Mixed Element Type Unstructured Grid Generation for Viscous Flow Applications

by

David L. Marcum and J. Adam Gaither

MSSU–COE–ERC–00–06

Computational Fluid Dynamics Laboratory

NSF Engineering Research Center

July 2000

Mississippi State University
P.O. Box 9627
Mississippi State, MS 39762

MIXED ELEMENT TYPE UNSTRUCTURED GRID GENERATION
FOR VISCOUS FLOW APPLICATIONS

David L. Marcum* and J. Adam Gaither[†]

NSF Engineering Research Center for Computational Field Simulation
Mississippi State University

Abstract

A procedure is presented for efficient generation of high–quality unstructured grids suitable for CFD simulation of high Reynolds number viscous flow fields. Layers of anisotropic elements are generated by advancing along prescribed normals from solid boundaries. The points are generated such that either pentahedral or tetrahedral elements with an implied connectivity can be be directly recovered. As points are generated they are temporarily attached to a volume triangulation of the boundary points. This triangulation allows efficient local search algorithms to be used when checking merging layers. The existing advancing–front/local–reconnection procedure is used to generate isotropic elements outside of the anisotropic region. Results are presented for a variety of applications. The results demonstrate that high–quality anisotropic unstructured grids can be efficiently and consistently generated for complex configurations.

Introduction

Unstructured grid technology is a promising approach offering geometric flexibility for handling of both complex geometry and physics. As such, it can provide a powerful capability for accurately and efficiently computing complex flow fields about realistic aerospace configurations. Several unstructured grid generation and flow solver procedures have been developed and successfully demonstrated for inviscid flow about complex configurations. For isotropic elements, existing procedures are robust and capable of generating high–quality grids efficiently. For anisotropic elements in viscous flow applications, further improvements in efficiency, robustness, and quality of unstructured grid generation procedures are needed. In this paper, a grid generation procedure is presented which offers the potential for overall improved performance and quality.

The most common approach used to generate anisotropic unstructured grids is to use a layered approach and generate points along normals from solid boundaries. Unstructured grid generation for viscous

*Professor, Mechanical Engineering Department,
Senior Member AIAA.
[†]Research Assistant II, Engineering Research Center.

applications have been developed using a modified advancing–front method by Hassan, et al[1], a semi–structured approach by Lohner[2], advancing–normal by Marcum[3], and advancing–layers by Pirzadeh[4]. Hybrid methods for prismatic/tetrahedral grid generation have been developed by Kallinderis[5] and Sharov and Nakahasi[6]. While all of these methods differ in how points and elements are generated, they all produce very structured and aligned elements adjacent to solid boundaries and use isotropic tetrahedral elements outside of the anisotropic or boundary–layer region. Use of prismatic elements within the anisotropic region can reduce subsequent memory and CPU requirements for the flow solver without any loss of accuracy.

The goal of the present work is to modify the existing advancing–normal and advancing–front local–reconnection method[3] for efficient generation of high–quality, mixed element type unstructured grids for viscous flow applications. While this method has many advantages, the local–reconnection process is unable to remove trapped sliver elements. Such elements can be formed between prismatic groups of elements as shown in Fig. 1. Local–reconnection or connectivity optimization can not remove these trapped slivers as they represent a local minimum state that cannot be removed without reconnecting a potentially very large number of elements. An alternative is to use the same point placement strategy and discard the connectivity in favor of a hybrid approach. With a hybrid approach the element connectivity is directly implied. Also, the elements can be recovered as either all tetrahedra or a mixture of five and six node pentahedra and tetrahedra. This combined approach retains most of the generality of the original and is very efficient.

Unstructured Grid Generation Procedure

The approach used in the present work uses the advancing–normal point placement algorithm[3] to generate points within the anisotropic region. The advancing–front/ local–reconnection (AFLR) procedure[7,8] is used to generate tetrahedral elements in the isotropic region. The basic steps in the overall procedure are listed below.

1) Generate a boundary surface grid.
2) Generate a volume triangulation of the boundary points. No boundary recovery is required.

3) Create new points using advancing–normal point placement.

4) Attach new points to the existing triangulation for searching and checking.

5) Create a new boundary surface grid using the inflated surface from advancing–normal point placement.

6) Use AFLR to generate an isotropic tetrahedral element grid for the remaining regions.

7) Merge anisotropic and isotropic regions. Element connectivity within the anisotropic region is directly determined from the point ordering.

Advancing–Normal Anisotropic Grid Generation

With advancing–normal type point placement for high–aspect–ratio elements, the standard AFLR procedure[3] does produce sliver elements of the type shown in Fig. 1. These elements are generated only in regions of high–aspect–ratio elements with a very structured alignment. Elimination of these elements with local–reconnection is not feasible. There may be no nearby optimization path which produces a better connectivity. The problem is inherently due to the very structured nature of the grid in these regions. Only a limited set of possible triangulations, that do not contain sliver elements, exists for a set of tetrahedra aligned in prismatic groups. A modified process is proposed here which eliminates the sliver problem and retains the generality and efficiency of the original procedure. In the present approach, local–reconnection is not used to determine the connectivity in these regions, Instead, the connectivity is directly determined by the order in which points were generated. This produces a very structured connectivity and allows the elements created to either be all tetrahedra or of mixed type.
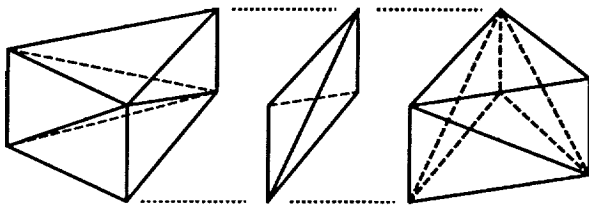

Fig. 1. Trapped sliver element between prismatic groups of tetrahedral elements.

The basic steps in the modified procedure are listed below.

1) Determine a normal vector at each active boundary–layer point. Initially the normals are

based solely on the original boundary surface geometry. As the generation advances the normals are generated using the geometry of the outer layer of the boundary–layer grid.

2) Smooth the normal vectors with a weighting dependent upon the distance from the boundary. Initially the normals are unsmoothed. At the estimated end of the boundary–layer region full smoothing is applied. The end of the boundary–layer region is based on a estimate of where the element aspect ratio will be near isotropic.

3) Generate new points one layer at a time. New points are created along the normal vector with the normal spacing determined using geometric growth from the boundary surface. Generation of new points along a normal is shown in Fig. 2.

4) Check distance between new points and surrounding element quality. The volume triangulation is used to efficiently check nearby points. As boundary–layers merge new points may be too close and advancement should terminate locally. A new point is rejected if the distance between it and any nearby new (or existing) point is less than a preset fraction of the local element length scale. Boundary–layer advancement is terminated locally if a new point is rejected.

5) New points are also rejected if any of the surrounding elements that they may produce fail a quality check (maximum angle < 160 deg.). Boundary–layer advancement is terminated locally if a new point is rejected for quality.

6) Active points can become isolated as boundary–layer advancement is locally terminated. This can be prevented by rejecting a new point and terminating local advancement if more than some fraction of its neighbors have been terminated.

7) Check element aspect–ratio. As the grid advances and the normal spacing increases the element aspect–ratio will eventually be isotropic. Boundary–layer advancement is terminated locally when the aspect–ratio on the next layer would be greater than unity.

8) Attach accepted new points to the volume triangulation. New points are connected and attached to the existing element that contains them.

9) Generate a new boundary surface grid by inflating the previous surface at points that have continued to advance.

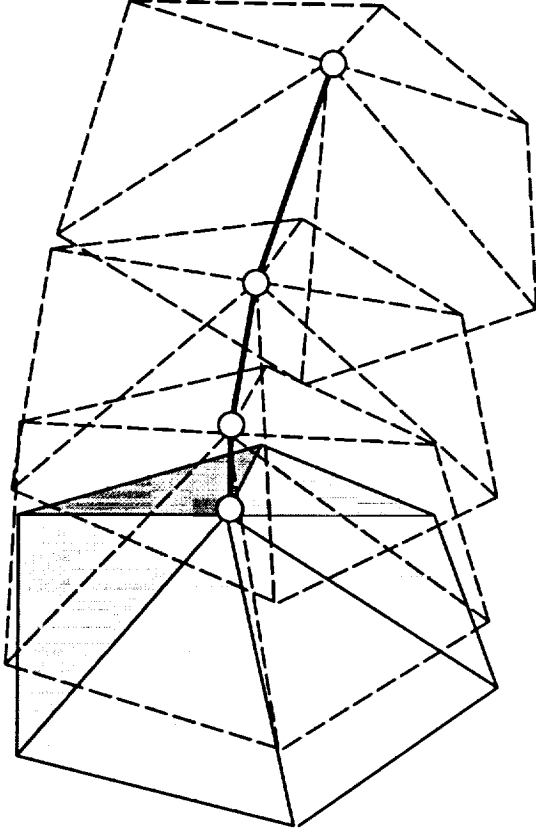10) Repeat steps 1 through 9 until no new points are accepted.

Fig. 2. Advancing–normal point placement along normal line from boundary surface.

Normal Spacing

The normal spacing is determined using accelerated geometric growth. The initial normal spacing can be specified globally or at each boundary point. Standard geometric growth is used with an accelerated growth factor. The normal spacing is determined from

$$\Delta s_{n+1} = \alpha_n \, \Delta s_n \qquad (1)$$

$$\alpha_{n+1} = min \, (\beta \, \alpha_n \, , \, \alpha_{max}) \qquad (2)$$

where Δs_n is the normal spacing for layer n, α_n is the growth factor for layer n, α_{max} is the maximum allowable growth factor, and β is the growth acceleration factor.

Boundary Normal Vectors

A boundary normal vector is required for the advancing–normal procedure. This normal is determined on the inflated surface during each pass. On the first pass the surface is the same as the original boundary surface. As only the boundary face normals are unique (since planar faces are used), some form or averaging or optimization procedure must be used to obtain the normal vector at nodes. Weighted averages can produce variations in normals due purely to topology or local face area differences. In the present work, a least–squares optimization procedure is used to

eliminate those variations. An error function is defined as

$$e_j = 1 - b_i \bullet n_j \qquad (3)$$

where e_j is the error function for face j, n_j is the face unit normal vector for face j, b_i is the node unit normal vector for node i. Node and face normals are shown in Fig. 3. The error function is also directly related to the volume of the element that will be produced from a given face. Minimizing the error function also maximizes the element volume. Least–squares optimization can be used to find b_i such that $\Sigma \, e_j^2$ is minimized. The resulting equations are

$$\Sigma \, [(b_i \bullet n_j) \, n_j^x] = \Sigma n_j^x \qquad (4)$$

$$\Sigma \, [(b_i \bullet n_j) \, n_j^y] = \Sigma n_j^y \qquad (5)$$

$$\Sigma \, [(b_i \bullet n_j) \, n_j^z] = \Sigma n_j^z \qquad (6)$$

where Σ denotes the sum over all faces surrounding node i and n_j^x, n_j^y, and n_j^z are the x, y, and z components of the unit normal vector for face j.
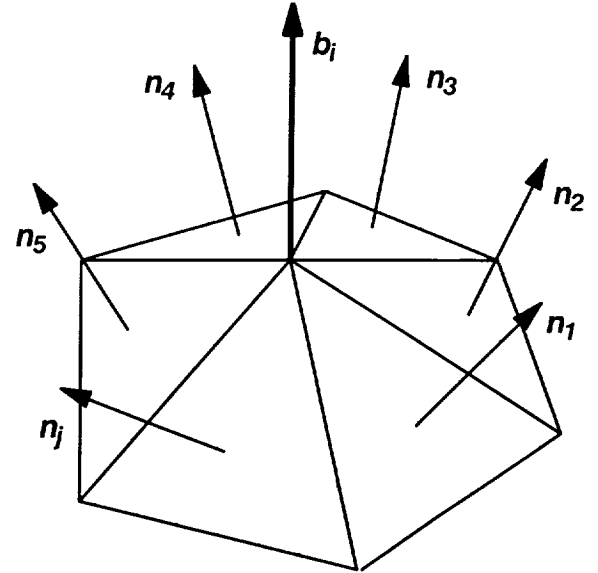

Fig. 3. Node normals determined from surrounding face normals.

Element Connectivity

The element connectivity for the points created using advancing–normal point placement is determined directly from the order in which they were created. The initial point ordering is re–ordered so that optimal element quality will be produced (concave first and convex last). Tetrahedral elements are created (only temporarily if mixed elements are desired) for a given new point by inflating all surrounding boundary faces as shown in Fig. 4. Pentahedral elements can also be created using points generated on subsequent layers. Five and six node pentahedra are formed by combining tetrahedra as shown in Fig. 5. With mixed element types, five–node pentahedra and tetrahedra are created

3

only on the outer layer of the anisotropic region as shown in Fig. 6. These elements are required so that the anisotropic region is bounded only by triangular faces. All of the elements in the combined grid have strict node, edge, and face matching to each other and to neighboring tetrahedral elements.
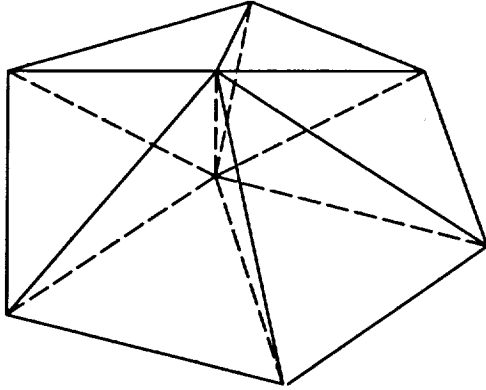

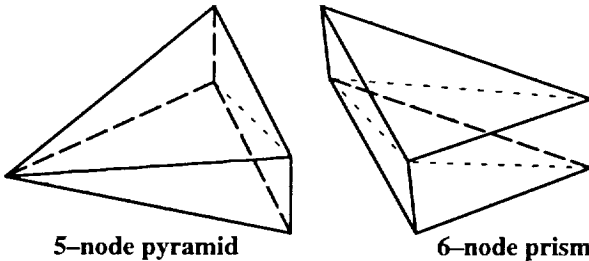**Fig. 4. Creation of tetrahedral elements using advancing–normal point placement.**

5–node pyramid   6–node prism

Fig. 5. Pentahedral elements formed by combining tetrahedral elements.
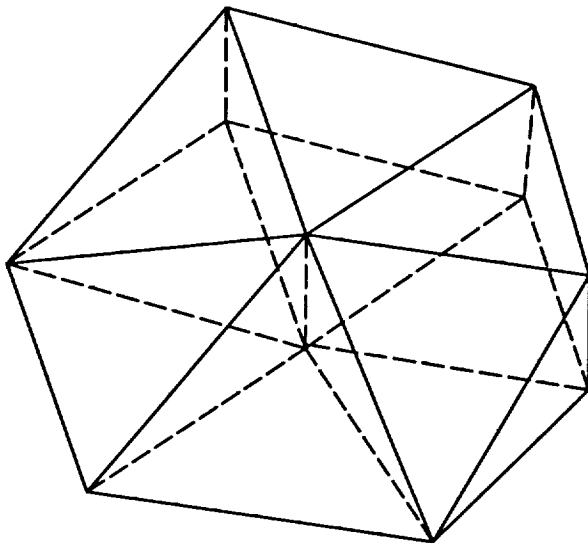
Fig. 6. Use of pyramid and tetrahedral elements to transition between prismatic and tetrahedral regions.

Isotropic Grid Generation

The AFLR grid generation procedure used in the present work is a combination of automatic point creation, advancing type ideal point placement, and connectivity optimization schemes. A valid grid is maintained throughout the grid generation process. This provides a framework for implementing efficient local search operations using a simple data structure. It also provides a means for smoothly distributing the desired point spacing in the field using a point distribution function. This function is propagated through the field by interpolation from the boundary point spacing or by specified growth normal to the boundaries. Points are generated using advancing–front type point placement. The connectivity for new points is initially obtained by direct subdivision of the elements that contain them. Connectivity is then optimized by local–reconnection with a combined Delaunay and min–max type (minimize the maximum angle) type criterion. The overall procedure is applied repetitively until a complete field grid is obtained. Complete details are presented in Refs. 7 and 8.

Application Examples

Selected application examples are presented here to demonstrate the capabilities of the present procedure for generation of three–dimensional unstructured grids of mixed element types that are suitable for Reynolds–Averaged Navier–Stokes simulations. All geometry preparation and surface grid generation work was done using SolidMesh[9] with AFLR surface grid generation[8].

Grid quality distributions and statistics are presented for all examples in Fig. 7. Element angle is used as the grid quality measure. The complete set of grid quality data consists of the six, eight, and nine dihedral angles for all tetrahedra, five–node pentahedra, and six–node pentahedra respectively. In Fig. 7 the distribution plot is in 5 deg. increments. As shown, the distribution has peaks at 60 and 90 deg. from the six–node pentahedra and a peak near 70 deg. from the tetrahedra. The results for the examples presented are representative of those obtained for a variety of configurations. Typically, for isotropic elements in the grid, the maximum element angle is 160 deg. or less, the standard deviation is 17 deg. or less, and 99.5% or more of the element angles are between 30 and 120 deg. And, for anisotropic elements in the grid, the maximum element angle is 170 deg. or less and 99.5% or more of the element angles are between 30 and 135 deg. The minimum angle is usually dictated by the geometry. Convex or concave edges with an included angle less than 20 deg., such as a sharp trailing edge or the interior of a wedge, can produce larger maximum angles in the anisotropic region. The maximum anisotropic element angle can be controlled by specifying the maximum allowable angle (which

4

eliminates generation of such elements) or by use of multiple normals at convex edges[3].
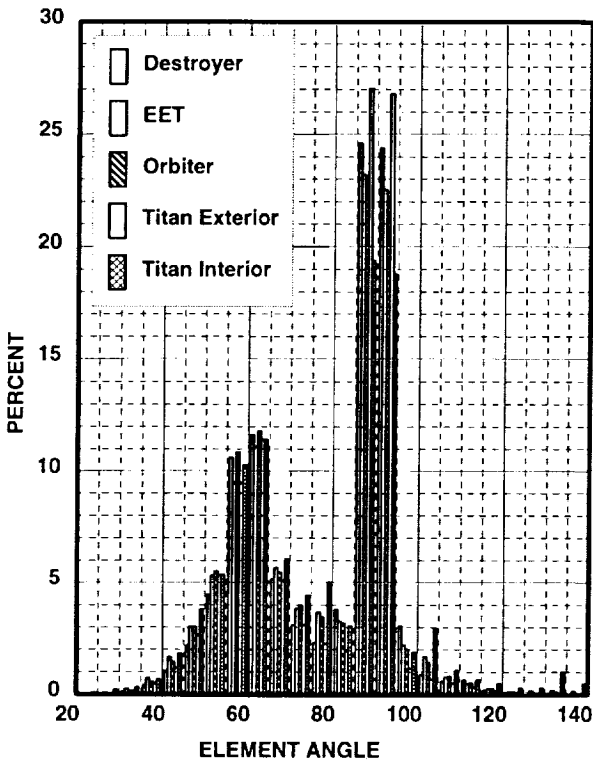

Fig. 7. Volume element angle distributions.

CPU time required on a SUN Ultra 60 workstation is presented in Table 1 for each example. Computer routines for the three–dimensional grid generator are written in C with dynamic memory that is automatically reallocated based upon actual requirements. All floating–point calculations are performed using 64 bit precision with 8 byte data. The CPU times reported are for one processor and include all I/O and generation of grid quality data. A boundary surface grid file is the input. The output includes a grid coordinate and connectivity file and a quality data file. Memory required is about 300 bytes per node generated. Requirements for memory and CPU time vary with the percentage of anisotropic elements, as those requirements for anisotropic generation are considerably less than those for isotropic generation.

User input required to generate a complete grid is minimal and includes specifying the point spacing at selected control points on the boundary curves for surface grid generation. Selection of options such as which boundaries to generate anisotropic elements from and initial normal spacing are the only required user input for volume grid generation. There are no user adjustable parameters that need to be changed from case to case. In all cases presented here, the initial normal spacing was set suitable for high Reynolds number viscous CFD analysis. Initial normal spacing was determined such that the first node adjacent to a viscous surface would have a y^+ value near 1.

Table 1. Number of nodes and elements generated and CPU time required for example cases.

Case	Boundary Faces	Nodes	Five–node Pentahedra	Six–node Pentahedra	Tetrahedra	CPU Time (min)
Destroyer Hull	85,656	1,269,048	17,140	2,234,663	702,609	9.6
EET Wing Body	272,920	2,290,661	33,001	3,744,105	2,004,663	41.9
Space Shuttle Orbiter	152,810	1,102,869	10,269	1,709,948	1,181,232	16.6
Titan IV–A Exterior	337,596	2,980,493	25370	5,113,326	1,962,705	59.4
Titan IV–A Interior	88,502	571,399	31,548	861,492	642,859	5.9

Destroyer Hull

The first case presented is for the Navy model 5415 destroyer hull. Multiple views of the surface grid on the water–line, hull, and propellor surfaces are shown in Fig. 8. A field cut near the propellors is shown in Fig. 9. Element size varies smoothly in the field and there is a smooth transition between the anisotropic and isotropic regions. Grid quality distributions are shown in Fig. 7. Number of boundary faces, nodes, elements and CPU time are presented in Table 1.
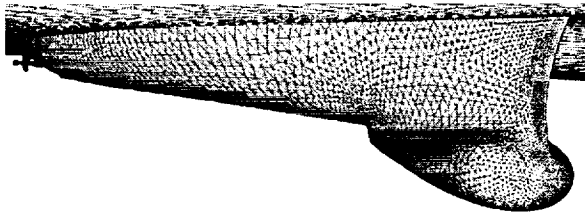
a) bow
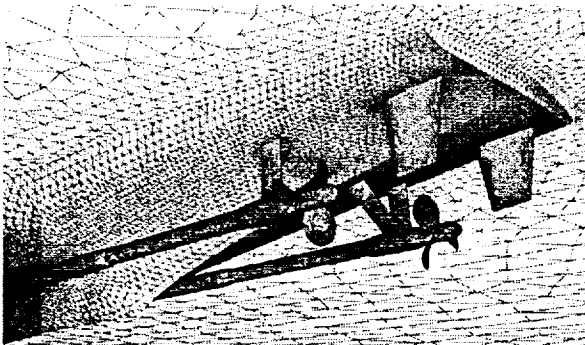
b) stern and propellors

Fig. 8. Destroyer hull surface grid.
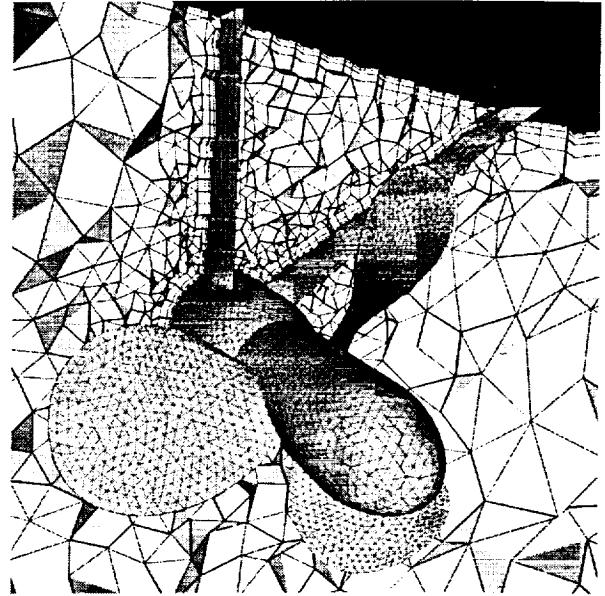
Fig. 9. Field cut near propellors for destroyer hull grid.

Energy Efficient Transport (EET)

A grid suitable for high Reynolds number viscous CFD analysis was generated for a high–lift wing/body configuration of the Energy Efficient Transport (EET). The surface grid on the underside of the wing is shown in Fig. 10. Field cuts near the wing/body, wing/slat, and wing/flap regions are shown in Fig. 11. Element size varies smoothly in the field and there is a smooth transition between the anisotropic and isotropic regions. In narrow regions between components, the number of anisotropic layers is reduced to produce high–quality elements between them. Also, the symmetry plane grid has been re–generated to match the interior anisotropic elements exactly. Grid quality distributions are shown in Fig. 7. Number of boundary faces, nodes, elements and CPU time are presented in Table 1. An all tetrahedral element version of this grid was used by Sheng, et al[10] for incompressible flow simulation with an implicit multi–block flow solver.
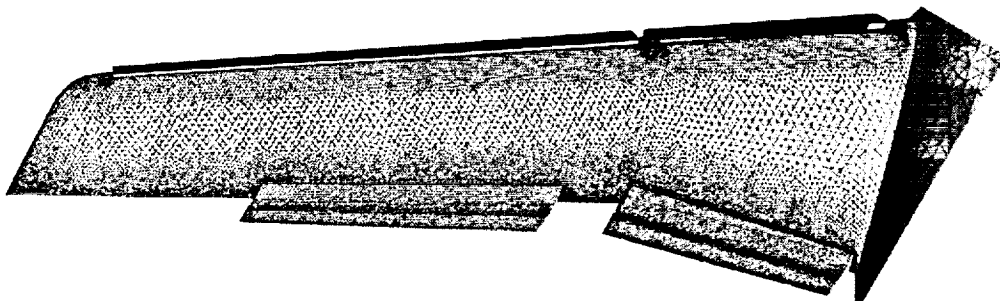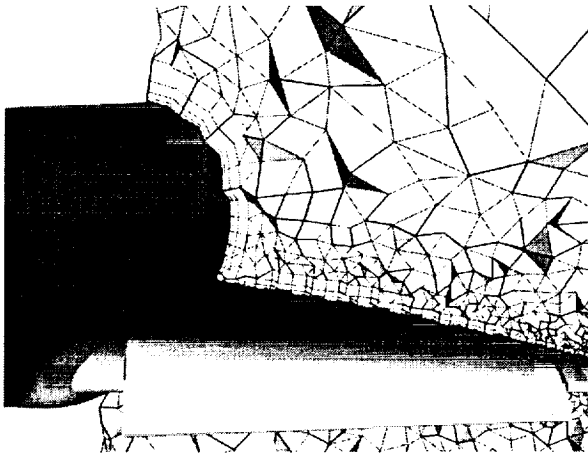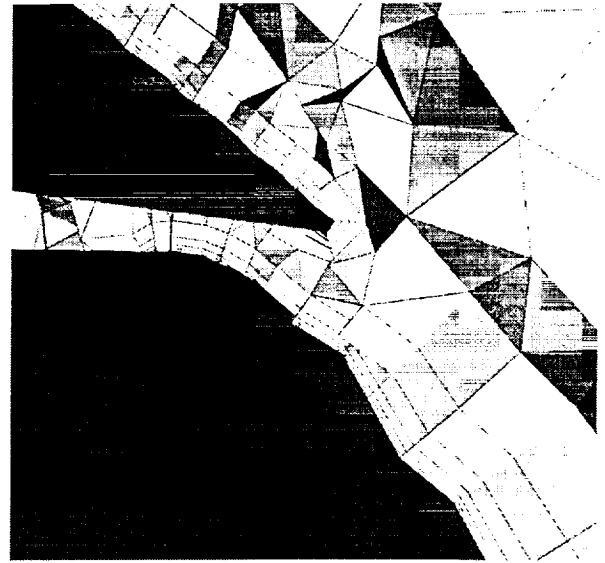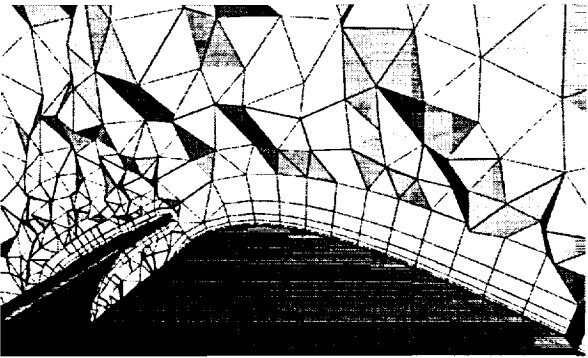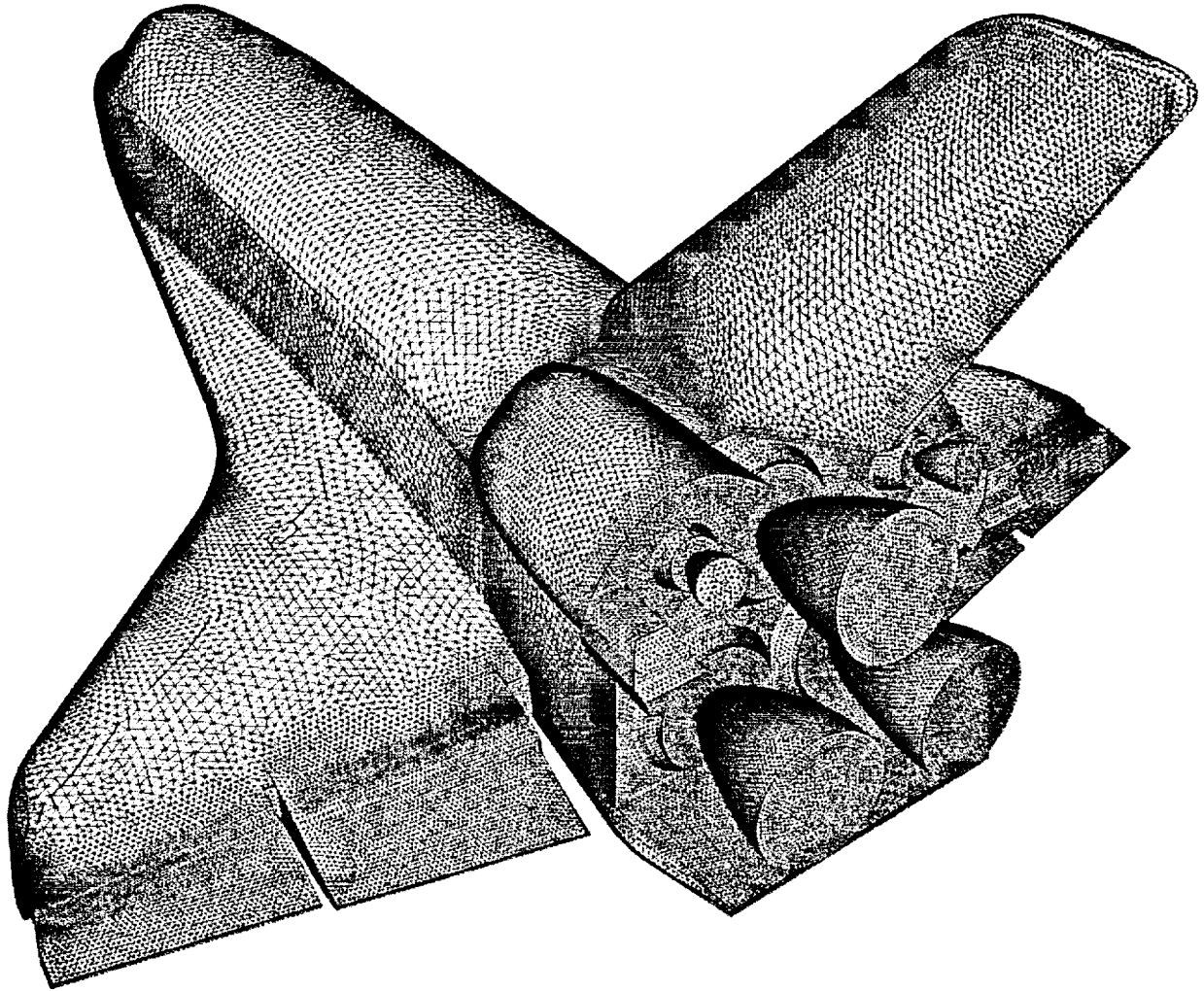

Fig. 10. EET wing surface grid.

a) wing region

b) slat region

c) flap region

Fig. 11. Field cuts for EET grid.

NASA Space Shuttle Orbiter

A grid suitable for high Reynolds number viscous CFD analysis was generated for the NASA Space Shuttle Orbiter. The surface grid is shown in Fig. 12. Field cuts near the rocket motor and inboard flap regions are shown in Fig. 13. Element size varies smoothly in the field and there is a smooth transition between the anisotropic and isotropic regions. In narrow regions between components, the number of anisotropic layers is reduced to produce high–quality elements between them. Also, the symmetry plane grid has been re–generated to match the interior anisotropic elements exactly. Grid quality distributions are shown in Fig. 7. Number of boundary faces, nodes, elements and CPU time are presented in Table 1.
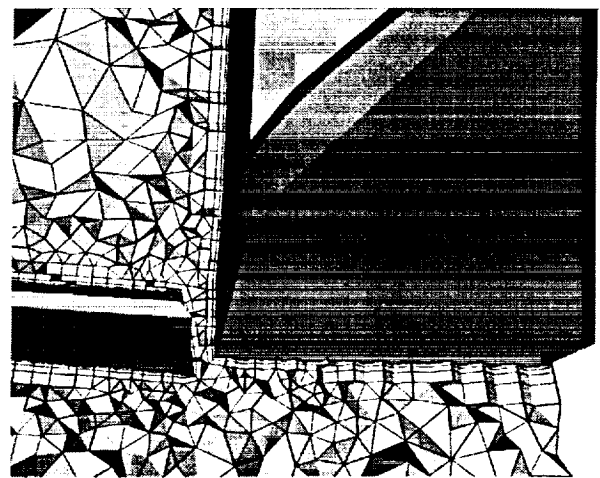
Fig. 12. NASA space shuttle orbiter surface grid.
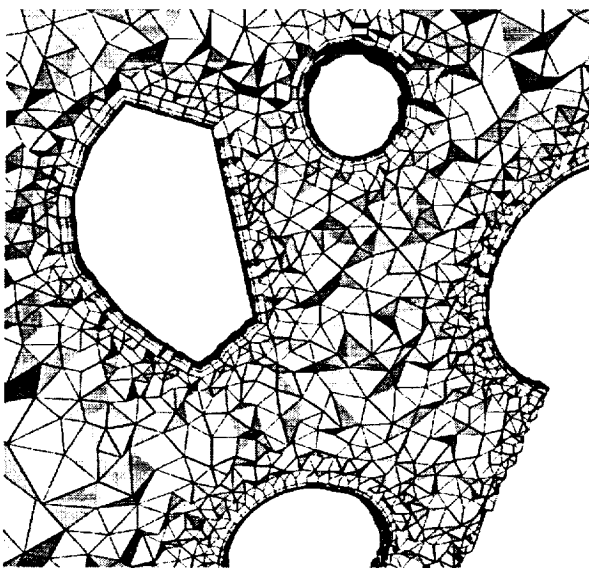
a) rocket motor region

b) inboard flap region

Fig. 13. Field cuts for NASA space shuttle orbiter grid.

8

Titan IV-A Launch Vehicle

A grid suitable for high Reynolds number viscous CFD analysis was generated for a Titan IV-A launch vehicle wind tunnel test model configuration. This configuration includes two strap-on solid rocket motors (SRM), thrust vector control (TVC), stage separation motor (SSM), interstage cavity, and wind tunnel test model sting. The overall configuration and surface grid near the SRM, TVC and SSM regions are shown in Fig. 14. Field cuts near the SRM, TVC, and SSM regions are shown in Fig. 15. A field cut within the interstage cavity is shown in Fig. 16. Element size varies smoothly in the field and there is a smooth transition between the anisotropic and isotropic regions. In narrow regions between components, the number of anisotropic layers is reduced to produce high-quality elements between them. Grid quality distributions are shown in Fig. 7. Number of boundary faces, nodes, elements and CPU time are presented in Table 1. This configuration, and the others presented here, are representative of the high level of geometric complexity that can be handled routinely using the present approach.
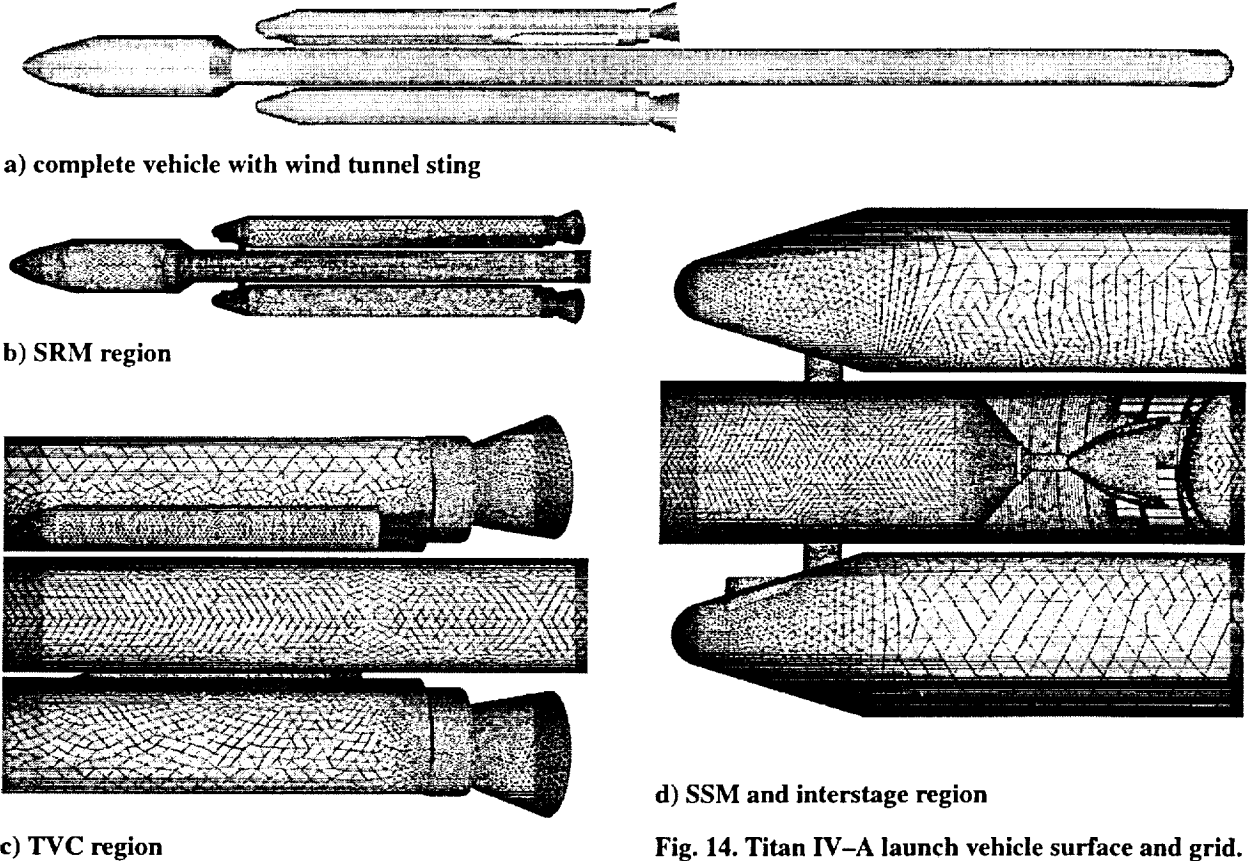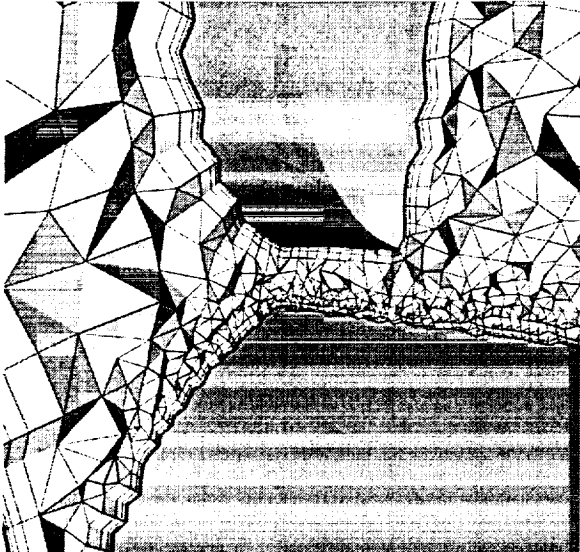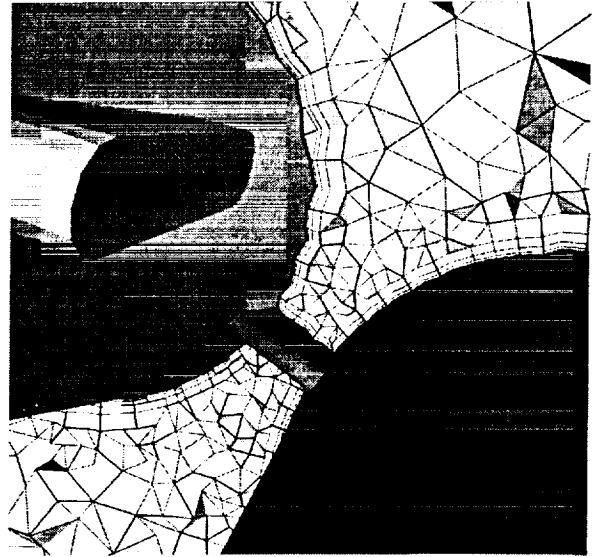

a) complete vehicle with wind tunnel sting

b) SRM region

c) TVC region

d) SSM and interstage region

Fig. 14. Titan IV-A launch vehicle surface and grid.

a) SRM end region

b) TVC end region

c) SSM region

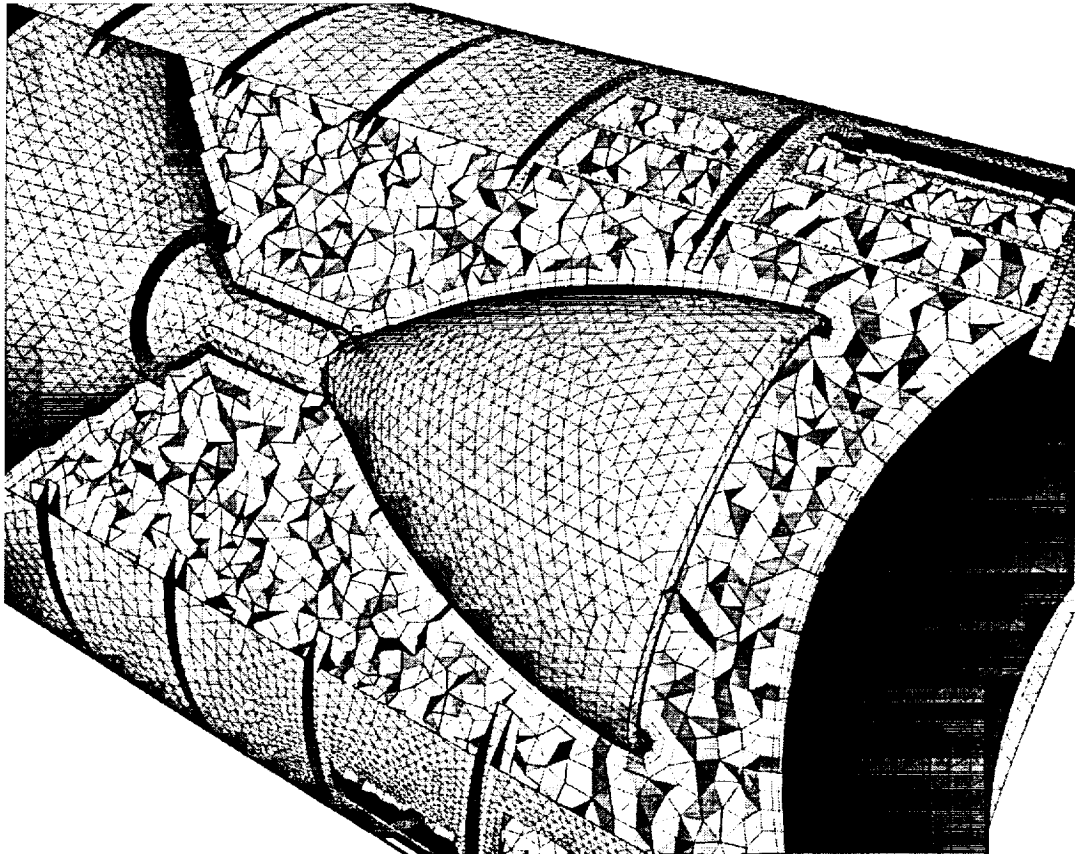Fig. 15. Field cuts for Titan IV–A launch vehicle external grid.

Fig. 16. Field cut and surface grid for Titan IV–A interstage cavity.

Summary

A procedure has been presented for efficient generation of high–quality unstructured grids of mixed element types suitable for CFD simulation of high Reynolds number viscous flow fields. Layers of anisotropic elements are generated by advancing along prescribed normals from solid boundaries. The points are generated such that either pentahedral or tetrahedral elements with an implied connectivity can be be directly recovered. As points are generated they are temporarily attached to a volume triangulation of the boundary points. This triangulation allows efficient local search algorithms to be used when checking merging layers. The existing AFLR procedure is used to generate isotropic elements outside of the anisotropic region. Results were presented for a variety of applications. The results demonstrate that high–quality anisotropic unstructured grids can be efficiently and consistently generated for complex configurations.

References

[1]Hassan, O., Marchant, K., Probert, E., Weatherill, N. P. and Marcum, D. L., "The Numerical Simulation of 3D Turbulent Transonic Flows Using Unstructured Grids," AIAA Paper 94-2346, 1994.

[2]Lohner, R., "Matching Semi–Structured and Unstructured Grids for Navier–Stokes Calculations,"AIAA Paper 93-3348, 1993.

[3]Marcum, D.L., "Generation of Unstructured Grids for Viscous Flow Applications," AIAA Paper 95–0212, 1995.

[4]Pirzadeh, S., "Unstructured Viscous Grid Generation by the Advancing–Layers Method," *AIAA Journal*, Vol. 32, No. 8, p. 1735, 1994.

[5]Kallinderis, Y., Khawaja, A., and McMorris, H., "Hybrid Prismatic/Tetrahedral Grid Generation for Viscous Flows Around Complex Geometries," *AIAA Journal*, Vol. 34, No. 2, p. 291, 1996.

[6]Sharov, D. and Nakahasi, K., "Hybrid Prismatic/Tetrahedral Grid Generation for Viscous Flow Applications," *AIAA Journal*, Vol. 36, No. 2, p. 157, 1998.

[7]Marcum, D.L. and Weatherill, N.P., "Unstructured Grid Generation Using Iterative Point Insertion and Local Reconnection," *AIAA Journal*, Vol. 33, No. 9, p. 1619, 1995.

[8]Marcum, D. L., "Unstructured Grid Generation Using Automatic Point Insertion and Local Reconnection," *The Handbook of Grid Generation*, edited by J.F. Thompson, B. Soni, and N.P. Weatherill, CRC Press, p. 18-1, 1998.

[9]Gaither, J. A., "A Solid Modelling Topology Data Structure for General Grid Generation," MS Thesis, Mississippi State University, 1997.

[10]Sheng, C., Hyams, D., Sreenivas, K., Gaither, A., Marcum, D. Whitfield, D., "Three–Dimensional Incompressible Navier–Stokes Flow Computations About Complete Configurations Using a Multi–block Unstructured Grid Approach," AIAA Paper 99–0778, 1999.