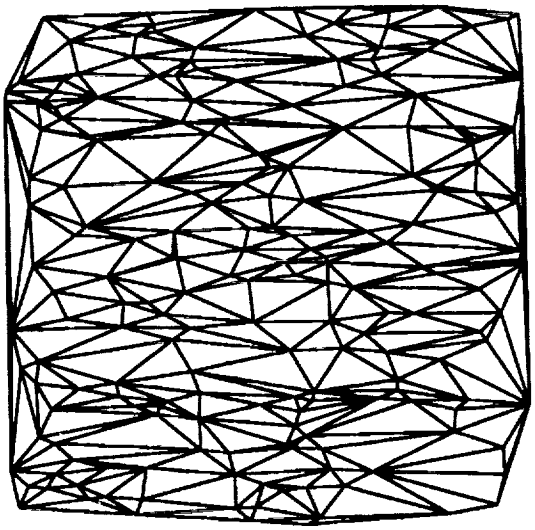
Circumshapes derived from paraboloid $x^2 + y^2$



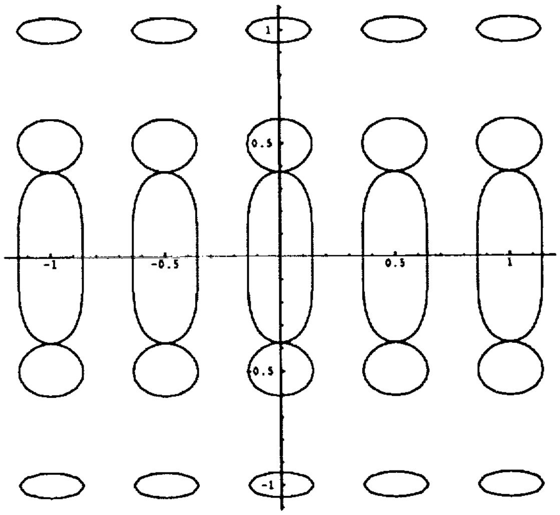
Triangulation derived from paraboloid $x^2 + y^2$
(Delaunay triangulation)



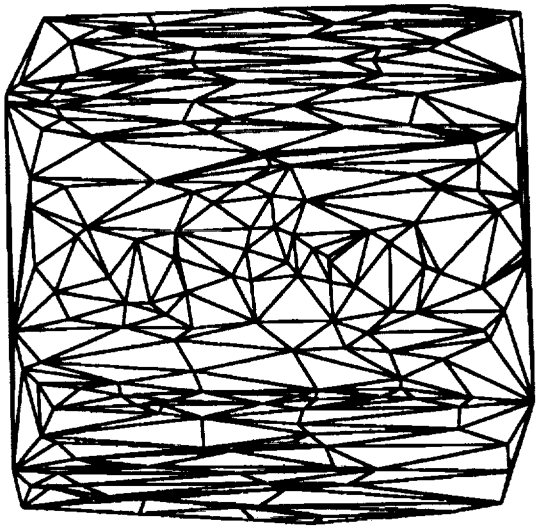
Circumshapes derived from $x^2 + 10y^2$, $\delta = 0.05$



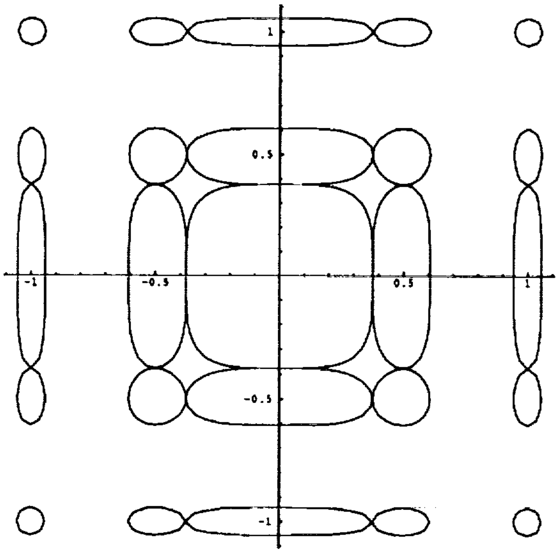
Triangulation derived from $x^2 + 10y^2$



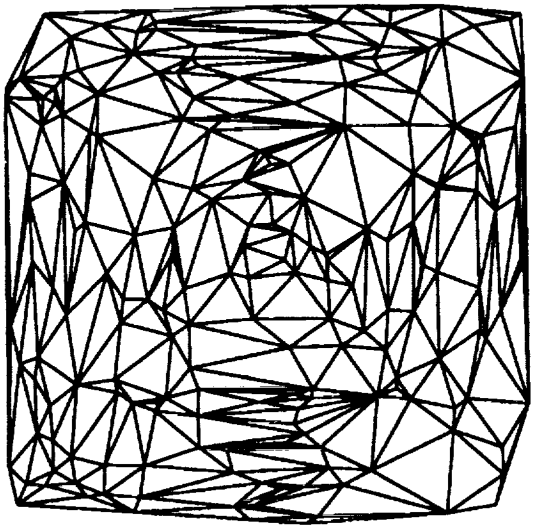
Circumshapes derived from $x^2 + y^4$, $\delta = 0.02$



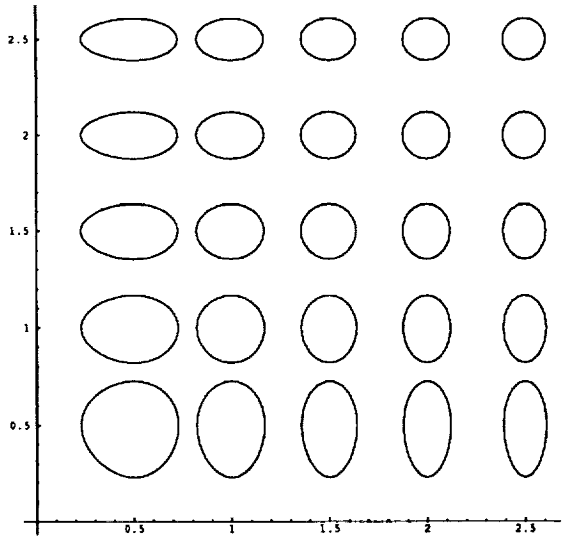
Triangulation derived from $x^2 + y^4$



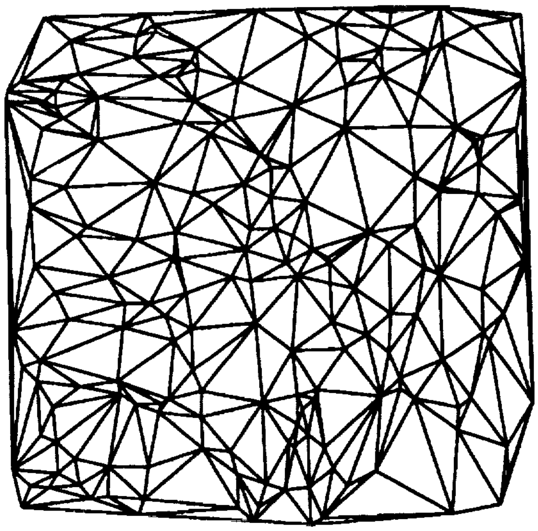
Circumshapes derived from $x^4 + y^4$, $\delta = 0.02$



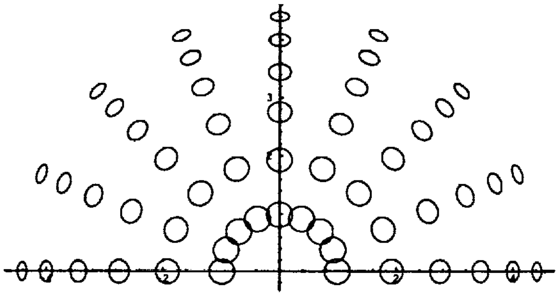
Triangulation derived from $x^4 + y^4$



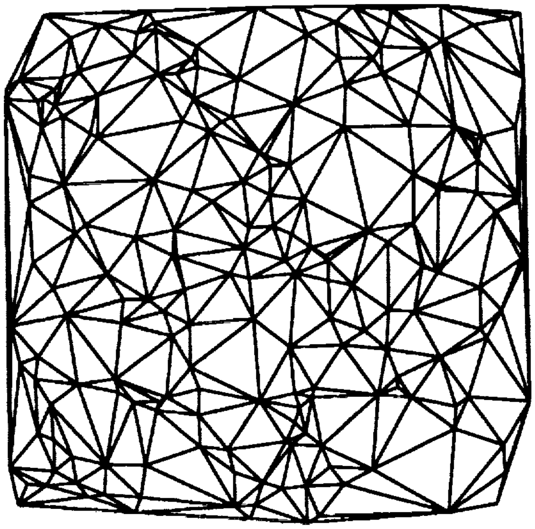
Circumshapes derived from $x^3 + y^3$, $\delta = 0.09$



Triangulation derived from $x^3 + y^3$



Circumshapes predicted from perspective projection, $z_{proj} = -100$, $\delta = 0.05$



Triangulation derived from perspective projection, $z_{proj} = -100$

CONCLUSIONS

Benefits of computational geometry - guarantees of

grid quality

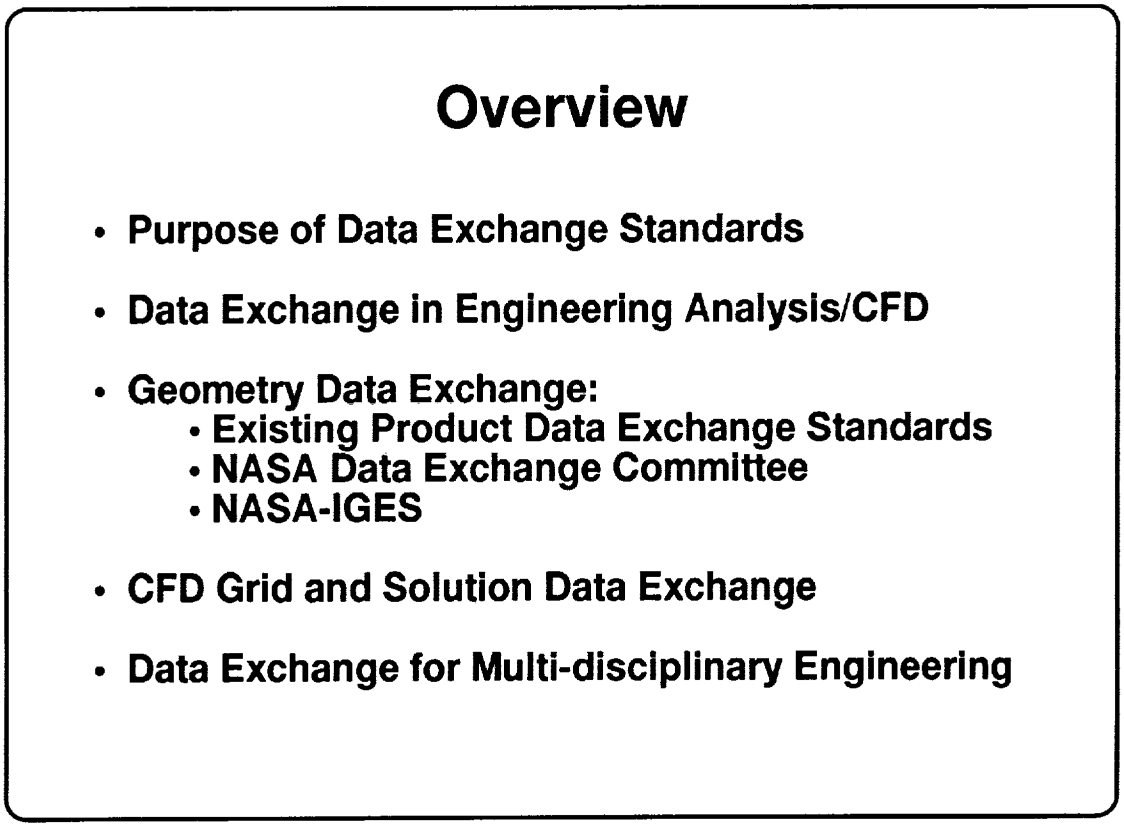
efficient algorithms

Many efficient triangulation algorithms are available,
but high aspect ratio triangulations are not among them.

Interdisciplinary cooperation will benefit grid generation
and computational geometry.

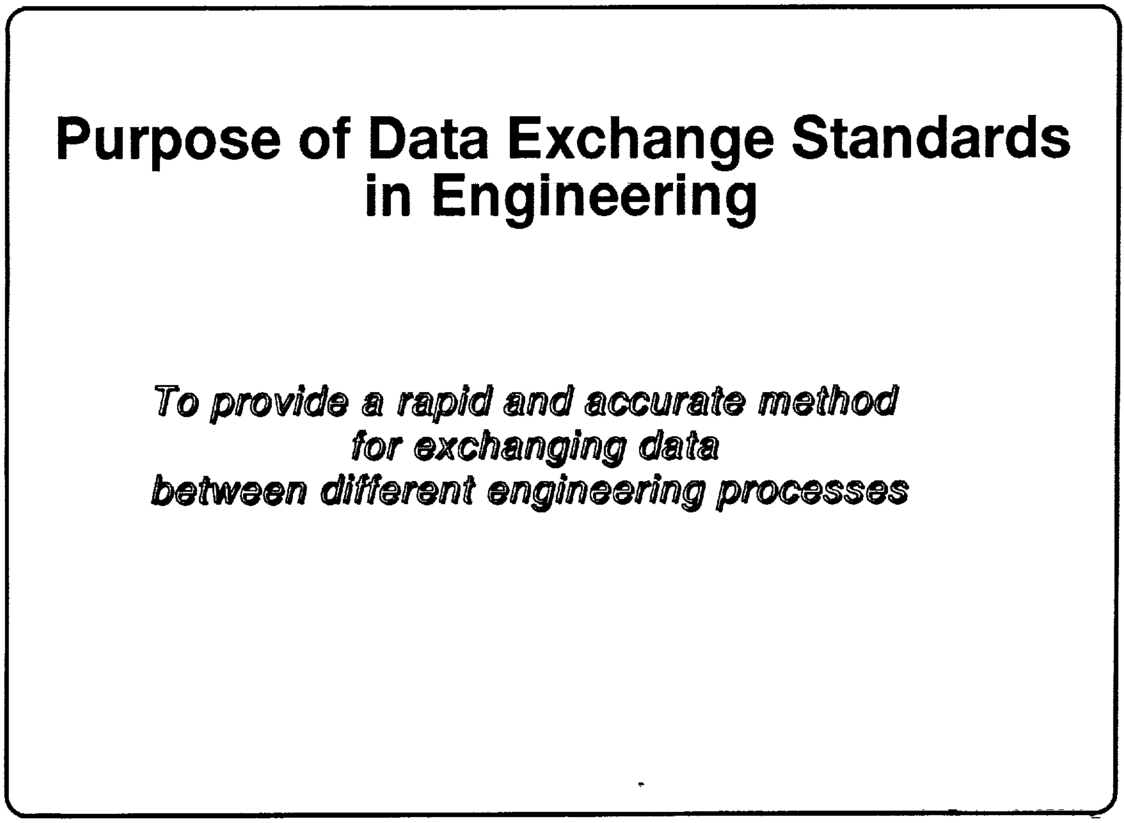
NASA DATA EXCHANGE STANDARDS FOR COMPUTATIONAL FLUID DYNAMICS

MATTHEW BLAKE
NASA AMES RESEARCH CENTER



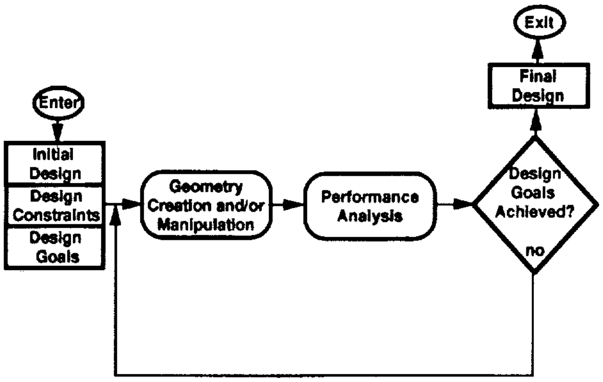
Overview

- Purpose of Data Exchange Standards
- Data Exchange in Engineering Analysis/CFD
- Geometry Data Exchange:
  - Existing Product Data Exchange Standards
  - NASA Data Exchange Committee
  - NASA-IGES
- CFD Grid and Solution Data Exchange
- Data Exchange for Multi-disciplinary Engineering

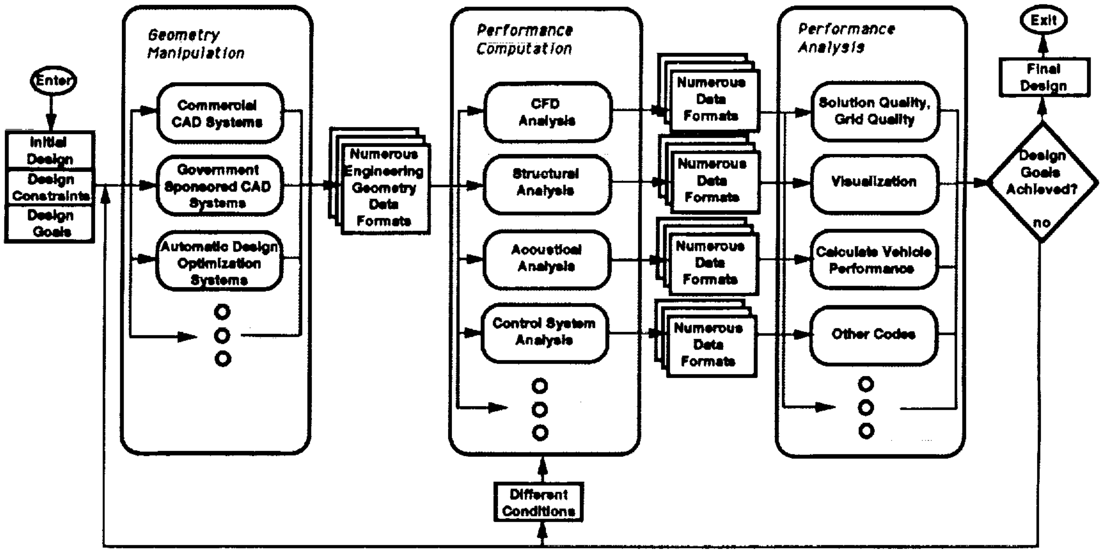


Purpose of Data Exchange Standards in Engineering

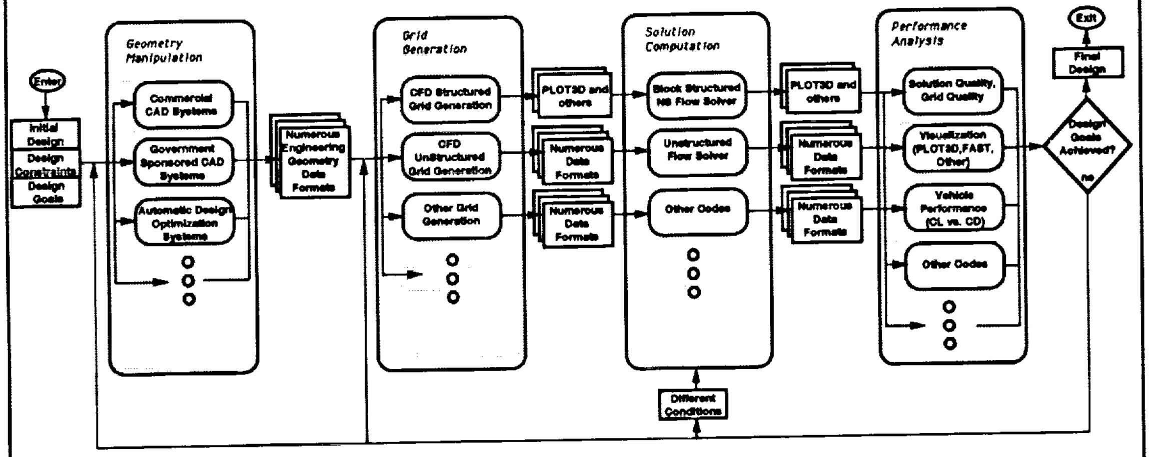
To provide a rapid and accurate method
for exchanging data
between different engineering processes



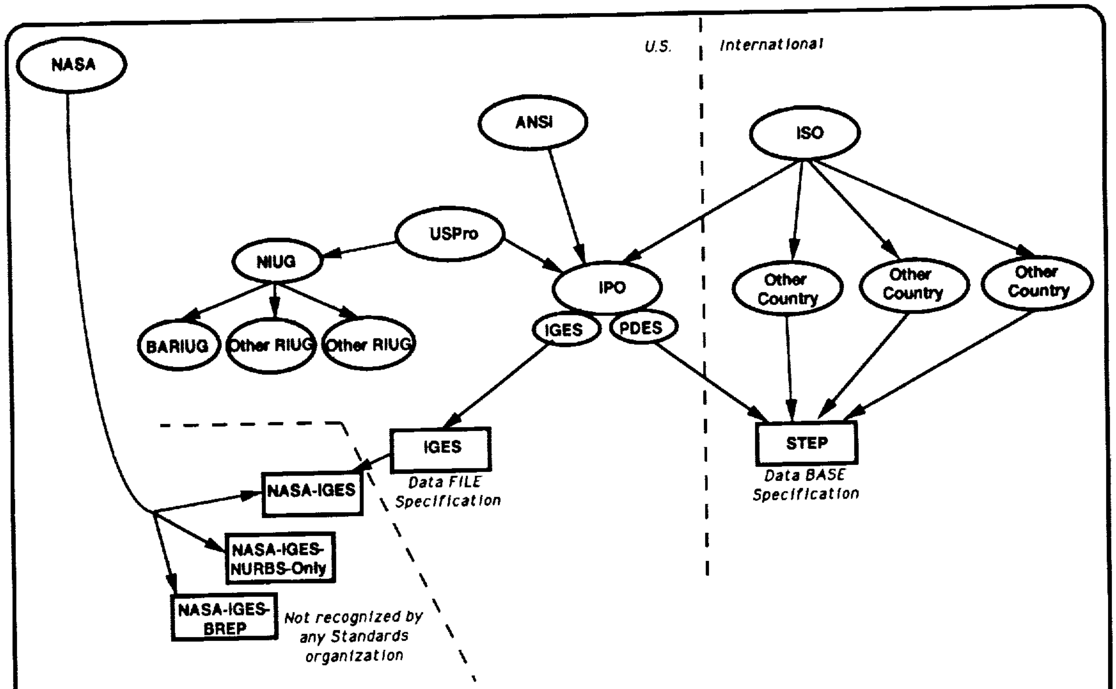
ENGINEERING ANALYSIS IN THE DESIGN CYCLE



ENGINEERING ANALYSIS IN THE DESIGN PROCESS



ENGINEERING DESIGN PROCESS COMPUTATIONAL FLUID DYNAMICS



U.S. and International Standards Organizations Related to Product Data

US & International Standards Organizations and Acronyms Related to Product Data

Organizations:

ISO	International Standards Organization
ANSI	American National Standards Institute
USPRO	U.S. Product Data Association
IPO	IGES / PDES Organization
PDES	Product Data Exchange Using STEP
NIUG	National IGES User Group
BARIUG	Bay Area Regional IGES User Group

Documents:

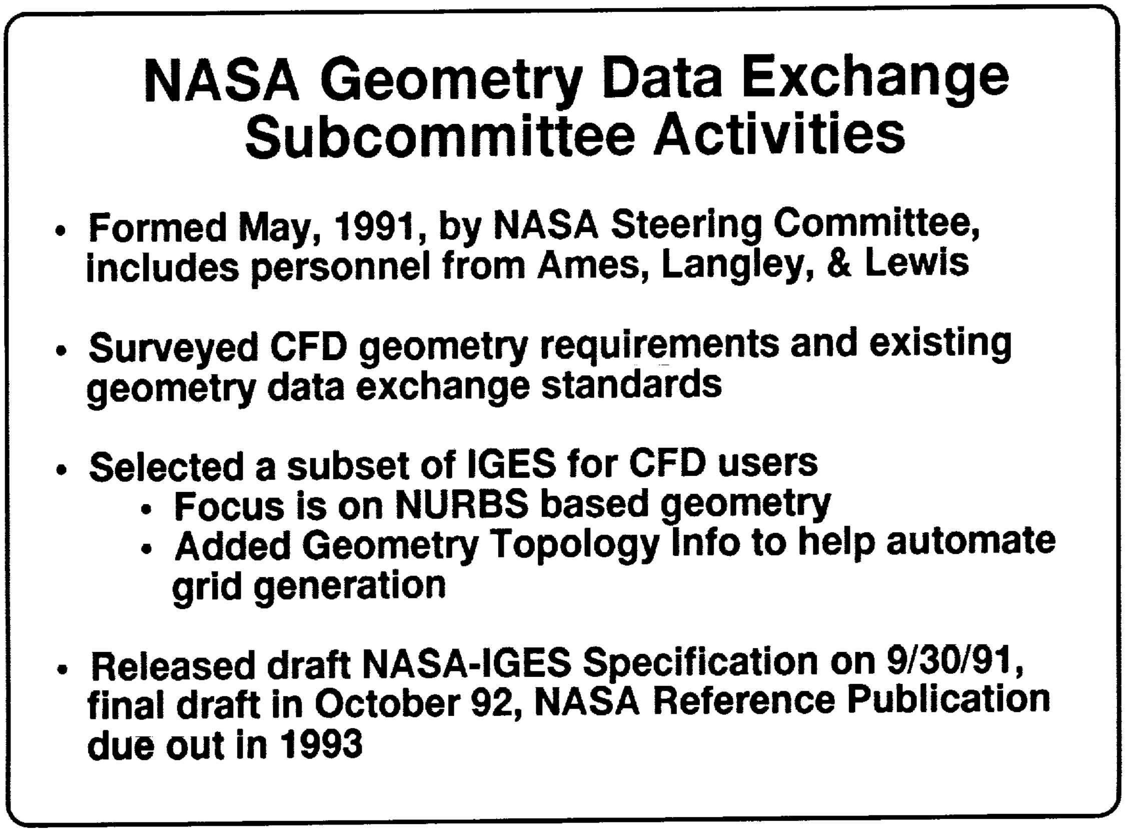
IGES	Initial Graphics Exchange Specification
STEP	Standard for the Exchange of Product Data
NASA-IGES	NASA subset of IGES
NASA-IGES-NURBS-Only:	NURBS only subset of NASA-IGES
NASA-IGES-BREP:	NURBS only geometry with B-Rrep topo. info
Superpatch	same as NASA-IGES-BREP

Other:

NURBS	Non Uniform Rational B-Splines
B-rep	Boundary Representation method for geometry topology

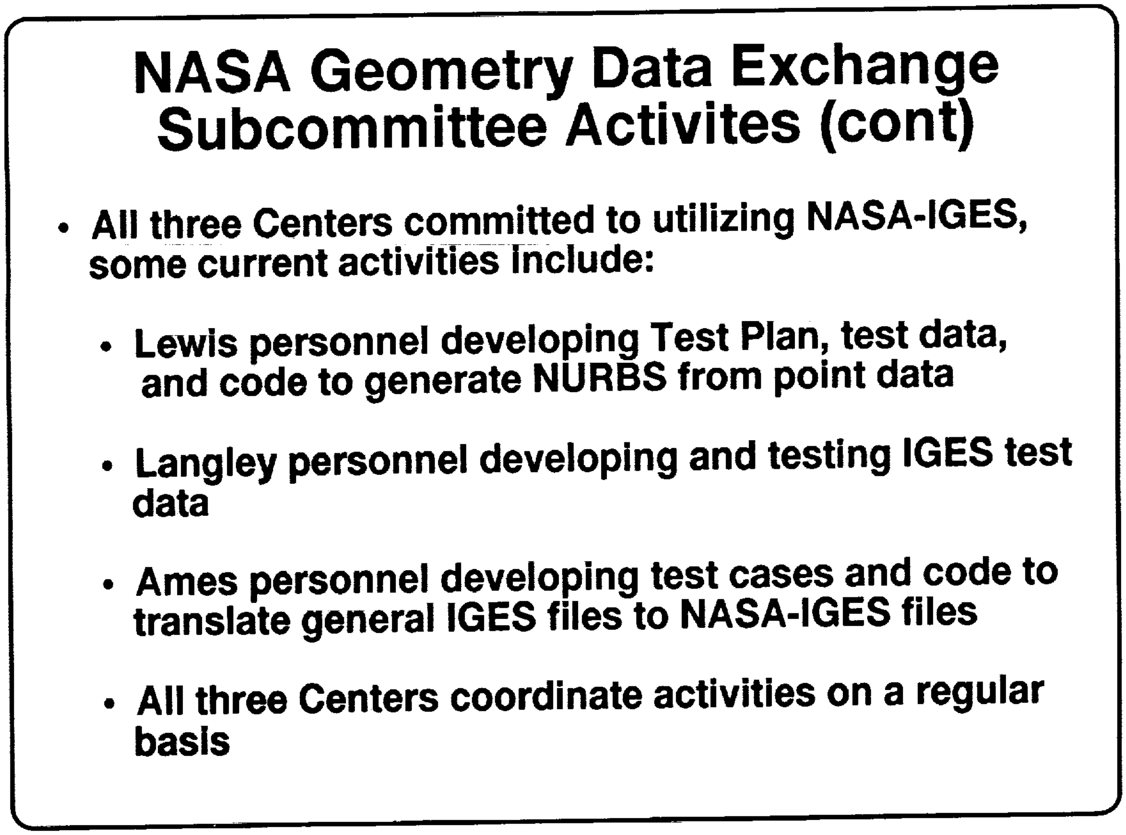
IGES Description

- Currently the most widely used method for product data exchange (including geometry)
- Large data file specification for all product information, superset of info, many ways to represent one item
- Version 4.0: Supported by all(?) CAD vendors
- Version 5.1: Current, supported by many vendors, includes NASA-IGES entities
- Version 5.2: Includes Open Shell (B-rep) in "grey pages", no vendor support yet, due out middle 1993
- Version 6.0: Final version, due in 1994



NASA Geometry Data Exchange Subcommittee Activities

- Formed May, 1991, by NASA Steering Committee, includes personnel from Ames, Langley, & Lewis
- Surveyed CFD geometry requirements and existing geometry data exchange standards
- Selected a subset of IGES for CFD users
  - Focus is on NURBS based geometry
  - Added Geometry Topology Info to help automate grid generation
- Released draft NASA-IGES Specification on 9/30/91, final draft in October 92, NASA Reference Publication due out in 1993



NASA Geometry Data Exchange Subcommittee Activites (cont)

- All three Centers committed to utilizing NASA-IGES, some current activities include:
  - Lewis personnel developing Test Plan, test data, and code to generate NURBS from point data
  - Langley personnel developing and testing IGES test data
  - Ames personnel developing test cases and code to translate general IGES files to NASA-IGES files
  - All three Centers coordinate activities on a regular basis

NASA Geometry Data Exchange Specification for CFD (NASA-IGES)

- Written for use by CFD scientists and engineers as well as CAD vendors
- Includes mathematical formulation of each type of geometric representation
- Includes an abstract representation of the database requirements for each entity
- Appendix contains the IGES protocol for NASA-IGES and NASA-IGES-NURBS-ONLY

Geometry Topology: NASA-IGES-BREP / Superpatch

- Provides connectivity/topology information for the curve and surface geometry entities
- Allows grid generation software to traverse the geometry so the grid can be constructed independent of surface layout choices made by the original designer
- Supplies important information for development of automated grid generation software
- Similar to Boundary Representation (B-rep) solid modeling technique

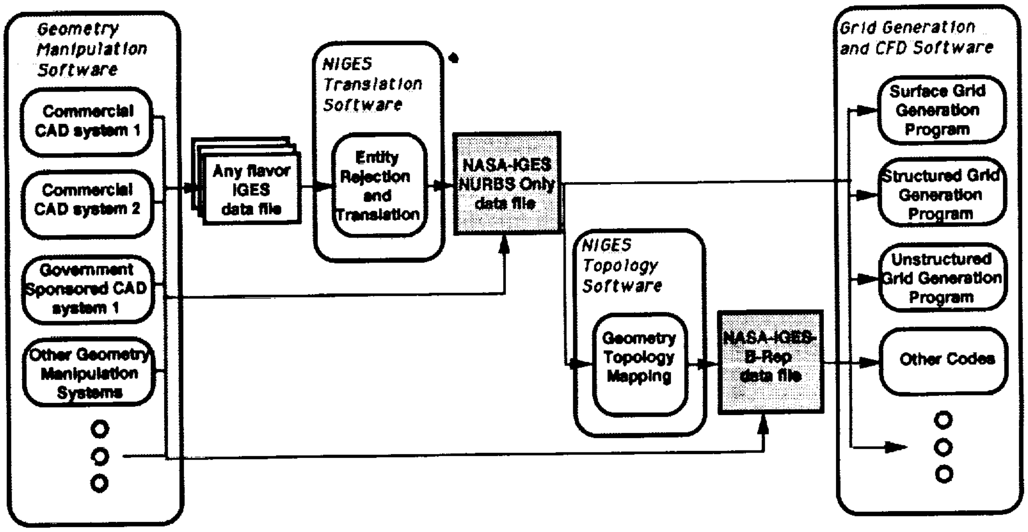
NASA-IGES ENTITIES

- NASA-IGES-NURBS-ONLY Geometry Entities:
  - Entity 126: Rational B-Spline Curve
  - Entity 128: Rational B-Spline Surface
  - Entity 141: Boundary
  - Entity 142: Curve on a Parametric Surface
  - Entity 143: Bounded Surface
  - Entity 102: Composite Curve
  - Entity 124: Transformation Matrix
- Other Geometry Entities Allowed in NASA-IGES:
  - Entity 100: Circular Arc
  - Entity 104: Conic Arc
  - Entity 106: Copious Data
  - Entity 110: Line
  - Entity 116: Point
- Non-Geometry Entities:
  - Entity 0: Null Entity
  - Entity 212: General Note
  - Entity 308: Subfigure Definition
  - Entity 314: Color Definition
  - Entity 402: Associativity Instance
  - Entity 406, Form 15: Name
  - Entity 408: Singular Subfigure Instance

NASA-IGES-BREP ENTITIES

- Topology Entities:
  - Entity 186: Manifold Solid B-Rep Object
  - Entity 514: Shell, Closed and Open
  - Entity 510: Face
  - Entity 508: Loop
  - Entity 504: Edge List
  - Entity 502: Vertex List
- Geometry Entities:
  - Entity 126: Rational B-Spline Curve
  - Entity 128: Rational B-Spline Surface
  - Entity 102: Composite Curve
  - Entity 124: Transformation Matrix
- Non-Geometry Entities:
  - Entity 0: Null Entity
  - Entity 212: General Note
  - Entity 314: Color Definition
  - Entity 402: Associativity Instance
  - Entity 406, Form 15: Name

CFD Geometry Data Exchange Utilizing NASA-IGES and NASA-IGES-BREP Data Files



Geometry Manipulation Software

Commercial CAD system 1

Commercial CAD system 2

Government Sponsored CAD system 1

Other Geometry Manipulation Systems

Any flavor IGES data file

NIGES Translation Software

Entity Rejection and Translation

NASA-IGES NURBS Only data file

NIGES Topology Software

Geometry Topology Mapping

NASA-IGES-B-Rep data file

Grid Generation and CFD Software

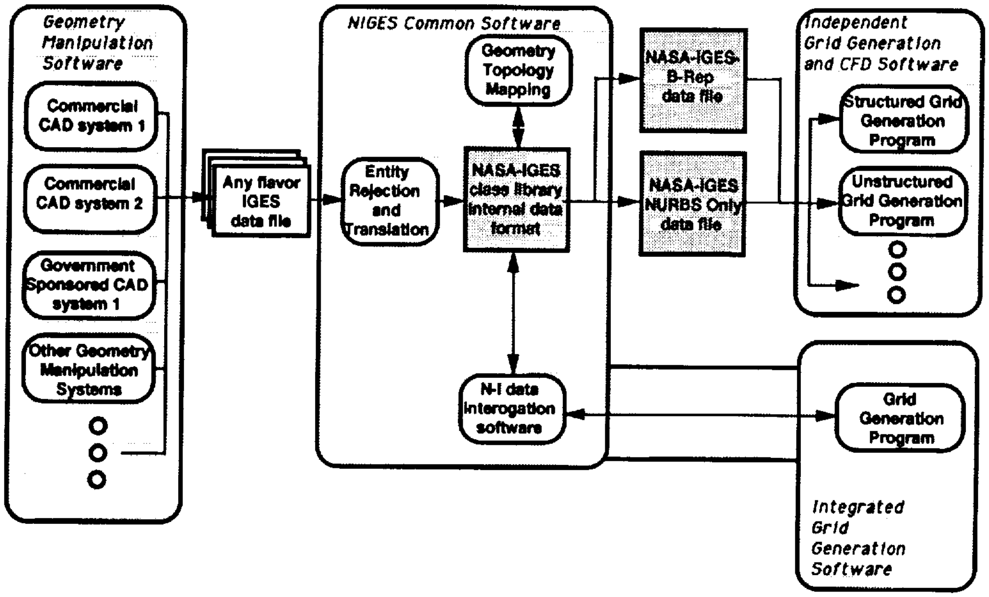
Surface Grid Generation Program

Structured Grid Generation Program

Unstructured Grid Generation Program

Other Codes

CFD Geometry Data Exchange
Utilizing
NASA-IGES and NASA-IGES-BREP
Data Files and Class Library


Geometry Manipulation Software
Commercial CAD system 1
Commercial CAD system 2
Government Sponsored CAD system 1
Other Geometry Manipulation Systems
O
O
O
Any flavor IGES data file
NIGES Common Software
Geometry Topology Mapping
Entity Rejection and Translation
NASA-IGES class library internal data format
N-I data interogation software
NASA-IGES-B-Rep data file
NASA-IGES NURBS Only data file
Independent Grid Generation and CFD Software
Structured Grid Generation Program
Unstructured Grid Generation Program
O
O
O
Grid Generation Program
Integrated Grid Generation Software

CFD Grid & Solution Data Standards: Design Goals

- **Include enough information to reconstruct connectivity information used by any specific application**
  HELP>>> Fill in the supplied table or provide documentation of your grid & solution data requirements

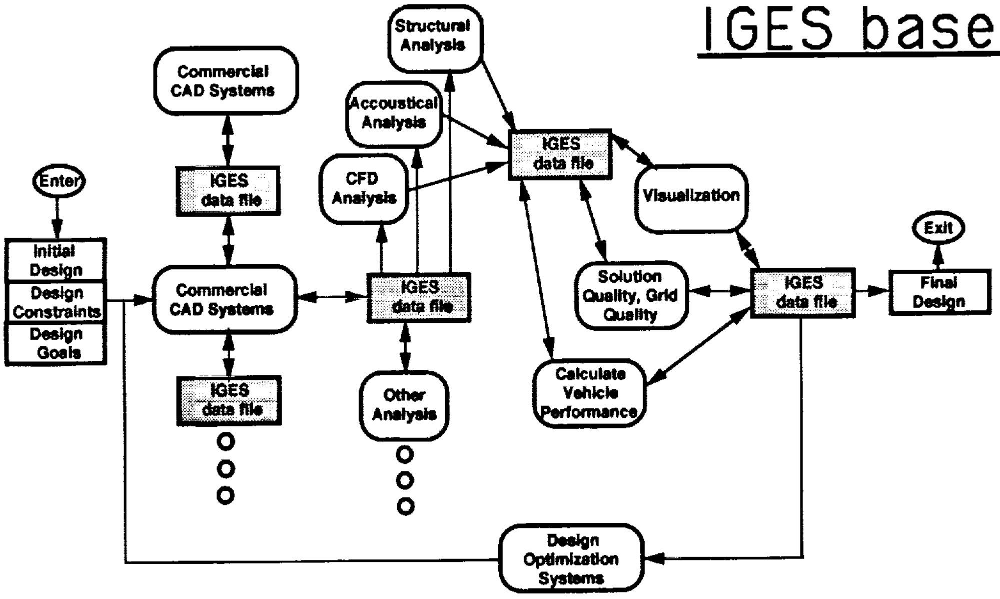
- **Insure reasonable space efficiency:**
  - **Disk space vs. ease of use**
    HELP>>> My calculations show Unstructured Grid Formats require 10 - 20 times the storage space of structured. If you disagree, describe your assumptions and calculations

  - **ASCII vs. binary**
    HELP>>> Why stick with ASCII? IEEE binary?

CFD Grid & Solution Data Standards: Design Goals (cont.)

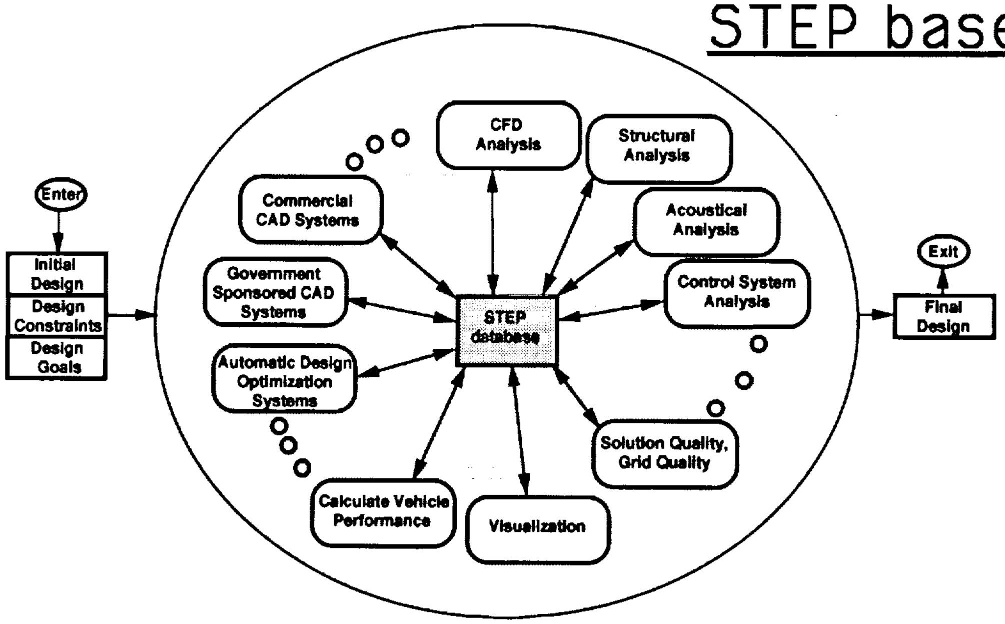
- **Select a format that is compatible or expandable for multi-disciplinary analysis:**
  - **Surface data only?**
    HELP>>> This is what CFD would exchange with a structural analysis package, why ship more?
  - **Linked to the geometry?**
    HELP>>> Required for accurate surface grid adaption
  - **Which other disciplines?**
    HELP>>> Structures, Controls, Thermal, ????

IGES based



DATA EXCHANGE FOR MULTI-DISCIPLINARY ENGINEERING ANALYSIS

STEP based



DATA EXCHANGE FOR MULTI-DISCIPLINARY ENGINEERING ANALYSIS

How To Help (or Get Help) on NASA Data Exchange Standards

- To get on the email foum for Grid Generation contact: siggrid-request@nas.nasa.gov (or my email below)
- To get a draft copy of the "NASA Geometry Data Exchange Specification for CFD" (NASA-IGES) contact me
- To assist with Grid & Solution Data Exchange Standards, fill out a data requirements sheet for your software (available at the back of the room) or provide documentation of your requirements, send to me
- Matthew Blake
  MS T045-2
  NASA Ames Research Center
  Moffett Field, CA 94041
  blake@nas.nasa.gov (415) 604-4978 FAX -3957
- NASA Langley, Pat Kerr, 804-864-5782, pkerr@eagle.larc.nasa.gov
- NASA Lewis, Scott Thorp, 216-433-8013, edthorp@opus.lerc.nasa.gov

SURFACE ACQUISITION THROUGH VIRTUAL MILLING

MARSHAL L. MERRIAM
NASA AMES RESEARCH CENTER

Surface Acquisition Through Virtual Milling

Marshal L. Merriam
CFD Branch
NASA Ames Research Center
Moffett Field, CA 94035

Kumaran Kalyanasundaram
Iowa State University
Ames, IA 50011

Abstract

Surface acquisition deals with the reconstruction of three-dimensional objects from a set of data points. The most straightforward techniques require human intervention, a time consuming proposition. It is desirable to develop a fully automated alternative. Such a method is proposed in this paper. It makes use of surface measurements obtained from a 3-D laser digitizer - an instrument which provides the (x, y, z) coordinates of surface data points from various viewpoints. These points are assembled into several partial surfaces, using a visibility constraint and a 2-D triangulation technique. Reconstruction of the final object requires merging these partial surfaces. This is accomplished through a procedure that emulates milling, a standard machining operation. From a geometrical standpoint the problem reduces to constructing the intersection of two or more non-convex polyhedra.

1. Introduction

The field of surface definition has gained considerable importance in the past couple of years. Advances in computers and numerical flow algorithms have made simulation of 3-D fluid flow computationally tractable. The single greatest impediment to the use of this technology on complex 3-D objects, such as complete aircraft, is defining the shape of the objects themselves. This observation has focused considerable attention on surface definition and surface modeling.

One commonly used technique for surface definition involves re-creating an object from a series of body cross-sections, coordinates of which are available from a computer-aided design (CAD) database or direct measurements [1,2,3]. This process requires human intervention and is susceptible to human error. A more automated approach, both for measuring the object and for constructing a surface conforming to the measurements, is needed.

Three-dimensional objects can be measured quickly and automatically using a laser digitizer [4]. This device, like a coordinate measuring machine, returns the coordinates of a number of surface points. Instead of a mechanical probe, the digitizer uses optics for its measurements. The lack of mechanical inertia and physical contact in the measurement process allows a five order of magnitude improvement in speed over a coordinate measuring machine. The digitizer collects points at the rate of 14500/second, to an accuracy of about $0.2-0.5$ millimeters depending on the surface albedo and orientation. The object is held on a solidly built machinist's table on which it can be translated or rotated by computer driven

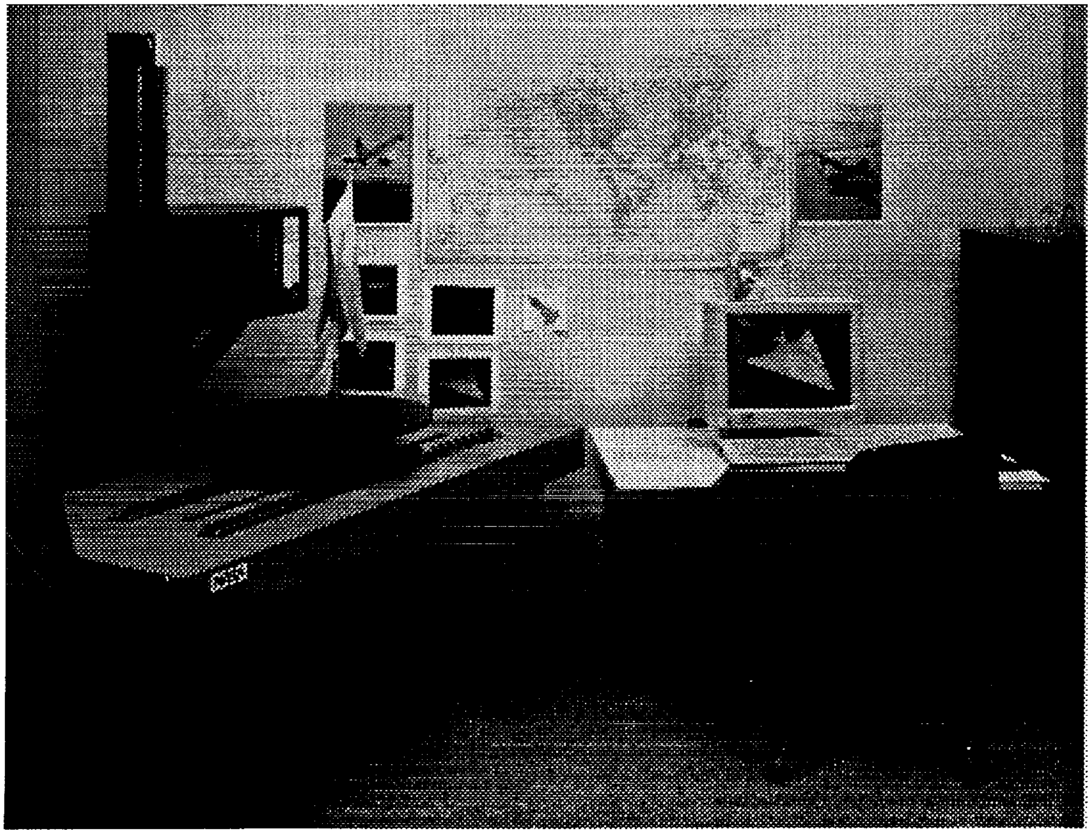


FIGURE 1. A Cyberware laser digitizer at NASA Ames Research Center.

servo motors, thus allowing observations from several different viewpoints to be expressed in a single coordinate system (see Figure 1). Merriam & Barth [5,6] have described the laser digitizer at length, interested readers are directed to their paper for further details.

Once the surface measurements are done, the surface incorporating the measured data must be reconstructed. Maksymiuk *et al.* [7] have proposed an algorithm based on pruning of unstructured grids. The method involves performing a Delaunay tessellation of the data points in three dimensions. This results in a solid body made up of tetrahedra, a valid reconstruction only for convex objects. The key insight is that no part of the object can obstruct the line of sight between the laser source and the object. This allows an improved shape to be reconstructed by deleting tetrahedra that intersect that line of sight. The algorithm is of $O(N \log N)$ complexity, where N is number of observed points. It has two main drawbacks: it generally removes more material than virtual milling, and the topological correctness of the reconstructed surface is sensitive to small errors in the experimental measurement. Both problems come from the discreteness of the pruning process: a tetrahedra is either removed or left untouched.

A very similar, but independently developed, procedure has been used by Faugeras *et al.* [8] for reconstructing 3-D scenes from stereo photographs. They also have shown how the reconstructed surface converges to the true surface when the sampling density increases.

Other algorithms for surface reconstruction exist. For example see Uselton[9] or Hoppe *et al.* [10]. We believe our algorithm to be fundamentally different and to have the following good properties.

i) It is reasonably efficient, having a formal complexity of $N \log N$ where N is the number of observed points.

ii) It always yields a topologically correct surface.

The remainder of the paper covers some of the algorithmic details of virtual milling including the relevant data structures and search techniques.

2. Surface Reconstruction From Digitizer Data

Our input data comes from a 3-D laser digitizer. This device provides prodigious amounts of data, but the data is given as a set of independent measurements. The desired output is a triangular faceted polyhedron which approximates the shape of the object being scanned.

Physical milling is the process of carving away material from an initial "blank" until the remaining material has the desired shape. Virtual milling (VM) simulates this process using computational geometry techniques. This immediately solves the most difficult problem; incorporating information from many different scans into a single part. The virtual cutting head resolves any small inconsistencies between scans. Whichever scan cuts the deepest prevails.

Two problems remain. First, the information from a single scan must be formed into a polyhedron which represents the volume to be milled out. Second, that polyhedron must be subtracted (in a solid modeling sense) from the workpiece.

2.1 Forming Surface Fragments From Individual Scans

The first job is to establish a triangular faceted surface fragment, an open two-manifold in 3-D, such that every measured point is fairly close to it. We give two separate strategies for doing this. One involves continuously adding points to gradually improve the surface approximation in the L_∞ norm. The other, which will be covered first, simply includes all the measured points from the outset, thereby avoiding the considerable expense of repeatedly computing the norm, but often resulting in a surface with many more vertices. In our experience, the difference is often a factor of 10.

There are a very large number of ways to triangulate N measured points. Each of these triangulations results in a surface fragment which includes all N points (by construction). Most of them can be eliminated by the use of visibility constraints.

It is known that the laser passes unimpeded from its source to the each point because observation requires illumination by the laser. This means that any triangulation which puts a triangle between the laser source and any observed point can be immediately discarded. One way to efficiently avoid such triangulations is to use projection methods.

Imagine for a moment that the laser originates from a point infinitely far from the workpiece ($z = -\infty$) so that all the rays are parallel to the z axis (the coordinate system is illustrated by Figure 2). Now project all the measured points onto the x, y plane (ignore the

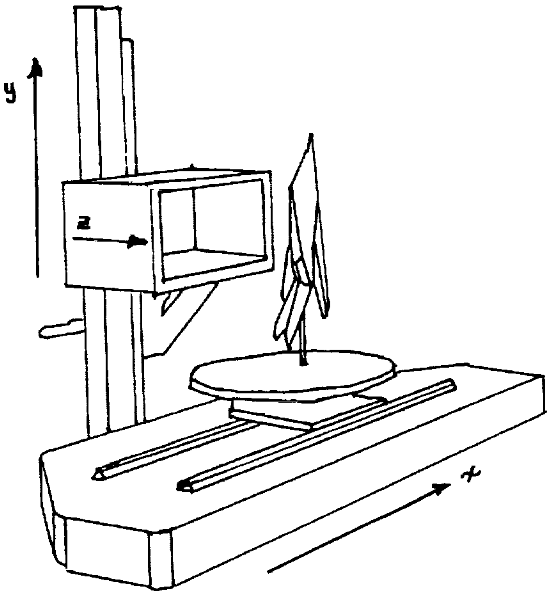


FIGURE 2. Coordinate systems for the laser digitizer.

z component) and perform some 2-D triangulation to establish connectivity between points. The corresponding 3-D triangulation (the one which has the same connectivity between points) does not violate the visibility constraint. That would imply that some edges of the projected triangulation cross. The converse is also true; every set of connections which satisfies visibility is a valid triangulation in the projected plane.

Now in practice, the focal length of the laser is not infinite, but only about three times its field of view[6]. The rays are perpendicular to the x axis, but are not parallel, forming an angle with the (x, z) plane that can be as much as 8.5 degrees. The appropriate projection in this case is cylindrical, rather than orthogonal, with the axis of the cylinder running parallel to the x axis and containing the laser source. In this coordinate system, the location in the (x, y) plane is given in polar coordinates (r, θ), the origin of which is at the laser source.

There are still an exponentially large number of ways to triangulate N points in the projected plane. The Delaunay triangulation in two dimensions is employed here. Delaunay triangulation is a classical problem in computational geometry and a well established technique for connecting scattered points [11 & 12].

A variation [13] involves using an incremental insertion algorithm for the Delaunay triangulation. After each insertion, the projected distance from the surface fragment to each measured point is computed and the one farthest away is determined and inserted. This process continues until the largest error falls below 0.25 mm, the nominal accuracy of the measurements.

Once the triangulation is done, the connectivity information is retained and the points are transformed back to their original (x, y, z) values. This gives a reasonable surface

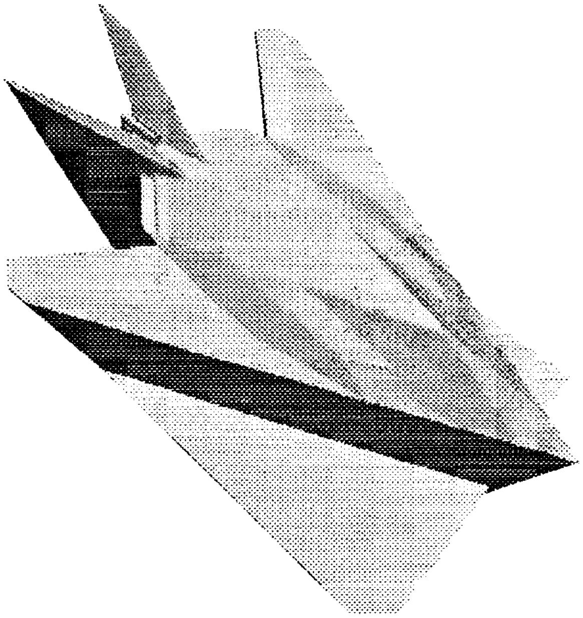


FIGURE 3. Unstructured triangulation of the top view of the F117A obtained using 2D Delaunay triangulation.

description (Figure 3), valid for shapes which can be reconstructed from a single scan, *e.g.*, sheet metal stampings.

For complex geometries, such as aircraft, multiple viewpoints are required. It is necessary to have, at least, a scan of the front and back views for reconstruction. In these situations a number of surface fragments have to be assembled. The end result is an approximation of the model as a non-convex polyhedron with triangular faces.

2.2 Combining Surface Fragments From Different Views

Combining the different views is achieved through a technique that emulates milling, a machining operation in which a workpiece is cut to the desired shape by careful removal of material. During each operation, the motion of the cutter is constrained to remove as much stock as possible without touching the finished part. Finally, only the finished part remains.

In the numerical analog the workpiece is any polyhedron (*e.g.*, a bounding box) with triangular faces, chosen to enclose the entire model. For each scan (view), it can be inferred that a polyhedral volume between the laser source and the object contains no material. This volume is excluded from the workpiece through a polyhedral intersection algorithm to be described later. In a solid modeling sense, the excluded volume is subtracted from the workpiece. Subtraction in this sense is commutative. Combining views consists of constructing the excluded polyhedra for each one and subtracting it from the partially finished part.

The problem here, is to subtract the volume of the polyhedron generated from a surface fragment, from the polyhedral workpiece ($P - Q$). In the following sections an algorithm for computing the intersection between two non-convex polyhedra with triangular faces is described in detail. The change needed to adapt this algorithm to perform the problem at hand, *i.e.*, to construct the intersection between a polyhedron and the compliment of another polyhedron, is also described.

3. Forming the Intersection of Two Non-Convex Polyhedra

Intersection problems have a wide variety of industrial applications [14], related to the fact that two objects cannot occupy the same space at the same time. Efficient, even optimal, algorithms have been developed for solving polygon intersection problems, but comparatively little is known about polyhedron intersections. Generalizations of the 2-D algorithms to 3-D are not straightforward.

In this work, only polyhedra with triangular faces are considered. This is done without loss of generality, since any higher degree polygon can be triangulated. This simplification allows reasonably efficient solution of polyhedron intersection problems in three-dimensions.

The intersection of an arbitrary number of non-convex polyhedra reduces to finding the intersection of two polyhedra. Given two non-convex polyhedra, P & Q, with triangular faces, form their intersection, $R \equiv P \cap Q$, such that the resulting polyhedron has only triangular faces.

The analogous problem in 2D is considered first. The intersection of two simple polygons A & B is a simple polygon C (Figure 4). Constructing C, from A and B, involves locating its vertices and its edges. Some of the vertices of C are vertices of polygon A, those which lie inside polygon B. Similarly, the vertices of polygon B which lie inside polygon A are vertices of C. The intersections of the edges of A and edges of B form the remaining vertices.

The edges of polygon C are all complete edges or edge fragments from polygons A or B. Edges of polygon A which lie entirely within polygon B, (*e.g.*, edge 1 in Figure 4), are edges of C. On the other hand, an edge of polygon A which lies entirely outside polygon B (*e.g.*, edge 2), is not. When an edge of A intersects one or more edges of B (*e.g.*, edge 3) only the edge fragments which lie inside polygon B are edges of C. Similar rules apply to edges of B.

Finding the intersection of two polyhedra can be accomplished by applying a similar procedure. The polyhedron R, formed from P ∩ Q, has nodes which are either nodes of P, nodes of Q, or intersections between the faces of P and the faces of Q. This problem, finding the intersection of two triangles in three-space, involves finding the endpoints of the line segment of intersection.

These line segments themselves constitute some of the edges of R. The other edges are formed from existing edges (the edges of P & Q) by treating them the same way as in polygon intersections. At this point, the intersection is a polyhedron with planar polygonal faces, some of which are not triangular. The higher degree polygons are triangulated so that the final polyhedron (R) has only triangular faces.

Summarizing then, computing the intersection of two polyhedra involves three main algorithms: polyhedron inclusion, line segment of intersection of two triangles in three-

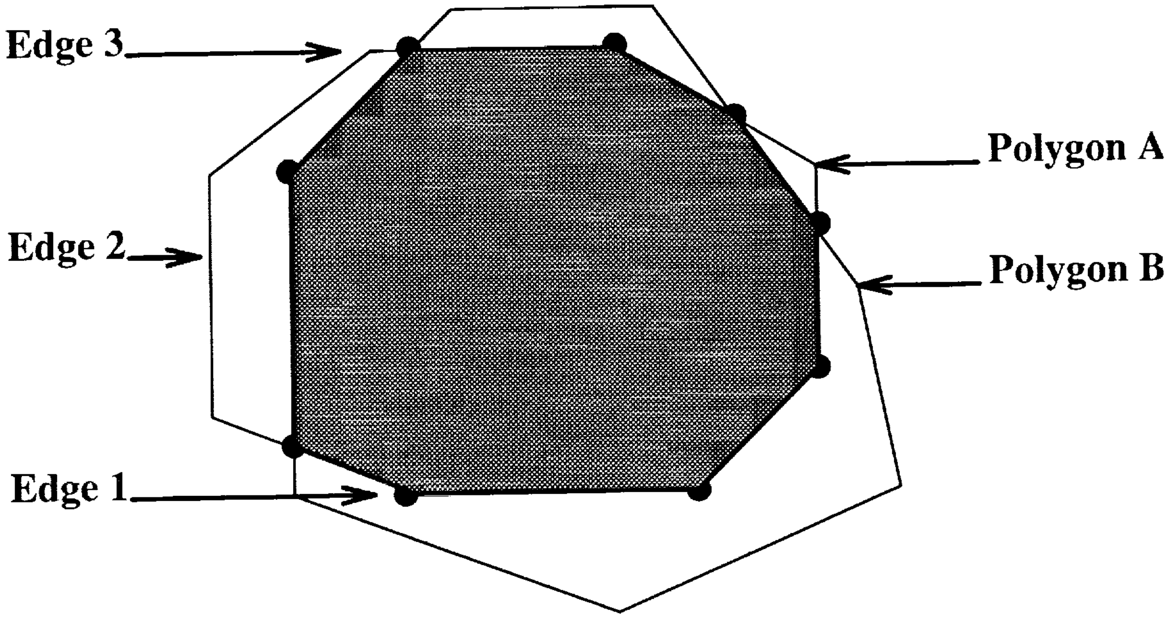


FIGURE 4. Intersection of two simple polygons A&B. The intersection is a simple polygon C, shown shaded. The thick dots denote the vertices of C.

space, and triangulating the interior of a simple polygon. These are described in the following sections.

3.1 Polyhedron Inclusion

Given a polyhedron and a point, is the point inside the polyhedron? To answer this a ray is drawn, emanating from the given point, typically along one of the three coordinate directions. The number of intersections between the ray and the polyhedron are counted. If the number is odd the point lies inside the polyhedron. Otherwise it lies outside the polyhedron. This algorithm is well known in 2D [14] and the 3D case is completely analogous.

Since the polyhedron is entirely composed of triangles, this only requires finding the 3D intersection of a ray with a triangle. By projecting both the ray and the triangle onto a plane normal to the ray, this problem is largely reduced to the 2D problem of point inclusion in a triangle.

An exhaustive search of all triangles will give the correct number of intersections. This is expensive. Sorting the triangles into a tree like structure drastically decreases the number of triangles searched each time. The data structure employed here is the alternating direction binary tree developed by Bentley [15]. Exposition of the search and sort algorithms is done, briefly, in a later section.

3.2 Intersection of Two Triangles in Three Dimensions

The polyhedron inclusion test determines which of the original vertices of P and Q will appear in R. The next step is to compile a list of intersecting pairs of triangles. These

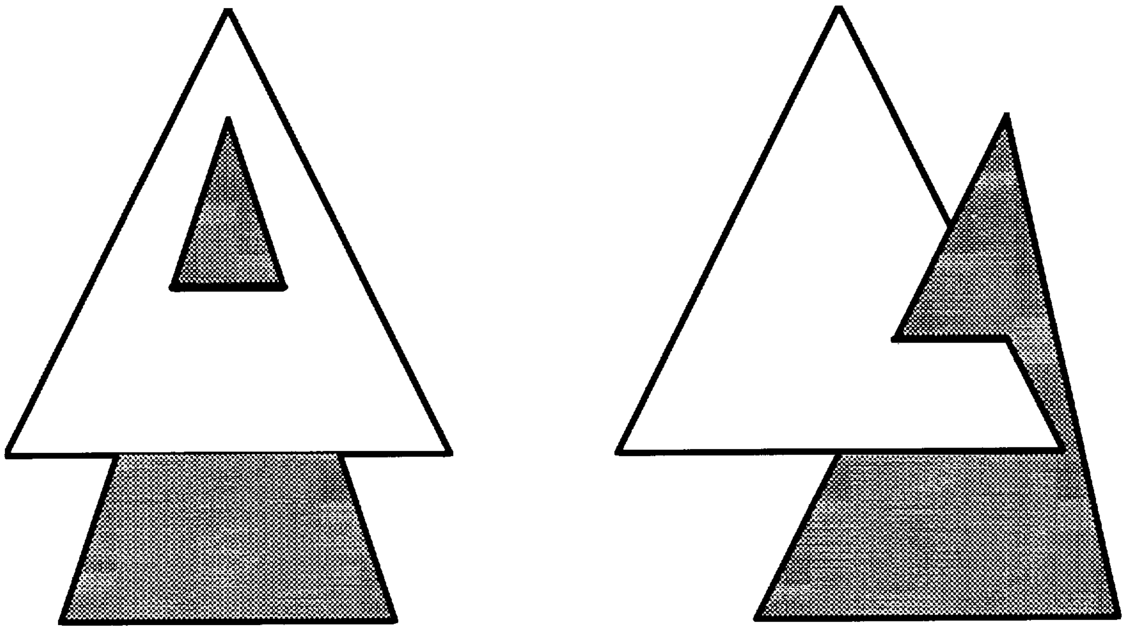


FIGURE 5. Two types of intersections of two triangles in three-space.

provide some of the edges of the final polyhedron along with all the remaining vertices.

Two triangles in 3D, A and B, intersect (if at all) along a line segment, each end of which lies on a separate triangle edge. To find these endpoints, each edge of triangle A is tested for intersection with triangle B, a problem essentially covered in the previous section. Similarly, edges of B are tested for intersection with triangle A.

Figure 5 shows two possible ways two triangles can intersect. Degenerate cases such as two intersecting, coplanar triangles, were not encountered. There are $O(N^2)$ pairs of triangles to test, most of which do not come close to intersection. Once again, the triangles have been sorted into a binary tree to avoid the expensive exhaustive search.

3.3 Constrained Triangulations

Edges of the two intersecting polyhedra (P ∩ Q) can be classified into three categories: a) edges of one polyhedron which lie entirely outside the other. Such edges are not part of the final polyhedron. b) The opposite situation, where edges of one polyhedron lie completely inside the other. Such edges are part of the final polyhedron. c) Edges of one polyhedron which intersect one or more triangular faces of the other. For such edges only those portions which lie inside the other polyhedron remain as part of the final polyhedron.

Figure 6a illustrates a situation where all vertices of a particular triangle lie inside the other polyhedron, and five new nodes have been added. The new node on the face of the triangle, N_5, is the point where an edge of Q intersects. The nodes on the edges (N_1, N_2, N_3, N_4) are the points of intersection between the edges of this triangle and the triangular faces of Q. The polygons shown in Figure 6b, are faces of P ∩ Q. The interiors of these simple polygons have to be triangulated in order to assure that the final polyhedron has only triangular faces.

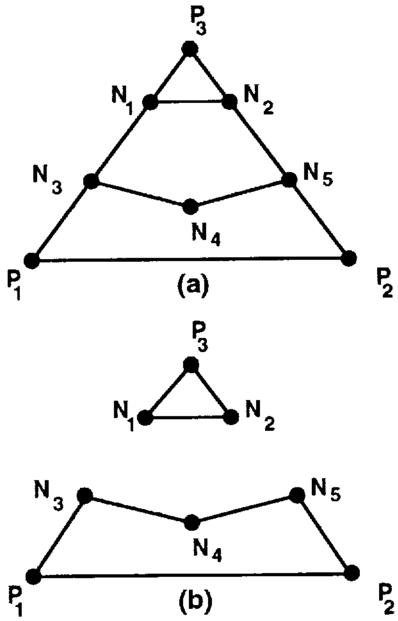


FIGURE 6. (a) This triangle has all three vertices (P_1, P_2, P_3) lying inside Q. N_1, N_2, N_3, N_4 are intersections between edges of P and faces of Q. N_5 is an intersection between this triangle and an edge of Q. All five points are vertices of P ∩ Q. (b) These regions are part of P ∩ Q.

Decomposing a polygon of degree N into $(N - 2)$ triangles is a classical problem in computational geometry. The best algorithms operate in linear expected time (Chazelle [16]). Since we rarely encountered polygons of very high degree, programming simplicity determined our choice of an $O(N^2)$ complexity algorithm by Bern & Eppstein [17].

4. Data Structures

Finding the intersection of two non-convex polyhedra involves answering two types of geometric questions:

a) Which triangles (if any) in a given set, contain a given point?

b) Which intersect a given triangle? Since both queries appear many times, it is essential to use an efficient algorithm in answering them.

One approach that will work is to test each triangle for intersection with the relevant point or triangle. This technique (exhaustive search) has a run time of $O(NM)$, where N is the number of triangles, and M, the number of queries. Such a search can be prohibitively slow.

A quicker approach searches some of the triangles each time, with full confidence that the unsearched triangles would return negative responses. This involves presorting the triangles. The domain containing all the triangles is partitioned spatially [18]. Searching is then restricted to the partition where the given point (or triangle as the case may be) lies, and, possibly, a few of the neighboring partitions. Algorithms of this type typically have run times of $O(M \log N)$.

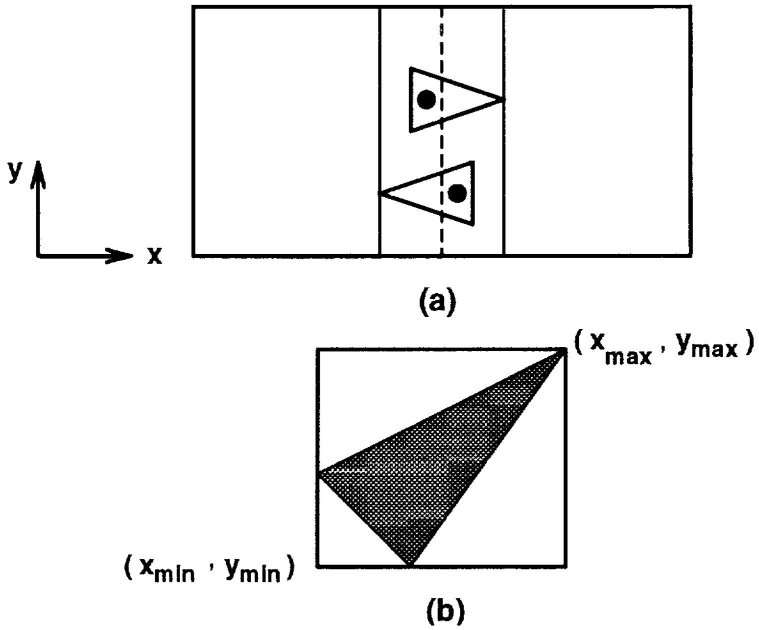


FIGURE 7. (a) A domain containing a set of triangles is partitioned into two regions with a slight overlap. (b) Triangle queries are handled by enclosing the triangle in a bounding box and searching all partitions which contain a portion of the box.

To achieve a suitable partitioning, triangles are treated as points. Each triangle is replaced by a unique point. The centroid has been chosen here, because it always lies inside the triangle. Then, the domain is divided into pieces with roughly equal numbers of triangles. The number of sub-domains created, during each division, depends on the type of data structure. In binary trees [15] the domain is divided into two halves at each level.

A one level example is illustrated in Figure 7a. The rectangular region containing the triangles has been divided into two. The dividing line is chosen to put an (approximately) equal number of triangles into each half. In this case the average of the x coordinates of the centroids was used to locate the vertical divisor. It is represented as a dotted line in Figure 7a. Bounding boxes can be constructed by determining the smallest and largest coordinate values of the vertices of the triangles contained in each half. Notice a small portion of the domain is common to both bounding boxes. Search efficiency depends on this overlap region being small compared to the size of the overall domain.

Now suppose we seek all triangles which contain a given point. By comparing x coordinate values, It can easily be determined which (if either) of the two bounding boxes contain the point. Clearly, if a point lies outside of a bounding box, it lies outside of all the triangles contained therein. If the point does not lie in the overlap region, at most half of the triangles will be searched.

For handling triangle queries, the triangle is enclosed in a bounding box (Figure 7b), and all the partitions which contain a portion of the box are searched. This effectively enlarges the overlap region, but the algorithm is otherwise identical.

A two-dimensional tree search has been implemented here for finding intersections

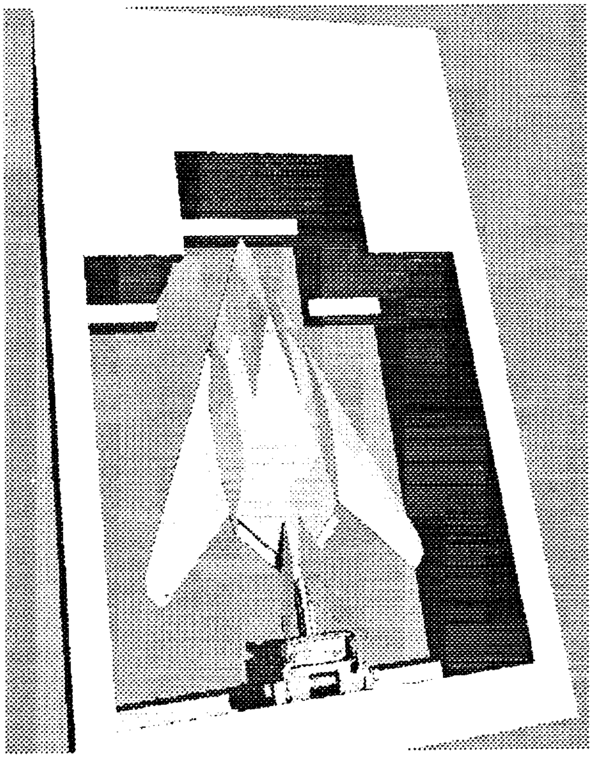


FIGURE 8. The F117a is still attached to the original blank and must be carefully separated. At this point, eight scans have been milled away.

between triangles in 3-D. The triangles are projected on to a plane, say the $x - y$ plane, and then partitioned according to the x and y coordinate values of their centroids. This approach was found to be more efficient than a three-dimensional tree search.

The polyhedron generated from a surface fragment is much smaller in size than the workpiece that encloses the entire model. It is useful to determine the bounding box of each polyhedron. It is only necessary to test for polyhedron intersection where these bounding boxes intersect, a significant optimization.

5. Finishing Operations

The physical model being scanned has to be supported securely in the digitizer. This usually implies a sting, but sometimes the model rests directly on the turntable. In any case, the desired geometry is usually attached to the remnants of the original blank. This situation is shown in Figure 8. When this happens in actual machining, the finished part is separated very carefully by hacksawing through the last connection. A similar approach is followed here. The cutting operation is simulated, again, through the intersection algorithm.

A number of other finishing operations are performed. The model is separated from the remains of the blank (clearing chips), the surface is given a consistent orientation, and some very small edges and triangles are removed (polishing). Finally, pinholes near the trailing edges are identified for later treatment.

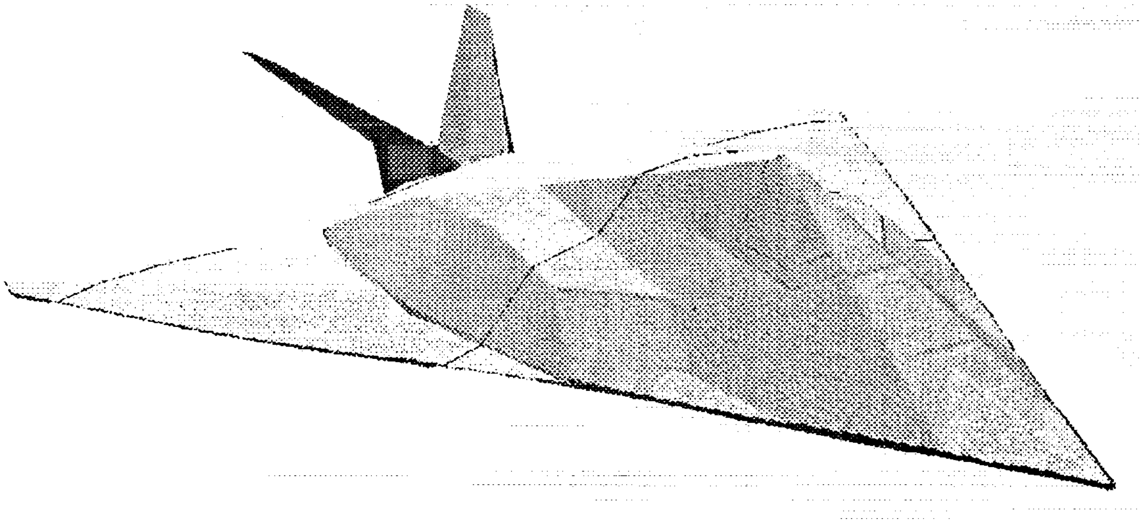


FIGURE 9. Front view of the reconstructed F117A. This incorporates information from eight scans.

6. Results and Discussion

The algorithm for generating an accurate geometric definition of three dimensional objects through virtual milling has been implemented on a Silicon Graphics workstation. The procedure has been tested on a F117A model scanned from several different viewpoints. Figure 9 shows the reconstructed F117A aircraft. The reconstruction involved eight scans and resulted in about 200,000 triangular faces. The sting, which supported the model during the digitizing process, was not completely removed. A portion of it is visible near the tail.

The runtime on an Iris 320/VGX was about six hours, including the scanning time. In an effort to reduce this time, the parallel intersection algorithm has been ported to the iPSC/860. Preliminary indications show a run time of under 10 minutes for the polyhedron intersection operations. The two intersecting polyhedra, P & Q, are equally split among the different processors by employing recursive coordinate bisection. Each processor is then responsible for constructing a portion of the intersection region.

7. Concluding Remarks

An algorithm for reconstructing 3-D objects from scattered data has been presented. The algorithm utilizes the Delaunay triangulation in two dimensions to generate partial surfaces from single views. Combining the different views is accomplished through virtual milling, a numerical analog of the physical machining operation. The technique is a general and automated method for reconstructing surfaces and assembling data from multiple views. Results for an F117A model clearly demonstrate these aspects. Ample research opportunities remain. For example, The resulting part is as accurate as the digitizer itself but is not very smooth. It would be nice to have an algorithm to produce the smoothest possible part without moving any point more than the nominal measurement accuracy. On another front, Kalyanasundaram [18] has demonstrated a parallel implementation of polyhedron intersection on the Intel iPSC/860.

8. Acknowledgement

The authors would like to thank Ms. Catherine Maksymiuk for her implementation of DeFloriani's decimation algorithm. It is implemented efficiently $O(NlogN)$ and greatly simplifies the virtual milling process without degrading the reconstructed surface.

9. References

1 Edwards, T.A., "Definition and Verification of a Complex Aircraft for Aerodynamic Calculations," AIAA-86-0431, 24th Aerospace Sciences Meeting, Reno, Nevada, January 6-9, 1986.

2 Edwards, T.A., "Geometry Definition and Grid Generation for a Complete Fighter Aircraft," 58th Meeting of the Fluid Dynamics Panel Symposium on Applications of CFD in Aeronautics, Aix-en-Provence, France, April 7-10, 1986.

3 Atwood, C.A. and Van Dalsem, W.R.,"Flowfield Simulation about the SOFIA Airborne Observatory," AIAA-92-0656, AIAA 30th Aerospace Sciences Meeting, Reno, NV, January 1992.

4 Cyberware Laboratory Inc., 8 Harris Court, Monterey, Ca 93940

5 Merriam, M.L. and Barth, T.J., "3D CFD in a Day The Laser Digitizer Project," AIAA-91-1654, 26th Fluid Dynamics, Plasma Dynamics & Lasers Conference, Honolulu, June 24-27, 1991.

6 Merriam, M.L. "Experience With The Cyberware 3D Digitizer," Proceedings of CAD and Engineering Workstations '92, NCGA, Anaheim, California, March 9-12, 1992.

7 Maksymiuk, C. and Merriam, M.L., "Surface Reconstruction from Scattered Data Through Pruning of Unstructured Grids," AIAA-91-1584, 10th Computational Fluid Dynamics Conference, Honolulu, June 24-27, 1991.

8 Faugeras, O.D., Le Bras-Mehlman, and Boissonnat, J.D., "Representing Stereo Data with the Delaunay Triangulation," Artificial Intelligence, v44, pp 41-87, July, 1990.

9 Uselton, S.P., "Surface Reconstruction From Limited Information," Ph.D. dissertation, The University of Texas, Dallas, Texas, December, 1981.

10 Hoppe, H., DeRose, T., Duchamp, T., McDonald, J., and Steutzle, W., "Surface Reconstruction From Unorganized Points," Computer Graphics 26, July 2nd, 1992.

11 Merriam, M.L., "An Efficient Advancing Front Algorithm For Delaunay Triangulation," AIAA-91-0792, 29 Aerospace Sciences Meeting, Reno, Jan 7-11, 1991.

12 Tanemura, M., Ogawa, T.,and Ogita, N., "A New Algorithm For Three Dimensional Voronoi Tessellation," Journal of Computational Physics 51, pp 191-207, 1983.

13 De Floriani, L., Falcidieno, B., and Pienovi, C., "Delaunay-based Representation of Surfaces Defined Over Arbitrarily Shaped Domains," Computer Vision, Graphics, and Image Processing, vol. 32, pp 127-140, 1985.

14 Preparata, F.P. and Shamos, M.I., "Computational Geometry An Introduction," Springer-Verlag, 1985.

15 Bentley, J.L., "Multidimensional Binary Search Trees Used For Associative Searching," Communications of the ACM, 18(9), pp 509-517, September, 1985.

16 Chazelle, B., "Triangulating a Simple Polygon in Linear Time," In Proc. 31st IEEE Symposium Foundations of Computer Science, pp 220-230, 1990.

17 Bern, M. and Eppstein, D., "Mesh Generation and Optimal Triangulation," Xerox PARC Technical Report, CSL-92-1.

18 Knuth, D.L, "The Art of Computer Programming, Vol.3," Addison Wesley, 1973.

19 Kalyanasundaram, K., "Parallel Surface Definition Through Virtual Milling," Ph.D. dissertation in progress, Iowa State University, Ames, Iowa.

Dr. William Van Dalsem
Computational Technology Branch
NASA Ames Research Center
Moffett Field, CA 95050
(415) 604-4469
e-mail: vandal@nas.nasa.gov

Mr. Guru R. Vemaganti
Lockheed Eng. & Sciences Co.
144 Research Drive
Hampton, VA 23666
(804) 766-9473

Mr. David L. Whitaker
Analytical Services and Materials, Inc.
MS 128
NASA Langley Research Center
Hampton, VA 23681
(804) 864-2150
e-mail: d.l.whitaker@larc.nasa.gov

Dr. Laurence B. Wigton
BOEING
MS 128
NASA Langley Research Center
Hampton, VA 23681
(804) 864-7885
e-mail: lbw9902@tab00.larc.nasa.gov

Dr. Thomas A. Zang, Jr.
MS 150
NASA Langley Research Center
Hampton, VA 23681
(804) 864-4082
e-mail: t.a.zang@larc.nasa.gov

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE
September 1993

3. REPORT TYPE AND DATES COVERED
Conference Publication

4. TITLE AND SUBTITLE
Unstructured Grid Generation Techniques and Software

5. FUNDING NUMBERS
WU 505-90-53-02

6. AUTHOR(S)
Mary-Anne K. Posenau, Editor

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
NASA Langley Research Center
Hampton, VA 23681-0001

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
National Aeronautics and Space Administration
Washington, DC 20546-0001

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
NASA CP-10119

11. SUPPLEMENTARY NOTES
Proceedings of the NASA Workshop on Unstructured Grid Generation Techniques and Software held at the NASA Langley Research Center, April 27-28, 1993

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Unclassified–Unlimited

Subject Category 61

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*
The Workshop on Unstructured Grid Generation Techniques and Software was conducted for NASA to assess its unstructured grid activities, improve the coordination among NASA centers, and promote technology transfer to industry. The proceedings represent contributions from Ames, Langley, and Lewis Research Centers, and the Johnson and Marshall Space Flight Centers. This report is a compilation of the presentations made at the workshop.

14. SUBJECT TERMS
Unstructured Grid Generation; Unstructured Grids; Computational Fluid Dynamics

15. NUMBER OF PAGES
360

16. PRICE CODE
A16

17. SECURITY CLASSIFICATION OF REPORT
Unclassified

18. SECURITY CLASSIFICATION OF THIS PAGE
Unclassified

19. SECURITY CLASSIFICATION OF ABSTRACT
Unclassified

20. LIMITATION OF ABSTRACT