

**NASA
Technical
Paper
2151**

April 1983

A Common Geometric Data-Base Approach for Computer-Aided Manufacturing of Wind-Tunnel Models and Theoretical Aerodynamic Analysis

**Michael J. See and
James V. Cozzolongo**

(NASA-TP-2151) A COMMON GEOMETRIC DATA-BASE
APPROACH FOR COMPUTER-AIDED MANUFACTURING OF
WIND-TUNNEL MODELS AND THEORETICAL
AERODYNAMIC ANALYSIS (NASA) 27 p
HC A03/MF A01

N83-22166

CSC L 01A H1/02

Unclas
09864

NASA



25th Anniversary
1958-1983

**NASA
Technical
Paper
2151**

1983

**A Common Geometric Data-Base
Approach for Computer-Aided
Manufacturing of Wind-Tunnel
Models and Theoretical
Aerodynamic Analysis**

Michael J. See
*University of Kansas
Lawrence, Kansas*

James V. Cozzolongo
*Ames Research Center
Moffett Field, California*

NASA
National Aeronautics
and Space Administration
**Scientific and Technical
Information Branch**

TABLE OF CONTENTS

	Page
SUMMARY	1
INTRODUCTION	1
History	2
Approach and Scope	2
INTERACTIVE PARAMETRIC EQUATIONS GEOMETRY SYSTEM	2
Functional Description	3
Data Base Description	5
AD-2000 CAD/CAM SYSTEM	5
Functional Description	7
Data Base Description	7
IPEGS/AD-2000 INTERFACE	7
Mathematical Form Conversion	7
File Format Conversion	9
Patch Management	11
Scaling	11
Patch Corner-Point Extraction	11
TEST CASE	11
Geometry Selection	12
Data Transfer	12
Data Conversion	13
Geometry Reconstruction in AD-2000	17
NC Code Generation	17
NC Machining	17
CONCLUSIONS	17
APPENDIX - PARAMETRIC BICUBIC SURFACE THEORY	21
REFERENCES	23

PRECEDING PAGE BLANK NOT FILMED

SUMMARY

A project to develop a more automated process to produce wind-tunnel models using existing facilities at NASA Ames Research Center is discussed. A new process was sought to more rapidly determine the aerodynamic characteristics of advanced aircraft configurations.

Such aerodynamic characteristics are determined from theoretical analyses and wind-tunnel tests of the configurations. At Ames, computers are used to perform the theoretical analyses, and a computer-aided manufacturing system is used to fabricate the wind-tunnel models. In the past a separate set of input data describing the aircraft geometry had to be generated for each process. The new process establishes a common data base by enabling the computer-aided manufacturing system to use, via a software interface, the geometric input data generated for the theoretical analysis. Thus, only one set of geometric data needs to be generated.

Tests reveal that the new process can reduce by several weeks the time needed to produce a wind-tunnel model component. In addition, this process increases the similarity of the wind-tunnel model to the mathematical model used by the theoretical aerodynamic analysis programs. Specifically, the wind-tunnel model can be machined to within 0.008 in. of the original mathematical model. However, the software interface is highly complex and cumbersome to operate, making it unsuitable for routine use. The procurement of an independent computer-aided design/computer-aided manufacturing system with the capability to support both the theoretical analysis and the manufacturing tasks was recommended.

INTRODUCTION

One function of the aeronautics research process at NASA Ames Research Center is to predict the aerodynamic characteristics of advanced aircraft configurations quickly and accurately. The predictions are based on aerodynamic data which consist of a combination of theoretically computed data and wind-tunnel test data. The theoretical methods and wind-tunnel data acquisition and reduction techniques are continually improved to increase the speed and accuracy of the predictions.

History

In 1972 a project was begun to automate the process of generating the input data required for the theoretical aerodynamic analysis programs that were under development and in use at that time. These data, the geometric information about the aircraft configuration to be studied, had been obtained from the engineering drawings of the aircraft. This process was time-consuming because each analysis program required geometric information in a different form.

The objective of the project was to develop a computer system that would extract the appropriate data from a mathematical model of the aircraft geometry. This model

was to be stored in the system's data base and created from data obtained by digitizing engineering drawings of the aircraft. The system was also to have the ability to present a three-dimensional, dynamic display of the configuration, thereby permitting visual verification of the model. The system that was developed was called the Interactive Parametric Equations Geometry System (IPEGS).

After the development of IPEGS, one of the most costly and time-consuming procedures in a comprehensive aerodynamic analysis was the construction of the wind-tunnel models that were to be tested. Numerical-control (NC) machining was used extensively in manufacturing these models. A commercially available, computer-aided design/computer-aided manufacturing (CAD/CAM) system called AD-2000¹ was used to generate the cutter-path data needed to drive the NC machines. As with IPEGS, AD-2000 required mathematical models of the geometries, which were also generated from the engineering drawings. These models were stored in AD-2000's data base. The task of generating a mathematical model from drawings was therefore done

¹Registered trademark of Manufacturing and Consulting Services, Inc.

The use of trade names of manufacturers in this report does not constitute an official endorsement of such products or manufacturers, either expressed or implied, by the National Aeronautics and Space Administration.

twice, once for the theoretical analysis and again for the wind-tunnel model construction.

If the theoretical analysis and the wind-tunnel model construction could draw from a single mathematical model in a common data base, considerable time could be saved. A new project was therefore initiated in January 1981 to develop a system based on such a common geometric data base. The new system, in addition to saving time, would improve the compatibility of theoretical and wind-tunnel data since both types of data would be derived from precisely the same geometry. This report documents the results of this project.

Approach and Scope

The most efficient way to achieve a common data base was to develop an interface between IPEGS and AD-2000 that would allow the models to be shared. This approach eliminated the need to alter any software internal to either IPEGS or AD-2000. The bulk of the project work involved the development of this interface.

Figure 1 illustrates the overall aerodynamic analysis process with the IPEGS/AD-2000 interface, and shows that the transition from drawings to computer data base need occur only once. With the aid of the interface, this data base can be used for either theoretical analysis or wind-tunnel model construction.

The project was limited to the problem of machining only the external surfaces of the geometry. It was assumed that

there were sufficient capabilities to allow for the addition of the various fittings and connectors required to assemble the components (e.g., wings and fuselage) into a complete model.

INTERACTIVE PARAMETRIC EQUATIONS GEOMETRY SYSTEM

IPEGS was developed to help meet the need for faster and more accurate theoretical aerodynamic analysis. IPEGS is a 3-D, geometric modeling system operated via an interactive graphics computer with dynamic display capabilities. The system performs three primary functions: It (1) permits the visual review and verification of the geometry to be analyzed; (2) provides the aerodynamicist with the ability to interactively modify the surface geometry as needed; and (3) generates the geometric input data for several aerodynamic analysis programs.

Functional Description

Before conducting any aerodynamic analyses, the aerodynamicist verifies that the geometric data represent the mathematical model of the geometry. These data are usually presented as an extensive list of numbers; checking such data is time-consuming and subject to error. A pictorial representation of the geometry defined by the data has fewer

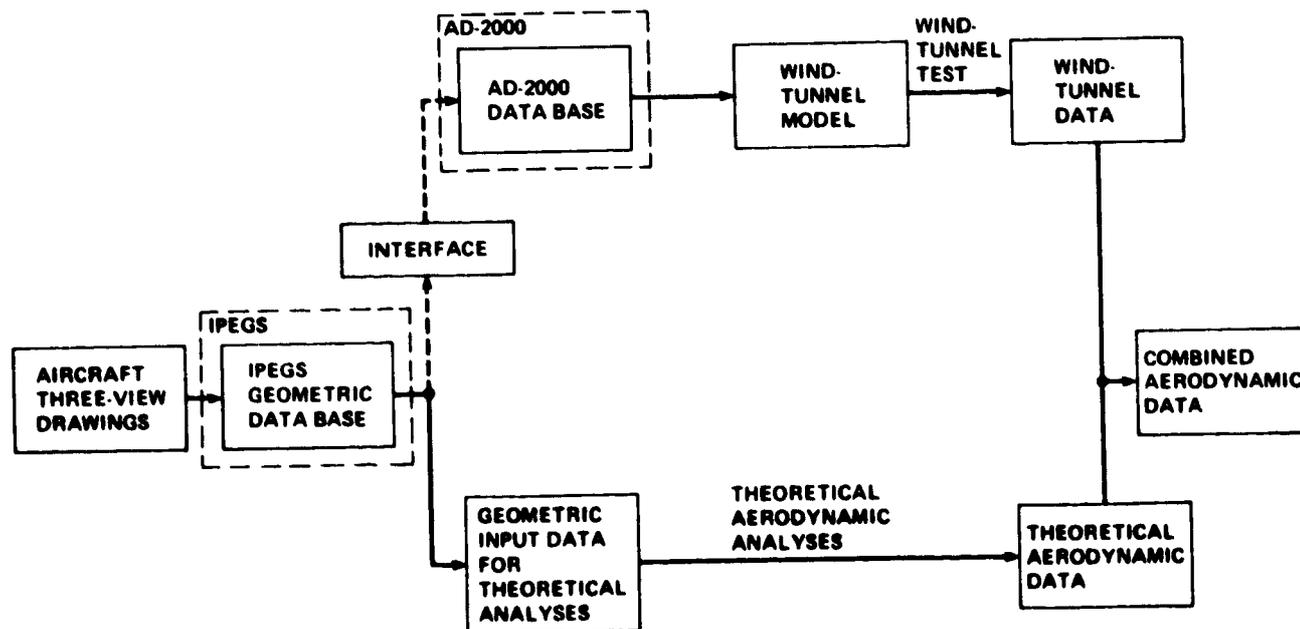


Figure 1.— Common geometric data base approach for acquiring combined aerodynamic data.

drawbacks; it allows quicker identification of errors and provides the aerodynamicist with a better "feel" for the geometry.

IPEGS describes surfaces as collections of parametric bicubic (PC) "patches" displayed on a terminal screen as a mesh of straight-line segments (the density of this mesh is specified by the operator), whose endpoints lie on the true surface (fig. 2), or as a number of straight-line segments (also specified) whose endpoints lie on the patch edges only (fig. 3). The operator can dynamically rotate, translate, and zoom into or out from the geometry. The operator may also display cross sections of the geometry by viewing the portion of the geometry that resides between two planes (parallel to the plane of the screen), which then can be positioned interactively. A hard copy of the display can be obtained from an electrostatic printer/plotter, flatbed plotter, or film writer. The ability to generate nondynamic, color, shaded-surface images of the geometry was recently added to the system (fig. 4).

IPEGS also allows the operator to modify surfaces by interactively deleting, adding, and splitting patches. This interactive graphics capability is quicker than manually editing numbers in the data base. This feature may be used for minor modifications only.

The third IPEGS function is the generation of the geometric input data for two levels of aerodynamic analysis. Level One analyses provide approximate predictions of the aerodynamic characteristics of aircraft configurations. An example

of a Level One program is AEROX (ref. 1). The IPEGS module that generates the input for this program is AEROG I. This module enables the operator to interactively extract and file such quantities as aspect ratio, camber, leading-edge sweep angle, and wingspan.

Level Two analyses employ more complex, linearized potential-flow equations. An example of a Level Two program is PANAIR (ref. 2), which requires the input geometry to be described as a collection of planar polyhedrons, or flat panels. The NETWORK module in IPEGS assembles this description from IPEGS' centralized PC model.

Data Base Description

IPEGS uses PC patches to model geometries to be analyzed (refs. 3 and 4). A patch describes a four-edged surface in space. The edges of a PC patch can have as much curvature as allowed by a third-order polynomial equation. Any one edge can have zero length.

Patches are grouped to form a particular component of a configuration, such as a wing or a vertical tail. These groups are referred to as "objects," and each complete configuration may be made up of a collection of one or more objects. The vertical tail shown in figures 2 and 3 is an "object," and is part of the configuration shown in figure 5.

A PC patch is mathematically described by a set of matrix equations that can be written in either "geometric" or

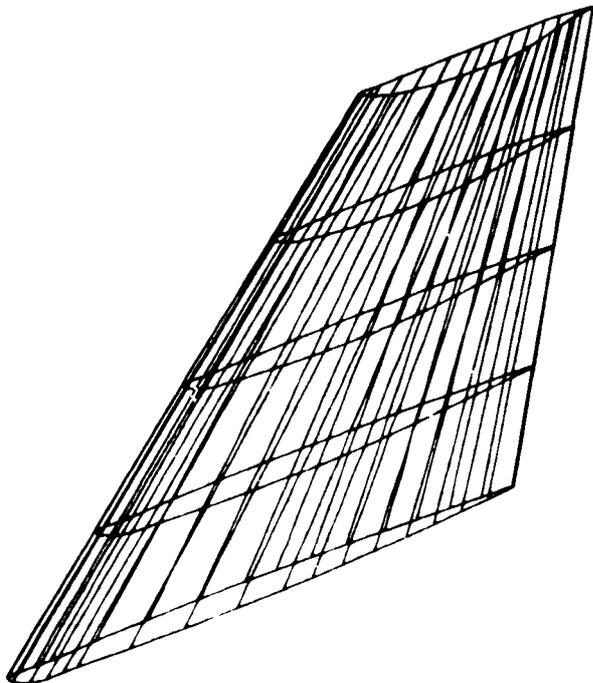


Figure 2.— IPEGS mesh display of vertical tail. (Mesh is 5 X 5.)

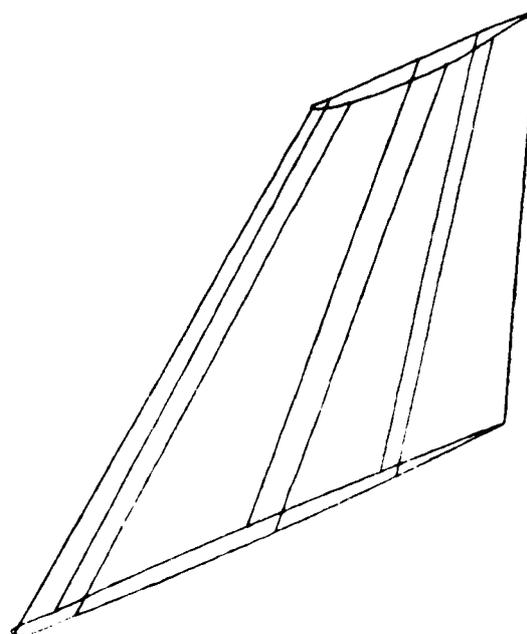


Figure 3.— IPEGS patch-edge display of vertical tail. (Edge density is 20.)

ORIGINAL PAGE IS
OF POOR QUALITY

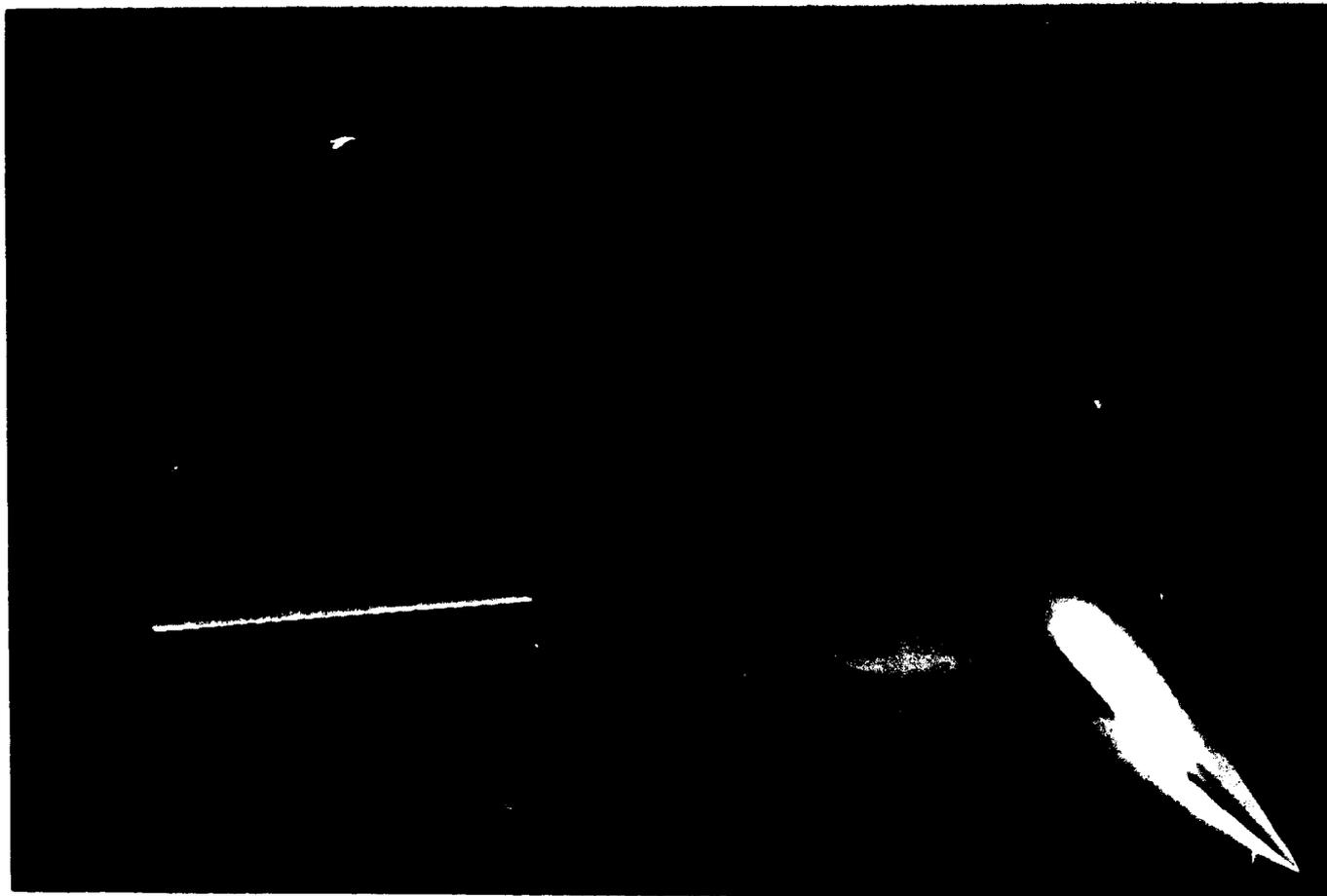


Figure 4.— Shaded surface display of V/STOL fighter configuration.

“algebraic” form (appendix). Three equations are required to describe a patch in three dimensions, one for each of the three coordinate axes. Both forms of the equation contain a matrix of coefficients that embodies the geometric character of the patch.

In the geometric equation, this matrix is called the boundary matrix, and contains four elements of information about the geometric behavior of the patch at each of its four corner points. Thus the boundary matrix has a total of 16 elements, or coefficients. The four elements of information at each patch corner point are: (1) the coordinate of the corner point, (2) the component of the slope (first derivative) of one of the two patch edges which meet at that point, (3) the component of the slope of the other edge, and (4) the component of the cross derivative, or “twist” vector, which controls the interior character of the patch. Since there are three equations for each patch, three boundary matrices are required. Therefore, the geometric character of a PC patch is completely defined by 16×3 , or 48 coefficients. An IPEGS file for a particular geometry contains $48 \times n$ coefficients, where n is the number of patches that make up

the geometry. The geometry of figure 5 consists of 248 patches.

The equivalent matrix in the algebraic equation is called the surface matrix, which also contains 16 coefficients. These coefficients, unlike those of the boundary matrix, have no physical interpretation.

An IPEGS geometric patch file can be created in one of two ways. The first method is to digitize drawings of cross sections of the geometry, and then fit PC patches between the digitized data points using patch-fitting programs. The boundary-matrix coefficients are generated and written into a patch file during this process. The patch-fitting programs are not presently part of IPEGS. The second method is to “read in” previously generated patch files from magnetic tape provided by industry. The latter method has been used more often since many of the larger aircraft manufacturers use PC surface modeling in their computer-aided design systems.

The boundary-matrix coefficients must be arranged in a patch file for storage (table 1). The patch-local coordinate system is illustrated in figure 6. The order of the patches in

ORIGINAL PAGE IS
OF POOR QUALITY



Figure 5.- IPEGS display of V/STOL fighter configuration.

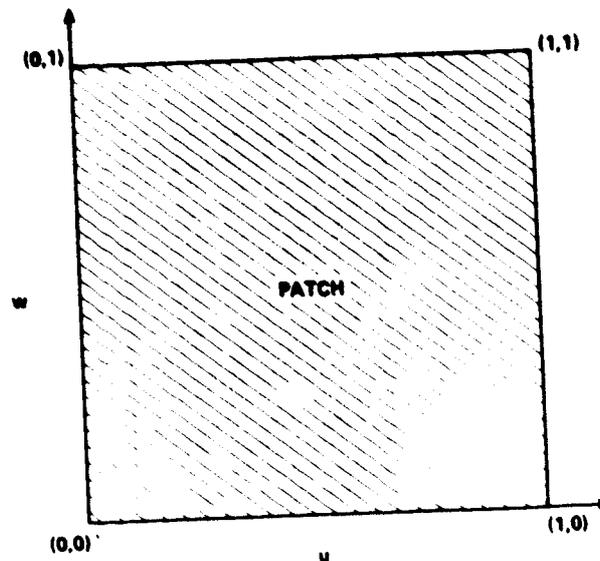


Figure 6.- Patch local coordinate system.

TABLE 1. REQUIRED ORDER OF BOUNDARY-MATRIX COEFFICIENTS FOR IPEGS PATCH FILE (Fig. 6)

Line no.	Description
1	Coordinates of corner point (0,0)
2	Coordinates of corner point (1,0)
3	First derivative with respect to u at corner point (0,0)
4	First derivative with respect to u at corner point (1,0)
5	Coordinates of corner point (0,1)
6	Coordinates of corner point (1,1)
7	First derivative with respect to u at corner point (0,1)
8	First derivative with respect to u at corner point (1,1)
9	First derivative with respect to w at corner point (0,0)
10	First derivative with respect to w at corner point (1,0)
11	Cross-derivative at corner point (0,0)
12	Cross-derivative at corner point (1,0)
13	First derivative with respect to w at corner point (0,1)
14	First derivative with respect to w at corner point (1,1)
15	Cross-derivative at corner point (0,1)
16	Cross-derivative at corner point (1,1)
17	Same as line 1 for next patch etc.

a file is arbitrary. The file represented in table 2 contains only one patch, which is presented graphically in figure 7.

Once the patch file is created, it is read into IPEGS using an input/output module called IPGIO. The management system associated with the IPEGS data base, called XIO, works with patch files in binary form. Thus, another patch file is created by IPGIO with the coefficients in binary form, which can then be read by XIO during the performance of the various IPEGS functions.

AD-2000 CAD/CAM SYSTEM

AD-2000 is an interactive graphics CAD/CAM system (refs. 5 and 6). AD-2000 consists of a set of integrated computer programs which is operated from an interactive graphics terminal. The system's purpose is to facilitate the design, visualization, and construction of geometric entities. AD-2000 is particularly useful in defining and machining objects having complex surfaces.

Functional Description

AD-2000, like other CAD systems, has the ability to create and display basic geometric elements (e.g., points, lines, circular arcs), as well as different types of curves and surfaces. These functions enable the operator to create virtually any geometric object. The system's drafting and display functions allow faster production of complete engineering drawings.

ORIGINAL PAGE IS
OF POOR QUALITY

TABLE 2.- LISTING OF PATCH FILE FOR GEOMETRY OF FIGURE 7.

x	y	z
-0.2500000E+01	-0.2500000E+01	0.0000000E+00
-0.2500000E+01	0.2500000E+01	0.0000000E+00
0.0000000E+00	0.5000000E+01	0.0000000E+00
0.0000000E+00	0.5000000E+01	0.0000000E+00
0.2500000E+01	-0.2500000E+01	0.0000000E+00
0.2500000E+01	0.2500000E+01	0.0000000E+00
0.0000000E+00	0.5000000E+01	0.0000000E+00
0.0000000E+00	0.5000000E+01	0.0000000E+00
0.5000000E+01	0.0000000E+00	0.0000000E+00
0.5000000E+01	0.0000000E+00	0.0000000E+00
0.0000000E+00	0.0000000E+00	0.5000000E+02
0.0000000E+00	0.0000000E+00	-0.5000000E+02
0.5000000E+01	0.0000000E+00	0.0000000E+00
0.5000000E+01	0.0000000E+00	0.0000000E+00
0.0000000E+00	0.0000000E+00	-0.5000000E+02
0.0000000E+00	0.0000000E+00	0.5000000E+02

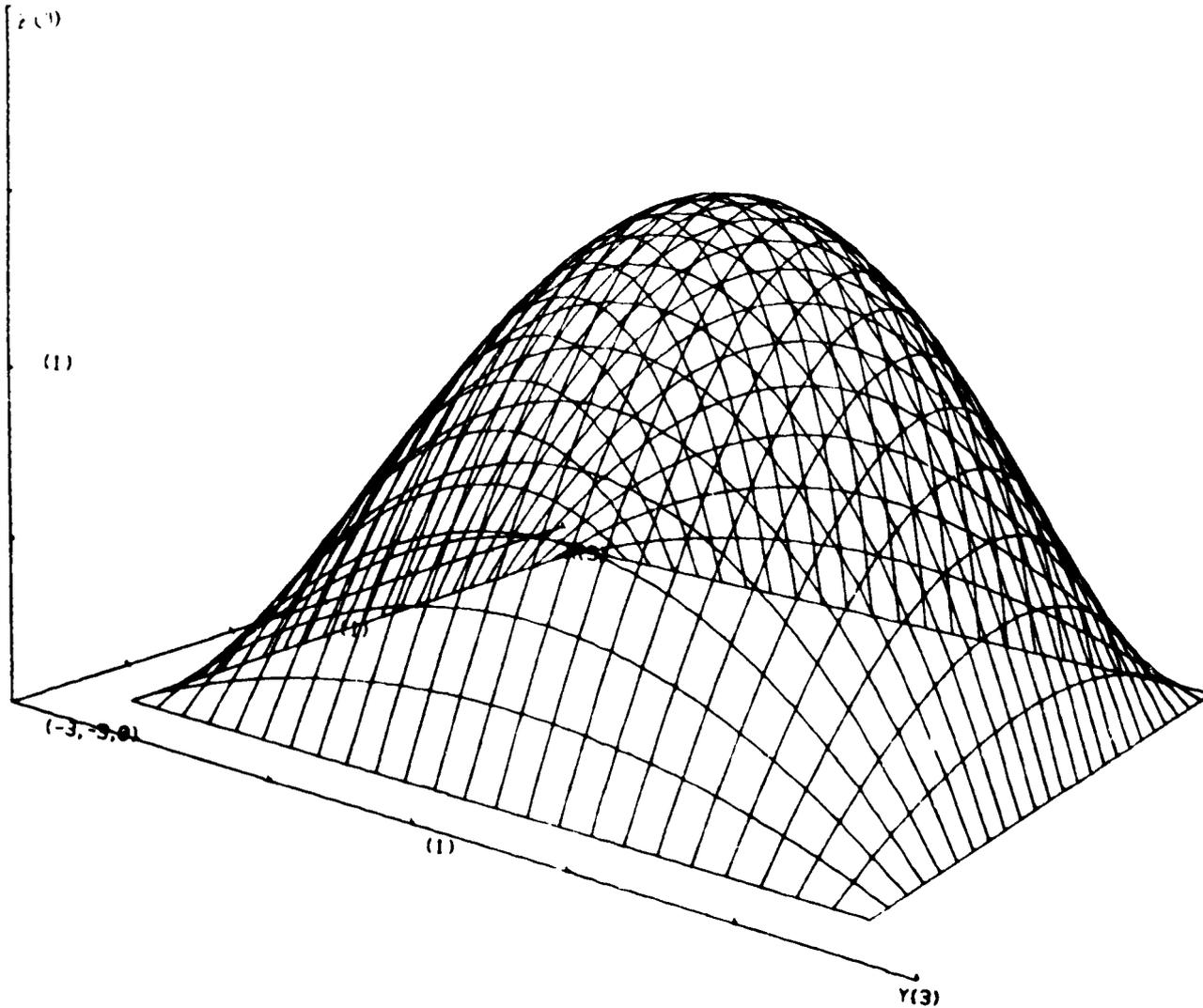


Figure 7.- IPEGS mesh display of single patch described in table 2. (Mesh is 25 X 25.)

ORIGINAL PAGE IS OF POOR QUALITY

The term "computer-aided manufacturing" applies only to those machining operations which call for the use of numerically controlled (computer-driven) machines. The most common of these machines are guided by commands coded in a special computer language called the Automatically Programmed Tool (APT) code.

AD-2000 automatically generates and files NC machine commands in man-readable ("GO TO") form for the desired surface. Typically, the operator specifies the surface to be machined by indicating it with a movable cursor on the display and then keys in information such as tool diameter, tolerance or number of cuts, feed rate, and start position. AD-2000 subsequently calculates and graphically displays the machine tool position commands necessary to machine the surface within the specified constraints. The commands are filed and later converted to APT code using post processors that are not part of AD-2000.

Data Base Description

The version of AD-2000 used in this research cannot create PC surfaces. However, there is a module that, with certain restrictions, permits AD-2000 to accept, display, and manipulate PC surfaces which are generated from other sources in a manner similar to other geometric entities. This means that the PC data must be stored in a file in a specific arrangement so it can be used by AD-2000. The required arrangement is different from that of IPEGS.

A partial listing of a typical AD-2000 PC patch file is presented in table 3. A general description of the contents of an AD-2000 patch file is given in table 4. Note that the patch data must be in algebraic form (appendix). It should also be noted that many other constraints apply to the contents of an AD-2000 patch file.

IPEGS/AD-2000 INTERFACE

The simplest approach for establishing a common data base for theoretical aerodynamic analysis and wind-tunnel model construction was to treat both IPEGS and AD-2000 as "black boxes," each with its own type of input and output data. This approach suggested an interface to allow for the use of an IPEGS output as an AD-2000 input. Internal modifications to either system were unnecessary.

The interface is a computer program that presents the operator with a number of options for manipulating IPEGS patch data to suit both the operator's and AD-2000's needs. It resides on the same computer hardware as AD-2000 and is available to any AD-2000 operator. The interface was written to be operated by NC programmers because NC programmers are familiar with AD-2000 input requirements, and some of the decisions needed during the data conversion

depend upon the specific machining technique to be employed.

The interface has five basic functions: (1) mathematical form conversion, (2) file format conversion, (3) patch management, (4) scaling, and (5) patch corner-point extraction. The first two functions were developed to convert an IPEGS patch file into a form that is understandable to AD-2000. The last three were included to help circumvent some of AD-2000's inherent functional limitations with respect to PC surfaces. Documentation on both AD-2000's PC surface input data requirements and its functional limitations either did not exist or was unavailable to the authors. These characteristics were uncovered during the development of the interface.

Mathematical Form Conversion

AD-2000's bicubic surface input module requires the PC patch coefficients to be in algebraic form. Since the coefficients are geometric in an IPEGS output patch file, the interface converts these data to algebraic form by pre- and post-multiplying the boundary matrix by a matrix (referred to as the "M" matrix) to generate an equivalent surface matrix (appendix). The operation is performed just before the patch data are written to an AD-2000 input file (after all other alterations to the data have been made).

File Format Conversion

The interface also converts the format of the patch data from the IPEGS format to the AD-2000 format. The general format of an AD-2000 PC patch file was given in table 4. Line 1 of the file contains the user-specified file name and the number of patches in the file. Line 2 contains various patch control data for the first patch in the file. These data are used by AD-2000 for various cataloging tasks. A line of this type precedes each patch in the file (table 3). The next 16 lines (3 through 18) contain the algebraic (surface-matrix) patch coefficients. Their arrangement in the file is based on their arrangement when written in surface-matrix form. These coefficients are stored in the file in reverse order with respect to those in the IPEGS patch file. They are stored by columns, beginning with the last element in the last column of the matrix. There are three coefficients on each line that come from each of the x , y , and z surface matrices. As shown in table 4, since a surface matrix has 4×4 dimension, line 3 of the file (the first line of coefficients) must contain the row 4, column 4 elements of the surface matrices. Line 4 of the file must contain the row 3, column 4 elements; line 5, the row 2, column 4 elements; and so on, until line 18, which contains the row 1, column 1 elements. Line 19 contains the control data for the next patch, with lines 20 through 35 containing the coefficients of that patch, and

ORIGINAL PAGE IS
OF POOR QUALITY

TABLE 3.- SAMPLE AD-2000 PATCH-FILE LISTING

x	y	z
<p>AC/578/S13 3 49 -0.2794099E 00 0.7472390E #1 0.0 0.0 0.4999036E-03 -0.1004271E-03 0.0 0.0 0.9313226E-09 -0.2328306E-09 0.0 0.0 -0.2072193E-07 0.6300639E-07 0.0 0.0</p>	<p>1 0 1001 1 -0.1209000E 00 0.4336290E #1 0.0 0.0 -0.6012407E-03 0.5007060E-03 0.0 0.0 -0.1629015E-08 0.1396904E-08 0.0 0.0 0.1261942E-05 -0.9420619E-06 0.0 0.0</p>	<p>1 2 49 0 -0.0600000E 00 0.2744000E 00 0.0 0.0 0.6000000E-02 -0.1000000E-02 0.0 0.0 0.2235174E-07 -0.1117507E-07 0.0 0.0 0.1000000E-06 -0.1043001E-06 0.0 0.0</p>
<p>3 48 -0.2790000E 00 0.7472290E #1 0.0 0.0 0.1662701E 00 0.5107403E-02 0.0 0.0 -0.6063594E-02 0.2663940E-01 0.0 0.0 0.1006404E-02 -0.7346392E-02 0.0 0.0</p>	<p>2 0 1001 1 -0.1215000E 00 0.4336790E #1 0.0 0.0 -0.1907190E 00 0.1660516E 00 0.0 0.0 0.4209027E 00 -0.2440600E 00 0.0 0.0 -0.9090302E-01 0.9931446E-01 0.0 0.0</p>	<p>2 2 49 0 -0.0630000E 00 0.2724000E 00 0.0 0.0 0.2320313E #1 -0.1041431E 00 0.0 0.0 0.3344727E-01 -0.2394772E-01 0.0 0.0 -0.2700770E-01 0.1340702E-01 0.0 0.0</p>
<p>3 40 -0.1100000E 00 0.7496599E #1 0.0 0.0 0.1235006E 00 0.3970000E-01 0.0 0.0 0.4510032E-03 0.1614332E-02 0.0 0.0 -0.1702407E-02 0.1034069E-02 0.0 0.0</p>	<p>3 0 1001 1 0.1700000E-02 0.4347099E #1 0.0 0.0 0.2710076E 00 -0.6737702E-02 0.0 0.0 0.0016251E-01 0.5106305E-01 0.0 0.0 0.2620952E-01 -0.2061439E-01 0.0 0.0</p>	<p>3 2 49 0 0.1400100E #1 0.7779900E-01 0.0 0.0 0.1010070E #1 -0.7267002E-02 0.0 0.0 -0.5100000E-02 -0.7600000E-02 0.0 0.0 -0.1300359E-01 0.1020104E-02 0.0 0.0</p>

TABLE 4.— REQUIRED ARRANGEMENT OF
DATA FOR AD-2000 PATCH FILE

Line no.	Description
1	Patch file title, total number of patches.
2	Control data for Patch 1.
3	↑ Algebraic coefficients; no physical interpretation (16 lines). ↓
4	
5	
6	
7	
8	
9	
10	
11	
12	
13	
14	
15	
16	
17	
18	
19	Same as line 2 for Patch 2.
20	Same as line 3 for Patch 2. ----- etc. -----

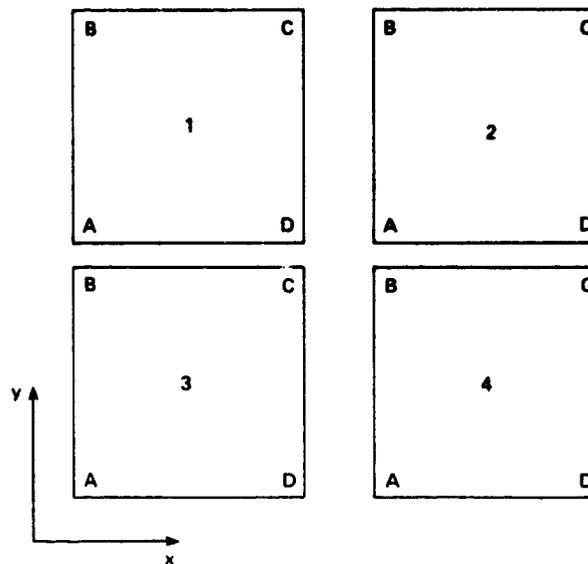


Figure 8.— Four square, planar, PC patches.

so on. The interface creates AD-2000 patch files with this format.

Patch Management

The patches in an IPEGS file are mutually independent entities, and thus the order in which they are listed is arbitrary. All of the patches needed to describe a geometry can be stored in a single IPEGS patch file (up to the physical storage limits of the computer). Any corner point of any given patch can be point (0,0) in the u - w (patch-local) coordinate system, and the direction of the u and w axes is arbitrary for each patch.

Rarely, however, can all of the patches from an IPEGS patch file be stored in a single AD-2000 patch file because (1) the patches in the AD-2000 file must be continuous with respect to one another and (2) the orientation of their u and w axes must be consistent. The continuity requirement applies only to the position of each patch relative to the other patches in the AD-2000 patch file. The surface itself need not be continuous in either the first or second derivative.

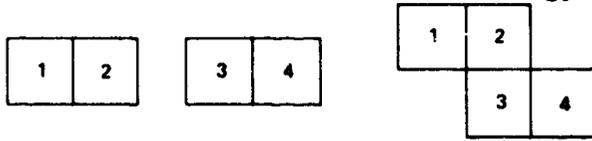
These two requirements can be illustrated. The four generic PC patches shown in figure 8 are square and have no interior curvature. All first-derivative and twist vectors are therefore zero. The patches lie in the x - y plane and are numbered so that they may be distinguished from each other.

- The continuity requirement has three major components:
1. Each patch must be physically connected to the next patch in the file.
 2. Each patch must have common sides with no more than two other patches in the file, with the common sides opposite to each other.
 3. Patches must be listed in the same consecutive order as their physical sequence on the surface.

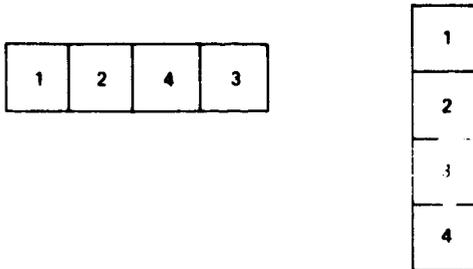
This requirement can be summarized for a surface made up of a matrix of $n \times m$ patches as follows: For an IPEGS surface to be properly incorporated into the AD-2000 system, it must be divided into either $n \times 1$ or $1 \times m$ "strips" of patches, each strip occupying a separate AD-2000 input patch file (fig. 9). All four patches in the first arrangement of figure 9(a) cannot be stored in one AD-2000 patch file because patch 3, which would follow patch 2 in the file, is not physically connected to patch 2. In the second arrangement of figure 9(a), the patch 2 sides that are common with patches 1 and 3 are not opposite to each other (note that patch 3 is in a similar situation) and thus all four cannot be stored in one file. The four patches in the first arrangement of figure 9(b) can be stored in a single AD-2000 patch file, provided that the first patch listed is patch 1; the second, patch 2; the third, patch 4; and the fourth, patch 3. The patches in the second arrangement of figure 9(b) can be stored in one file in 1-2-3-4 order.

The common orientation requirement of patch files is that each patch's local coordinate system must be oriented in the x - y - z coordinate system consistently with that of the other patches in the file. This requirement dictates that the origin of the u - w coordinate system (fig. 6) and the directions of the u and w axes must be declared for each patch in the file such that the side between corner points (1,0) and

ORIGINAL PAGE IS
OF POOR QUALITY



(a) CANNOT BE STORED IN A SINGLE PATCH FILE

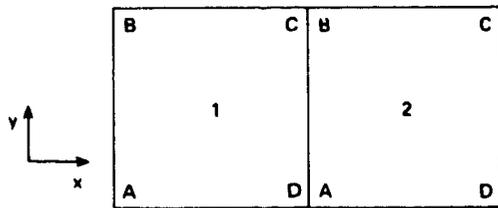


(b) CAN BE STORED IN A SINGLE PATCH FILE

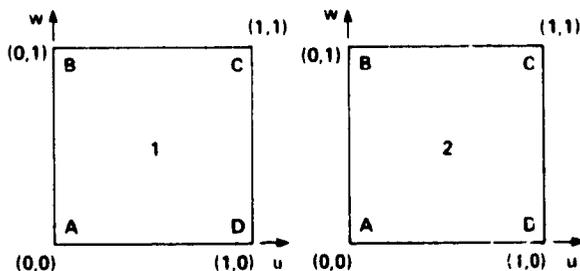
Figure 9.— Continuity requirement in AD-2000.

(1,1) for any given patch must coincide with the side between points (0,0) and (0,1) of the following or preceding patch in the file (fig. 10(a)). Side CD of patch 1 and side AB of patch 2 coincide. The patch data must be arranged in the AD-2000 patch file based on the orientation of the u and w axes (fig. 10(b) corner points (1,1) and (1,0) of patch 1 coincide with points (0,1) and (0,0) of patch 2, respectively).

The IPEGS/AD-2000 interface enables the NC programmer to ensure that the AD-2000 input patch files satisfy the previously mentioned continuity and orientation



(a) SAMPLE ARRANGEMENT



(b) REQUIRED ORIENTATIONS OF u - w COORDINATE SYSTEMS

Figure 10.— Common orientation requirement in AD-2000.

requirements. To use these features, the NC programmer needs a topological map of the patches that make up the geometry to be machined (fig. 11). This map indicates the order of the patches in the IPEGS patch file and can be generated using one of the surface modification functions available with IPEGS.

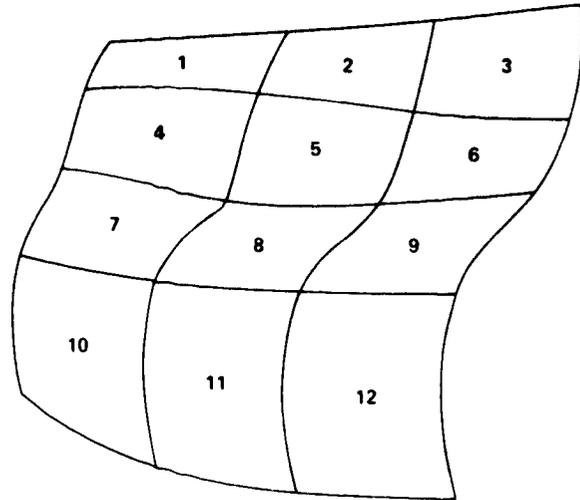


Figure 11.— Topological map of arbitrary surface.

The interface can extract a user-specified group of patches from an IPEGS patch file to satisfy the continuity requirement. The patches are written to a separate AD-2000 patch file after other required manipulations are performed. For example, if the numbers of the patches of the first arrangement shown in figure 9(a) were to their order in the IPEGS patch file, the operator would specify patch numbers 1, 2, 4, and 3, in that order, to be written to an AD-2000 patch file.

The IPEGS/AD-2000 interface can also reorient the u - w axis system of the selected patches with respect to the x - y - z axis system. This capability is limited to swapping the u axis with the w axis. The origin of the u - w axis system cannot be moved from one corner point to another; however, it is rarely necessary to do so.

The surface shown in figure 12 is made up of a 2×2 matrix of patches. The u - w axis systems of the patches are consistently oriented in the IPEGS file (fig. 12). This orientation permits patches 1 and 2 to be written to one AD-2000 file and patches 3 and 4 to another file without any reorientation. If the surface patches need to be divided into columns (a 1-3/2-4 grouping), the u and w axes must be swapped. This is done by interchanging corner points (1,0) and (0,1) and their associated first and cross derivatives in the IPEGS file.

The decision to reorient the local axis systems of the selected patches is made by first assuming that reorientation

**ORIGINAL PAGE IS
OF POOR QUALITY**

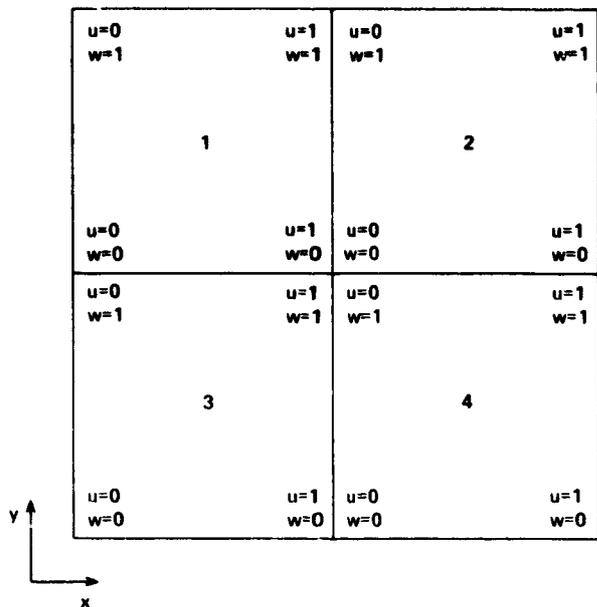


Figure 12.— IPEGS surface with consistent patch orientation.

is unnecessary and by displaying the patches in AD-2000. An incorrect display (such as the one shown in figure 13(a)) will

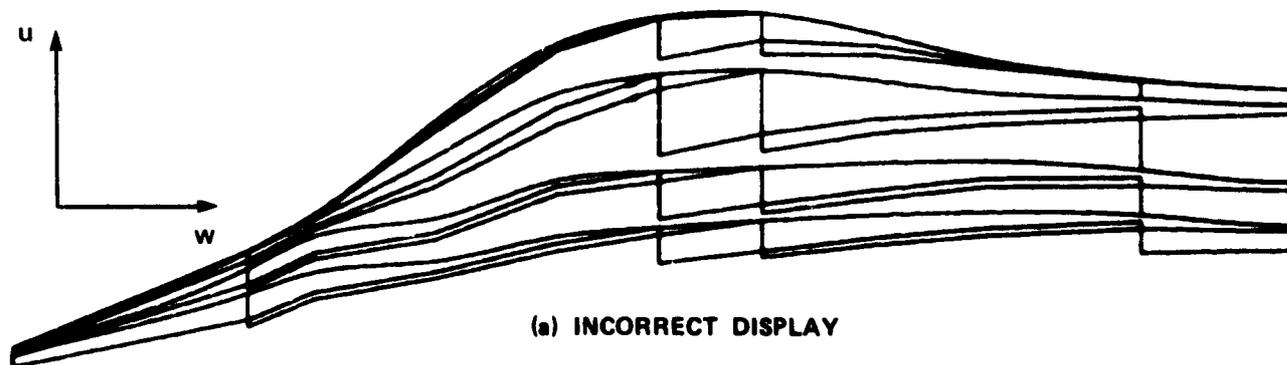
indicate that reorientation is necessary. A correct display will appear as shown in fig. 13(b).

Scaling

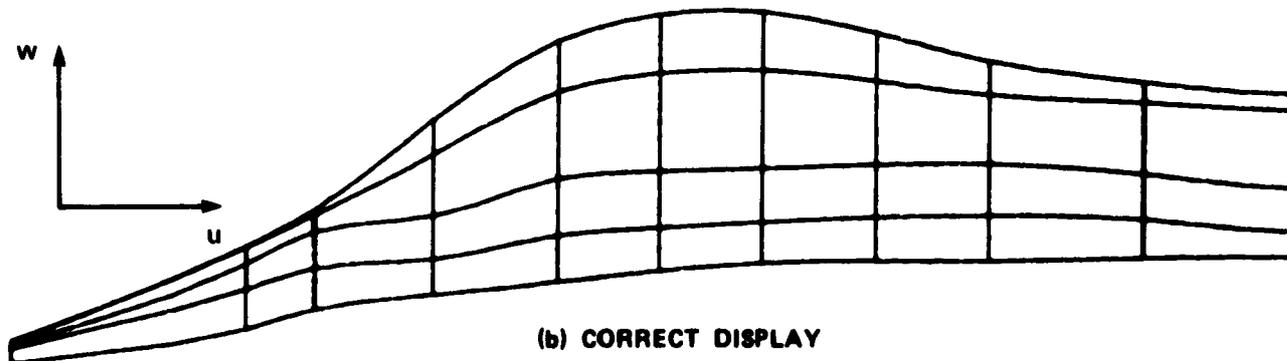
The scale of the geometry may be changed to produce different sizes of models for different wind tunnels or other applications. AD-2000 cannot scale PC surfaces; the interface performs this function by simply multiplying all patch coefficients by the user-specified scale factor.

Patch Corner-Point Extraction

Before generating the NC code, it is helpful to know the coordinates of certain points on the surface. The coordinates of the patch corner points are sufficient for most situations. AD-2000 cannot evaluate the coordinates of the patch corner points, so this capability was added to the IPEGS/AD-2000 interface. The interface extracts the corner points from the IPEGS patch file and creates a separate data file of these points for input to AD-2000. These points can be displayed on the surface (fig. 14).



(a) INCORRECT DISPLAY



(b) CORRECT DISPLAY

Figure 13.— Method of determining need for patch reorientation.

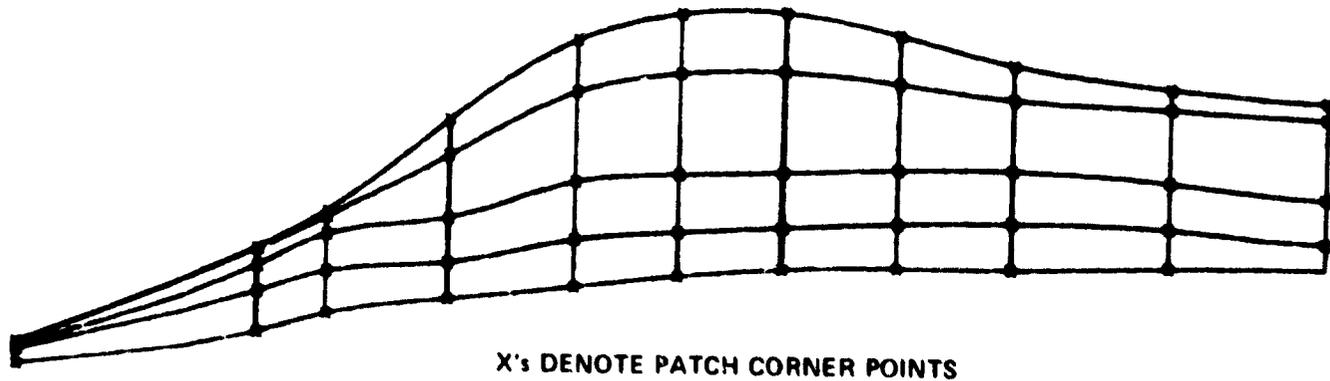


Figure 14. AD-2000 PC surface with separate patch corner points.

TEST CASE

Several geometries were carried through the complete system to test it for continuity and simplicity. Two test geometries were ultimately machined, one of which will be described here. The process can be divided into six phases: (1) geometry selection, (2) data transfer, (3) data conversion, (4) geometry reconstruction in AD-2000, (5) NC code generation, and (6) NC machining.

Geometry Selection

The machining of an aircraft component surface was considered to be sufficient to test the system. It was assumed that fittings and other attachments on the component could be added later.

The test geometry described here is the forebody of the aircraft configuration shown in figures 4 and 5. It included the canopy and a portion of the forward fuselage section. The forebody was selected because it was a relatively complex surface and could be easily separated from the complete aircraft geometry. The test geometry is shown in wire mesh form in figure 15 and in shaded surface form in figure 16.

Only half of the forebody is shown in figures 15 and 16 because that geometry was all that was necessary to describe the part. The other half is created for display and modeling by reflection about the aircraft's natural plane of symmetry. Furthermore, only one half of the forebody is necessary for NC machining because the other half is easily cut using the "mirror" feature available on most NC machines.

Data Transfer

The data transfer phase involves sending the IPEGS patch data to a location accessible by an NC programmer using

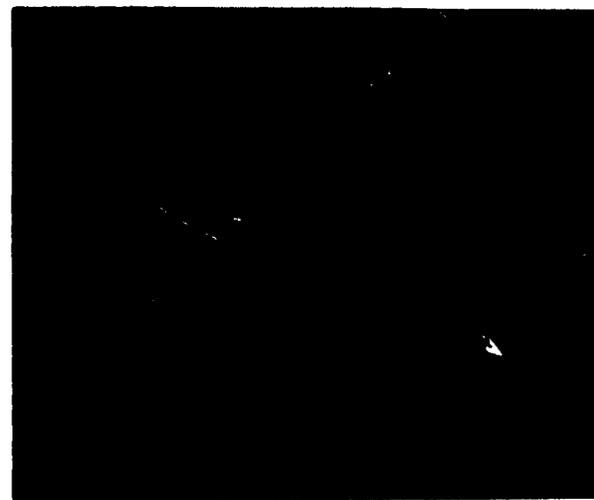


Figure 15. - IPEGS display of test geometry.



Figure 16. Shaded surface display of test geometry.

ORIGINAL PAGE IS
OF POOR QUALITY

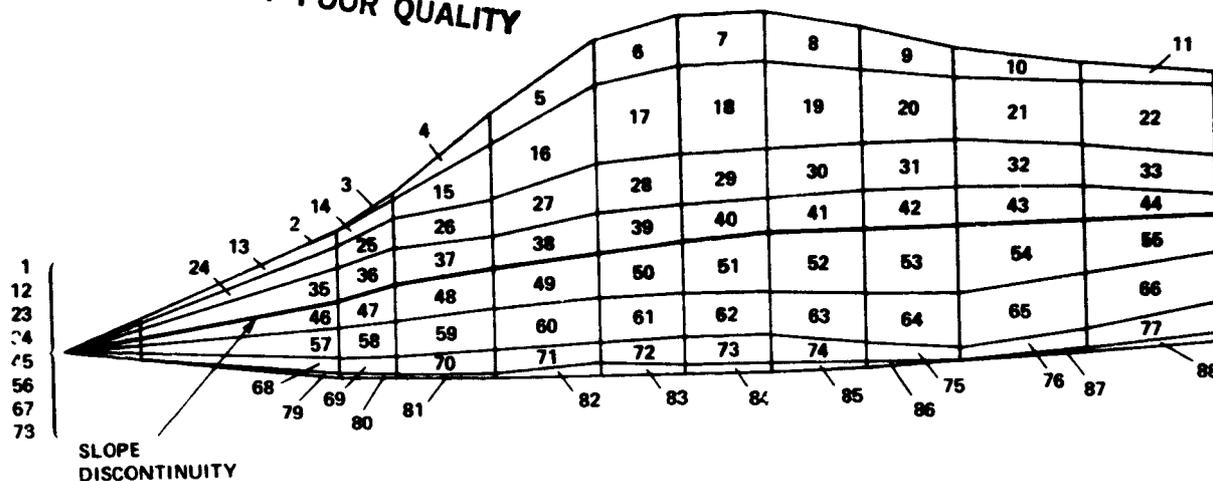


Figure 17.— Topological map of test geometry.

AD-2000. Before this can be done, a topological map of the patches must be produced using IPEGS (fig. 17). After generating this map, the patch data are then written to a standard IPEGS output patch file in the form outlined in table 1. This file is transferred through the computer network to the computer used by machine shop personnel.

Data Conversion

In the data conversion phase, the NC programmer creates a set of AD-2000-input patch files from the IPEGS file by dividing the surface into patch strips using the machine shop's graphics terminal and the IPEGS/AD-2000 interface. The separation of the geometry into patch strips depends on the specific machining techniques to be used. Therefore, these techniques must be determined before the interface is executed.

There were two machining decisions to be made with respect to the test geometry. The first decision resulted from the sharp slope discontinuity, caused by the leading edge of the wing strake, which ran lengthwise along the forebody midway between the fourth and fifth rows of patches (fig. 17). Since a three-axis machine was to be used, the best solution was to cut the forebody into upper and lower sections, with the discontinuity between the two sections. This was a convenient cut because the discontinuity was on the boundary between two patch strips. Furthermore, the cutter was to be driven in a lattice pattern lengthwise along the forebody, beginning at the extreme edge of the surface and working toward the line of discontinuity. This method was used for both the upper and lower sections as illustrated in figure 18.

The second decision was to machine all but the tip of the forebody, and to fair the tip in later by hand. This decision was made for two reasons. First, the cutter would have freed

the tip from the surrounding material before machining was complete, causing undesirable vibration and possible damage. Second, AD-2000 has difficulty computing cutter paths to the vertex of a conic surface. The small cluster of patches at the tip of the forebody were omitted for this purpose.

These machining considerations led to dividing the forebody into eight separate AD-2000 patch files, each containing a longitudinal strip of patches. The IPEGS patch file of the complete forebody was first separated into two IPEGS files using the standard editor available on the computer system. The first file, labeled UQNOSE.GEO, contained the portion of the forebody above the line of discontinuity. The second, labeled LQNOSE.GEO, contained the lower portion.

The execution of the interface program is shown in figure 19. This program was first executed for the upper forebody section, thus UQNOSE.GEO was specified as the IPEGS patch file to be converted. A simplified topological map of this section is shown in figure 20. The program then counted the number of patches on the file.

A name for the AD-2000 file that would contain the first set of patches extracted from UQNOSE was requested next. The file for the first patch strip was called U1.NMG. The uppermost lengthwise strip of patches, 2-11, were selected. (Patches 1, 12, 23, and 34 were at the tip and were therefore omitted.) This group of patches was specified sequentially since it did not violate the continuity requirement. The "random" mode would have been selected if the geometry were to be divided into column strips (the first set consisting of patches 2, 13, 24, and 35), and each patch would have been specified individually.

The next prompt in the program requested the patch direction, or the assumed orientation of the patch-local axis system in the IPEGS file. Specifying the "X" direction tells the interface to perform no reorientation; specifying "Y" will reorient the patches. This prompt is given only once because the program also assumes that all the patches in the

ORIGINAL PAGE IS
OF POOR QUALITY

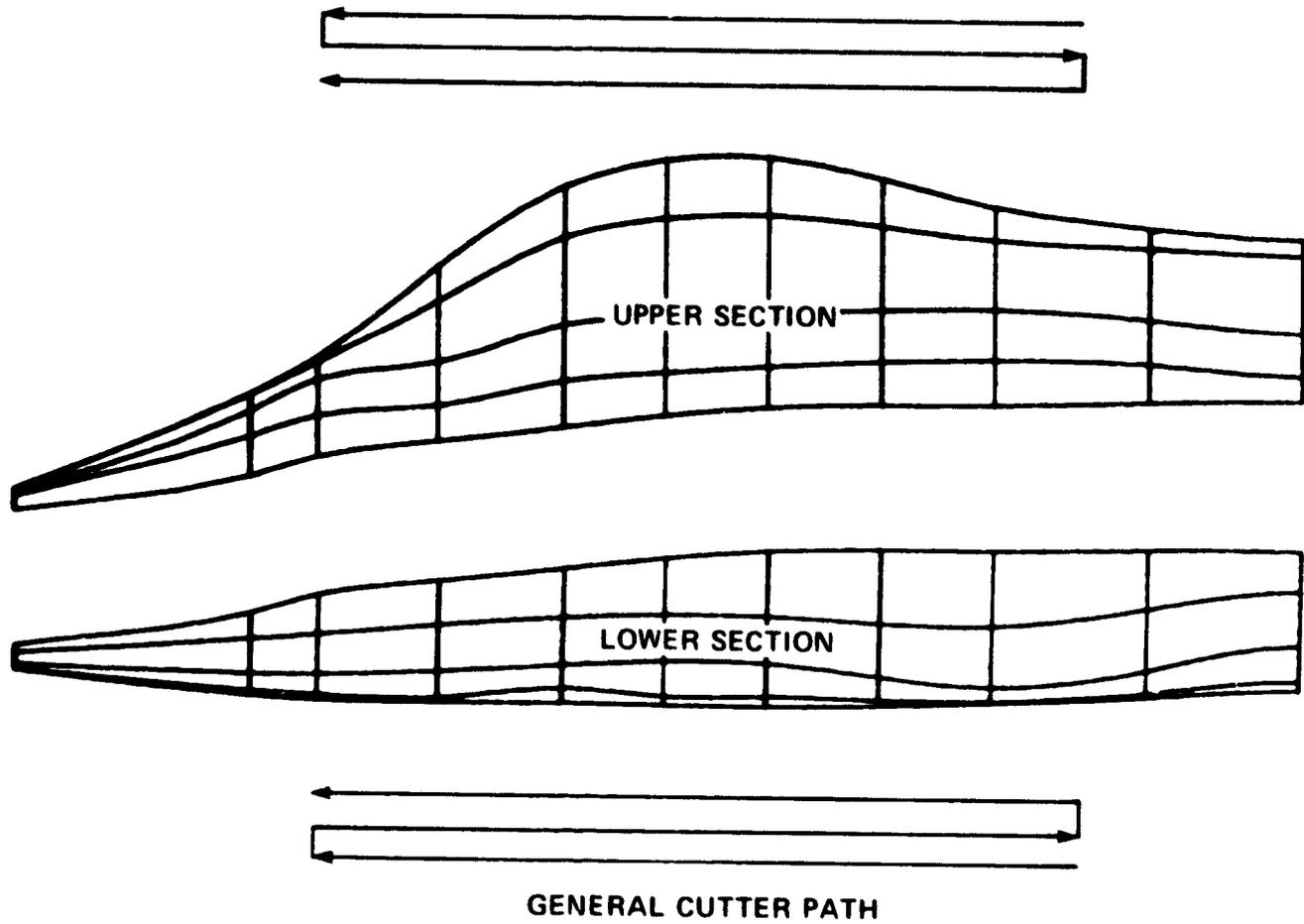


Figure 18.-- Machining test geometry approach.

ORIGINAL PAGE IS
OF POOR QUALITY

```
8 RUN CITA
INPUT FILE 'name.GEO' U0NOSE.GEO
DO YOU WANT A PATCH COUNT MAIN FILE? Y
NUMBER OF PATCHES IN FILE- 44

OUTPUT FILE 'name.NMG' U1.NMG
SCALE FACTOR- 1.0
Selection can be Sequentially or Random
(ENTER S or R) S
PATCHES N1 thru N2 2,11
(Enter X or Y) FOR DIRECTION OF SURFACE PATCHES X
NUMBER OF PATCHES- 10
CONTINUE (Y OR N) ?Y
OUTPUT FILE 'name.NMG' U2.NMG
SCALE FACTOR- 1.0
Selection can be Sequentially or Random
(ENTER S or R) S
PATCHES N1 thru N2 13,22
NUMBER OF PATCHES- 10
CONTINUE (Y OR N) ?Y
OUTPUT FILE 'name.NMG' U3.NMG
SCALE FACTOR- 1.0
Selection can be Sequentially or Random
(ENTER S or R) S
PATCHES N1 thru N2 24,33
NUMBER OF PATCHES- 10
CONTINUE (Y OR N) ?Y
OUTPUT FILE 'name.NMG' U4.NMG
SCALE FACTOR- 1.0
Selection can be Sequentially or Random
(ENTER S or R) S
PATCHES N1 thru N2 35,44
NUMBER OF PATCHES- 10
CONTINUE (Y OR N) ?N
FORTRAN STOP
```

```
8 DIR/SIZE/DATE U*.NMG
```

```
Directory F:\01EGUEST\
```

U1.NMG;1	21	10-DEC-1982	17:22
U2.NMG;1	21	10-DEC-1982	17:23
U3.NMG;1	21	10-DEC-1982	17:24
U4.NMG;1	21	10-DEC-1982	17:24

```
Total of 4 files, 84 blocks.
```

```
8
```

Figure 19 Interface program interaction listing for upper forebody section

ORIGINAL PAGE IS
OF POOR QUALITY

1	2	3	4	5	6	7	8	9	10	11
12	13	14	15	16	17	18	19	20	21	22
23	24	25	26	27	28	29	30	31	32	33
34	35	36	37	38	39	40	41	42	43	44

Figure 20.- Simplified topological map of upper forebody section.

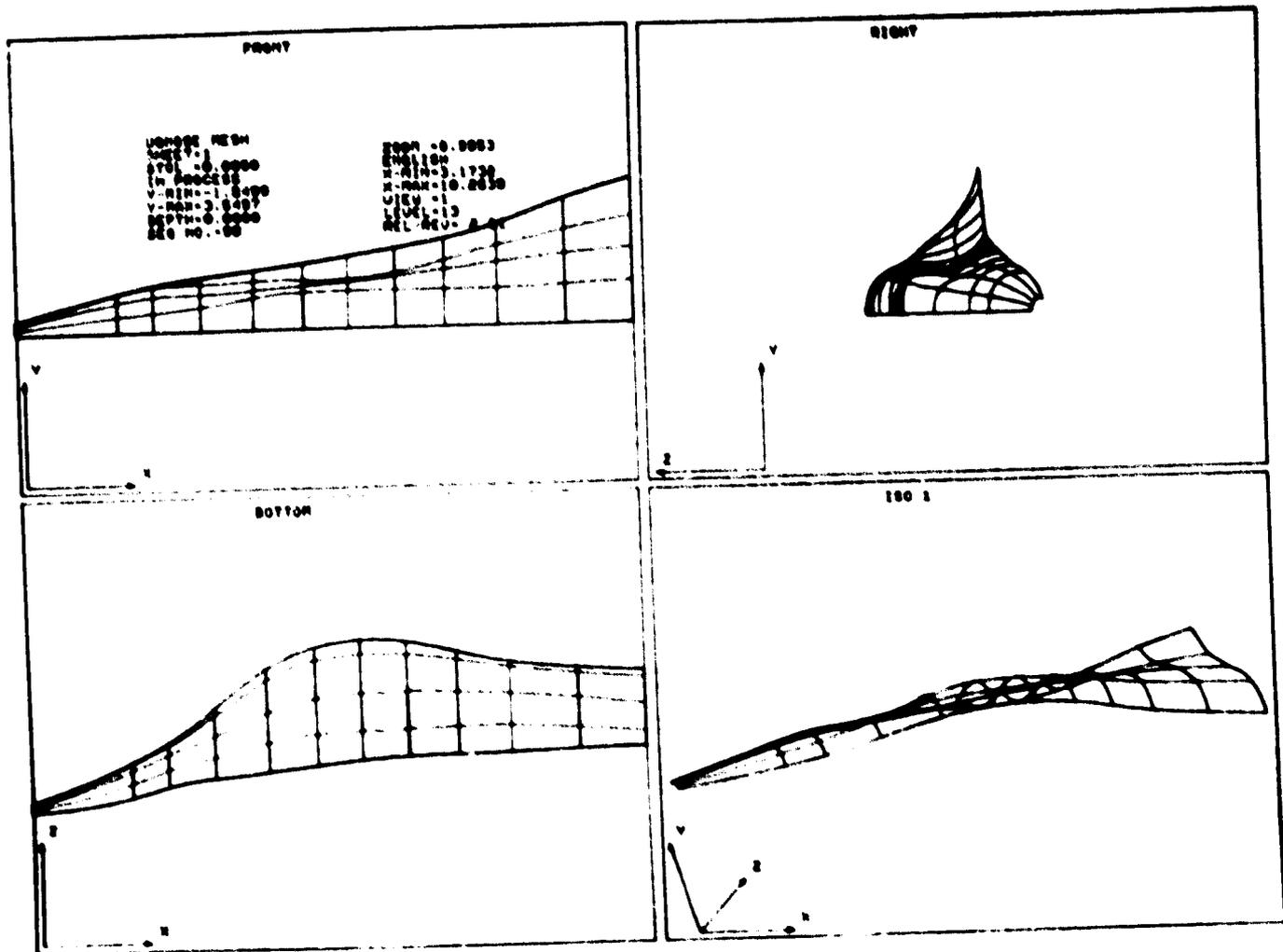


Figure 21. AD-2000 display of upper forebody section.

ORIGINAL PAGE IS OF POOR QUALITY

IPEGS file are consistently oriented. The X-direction was specified in this case because it was assumed (for the first try) that the IPEGS patches were correctly oriented for this type of division.

The selected patches were then written to the specified file in the proper format by the interface. The total number of patches selected was then displayed.

The three other longitudinal strips of patches were then processed and placed in files labeled U2, U3, and U4.NMG. The directory listing (fig. 19) verified that the specified AD-2000 input patch files were properly generated. (Each file contains 10 patches.) The entire procedure was then repeated for the lower surface (LQNOSE.GEO).

Geometry Reconstruction in AD-2000

In the geometry reconstruction phase, the individual patch strips were brought into the AD-2000 system and reconnected to form the original single surface. The upper surface was reconstructed first.

The AD-2000 system was used to bring in the four patch strips for display (fig. 21). Since the display appeared to be correct, the assumption about the original patch orientation was therefore correct.

The next step normally combines the separate surfaces into one composite surface using standard AD-2000 functions. However, since a high-tolerance cutter path (which requires more tool commands than a low-tolerance cutter path) was to be generated for this composite surface, the NC data would have been too numerous to process into a single punch tape for the NC machine. Therefore strips 1 and 2 were combined into one composite surface, and strips 3 and 4 were combined into another. This approach allowed two separate cutter paths to be generated for the upper forebody section, each of which could be punched on a single NC tape.

The reconstructed upper forebody section was then stored as a "part" within the AD-2000 system, and the procedures in this phase were repeated for the lower forebody section. The complete forebody was thus composed of four parts.

NC Code Generation

The NC code generation phase computes and files the data needed to drive an NC cutter to machine the surface. The upper forebody section was again considered first.

The machining function in AD-2000 requests certain information from the NC programmer, such as cutter diameter, tolerance, and maximum allowable scallop height (a measure of the height above the true surface of the excess material between the cutter paths). A 0.5-in. ball cutter was selected to cut the forebody section within a tolerance of 0.001 in. The maximum scallop height was specified to be

0.002 in. Two cutter paths, one for each of the two upper surfaces, were then generated and displayed. A close-up view of a portion of these cutter paths is shown in figure 22, where the dashed lines represent the position of the center of the cutter on the path.

An indication that the cutter would violate the surface in this region is seen near the middle of the surface where several of the paths crossed each other. The reason for this is that the radius of the surface curvature in this area was smaller than the radius of the cutter. The cutter, in other words, was too large to fit into this contour.

The standard solution to this problem is first to construct a check surface, or boundary line, around the problem area. The original cutter (in this case, 0.5 in.) is used for the region outside the check surface. A smaller cutter is used inside the check surface. AD-2000, however, cannot create check surfaces on PC surfaces of this complexity. (Another solution, to use the smaller cutter over the entire surface, results in an excessive amount of NC code and a correspondingly long machining time.)

The solution was to isolate the patches in the problem area from the patches on the rest of the surface. The upper forebody section was reconstructed (using the IPEGS/AD-2000 interface program) as a set of three separate surfaces. The isolated region could then be machined using the smaller cutter.

Figure 23 shows the two cutter paths generated for the outer region using the 0.5-in. ball cutter. Figure 24 shows a magnified view of the inner (isolated) region with a cutter path generated using a 0.125-in. ball cutter. Some crossover is still evident, but it is considered negligible.

These procedures then generated two cutter paths for the lower forebody section and no radius-of-curvature problems were encountered. Finally, all five cutter paths for the complete forebody section were stored in files outside AD-2000.

NC Machining

The five cutter-path files were written on magnetic tape for processing into five NC punch tapes. Machine setup sketches were drawn and given to the NC machinist along with the punch tapes and corresponding listings, and the test geometry was then machined. The resulting forebody machined in aluminum is shown in figure 25.

CONCLUSIONS

The research program to integrate an in-house graphics system with a commercially available CAD/CAM system revealed that (1) substantial time can be saved using the IPEGS/AD-2000 interface, and (2) the accuracy of the

ORIGINAL PAGE IS
OF POOR QUALITY

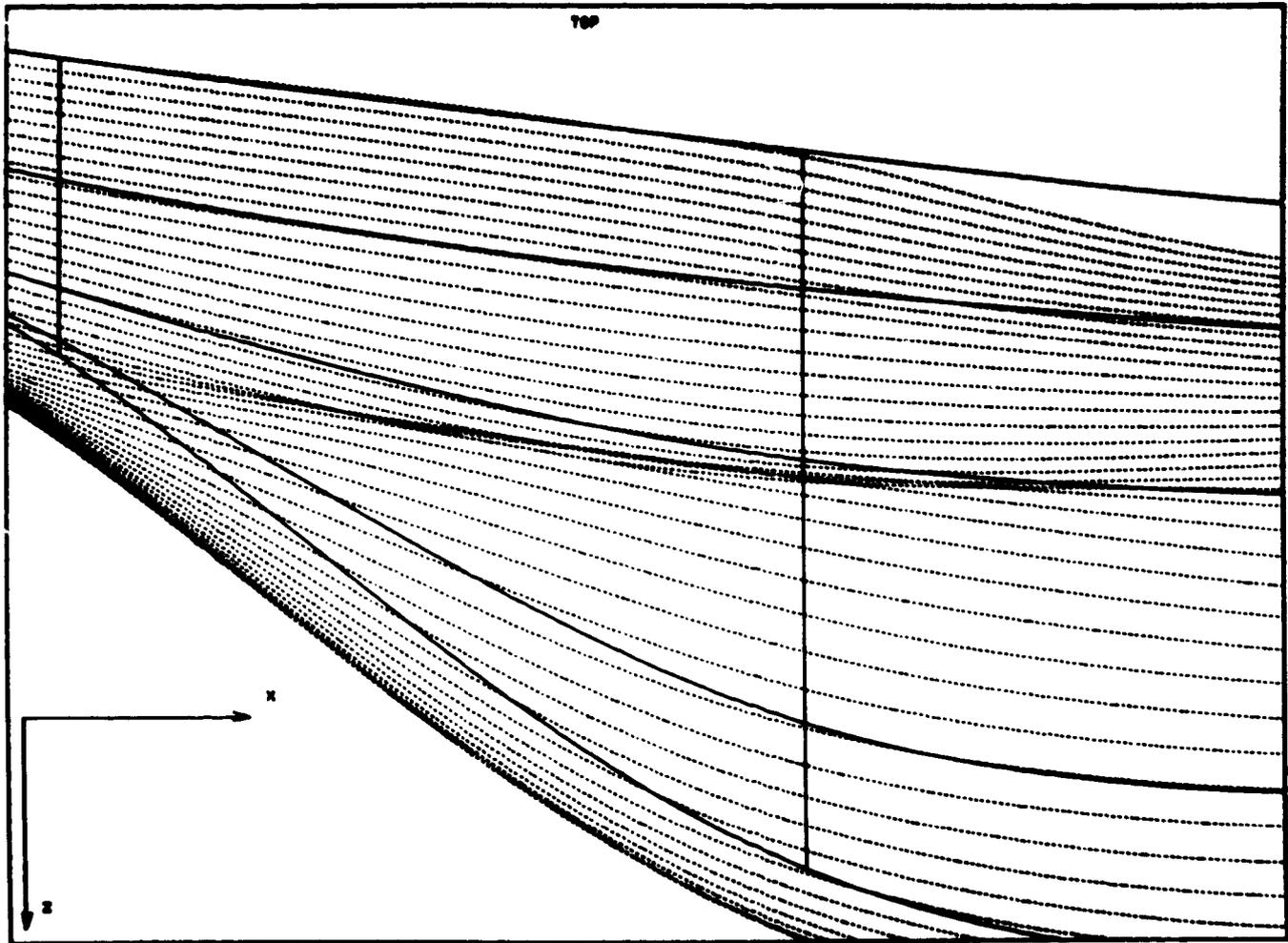


Figure 22.— Close-up view of cutter paths showing path crossover.

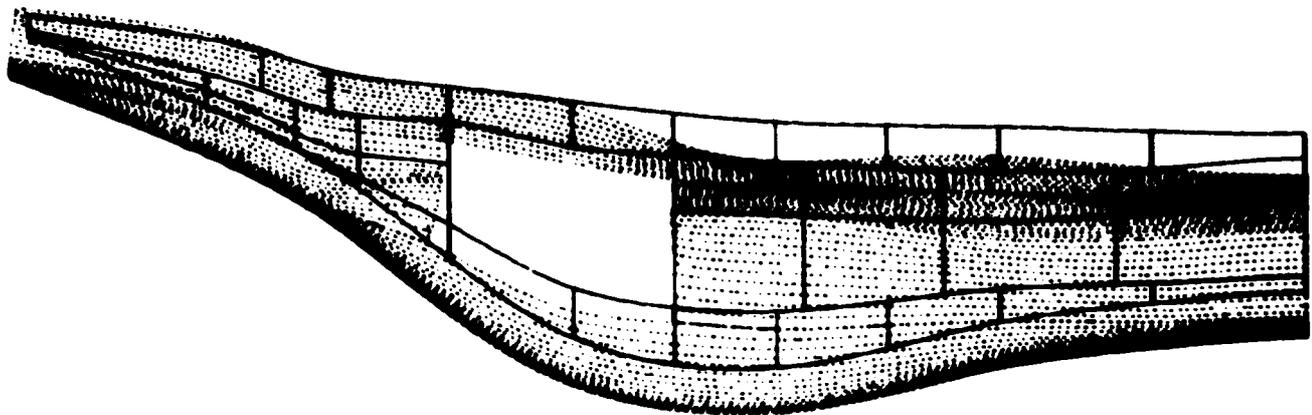


Figure 23.— Cutter paths for outer region of redefined upper nose section.

ORIGINAL PAGE 13
OF POOR QUALITY

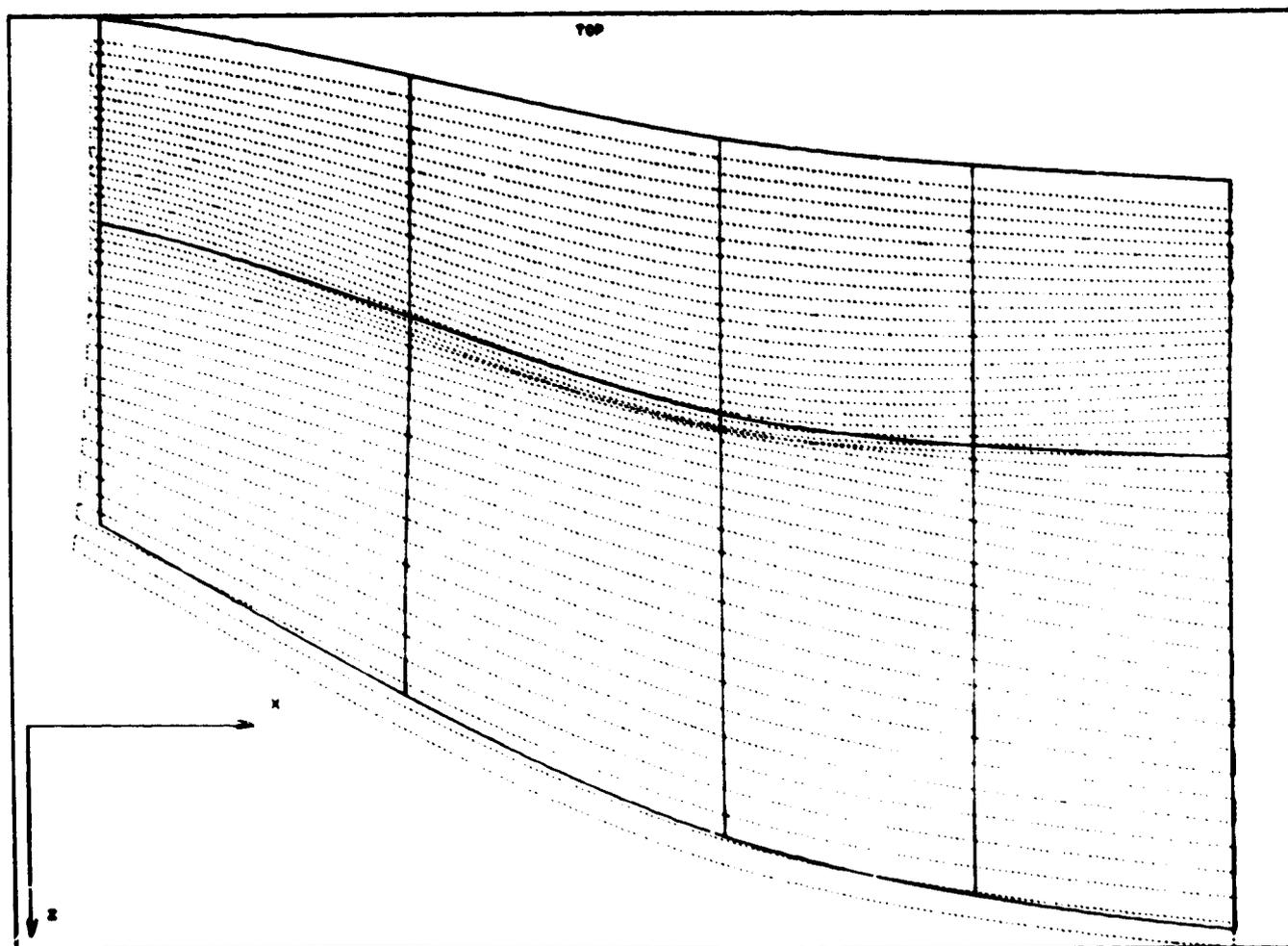


Figure 24. — Cutter paths with inner (isolated) region of redefined upper nose section.

ORIGINAL PAGE IS
OF POOR QUALITY



Figure 25.—Machined test geometry.

machined surfaces is acceptable. However, the system is unacceptably complex and cumbersome to operate. Because operational difficulties outweighed the advantages of time saving and accuracy, a recommendation was made to procure a different, independent CAD/CAM system that had the necessary IPEGS and AD-2000 capabilities.

The use of the IPEGS/AD 2000 interface to machine IPEGS geometries saves a substantial amount of time as compared to the previous process. The amount of time saved, however, is difficult to assess quantitatively because wind-tunnel models are not fabricated on a regular production-oriented basis. About 2-3 weeks was saved in fabricating the test geometry, but the time saved would vary with different geometries.

The accuracy of the surfaces machined using the new method was acceptable for most applications. The total accuracy of the new system is the sum of the accuracies of the PC surface, the AD-2000-generated cutter path, and the

NC machine. These and other factors combine so that a machined and finished PC surface can be expected to fall within 0.008 in. of the original PC surface. This estimate was verified by measuring the coordinates of a set of points on the machined surface and comparing them with corresponding points on the mathematical model.

A major disadvantage of the new process compared to the previous process is that it is cumbersome to use. It takes time for the NC programmers to learn how to operate the IPEGS/AD-2000 interface proficiently. In addition, the process requires extensive input from both the aerodynamicist and the NC programmer to make IPEGS patch files compatible with AD-2000.

Ames Research Center
National Aeronautics and Space Administration
Moffett Field, California, November 23, 1982

PARAMETRIC BICUBIC SURFACE THEORY
(Adapted from ref. 3)

A continuous, three-dimensional surface can be expressed as a general transformation of the form

$$x = f(u,w)$$

$$y = g(u,w)$$

$$z = h(u,w)$$

with domain D in the u - w plane. From this expression, the specific equation for a parametric bicubic (PC) surface patch can be derived, and is written:

$$P(u,w) = [u^3 u^2 u \ 1] [M] [B] [M]^T \begin{bmatrix} w^3 \\ w^2 \\ w \\ 1 \end{bmatrix} \quad (1)$$

where P represents x , y , or z .

$$[M] = \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

and

$$[B] = \begin{bmatrix} P_{00} & P_{01} & P_{00w} & P_{01w} \\ P_{10} & P_{11} & P_{10w} & P_{11w} \\ P_{00u} & P_{01u} & P_{00uw} & P_{01uw} \\ P_{10u} & P_{11u} & P_{10uw} & P_{11uw} \end{bmatrix}$$

The physical interpretation of equation (1) is illustrated in figure A1. Equation (1) is the so-called "geometric" form of the expression for a PC patch.

The matrix B is called the boundary matrix, and its elements are the PC patch coefficients in geometric form. Each of these coefficients represents a physical characteristic of the patch, as signified by its notation. Numerical subscripts

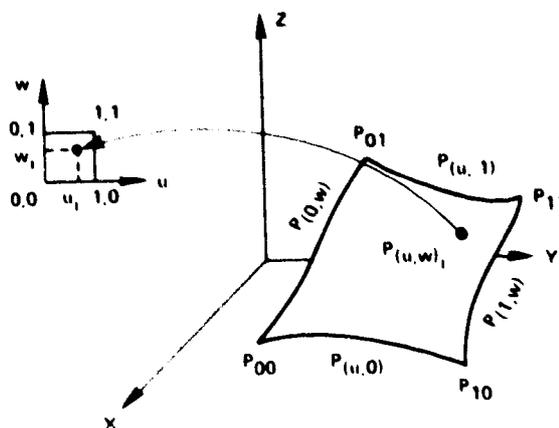


Figure A1. - Relationship between u - w and x - y - z coordinate systems.

indicate specific values of u and w (corner points), and the explicit subscripts u and w indicate that the coefficient is a derivative with respect to that variable. For example:

$$P_{01} = [P(u,w)]_{\substack{u=0 \\ w=1}} : \text{point}$$

$$P_{10w} = \left[\frac{\partial P(u,w)}{\partial w} \right]_{\substack{u=1 \\ w=0}} : \text{first-derivative or slope}$$

$$P_{11uw} = \left[\frac{\partial^2 P(u,w)}{\partial u \partial w} \right]_{\substack{u=1 \\ w=1}} : \text{cross-derivative or "twist"}$$

Thus, the matrix B embodies the geometric character of the patch. It can be seen that this matrix can be partitioned into four 2×2 submatrices and expressed as follows:

$$[B] = \begin{bmatrix} P & | & S_w \\ S_u & | & T \end{bmatrix}$$

where P (in this case) is the submatrix of point data at the four corner points; S_u and S_w contain the slopes, or first derivatives with respect to u and w at those points; and T contains cross-derivative, or "twist" data, which govern the character of the interior of the patch.

**ORIGINAL PAGE IS
OF POOR QUALITY**

Equation (1) can also be written in "algebraic" form:

$$P(u, w) = [u^3 \ u^2 \ u \ 1] [S] \begin{bmatrix} w^3 \\ w^2 \\ w \\ 1 \end{bmatrix}$$

where

$$[S] = [M] [B] [M]^T$$

The matrix S is called the surface matrix, and its elements are the PC patch coefficients in algebraic form. The coefficients in this form have no physical interpretation.

ORIGINAL PAGE IS
OF POOR QUALITY

REFERENCES

1. Axelson, J. A.: AEROX - A Computer Program for Transonic Aircraft Aerodynamics to High Angles of Attack. Volumes I, II, and III. NASA TM X-73208, 1977.
2. Carmichael, R. L.; and Erickson, L. L.: PAN AIR - A Higher Order Panel Method for Predicting Subsonic or Supersonic Linear Potential Flows about Arbitrary Configurations. AIAA Paper 81-1255, 1981.
3. Peters, G. J.: Interactive Computer Graphics Application of the Parametric Bi-cubic Surface to Design Problems. Computer Aided Design, R. E. Barnhill and R. F. Riesenfeld, eds., Academic Press, Inc., New York, 1974, pp. 259-302.
4. Roland, D. P.: Parametric Cubic Surface Representation. Proc. Workshop on Aircraft Surface Representation for Aerodynamic Computation. NASA TM 81170, March 1978.
5. Manufacturing and Consulting Services, Inc.: AD-2000 Training Manual, Release 2/Revision 1. Manufacturing and Consulting Services, Inc., Santa Ana, California, 1980.
6. Manufacturing and Consulting Services, Inc.: AD-2000 Reference Manual, Release 2/Revision 1. Manufacturing and Consulting Services, Inc., Santa Ana, California, 1980.