N95- 28754

GENERATING GRIDS DIRECTLY ON CAD DATABASE SURFACES USING A PARAMETRIC EVALUATOR APPROACH

Timothy D. Gatzke* and Thomas G. Melson**
McDonnell Douglas Corporation
St. Louis, Missouri

ABSTRACT

A very important, but often overlooked step in grid generation is acquiring a suitable geometry definition of the vehicle to be analyzed. In the past, geometry was usually obtained by generating a number of cross-sections of each component. A number of recent efforts have focussed on non-uniform rational B-spline surfaces (NURBS) to provide a single type of analytic surface to deal with inside the grid generator. This approach has required the development of tools to read other types of surfaces and convert them, either exactly or by approximation, into a NURBS surface. This paper describes a more generic parametric evaluator approach, which does not rely on a particular surface type internal to the grid generation system and is less restrictive in the number of surface types that can be represented exactly. This approach has been implemented in the McDonnell Douglas grid generation system, MACGS, and offers direct access to all types of surfaces from a Unigraphics part file.

INTRODUCTION

Grid generation tools have progressed rapidly over the past decade with a proliferation of grid generation systems. In order to shorten the time required for grid generation, these tools have focussed on batch[1], interactive[2,3], unstructured grid[4,5,6], and overset grid[7,8] methods. However, all of these tools require a geometry definition in a particular format before grid generation can begin. For the standard test cases such as a NACA 0012 airfoil or the ONERA M6 wing, this geometry is readilly available. Alas, Computational Fluid Dynamics has progressed to the complex configurations of today's and tomorrow's real vehicles and this geometry is primarily generated within Computer Aided Design (CAD) systems.

Several popular CAD systems have developed along parallel paths. Due to the competitiveness of the CAD industry, most of this development has remained proprietary. Coupled with the variety of design needs, this has led to development of a large number of surface types. Attempts to standardize data exchange such as the Initial Graphics Exchange Specification (IGES)[9] and Product Data Exchange Using STEP (PDES)[10] still allow for user defined surface types which are not common to all systems. As a result, today's CAD models are composed of a wide range of surface types, complicating efforts to work directly on the CAD surface types.

In the past, geometry was usually obtained by generating a number of cross-sections of each component. This either involved communicating the CFD engineer's needs to a designer, or

---

* Principal Technical Specialist - Computational Fluid Dynamics Project

** McDonnell Douglas Fellow - Advanced Integrated Mathematical Systems teaching the CFD engineer to use the CAD system. The time for this step is dependent on the complexity of the CAD model and could take a few weeks for a full vehicle such as an F/A-18 aircraft. Other efforts have approached the problem of acquiring geometry by converting the initial CAD surfaces to non-uniform rational B-spline surfaces (NURBS)[11] to provide a single type of analytic surface to deal with inside the grid generator. However, for many surface types there is no exact NURBS representation and so the surface must be approximated. This introduces the additional step to convert surfaces to NURBS and the associated development cost for tools to convert or approximate every surface type. This paper describes a more generic parametric evaluator approach which does not rely on a particular surface type internal to the grid generation system and is less restrictive in the number of surface types that can be represented exactly. This approach has been implemented in the McDonnell Douglas grid generation system, MACGS, and offers direct access to all types of surfaces from a Unigraphics part file.

UNIGRAPHICS AND LOFT DATA

The rapid development of CAD tools over recent years has led to current aircraft defined on a wide range of CAD systems. Over the last three years, McDonnell Douglas has been transitioning to the Unigraphics II CAD sytem. One of the important issues in this transition is how to handle data from aircraft designed earlier on the legacy CAD systems. CAD systems have been proprietary and their internal geometry representation, and the routines to work with geometry have been closely guarded. As a result, each system has its own set of surfaces and formats. A few of these may be common to all systems but a large number are unique to each system.

Another driver towards a large number of surface types is the lofting process. Lofting is the process of defining the external surface of a vehicle. This surface definition is then used for the detail design of the vehicle. Manufacturability is a key concern which requires attention to surface continuity and smoothness along with knowledge of manufacturing processes such as machining, extrusion, etc. This has led to an entire set of tools separate from the basic CAD operations used to design detail parts and generate manufacturing drawings. As an illustration of the magnitude of this issue, between the McDonnell Douglas, Northrop, and Dassault lofting tools, there at least 71 different surface types. This number is continuing to increase as new surface types are introduced to reflect new design and manufacturing proceses.

This makes porting CAD models between systems very difficult. It is prohibitively time consuming to restrict the designer to surface types that are common to multiple systems. These unique surface types were added to improve the design process and it would be a step backwards not to use them. A second approach is to generate conversions of surfaces of one type to another. The most common choice is NURBS since many surfaces, such as parametric bicubics, conic surfaces, and surfaces of revolution can be represented exactly. However, many of these unique surface types are procedurally rather than mathematically defined. An example of a procedural surface is the Spine Controlled Dependent Fillet Surface with Functional Radius. In this surface, circular elements lie in planes normal to a controlling space curve, called the spine curve. The surface defines a fillet between any two supporting surfaces with the radius of any given element defined by a function. The surface is a dependent procedural surface that conforms exactly, in both position and tangency conditions with the supporting surfaces. The radius function can be defined by any of three methods, the most common being interpreted as a function of percent of arc length of the spine curve over the defined range of the fillet surface.

PARAMETRIC EVALUATOR APPROACH

An alternate approach is to use a parametric evaluator[12,13] for interogating analytic surfaces. The basis for the parametric evaluator approach is defining a set of two-dimensional curvilinear coordinates to represent each surface. In defining this coordinate system, each coordinate direction can be restricted to the range from zero to one, such that any point on the surface has an associated unique coordinate pair in this range. The parametric evaluator is the internal module of the CAD system which takes a parametric coordinate and returns the physical coordinates for the associated point on the surface. Parametric evaluators have been written as stand-alone modules for several CAD and lofting systems including Dassault CATIA, EDS Unigraphics II, McDonnell Douglas CADD/CALL and GOLD, and Northrop-Grumman NCAD/NCAL. Having parametric evaluators for each surface type, all that remains is to write tools that access these surfaces to operate on the unit parametric square. This is accomplished through the Advanced Integrated Mathematical System (AIMS) software. This software, developed by McDonnell Douglas, includes a variety of capabilities for surface interrogation, intersection, etc. All of the dependencies based on the surface type are transparent to the user. Internally, AIMS determines the surface type and accesses the appropriate parametric evaluator.

AIMS does make use of surface type information in specific cases. Operations on certain types of surfaces may produce an exact analytic solution. For example, in intersecting a plane with conic section there is an exact solution which will be a conic curve. AIMS will use this information when appropriate to improve speed and/or accuracy transparent to the user.

To allow Unigraphics to access data from other CAD systems, a major effort was undertaken in 1991 by a group at McDonnell Douglas to implement AIMS into Unigraphics. As of Version 9, in addition to its internal entity types, Unigraphics can access surfaces from CATIA, CADD/CALL, GOLD, and NCAD/NCAL through AIMS and the appropriate parametric evaluator. Through AIMS, Unigraphics can perform operations, such as the intersection shown in Figure 1, on surfaces from other CAD systems without converting the surface to another type. Support for surfaces from another CAD system would only require the development and linking in of a stand-alone parametric evaluator for its surface types.

IMPLEMENTATION IN MACGS

The McDonnell Aircraft Computational Grid System (MACGS)[14,15,16] is a general purpose interactive grid generation system which has capabilities for geometry manipulation, and generation of structured, overlapping, and unstructured grids. Implementation of AIMS into MACGS eliminates the need to obtain a point definition of a geometry from the CAD system. This is significant since acquiring a point definition geometry for a full aircraft configuration can take up to two weeks.

Several options were considered in implementing analytic surfaces in MACGS. MACGS stores the user created surfaces and grids in a working file. Storing analytic surface definitions in this file as well was considered but this could require a lot of file space for a large CAD model. The method chosen was to have two files open. In addition to the users working file, the user can also open a Unigraphics part file. There are two ways that the analytic surfaces of the CAD model can be stored in the Unigraphics part file. Surfaces which are created using Unigraphics or read into Unigraphics from other CAD systems through IGES files or some other format are stored in the Unigraphics file directly. The true LOFT surfaces are stored in a separate database and the Unigraphics part file will contain only a reference to these surfaces. Unigraphics user functions are used to interrogate the part file to get a list of surfaces. Through AIMS, access to these surfaces is transparent to the user whether they are stored in the part file, or in the LOFT database which may reside on a different physical system and is accessed via the network. The dashed surfaces in Figure 2, which were defined in the older McDonnell Douglas CAD system, CADD, are accessed from the loft data base. The solid line surfaces reside in the Unigraphics part file along with references to the loft surfaces. Since Unigraphics user functions are used to access the Unigraphics part file, Unigraphics must be running on the grid generation platform.

The AIMS software is used to retrieve points along the edges of the surfaces which are used to display the surfaces. The aircraft forebody configuration of Figure 2 is accessed from the Unigraphics part file and displayed in MACGS as shown in Figure 3. The number of points displayed on an edge is determined by a maximum chordal tolerance between the actual surface and the straight segment between successive points. The user can generate structured or unstructured grids on an analytical surface by specifying distributions of points along each edge of the surface. Parametric coordinates along the edges of the surface are computed from these distributions and a grid is generated in parametric coordinates. Then AIMS is used to compute the corresponding physical coordinates of the grid points. Structured and unstructured grids generated on an analytic surface are shown in Figures 4 and 5 respectively.

Often, there is a need to manipulate geometry[17]. Doing this in the CAD system involves getting a CAD designer involved or extensive CAD training for the CFD engineer. Building capabilities to manipulate the analytic surfaces within the grid system would add another level of complexity and increase the required user skill level dramatically. A simpler method is to use tools that already exist in the MACGS grid generation system. MACGS capabilities include splitting, combining, smoothing, etc., for surfaces in an interactive graphical environment. Since the user is modifying the geometry for a discrete analysis, some degree of approximation is acceptable.

The capability to project user-created geometry or grids onto one or multiple analytic surfaces was also incorporated using AIMS functionality. A point definition surface representation is used to define grids and surfaces generated within MACGS and interpolation on a curve fit in each direction is performed to redistribute points on the surface. This requires care by the user since inappropriate operations can introduce variation from the original surface. To ensure that points are precisely on the CAD geometry, the new grids can be projected back onto the original CAD surfaces at any point. Figure 6 illustrates projection of a structured grid onto multiple analytic surfaces.

COMPENSATING FOR ANALYTIC SURFACE PARAMETERIZATION

It is desirable to have grid distributions based on the physical arclength of a surface, rather than its parametric coordinates. This is especially true for analytic surfaces. There may be a large range of parameterizations for a given surface and the parametric coordinates are generally non-linear with respect to arclength on the surface. Therefore some method is needed to assure that the distribution in physical space satisfies the desired spacing, stretching, and smoothness characteristics.

This same problem has been addressed for surfaces defined by a set of discrete points. When interpolating between the discrete points, some type of fit is assumed, such as a bi-cubic spline. The distribution of the discrete points can be thought of as the surface parameterization. For example, redistributing points on a surface defined by points clustered toward one edge or with a pronounced variation in the distribution across the surface can yeild an undesirable result if some method to base the distribution of points on arc length along the surface is not used.

This is accomplished for analytic surfaces by first calculating a distribution of points on the surface, uniform in parametric space. The arclength distribution of these points is used to define a mapping from the desired arclength parametric coordinates to coordinates based on the parameterization of the surface. Since all physical points are generated by evaluating parametric coordinates on the CAD surface description using AIMS, all points are guaranteed to lie on the surface. The intermediate mapping affects only where they lie on the surface. Also, points on the bounds of the parametric space will lie on the edge of the surface.

Generation of a uniform distribution of points is illustrated in Figure 7. Figure 7b shows the effect of applying a uniform distribution in the parametric coordinates of the analytic surface. Note that the spacing along the body is larger near the singularity than at the opposite edge. Figure 7c represents the desired uniform distribution in an arc length space. To achieve this uniform distribution, the point distribution in the analytic surface parametric coordinates shown in Figure 7d is evaluated to produce the more uniform grid of Figure 7e. This effect of the surface parameterization can be even more pronounced for unstructured grids as shown in Figure 8. Figure 8b shows the problem that the parameterization near the singularity can have on the unstructured grid generation algorithm when compensation is not used. To overcome this problem for unstructured grids, it is useful to generate the grid in a parametric space that more closely approximates the shape of the surface, as shoen in Figure 8c. Figures 8d and 8e again show the intermediate mappings that produce the surface grid presented in Figure 8f.

RESULTS

The initial implementation provides a useful capability to read surfaces and put a distribution of points on the surfaces. There are still several issues which are being addressed. One of the most important deals with trimmed surfaces. Currently, for a trimmed surface the entire untrimmed surface is read. Trimmed surfaces can be very general in nature with multiple holes and irregular shaped cut-outs. It is often not possible to come up with a single unit parameterization of only the trimmed part of the surface. Therefore it is necessary to maintain the analytic definitions of the surface trim curves so that the grid can conform to thes curves as well as the shape of the surface. Future enhancements will also bring in the trimming information. Another important related function is the ability to deal with analytic curves extracted from the edge of a surface. These curves are necessary for constructing the edges of a single grid which spans multiple surfaces.

Manipulations inside the grid generation system on a point-definition surface created from the analytic surface, are based on a fit through the points. This is not generally a problem since the fit through the surface points is smooth and deviation from the surface is usually small compared to the spacing of points on the surface. When necessary, these surfaces can be projected back to the analytic surface.

Another issue involves speed and memory requirements. Memory for the analytic surfaces is allocated dynamically as the surfaces are accessed from the file. When working with a lot of surfaces, there is a tradeoff between keeping everything in memory (and often hitting a memory limit), and cleaning up memory after each operation (and paying a penalty for increased time to access the surface (sometimes across a network) each time it is needed.

For a model with a large number of surfaces, it is desirable to generate a grid spanning multiple surfaces, rather than gridding each one and combining them. This can be accomplished by generating a grid from edge curves and then projecting this grid onto the analytic surfaces. However, the distribution of the grid points after projecting may not match the distribution of the original surface, particularly when the surfaces have a lot of curvature. This is especially important for the boundary spacing. This is generally not as critical for newer models developed early in the design phase, since these often have fewer surfaces defining the geometry. But we have observed that as the maturity of the design increases, the complexity of the surface definition and the number of surfaces defining it increase rapidly. As a result, developing the capability to generate grids across multiple surfaces directly is high on our priorities.

There were several benefits to implementing support for analytic surfaces using AIMS. First of all, the package is very modular and was incorporated rather easily. Since memory for the analytic surfaces is allocated dynamically within AIMS, no major changes to the MACGS data structures were required. The biggest issues in implementing AIMS were related to the linker. The interrelation between AIMS and Unigraphics is at times confusing because some AIMS routines call UG routines and other UG routines call AIMS routines, and it took some effort to sort this out. Secondly, AIMS is robust and users know they have the exact surfaces from the CAD model. Also, the accuracy of operations such as projection are consistent with the equivalent operations performed within the CAD environment.

CONCLUSIONS

Implementation of the AIMS routines into MACGS provides the capability to access any surface in a Unigraphics model, regardless of type. This greatly simplifies the process of geometry acquisition for CFD analyses. However, the amount of manipulation of the geometry within the grid generation system depends on several things including the quality of the model (gaps, overlap, etc.) and simplifications to be made to the geometry for analysis reasons. This approach worked well due to the availability of a mature library for accessing surfaces and evaluating them parametrically. This proved to be much simpler than converting a vast range of surfaces to a particular type, such as NURBS, and making extensive internal changes to the grid generation system to support this type.

REFERENCES

1. Lijewski, L.E., Cipolla, J., Thompson, J.F., and Gatlin, B., "Program EAGLE User's Manual," AFATL-TR-88-117, Vols. I-III, September 1988.

2. Steinbrenner, J.P., Chawner, J.R., and Fouts, C.L., "The GRIDGEN 3D Multiple Block Grid Generation System," WRDC-TR-90-3022, Vol. I,II, July 1990.

3. Remotigue, M.G., Hart, E.T., and Stokes, M.L., "EAGLEView: A Surface and Grid Generation Program and its Data Management," NASA CP 3143, pp. 243-251, April 1992.

4. Lohner, R., and Parikh, P., "Generation of Three-Dimensional Unstructured Grids by the Advancing-Front Method." AIAA Paper No. 88-0515, January 1988.

5. Parikh, P., Pirzadeh, S., and Frink, N.T., "Unstructured Grid Solutions to a Wing/Pylon/Store Configuration Using VGRID3D/USM3D," AIAA Paper No. 92-4572, August 1992.

6. Weatherill, N.P., "A Method for Generating Irregular Computational Grids in Multiply Connected Planar Domains," International Journal for Numerical Methods in Fluids, Vol. 8, pp. 181-197, 1988.

7. Benek, J.A., Steger, J.L., and Dougherty, F.C., "A Flexible Grid Embedded Technique with Application to the Euler Equations," AIAA Paper No. 83-1944, July 1983.

8. Baysel, O., Fouladi, K, and Lessard, V.R., "Multi-grid Upwind Viscous Flow Solver on Three-Dimensional Overlapped and Embedded Grids," AIAA Journal, Vol. 29, No. 6, 1991.

9. Smith, B., and Wellington, Jr., "Initial Graphics Exchange Specification (IGES), Version 3.0," NBSIR 86-3359, April 1986.

10. "U.S. Product Data Association and IGES/PDES Organization Reference Manual," October 1994.

11. Piegl, L., "On NURBS: A Survey," IEEE Computer Graphics and Applications, Vol. II, No. 1, 1991.

12. Melson, T.G., "Analytic/Parametric Conversions," Proceedings CAM-I Seminar on Geometric Representation - Which Way to Go?, Tucson, Arizona, P-84-MM-01, pp. 184-206, January 1984.

13. Moreau, P., "Applying Object-Oriented Architecture to Industrial Geometric Modelers," International Symposium on Advanced Geometric Modelling for Engineering Applications, Berlin (West), FRG, pp. 365-374, November 1989.

14. Gatzke, T.D., LaBozzetta, W.F., Finfrock, G.P., Johnson, J.A., and Romer, W.W., "MACGS: A Zonal Grid Generation System for Complex Aero-Propulsion Configurations," AIAA Paper No. 91-2156, June 1991.

15. Gatzke, T.D., and Weatherill, N.P., "Unstructured Grid Generation Using Interactive Three-Dimensionsl Boundary and Efficient Three-Dimensional Volume Methods," AIAA Paper No. 93-3452, August 1993.

16. LaBozzetta, W.F., Gatzke, T.G., Ellison, S., Finfrock, G.P., and Fisher, M.S., "MACGS - Toward The Complete Grid Generation System," AIAA Paper No. 94-1923, June 1994.

17. Gatzke, T.D., LaBozzetta, W.F., Cooley, J., and Finfrock, G.P., "Geometry Acquisition and Grid Generation -- Recent Experiences with Complex Aircraft Configurations," NASA CP 3143, pp. 31-43, April 1992.

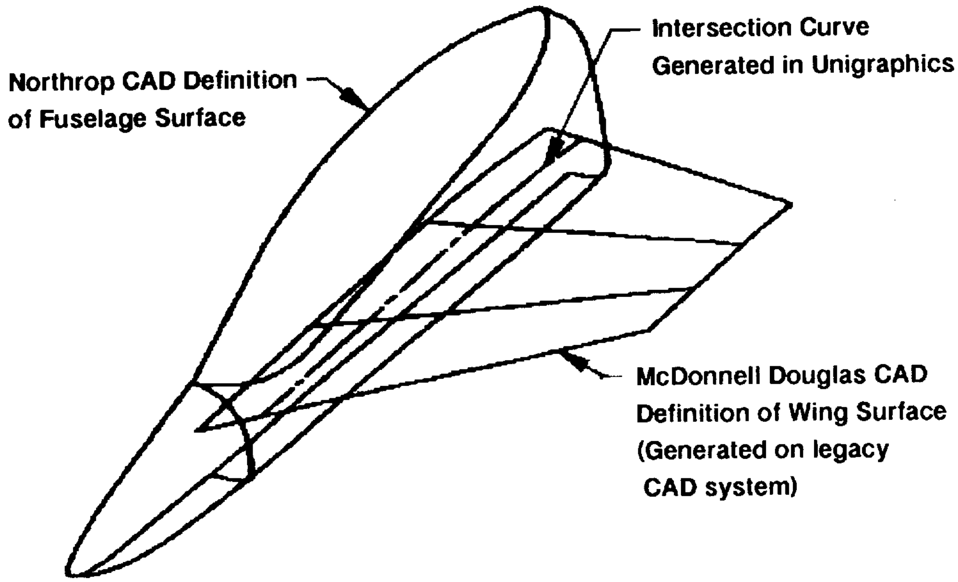


Figure 1. – Unigraphics Intersection of Surfaces from Different CAD Systems Using Parametric Evaluators.

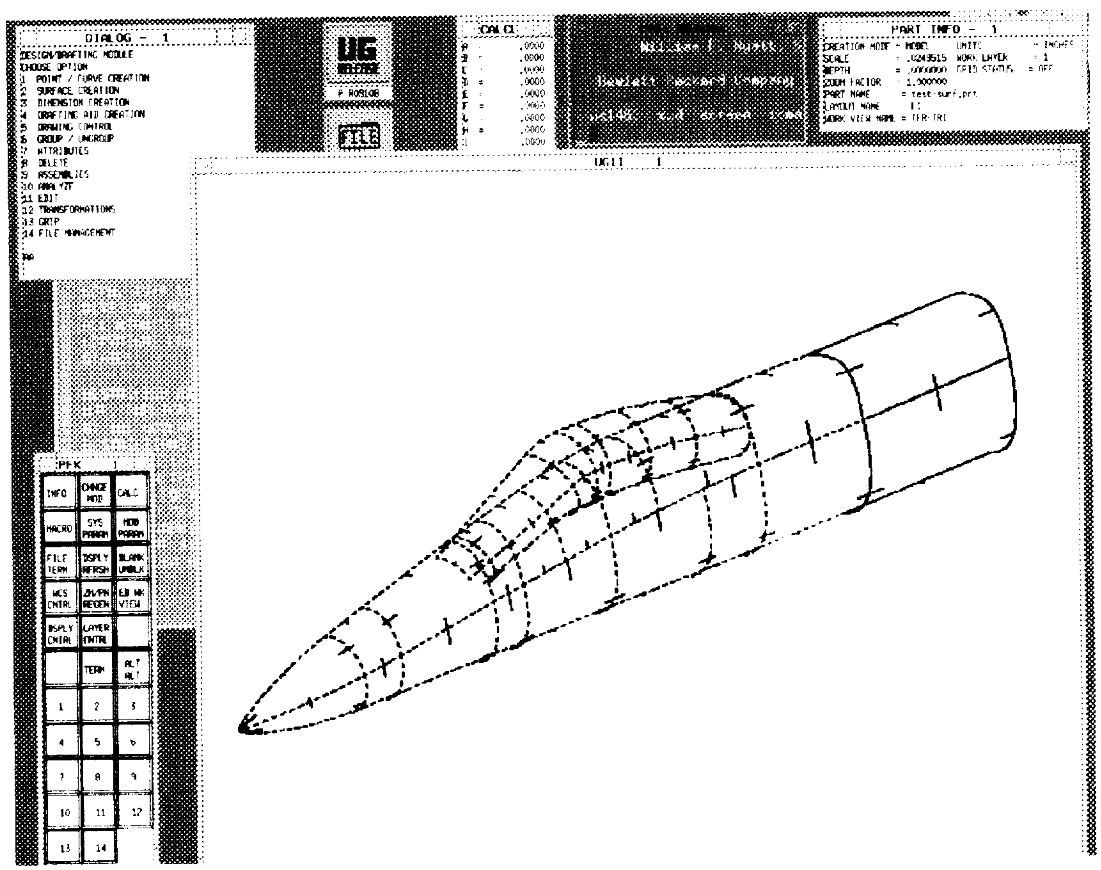


Figure 2. – An Aircraft Forebody and Canopy Definition in Unigraphics.

(Dashed surfaces defined in legacy CAD system.)

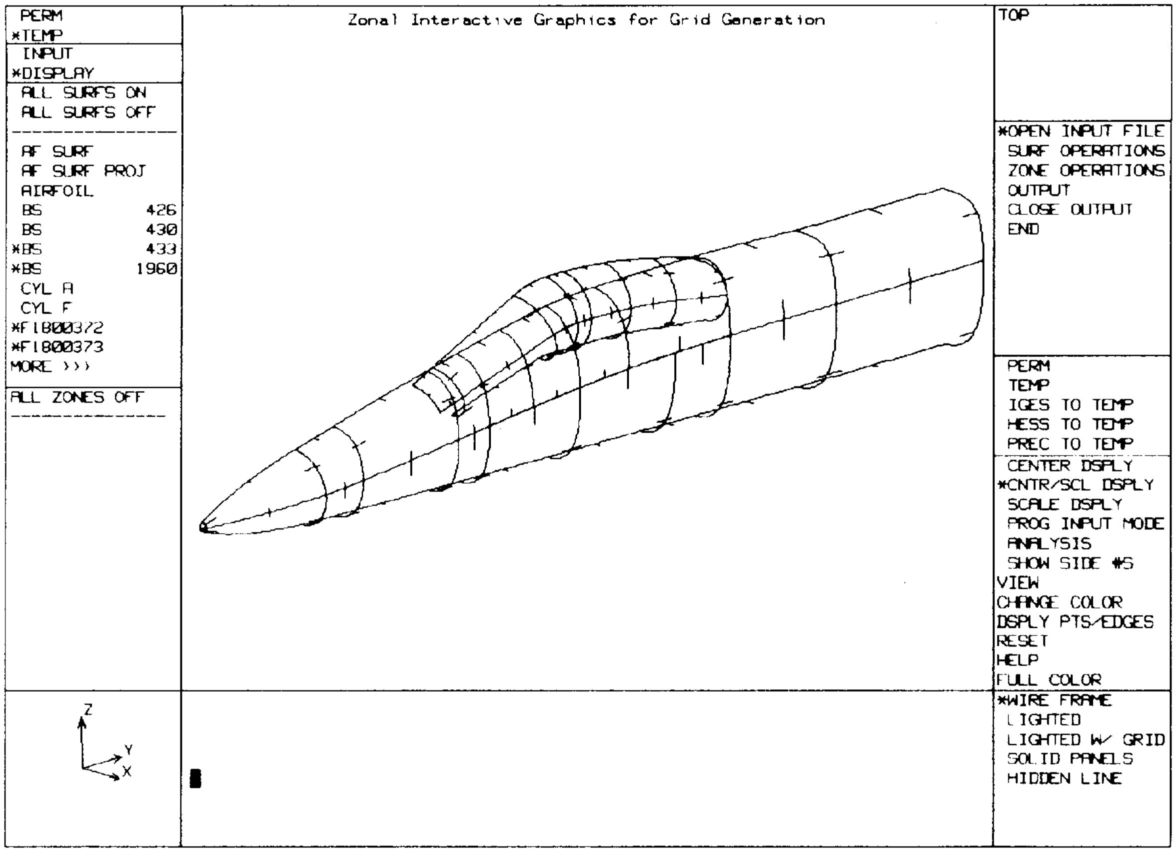


Figure 3. – Analytic Surfaces for Aircraft Forebody and Canopy Displayed in MACGS Surface Gridding Module, ZONI3G.

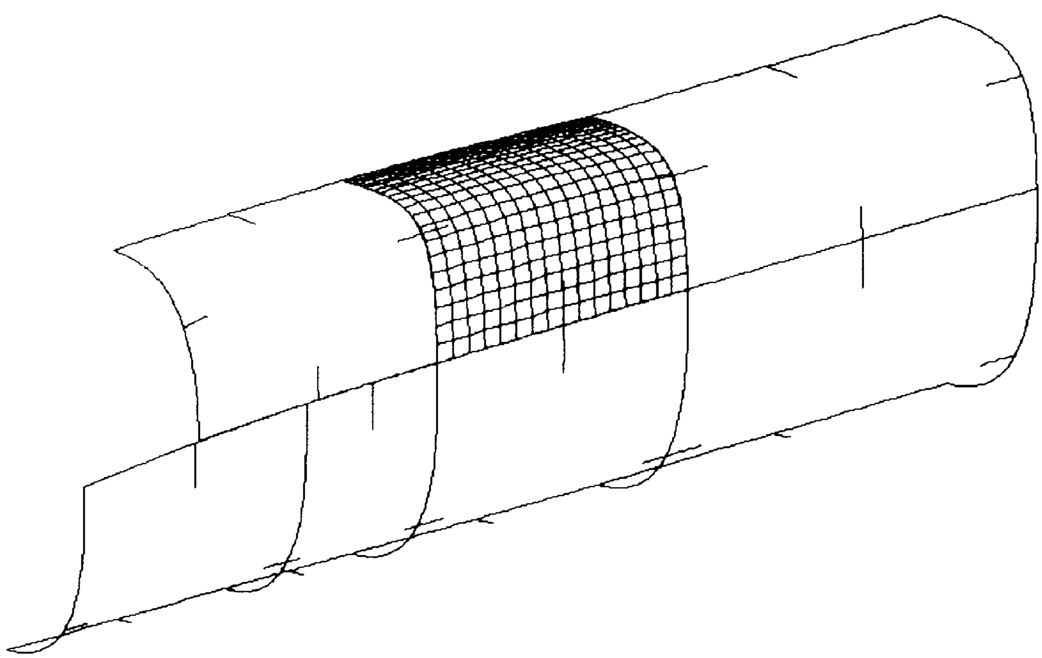


Figure 4. – Structured Grid Generated Directly on Analytic Surface with Clustering Toward Top of Fuselage.

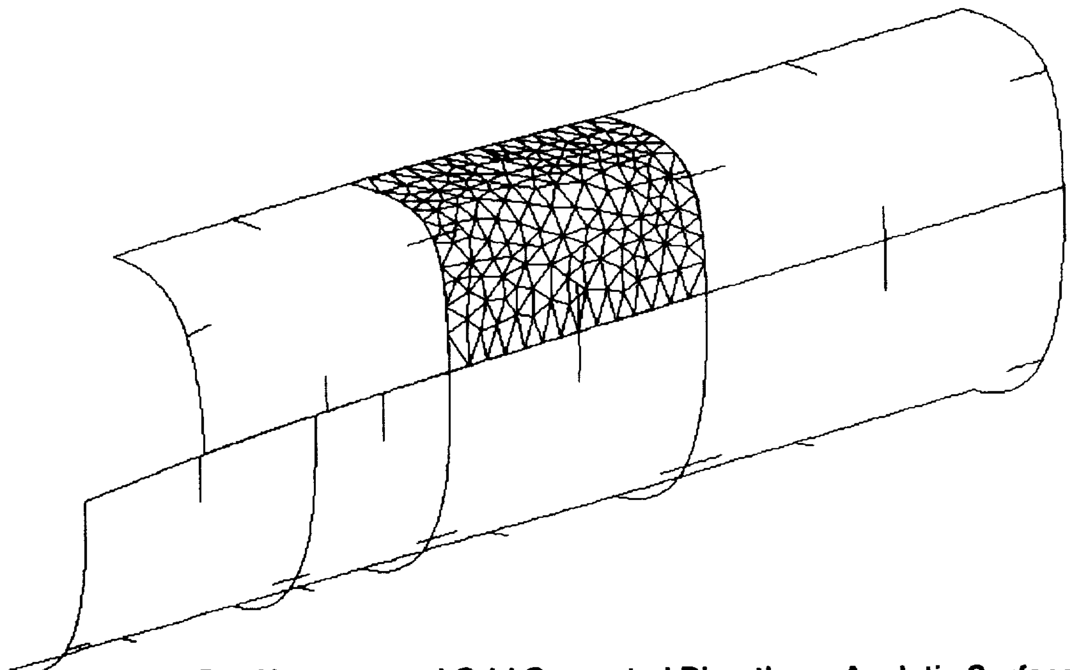


Figure 5. – Unstructured Grid Generated Directly on Analytic Surface.

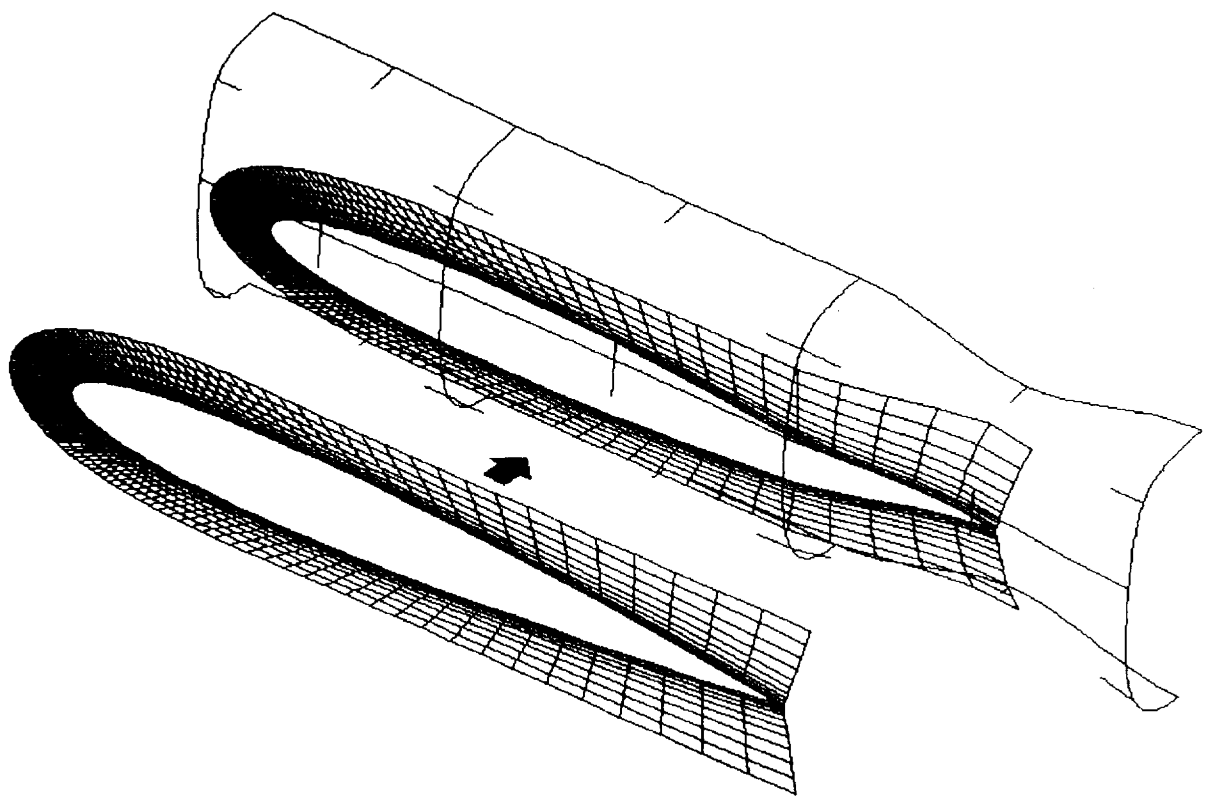
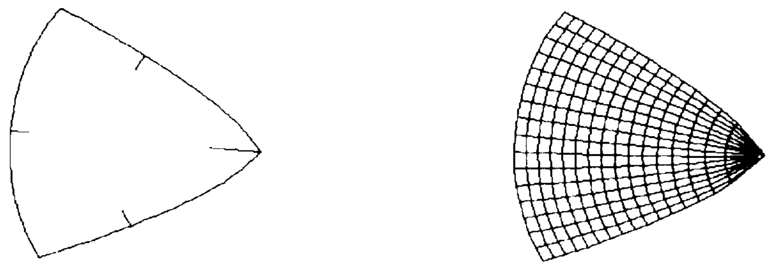
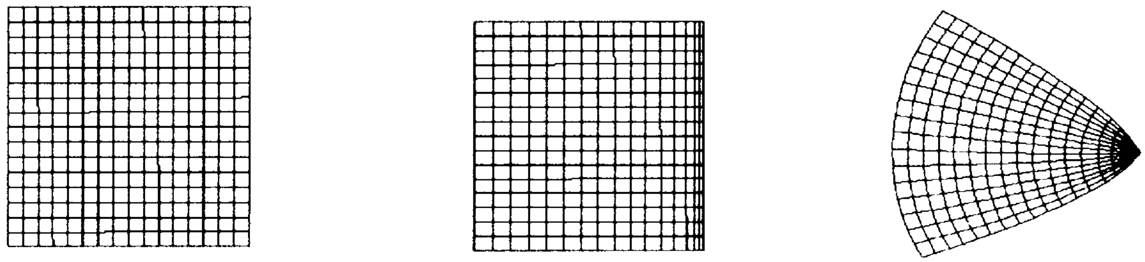


Figure 6. – Structured Grid Projected onto Multiple Analytic Surfaces.



a) Analytic surface

b) Structured grid using analytic surface parameterization

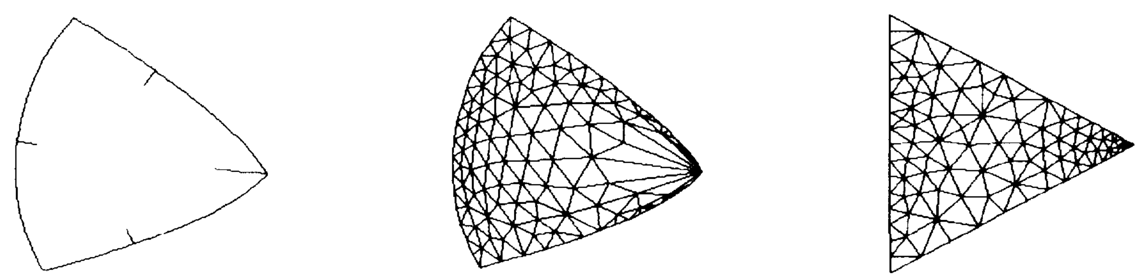


c) Grid generated in arc-length parametric space

d) Grid mapped to analytic surface parameterization

e) Structured grid mapped to analytic surface

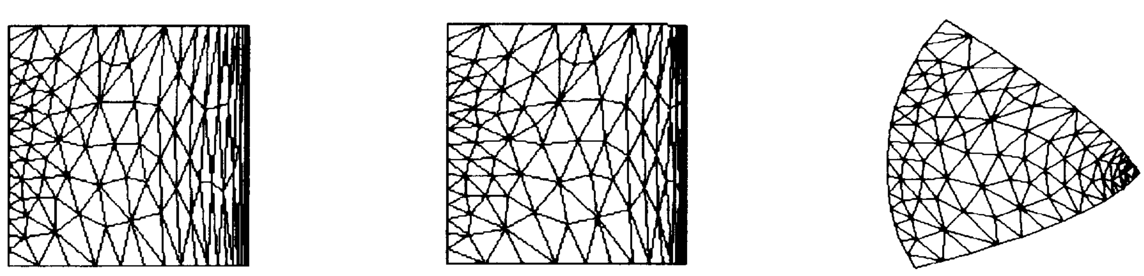
Figure 7. - Structured Grid Generation: Compensation for Analytic Surface Parameterization.



a) Analytic surface

b) Structured grid using analytic surface parameterization

c) Grid generated in modified parametric space approximating surface shape



d) Grid mapped to arc-length parameterization

e) Grid mapped to analytic surface parameterization

f) Unstructured grid mapped to analytic surface

Figure 8. - Unstructured Grid Generation: Compensation for Analytic Surface Parameterization.