NASA Conference Publication 10089

Workshop on Grid Generation and Related Areas

Proceedings of a workshop hosted by
NASA Lewis Research Center
Cleveland, Ohio
November 14–15, 1991

NASA

Workshop on Grid Generation and Related Areas

NASA

National Aeronautics and
Space Administration

Office of Management

**Scientific and Technical
Information Program**

1992

PREFACE

This publication is a collection of presentations given at the Workshop on Grid Generation and Related Areas held at NASA Lewis Research Center, Cleveland, Ohio, November 14-15, 1991. The purpose of this workshop was to assemble engineers and scientists from Lewis Research Center (including ICOMP (Institute for Computational Mechanics in Propulsion), SSC (Support Service Contractors), and OAI (Ohio Aerospace Institute)) working on grid generation for Computational Fluid Dynamics (CFD), surface modeling, and related areas. Specifically, the objectives were (1) to provide an informal forum on grid generation and related topics; (2) to assess user experience, skills, current activities, and available tools; (3) to identify needs; and (4) to help promote synergy among engineers and scientists working in this subject area.

The workshop consisted of four sessions representative of grid generation and surface modeling research and application within NASA Lewis Research Center. Each session contained presentations and an open discussion period.

Session I consisted of presentations on data management and grid generation for inlets, ducts, and nozzles. Data management, file manipulation, and binary data exchange between different computers are all important factors as CFD engineers work in teams. A general purpose code GRIDGEN is the primary workhorse used by groups to avoid the need for several smaller, specialized codes. Even though it needs further improvement, GRIDGEN appears to be well received by engineers who generate grids for various geometries because of its useful and versatile interactive features, including interactive zoning. However, since it is a complex code, it takes time for a user to acquire full knowledge of all available features.

Session II consisted of presentations on grid generation for turbomachinery and turboprops. Use of customized codes is the current trend in turbomachinery research at Lewis. Each grid code is closely tied to a turbomachinery flow solver. Lewis efforts have been focused on the component research and analysis of given designs, for which hard-wired, customized codes have met the needs of turbomachinery flow solvers. This resulted in the development of several small, specialized grid generation codes that provide the grid structure and quality required by the flow solvers. As Lewis extends its computational activities from single stage to complete engine configurations, and from single discipline to multidiscipline, additional software capability will be needed. Multicomponent and multidisciplinary research and applications will likely influence future trends.

Session III consisted of presentations on techniques for flow simulation in complex geometries, and grid adaptation techniques for complex flows: automatic domain decomposition, structured grid adaptation, unstructured grid generation and adaptation, and hybrid elliptic/hyperbolic grid generation schemes. Since most project-oriented CFD within Lewis currently uses a patched structured grid, automatic zoning and structured grid adaptation are important technologies that require continuing development. The unstructured grid generation and adaptation schemes will require further advances before widespread application.

iii

Session IV consisted of presentations on the activities of the NASA Surface-Modeling and Grid-Generation Steering Committee, Lewis activities for the Numerical Propulsion Systems Simulations (NPSS), and the NASA Geometry Data Exchange Specifications. The geometry data exchange standard was developed by a NASA subcommittee to provide rapid and accurate geometry data transfer and to promote cooperation among interested parties in the United States. It is clear that geometry modeling and grid generation techniques need to be extended across disciplines (aerodynamics, structures, heat transfer, etc.).

Lewis grid generation capability has been substantially enhanced since the in-house survey conducted in October 1989. Coordination and interaction among interested partners within Lewis, as well as within NASA, are important contributing factors. Surface modeling through Computer Aided Design (CAD) is quite advanced, but not well coordinated with grid generation. The interface between CAD and grid generation will need to be improved through the use of the NASA geometry data exchange standard if CFD is to become a useful design tool as well as an analysis tool. Lewis efforts have been toward component-oriented research and analysis. For multicomponent and multidisciplinary efforts for propulsion systems studies, Lewis needs better surface-modeling and grid-generation tools, human skills, and planned teamwork, since these efforts require expertise from several disciplines.

The workshop concluded with software demonstrations. Tim Beach demonstrated an interactive grid generation code for turbomachinery, called IGB, which uses an interactive definition of Bezier curves as a means to control grid structures. Jim Bruns demonstrated the use of a general purpose code called GRIDGEN. Developed by General Dynamics and the USAF, it is capable of interactive domain decomposition, surface grid generation, field grid generation, and interactive viewing. Yung Choo demonstrated a 2D interactive solution adaptive grid generation code, called TURBO-AD. David Miller demonstrated an interactive 2D multiblock grid generator for turbomachinery, called TIGGERC. Mark Stewart demonstrated a multiblock grid generation code called TOPOS3, which automatically decomposes the flow domain of jet engine configurations. Kin Wong demonstrated a grid and solution file manipulation program called QX_MAN.

Yung K. Choo
April, 1992

WORKSHOP ON GRID GENERATION
AND RELATED AREAS

CONTENTS

Session I-Grid Generation For Inlets/Nozzles/Ducts
Chair: Danny Hwang

Session II-Grid Generation For Rotating Machinery

Chairs: Dennis L. Huff and David P. Miller

Session III-New Techniques In Grid Generation
Chair: Mark E.M. Stewart

Session IV-NASA and Lewis Activities
Chair: Austin L. Evans

SESSION I

GRID GENERATION FOR INLETS/NOZZLES/DUCTS

Chair: Danny Hwang

GRID MANAGEMENT
by
Danny Hwang
NASA Lewis Research Center

ABSTRACT

Special computational environment exists in the Special Project Office (SPO) to work for the same project by many CFD engineers. Grid generation is not a task of a single person and also the grid created is not used by a single user. The presentation will be a brief overview of the grid management used by the engineers of the SPO. The topics will include the grid file naming system, grid-generation procedure, grid storage, and our grid format standard (plot3d BDX binary).

3

- *INTRODUCTION*

- *GRID FILE NAMING SYSTEM*

- *GRID GENERATION PROCEDURE*

- *GRID STANDARD -- PLOT3D BDX BINARY*

- *GRID STORAGE*

- *INTRODUCTION*

- *GRID FILE NAMING SYSTEM*

- *GRID GENERATION PROCEDURE*

- *GRID STANDARD -- PLOT3D BDX BINARY*

- *GRID STORAGE*

```
          ┌─────────────────┐              ┌─────────────────┐
          │     USER 3      │              │    USER 2,3     │
          │ MODIFIED GRID 1 │─────────────▶│   USE GRID 2    │
          │  CALLED GRID 2  │              │ FOR CFD ANALYSIS│
          └─────────────────┘              └─────────────────┘
                   ▲
                   │
┌─────────────┐   ┌─────────────────┐
│   USER 1    │   │   USER 1,2,3,4  │
│CREATE GRID 1│──▶│   USE GRID 1    │
│             │   │ FOR CFD ANALYSIS│
└─────────────┘   └─────────────────┘
                   │
                   ▼
          ┌─────────────────┐              ┌─────────────────┐
          │     USER 4      │              │    USER 1,4     │
          │ADD ANOTHER BLOCK│              │   USE GRID 3    │
          │   TO GRID 1     │─────────────▶│ FOR CFD ANALYSIS│
          │ NEW GRID CALLED │              │                 │
          │     GRID 3      │              └─────────────────┘
          └─────────────────┘
```

● *INTRODUCTION*

┌──────────────────────────────────┐
│ ● *GRID FILE NAMING SYSTEM* │
└──────────────────────────────────┘
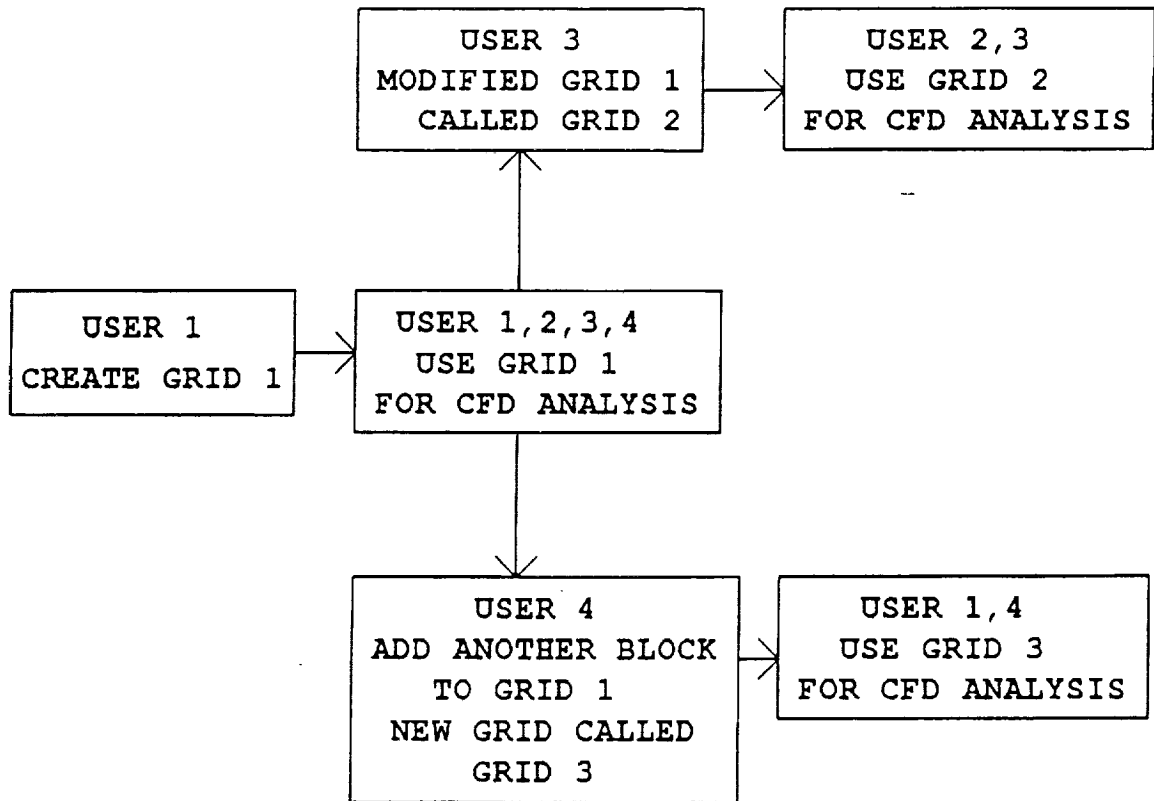
● *GRID GENERATION PROCEDURE*

● *GRID STANDARD -- PLOT3D BDX BINARY*

● *GRID STORAGE*

GRID FILE NAMING SYSTEM

GEOMETRY FILE NAME:

g(geometry name)(grid number).(plot3d grid extension)

FOR EXAMPLES: gsd07.x

gsd03.xb --(iblank file)

SOLUTION FILE NAME:

(code name)(geometry name)(grid number)(case number)

(iteration ID).(plot3d solution extension)
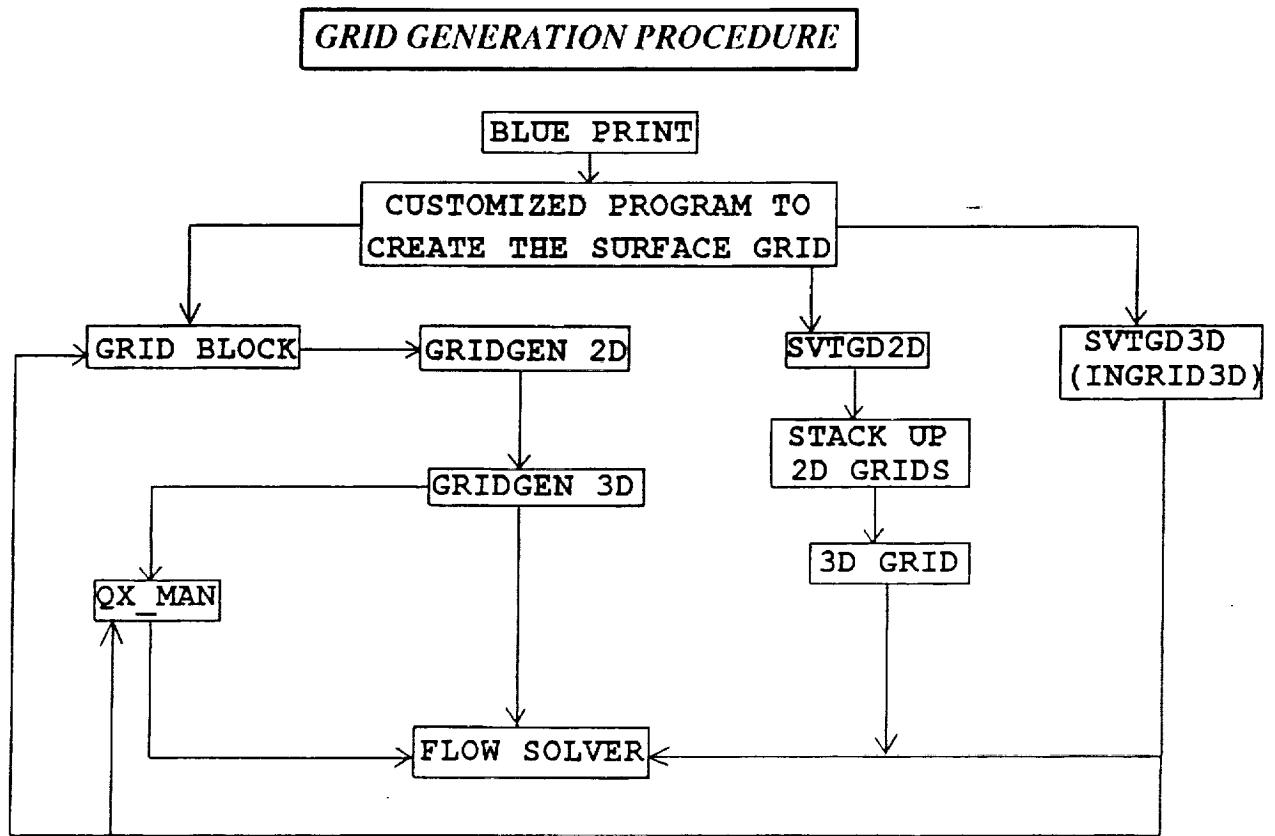
FOR EXAMPLES: psd0701a.q

    p : parc3d solution
    sd: geometry name
    07: grid number
    01: case name
     a: iteration ID (e.g. from iteration 1 to 1000)
     q: plot3d q file

PLOT3D Version 3.6 [IRIS 4D] 1 February, 1989

An interactive graphics program for 2D and 3D CFD datasets

If you have problems with this version contact Tony Facca
For general PLOT3D questions, enter HELP

date: 24/Oct/91   time: 09:33:14 AM

PLOT3D [IRIS 4D]: read/x=gsd07.x/q=psd0701a.q▯

PARC3D Log

Case Name: case 1			Grid Name: gsd07.x				Brief Description:		
Grid Size: 89 X 33X 49		Boundary Conditions:						Computer/ Directory:		
Job Name	Start/End Time	Iter. Range	DIS2	DIS4	DTCAP	PCQMAX	Start Resid.	End Resid.	CPU Time	Notes

Directory Name:		Grid Name:	$gsd07.X$		Grid Size:	$89 \times 33 \times 49$	
Case Name:	*Case 1*	Mode:		Notes:			

Input Parameters	XMach:	Pr:	Chord:	I1BC:	IMBC:	J1BC:	JMBC:

Dimension Parameters	WW:	X:		D:	RP:	RO:	

Run #	Date	Iteration		Residual		CPU Time	CNUM	Batch ID	Comments

- *INTRODUCTION*

- *GRID FILE NAMING SYSTEM*

- *GRID GENERATION PROCEDURE*

- *GRID STANDARD -- PLOT3D BDX BINARY*

- *GRID STORAGE*

```
                    ┌─────────────────────────────┐
                    │  GRID GENERATION PROCEDURE   │
                    └─────────────────────────────┘

                         ┌─────────────┐
                         │ BLUE PRINT  │
                         └─────────────┘
                                │
                    ┌───────────────────────────┐
                    │ CUSTOMIZED PROGRAM TO      │
                    │ CREATE THE SURFACE GRID    │
                    └───────────────────────────┘

  ┌─────────────┐    ┌─────────────┐    ┌───────────┐    ┌───────────────┐
  │ GRID BLOCK  │───▶│ GRIDGEN 2D  │    │ SVTGD2D   │    │ SVTGD3D       │
  └─────────────┘    └─────────────┘    └───────────┘    │ (INGRID3D)    │
                                                          └───────────────┘
                          ┌─────────────┐    ┌─────────────┐
                          │ GRIDGEN 3D  │    │ STACK UP    │
                          └─────────────┘    │ 2D GRIDS    │
                                             └─────────────┘
        ┌─────────────┐
        │ QX_MAN      │                      ┌─────────────┐
        └─────────────┘                      │ 3D GRID     │
                                             └─────────────┘

                          ┌─────────────┐
                          │ FLOW SOLVER │
                          └─────────────┘
```

- *INTRODUCTION*

- *GRID FILE NAMING SYSTEM*

- *GRID GENERATION PROCEDURE*

┌──┐
│ • *GRID STANDARD -- PLOT3D BDX BINARY*   │
└──┘

- *GRID STORAGE*

```
            ┌─────────────────────────────┐
            │      GRID STANDARD          │
            │                             │
            │   -- PLOT3D  BDX  BINARY -- │
            └─────────────────────────────┘
```

BDX -- BINARY DATA EXCHANGE

● WRITTEN IN FORTRAN 77

   --CAN BE EASILY MODIFIED FOR A SYSTEM

● PROVIDES A SYSTEM INDEPENDENT BINARY FORMAT

   --BINARY DATA TRANSFER THROUGH TCP/IP

● THE STORAGE SIZE OF A BDX FILE IS VERY SMALL

CRAY

```
SUBROUTINE QDINIT

PARAMETER (MUN=10,MRL=1024,MWD=MRL/8,MAX=MRL/4)
INTEGER II,IU,IP,IR,IB
LOGICAL LINIT
```

VAX or IRIS

```
SUBROUTINE QDINIT

PARAMETER (MUN=10,MRL=1024,MWD=MRL/4,MAX=MRL/4)
INTEGER II,IU,IP,IR,IB
LOGICAL LINIT
```

```
#****** dtox.com *****
# change dlum3dvc d file to plot3d x file
#  d file -- z is the flow direction
#  x file -- x is the flow direction
ja
#****** input *****
set input=g_wh_4.dp
#******** output *******
# output name is in dtox.f -- change before compile
#**********************
cft77 dtox.f
segldr -o tempx dtox.o,/u/ne/hwang/plot3d/sdblib.a
ln $input      fort.10
tempx
rm tempx
ls -l $input
ls -l g_wh_4.xp
rm sonigrid.o
rm tempx
rm temp.f
rm temp.o
rm fort.10
ja -s
```

```fortran
      PARAMETER (II=89,JJ=41,KK=41)
      CHARACTER*33 TITLE
      dimension x(ii,jj,kk),y(ii,jj,kk),z(ii,jj,kk)
C********************** READ *******************************
c
      READ(10) NZON
      READ(10) JMAX,KMAX,IMAX
      write(*,*) 'ii(flow direction),jj,kk from parameter are'
     .           ,ii,jj,kk
      write(*,*)  'imax(flow direction),jmax,kmax from input file'
     .           ,imax,jmax,kmax
      if (ii.ne.imax.or.jj.ne.jmax.or.kk.ne.kmax) go to 999
      READ(10)
     > (((Y(i,j,k),J=1,JMAX),K=1,KMAX),I=1,IMAX),
     > (((Z(i,j,k),J=1,JMAX),K=1,KMAX),I=1,IMAX),
     > (((X(i,j,k),J=1,JMAX),K=1,KMAX),I=1,IMAX)
********************** WRITE *******************************
      call  qdopen(11,'grid.x',je)
      call qdputi(11,imax,je)
      call qdputi(11,jmax,je)
      call qdputi(11,kmax,je)
      idh=imax*jmax*kmax
      call qdpuea(11,x,idh,je)
      call qdpuea(11,y,idh,je)
      call qdpuea(11,z,idh,je)
      call qdclos(11,je)
      go to 9999
  999 write(*,*) 'imax,jmax,kmax are:',imax,jmax,kmax
      write(*,*) 'change parameter and rerun'
 9999 stop
      end
```

THE GRID STORAGE SIZE

	STORAGE SIZE(BYTES)	RATIO
ASCII	5108119	4
CRAY BINARY	2531328	2
BDX BINARY	1261568	1

THE GRID SIZE : 65 BY 33 BY 49

- *INTRODUCTION*

- *GRID FILE NAMING SYSTEM*

- *GRID GENERATION PROCEDURE*

- *GRID STANDARD -- PLOT3D BDX BINARY*

- *GRID STORAGE*

13

GRID STORAGE

- ● ALL USERS PUT A NEW GRID

    IN A PUBLIC DIRECTORY

- ● WRITE GRID INFORMATION IN A GRID

    LOG BOOK

GRID INFORMATION	
GRID NAME:	SIZE:
TYPE:	PACKING:
SYMMETRY:	GRID GROGRAM:
CREATED BY:	DATE:
NONDIMENSIONALIZED BY:	
INITIAL SPACE FROM WALL:	
INPUT FILES USED TO CREATE GRID:	
COMMENTS:	

CONCLUDING REMARKS

AFTER MANY YEARS OF TRIAL AND ERROR, THE SIMPLE GRID NAMING SYSTEM WAS ESTABLISHED. THE PLOT3D BDX BINARY WAS SELECTED TO BE THE STANDARD OF GRID STORAGE. UNLESS A BETTER WAY OF GRID MANAGEMENT IS FOUND, THIS SYSTEM WILL BE USED FOR YEARS TO COME.

GRID GENERATION FOR A COMPLEX AIRCRAFT CONFIGURATION

by
Jim Bruns
Sverdrup Technology Inc.
Lewis Research Center Group

ABSTRACT:

The procedure used to create a grid around the F/A-18 fighter aircraft will be presented. This work was done for the NASA High Alpha Technology Program. As part of this program, Lewis is numerically and experimentally investigating the flow in the F/A-18 inlet duct at high angle of attack. A grid was needed which could be used to calculate both the external and internal flow around the F/A-18. The grid had to be compatible with the cfd codes PARC3D and CFL3D.

The programs used to create this grid were I3GVIRGO and GRIDGEN. A surface definition used to create the grid was obtained from MCAIR and was composed of numerous files each containing point definition of a portion of the aircraft. These files were read into the geometry manipulation program I3GVIRGO, where they were modified and grouped into smaller GRIDGEN database files. Next, the block outlines and boundary conditions were specified in the GRIDBLOCK program. The GRIDGEN2D program was used to create the surface grid on the block faces, and GRIDGEN3D was used to create the full 3-D grid.

Some common problems with block interfaces and surface definition will be discussed. Also, some general feelings about GRIDGEN will be given.

GRIDGEN PROGRAM FLOWCHART

```
            ┌─ I3GVIRGO      : Geometry database manipulation.
            │       ↓
    IRIS ───┼── GRIDBLOCK    : Setup blocking structure.
            │       ↓
            └── GRIDGEN2D     : Create grid on faces of each block.
                    ↓
    CRAY ──── GRIDGEN3D       : Create grid in the interior of each block.
                    ↓
    IRIS ──── GRIDVUE3D       : Examine finished grid.
```
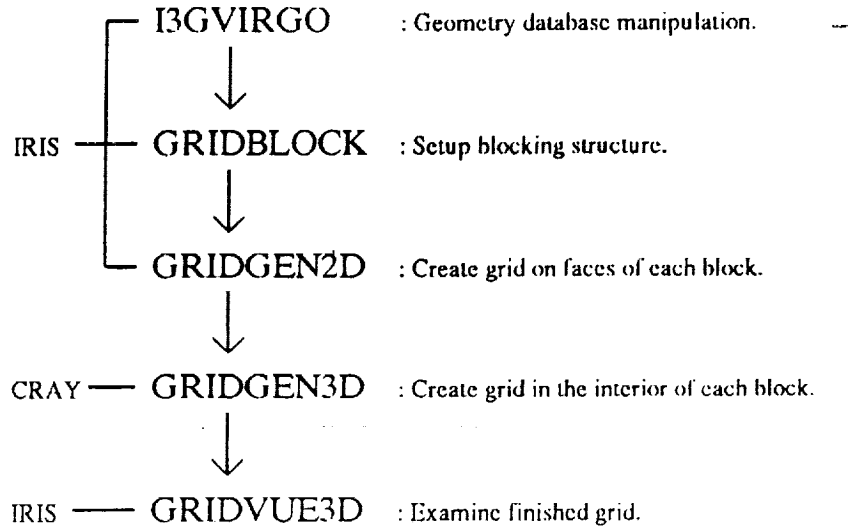
Geometry Database Manipulation

Most geometry databases come from CAD programs.

GRIDGEN needs databases which are a set of points defining surface.

I3GVIRGO is used mainly for geometry database manipulation.

-Join together or split surface definitions.
-Scale, translate, and rotate surfaces.
-Redistribute points on surface.
-Intersect surfaces.
-Can output files in GRIDGEN database format.

GRIDGEN allows very general database surface definitions.

-Database topology and grid surface topology are independent.

18

I3GVIRGO Screen

GRIDBLOCK Screen

GRIDBLOCK Screen

GRIDGEN2D
Screen

20
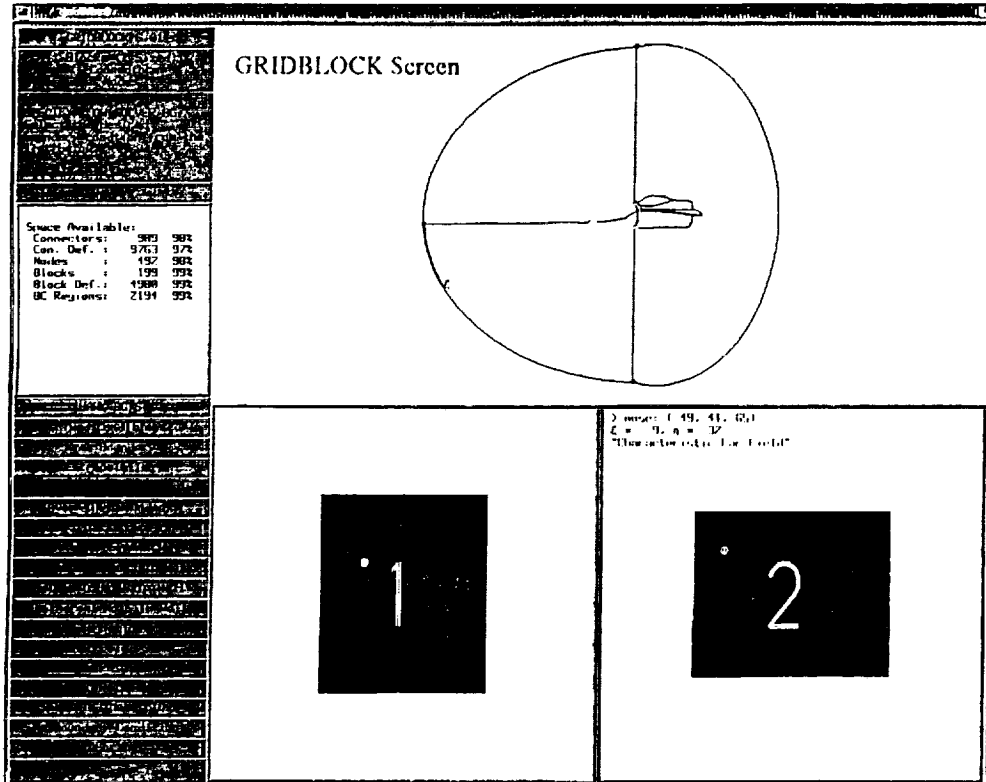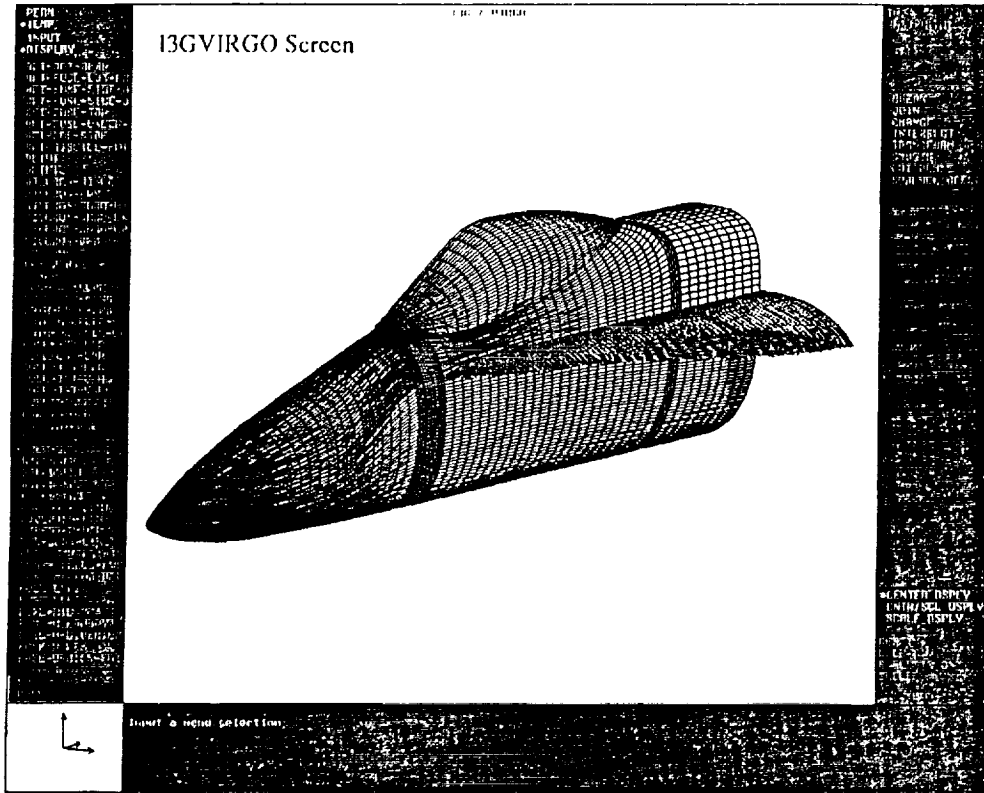
-Creates the 3-D volume grid.

-Runs on Cray but can easily be modified to run on IRIS.

-Algebraic and Elliptic solvers.

-Notifies user of negative Jacobians, cell aspect ratio, and truncation error.

-Batch code.
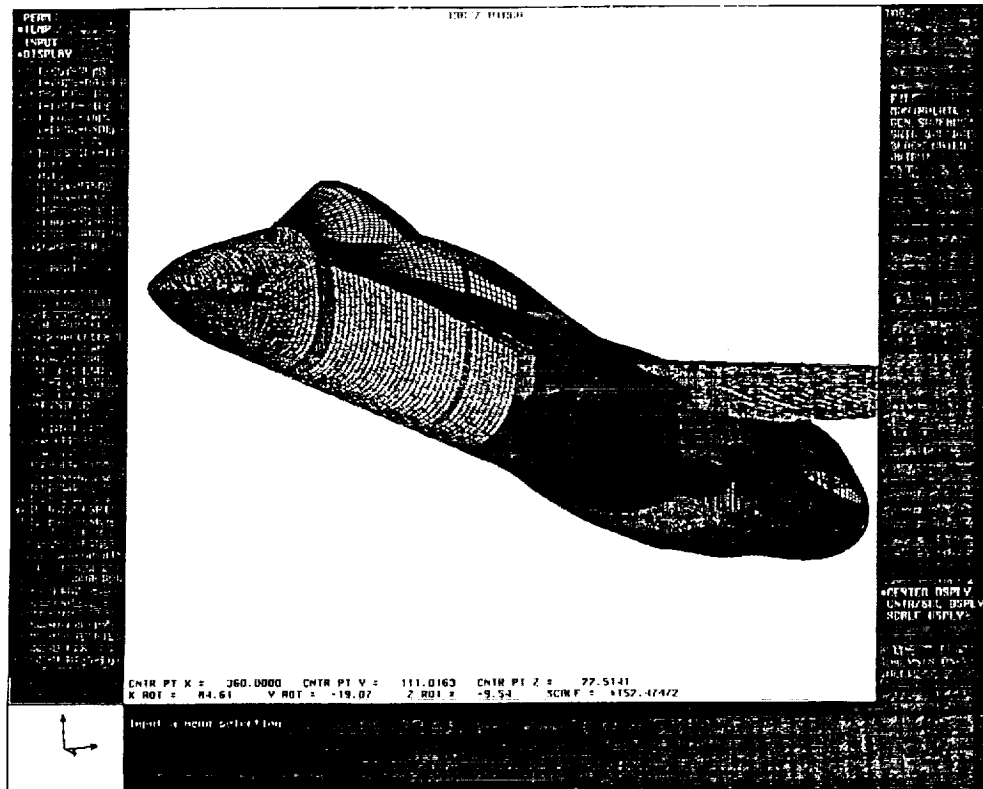
Grouping of F-18 Geometry into Different Database Files
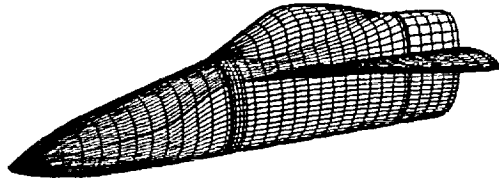
Forebody and LEX
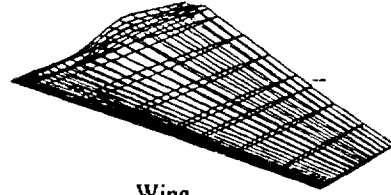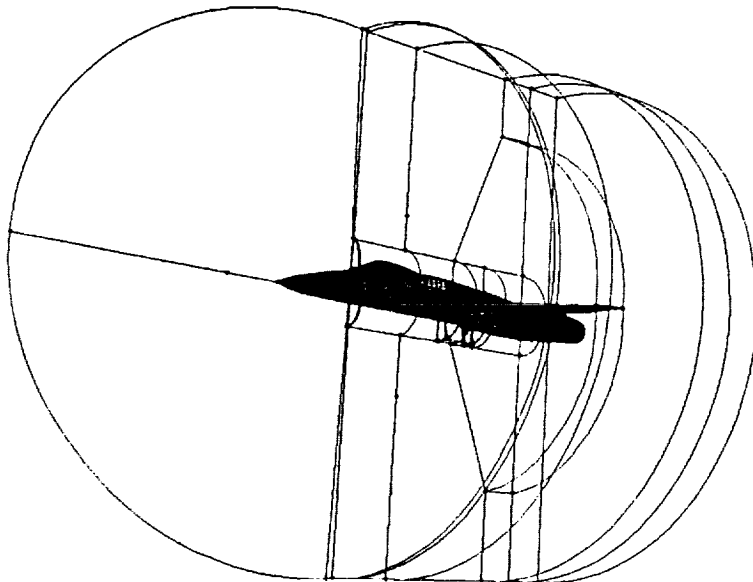
Wing
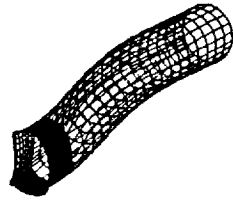
Inlet
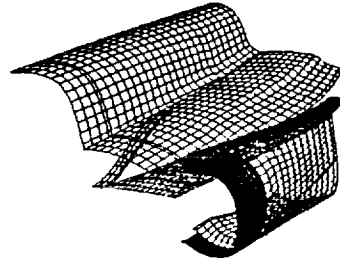
Fuselage and Cowl

GRIDBLOCK Screen

Space Available:
Connectors: 681 68%
Con. Def. :   2285 77%
Nodes   :   332 66%
Blocks   :   161 87%
Block Def.:   3385 79%
BC Regions: 1582 98%

22

GRIDBLOCK Screen

F-18 Grid Around
Inlet Area
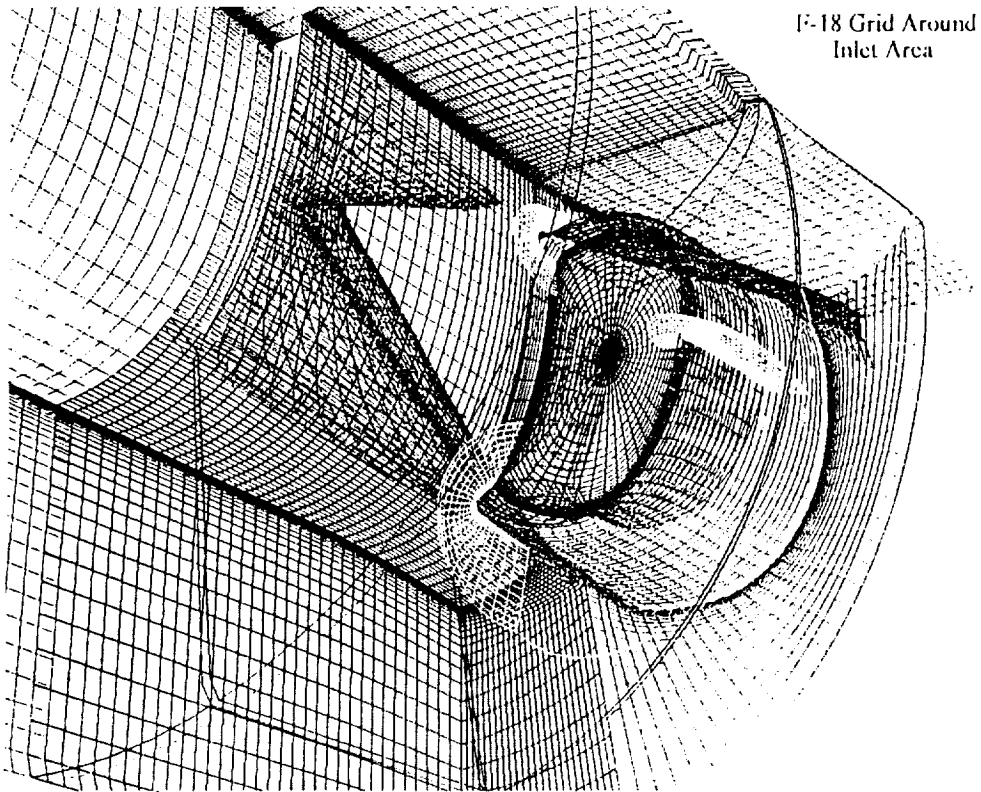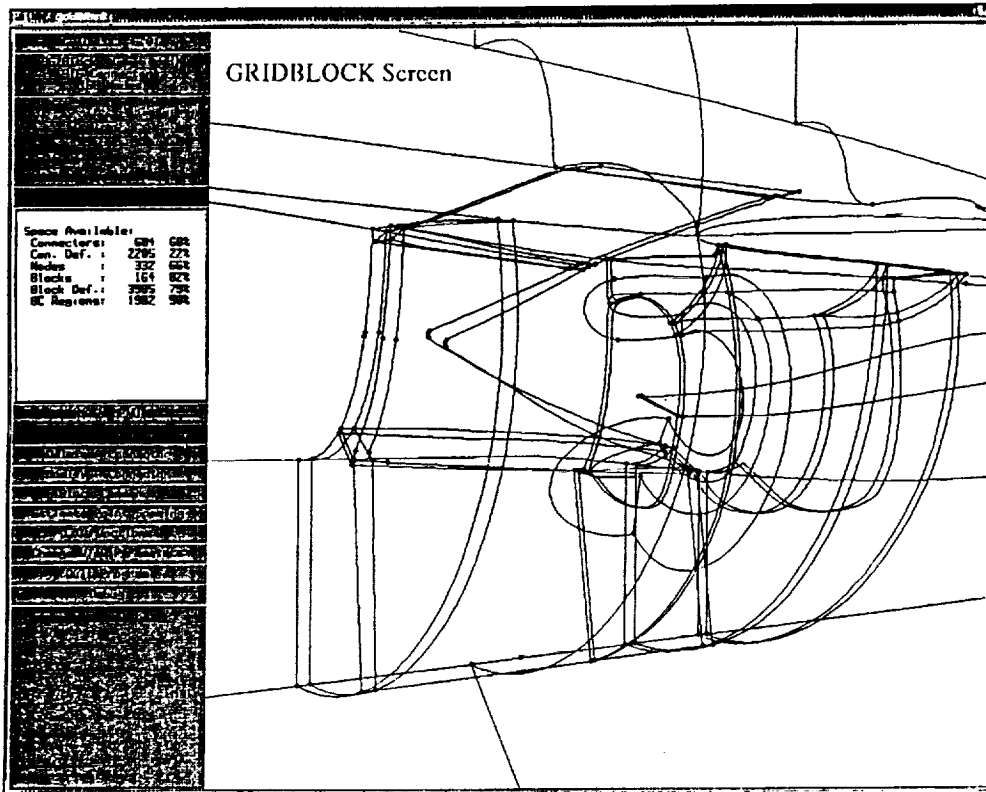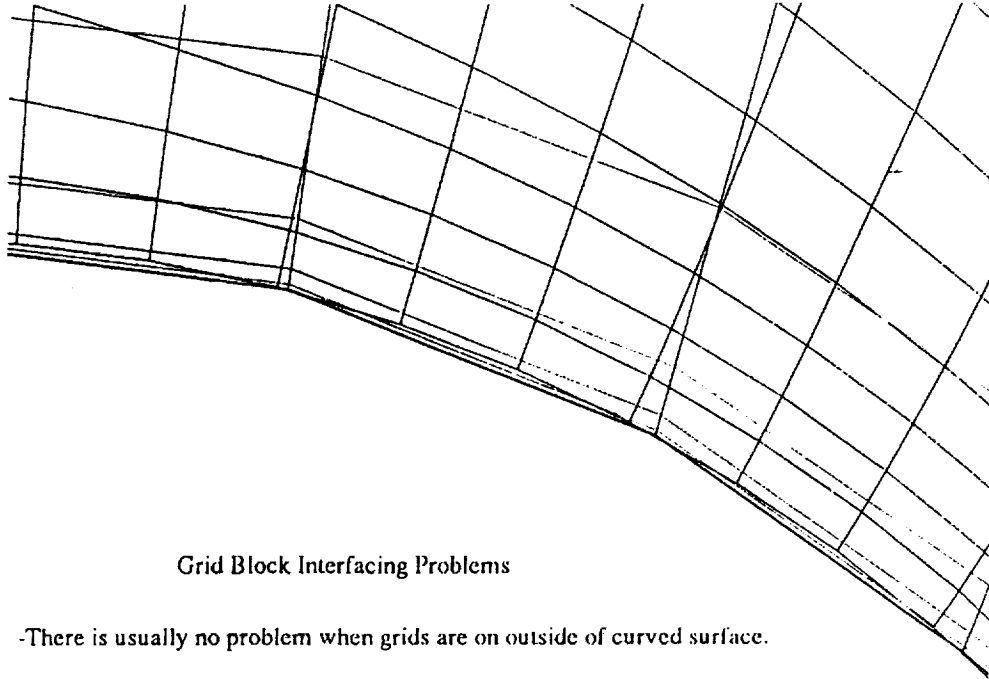
Grid Block Interfacing Problems

-There is usually no problem when grids are on outside of curved surface.
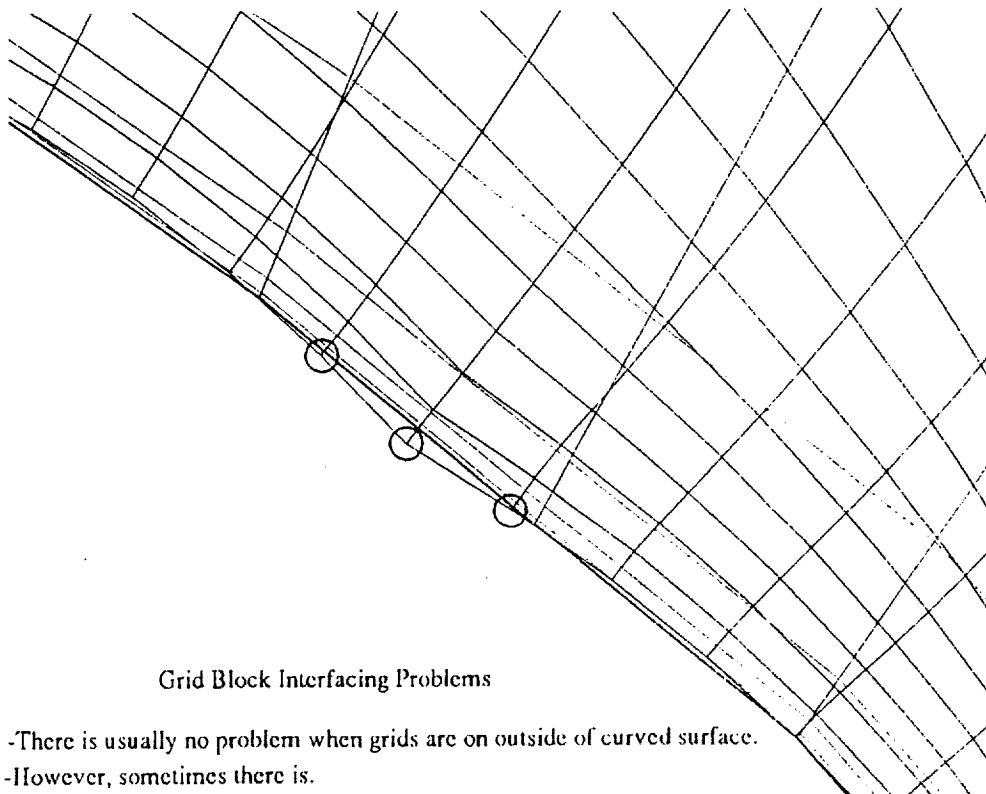
Grid Block Interfacing Problems

-There is usually no problem when grids are on outside of curved surface.
-However, sometimes there is.

24

Possible Grid Interfacing Problem

Circled points are outside of other block.

F-18 Geometry
Red = Viscous
Green = Euler

F-18 Geometry: Grid #2
Red = Viscous Blocks
Green: Euler Blocks
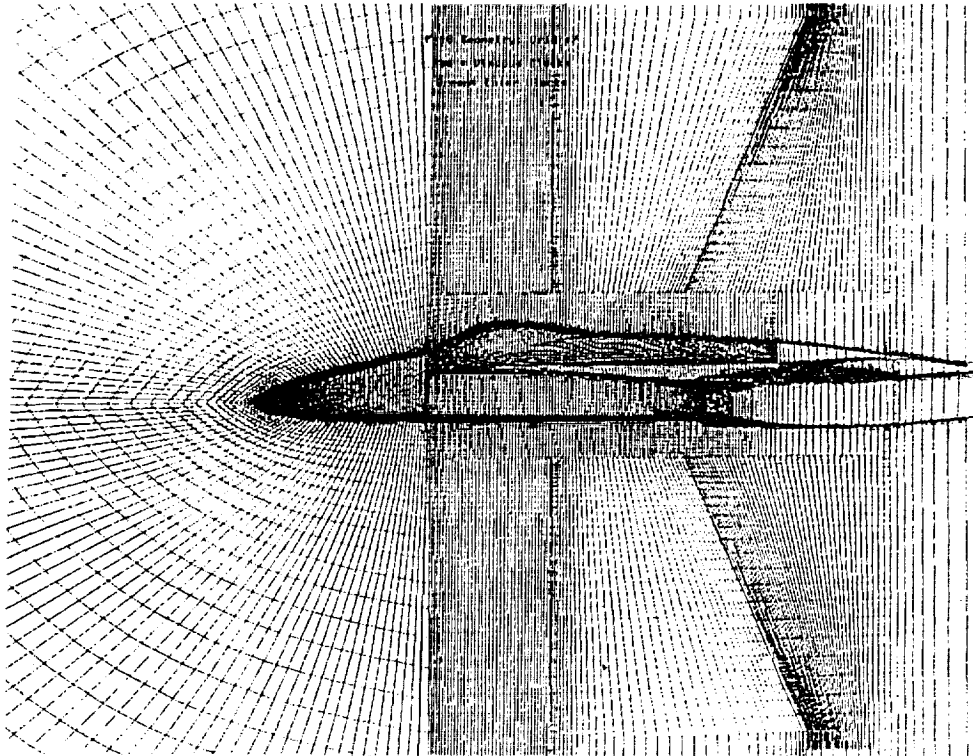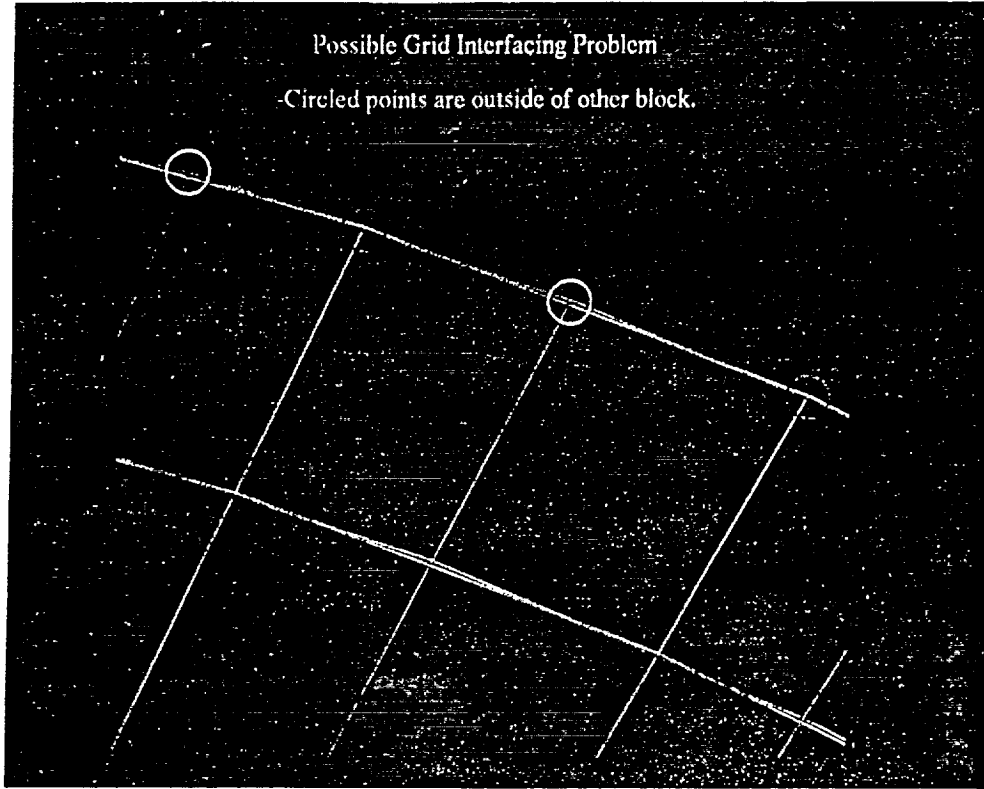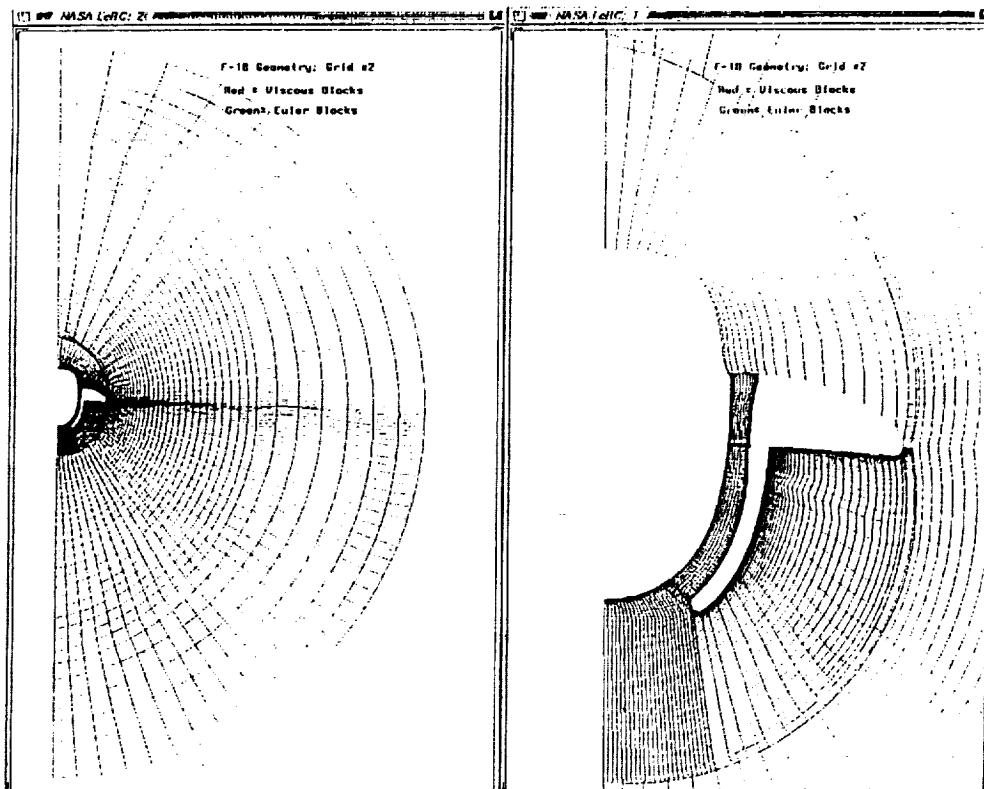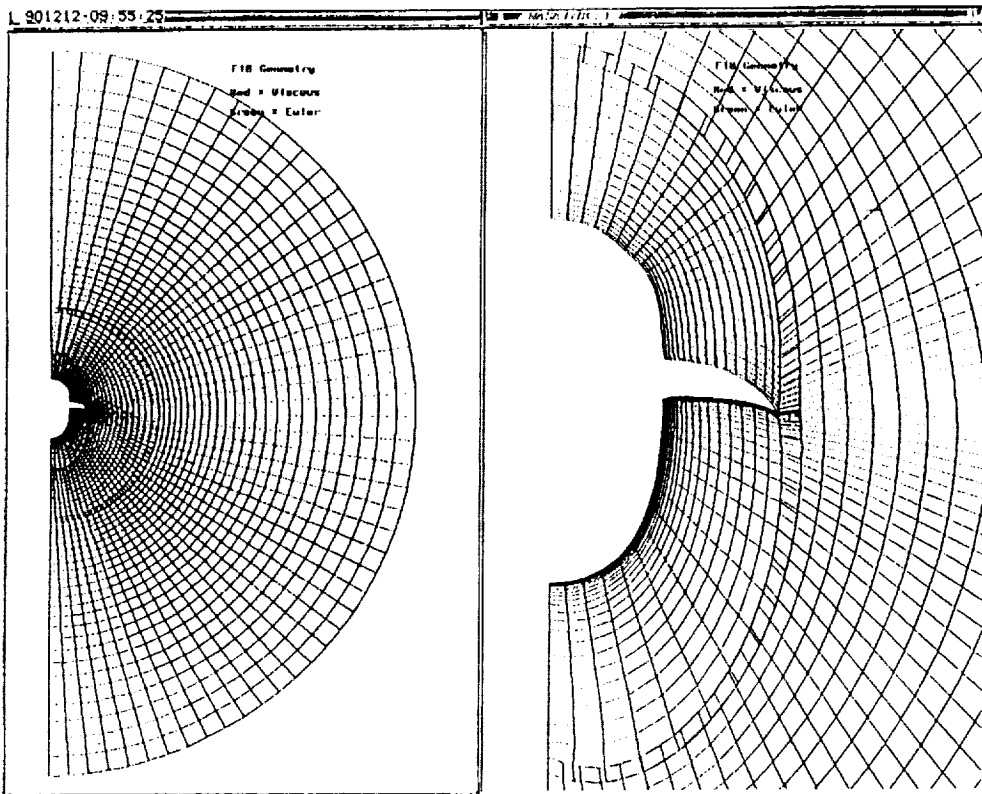
Summary

-GRIDGEN grid generator is an effective program for producing complex multi-blocked grids.

-Accurate surface defintions and block interfaces must be handled carefully.

27

QX_MAN: Q and X File MANipulation
written by
Mark A. Krein
NASA Lewis Research Center

presented by
Kin C. Wong
NASA Lewis Research Center

QX_MAN is a grid and solution file manipulation program written
primarily for the PARC code and the GRIDGEN family of grid
generation codes. QX_MAN combines many of the features frequently
encountered in grid generation, grid refinement, setting up initial
conditions, and post processing. QX_MAN allows the user to
manipulate single block and multi-block grids (and their
accompanying solution files) by splitting, concatenating, rotating,
translating, re-scaling, and stripping or adding points. In
addition, QX_MAN can be used to generate an initial solution file
for the PARC code. The code was written to provide several formats
for input and output in order for it to be useful in a broad
spectrum of applications.

BASIC FEATURES

- CONCATENATE TWO OR MORE GRIDS

- EXTRACT GRID

- ENHANCE OR DE-ENHANCE GRID

- ROTATE, TRANSLATE, RESCALE GRID

- CREATE PARC INITIAL CONDITION FILE

- CONVERT TO GRIDGEN FORMAT

QX_MAN FLOW DIAGRAM

QX_MAN IS A GRID AND SOLUTION FILE MANIPULATION PROGRAM WRITTEN PRIMARILY FOR USERS OF THE PARC CODE AND THE GRIDGEN FAMILY OF GRID GENERATION CODES. QX_MAN PROVIDES SEVERAL FEATURES WHICH ARE REPEATED ON A FREQUENT BASIS IN THE COURSE OF GRID GENERATION, ESTABLISHING CODE INITIAL CONDITIONS, AND DISPLAYING CODE RESULTS. THESE FEATURES HAVE BEEN COMBINED INTO A GENERAL PROGRAM WHICH PROVIDES CONVENIENT IMPLEMENTATION.

USER INPUT:
IMULTI=1 FOR OUTPUT GRID FILE
IPARC=1   FOR OUTPUT SOLUTION / PARC RESTART FILE
IONE=1   READ ONE GRID FILE AND USE AS IS
          (THIS OPTION SKIPS MUCH CODE)

IONE=1

IONE.NE.1

ADDITIONAL USER INPUT:
NTOT=   NUMBER OF GRID BLOCKS TO READ
NCREATE= NUMBER OF BLOCKS TO CREATE
IODUP=1   TO DUPLICATE OPERATIONS ON
          SOLUTION FILE
NBLOCK=   FINAL NUMBER OF OUTPUT BLOCKS
IMANY=   NUMBER OF FILES TO READ

READ ALL INPUT FILES:
ENTER FILE FORMAT AND FILE NAMES AS PROMPTED

IONE.NE.1

IONE=1

MAIN PROGRAM LOOP:
EXECUTED ONCE FOR EACH INTERMEDIATE
BLOCK AND EACH OUTPUT BLOCK
[DO 40 I=1,(NCREATE + NBLOCK)]

IF (I LE NCREATE)

IF (I GT NCREATE)

IF (I LT (NCREATE + NBLOCK))

INTERMEDIATE BLOCK CREATION:
USER SELECTED OPTIONS FOR CREATING A NEW BLOCK:
0 : ROTATION, TRANSLATION, RESCALING
1 : CONCATENATION
2 : SPLITTING

OUTPUT BLOCK:
USER SELECTS DESIRED OPTIONS
(AN OUTPUT BLOCK CAN BE A NEWLY CREATED BLOCK)
0 : OUTPUT A BLOCK AS IS (ROTATION, TRANSLATION, RESCALING
    MAY BE PERFORMED)
1 : CONCATENATION (CREATES A NEW BLOCK)
2 : SPLITTING (CREATES A NEW BLOCK)
3 : ENHANCEMENT / DE-ENHANCEMENT (CREATES A NEW BLOCK)

IF (I GT (NCREATE + NBLOCK))

OPTION TO PERFORM GLOBAL MODIFICATIONS:
MODIFICATIONS PERFORMED ON ALL OUTPUT GRID BLOCKS:
ROTATION, TRANSLATION, AND RESCALING MAY BE SELECTED

FINAL OUTPUT:
WRITE DESIRED OUTPUT FILES IN
DESIRED FORMAT AND QUIT

CONCATENATING AND EXTRACTING GRIDS
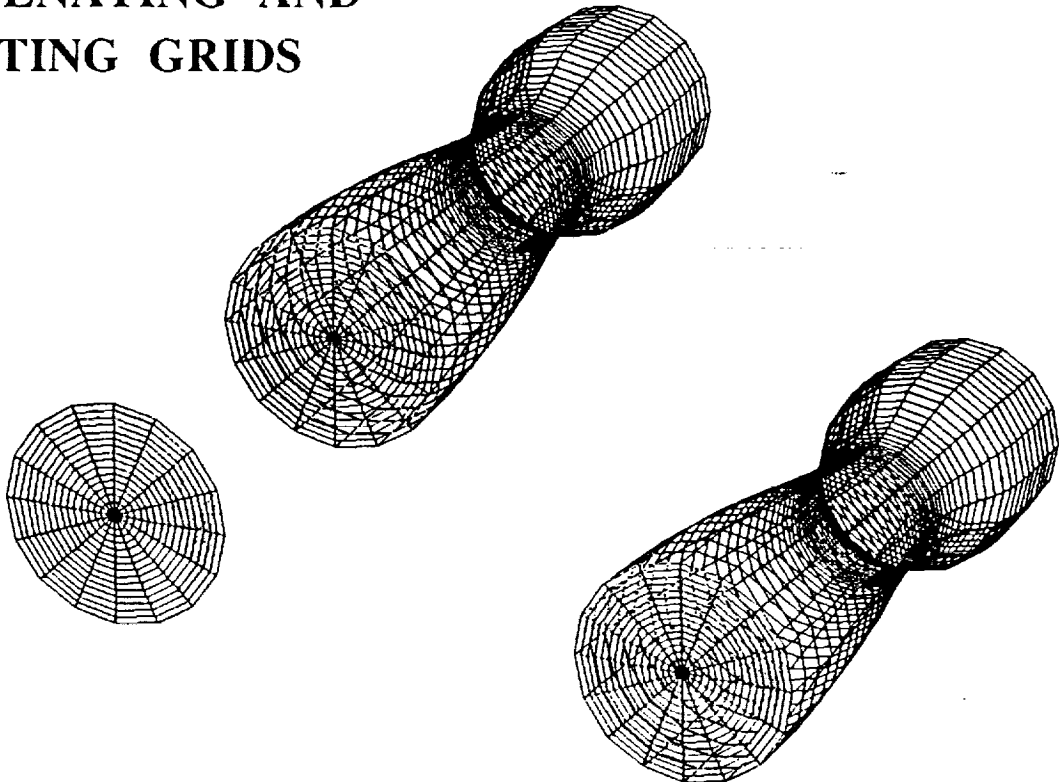
ENHANCING AND DE-ENHANCING GRIDS
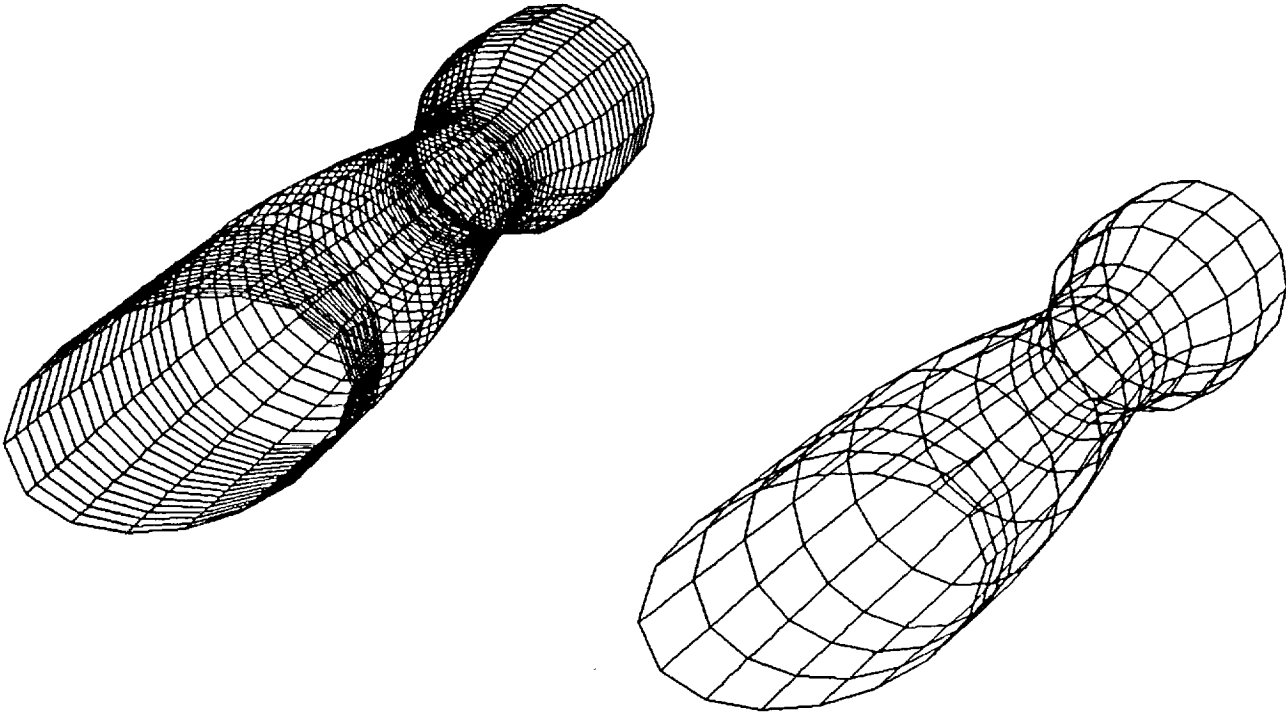
GENERATING PARC INITIAL CONDITIONS

- FROM OLD RESTART FILE

- FROM OLD Q FILE

- FROM SPECIFIED REFERENCE CONDITIONS

INPUT AND OUTPUT FORMATS

INPUT

- MULTI-BLOCK BDX
- NON-BLOCKED BDX
- MULTI-BLOCK UNFORMATTED
- NON-BLOCKED UNFORMATTED
- PARC3D RESTART FILE
- GRIDGEN3D FORMAT

OUTPUT

- MULTI-BLOCK BDX
- GRIDGEN3D FORMAT

Summary of Discussion
Session I
Chair: Danny Hwang

GRIDGEN is a useful and powerful interactive software package that is available in the United States. Its capabilities include interactive domain decomposition (GRIDBLOCK) and surface grid generation (GRIDGEN2D), 3D field grid generation (GRIDGEN3D), and interactive viewing capability (GRIDVIEW). GRIDGEN has been the primary workhorse for the forebody-inlet/nozzle-afterbody/duct computational grids at Lewis Research Center since early 1990.

GRIDGEN appears to be well received by engineers who are dealing with complex geometries because of its interactive zoning capability. Since it is a large code, it takes time for a user to acquire full knowledge of all available features. However, some engineers indicated that GRIDGEN is too large to learn and use for their component research. GRIDGEN has not yet been tested for rotating machinery, which does not require many blocks, but requires a special grid structure and quality by its flow solvers.

Addition of new schemes such as the elliptic/hyperbolic hybrid grid generation scheme, grid adaptation scheme, and integration of GRIDBLOCK with GRIDGEN2D would further enhance GRIDGEN's capability.

The use of multiple grid generation codes depends on the application, complexity of geometry, and user preference. Engineers who repeatedly use similar geometry for their analysis prefer hardwired, customized codes to large general purpose codes. As design changes, however, another code is likely to be needed. Those who need multiblock grids for complex geometries and work on various different geometries prefer a versatile general purpose code to the customized codes.

The use of "itrans" is sufficient to transfer binary data between Cray and IRIS. The binary file created by "itrans," however, is transferable only between Cray and IRIS workstations.

The Plot3D BDX binary should be considered as a candidate standard for binary data exchange and storage because it not only can be transferred between all mainframe computers and workstations, but also requires substantially less disk space.

SESSION II

GRID GENERATION FOR ROTATING MACHINERY

Chairs: Dennis L. Huff and David P. Miller

TCGRID - A Three-Dimensional C-Grid
Generator for Turbomachinery
by
Rodrick V. Chima
NASA Lewis Research Center

ABSTRACT

A fast three-dimensional grid code for turbomachinery has been developed. The code, called TCGRID for Turbomachinery C-GRID, can generate either C or H-type grids for fairly arbitrary axial or radial turbomachinery geometries. The code also has limited blocked-grid capability and can generate an axisymmetric H-type grid upstream of the blade row or an O-type grid within the tip clearance region.

Hub and tip geometries are input as a simple list of (z,r) pairs. All geometric data is handled using parametric splines so that geometries that turn 90 degrees (e.g. centrifugal impellers) can be handled without difficulty. Blade input is in standard MERIDL or Lewis compressor design code format. TCGRID adds leading and trailing edge circles to MERIDL geometries and intersects the blade with the hub and tip if necessary using a novel intersection algorithm.

The following procedure is used to generate the grids:

1. A coarse, equally spaced meridional grid is generated algebraically between the specified hub and tip.

2. Blade coordinates are found at the meridional grid points by interpolation of the input blade geometry.

3. Two-dimensional blade-to-blade grids are generated along the meridional grid lines in $(m, \bar{r}\theta)$ coordinates using a version of the GRAPE code developed by Steger and Sorenson. Here, m is the arc length along the meridional surface, and $\bar{r}$ is some mean radius. The GRAPE code allows arbitrary specification of inner and outer boundary points, then generates interior points as the solution of a Poisson equation. Forcing terms in the Poisson equation are chosen to maintain the desired grid spacing and angles at the boundaries.

4. The $(m, \bar{r}\theta)$ coordinates are transformed back to (z, r, θ).

5. The two-dimensional grids are reclustered spanwise using a hyperbolic tangent stretching function to make a full three-dimensional grid.

6. The (z, r, θ) coordinates are transformed back to (x, y, z).

7. Axisymmetric upstream blocks are generated using transfinite interpolation and tip clearance blocks are generated using Hermite polynomials.

Output is in PLOT3D format, which can also be read by the RVC3D (Rotor Viscous Code 3-D) Navier-Stokes code for turbomachinery. Intermediate 2-D and 3-D grids useful for debug and other purposes can also be output using a convenient output flag.

The attached figure shows a 185x40x49 grid generated for a transonic fan (NASA rotor 67.) The grid has 362 600 points and was generated in less than a minute on a Cray Y-MP. The initial grid spacing is about 0.015 mm at the blade, 0.03 mm at the hub and 0.045 mm at the tip. The C-shaped grid gives good resolution of the round leading edge of the blade and of the wake. It has been used by the author for detailed calculations of the performance of this fan.

PRECEDING PAGE BLANK NOT FILMED

TCGRID (TURBOMACHINERY C-GRIDS)

BY R. V. CHIMA

FEATURES

- 3-D C- OR H-GRID GENERATOR FOR TURBOMACHINERY
- APPLICABLE TO AXIAL OR RADIAL TURBOMACHINES
- FAST ELLIPTIC GRID SCHEME WITH CONTROL OVER ANGLES & SPACING
- GENERAL BLADE INPUT
  MERIDL OR COMPRESSOR DESIGN CODE FORMAT
  CODE ADDS L.E. & T.E. CIRCLES TO MERIDL DATA
  CODE INTERSECTS BLADE WITH HUB & TIP
- BLOCKED GRID CAPABILITY
  O-GRID FOR TIP CLEARANCE (HERMITE POLYNOMIALS)
  H-GRID FOR INLET REGION (TRANSFINITE INTERPOLATION)
- OUTPUT IN PLOT3D, RVC3D FORMAT

DETAILS

- 2-D BLADE-TO-BLADE GRIDS GENERATED ON SEVERAL
  SURFACES OF REVOLUTION USING SORENSON-STEGER GRAPE CODE
- 2-D GRIDS STACKED & RECLUSTERED ALONG SPAN
  TANH CLUSTERING RESOLVES ENDWALL BOUNDARY LAYER REGIONS

STATUS

- STACKED 2-D GRIDS SIMPLER AND FASTER THAN FULL 3-D SOLVER
  CPU TIME ABOUT 1 MIN ON CRAY Y-MP FOR 185x40x49 GRID
- TCGRID IS TESTED, DOCUMENTED, & AVAILABLE

TCGRID - PROCEDURE

0. INPUT, INPUT PROCESSING

1. GENERATE ALGEBRAIC MERIDIONAL GRID

2. INTERPOLATE BLADE ONTO MERIDIONAL GRID

3. GENERATE SEVERAL BLADE-TO-BLADE GRIDS
    WITH GRAPE CODE IN $(m, \bar{r}\theta)$ SYSTEM

4. TRANSFORM $(m, r, \bar{r}\theta) \Rightarrow (z, r, \theta)$

5. RECLUSTER SPANWISE USING TANH STRETCHING

6. TRANSFORM $(z, r, \theta) \Rightarrow (x, y, z)$

7. GENERATE UPSTREAM BLOCKS BY TRANSFINITE INTERPOLATION
    TIP CLEARANCE BLOCKS USING HERMITE POLYNOMIALS

TCGRID - PROCEDURE

0. SIMPLE NAMELIST INPUT

```
&nam1 merid=0 im=97 jm=31 km=33 itl=17 icap=12 k2d=3 &end
&nam2 nle=19 nte=8 dsle=.0028 dste=.0012 dshub=.00004 dstip=.00004
      dswte=.0012 dswex=.040 dsthr=1. dsmin=.00004 dsmax=.002
      dsra=.452 rcorn=.017 &end
&nam3 iterm=100 idbg=0 0 0 0 0 0 0 0 aabb=.5 ccdd=.45 &end
&nam4 zbc=-.300    -.080      .215   -.300    -.080      .215
      rbc= .708333 .708333 .708333 .833333 .833333 .833333 &end
'Lou Goldmans annular turbine cascade'
    2    2
Hub z-coordinates
Hub r-coordinates
Tip z-coordinates
Tip r-coordinates
    2   76   36
Blade coordinates
```

TCGRID - PROCEDURE

0. INPUT PROCESSING
   CODE CAN ADD L.E., T.E. CIRCLES TO MERIDL DATASETS
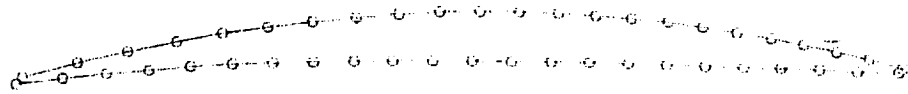
ROTOR 67 - MERIDL BLADE SECTION
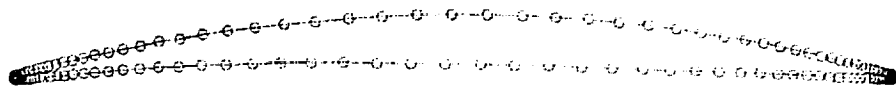
ROTOR 67 - ARBITRARY BLADE SECTION

MERIDIONAL GRID

43

MERIDL BLADE
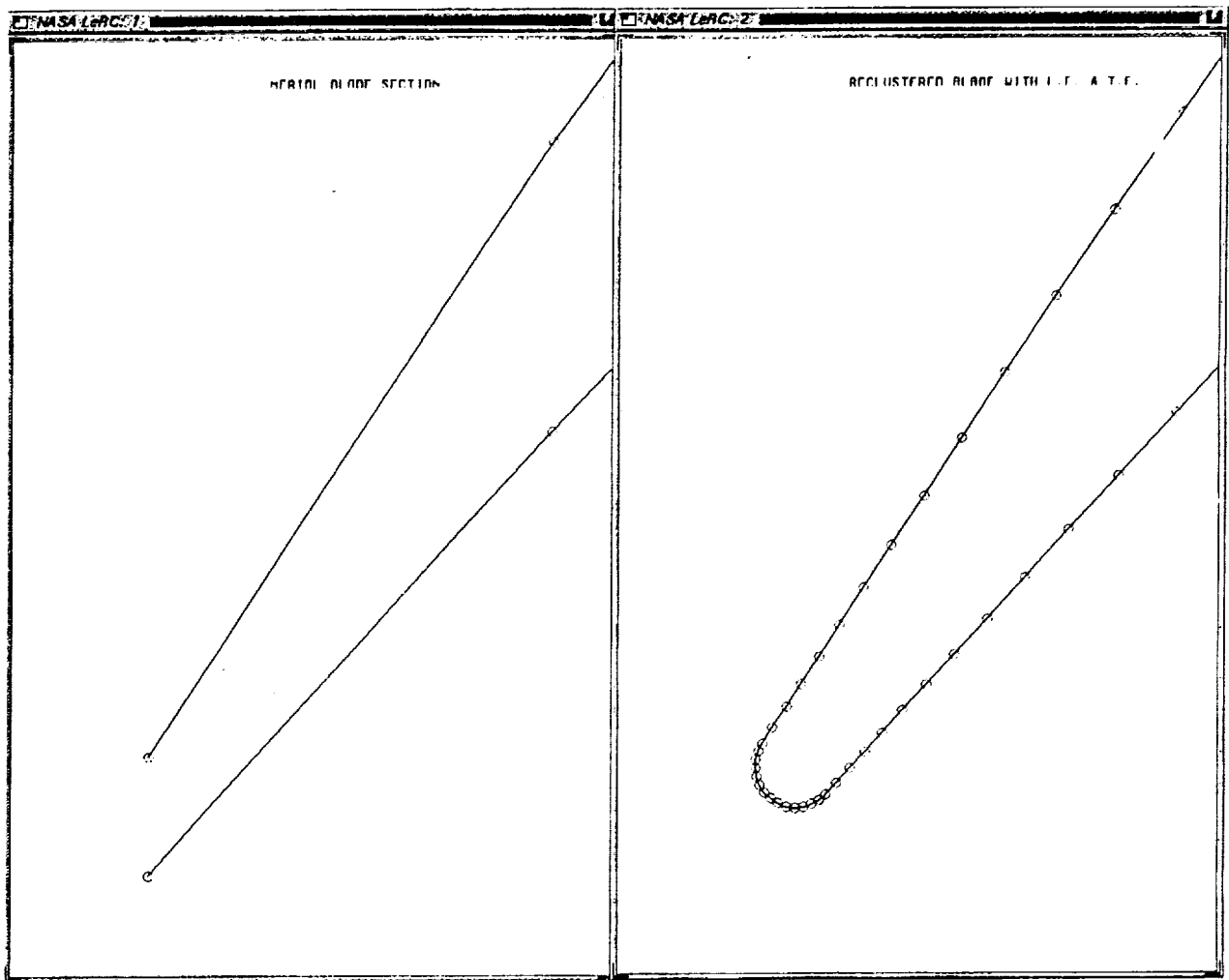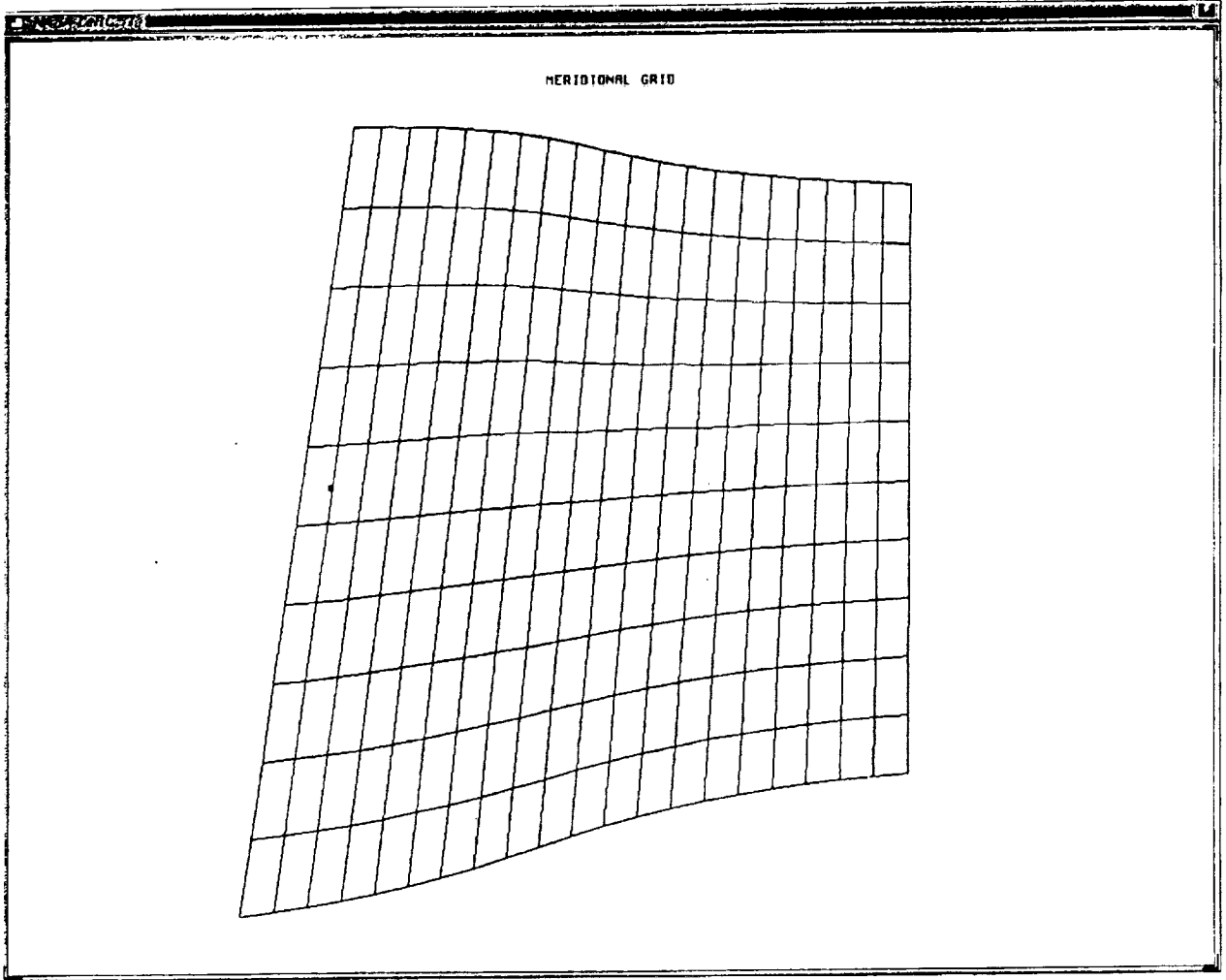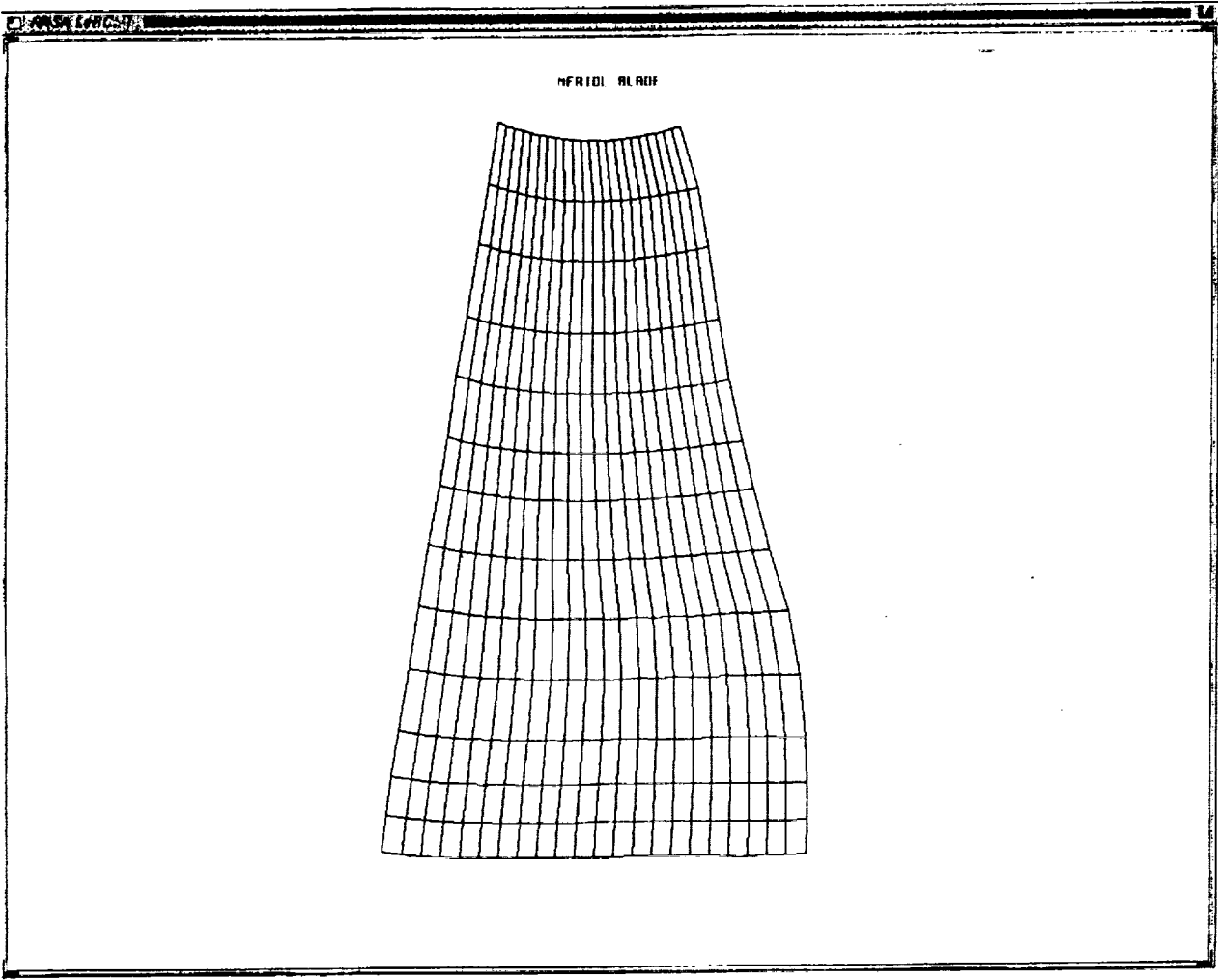
INTERPOLATED BLADE
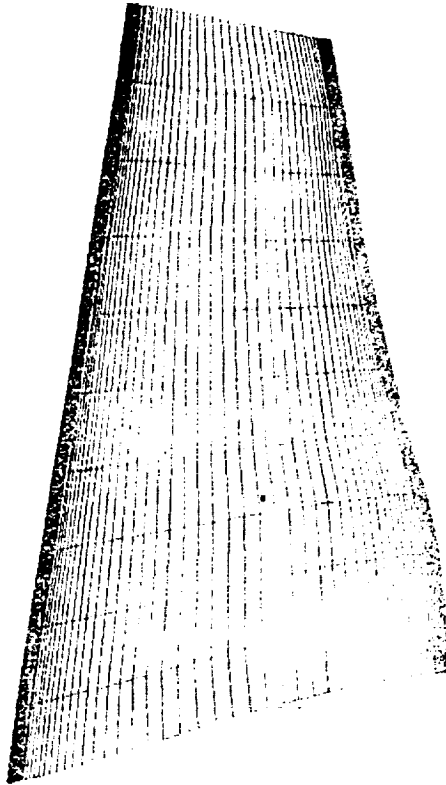
INTERSECTION OF 2 DISCRETE LINES
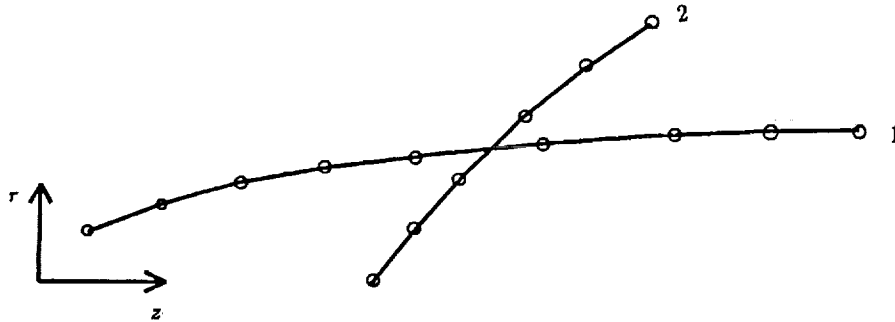
1. Fit lines with parametric splines.

$$z_1(s_1), \qquad r_1(s_1), \qquad z_1' = \frac{dz_1}{ds_1}, \qquad r_1' = \frac{dz_1}{ds_1}$$

$$z_2(s_2), \qquad r_2(s_2), \qquad z_2' = \frac{dz_2}{ds_2}, \qquad r_2' = \frac{dz_2}{ds_2}$$

2. Pick a point on each line.

$$s_1 \to (z_1, r_1)$$

$$s_2 \to (z_2, r_2)$$

INTERSECTION OF 2 DISCRETE LINES
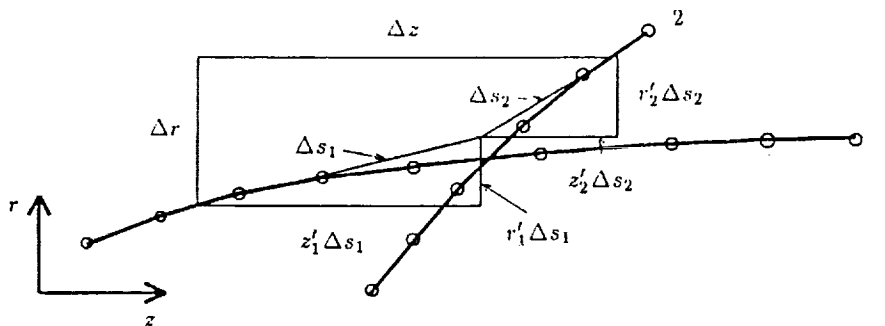
3. Find corrections Δs_1 and Δs_2 such that:

$$z_1' \Delta s_1 + z_2' \Delta s_2 = \Delta z$$

$$r_1' \Delta s_1 + r_2' \Delta s_2 = \Delta r$$

4. Solve for Δs_1 and Δs_2.

$$\Delta s_1 = \frac{r_2' \Delta z - z_2' \Delta r}{z_1' r_2' - z_2' r_1'}$$

$$\Delta s_2 = \frac{z_1' \Delta r - r_1' \Delta z}{z_1' r_2' - z_2' r_1'}$$

5. New points are

$$s_1 = s_1 + \Delta s_1$$

$$s_2 = s_2 + \Delta s_2$$

6. Converges to 8 digits in 3-4 iterations

46

TCGRID - PROCEDURE

3. GENERATE SEVERAL BLADE-TO-BLADE GRIDS WITH GRAPE CODE

- GRAPE CODE BY REECE SORENSON, NASA AMES
  TURBOMACHINERY MODIFICATIONS BY ROD CHIMA, NASA LERC

- 2-D ELLIPTIC GRID GENERATION CODE FOR ARBITRARY AIRFOILS
  POISSON FORCING FUNCTIONS ALLOW SPECIFICATION OF
  GRID SPACING AND ANGLE AT BOUNDARIES

- GRIDS GENERATED IN $(m, \bar{r}\theta)$ SYSTEM

- PERIODIC OUTER BOUNDARY MADE FROM MEAN CAMBER LINE

- C-TYPE GRIDS GIVE GOOD LEADING EDGE RESOLUTION

- LOWER TRAILING EDGE RESOLUTION AVOIDS UNSTEADY FLOWS

- GOOD RESOLUTION OF WAKE

ROTOR 67 BLADE-TO-BLADE GRIDS

ROTOR 67 LEADING EDGE GRID

ROTOR 67 TRAILING EDGE GRID
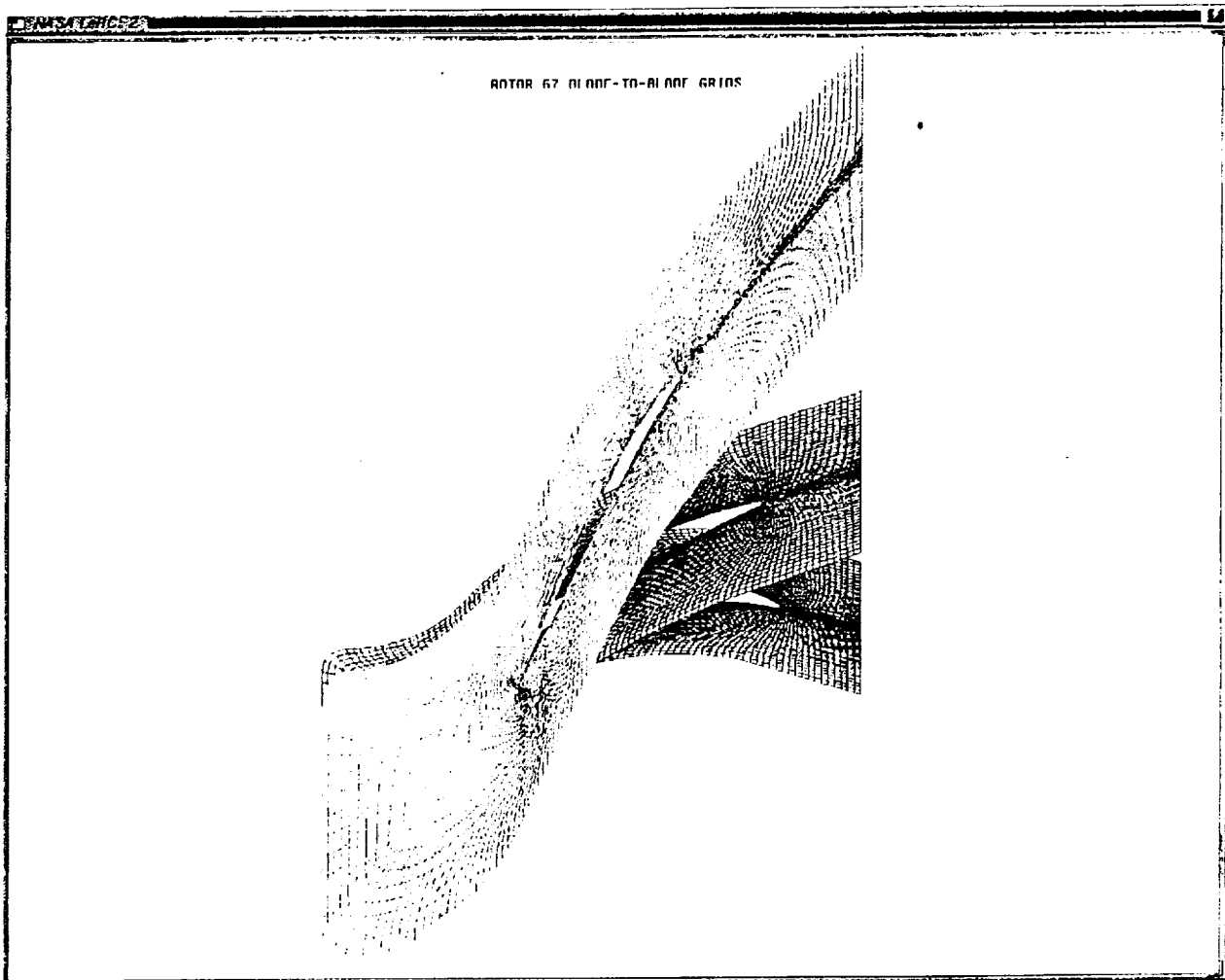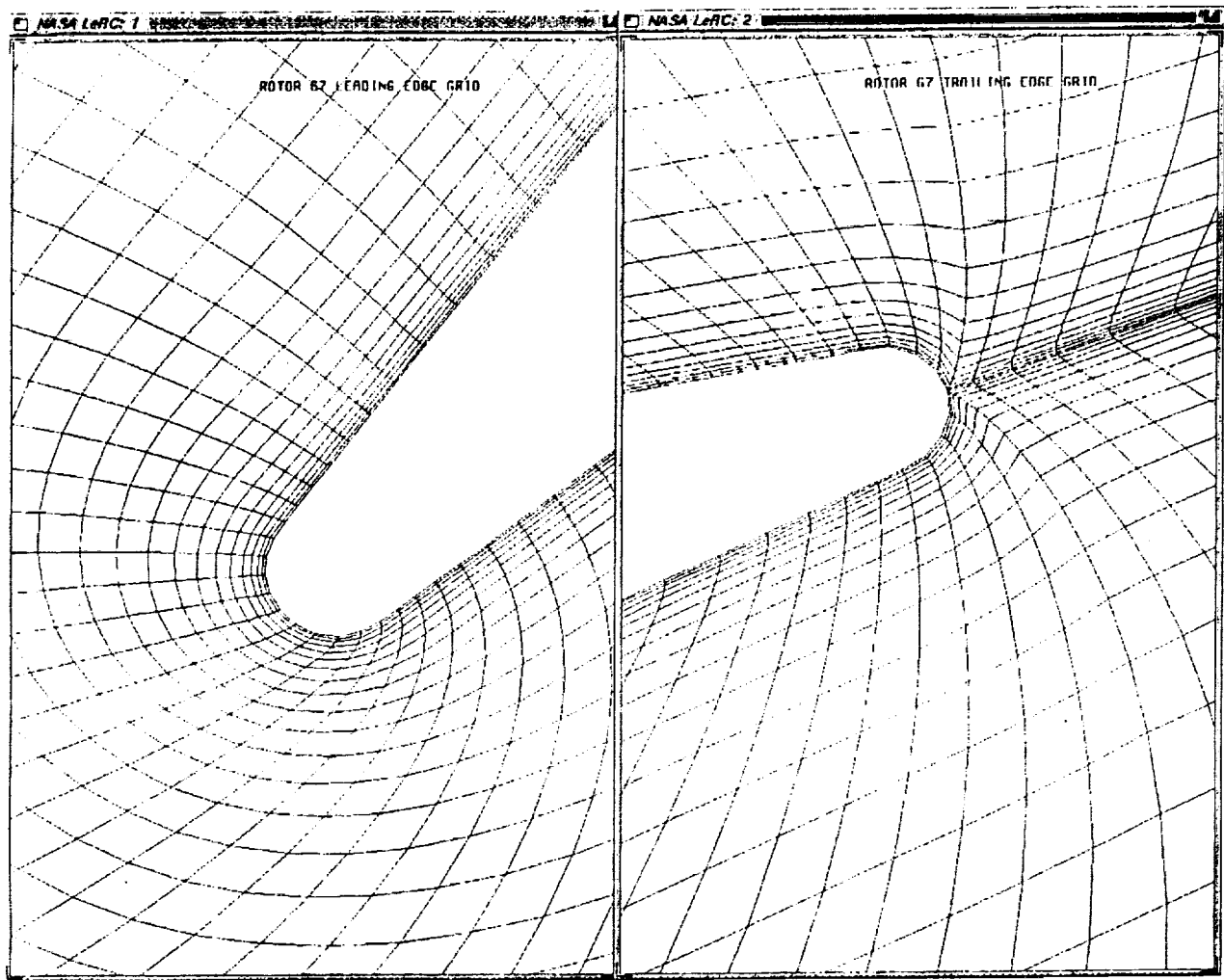
TCGRID - PROCEDURE

4. TRANSFORM $(m, r, \bar{r}\theta) \Rightarrow (z, r, \theta)$

5. RECLUSTER SPANWISE USING TANH STRETCHING

6. TRANSFORM $(z, r, \theta) \Rightarrow (x, y, z)$

ROTOR G7 UNIFORM GRID   ROTOR G7 RECLUSTERED GRID
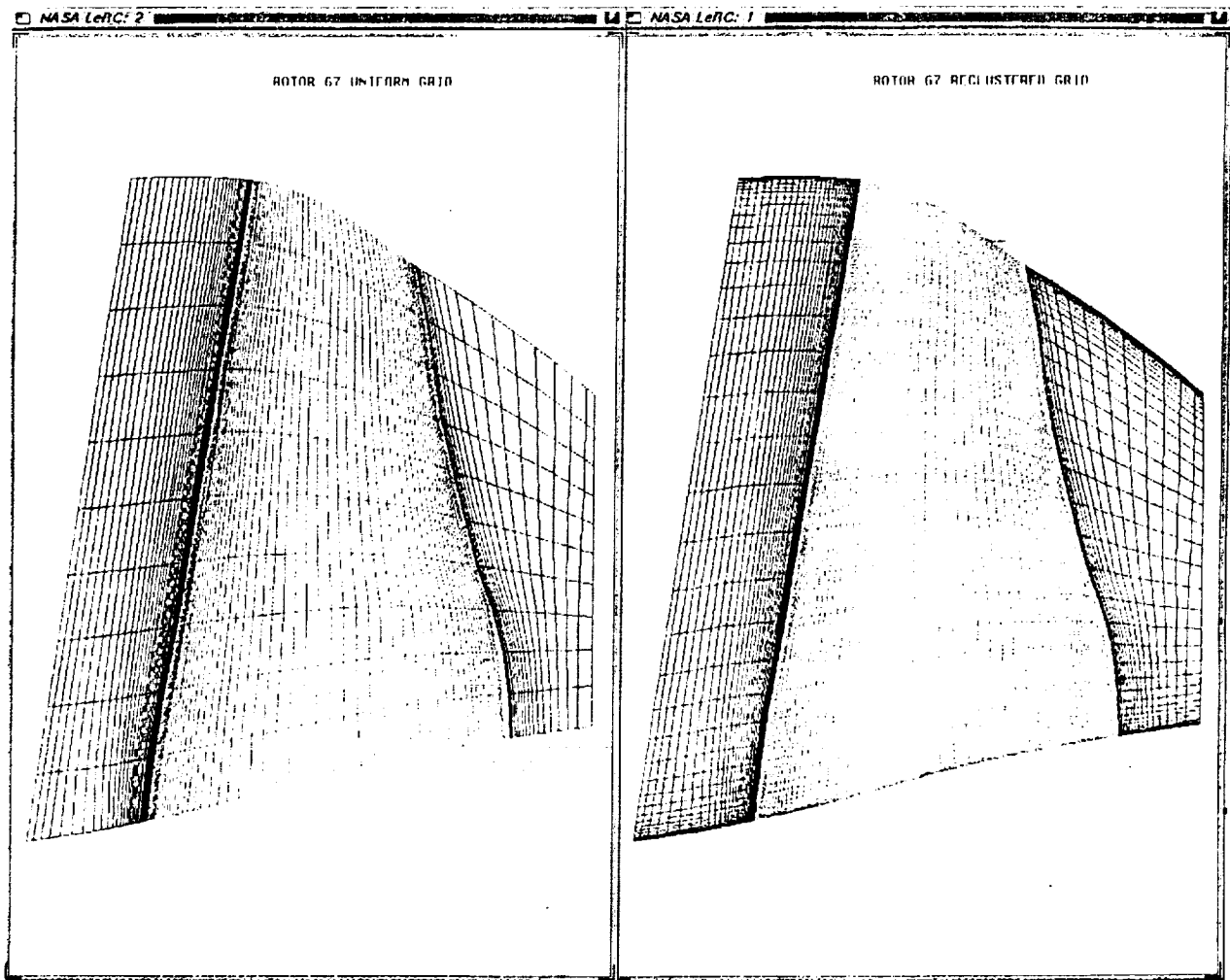
TCGRID - NEAT STUFF

- DEBUG OUTPUT

idbg(i)   Eight integer flag array for writing intermediate grids
to other fortran units. Useful for debug, graphics, and
possibly for grid generation in itself. <Default = 8*0.>
Available options are tabulated below.

i	value	unit	Size	Description
1	1	11	i2d,k2d	2-d meridional grid
2	1	12	npb,nbs,1	3-d blade w/le,te
3	1	13	npb,k2d,1	3-d interpolated blade
4	k	14	im,jm,1	2-d b-b grid at sta. k
5	1	15	im,jm,k2d	3-d grid before fill
6	1	16	imt,jmt,kmt	3-d tip clearance block
7	1	17	imi,jmi,kmi	3-d inlet block
8	1	18	nbs,npb,1	3-d MERIDL blade *

* idbg(8)=1 gives a MERIDL blade file without
modification. The file has two grids that must be read
with the /mgrid option in plot3d.

- OTHER EXAMPLES

ROTOR 67
185 x 40 x 49 GRID
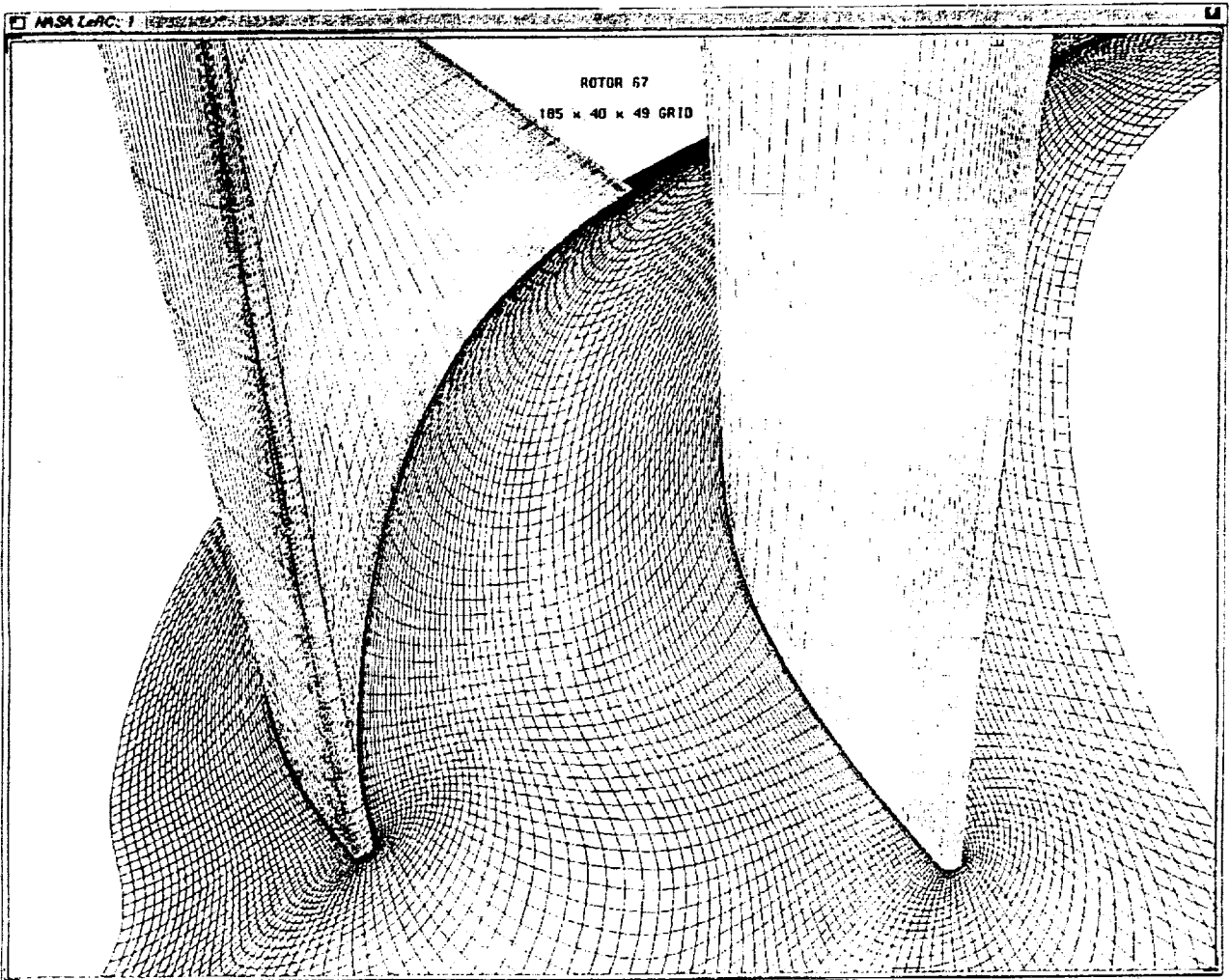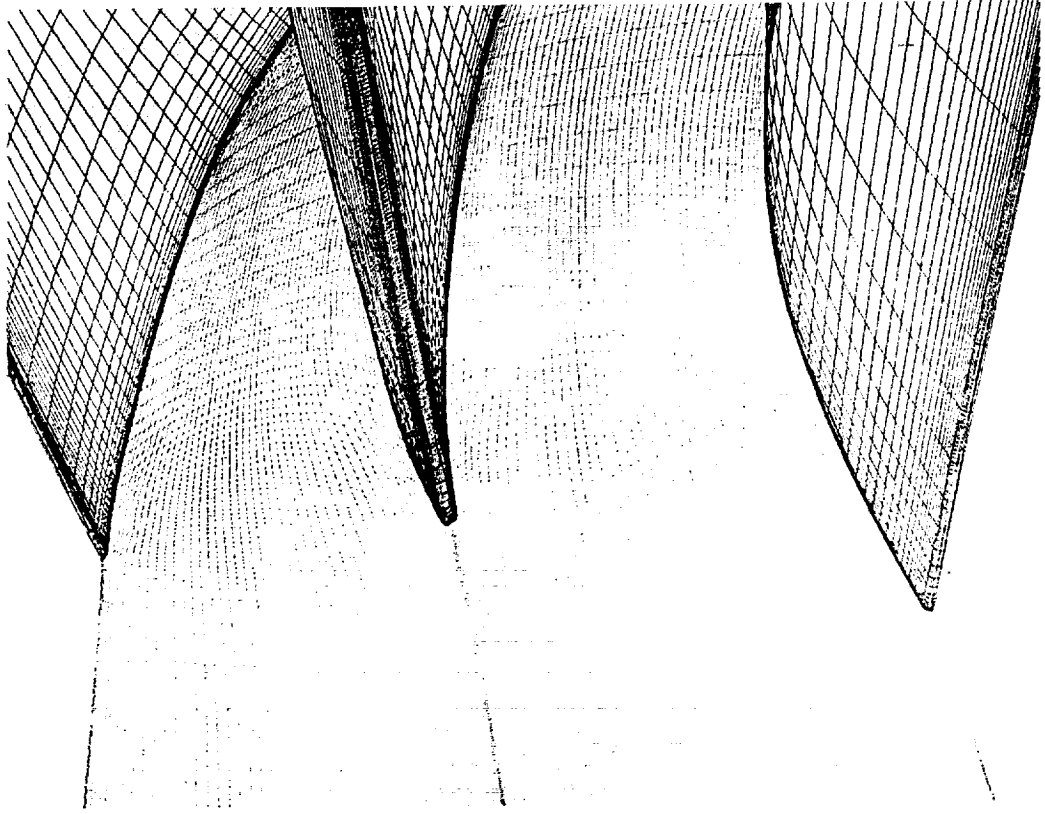
GOLDMANS ANNULAR CASCADE

CONVERGING HUB

97 × 31 × 33 GRID
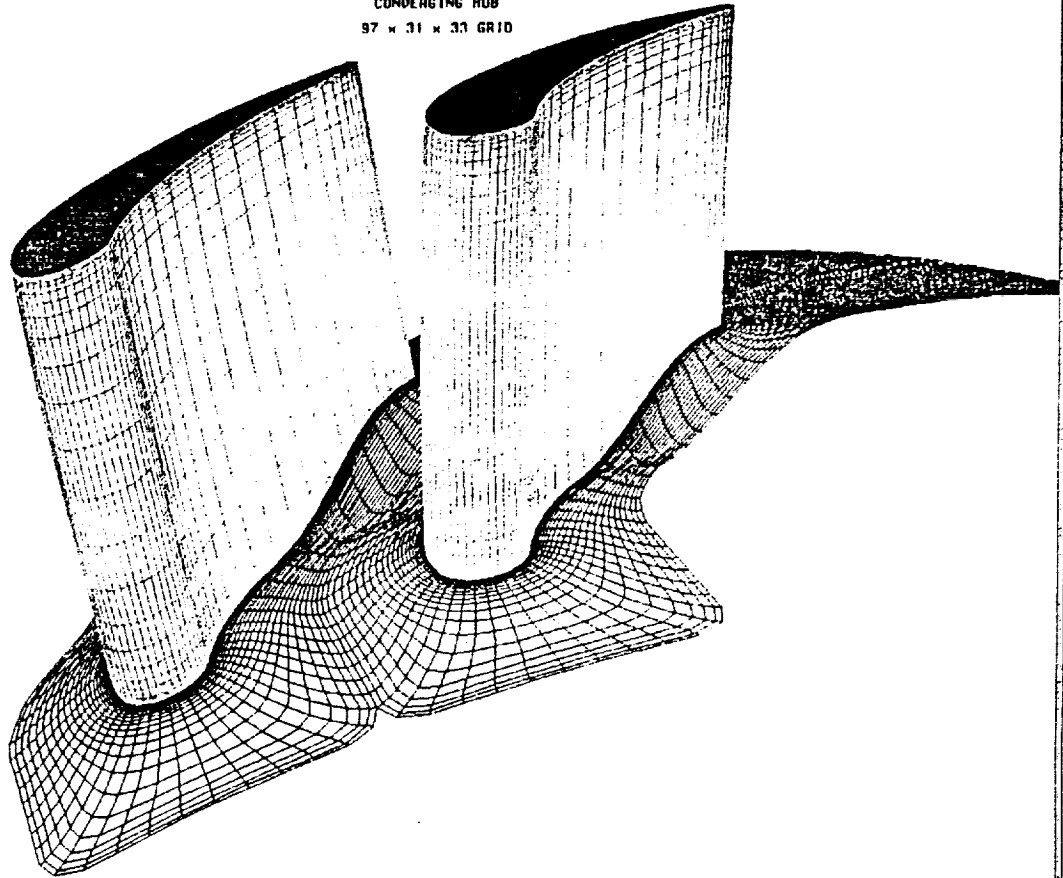
LARGE LOW SPEED CENTRIFUGAL IMPELLER
225 × 39 × 33 C-GRID
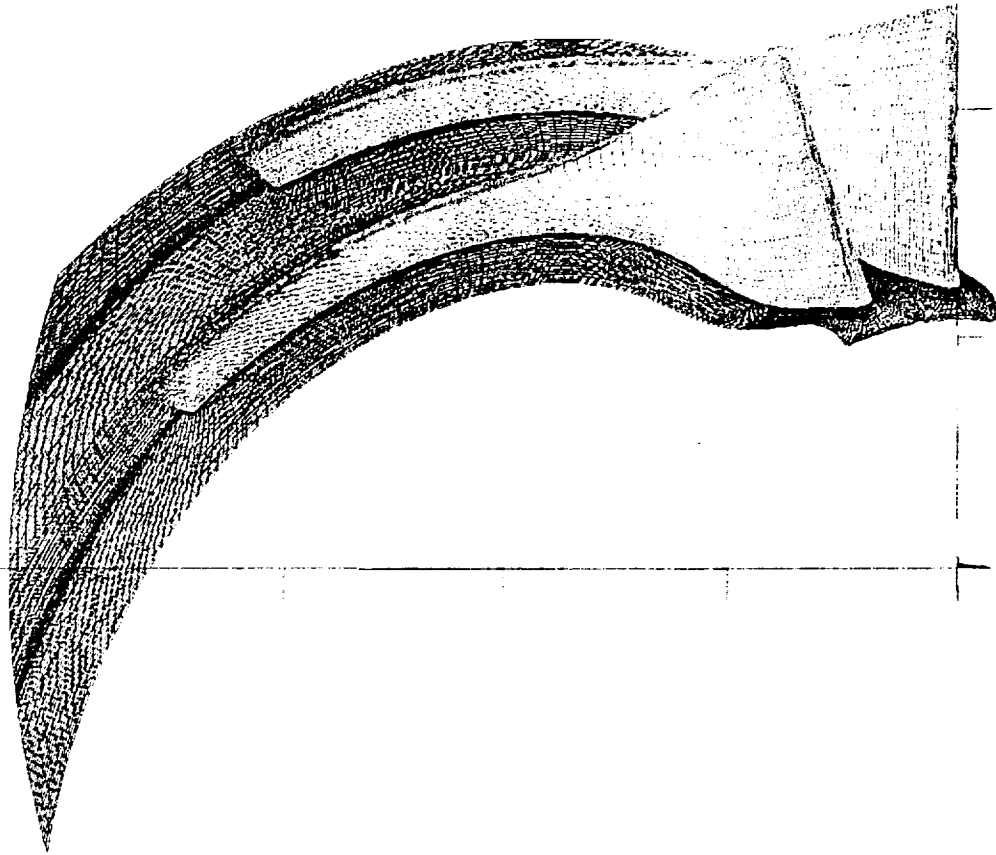
TCGRID - SUMMARY

- TCGRID IS FAST, CLEAN, EASY, GENERAL, ETC. ETC...

- TCGRID IS DOCUMENTED

- TCGRID IS AVAILABLE

An Interactive Grid Generation Technique
for Turbomachinery
by
Tim Beach
Sverdrup Technology, Inc.
Lewis Research Center Group

ABSTRACT

A combination algebraic/elliptic technique is presented for the
generation of 3-D grids about turbomachinery bladerows for both axial and
radial flow machinery. The technique is built around use of an advanced
engineering workstation to construct several 2-D grids interactivley on
predetermined blade-to-blade surfaces. A 3-D grid is generated by
interpolating these surface grids onto an axisymmetric grid. On each blade-to-
blade surface, a grid is created using algebraic techniques near the
blade to control orthogonality within the boundary layer region and elliptic
techniques in the mid-passage to achieve smoothness. The interactive
definition of bezier curves as internal boundaries is the key to simple
construction. The approach is adapted for use with the average passage solution
technique, although this is not a limitation for most other uses. A variety
of examples will be presented.
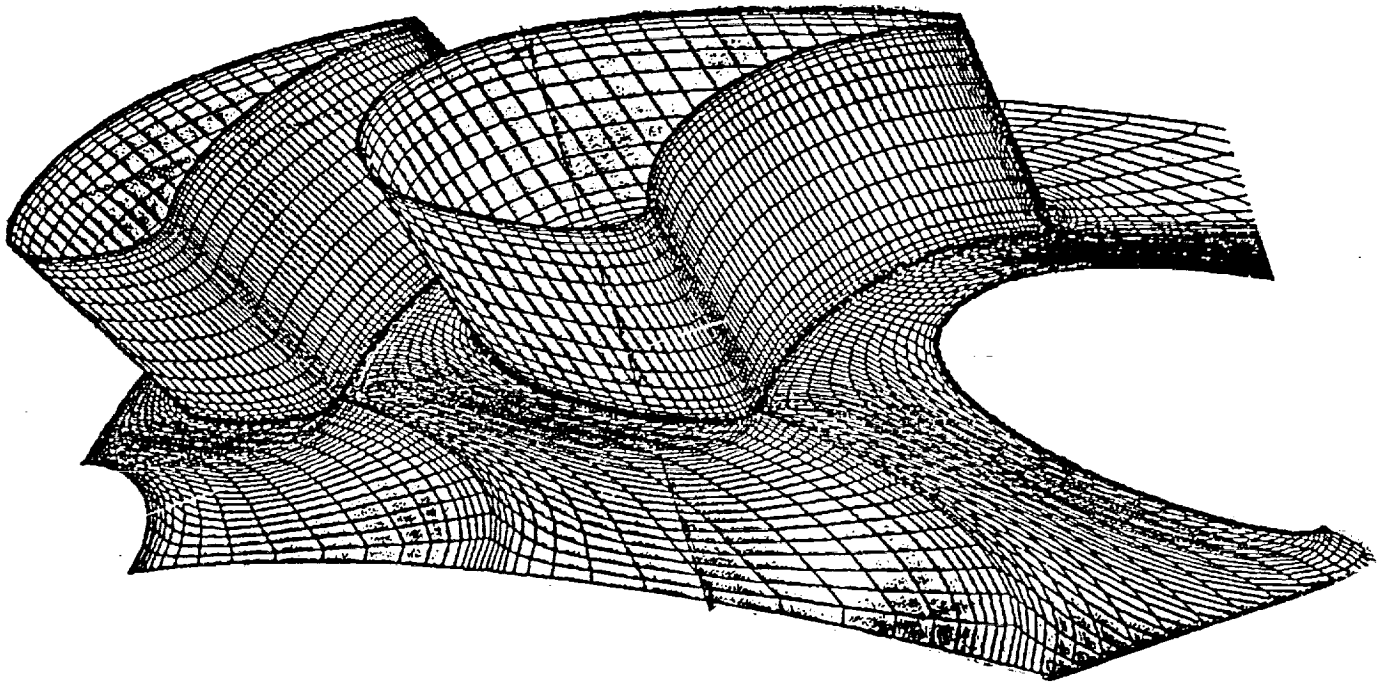
TIGER: Turbomachinery Interactive Grid
GenERation
presented by
Dennis L. Huff*
NASA Lewis Research Center

ABSTRACT

A three-dimensional, interactive grid generation code called "TIGER" is being developed for analysis of flows around ducted or unducted propellers. TIGER is a customized grid generator that combines new technology with methods from general grid generation codes (GENIE and EAGLE). The code generates multiple-block, structured grids around multiple blade rows with a hub and shroud for either C-grid or H-grid topologies. The code is intended for use with a Euler/Navier-Stokes solver also being developed by Mississippi State University, but is general enough for use with other flow solvers. Emphasis is being placed on developing both the flow code and the grid generation code simultaneously for ducted propeller geometries to ensure that the grids will be usable for real applications.

TIGER features a Silicon Graphics interactive graphics environment that displays a pop-up window, graphics window and text window. The geometry is read as a discrete set of points with options for several industrial-standard formats and NASA-standard formats. Various splines are available for defining the surface geometries. (Future work is planned that will incorporate standard data formats using IGES entities.) Once the geometries are defined, TIGER calculates intersections and divides the computational space into sub-blocks for better control of the grid distribution. The perimeters of each sub-block can be controlled in terms of spacing and packing parameters. Bezier curves are used to achieve grid line slope continuity and orthogonality. After the volume grid is generated for each sub-block, the global grid is assembled. The user can specify how the global grid is blocked for input into the flow solver.

Grid generation is done either interactively or through a batch mode operation using history files from a previously generated grid. The batch mode operation can be done either with a graphical display of the interactive session or with no graphics so that the code can be run on another computer system. A typical session running on an IRIS 4D/25 with full interactive graphics takes about 4 minutes (wall clock) to generate a grid with approximately 250,000 grid points for a ducted propeller with two blade rows. This run time can be significantly reduced by running on a faster workstation or batch on the CRAY-YMP.

To date, grids have been generated for the SR-7 propfan, ROTOR-67, a NAVY torpedo, the GE F4-A4 propfan, a propfan cruise missile, and the 1.15 Pressure Ratio Fan for both the forward thrust (design) configuration and a simulated reversed thrust configuration that emulates Pratt and Whitney's Advanced Ducted Propeller. Future work will continue streamlining the code for use with Euler and Navier-Stokes flow solvers. Also, a tool kit that improves the treatment of the leading and trailing edges of the blades and ducts will be incorporated based on a package developed at NASA Lewis (by Schumann and Chima).

---

* Written by Bharat K. Soni, Ming-Hsin Shih, and J. Mark Janus
Engineering Research Center, Mississippi State University

Objectives

- Develop a three-dimensional, interactive grid generation code for analysis of flows around ducted or unducted propellers

- Output multiple block, structured grids for use in TURBO (Euler/N-S flow code also being developed at MSU)

Philosophy

- Develop the flow solver and grid generation code simultaneously .

- Customized approach combines features from a number of generalized grid generators, including GENIE and EAGLE

- Generate a good algebraic grid and then do a few iterations of smoothing (if necessary)

Features

- Multiple input/output formats used by NASA and industry

- Interactive graphics (Silicon Graphics workstations) or batch mode

- Adjustable blade setting angles

- Saves journal file for quick replay or adjustments

- Uses Bezier curves for control of grid shape in both the blade-to-blade and radial directions

- Sub-blocking of the domain gives added control of grid in specific regions.

- Leading and trailing edge modeling uses either bezier curves or a package developed at Lewis by Schumann/Chima

- Any combination of C or H-grids is or will be possible around hub, duct, or blades.

Procedure

1 Read blade, hub, and duct coordinates

2 Compute intersections at hub and duct

3 Redistribute surface points based on packing input

4 Define "Ruler" and "Segment" lines through interactive display and adjustment

5 Compute perimeter planes for each sub-block

6 Compute volume grid for each sub-block

7 Smooth (if necessary) and assemble volume grids

8 Decompose grid into sub-blocks for the flow solver
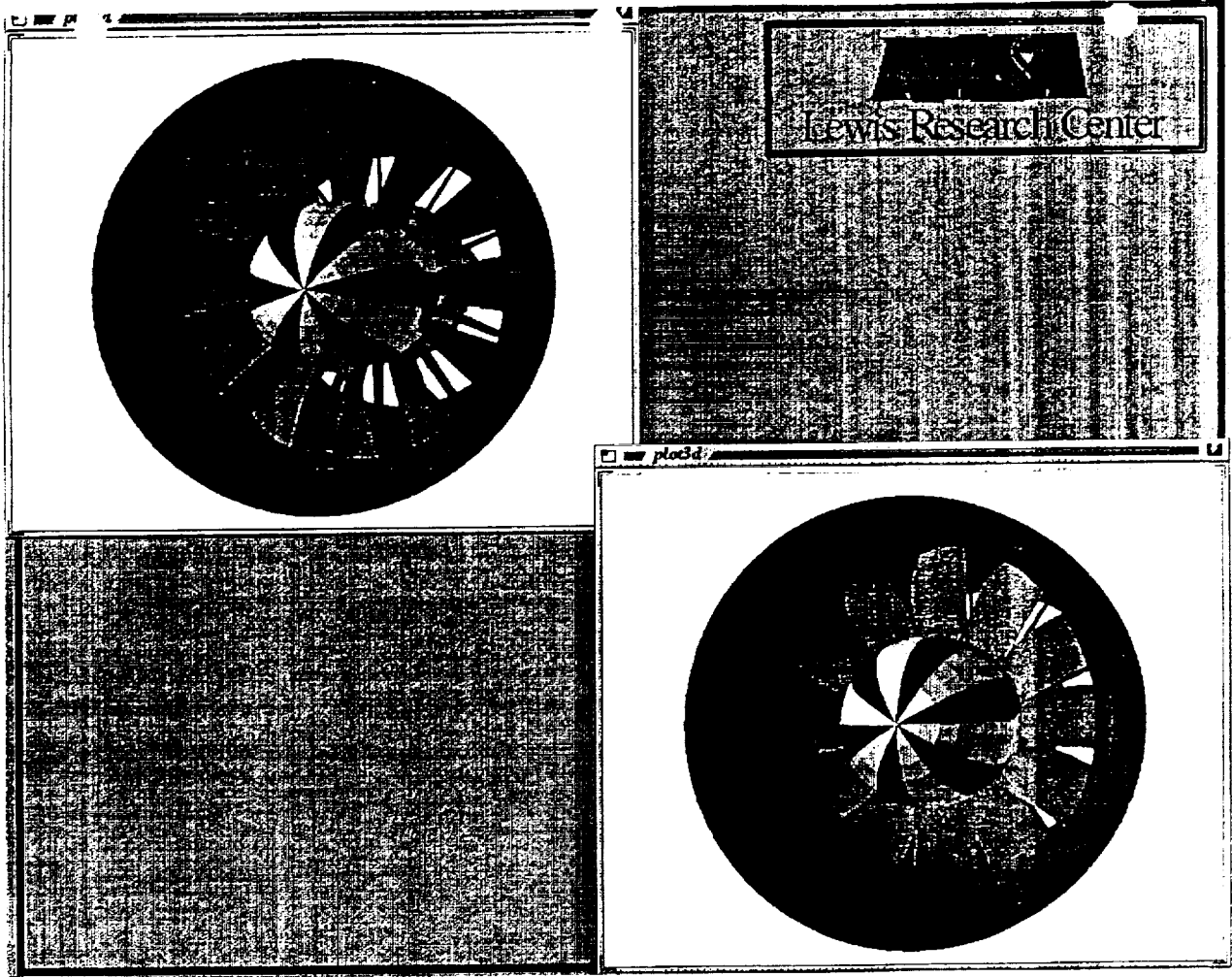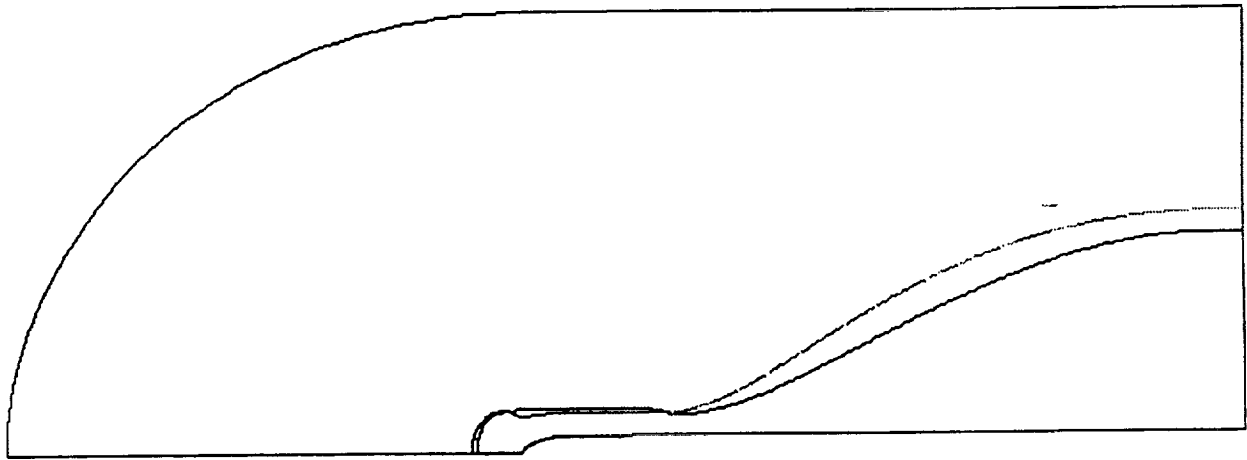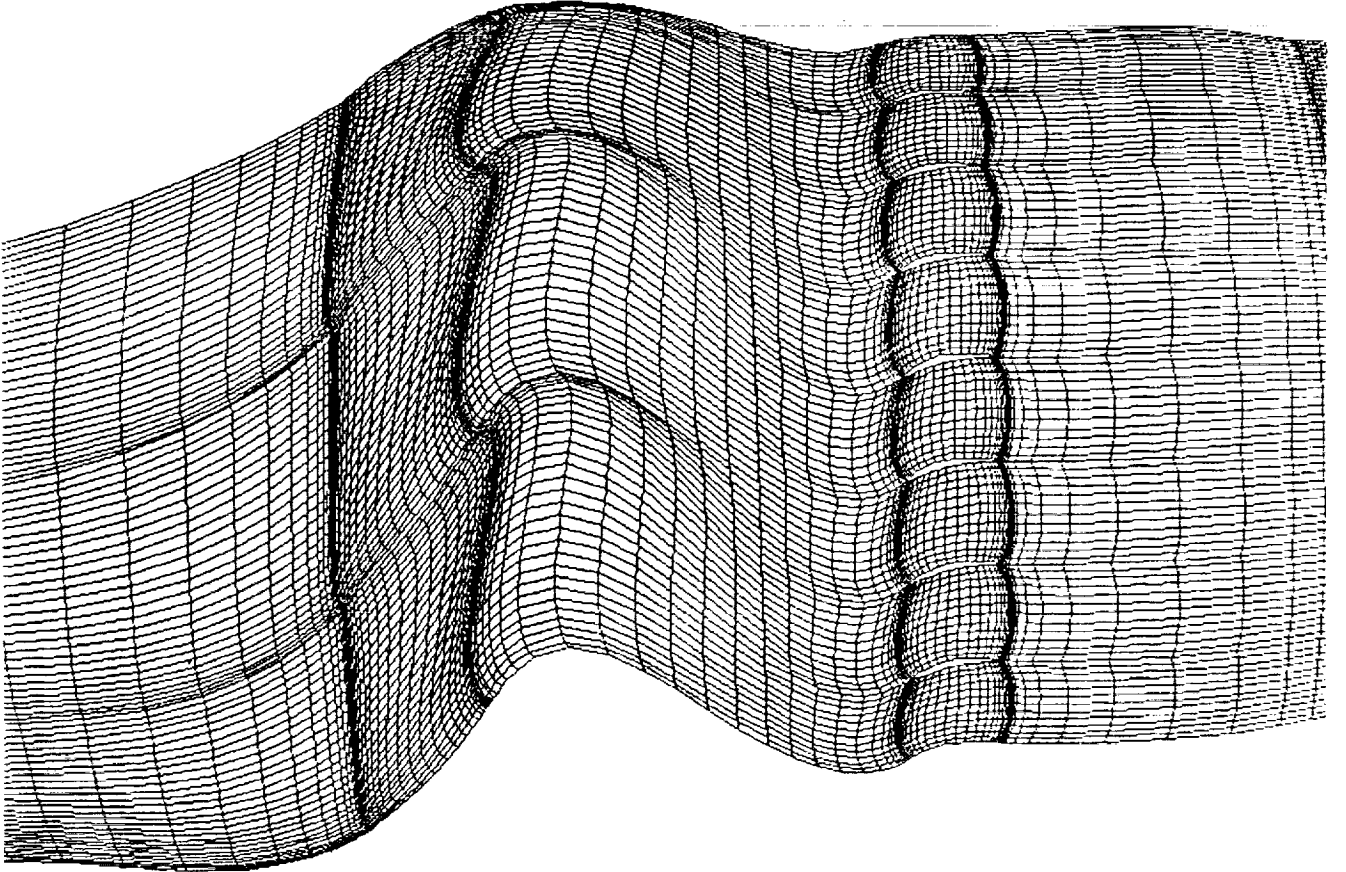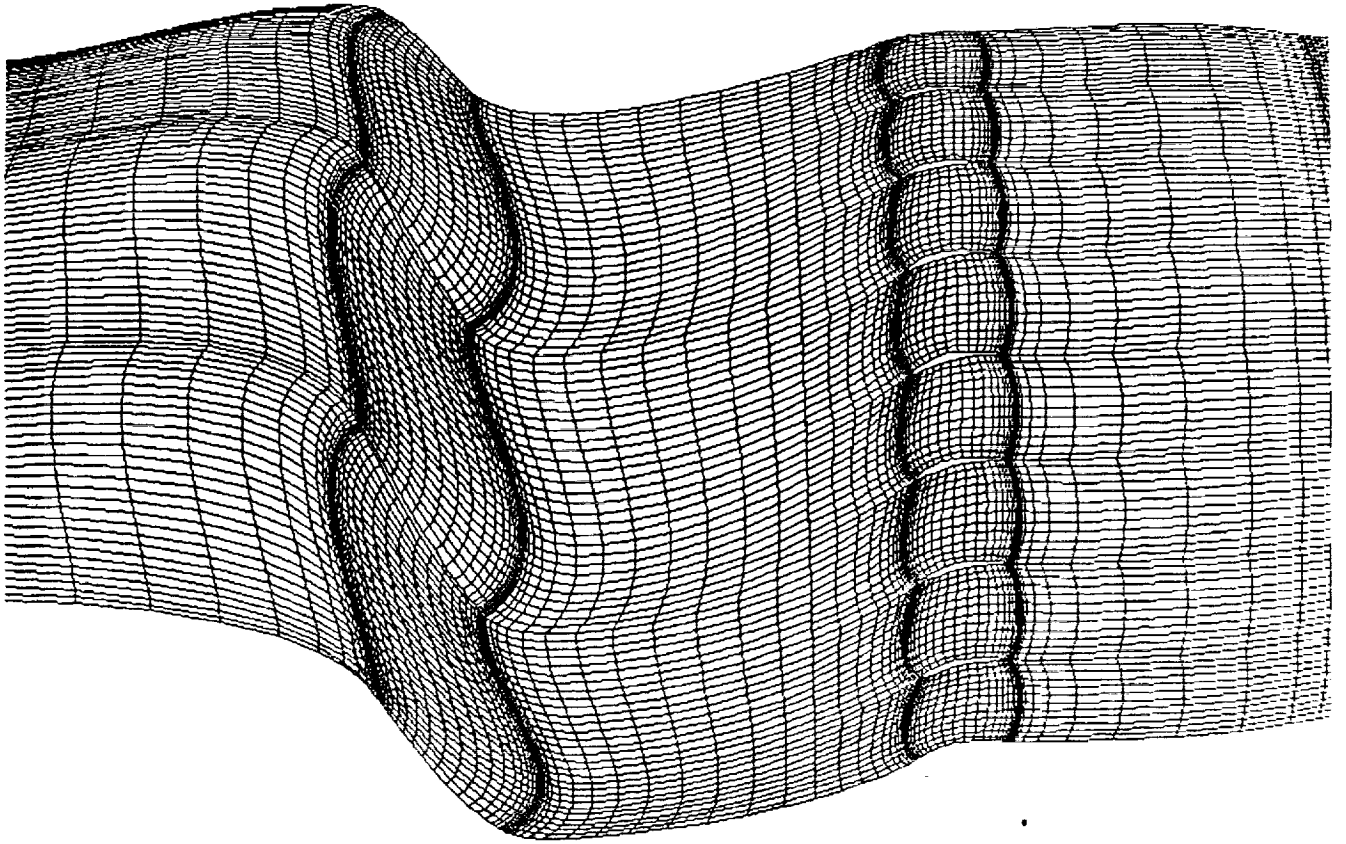
Status and Future Directions

- First generation of TIGER has been installed at Lewis

- Ready for input from people trying to model the flowfield

- Future work will include:
  Further development of leading/trailing edge modeling
  Streamlining code for viscous solutions
  New graphic interface
  Grid quality evaluation
  NASA-IGES interface for geometry input ?
  Any others ?

(Un)Ducted Propeller Grid Generators

CODE NAME	GEOM. TYPE	GRID TYPE	Blade ROWS	MODE	COMMENTS
MESHRM	I/E	H-H	1/2/3..	B	Original mesh generator.
PMESH	I/E	H-H	1/2	B	Better grid smoothness.
MEGEN2	I/E	H-H	1/2	B	Better nose (pusher) packing.
CHGRID	I/E/D	H-H	1/2/3..	B/S	(Blocked) grids for
	D	C-H	1	B/S	the AOA flow solver.
TIGG3D	I/E/D	H-H	1/2/3..	B/I	Multiple radial blocks/flowpaths; NPSS directed.

	Internal	"H-" in Z-R		Batch
Nomenclature:	External	"-H" in Z-Theta		Semi-interactive
	Ducted			Interactive

TIGGERC - Turbomachinery Interactive Grid Generator Energy Distributor and Restart Code

by
David P. Miller
NASA Lewis Research Center

Abstract

A two-dimensional multi-block grid generator has been developed for a new design and analysis system for studying multi-blade-row turbomachinery problems with an axisymmetric viscous/inviscid "average-passage" throughflow code. TIGGERC is a mouse driven, fully interactive grid generation program which can be used to modify boundary coordinates and grid packing. TIGGERC generates grids using a hyperbolic tangent or algebraic distribution of grid points on the block boundaries and the interior points of each block grid are distributed using a transfinite interpolation approach. TIGGERC generates a blocked axisymmetric H-grid , C-grid, I-grid or O-grid for studying turbomachinery flow problems. TIGGERC was developed for operation on small high speed graphic workstations.

Flow Chart for NASA LERC Turbomachinery Design/Analysis System

TIGGERC

T urbomachinery
I nteractive
G rid
G enerator
E nergy
R estart
C ode

IBEGG

I nteractive
B lade
E lement
G ometry
G enerator

Interface Block

Axisymmetric Streamline Curvature Solution

VIADAC

V iscous
I nviscid
A xisymmetric
D esign
A nalysis
C ode

Quasi-3D Blade-to-Blade Solution

Full-3D Solution
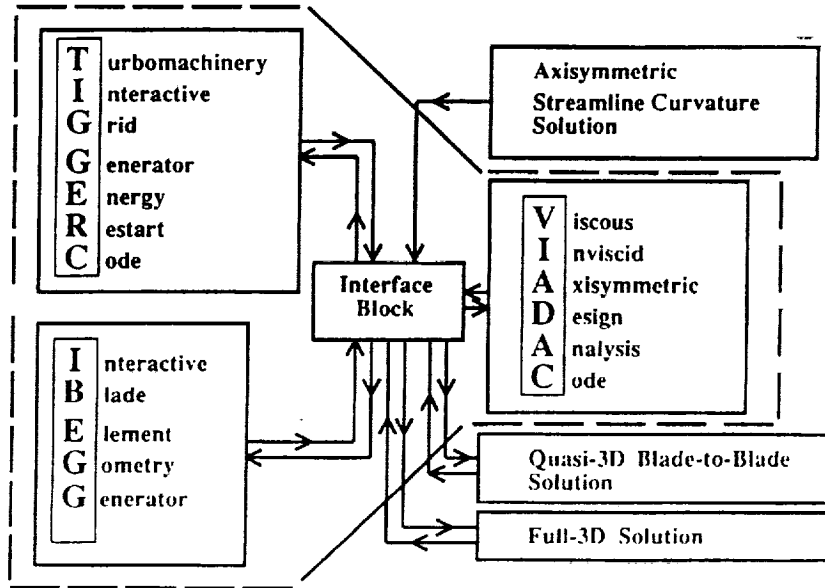
TIGGERC

Turbomachinery I nteractive G rid G enerator E nergy Distributor   R estart   C ode

- Create an Axisymmetric Blocked Grid for any Generic Turbomachinery Problem

- Interactively Generate a Viscous/Inviscid Grid

- Interactively Modify Coordinates

  - ☐ Bezier Curves
  - ☐ Damped Cubic Splines
  - ☐ Mouse Driven & Manual Input

- Algebraic & Hyperbolic Tangent Block Boundary Point Distributions

- Distribute Work or Turning on a new Blade

- Creates Generalized File Format for Iris or Cray

TIGGERC BLOCK STRUCTURE
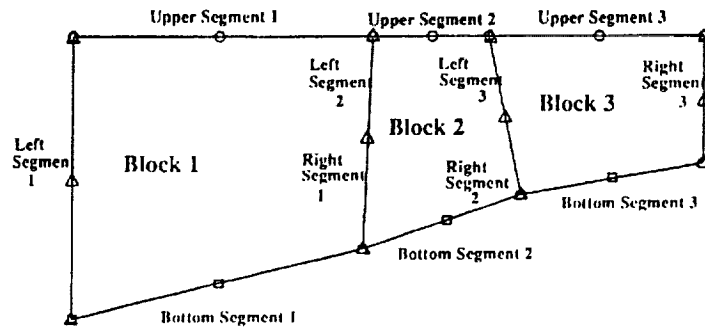
Examples of TIGGERC Block Boundaries

A Small Axial-Centrifugal Simulated Engine
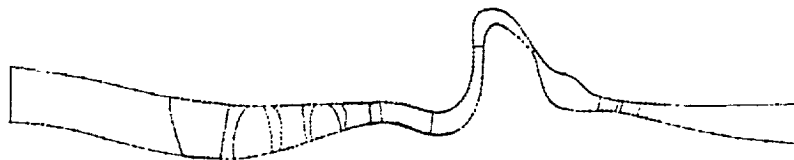
NASA Rotor67 Geometry

NASA Lewis Research Center
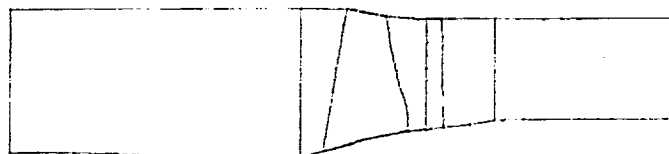
Examples of Tiggerc H-Grid Mesh Generation

A Small Axial-Centrifugal Simulated Engine

442x31 Grid
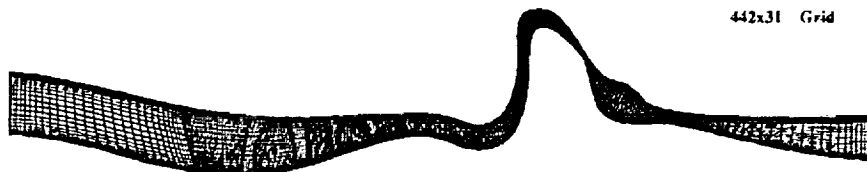
NASA Rotor67 Test Bed Grid

162x61 Grid

NASA Lewis Research Center
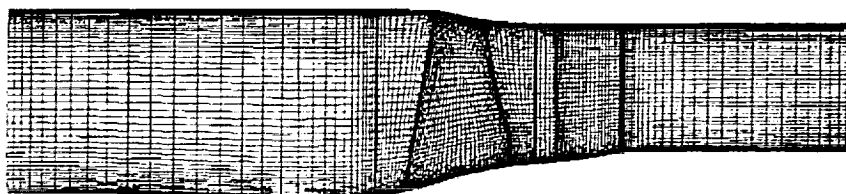
Examples of O-Grid, I-Grid and C-Grid Geometries

O-Grid

I-Grid
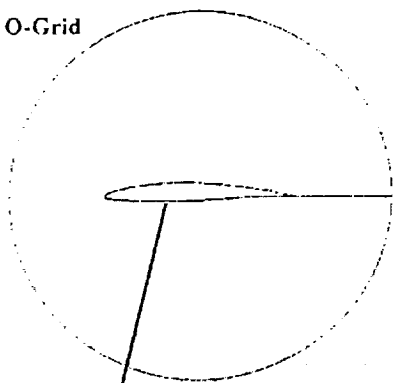
Simulated Turbine Vane
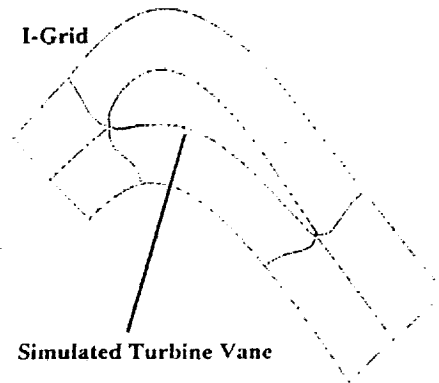
NACA 64-410
Series Airfoil
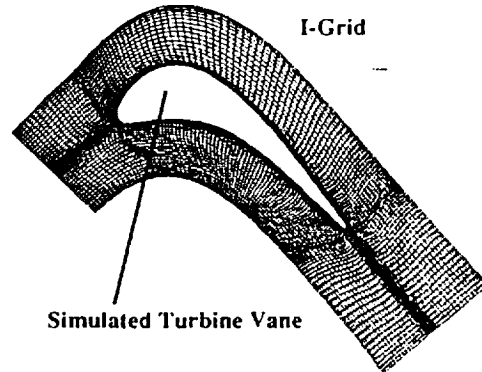
C-Grid

Examples of O-Grid, I-Grid and C-Grid Meshes

O-Grid

I-Grid

Simulated Turbine Vane

NACA 64-410   C-Grid
Series Airfoil
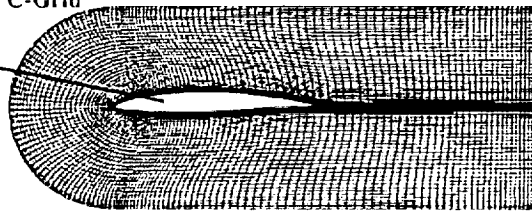
Examples of TIGGERC Multi-Row-Block Boundaries

An Advance Ducted Propfan Geometry
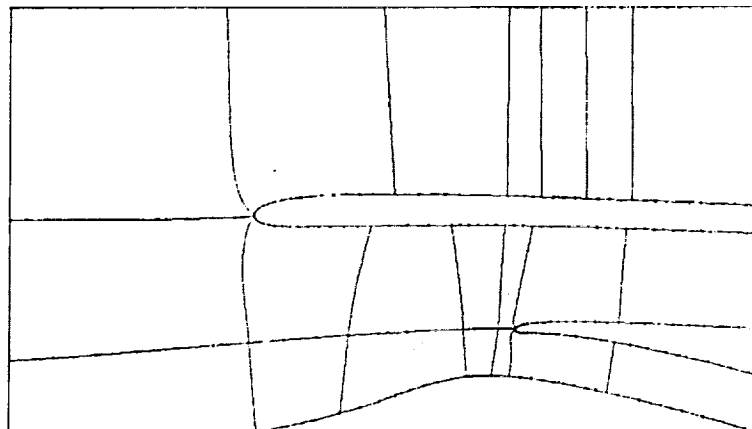
NASA Lewis Research Center

Examples of TIGGERC Multi-Row-Block Boundaries

An Advance Ducted Propfan Grid

NASA Lewis Research Center
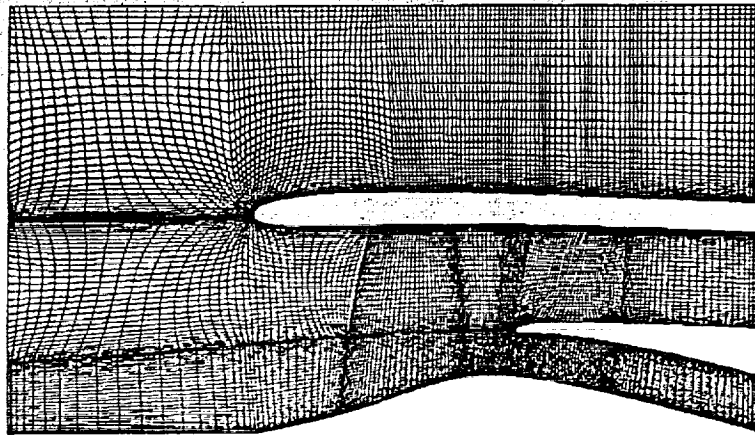
Other Examples of TIGGERC Grids

An Axisymmetric Rocket Nozzle
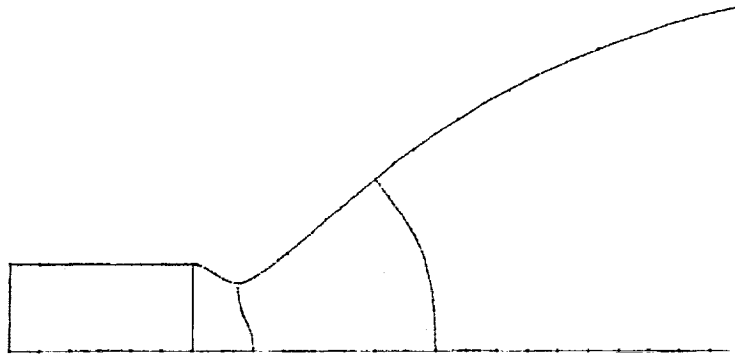
Other Examples of TIGGERC Grids

An Axisymmetric Rocket Nozzle
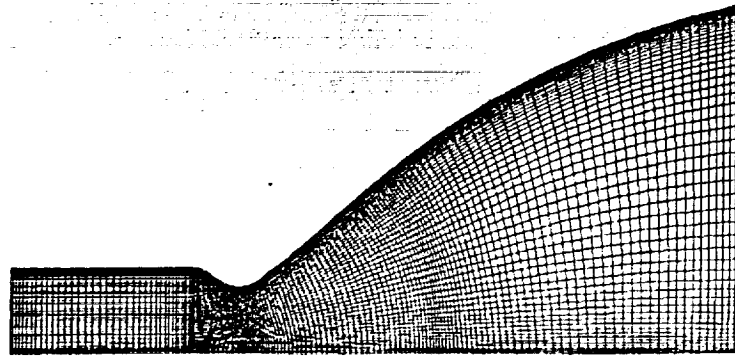
TIGG3D Grid Generation About the NASA/GE Energy Efficient Engine Fan Section

71
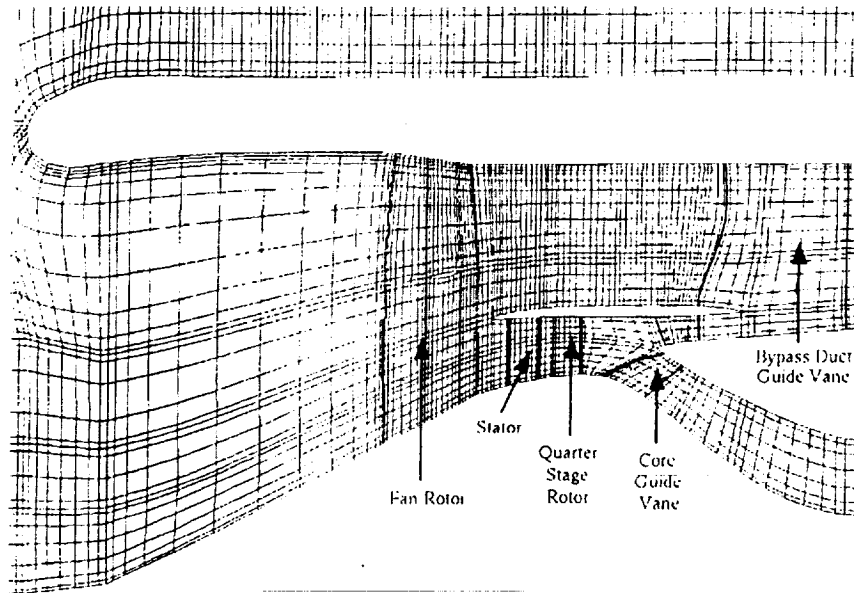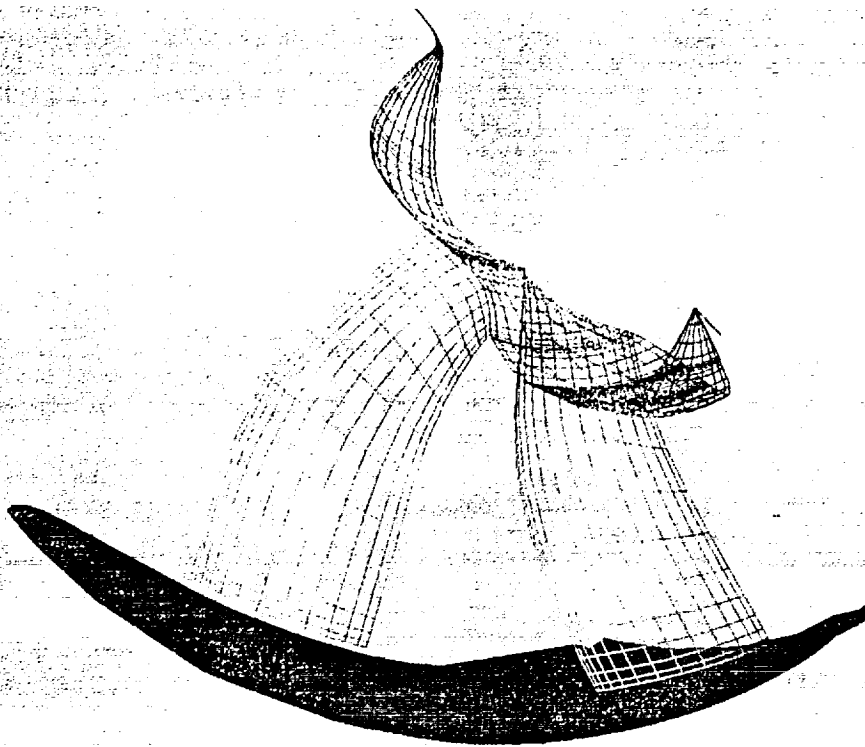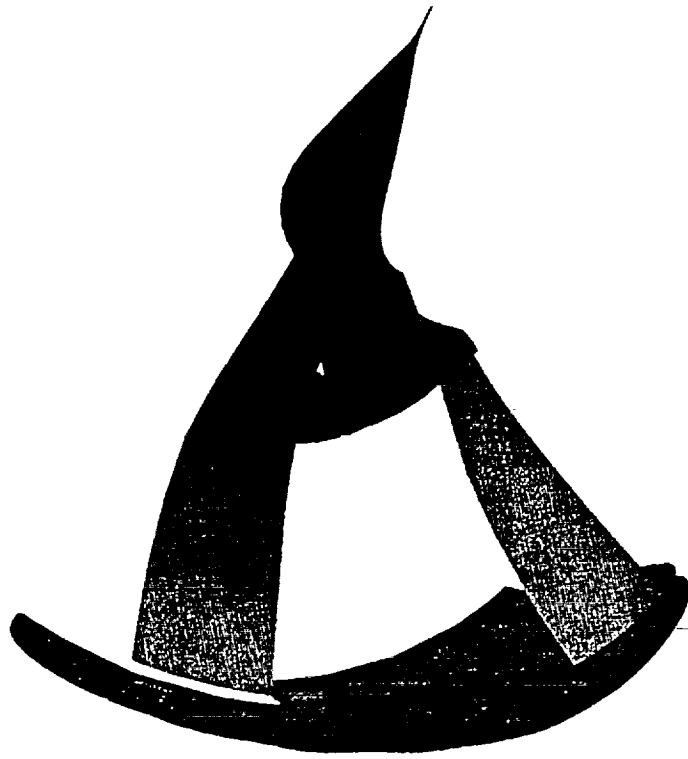
Summary of Discussion
Session II
Chairs: Dennis L. Huff and David P. Miller

A noticeable trend for grid generation in turbomachinery is the use of customized codes. All four papers presented in this session had some level of customizing. The general feeling from the audience was that the user should use whichever code works for a given problem. The maturity of CFD is not as advanced as it is for finite element analysis and there are many questions that depend on the flow physics being modeled. Since the desired topology for turbomachinery applications is generally known, the amount of time required to generate a grid is significantly less than the time required to grid an entire aircraft. Lewis has been mainly interested in analyzing components of turbomachinery, which greatly simplifies the grid generation process. However, this may change with system integration and multidisciplinary applications. There was some concern expressed by people interested in system integration that the use of customized grid codes will present them with a problem.

The four papers in this session used structured grids with no adaptation. There was some concern from the audience that other NASA centers do more research in this area and that adaptation is the only realistic way to address three-dimensional problems. At this point, it was mentioned that there was some level of research being done at Lewis for unstructured meshes, which is a natural home for adaptation. There was no discussion of whether adaptation should be done for structured grids in turbomachinery.

Another concern addressed in this session was interactive versus batch codes. The general concern for batch codes is that time must be spent looking at the grid anyway, and interactive grid generation is much faster unless the user is regridding from a previously generated grid.

SESSION III

NEW TECHNIQUES IN GRID GENERATION

Chair: Mark E.M. Stewart

Multiblock Grid Generation for

Jet Engine Configurations
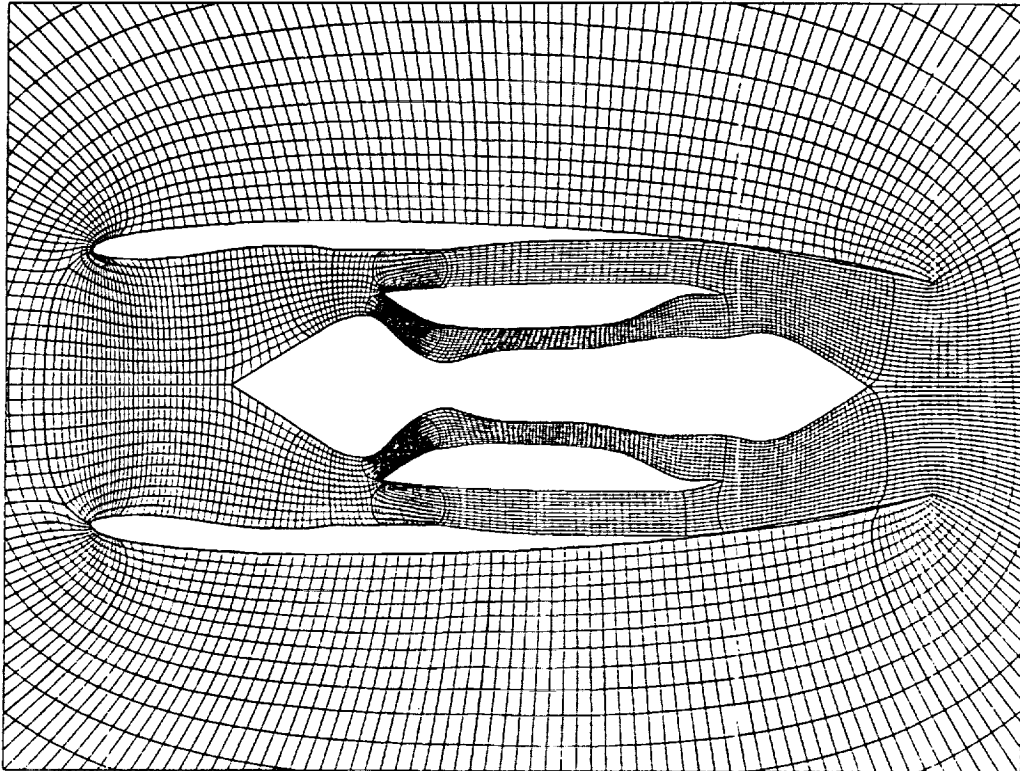
Mark E.M. Stewart
Institute for Computational Mechanics in Propulsion
NASA Lewis Research Center

Overview

- Goal

- Multiblock Grid Generation

  - Finding a Domain Decomposition

  - Dimensioning Grids

  - Grid Smoothing

  - Manipulating Grids and Decompositions

  - Some Specializations for Jet Engine Configurations

Goal

Methods for Generating Grids with,

- Minimal Human Intervention

- Applicable to a Wide Range of Problems

- Compatible with Existing Numerical Methods

- Compatible with Existing and Proposed Computers

Domain Decomposition

- Search Algorithm for Determining Boundary Conforming Subregions

- Algorithm is,

    - Efficient

    - General

    - Substantially Automates Domain Decomposition

Segment Refinement Example

81
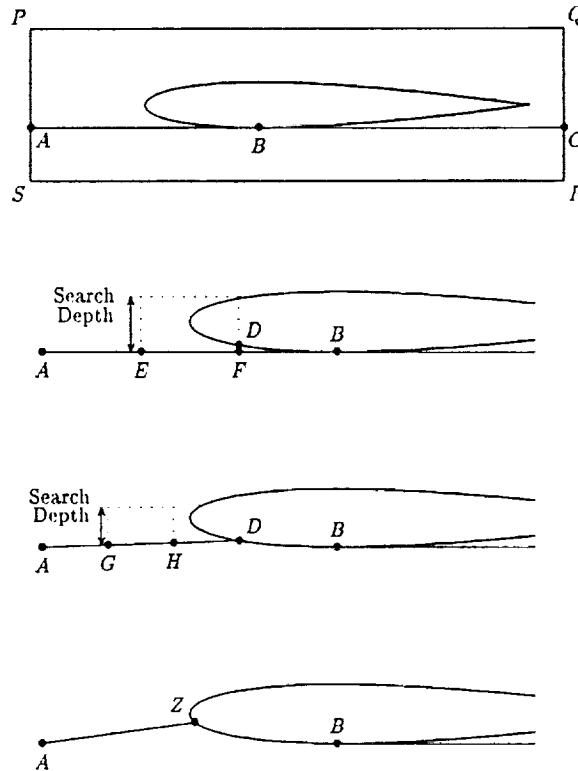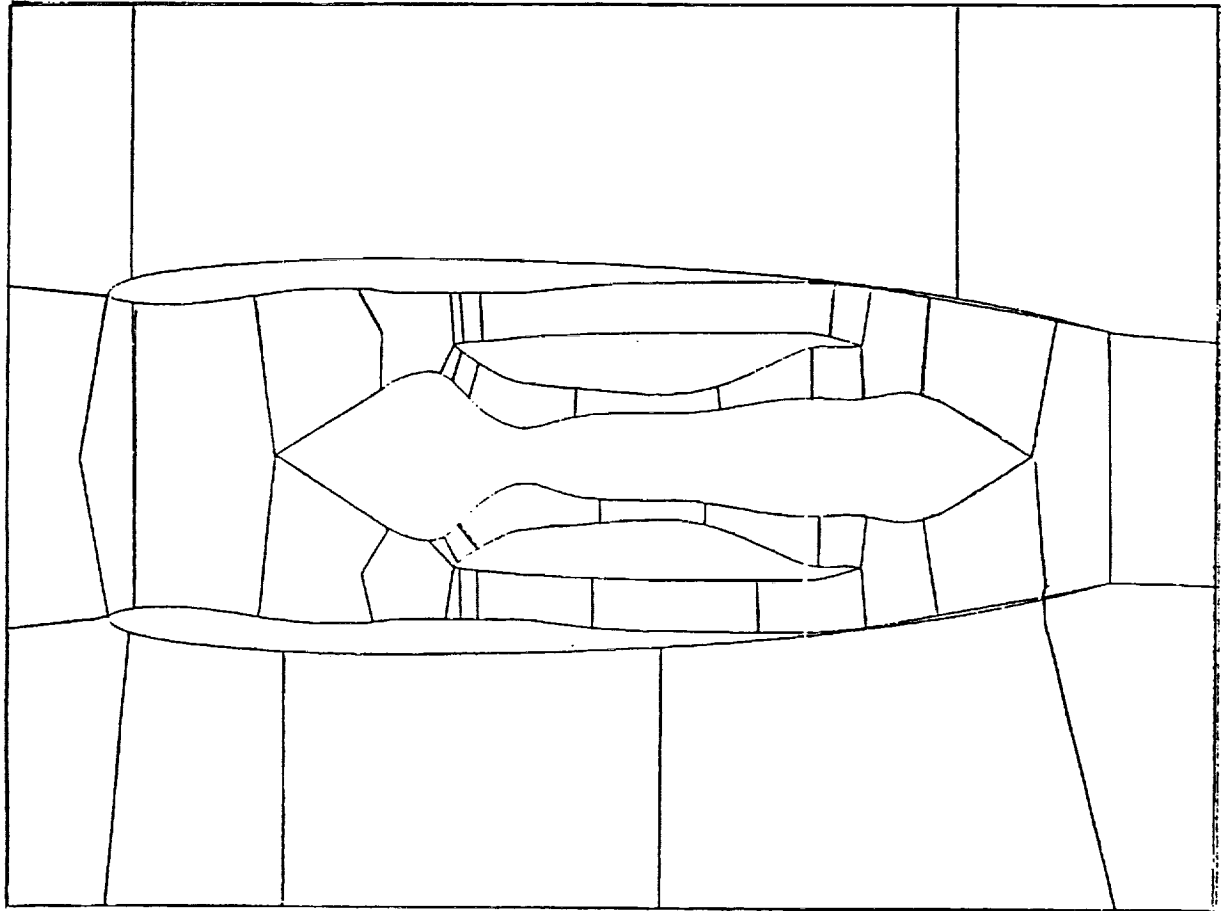
Dimensioning Grids in a Domain Decomposition

- Domain Decomposition is Globally Unstructured/Locally Structured (No background structured grid)

- Constraints:
    - Structured Grid
    - Coordinate Line Continuity at Interfaces

- Constraints Form a System of Linear Equations
    - underconstrained
    - constant, integer coefficients
    - positive, integer solutions

- Simplex Method for finding a solution

- Since Underconstrained, Solution Parameters Allow Flexibility in Grid Dimensions

Grid Generation

- Algebraic Methods (Coons' Patch) to Initialize the Grid

- Elliptic Grid Smoothing
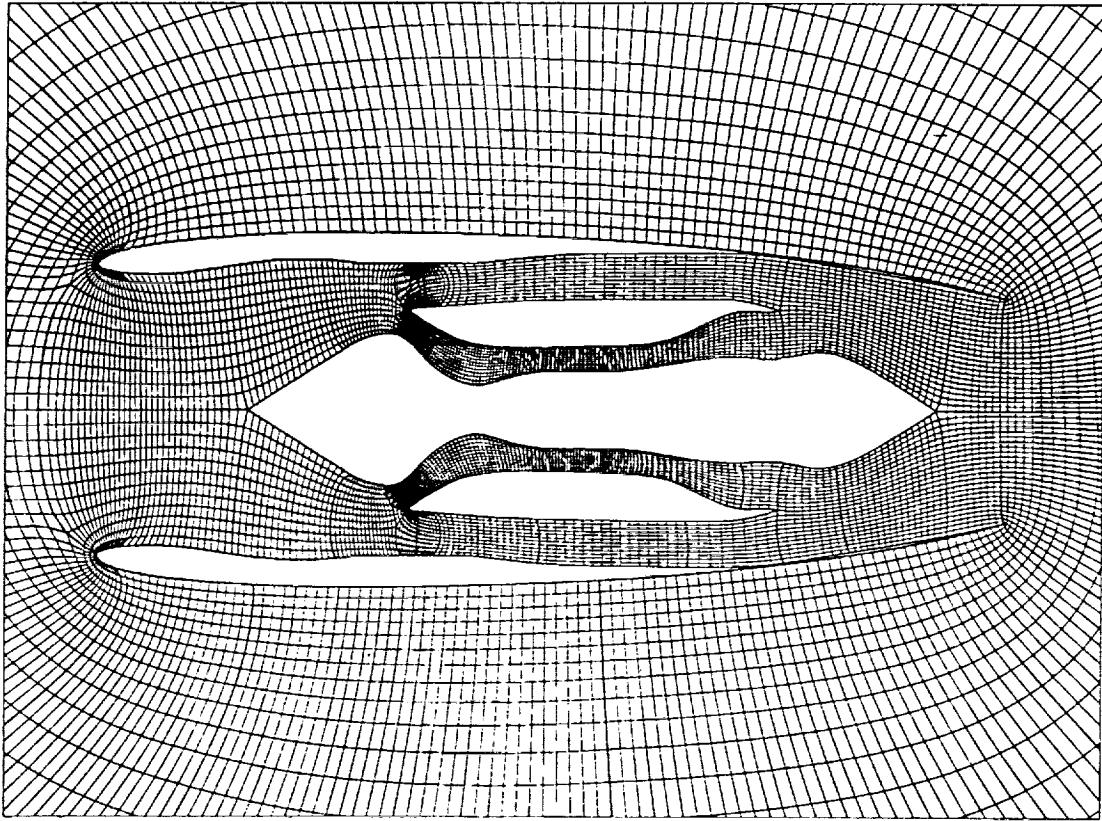
Conclusions

- Multiblock Grid Generation Techniques

    - Finding a Domain Decomposition

    - Dimensioning Grids

    - Grid Smoothing

    - Manipulating Grids and Decompositions

Two-Dimensional Unstructured Triangular Grid Generation

by
Philip C.E. Jorgenson
NASA Lewis Research Center

Abstract

The development of a general flow code that can predict the flow about complex geometries which include complex flow structures is a desirable feature in computational fluid dynamics. Many flow codes to date which use the finite difference or finite volume formulation of the flow equations were written with some basic inherent structure which permits flow solutions to be obtained efficiently. This structure, which makes the solver so efficient, often makes it difficult to obtain reasonable grids about complex flow geometries. Grid patching or grid overlaying techniques have been used with structured grids; however, the use of these techniques usually requires special coding in the flow solver to circumvent problems like metric discontinuities and function interpolations between the various grids. An unstructured grid flow solver can alleviate many of the problems associated with structured grids. One advantage of using an unstructured triangular grid formulation is the ability to generate grids about arbitrary geometries. Another advantage is the ability to add cells in high gradient regions of the flow field as well as those regions of the flow that are of interest without concern for the surrounding cells. The main disadvantage of using an unstructured mesh lies in the added complexity of the flow solver.

The current work will demonstrate a capability for the generation of two-dimensional unstructured triangular meshes about arbitrary geometries. This work uses a distribution of boundary points and triangulates the computational domain using a Delaunay triangulation algorithm. Typically initial cells are added based on cell aspect ratios or cell areas. A resulting mesh can then be used along with the connectivity of the cells to solve either an Euler or Navier-Stokes flow problem.

Motivation: Develop a triangular unstructured grid generation package that outputs the necessary connectivity and geometric information for an Euler or Navier-Stokes flow code.

Outline:

- Structured vs Unstructured

- Details of triangular unstructured grid generation

- Two-dimensional examples

- Current status

- Future plans

Structured vs Unstructured
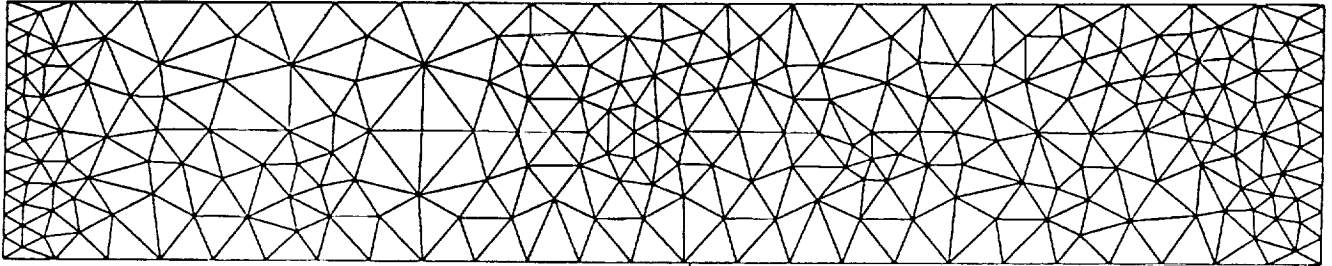
Structured vs Unstructured

Structured grid

Advantages

- Easy to grid simple geometries

- Flow solver can exploit the grid structure for increased speed

Disadvantages

- Complex geometries are difficult to grid

- Grid adaptation to flow features must be done within the restrictions of the grid structure

Structured vs Unstructured

Unstructured grid
Advantages

- Easy to grid complex geometries

- Grid adaptation can be done locally

Disadvantage

- A complete data structure is required for the flow code

Unstructured Triangular Grid Generation

Methods:

- Advancing front

- Delaunay triangulation
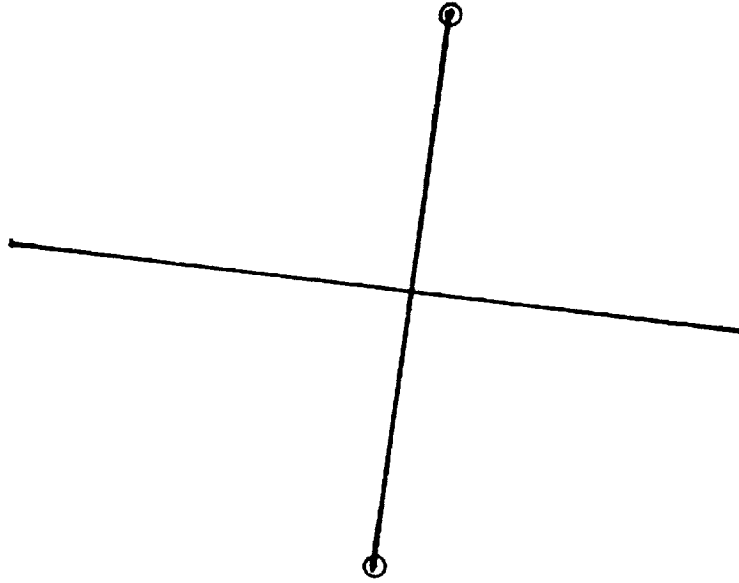
Delaunay Triangulation
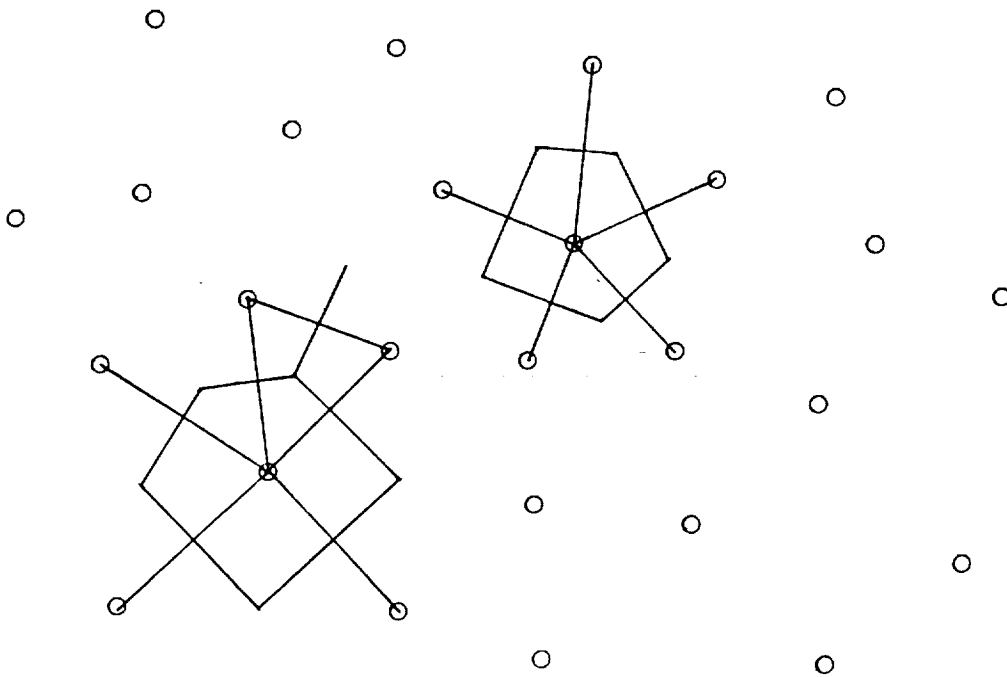
Delaunay Triangulation
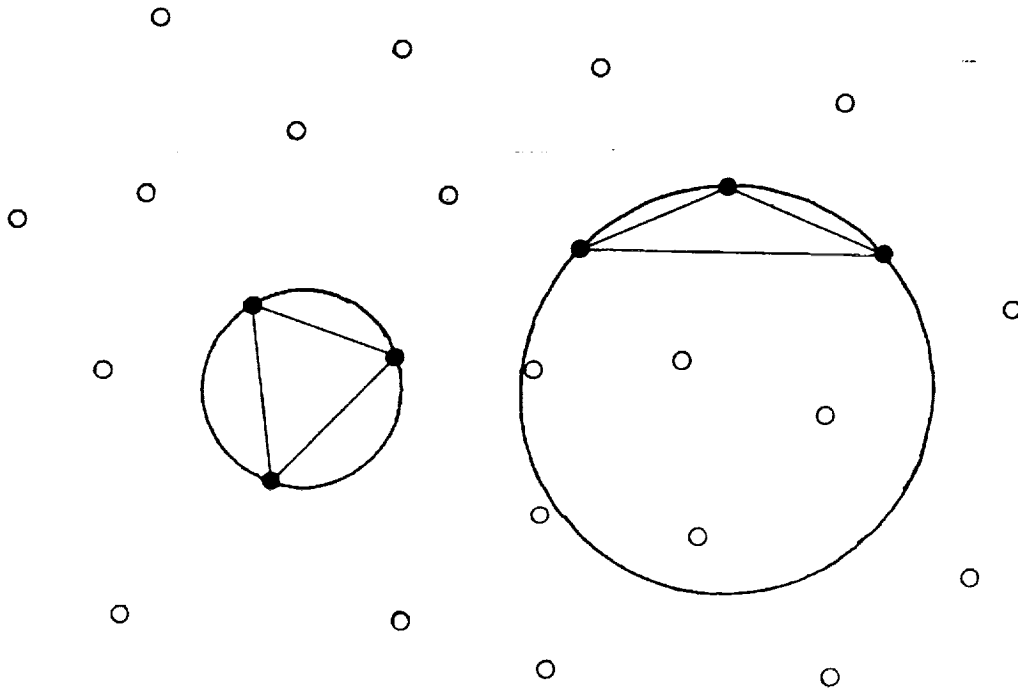
Delaunay Triangulation

- Boundary description

- Triangulate boundary points

- Refine until "good" initial grid

- Output grid geometry and connectivity

- Run flow code and adapt to interesting flow physics

Boundary Description
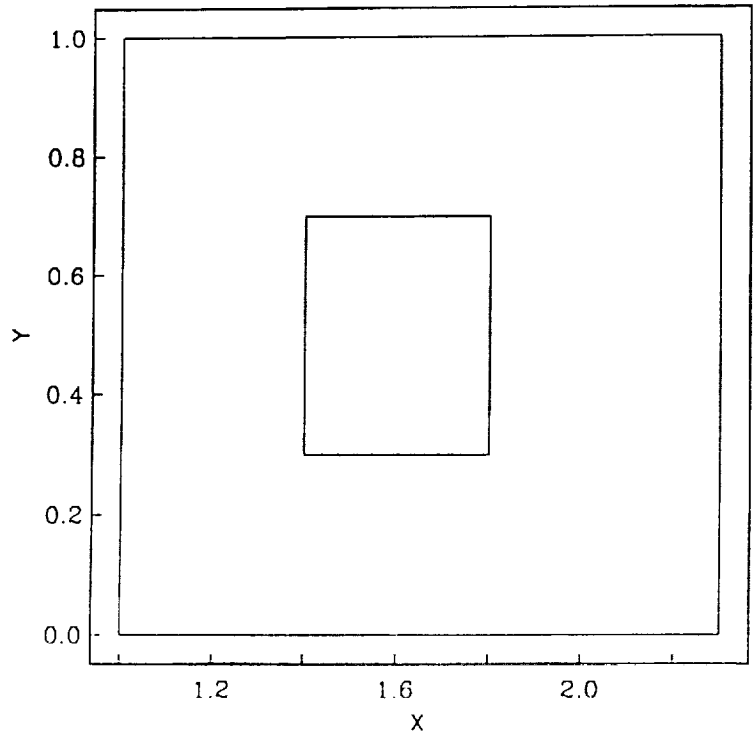
Boundary Points
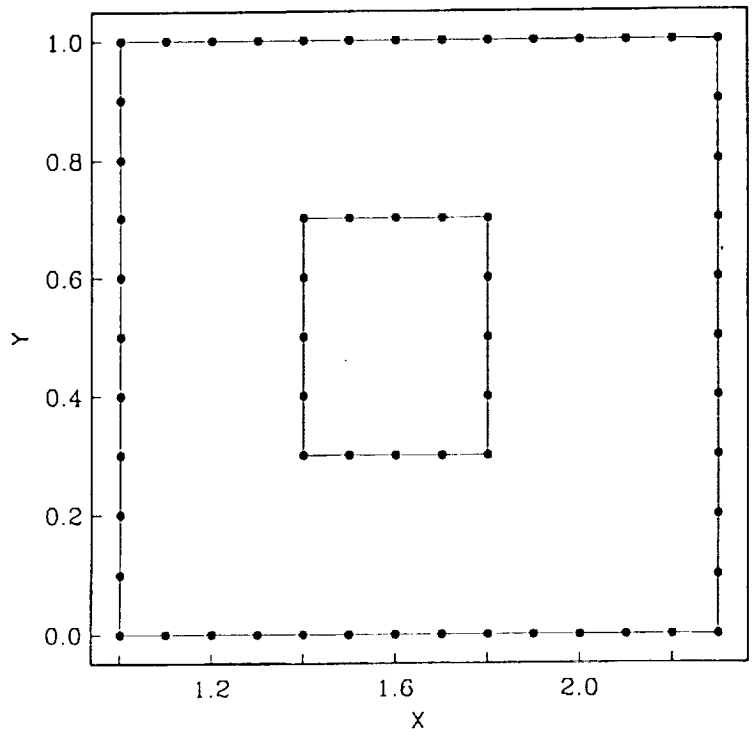
Triangulation of Boundary Points
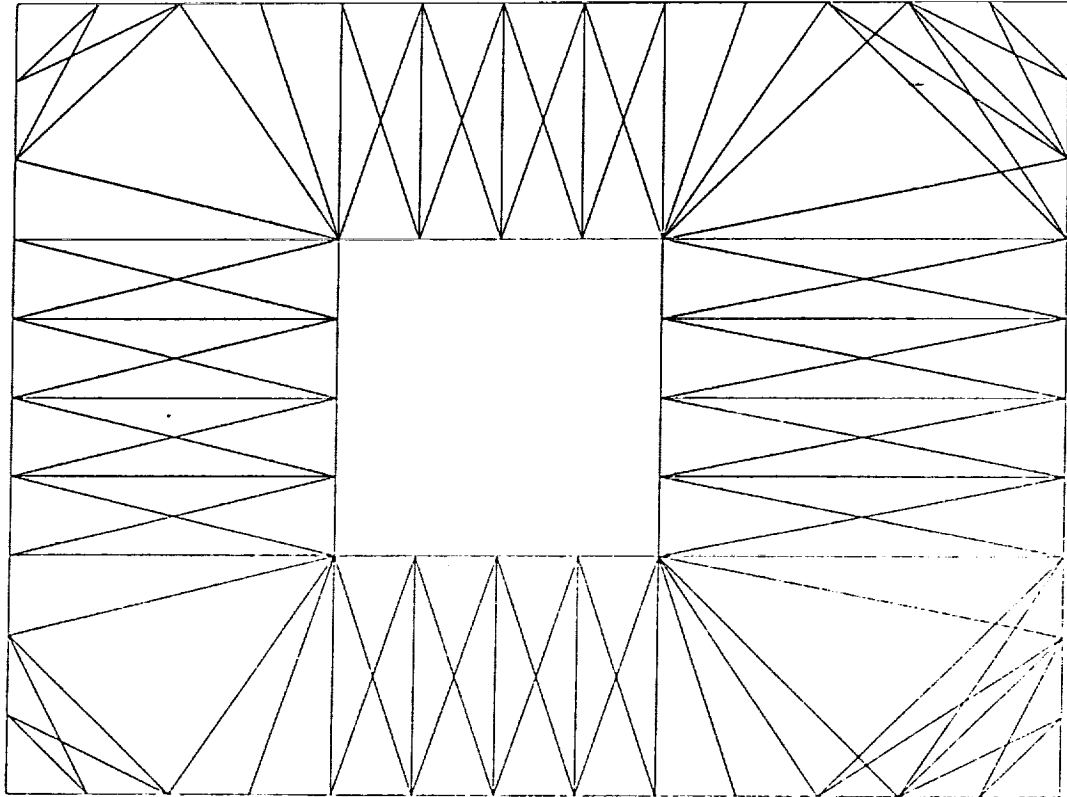
Grid Refinement Criterion

- Aspect ratio

- Area

- Circumcircle

Refinement-Aspect Ratio

Basis:

- Ratio of radius of inscribed circle to twice the radius of circumcircle

$$AR = \frac{ri}{2rc}$$

where

$$ri = \frac{\sqrt{s(s-a)(s-b)(s-c)}}{s}$$

$$rc = \frac{abc}{4\sqrt{s(s-a)(s-b)(s-c)}}$$

and

$$s = \tfrac{1}{2}(a + b + c)$$

Result:

- New point is placed at circumcenter of triangle with smallest aspect ratio

C-2

Refinement-Area

Basis:

- Area of triangle

$$AREA = \sqrt{s(s-a)(s-b)(s-c)}$$

where

$$s = \tfrac{1}{2}(a + b + c)$$

Result:

- New point is placed at circumcenter of triangle with largest area

Refinement-Circumcircle

Basis:

- Radius of circumcircle of triangle

$$RC = \frac{abc}{4\sqrt{s(s-a)(s-b)(s-c)}}$$

where

$$s = \tfrac{1}{2}(a + b + c)$$

Result:

- New point is placed at circumcenter of triangle with largest circumcircle

Add New Point
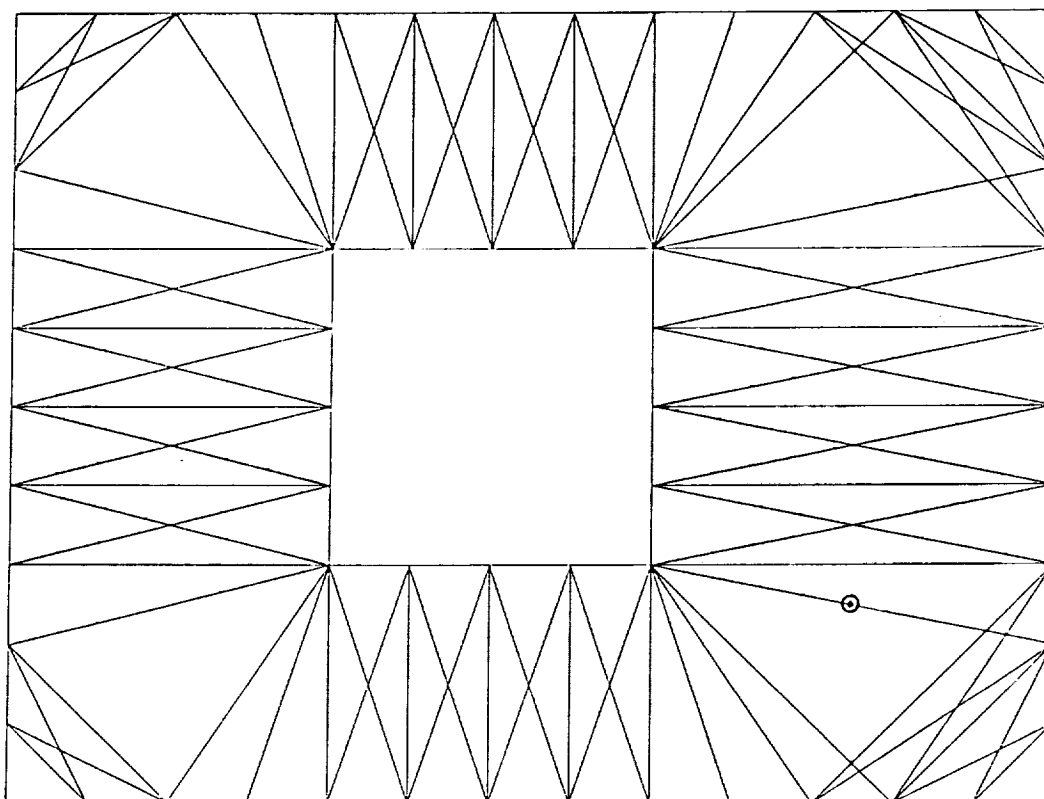
Delete Conflicting Cells
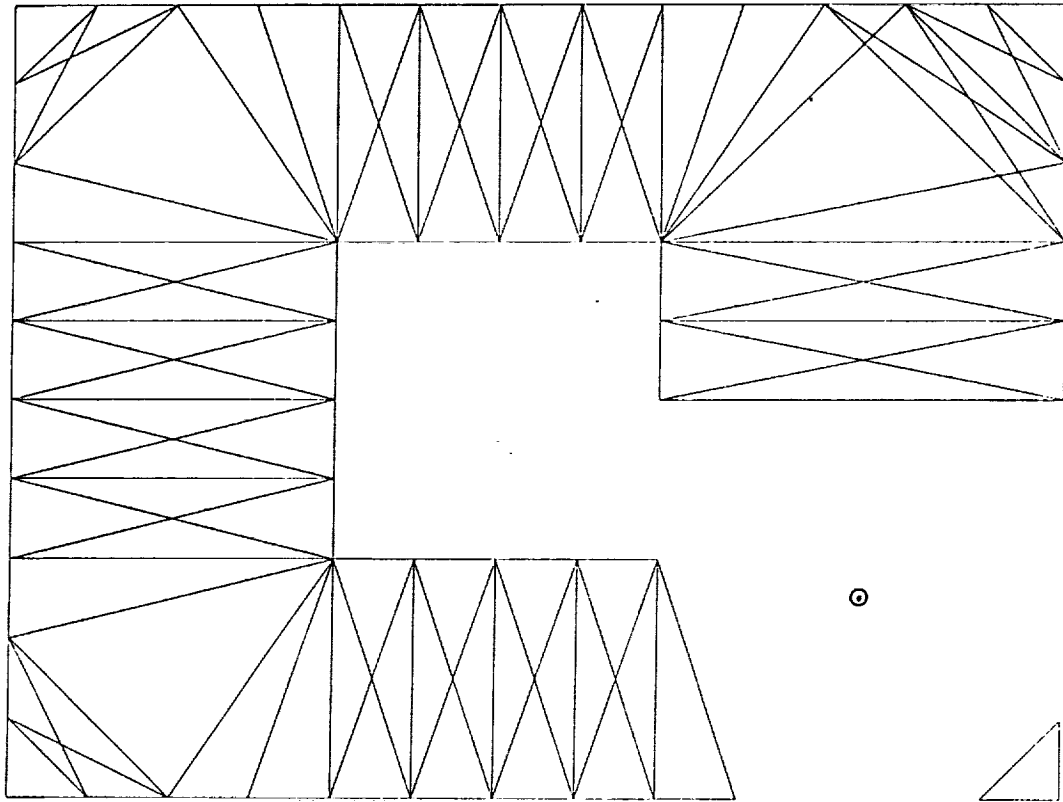
Reconnect Sides to New Point
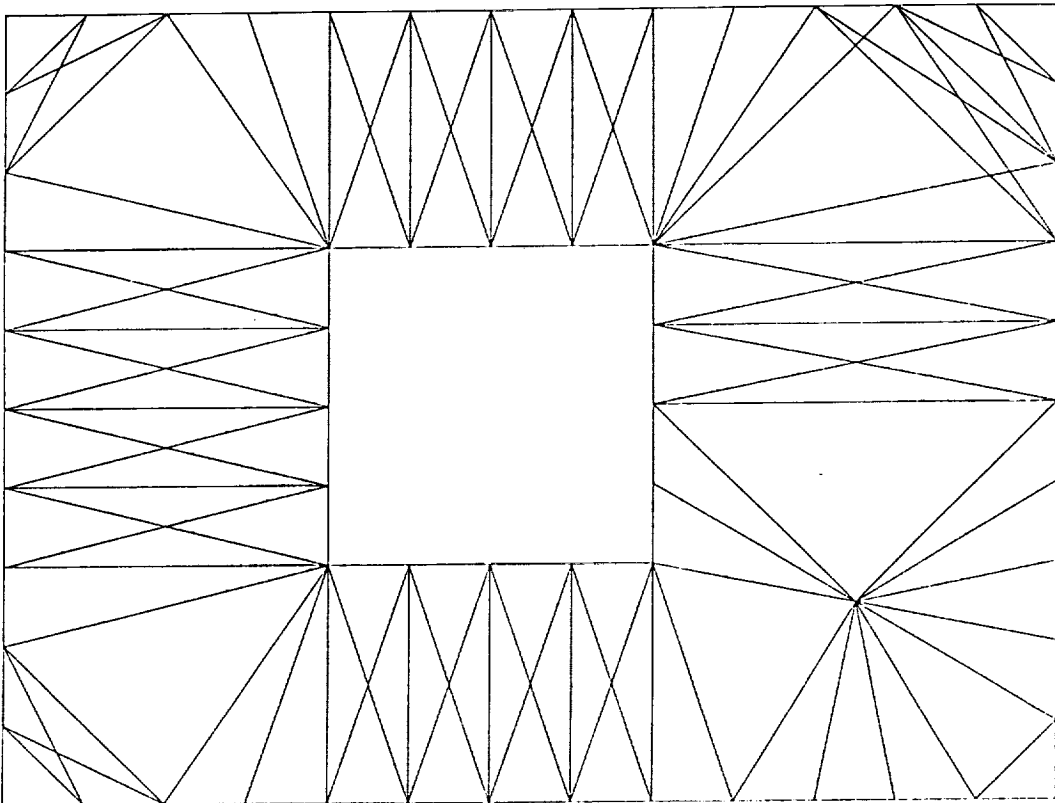
Initial Refinement
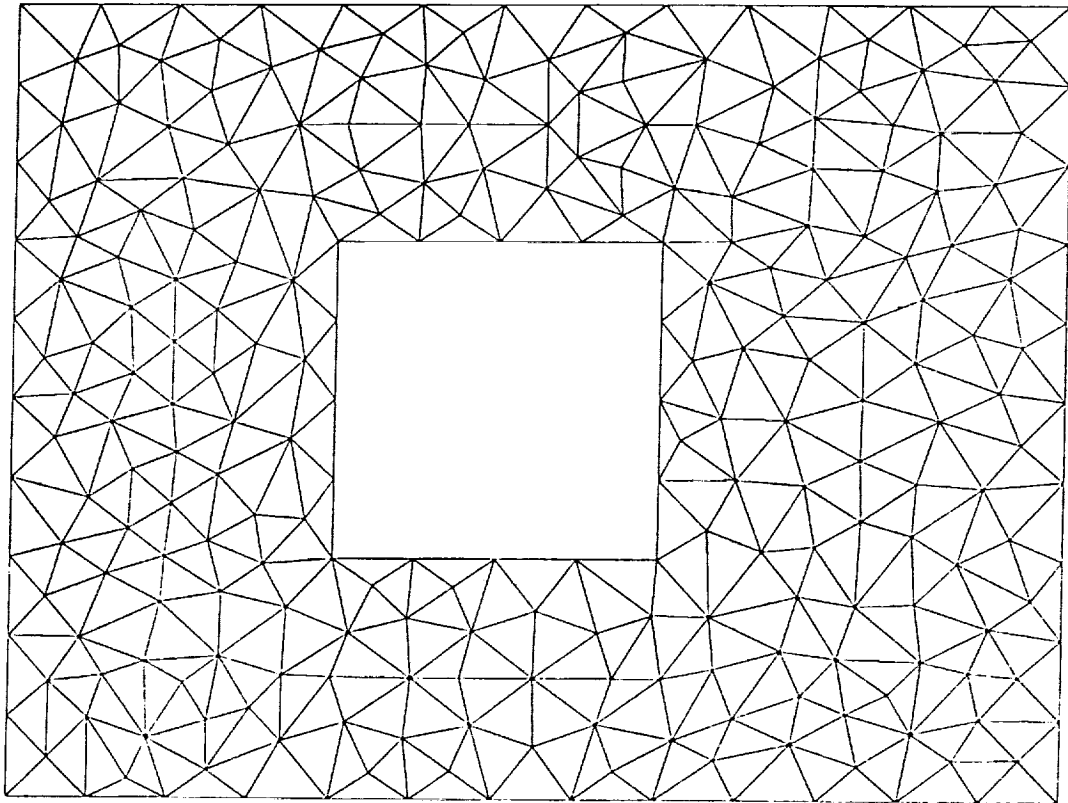
Grid Code Output

Flow code requirements

- Node point x, y coordinates

- Connectivity: cell nodes, cell faces, face cells

- Boundary information: inlet, exit, and solid wall faces
  (periodic faces are treated with special connectivity)
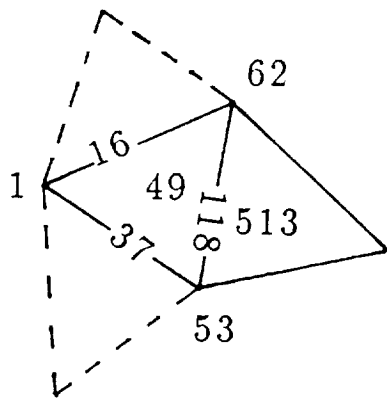
$$NCELL(1:3,49)=37,118,16$$
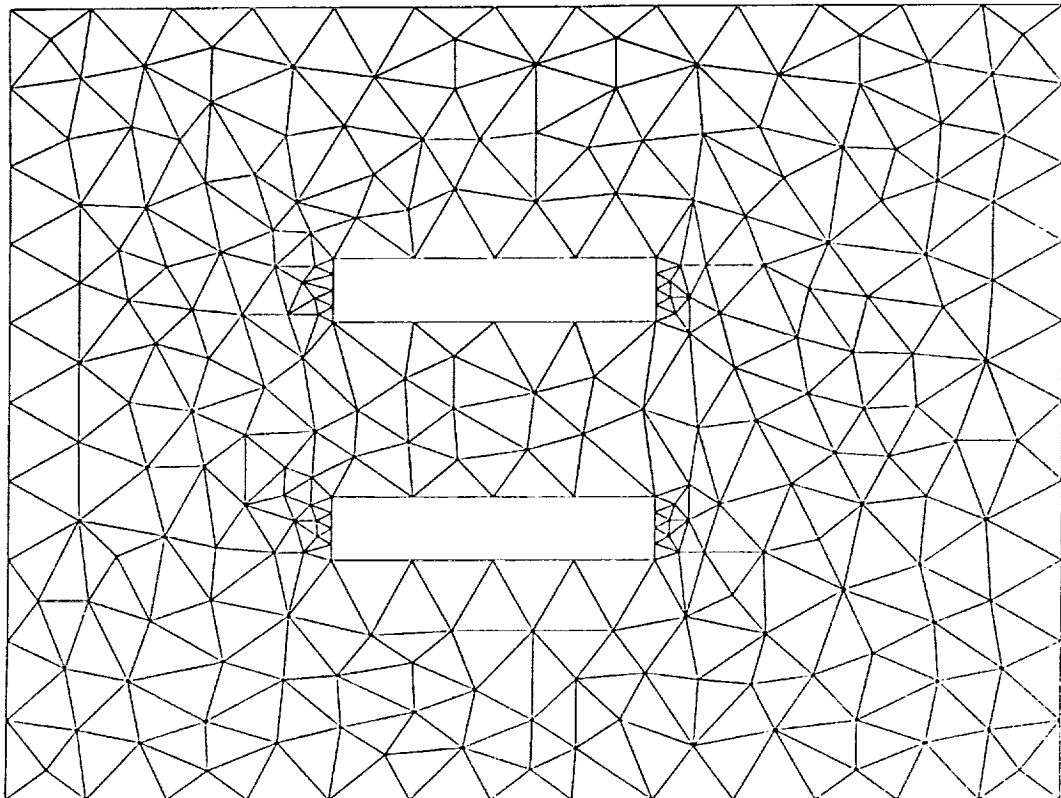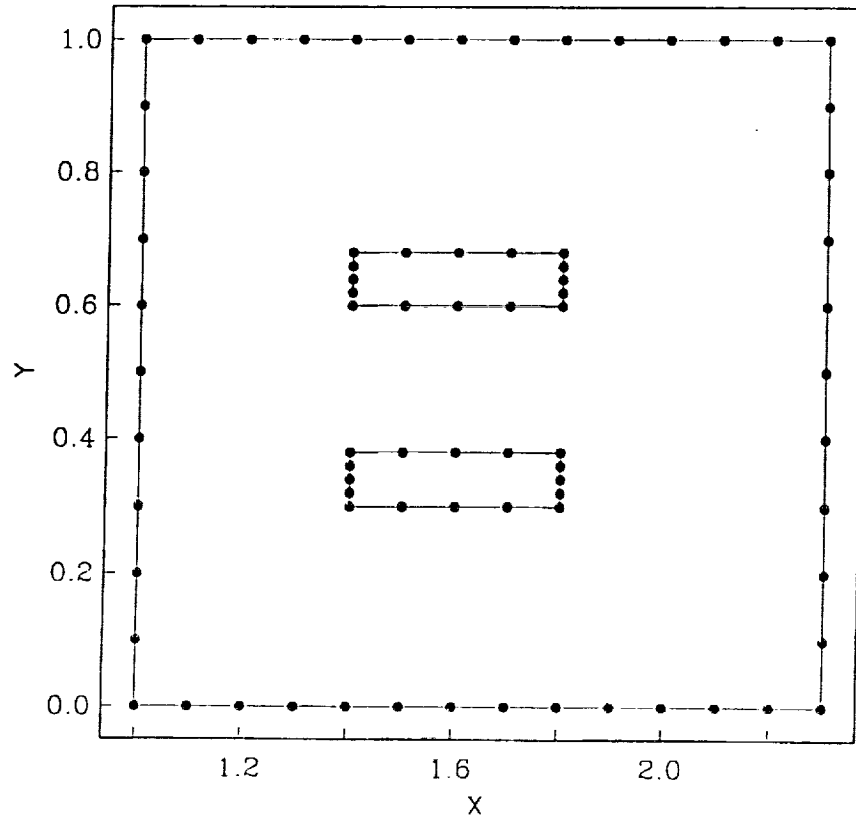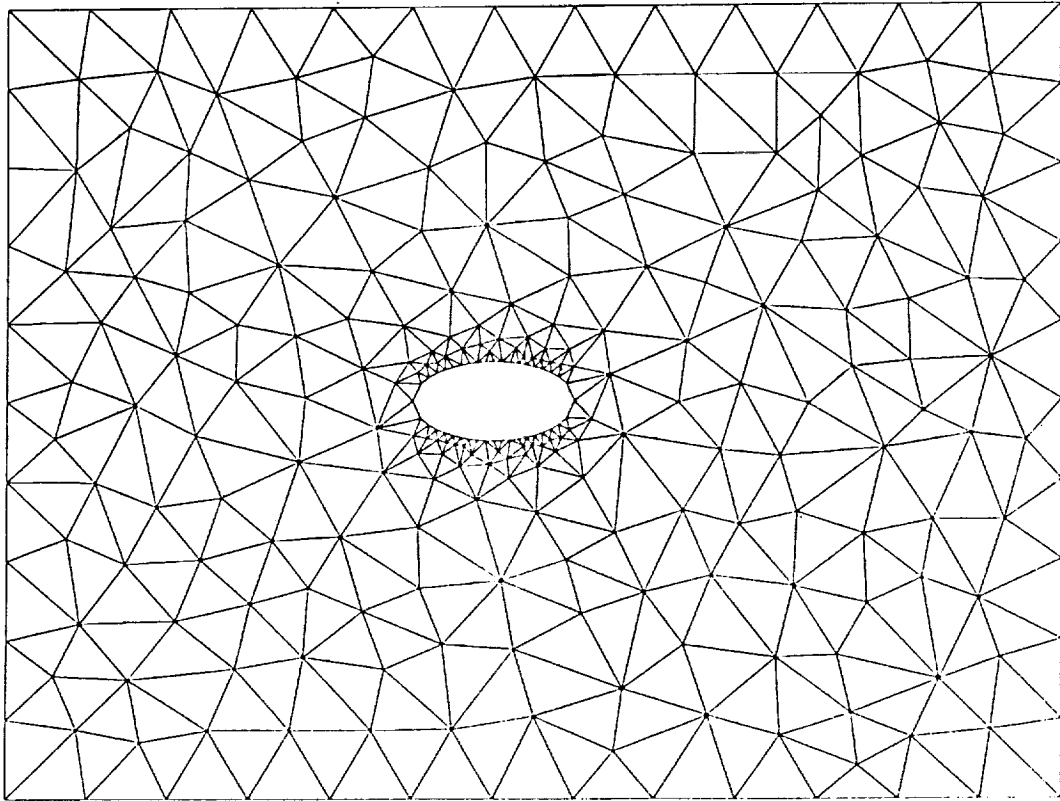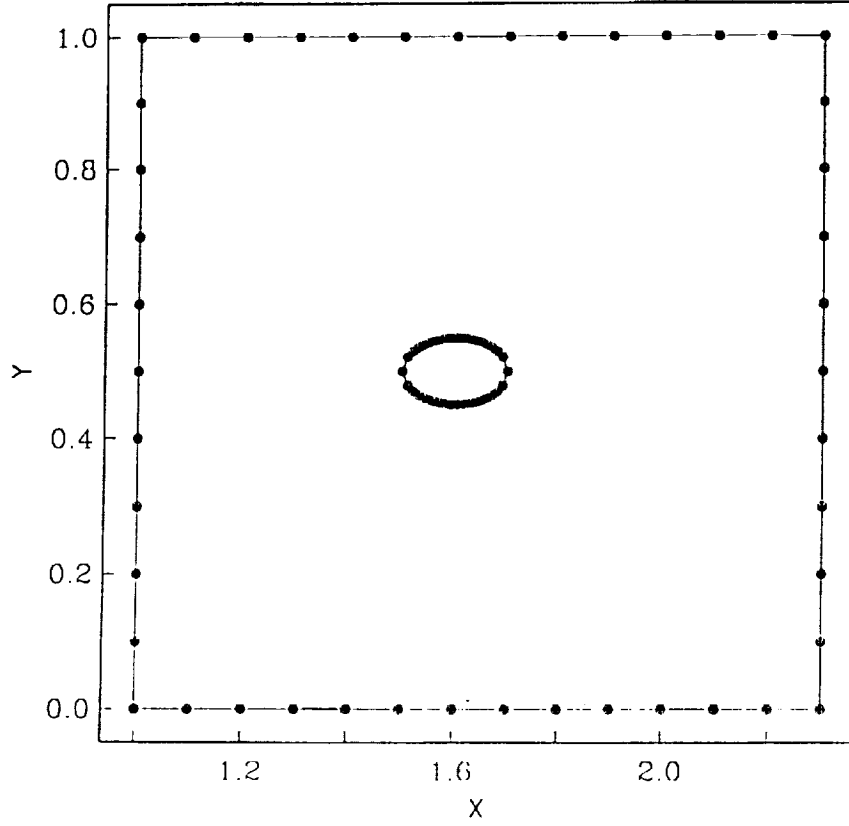$$NCELL(4:6,49)=1,53,62$$
$$NFACE(1:2,118)=49,513$$

BOUNDARY POINTS

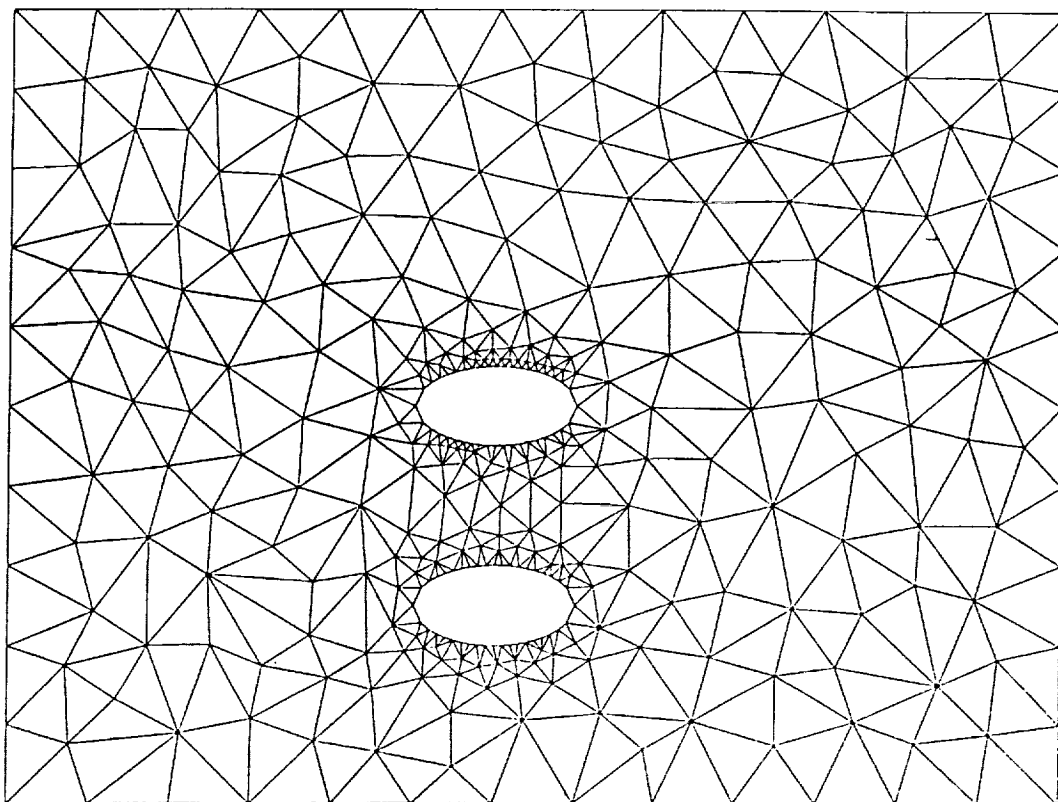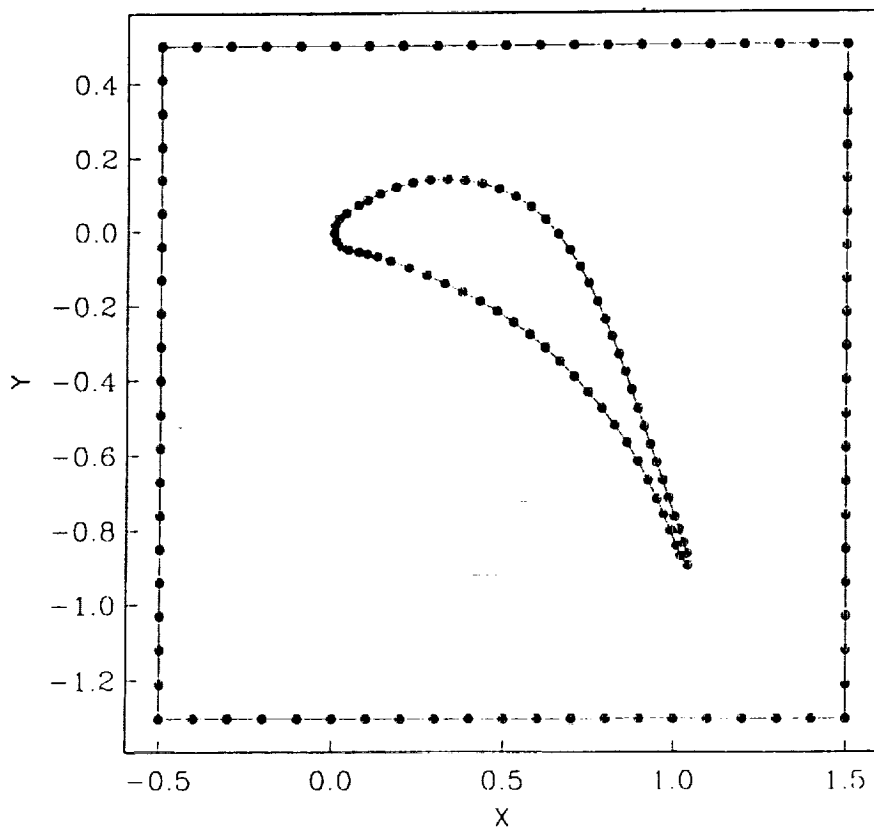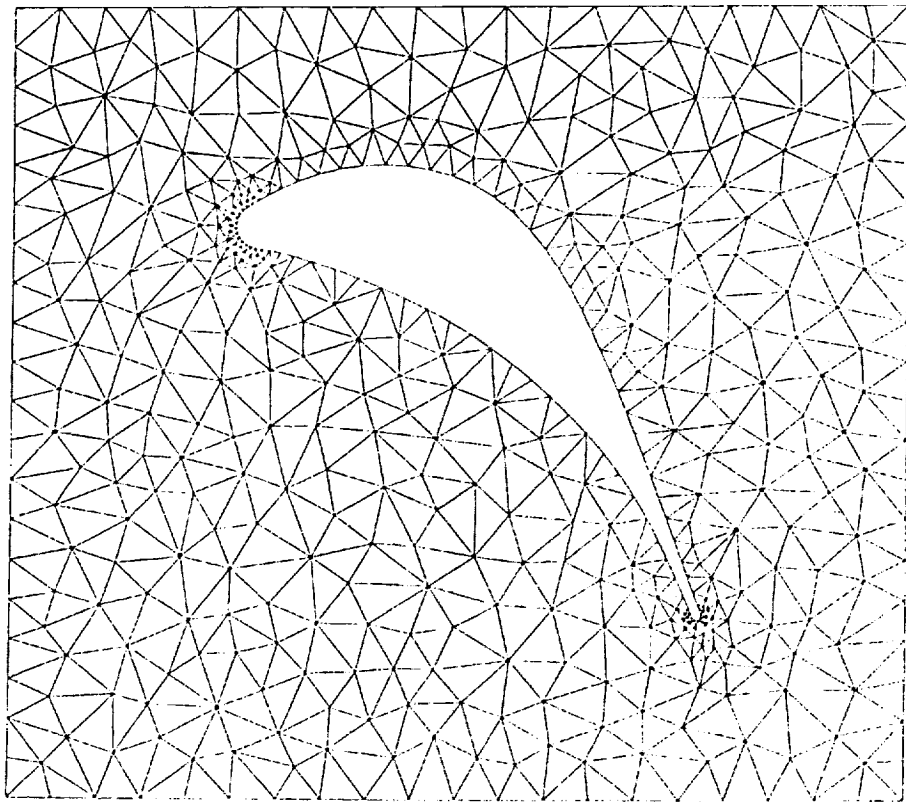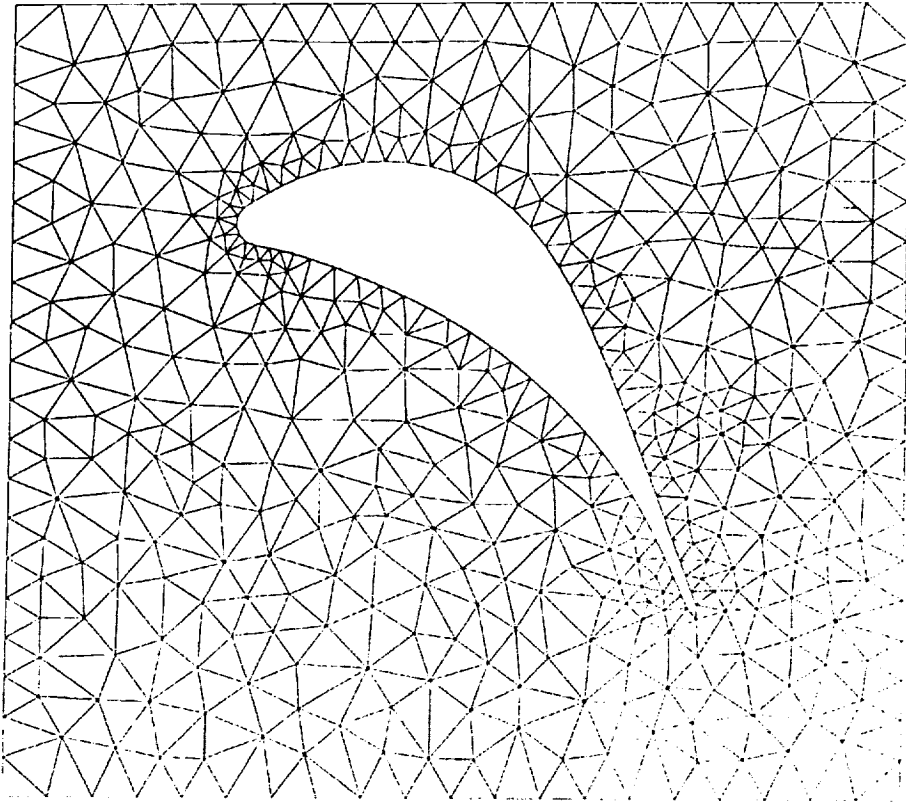# BOUNDARY POINTS

102
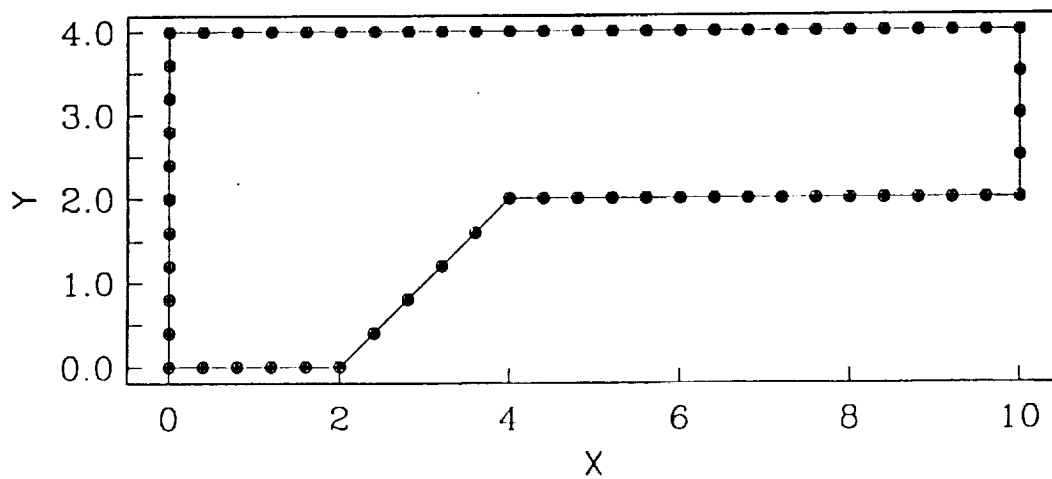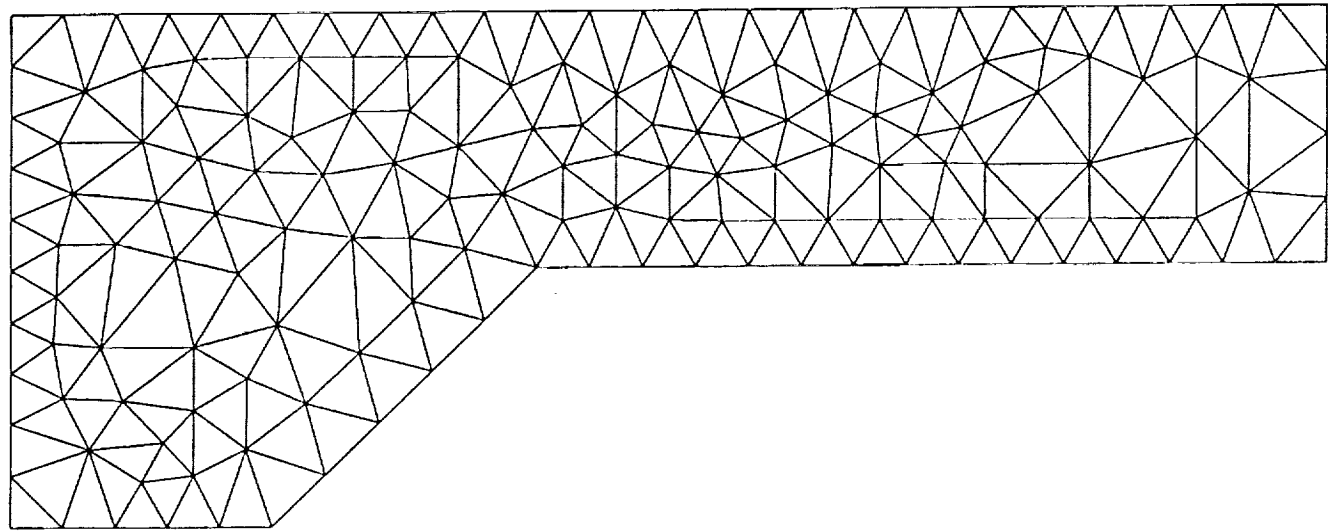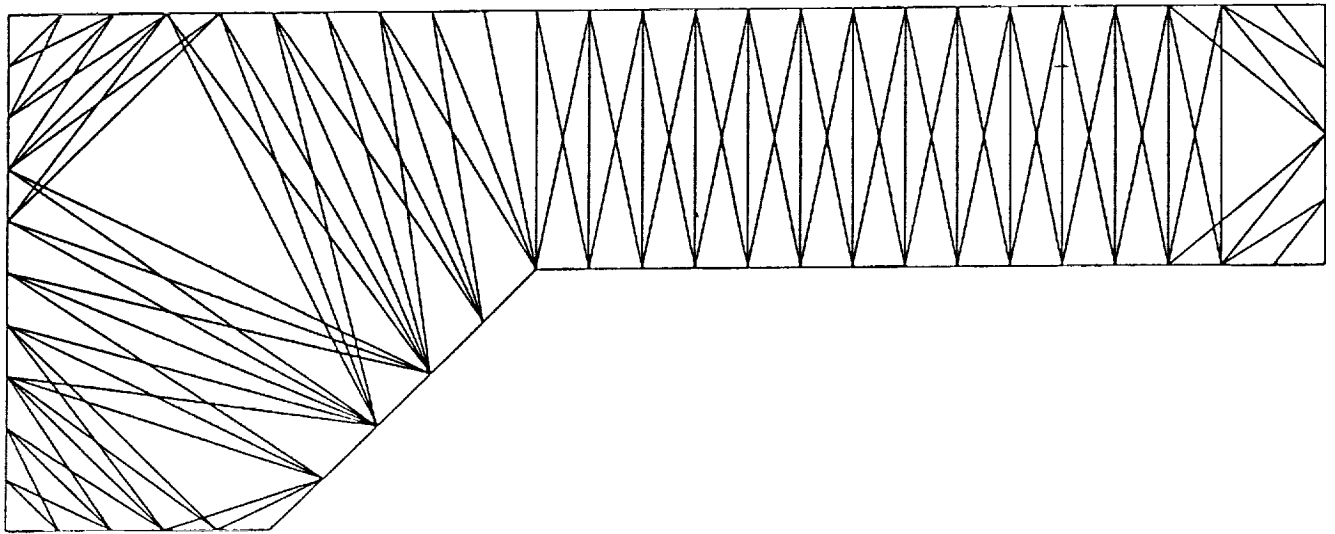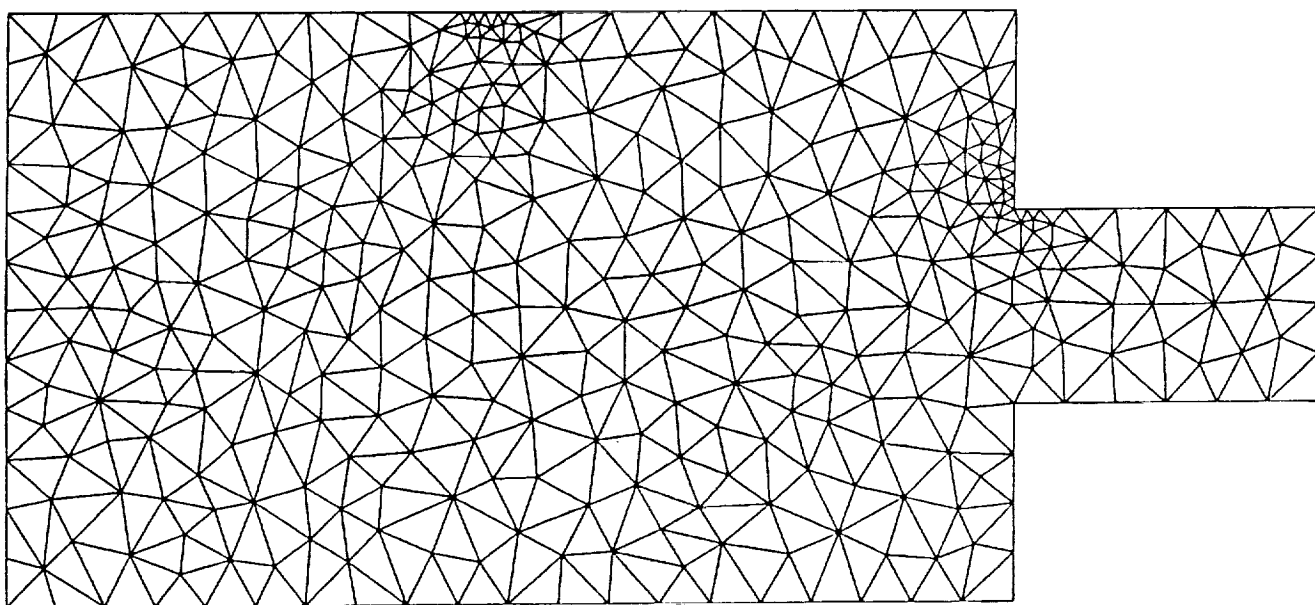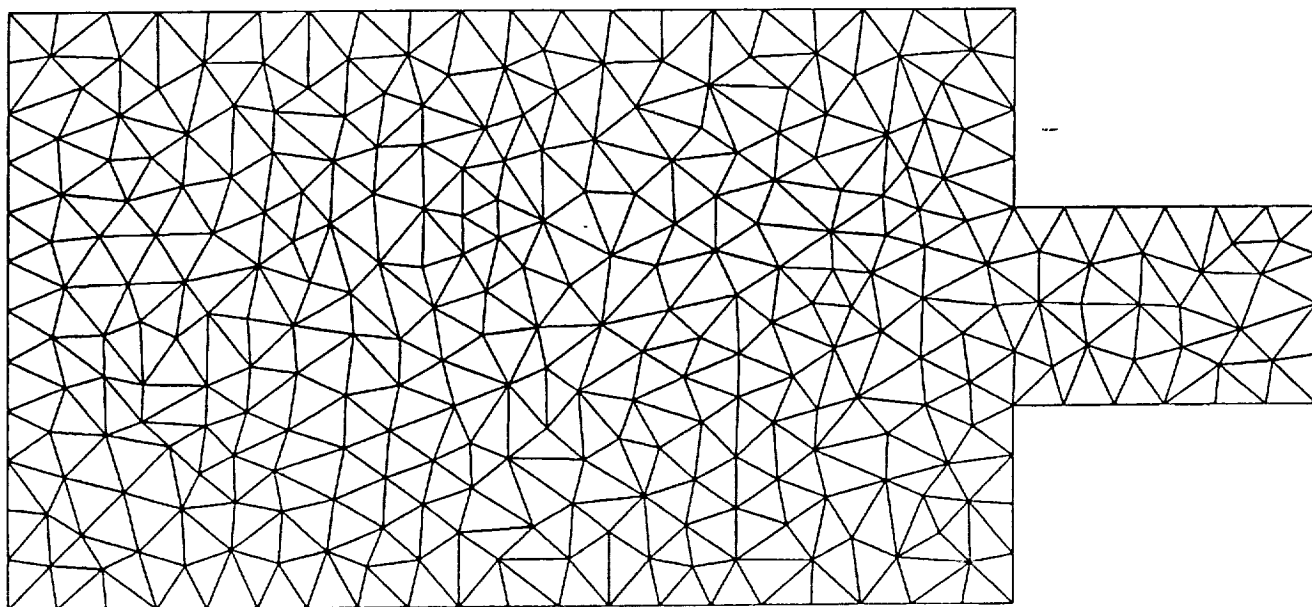
BOUNDARY POINTS

BOUNDARY POINTS

Current Status

- Interactive plotting package with PostScript dump

- Bandwidth minimizer

- Quantify a "good" unstructured triangular mesh for Euler or Navier-Stokes flow codes

Future Plans

- Add capability to refine boundaries using splines

- Incorporate regular unstructured mesh in portions of the domain for Navier-Stokes flow code

Cartesian Based Grid Generation/Adaptive Mesh Refinement
by
William J. Coirier
NASA Lewis Research Center

1 Abstract

Grid adaptation has recently received attention in the CFD community as a means to capture the salient features of a flowfield by either moving grid points of a structured grid or by adding cells in an unstructured manner. Grid adaption by moving the grid points of a structured grid has the advantage of an efficient data representation that vectorizes well on vector machines. But, considering the recent advances in speed available on current workstations, and the long waiting times to get an available processor due to a high number of users, this advantage is becoming less important. In addition, since this approach always has a fixed number of cells, it can result in moving grid points away from regions where they are actually needed ("robbing Peter to pay Paul"). An alternative formulation would be to adaptively refine: that is, add new cells where the flow physics/truncation error dictate. There are basically two subsets to this adaptive refinement approach. One is based on a "background" structured curvilinear grid, where the cells (and their children) of the background grid are refined. This necessitates generating a base structured curvilinear grid, which for realistic geometries (such as multi-element airfoils or inlets with bypass doors and bleed ducts) is a formidable if not impossible task. The resulting data structure from this approach is termed unstructured, in that cell and face connectivity must be stored, and is not implicitly found by the cell index (as in an array).

The approach investigated here is based upon a background cartesian mesh, from which the geometry is "cut" out of the mesh. Once the mesh is obtained, a solution on this coarse grid is found, that indicates which cells need to be refined. This process of refining/solving continues until the flow is grid refined in terms of a user specifed global parameter (such as drag coefficient etc.). The advantages of this approach are twofold: the generation of the base grid is independent of the topology of the bodies or surfaces around/through which the flow is to be computed, and the resulting grid (in uncut regions) is highly isotropic, so that the truncation error is low. The flow solver (which, along with the grid generation is still under development) uses a completely unstructured data base, and is a finite volume, upwinding scheme. Current and future work will address generating Navier-Stokes suitable grids by using locally aligned and normal face/cell refining. The attached plot shows a simple grid about two turbine blades.

OVERVIEW

- Why Unstructured Grids?

- What Really is an Unstructured Approach?

- Cartesian Based Grid Generation

- Examples of Adaptive Mesh Refinement

- Status and Future Directions

Why Unstructured Grids?

- Two Major Advantages:

  - "Automatic" Mesh Generation About/In Arbitrary (Multiple) Bodies/Geometries.

  - Resolution of Disparate Spatial Length Scales by Locally Adaptive Mesh Refinement.

- Disadvantages:

  - "Fully" Unstructured Requires More Memory/Cell
  - Choice of Data Structure is Crucial.

What is Unstructured?

- "Unstructured" is Really a Consequence of:
  - Grid Topology (Connectivity)
  - Flow Solver Approach

- DATA STRUCTURE

  - All Three Intimately Intertwined

- Structured Grid Data Structure:
  - Connectivity Found by Incrementing/Decrementing Indices of Arrays Containing Information (X,U etc.)

What is Unstructured? (cont.)

- Unstructured Data Structure(s) Must Be Able to Directly or Indirectly Obtain:
  - Cell to Cell Connectivity (Cell Neighbors)
  - Cell to Face Connectivity (Flux Balance)
  - Face to Cell Connectivity (Residual Distribution)
  - Cell to Node Connectivity (Flux and Plotting)

- Choice of What and How to Store and What to Infer Has Major Impact on Speed/Memory and Refinement Types

- Choice of Programming Language Can Greatly Hinder or Help:
  - Using C Language for Dynamic Memory Alloc/Dealloc, Memory Pointers, "Make Your Own" Data Structures . . .

Cartesian Based Grid Generation

- Data Structure: Linked Lists of Cells, Faces and Nodes.
- Cells Contain Lists of Edges Which Point to Faces
- Faces Point to Cells and Nodes

- Cells Split by Joining Midpoints of Edges

FACE

CELL FACE FACE EDGE C —|— C
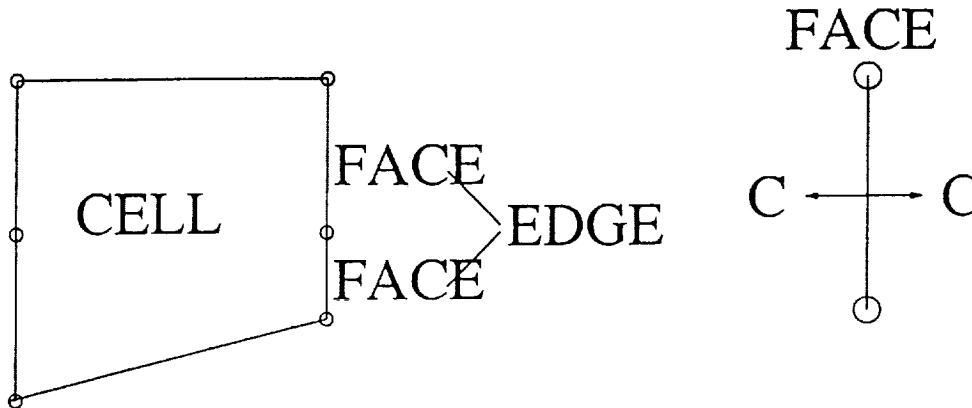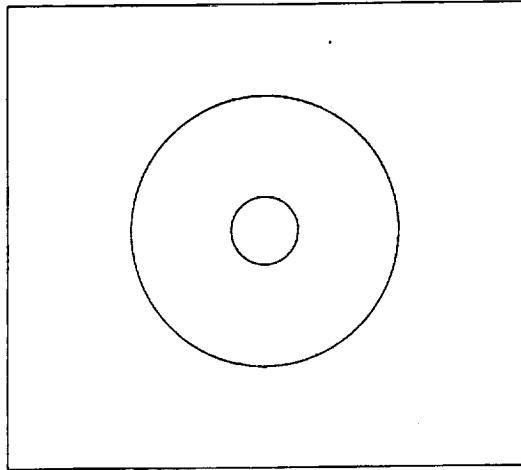
Cartesian Based Grid Generation

- Basic To The Approach:
  - Generate a Suitable "Background" Mesh of Cartesian Cells and "Cut" the Bodies/Geometry Out of the Mesh

- In Uncut Regions Mesh is Highly Isotropic
- Regularity of Background Mesh Helps in Cutting Geometry

- Method Can Generate Base Grids for Arbitrary Geometries Automatically and Inexpensively

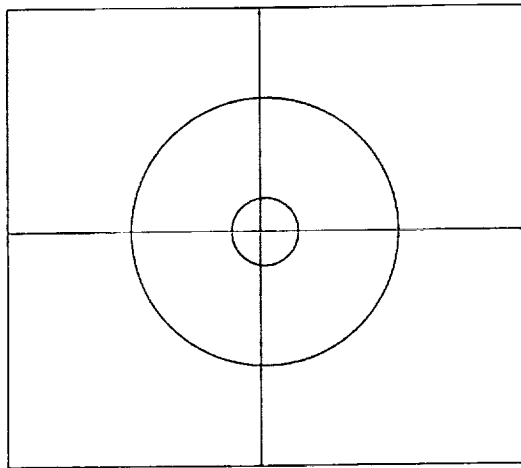- Flow Solver Then Successively Solves/Refines Starting With Base Grid

110

Cartesian Based Grid Generation

- Start With Single Cell Spanning All Bodies
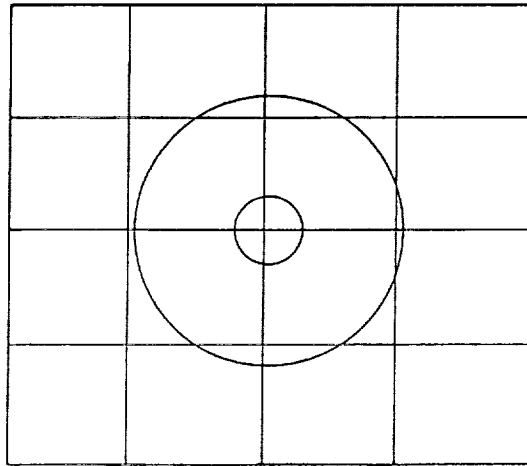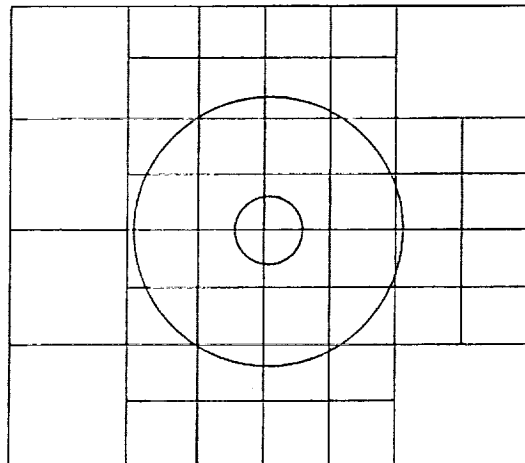- All Bodies Not Cut: Quad Refine

Cartesian Based Grid Generation

- All Bodies Cut, But Not Delta-s Refined:
- Quad Refine Cells That Need to Be Refined

Cartesian Based Grid Generation

- All Bodies Cut, But Not Delta-s Refined:
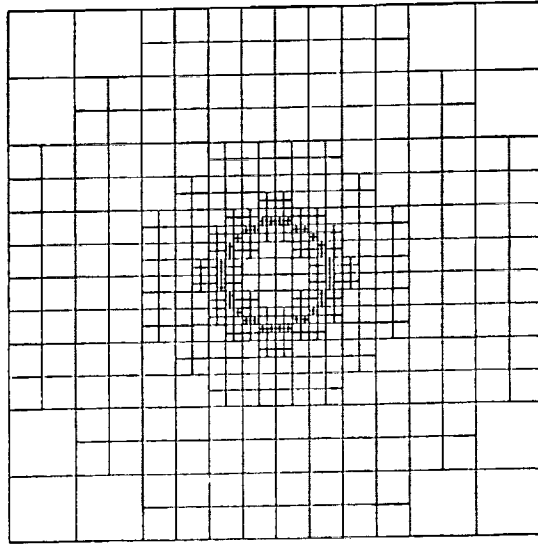- Quad Refine Cells That Need to Be Refined

Cartesian Based Grid Generation

- Still Not Delta-s Refined:
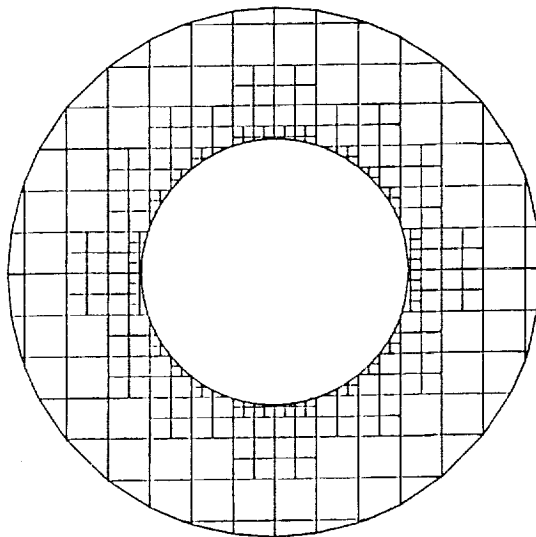- Quad Refine Cells That Need to Be Refined

112

Cartesian Based Grid Generation

- Delta-s Criteria is Met

Cartesian Based Grid Generation

- Cut Geometry Out of Background Mesh

Geometry Definition

- Arbitrary Number of Bodies Allowed
- Each Body Parameterized as Collection of Basis Functions (For Now Uses Cubic Splines)
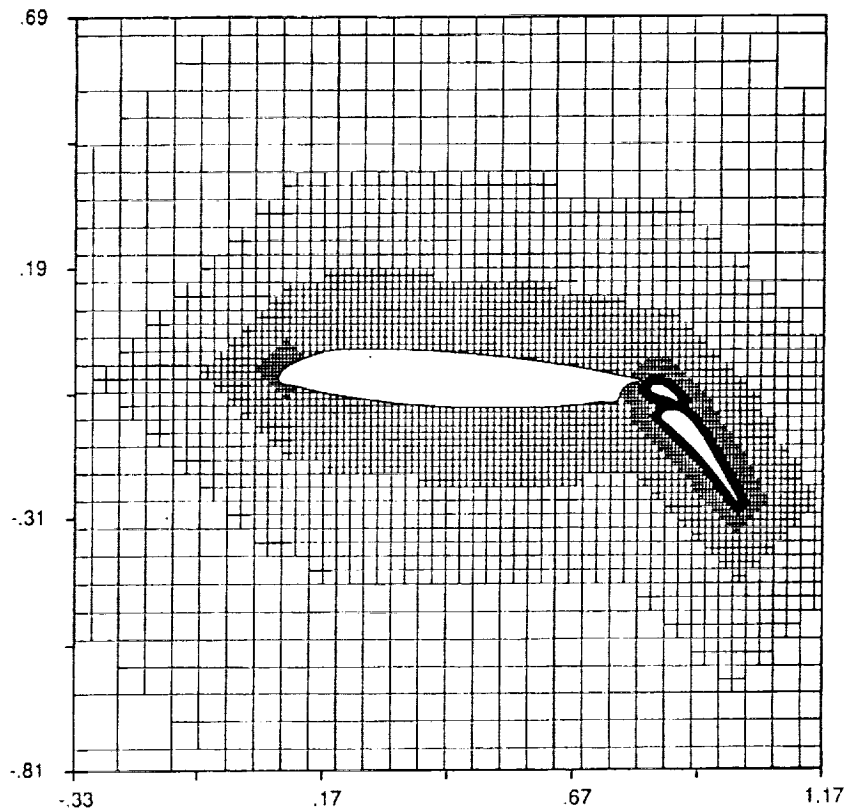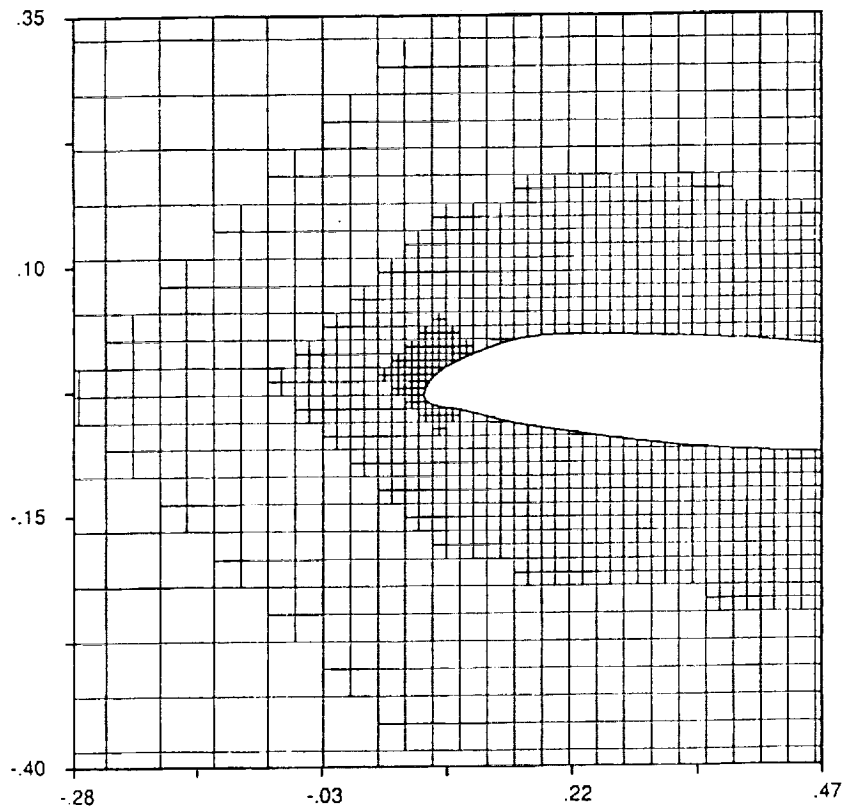
$$X(s) = \Sigma C_{0j}(s - s_j)^j$$

$$Y(s) = \Sigma C_{1j}(s - s_j)^j$$

- Bodies Are Cut Out of Background Mesh When Cut Face Lengths are Within Specified Surface Lengths:

$$\Delta S(s) = \Sigma C_{2j}(s - s_j)^j$$

- Allows Surface Face Lengths to be Governed by Some Geometric Parameter(s): e.g. - Radius of Curvature

114
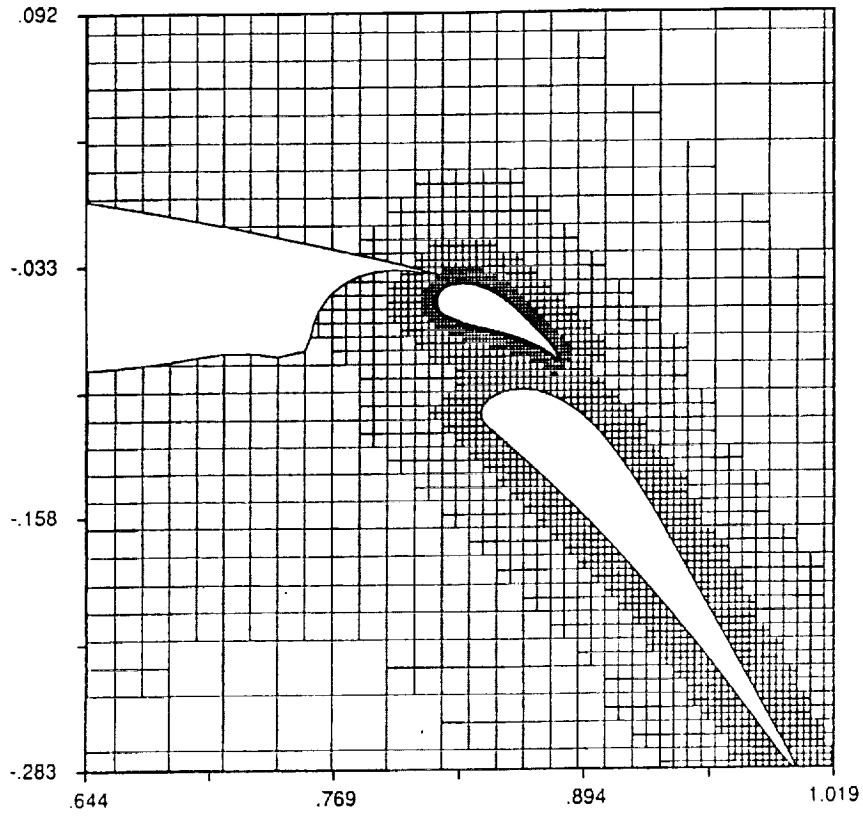
Cartesian Based Grid Generation

- For Navier-Stokes Calculations Need Cells Locally Aligned Near Body Surfaces

- Developed Data Structure Refines Cells by Joining Midpoints of Cell Edges

- "Locally" Aligned Cells Can be Made by Joining Midpoints of Edges Whose Faces are Cut By Body
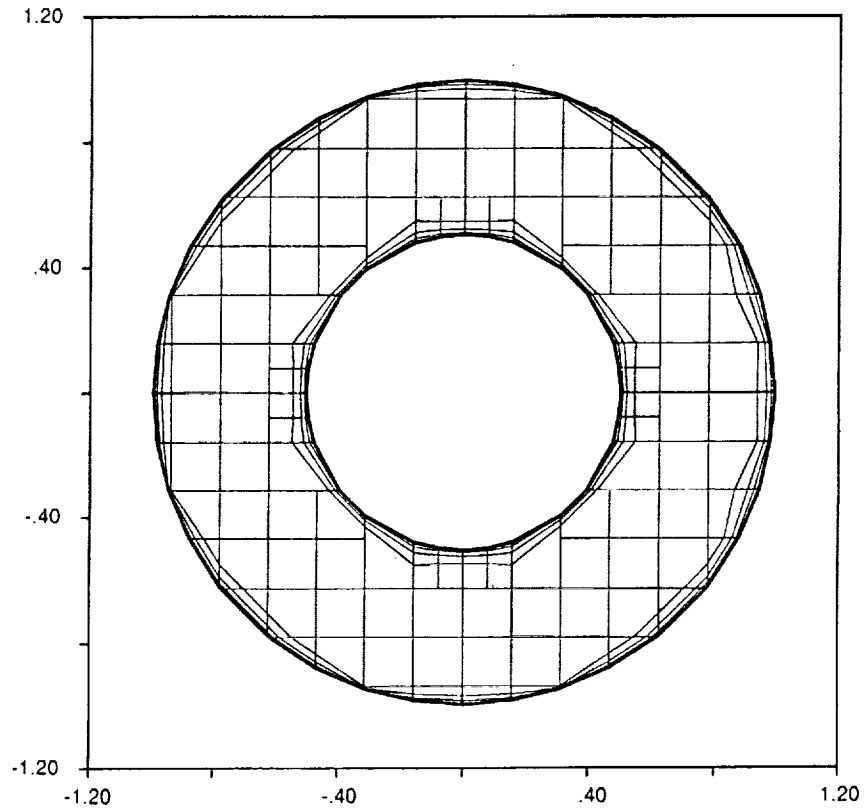
Adaptive Mesh Refinement

- Idea: Add Cells to Regions of Flow Where Gradients/Truncation Error Dictates

- 2D "Inviscid" Burger's Equation:

- Characteristics-like Solution:
    - Wave Steepening
    - Shock Formation

- 2nd Order Accurate, Upwinded, Monotone Scheme

- Simple Adaptation Criteria: Delta-u Across Face (Better Adaptation Criteria to be Used Later.)
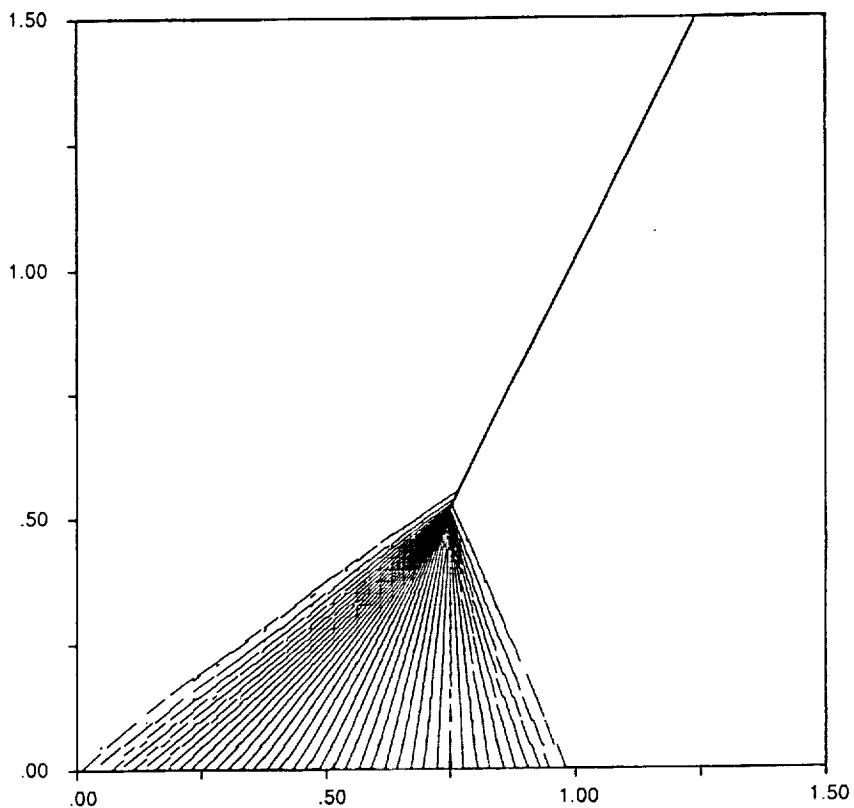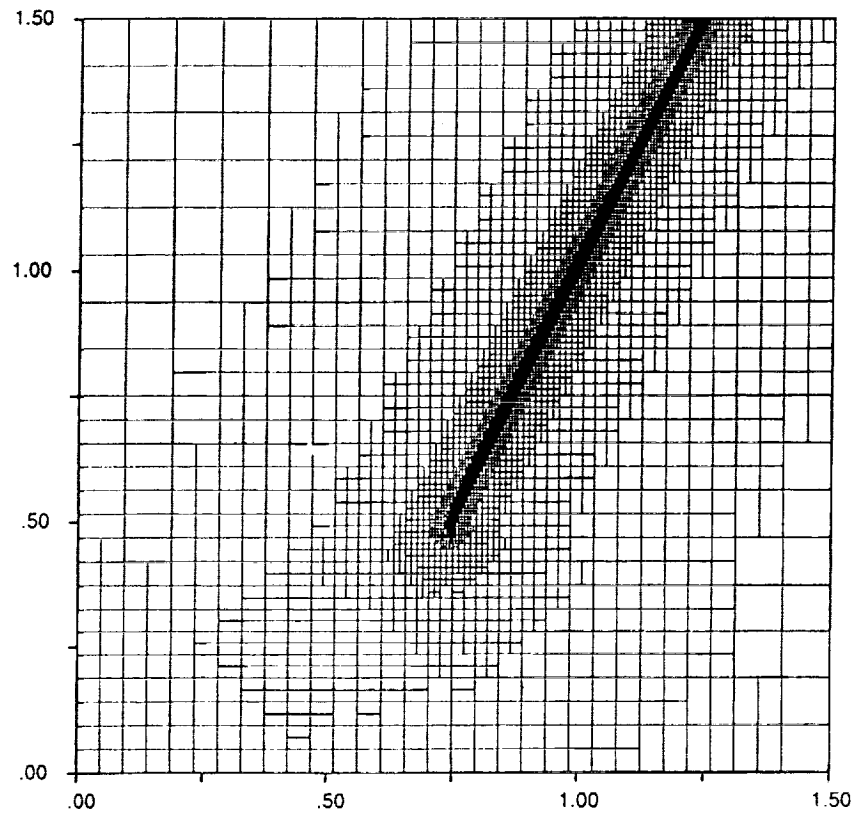
Adaptive Mesh Refinement

- Circular Convection Problem:
  - Linear Convection Equation (2D)
  - "Inflow" Convected About (X,Y) = (0,0.5)

- Inflow is a "Circular Gaussian" Profile (Smooth).

118
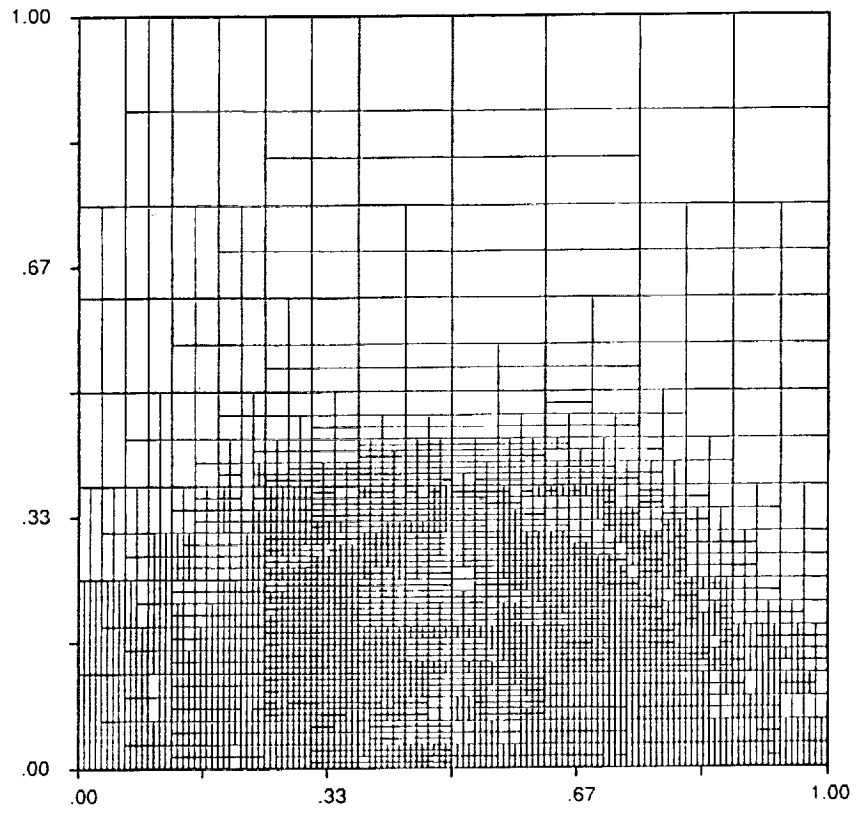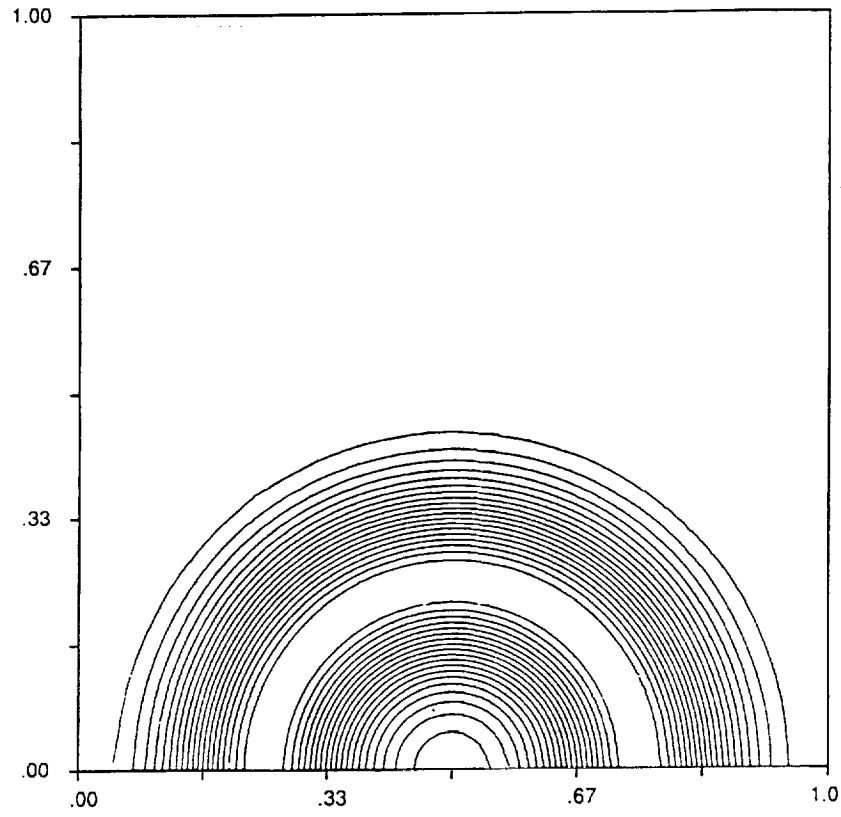
Status

- Grid Generation:
  - Euler Suitable Meshes Nearly Finished
  - NS Suitable Meshes Under Development

- Flow Solver
  - Development to Begin "Any Day Now"

Future

- Goal Is To Develop 2D Navier-Stokes Code:
  - High Order Upwind Scheme
  - Reconstruction Using L2-Norm or Green Gauss
  - Matrix Preconditioning
  - Residual Smoothing
  - Vector Sequencing (?)

Grid Generation Research at OSU
by
S. Nakamura
Ohio State University

The research of grid generation at OSU started in late 1970's. Our past contributions include development of adaptive grid generation procedure for transonic airfoil, and parabolic grid generation algorithm for wing-fuselage geometries. The latter evolved later as the parabolic-hyperbolic grid generation method, which has been used extensively in automobile industry.

In the last two years, our effort has been concentrated on (1) surface modeling, (2) surface grid generation, and (3) 3-D flow space grid generation.

The surface modeling shares the same objectives as the surface modeling in CAD, so software available in CAD can in principle be used for solid modeling. Unfortunately, however, the CAD software cannot be easily used in practice for grid generation purpose, because they are not designed to provide appropriate data base for grid generation. Therefore, we started developing a generalized surface modeling software from scratch, that provides the data base for the surface grid generation.

Generating surface grid is an important step in generating three-dimensional grid for flow space. To generate a surface grid on a given surface representaion, we developed a unique algorithm that works on any non-smooth surfaces.

Once the surface grid is generated, three-dimensional space grid can be generated. For this purpose, we also developed a new algorithm, which is a hybrid of the hyperbolic and the elliptic grid generation methods. With this hybrid method, orthogonality of the grid near the solid boundary ban be easily achieved without introducing empirical fudge factors.

During the stay in OAI, I have worked to develope two- and three-dimensional grids for turbomachinery blade geometries. As an extension of this research we are planning to develop an adaptive grid procedure with an interactive graphic environment.

121

Interactive Solution-Adaptive Grid Generation Procedure

by
Yung K. Choo
NASA Lewis Research Center
and
Todd Henderson
OAI Summer Intern
NASA Lewis Research Center

ABSTRACT

TURBO-AD is a 2D interactive solution-adaptive grid generation program. The code uniquely combines a grid adaptation technique that uses parametric mapping with control sources and an algebraic grid generation that uses control points into a single software package. The grid adaptation is achieved by first adapting the control points to a numerical solution in the parametric domain using control sources obtained from flow properties. Then a new grid is generated from the adapted control net using the control point formulation. The new adapted grid in the parametric domain is then mapped back to the physical domain. This solution-adaptive grid generation process is efficient because the number of control points is much less than the number of grid points and the grid generation from the adapted control net is an efficient algebraic process.

This presentation consists of three parts. Part one briefly discusses the control point form (CPF) and illustrates some of the features of a menu-driven interactive code, TURBO-I, that uses CPF. Part two discusses a solution-adaptive grid generation technique that uses parametric mappings. Finally, part three dicusses the interactive solution adaptive grid generation procedure which adapts the control net to solutions instead of the grid.

OUTLINE

PART ONE

CONTROL POINT FORM (CPF) OF ALGEBRAIC GRID GENERATION AND TURBO-I

OBJECTIVES

    TO DEVELOP EASY, EXPLICIT GRID MANIPULATION CAPABILITY

ADVANTAGES

    THE CPF-BASED GRID GENERATION:

- CAN EASILY ENHANCE GRID QUALITY BY VRIOUS LOCAL/GLOBAL
  GRID DISTRIBUTION STRATEGIES

- IS EFFICIENT AND EASY TO IMPLEMENT

  WORKS WELL IN THE INTERACTIVE COMPUTER GRAPHICS ENVIRONMENT

- CAN BE USED AS A COMPLEMENTARY TOOL AS WELL AS
  A STAND-ALONE TOOL

124

REFERENCES

Eiseman, P.R., "A Control Point Form of Algebraic Grid Generation," Int. J. Num. Methods in Fluids, Vol. 8, pp 1165-1181, 1988

Choo, Y.K., Miller, D. P., Reno, C., "Implementation of Control Point Form of Algebraic Grid Generation Technique," Numerical Grid Generation in Computational Fluid Dynamics and Related Fields, Edited by A.S.-Arcilla, et al, North-Holland, pp 331-341, 1991

TECHNICAL APPROACH - CONTROL POINT FORM (CPF)

CONSTRUCTION OF A CURVE & A CONTROL NET
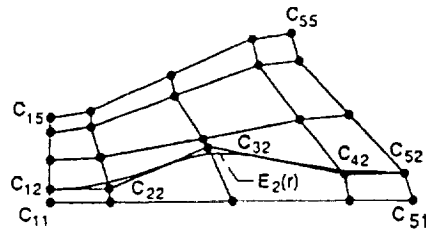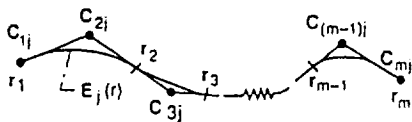
$$Q(r,t) = T(r,t) + a_1\left[1 - G_1(r)\right]\left[P(1,t) - F_1(t)\right]$$
$$+ a_2 G_{N-1}(r)\left[P(N-1,t) - F_N(t)\right]$$
$$+ a_3\left[1 - H_1(t)\right]\left[P(r,1) - E_1(r)\right]$$
$$+ a_4 H_{M-1}(t)\left[P(r,M-1) - E_M(r)\right]$$

125

INTERACTIVE GRID GENERATION
Program TURBO

CONSTRUCT A SIMPLE CONTROL NET GENERATE SURFACE GRID

COMPUTE & EXAMINE INITIAL GRID

SELECT A CONTROL POINT TO MODIFY GRID

EXAMINE NEW GRID -ACCEPT OR REPEAT THE PROCESS

COMPUTE NEW GRID

TRANSLATE THE CONTROL POINT
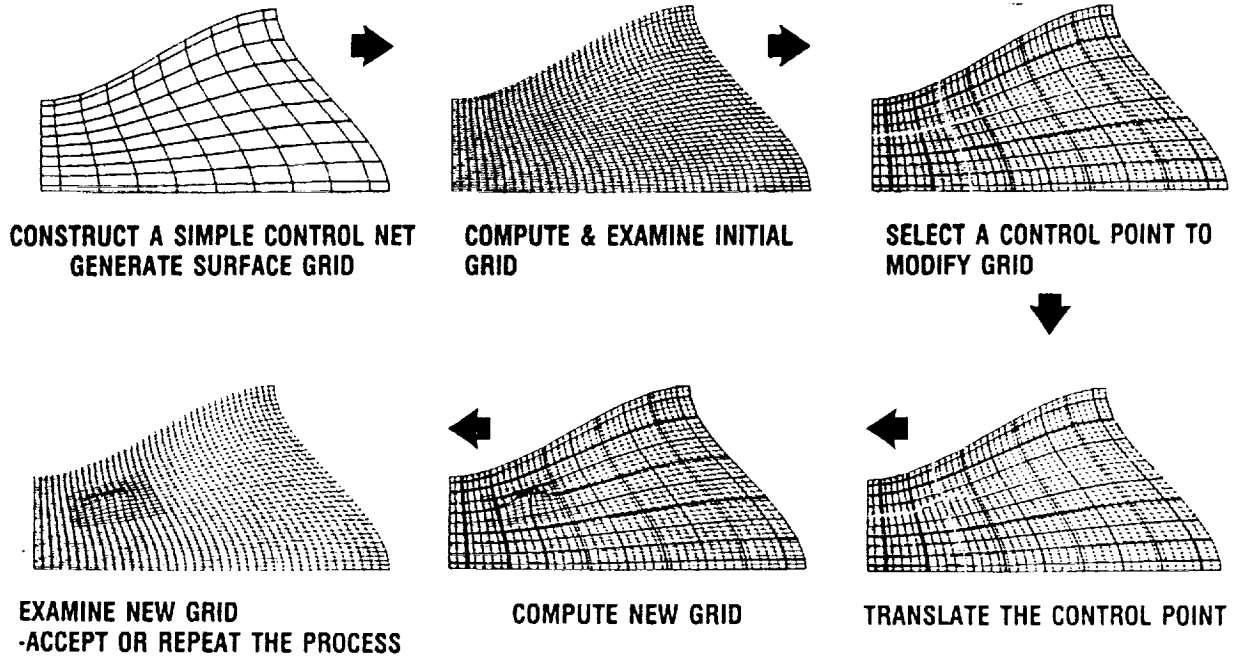
Eiseman/Choo

CD-88-37898

A SAMPLE CASE

FROM A GIVEN GEOMETRY, TO BOUNDARY GRID,

...., TO FINAL GRID

Given Geometry
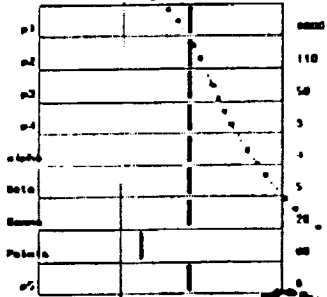
Choose Boundary
Segment and
Discretize

Boundary Grid
Can easily be
Redistributed
interactively

127

TURBO/I INTERACTIVE PROCESS - CONTINUED

STRETCHING

128
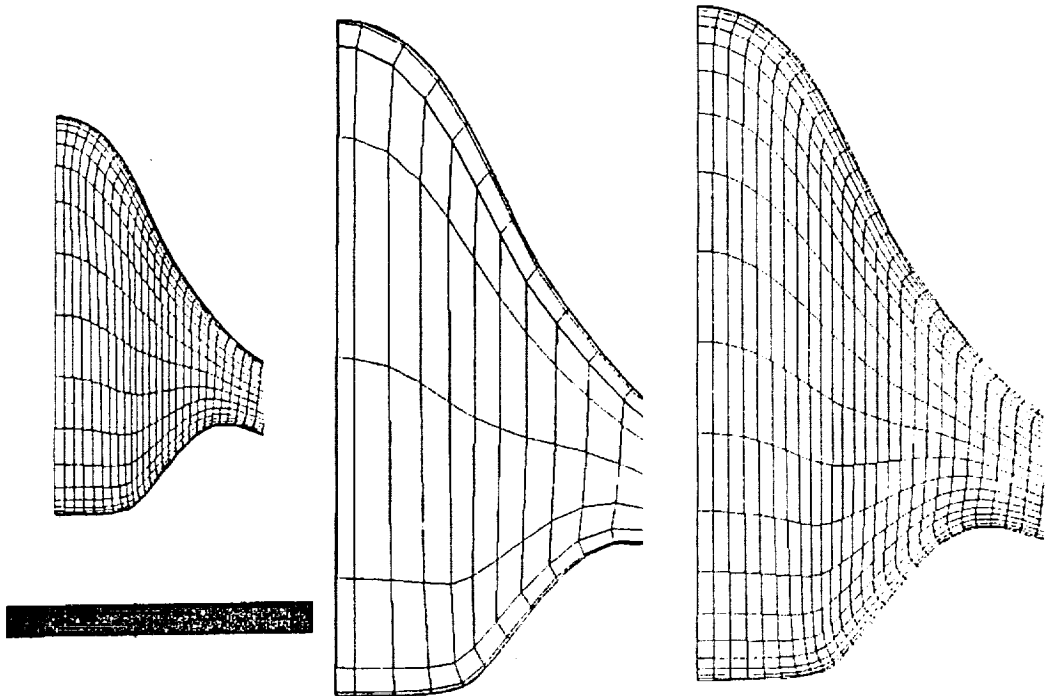
PART TWO

SOLUTION-ADAPTIVE GRID GENERATION USING PARAMETRIC MAPPINGS WITH SOURCE CONTROL

OBJECTIVES

TO DEVELOP GRID ADAPTATION CAPABILITY TO SOLUTIONS, GRID QUALITY, AND GEOMETRY TO OBTAIN ACCURATE PREDICTION OF COMPLEX FLOWS

ADVANTAGES

THE APPROACH IS EASY TO IMPLEMENT AS ONE OF THE REDISTRIBUTION SCHEME

THE USE OF GRID CONTROL SOURCES ALLOWS TO ADAPT THE GRID BY USING LINEAR COMBINATIONS OF FLOW PROPERTIES

THE APPROACH CONTROLS BOTH GRID DENSITY AND QUALITY WITHOUT ANY ADVERSE INFLUENCE OF ONE TO OTHER

THE GRID ADAPTATION SCHEME CAN BE INTEGRATED WITH WELL-DEVELOPED FLOW SOLVERS AS SUBROUTINES.

REFERENCES

Lee, K.D., Loellbach, J.M., Pierce, T.R., "Solution Adaptive Grid Generation Using a Parametric Mapping," 2nd Internation Conference n Computational Fluid Dynamics, Miami Beach, Fl, December 1988

Lee, K.D., Henderson, T.L., Choo, Y.K., "Grid Quality Improvement by a Grid Adaptation Technique," Numerical Grid Generation in Computational Fluid Dynamics and Related Fields, ed. A.S.-Arcilla, et al, North-Holland, pp 597-606, 1991

129
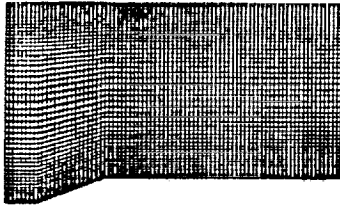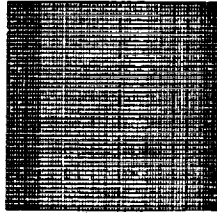
initial grid   (s,t) parametric domain   σ' distribution   solution-adapted grid
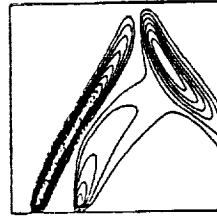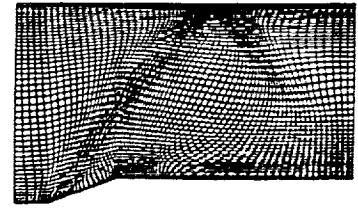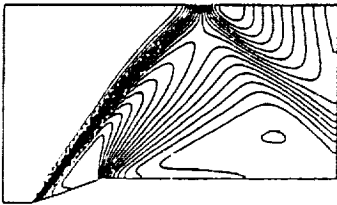
isobars on initial grid   σ' distribution   (s',t') parametric domain   isobars on adapted grid
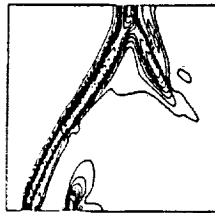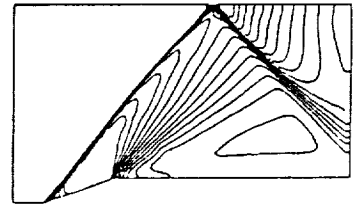
Grid control sources definition:

$$\sigma^s_{kl} = w^s_0 \left| \phi \right| + w^s_1 \left| \frac{\partial \phi}{\partial s} \right| + w^s_2 \left| \frac{\partial^2 \phi}{\partial s^2} \right|$$

$$\sigma^t_{kl} = w^t_0 \left| \phi \right| + w^t_1 \left| \frac{\partial \phi}{\partial t} \right| + w^t_2 \left| \frac{\partial^2 \phi}{\partial t^2} \right|$$

where w_i = weighting factors

ϕ = a grid quality parameter

130

Mapping modification:

$$s'_{ij} = s_{ij} + \sum_{k,l} K^s_{ijkl} \sigma^s_{kl}$$

$$t'_{ij} = t_{ij} + \sum_{k,l} K^t_{ijkl} \sigma^t_{kl}$$

where (s',t') = modified parametric coordinates

K^s_{ijkl}, K^t_{ijkl} = influence coefficients

SAMPLE SOLUTION-ADAPTIVE GRID GENERATION

DOUBLE WEDGED CORNER FLOW

CORNER FLOW
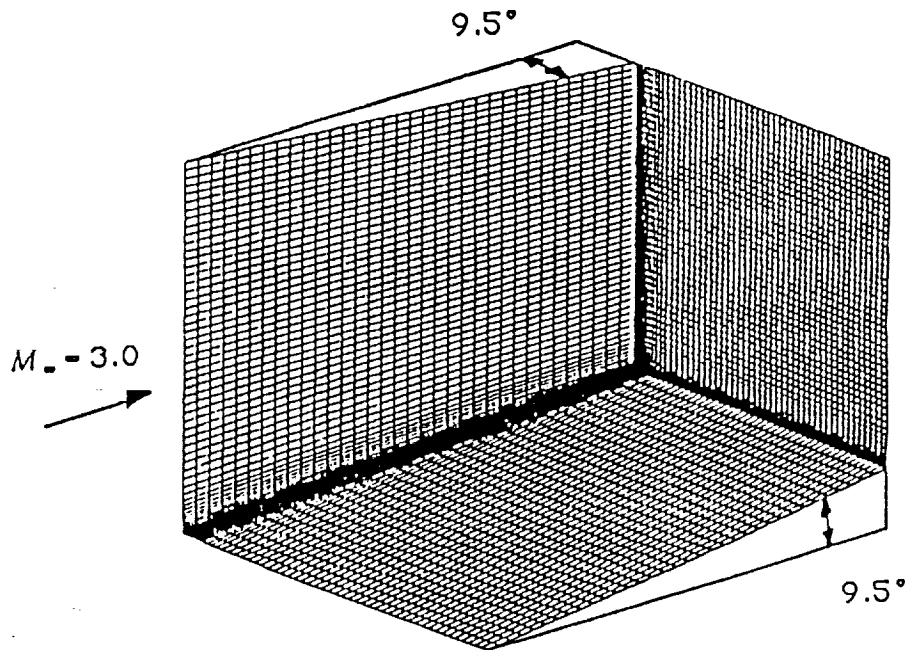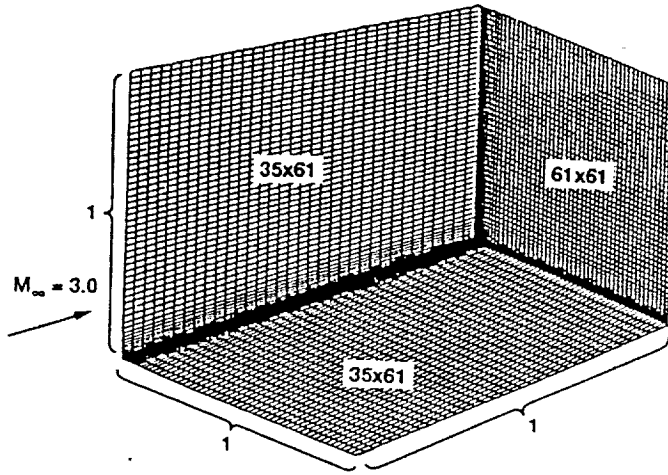
CORNER FLOW
GRID COMPARISON
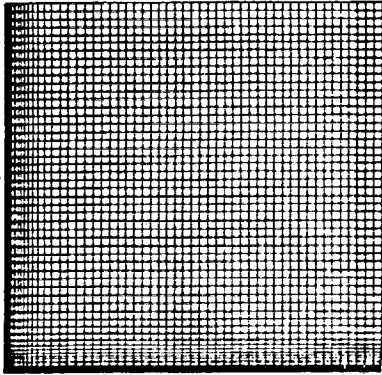
Initial grid
@ x = 0.407
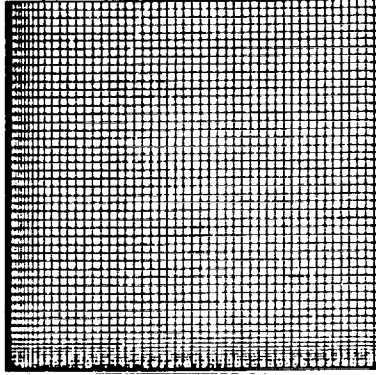
Initial grid
@ x = 0.688
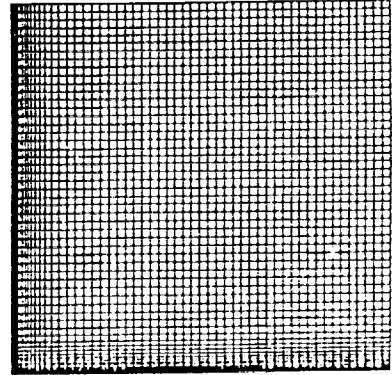
Initial grid
@ x = 1.000
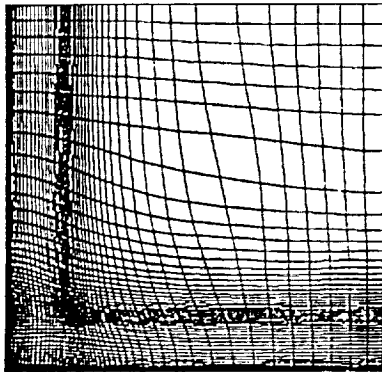
Adapted grid
@ x = 0.407
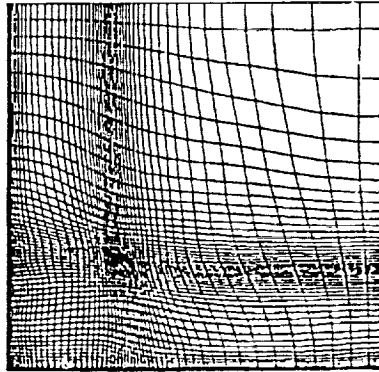
Adapted grid
@ x = 0.688
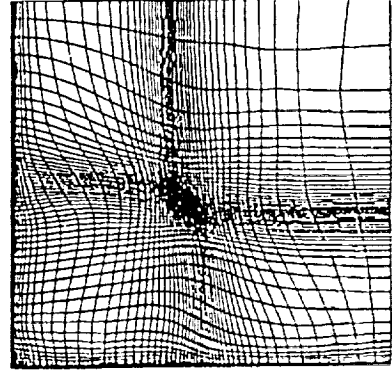
Adapted grid
@ x = 1.000

CORNER FLOW
SOLUTION COMPARISON
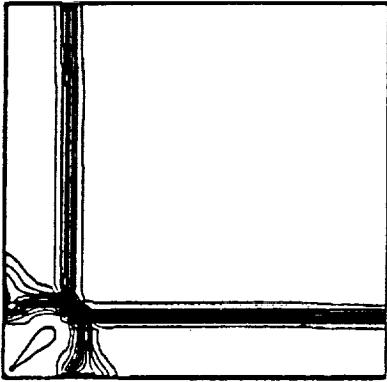
Density contours
on initial grid
@ x = 0.407
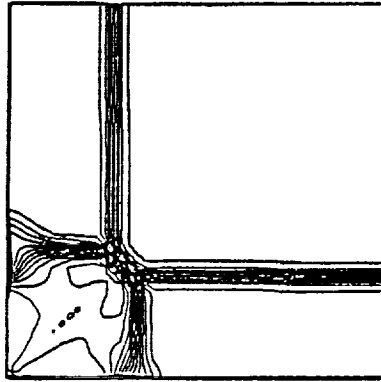
Density contours
on initial grid
@ x = 0.688
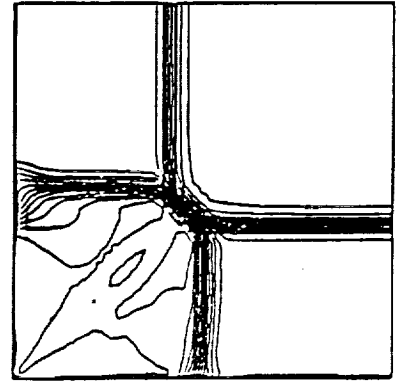
Density contours
on initial grid
@ x = 1.000
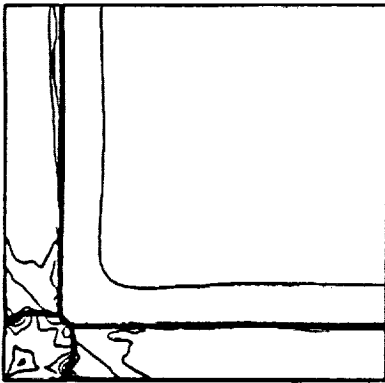
Density contours
on adapted grid
@ x = 0.407
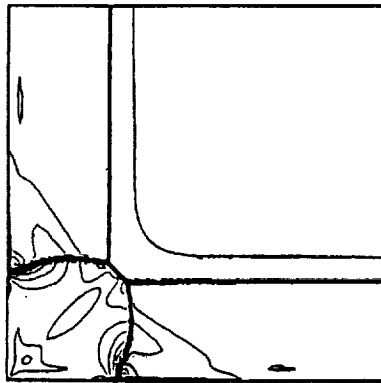
Density contours
on adapted grid
@ x = 0.688

Density contours
on adapted grid
@ x = 1.000

PART THREE

INTERACTIVE SOLUTION-ADAPTIVE CONTROL-NET GENERATION
AND NEW GRID COMPUTATION FROM THE ADAPTED CONTROL NET
(TURBO-AD)

OBJECTIVES

TO INTERACTIVELY PERFORM SOLUTION-ADAPTIVE GRID GENERATION ON
GRAPHICS WORKSTATIONS

TECHNICAL APPROACH/BENEFIT

TWO TECHNOLOGIES ARE COMBINED TO DEVELOP AN INTERACTIVE
SOLUTION-ADAPTIVE GRID GENERATION PROGRAM, TURBO-AD. ONE IS THE
SOLUTION-ADAPTION TECHNIQUE THAT USES THE PARMAETRIC MAPPING WITH
CONTROL SOURCES, THAT ARE DERIVED FROM THE FLOW SOLUTIONS. THE OTHER
IS THE CONTROL POINT FORM OF ALGEBRAIC GRID GENERATION TECHNIQUES THAT
PROVIDES PRECISE LOCAL CONTROL.

SIGNIFICANT COMPUTATIONAL SAVINGS ARE ATTAINED BY ADAPTING THE CONTROL
NET TO SOLUTIONS INSTEAD OF THE USUAL DIRECT GRID ADAPTATION TO
SOLUTIONS BECAUSE NUMBER OF CONTROL POINTS IS MUCH LESS THAN
THE NUMBER OF GRID POINTS. THIS NEW APPROACH CAN EASILY BE MORE
THAN TEN (10) TIMES FASTER IN 2-D THAN THE USUAL DIRECT GRID
ADAPTATION PROCEDURE

REFERENCE

Choo, Y.K., Henderson, T., "Interactive Solution-Adaptive Grid
Generation," NASA Technical Memonrandum, to be printed

135

SAMPLE CASE

A HIGH SPEED INLET WITH COMPRESSION AND EXPANSION

1. INITIAL GRID, INTIAL SOLUTION, AND INITIAL CONTROL NET

2. SOLUTION-ADAPTIVE CONTROL NET GENERATION

3. FINAL ADAPTED CONTROL NET, NEW GRID, AND NEW SOLUTION
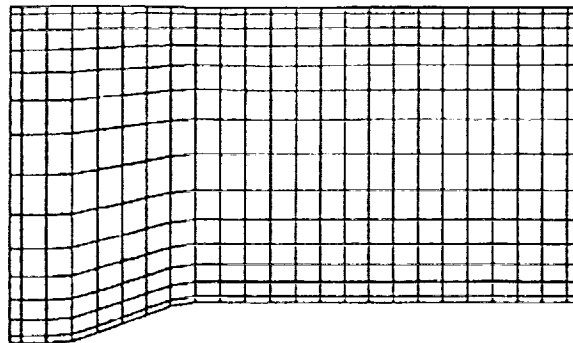
INITIAL CN
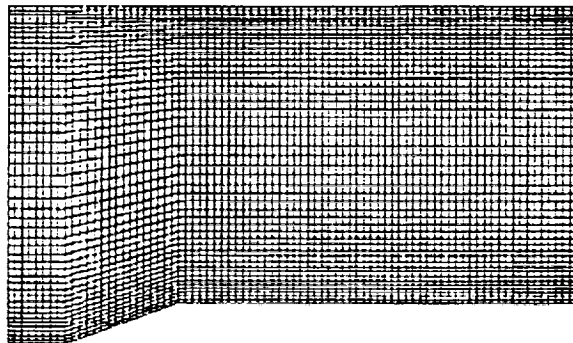
INITIAL
GRID
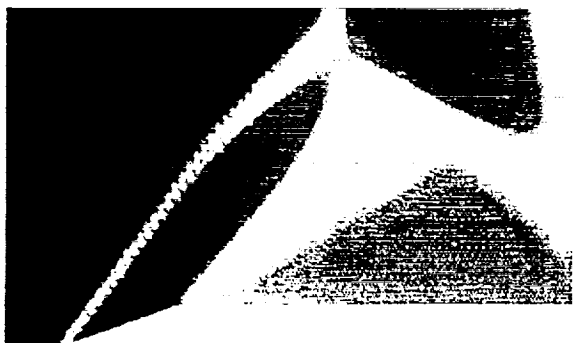
INITIAL
SOLUTION

SOLUTION-ADAPTIVE GRID GENERATION PROCEDURE
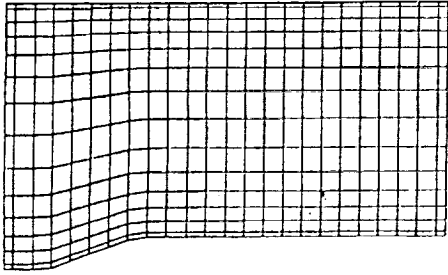
INITIAL CONTROL NET
IN PHYSICAL SPACE (x,y)
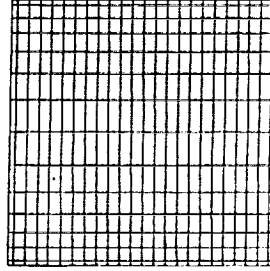
INITIAL CONTROL NET IN
A PARAMETRIC SPACE (s,t)
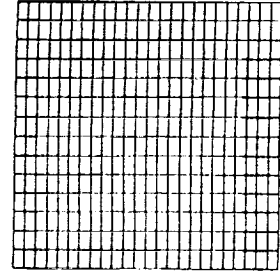
UNIFORM CONTROL NET IN
PARAMETRIC SPACE (u,v)
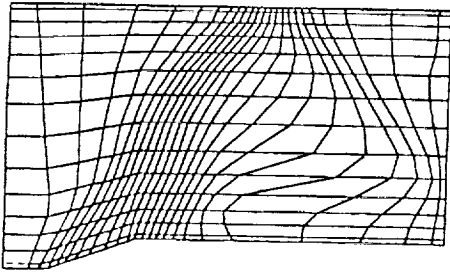
ADAPTED CONTROL NET IN
PHYSICAL SPACE (x,y)
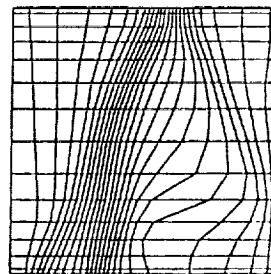
ADAPTED CONTROL NET IN
A PARAMETRIC SPACE (s,t)
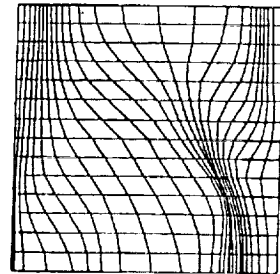
ADAPTED CONTROL NET IN
PARAMETRIC SPACE (u,v)

137
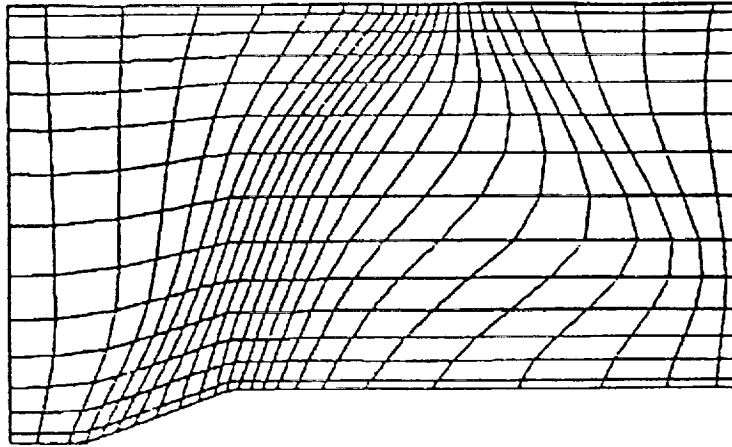
ADAPTED
CN
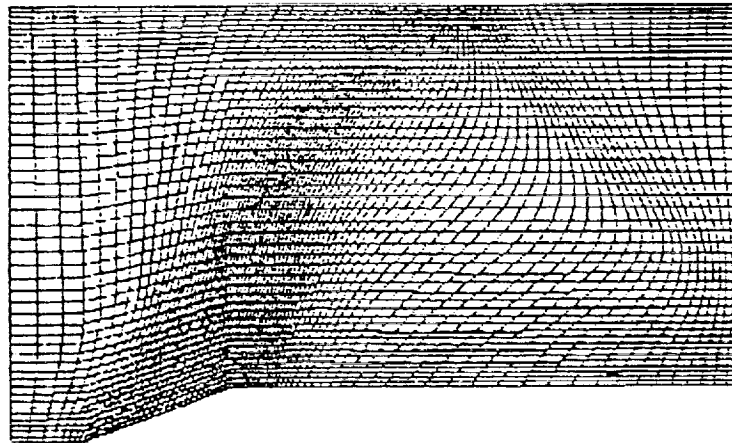
NEW
GRID

NEW
SOLUTION

138
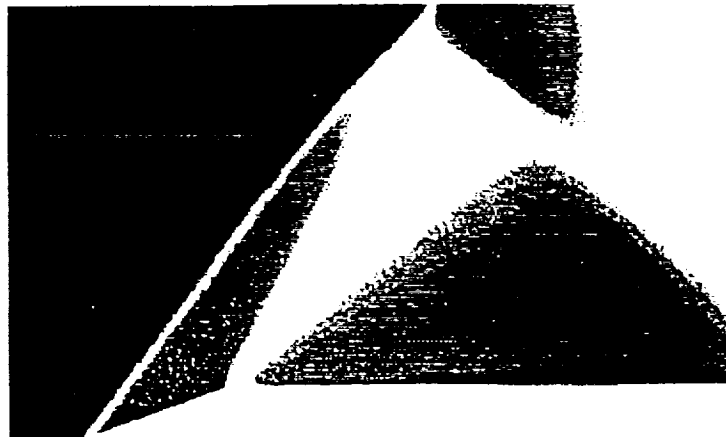
Summary of Discussion
Session III
Chair: Mark E.M. Stewart

The presentations in this session centered on techniques for simulation in complex geometries and adaptation. The discussion centered not so much on these techniques or their need, but where to use them and how to configure them. In particular, the discussion centered around three areas: the question of general techniques for grid generation versus custom software; important areas of application; and supporting technology.

First, the discussion of general grid generation techniques versus custom software was renewed from the perspective of problems and how they are solved. General approaches may lose sight of applications and result in long turn around times for practical problems. Custom software for an application may give rapid turn around, but be difficult to extend and result in duplication of work.

Second, given this emphasis on applications, what are the important applications at Lewis, in addition to turbomachinery blade rows, nozzles, ducts, and aircraft icing? Electromagnetics was suggested as an important area of application. However, it became clear that development and improvement of techniques are necessary to study in more detail the applications we are currently working on.

The third area of discussion was how supporting technology is influencing developments in grid generation and flow solutions. Topics discussed included graphics for unstructured grids, computer science and data structures, computer hardware and architectures, and numerical methods. The discussion centered on the need for graphics packages for use with unstructured meshes.

SESSION IV

NASA AND LEWIS ACTIVITIES

Chair: Austin L. Evans

NASA Surface-Modeling and Grid-Generation (SM/GG) Activities
by
Yung K. Choo
NASA Lewis Research Center

A NASA Steering Committee was formed to carry out the recommendations from

the NASA Workshop on FUTURE DIRECTIONS in SURFACE MODELING and GRID

GENERATION, held in December 1989. This committee consists of

representatives from three OAST Research Centers (ARC, LARC, LERC) and Code

RF at NASA Headquarters. Its function is to communicate and coordinate

within NASA the acquisition and distribution of geometry/grid generation

software/data, establish geometry data exchange standards, and interface

with other government, university, and industry efforts.

The NASA SM/GG Steering Committee plays an important role for the

coordination of NASA SM/GG activities. Coordination and cooperation of the

SM/GG activities among interested partners are increasing within Lewis as

well as within NASA. In May of 1990, the NASA SM/GG Steering Committee

formed a Subcommittee to develop a Geometry Data Exchange specifications in

order to provide rapid and accurate geometry data transfer for

computational aerospace design and to promote cooperation in the U.S. The

Subcommittee (ARC, LARC, LERC) completed to develop the geometry data

exchange specifications and distributed it to get industry review and

comments. Lewis projects such as the NPSS (Numerical Propulsion Systems

Simulations) benefit from this geometry data exchange standard. Two

speakers (Austin Evans on NPSS and Scott Thorp on the Subcommittee

activities) discuss them in detail.

NASA

SURFACE MODELING AND GRID GENERATION
IN COMPUTATIONAL AEROSPACE DESIGN
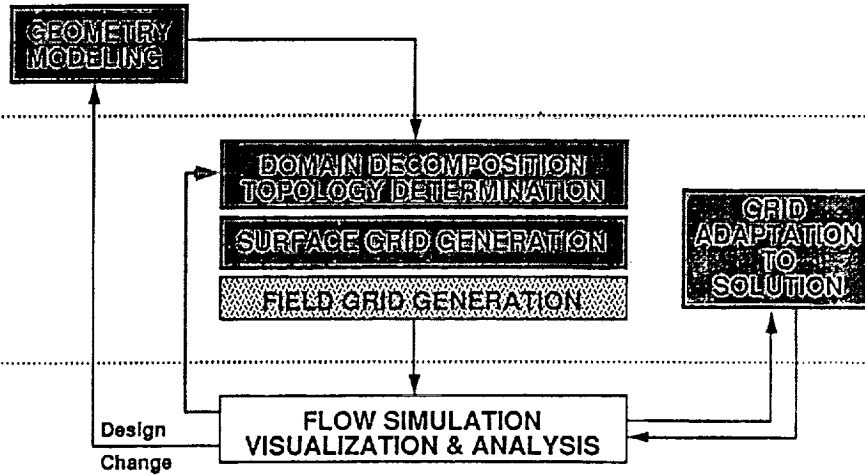
NASA

MANPOWER REQUIRED FOR CFD

	Percent of User's Time
Geometry Definition	25%
Surface/Field Grid Generation	35%
Compute Solution	20%
Viewing and Evaluation	20%

NASA WORKSHOP ON FUTURE DIRECTIONS IN SURFACE MODELING AND GRID GENERATION

DECEMBER 5-7, 1989

● **WORKSHOP ACTIVITIES**

   5 OVERVIEW SESSIONS, 4 GROUP DISCUSSIONS
   NASA PLANNING SESSION
   NASA/INDUSTRY/UNIVERSITY/DOD Participated

● **CONSENSUS**

   SM/GG ARE CRITICAL IN COMPUTATIONAL AEROSPACE
   DESIGN

   SM THROUGH CAD IS ADVANCED BUT NOT WELL
   COORDINATED WITH GRID GENERATION

   FASTER AND BETTER TOOLS ARE NEEDED

INTERNAL FLUID MECHANICS DIVISION

NASA WORKSHOP - CONTINUED

● **PROBLEMS**

   FEW ORGANIZATIONS HAVE THE LARGE, WELL-EQUIPPED, MULTI-DISCPLINARY
   EFFORTS REQUIRED TO PRODUCE USEFUL SM/GG TOOLS.

   ACTIVITIES IN THE U.S. LACK COORDINATION AND COOPERATION

● **RECOMMENDATIONS**

   ESTABLISH A NASA SM/GG STEERING COMMITTEE TO COORDINATE NASA'S
   SM/GG ACTIVITIES AND TO OVERSEE & ADVOCATE THESE ACTIVITIES

   IMPROVE COORDINATION OF U.S. ACTIVITIES THROUGH THE DEVELOPMENT
   AND IMPLEMENTATION OF AEROSPACE-GEOMETRY DATA EXCHANGE
   SPECIFICATIONS

   ESTABLISH A SM/GG FOCAL GROUP AT EACH OAET RESEARCH CENTERS
   (ARC, LARC, LERC)

INTERNAL FLUID MECHANICS DIVISION

SUBCOMMITTEE ON GEOMETRY DATA EXCHANGE STANDARD

GEOMETRY DATA TRANSFER PROBLEMS

- GEOMETRY DATA FOR ANALYSIS OFTEN REQUIRE MANY HOURS OF MANIPULATION TO ACHIEVE A FORMAT CAPABLE OF BEING UTILIZED

- THIS MODIFIED DATA SET OFTEN LOSES A LEVEL OF ACCURACY FROM THE ORIGINAL DATA

- U.S. ACTIVITIES NEED TO IMPROVE COORDINATION THROUGH THE DEVELOPMENT AND IMPLEMENTATION OF GEOMETRY DATA EXCHANGE SPECIFICATIONS

SUBCOMMITTEE - Continued

GOAL

TO PROVIDE RAPID AND ACCURATE GEOMETRY DATA TRANSFER WITH COMMON GEOMETRY REPRESENTATION FOR CFD RESEARCH

APPROACH

DEVELOP A NASA GEOMETRY DATA EXCHANGE SPECIFICATION BASED ON A SUBSET OF IGES TO PROVIDE RAPID ACCURATE GEOMETRY DATA TRANSFER

INTERACT WITH THE AEROSPACE INDUSTRY TO DISSEMINATE THE SPECIFICATION, OBTAIN COMMENTS/FEEDBACK AND TESTING ASSISTANCE

TEST AND IMPLEMENT THE SPECIFICATION AT THE THREE NASA OAET CENTERS

(IGES = Initial Graphics Exchange Specifications)

LEWIS ACTIVITIES
ON SURFACE MODELING AND GRID GENERATION

```
┌─────────────┐         ┌──────────────────────┐         ┌─────────────┐
│ INDUSTRY    │◄───────►│   LEWIS SM/GG        │◄───────►│ NASA HQS    │
│ UNIVERSITY  │         │   DEVELOPMENT TEAM   │         │ ARC, LARC   │
│ DOD         │         │                      │         │ MSFC, JSFC  │
└─────────────┘         └──────────────────────┘         └─────────────┘
                                   ▲
                                   │
                                   ▼
              ┌──────┬──────┬──────┬──────┬──────┬──────┐
              │ IFMD │ PSD  │ ITO  │ ED   │ CSD  │ SD   │
              └──────┴──────┴──────┴──────┴──────┴──────┘
```
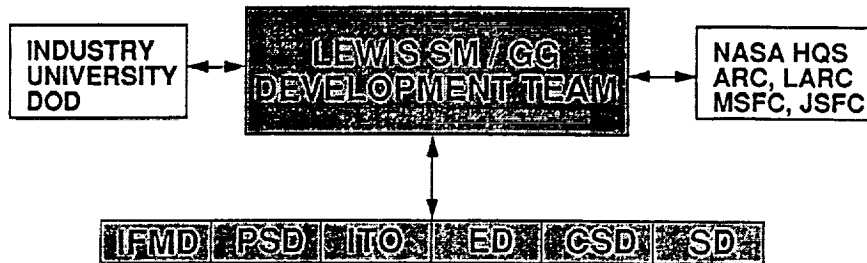
IFMD: INTERNAL FLUID MECHANICS DIVISION
PSD : POWER PROPULSION SYSTEMS DIVISION
ITO : INTERDISCIPLINARY TECHNOLOGY OFFICE
ED : ENGINEERING DIRECTORATE
CSD : COMPUTER SERVICES DIVISION
SD : STRUCTURES DIVISION

Lewis activities - Continued

LEWIS SM/GG DEVELOPMENT TEAM

GOALS

- DEVELOP SM/GG CAPABILITY FOR NUMERICALLY SIMULATING FLOWS IN PROPULSION SYSTEMS

APPROACH

- ENHANCE LEWIS' SM/GG CAPABILITY BY:
    - ASSESSING LEWIS SPECIFIC NEEDS
    - ENSURING USER INTERACTION AND TRAINING
    - UPGRADING TOOLS

- DEVELOP NEW CAPABILITY AS NEEDED

- MAINTAIN CLOSE INTERACTION WITH ARC, LARC, INDUSTRY, DOD,
    UNIVERSITY

- SUPPORT THE DEVELOPMENT OF THE NASA GEOMETRY EXCHANGE
    SPECIFICATION:
    - DEFINING LEWIS SPECIFIC GEOMETRIC TEST CASES AND TEST PLANS
    - DEFINING LEWIS SPECIFIC SOFTWARE REQUIREMENTS

SUMMARY REMARKS

● COORDINATION AND COOPERATION OF THE SM/GG ACTIVITIES AMONG
  INTERESTED PARTNERS ARE INCREASING. NASA SM/GG STEERING
  COMMITTEE PLAYS AN IMPORTANT ROLE FOR THE COORDINATION OF NASA
  SM/GG ACTIVITIES

● NASA GEOMETRY DATA SPECIFICATIONS WILL IMPROVE COORDINATION OF
  SM/GG ACTIVITIES IN THE U.S.

● LEWIS SPECIFICS

   - LEWIS ACTIVITIES ARE PRIMARILY COMPONENT ORIENTED

   - TURBOMACHINERY CODES USE SPECIALIZED GRID GENERATORS

   - GRIDGEN IS THE PRIMARY WORK HORSE FOR INLETS/NOZZLES/DUCTS

   - GRID ADAPTATION, QUALITY MEASURE, AUTOMATIC ZONING ARE
     IMPORTANT

   - NEED TO FOSTER MULTI-COMPONENT & MULTI-DISCIPLINARY EFFORT

Activities for Numerical Propulsion
Systems Simulation Program
by
Austin L. Evans
NASA Lewis Reseach Center

The Interdisciplinary Technology Office (ITO) has been tasked at LeRC with the responsibility of coordinating interdisciplinary research and technology programs (eg. IHPTET, HPCCP, and NPSS); establish and maintain interfaces between the various disciplines at LeRC, industry, government and academic organizations (eg. Software Environment Coordination Team and Surface Modeling and Grid Generation Development Team); and facilitating the exploitation of advances in the individual disciplinary efforts that have multidisciplinary implications.

The ITO works within the LeRC matrix organizational structure to form multidisciplinary coordination and development teams to deal with interdisciplinary issues. One such team on SM&GG was formed. Here, ITO coordinates SM&GG efforts, helps identify a challenging problem to serve as a research focal point, and helps the team develop an approach for the research. Funding is primarily provided by leveraging off of existing programs with ITO providing funds to serve as "seed" money and to ensure interdisciplinary closure.

The effort to identify and/or coordinate the development of appropriate standards for NPSS is a key technical issue for the program. The driver here is the requirement that the individual discipline analysis modules within NPSS be able to freely exchange analysis results and geometry data. It will be necessary to incorporate specific standards within NPSS to allow for this exchange of data.

ACTIVITIES FOR NPSS

- **BACKGROUND**

- **ITO-NPSS INVOLVEMENT**

- **DEVELOPMENT TEAM ACTIVITIES**

AUSTIN L. EVANS
November 14, 1991

NUMERICAL PROPULSION SYSTEM SIMULATION (N.P.S.S.)

VALIDATED MODELS

- FLUID MECHANICS
- HEAT TRANSFER
- COMBUSTION
- STRUCTURAL
  MECHANICS
- MATERIALS
- CONTROLS
- AEROELASTICITY

N.P.S.S.
INTEGRATED INTERDISCIPLINARY
ANALYSIS AND ASSESSMENT OF
PROPULSION SYSTEMS

RAPID COMPUTATION
WITH KNOWN
ACCURACY FOR

- PERFORMANCE
- STABILITY
- DURABILITY
- LIFE

HIGH PERFORMANCE COMPUTING
- PARALLEL PROCESSING
- EXPERT SYSTEMS
- INTERACTIVE 3-D GRAPHICS
- NETWORKS
- DATA BASE MANAGEMENT SYSTEMS
- AUTOMATED VIDEO DISPLAYS

A NUMERICAL TEST CELL
FOR AEROSPACE PROPULSION SYSTEMS

CD-90-47020

152

The Interdisciplinary Technology Office (ITO) has been tasked at LeRC with the responsibility of coordinating interdisciplinary research and technology programs (eg. IHPTET, HPCCP, and NPSS); establish and maintain interfaces between the various disciplines at LeRC, industry, government and academic organizations (eg. Software Environment Coordination Team and Surface Modeling and Grid Generation Development Team); and facilitating the exploitation of advances in the individual disciplinary efforts that have multidisciplinary implications.

PRELIMINARY SET OF NPSS ANALYSIS MODULES FOR EEE ENGINE

	AERODYNAMICS	STRUCTURES	THERMAL	CONTROLS
LEVEL 1	WWEP	WATE-2		
	ASCYNT-INLET			
LEVEL 2	DISTEN	WATE-2+		LAPINT
LEVEL 3	MBLOCK-FLOWPATH	STR/AERO	SINDA?	CONT.
	PANEL CODE-SOURCE TERMS			
	FAN	FAN		
	COMPRESSOR	COMPRESSOR		
	TURBINE	TURBINE		
	REAC 2D-COMBUSTOR	COMBUSTOR		
LEVEL 4	ADPAC-FAN	NASTRAN-FAN	SINDA-FAN	CONT.
	MSTAGE-COMPRESSOR	COMPRESSOR	COMPRESSOR	
	REACT3D-COMBUSTOR	COMBUSTOR	COMBUSTOR?	
	MSTAGE-TURBINE	TURBINE	TURBINE	
	PARC-NOZZLE	NOZZLE	NOZZLE	
LEVEL 5	TURBO-FAN	NASTRAN-FAN	SINDA-FAN	CONT.
	COMPRESSOR	COMPRESSOR	COMPRESSOR	
	TURBINE	TURBINE	TURBINE	
	KIVA-COMBUSTOR	COMBUSTOR	COMBUSTOR?	
	PARC-NOZZLE	NOZZLE	NOZZLE	

GEOMETRY DATA FLOW

COMPUTER AIDED DESIGN SYSTEM (CAD) → COMPUTER AIDED MANUFACTORING SYSTEM (CAM) → NC TAPE

INDUSTRY LaRC ARC LERC CFD

IGES FILE TRANSLATOR

NASA IGES FILTER

NPSS SYSTEM EXECUTIVE

ENGINE GEOMETRY DATA FILE
HOT SHAPE   COLD SHAPE

COMPONENT GEOMETRY DATA FILE
HOT SHAPE
COLD SHAPE

CFD FILTER
CFD GRID GENERATOR B.C.&I.V.
CFD FLOW SOLVER

STRUCTURES FILTER
STRUCTURES GRID GENERATOR B.C.&INITIAL V
STRUCTURES SOLVER

THERMAL FILTER
THERMAL GRID GENERATOR B.C.&I. VALUES
THERMAL SOLVER

154

AERODYNAMIC RESEARCH AND DESIGN
INTEGRATED CAE/CAD/CAM CAPABILITIES

INTERDISCIPLINARY DEVELOPMENT TEAMS AND INTERACTIONS

```
┌─────────────────────────────────────────────────────┐   ┌───────────────────────────┐
│ ITO SURFACE MODELING AND GRID GENERATION DEVELOPMENT  │   │ NASA SURFACE MODELING AND │
│ TEAM    AUSTIN EVANS   (YUNG CHOO)                     │   │ GRID GENERATION STEERING  │
│                                                       │   │ COMMITTEE                 │
│ YUNG CHOO       JOE OGONEK      DAVE MILLER            │   │ (PAM RICHARDSON-HQ-CODE RF)│
│ MARK STEWART    DON SOSOKA      TIM BEACH              │   ├──────┬───────────┬────────┤
│ MARK POTAPCZUK  DHARAT GONI/MSU VPI ACSYNTH REP        │   │ LERC │ AMES      │ LANGLEY│
│ SCOTT THORP     MATT MELIS      AUSTIN EVANS           │ ──│Y.CHOO│W.VAN DALSEN│R.SMITH│
└─────────────────────────────────────────────────────┘   │D.HUFF│M.BLAKE-NAS│        │
                                                           └──────┴───────────┴────────┘
   ┌────────────┐  ┌────────────┐  ┌────────────────┐       ┌───────────────────────────┐
   │ LERC GRID  │  │ LERC       │  │ LERC GEOMETRY  │       │ NASA GEOMETRY EXCHANGE    │
   │ GENER-     │  │ INDUST-    │  │ EXCHANGE       │       │ STANDARDS SUBCOMMITTEE    │
   │ ATION      │  │ RIAL       │  │ STANDARDS      │       │ (MATT BLAKE-AMES/NAS)     │
   │ WORKSHOP   │  │ LIAISON    │  │ SUBCOMMITTEE   │       ├──────┬───────┬────────────┤
   │            │  │            │  │                │       │ LERC │ AMES  │ LANGLEY    │
   │ YUNG CHOO  │  │ AUSTIN     │  │ AUSTIN EVANS   │ ──────│S.THORP│M.BLAKE│P.KERR     │
   │ AUSTIN     │  │ EVANS      │  │ SCOTT THORP    │       └──────┴───────┴────────────┘
   │ EVANS      │  │ YUNG CHOO  │  │ HAROLD RENKEL  │
   │ DALE       │  │ SCOTT THORP│  │ KEVIN DOWNEY   │
   │ HOPKINS    │  │ DAVE MILLER│  │ MARK STEWART   │
   │ DENNIS HUFF│  └────────────┘  │ TIM BEACH      │
   │ DANNY HWANG│                  │ PHIL JORGENSON │
   │ CHANTHY IEK│                  └────────────────┘
   │ MARK KERWIN│
   │ MATT MELIS │
   │ DAVE MILLER│
   │ MARK       │
   │ POTAPCZUK  │
   │ AARON      │
   │ SNYDER     │
   │ DON SOSOKA │
   │ MARK       │
   │ STEWART    │
   │ C. STEFFEN │
   │ RUSS CLAUS │
   └────────────┘
```

DEVELOPMENT TEAM PROGRESS

- ## DRAFT OF NASA-IGES GEOMETRY EXCHANGE SPECIFICATION GENERATED

  - PROVIDE RAPID AND ACCURATE EXCHANGE OF GEOMETRY DATA BETWEEN DESIGN AND INTERDISCIPLINARY ANALYSIS SOFTWARE
  - USES SUBSET OF IGES ENTITIES (B-SPLINES AND NURBS)
  - UNDER REVIEW: LERC INDUSTRIAL REVIEW 9/10/91, FINAL NASA/INDUSTRY REVIEW EARLY 1992.

- ## DEVELOPING PLANS FOR CONVERTING EXISTING CODES TO UTILIZE SPECIFICATION

- ## TRAINING AND INFORMATION EXCHANGES

  - PDES/IGES MEETING (WEEK OF 10/20/91)
  - LERC/MARSHALL INITIAL COORDINATION MEETING (10/28/91)
  - LERC WORKSHOP ON GRID GENERATION (11/14/91)
  - NURBS COURSE FOR LERC DESIGNERS/ANALYSTS (JANUARY 92)

PROBLEM DEFINITION
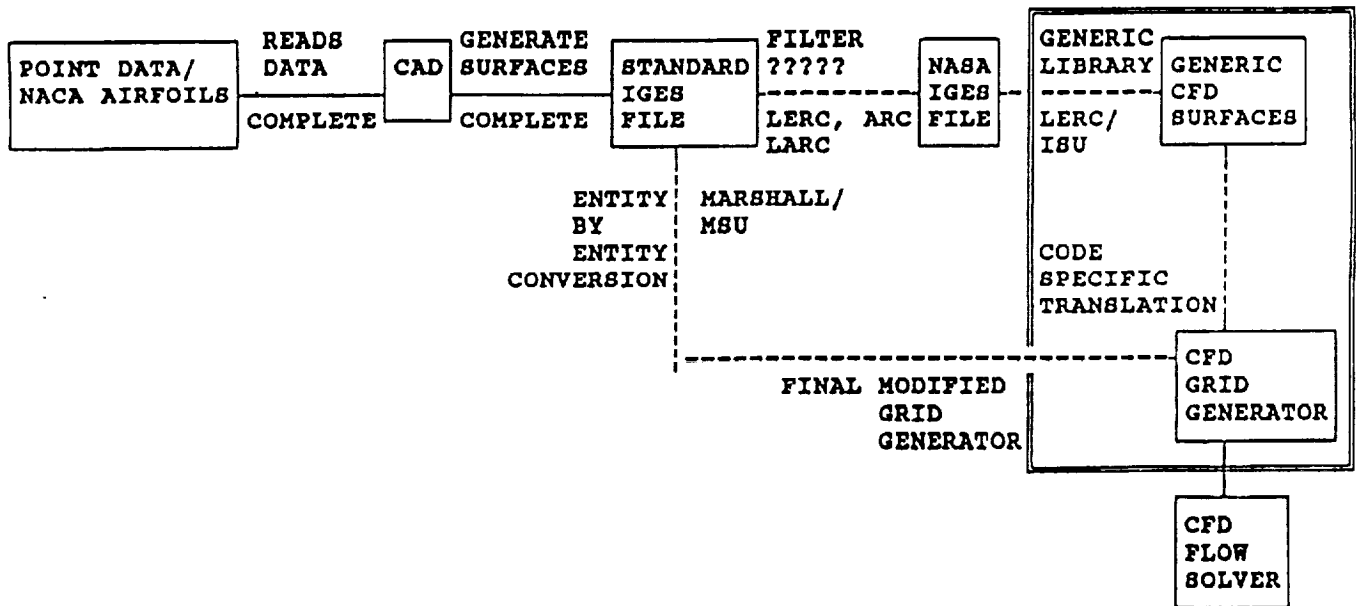
Integrating Aerodynamic Surface Modeling
for Computational Fluid Dynamics with
Computer Aided Structural Analysis, Design, and Manufacturing

by
Scott A. Thorp
NASA Lewis Research Center

ABSTRACT

This presentation will discuss the development of a NASA Geometry Exchange Specification (NASA-IGES) for transferring Aerodynamic surface geometry between LeRC CAE/CAD/CAM systems and grid generation software used for CFD research. The proposed specification is based on a subset of the Initial Graphics Exchange Specification (IGES).

The advantages of a standardized methods for describing and transferring surface data for aerodynamic and structural analysis will be discussed. Problems encountered in CAE/CAD/CAM when point data is used as the method of geometry transfer will be presented. The discussion will include problems associated with aero surface regeneration for design/fabrication and the resulting need for a data transfer standard. Some problems for discussion include: requirements for tighter surface profile tolerances, multiple data formats utilized for point data transfer, inconsistent surface point fidelity, lack of common curve and surface mathematics for fitting and extracting points.

The presentation will include discussion of how the NASA-IGES standard will accommodate improved CAD inspection methods and reverse engineering techniques currently being developed. These technologies will allow aerodynamic evaluation of manufacturing requirements through sensitivity analysis. Also, the reverse engineering aspects will allow analyses to be preformed on the "as-built" geometry to assure geometric integrity when correlating experimental test data to analytical predictions.

NASA   LEWIS RESEARCH CENTER

CIM
COMPUTER INTEGRATED MANUFACTURING
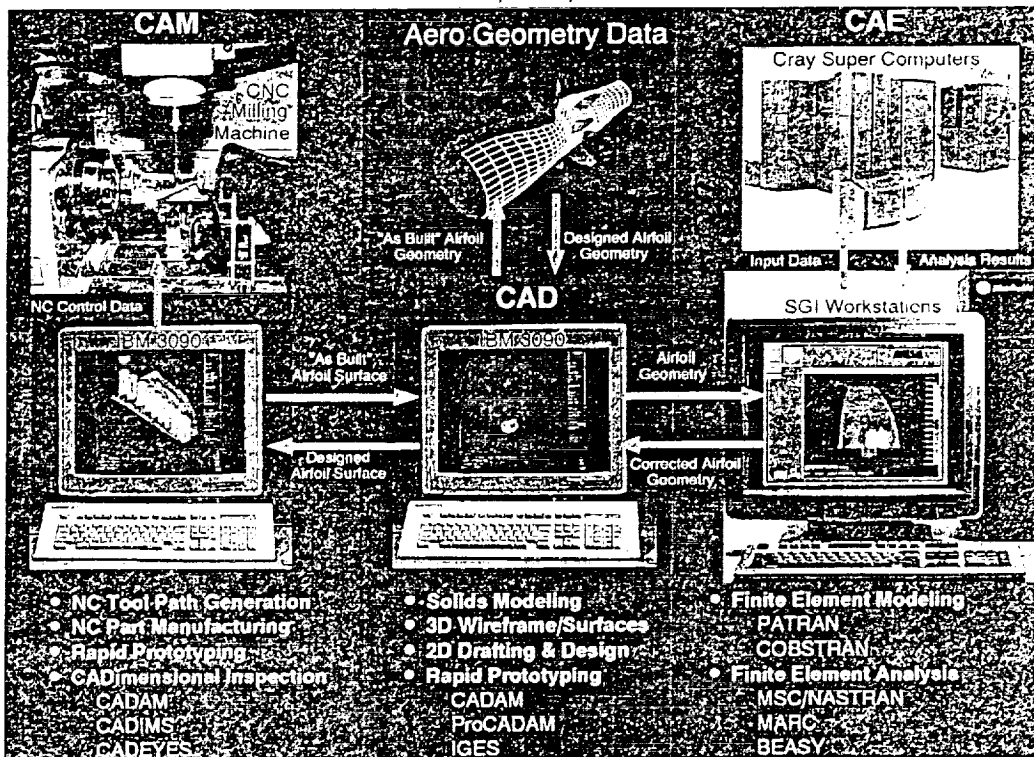
CAD
COMPUTER AIDED DESIGN

CAE
COMPUTER AIDED ENGINEERING

CAM
COMPUTER AIDED MANUFACTURING

CAQA
COMPUTER AIDED QUALITY ASSURANCE

AERODYNAMIC RESEARCH AND DESIGN
INTEGRATED CAE/CAD/CAM CAPABILITIES

160

NASA LEWIS RESEARCH CENTER

GEOMETRY DATA TRANSFER METHODS

- CUSTOM POINT DATA FORMATS

- IGES - INITIAL GRAPHICS EXCHANGE SPECIFICATION
  NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY (NIST)

- PDES - PRODUCT DATA EXCHANGE USING STEP
  NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY (NIST)

- STEP - STANDARD FOR THE EXCHANGE OF PRODUCT DATA
  INTERNATIONAL ORGANIZATION OF STANDARIZATION (ISO)

NASA LEWIS RESEARCH CENTER

GEOMETRY DATA TRANSFER METHODS & PROBLEMS
MULTIPLE CUSTOM DATA FORMATS & MULTIPLE DEFINITIONS

FORMATS: POINT DATA

| GENERAL ELECTRIC | PRATT & WHITNEY | GARRETT | ALLISON | LEWIS TURBO 1 | LEWIS TURBO 2 | PROPELLER |

CAE/CAD/CAM INTERFACE PROGRAM

CAE/CAD/CAM DATA BASE
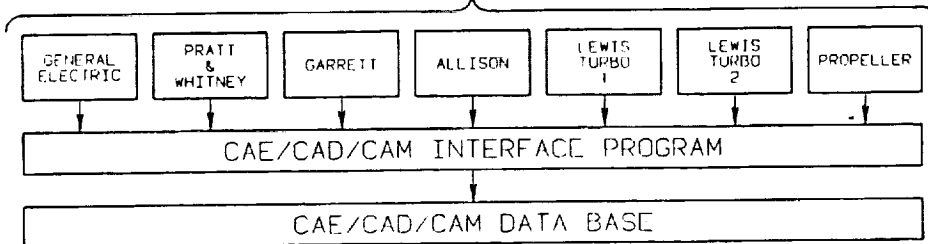
DEFINITIONS:

PLANAR CARTESIAN COORDINATE   STREAM LINES   CYLINDRICAL   NON-DIMENSIONAL

- INDIVIDUAL, UNIDIRECTIONAL INTERFACE PROGRAMS
- NO STANDARD GEOMETRY TRANSFER METHOD

NASA   LEWIS RESEARCH CENTER

GEOMETRY DATA TRANSFER METHODS & PROBLEMS
DUPLICATION OF MODELING EFFORTS

```
                    ┌──────┐
                    │POINT │
                    │DATA  │
                    └──────┘
```

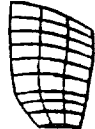| AERO SURFACE MODELING | STRUCTURAL SURFACE MODELING | CAD SURFACE MODELING | CAM SURFACE MODELING | INSPECTION |

CAE   CAD   CAM   CAQC
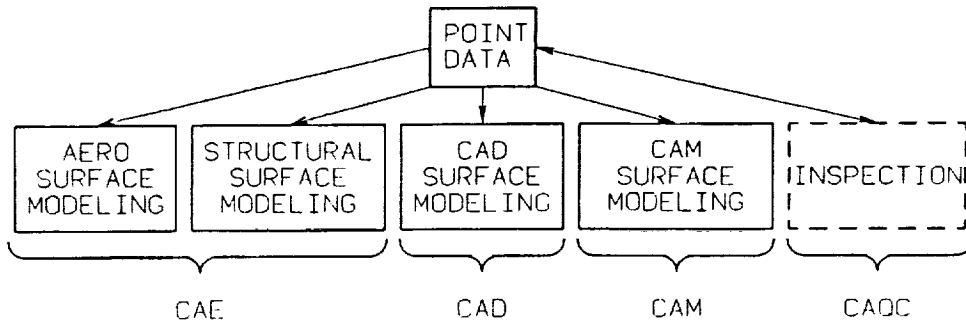
- 60%-70% OF THE TIME REQUIRED BY EACH DISIPLINE IS SPENT DEFINING GEOMETERY/SURFACE MODELING

- SURFACE MODELING EFFORTS DUPLICATED BY EACH DISIPLINE

NASA   LEWIS RESEARCH CENTER

GEOMETRY DATA TRANSFER METHODS & PROBLEMS
INCONSISTENT CURVE & SURFACE MATH

```
                    ┌──────┐
                    │POINT │
                    │DATA  │
                    └──────┘
```

BEZIER   CUBIC POLYNOMIAL   HERMITE   B-SPLINE

| AERO GEOMETRY | ≠ | STRUCTURE GEOMETRY | ≠ | CAD GEOMETRY | ≠ | CAM GEOMETRY | → | TESTING |
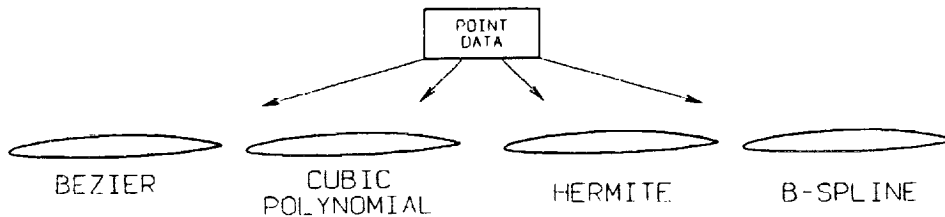
ANALYSIS ≠ DESIGN ≠ HARDWARE ≠ TEST RESULTS

- A COMMON SET OF CURVE AND SURFACE MATH IS NOT USED
- MORE COMPLEX GEOMETRY
- TIGHTER TOLERANCES
- LOSS OF ACCURACY DUE TO CONVERSIONS

GEOMETRY DATA TRANSFER METHODS & PROBLEMS

NO SURFACE VECTOR DATA PROVIDED IN POINT FORMATS
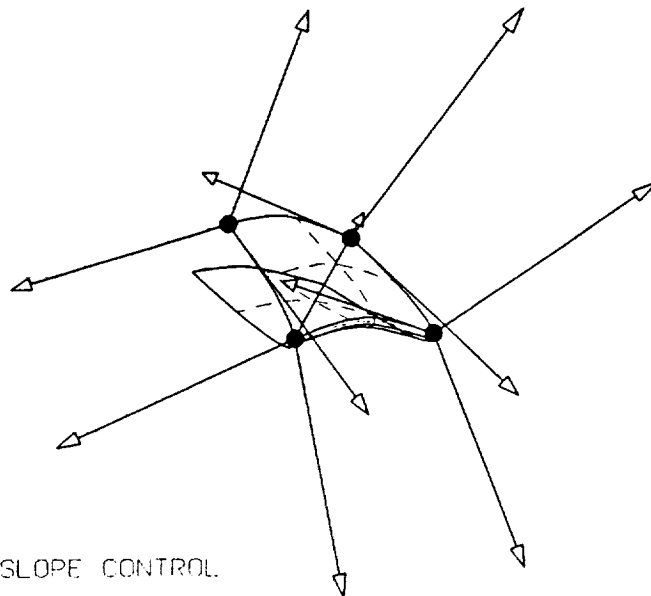
- I.0 SLOPE CONTROL.

GEOMETRY DATA TRANSFER METHODS & PROBLEMS

CHORDWISE WAVINESS
INACCURATE COORD. OR CURVE FITTING

MISMATCHED COORD DISCONTINUITIES
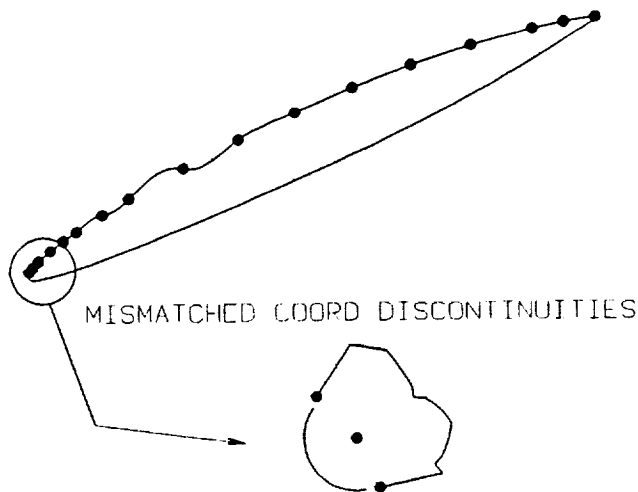
- GEOMETRY ALTERED AND SMOOTHED

NASA   LEWIS RESEARCH CENTER

GEOMETRY DATA TRANSFER METHODS & PROBLEMS
MISMATCHED SLOPE DISCONTUITIES

NASA   LEWIS RESEARCH CENTER
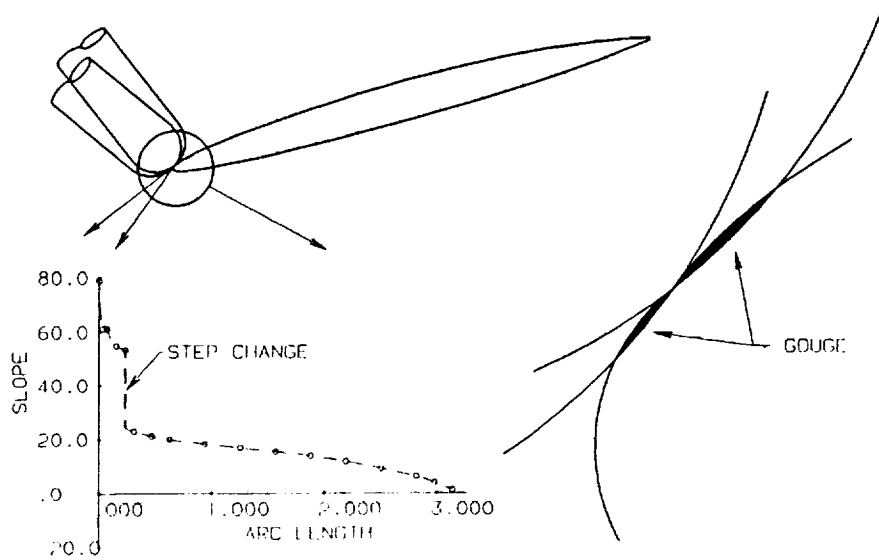
GEOMETRY DATA TRANSFER METHODS & PROBLEMS
SPANWISE WAVINESS
INACCURATE SECTION ORIENTATION OR CURVE FITTING

164
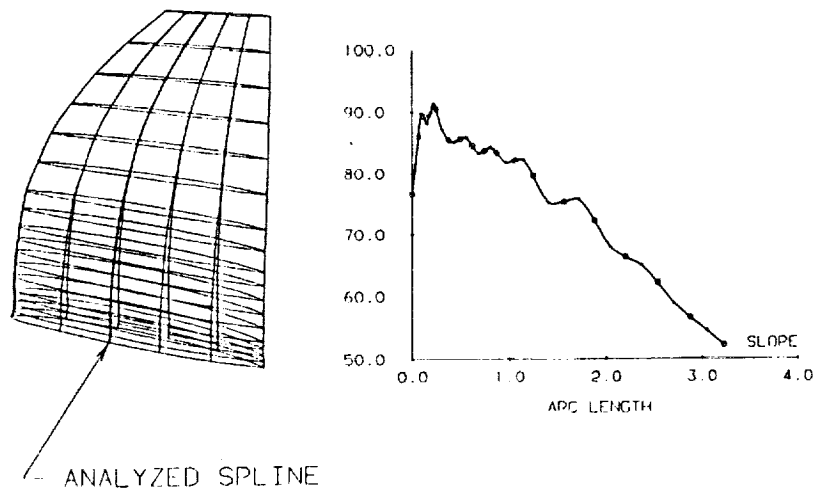
NASA-IGES
GEOMETRY DATA EXCHANGE
SPECIFICATION

NASA LEWIS RESEARCH CENTER

GOAL

- PROVIDE A RAPID AND ACCURATE EXCHANGE OF GEOMETRY DATA BETWEEN DESIGN AND ANALYSIS SOFTWARE

- ADDITIONALLY:

    INSURE LOW IMPACT ON INDUSTRY

    INCLUDE PROVISIONS FOR INTERDISCIPLINARY RESEARCH

    PROVIDE CAPABILITY FOR FUTURE ENHANCEMENTS

165

NASA

WHAT IS IGES ??

- IGES, THE INTIAL GRAPHICS EXCHANGE SPECIFICATION, DEFINES A NEUTRAL DATA FORMAT THAT ALLOWS FOR THE DIGITAL EXCHANGE OF INFORMATION AMONG COMPUTER-AIDED DESIGN (CAD) SYSTEMS.

- THERE ARE FOUR CLASSES OF IGES ENITES:

    CURVE AND SURFACE GEOMETRY ENTITES

    CONSTRUCTIVE SOLID GEOMETRY (CSG) ENTITES

    ANNOTATION ENTITIES

    STRUCTURE ENTITIES

NASA

NASA-IGES
SPECIFICATION FEATURES

- BASED ON A SUBSET OF AN EXISTING STANDARD FOR EASE OF INTEGRATION WITH INDUSTRY

- ALL GEOMETRY REPRESENTED BY A MINIMUM NUMBER OF ENTITIES PROVIDING EASIER INTERFACE DEVELOPMENT

- FULL NURB CURVE AND SURFACE DEFINITIONS CAN BE TRANSFERED WITH THE SPECIFICATION

- SPECIFIES BI-DIRECTIONAL DATA TRANSFER SO CFD ANALYSIS CAN EFFECT DESIGN

- PROVIDES FUTURE SOFTWARE DEVELOPERS WITH A STANDARD GEOMETRY DATA FORMAT TO RELY ON

NASA — LEWIS RESEARCH CENTER

INTEGRATED GEOMETRY DATA TRANSFER
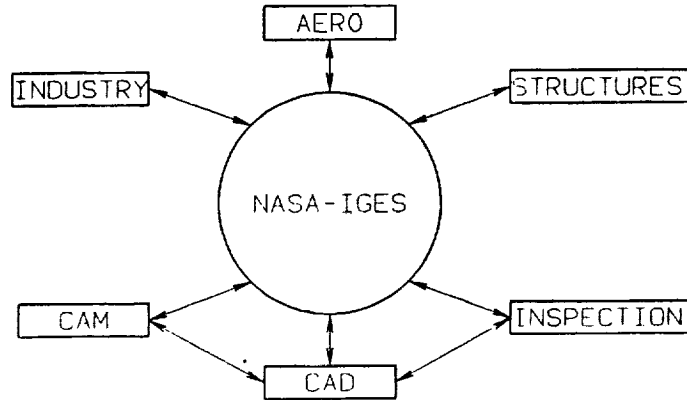
- BI-DIRECTIONAL INTERFACE TO OTHER DISCIPLINES

- CONSISTENT CURVE AND SURFACE MATH WITH A COMMON FORMAT

- ELIMINATES DUPLICATION OF SURFACE MODELING EFFORTS

- DEVELOP TOOLS TO CREATE SMOOTH CONTINOUS SURFACE
  THE FIRST TIME, BASE ON THE STANDARD

NASA — LEWIS RESEARCH CENTER

REVERSE ENGINEERING
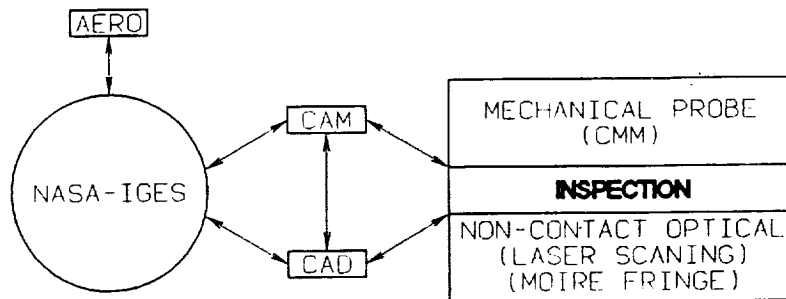
- IMPROVED INSPECTION

- EVALUATION AND COMPARISON OF "AS BUILT" TO
  DESIGN GEOMETRY

- EVALUATION OF SURFACE TOLERANCE SENSITIVITY

- DEVELOPMENT OF IMPROVED SURFACE TOLERANCE METHODS

- SURFACE COMPENSATION FOR IMPROVED QUALITY

NASA   **LEWIS RESEARCH CENTER**

CONCLUSIONS

- IMPROVES QUALITY AND ACCURACY

- REDUCES DESIGN CYCLE TIME

- REDUCES ENGINEERING AND DESIGN COSTS

- ALLOWS COMMON TOOLS TO BE DEVELOPED AND SHARED

- PROVIDES A STANDARD FOR FUTURE SOFTWARE
  DEVELOPMENT

- INTEGRATES ENGINEERING ANALYSIS, DESIGN,
  MANUFACURING, AND INSPECTION WITH A
  CONSISTENT GEOMETRIC METHODOLOGY

Summary of Presentations and Comments
Session IV
Chair: Austin L. Evans

Session IV consisted of presentations on the activities of the NASA Surface-Modeling and Grid-Generation Steering Committee, Numerical Propulsion Systems Simulations (NPSS), and NASA Geometry Data Exchange Specifications. The geometry data exchange standard was developed by a NASA subcommittee to provide rapid and accurate geometry data transfer, and to promote cooperation among interested parties in the United States.

Surface modeling through Computer Aided Design (CAD) is quite advanced, but not well coordinated with grid generation. The interface between CAD and grid generation will need to be improved through the use of the NASA geometry data exchange standard if CAD is to become a useful design tool as well as an analysis tool. Lewis efforts have been towards component-oriented research and analysis. For multicomponent and multidisciplinary efforts for propulsion systems studies, Lewis needs better surface-modeling and grid-generation tools, human skills, and planned teamwork since these efforts require expertise from several discipline areas.

Lewis grid generation capability has been substantially enhanced since the in-house survey conducted in October of 1989. Coordination and interaction among interested parties within Lewis, as well as within NASA, are important contributing factors. It is clear that geometry modeling and grid generation techniques need to be extended across disciplines. The NPSS program is serving as a focal point for this multidisciplinary extension at Lewis.

REPORT DOCUMENTATION PAGE

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	April 1992	Conference Publication

4. TITLE AND SUBTITLE

Workshop on Grid Generation and Related Areas

5. FUNDING NUMBERS

WU-505-62-52

6. AUTHOR(S)

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

National Aeronautics and Space Administration
Lewis Research Center
Cleveland, Ohio 44135-3191

8. PERFORMING ORGANIZATION REPORT NUMBER

E-6823

9. SPONSORING/MONITORING AGENCY NAMES(S) AND ADDRESS(ES)

National Aeronautics and Space Administration
Washington, D.C. 20546-0001

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

NASA CP-10089

11. SUPPLEMENTARY NOTES

Responsible person, Yung Choo, (216) 433-5868.

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Unclassified - Unlimited
Subject Categories 02 and 07

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

This publication is a collection of presentations given at the Workshop on Grid Generation and Related Areas held at NASA Lewis Research Center, Cleveland, Ohio, November 14-15, 1991. The purpose of this workshop was to assemble engineers and scientists from Lewis Research Center (including ICOMP, SSC, and OAI) working on grid generation for computational fluid dynamics (CFD), surface modeling, and related areas. Specifically, the objectives were (1) to provide an informal forum on grid generation and related topics, (2) to assess user experience, skills, current activities, and available tools, (3) to identify needs, and (4) to help promote synergy among engineers and scientists working in this subject area. The workshop consisted of four sessions representative of grid generation and surface modeling research and application within NASA Lewis Research Center. Each session contained presentations and an open discussion period. A summary of discussions for each session has been included in this publication.

14. SUBJECT TERMS

Grid generation; Computational fluid dynamics; Propulsion

15. NUMBER OF PAGES

183

16. PRICE CODE

A09

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	