

AD-778 444

THE MARK IV SUPERSONIC-HYPERSONIC
ARBITRARY-BODY PROGRAM. VOLUME II.
PROGRAM FORMULATION

Arvel E. Gentry, et al

Douglas Aircraft Company

Prepared for:

Air Force Flight Dynamics Laboratory

November 1973

Reproduced From
Best Available Copy

DISTRIBUTED BY:

NTIS

National Technical Information Service
U. S. DEPARTMENT OF COMMERCE
5285 Port Royal Road, Springfield Va. 22151

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER AFFDL-TR-73-159 VOLUME II	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER AD 778 444
4. TITLE (and Subtitle) THE MARK IV SUPERSONIC-HYPERSONIC ARBITRARY-BODY PROGRAM, VOLUME II PROGRAM FORMULATION		5. TYPE OF REPORT & PERIOD COVERED Final
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Arvel E. Gentry Douglas N. Smyth Wayne R. Oliver		8. CONTRACT OR GRANT NUMBER(s) F33615-72-1675
9. PERFORMING ORGANIZATION NAME AND ADDRESS Douglas Aircraft Company, McDonnell Douglas Corp. 3855 Lakewood Blvd. Long Beach, California 90846		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Air Force Flight Dynamics Laboratory AFFDL/FXG Wright-Patterson AFB, Ohio		12. REPORT DATE November 1973
		13. NUMBER OF PAGES 228
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE N/A
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited.		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES Reproduced by NATIONAL TECHNICAL INFORMATION SERVICE U S Department of Commerce Springfield VA 22151		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number)		
Aerodynamics Arbitrary Three-Dimensional Shapes	Supersonic Hypersonic Flow Computers (Digital)	FORTAN Graphics
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) This report describes a digital computer program system that is capable of calculating the supersonic and hypersonic aerodynamic characteristics of complex arbitrary three-dimensional shapes. This program is identified as the Mark IV Supersonic-Hypersonic Arbitrary-Body Computer Program. This program is a complete reorganization and expansion of the old Mark III Hypersonic Arbitrary-Body Program. The Mark IV program has a number of new capabilities that extend its applicability down into the supersonic speed range.		

20. ABSTRACT (continued)

The outstanding features of this program are its flexibility in covering a very wide variety of problems and the multitude of program options available. The program is a combination of techniques and capabilities necessary in performing a complete aerodynamic analysis of supersonic and hypersonic shapes. These include: vehicle geometry preparation; computer graphics to check out the geometry; analysis techniques for defining vehicle component flow field effects; surface streamline computations; the shielding of one part of a vehicle by another; calculation of surface pressures using a great variety of pressure calculation methods including embedded flow field effects; and computation of skin friction forces and wall temperature.

Although the program primarily uses local-slope pressure calculation methods that are most accurate at hypersonic speeds, its capabilities have been extended down into the supersonic speed range by the use of embedded flow field concepts. This permits the first order effects of component interference to be accounted for.

The program is written in FORTRAN for use on CDC or IBM types of computers.

The program is documented in three volumes. Volume I is primarily a User's Manual, Volume II gives the Program Formulation, and Volume III contains the Program Listings.

ia

NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

ACCESSION for	
NTIS	Write Center <input checked="" type="checkbox"/>
DCC	ERIC <input type="checkbox"/>
USAM SUBJECT	
JUSTIFICATION	
BY	
DISSEMINATION TO	
DATE	
A	

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.

ib

**THE MARK IV SUPERSONIC-HYPERSONIC
ARBITRARY-BODY PROGRAM.**

VOLUME II -- PROGRAM FORMULATION

*ARVEL E. GENTRY
DOUGLAS N. SMYTH
WAYNE R. OLIVER*

DOUGLAS AIRCRAFT COMPANY
MCDONNELL DOUGLAS CORPORATION

Approved for public release; distribution unlimited.

FOREWORD

This report describes a computer program developed at the Douglas Aircraft Division of the McDonnell Douglas Corporation, Long Beach, California. The development of the Douglas Arbitrary-Body Aerodynamic Computer Program was started in 1964 and greatly expanded in subsequent years under sponsorship of the Douglas Independent Research and Development Program (IRAD). From August 1966 to May 1967 the program development was continued under Air Force Contract No. F3361567-C-1008. The product of this work was the Mark II version of the program as released for use by government agencies in May 1967. Between 1967 and 1968 further Douglas IRAD work and another Air Force Contract (F33615-67-C1602) produced the Mark III Hypersonic Arbitrary-Body Program. The latest version of the program as presented in this report is identified as the Mark IV Supersonic-Hypersonic Arbitrary-Body Computer Program and was prepared in the period of 1972-73 under Air Force Contract F33615-72-C-1675. This contract was administered by the Air Force Flight Dynamics Laboratory, Flight Mechanics Division, High Speed Aero Performance Branch. The Air Force Project Engineers for this study were Verle V. Bland Jr., and Captain Hugh Wilbanks, AFFDL/FXG.

At the Douglas Aircraft Company, this work was conducted under the direction of Mr. Arvel D. Gentry as Principal Investigator. A number of major parts of the new program were prepared by Mr. Douglas N. Smyth. Mr. Wayne R. Oliver's work in applying the various versions of this program to practical design problems contributed both in program design and in program validation. A number of other people contributed to the various phases of this work for which the authors are grateful.

This report was submitted by the authors in November 1973.

This technical report has been reviewed and is approved.



PHILIP P. ANTONATOS

Chief, Flight Mechanics Division
Air Force Flight Dynamics Laboratory

ABSTRACT

This report describes a digital computer program system that is capable of calculating the supersonic and hypersonic aerodynamic characteristics of complex arbitrary three-dimensional shapes. This program is identified as the Mark IV Supersonic-Hypersonic Arbitrary-Body Computer Program. This program is a complete reorganization and expansion of the old Mark III Hypersonic Arbitrary-Body Program. The Mark IV program has a number of new capabilities that extend its applicability down into the supersonic speed range.

The outstanding features of this program are its flexibility in covering a very wide variety of problems and the multitude of program options available. The program is a combination of techniques and capabilities necessary in performing a complete aerodynamic analysis of supersonic and hypersonic shapes. These include: vehicle geometry preparation; computer graphics to check out the geometry; analysis techniques for defining vehicle component flow field effects; surface streamline computations; the shielding of one part of a vehicle by another; calculation of surface pressures using a great variety of pressure calculation methods including embedded flow field effects; and computation of skin friction forces and wall temperature.

Although the program primarily uses local-slope pressure calculation methods that are most accurate at hypersonic speeds, its capabilities have been extended down into the supersonic speed range by the use of embedded flow field concepts. This permits the first order effects of component interference to be accounted for.

The program is written in FORTRAN for use on CDC or IBM type computers.

The program is documented in three volumes. Volume I is primarily a User's Manual, Volume II gives the Program Formulation, and Volume III contains the Program Listings.

TABLE OF CONTENTS

	Page
SECTION I	
INTRODUCTION	1
SECTION II	
PROGRAM FRAMEWORK	4
SECTION III	
GEOMETRY	6
The Surface Element Geometry Method	7
Parametric Cubic	16
Aircraft Geometry Option	23
Pods or Nacelles	23
General Airfoil Surfaces	26
Control Surface Geometry	30
SECTION IV	
GENERAL INTERPOLATION METHOD	34
Surface Spline	34
Linear Spline	36
Symmetry	37
Application of the Surface Spline for Interpolation	39
Surface Data Normalization	40
Bodies	41
Lifting Surface	42
Flow Field Data Normalization	43
SECTION V	
FLOW FIELD ANALYSIS METHODS	46
Component Flow Field Analysis	48
Shock-Expansion Method	50
Surface Pressure	51
Shock Wave Shapes	56
Flow Field Calculation	79

Preceding page blank

SECTION VI

SHIELDING EFFECTS 91

SECTION VII

COMPUTATION OF VEHICLE FORCES 109

Calculation of Local Flow Conditions 109
Coefficient Transformations 112
Surface Velocity Vector 113
Pressure Coefficient Corrections 114

SECTION VIII

INVISCID PRESSURE METHODS 117

Basic Pressure Calculation Methods 118
 Modified Newtonian 119
 Modified Newtonian Plus Prandtl-Meyer 120
 Tangent-Wedge 121
 Tangent-Wedge and Delta Wing
 Newtonian Empirical Method 125
 Tangent Cone 134
 Inclined Cone 137
 Use as an Element Impact Method 148
 Van Dyke Unified Method 149
 Shock-Expansion Method 149
 Free Molecular Flow Method 150
 Hankey Flat-Surface Empirical Method 151
 Modified Dahlem-Buck 151
 Blast Wave Pressure Increments 152
The Stored Pressure Option 153

SECTION IX

STREAMLINE CALCULATIONS 154

SECTION X

VISCOUS FORCE CALCULATION METHODS 160

The Integral Boundary Layer Method 161
Mark III Skin Friction Option 162
 Skin Friction Geometry Model 163

Local Flow Conditions	165
Incompressible Flow	165
Compressible Flow	166
Surface Equilibrium Temperature	167
Real Gas Effects	169
Viscous-Inviscid Interaction	173
Viscous Force on Blunt Bodies	185

SECTION XI

AUXILIARY ROUTINES	191
General Cutting Plane Option	191

SECTION XII

COMPUTER GRAPHICS	201
Picture Drawing Methods	202

REFERENCES	214
----------------------	-----

SECTION I

INTRODUCTION

The basic objective of this work was to provide a theoretical analysis tool for use in studying the aerodynamic characteristics of vehicles operating at speeds from about Mach 2 on up into the hypersonic range. This program was to be capable of predicting the aerodynamic characteristics of arbitrary wing-body-fin configurations including the determination of embedded flow region effects, the effects of wing-body and wing-fin interference, and give improved viscous flow results. One key requirement was that the geometry data input be compatible with the Mark III Hypersonic Arbitrary-Body Aerodynamic Computer Program.

The basic tenet of this project was that it employ "engineering methods" that represent a realistic modeling of the actual flow about a shape. The basic guide line was that the program produced should be a flexible engineering tool, usable by the designer in day-to-day design and development work, rather than a specialized research program requiring extensive knowledge for successful operation and large amounts of computer time.

In addition to the above it was desirable that the new program retain as many of the capabilities of the old Mark III Hypersonic Arbitrary-Body Program as possible. This would make the program equally applicable to interceptors and fighters and to space shuttle vehicles.

The result of this work in response to these objectives is the Mark IV Supersonic-Hypersonic Arbitrary-Body Program. To a certain extent, the Mark IV program is a re-structuring of the old Mark III program. It does, of course, make extensive use of code from the Mark III program. Moreover, the geometry decks prepared for the Mark III program are still directly usable on the new program. However the framework for the Mark IV program differs from the old program in that each basic type of analysis is accomplished in a separate program component. Each of the major program functions are placed in separate components with the interface between components provided by an executive routine and access to appropriately stored and saved data.

The executive routine controls the order of calls to the Geometry, Aero, Graphics, and Auxiliary Routines. The Geometry component has all of the capabilities of the Mark III Mod 3 Hypersonic Arbitrary-Body Program. These include input element, ellipse generation, parametric cubic, and the Aircraft Geometry Option.

The Aero part of the program contains six major independent components: Flow Field Analysis, Shielding, Inviscid Pressures, Streamline Analysis, Viscous Methods, and Special Routines. Each of these components

generates data that is saved on storage units for subsequent use by other components. Because of this new framework for the program all of the input data to the program (except for the geometry data) is different from that used on the Mark III program.

The Flow Field Analysis component is one of the key new capabilities of the program. With it we can generate and store the external flow field of a vehicle component. This flow field can then be retrieved by the force part of the program and used to define the incident flow conditions for another component. In this way we can account for the first order interference effects between different parts of a vehicle. A surface spline method is used to interpolate data within the flow field and for several other purposes within the program.

The Shielding component also provides a new capability in the Mark IV program. This option may be used to account for the shielding from the external impact flow of one part of a vehicle by another part.

The new viscous parts of the program provide the capability of calculating skin friction properties using an integral boundary layer program. These computations are performed using external flow properties along the program calculated surface streamlines.

The first Douglas arbitrary body program was started in July 1964, almost ten years ago. The objective of the program at that time was to fill the aerodynamic analysis gap that existed between the linear theory methods (for simple shapes and low supersonic speeds), and the detailed gas dynamic solutions using the method of characteristics or finite difference techniques (simple shapes and very long computer times). Linear theory methods have been improved considerably in the past ten years, but they still cannot handle completely arbitrary shapes and they do not account for the non-linear effects as Mach number increases. Also, the detailed gas dynamic solutions still require too much machine time for them to be classified as tools useful in the many day-to-day studies in most vehicle design and evaluation efforts.

The Mark IV Supersonic-Hypersonic Arbitrary-Body Program is provided as an engineering rather than a research tool. As such, the accuracy of its results should not be expected to be as good as some of the more exact methods (when applied to shapes and conditions where they are specifically designed for). However, when solving problems outside the range of the linear or more exact methods, or when studying complex arbitrary shapes, the Mark IV program should produce very useful results.

Throughout this report it will be assumed that the reader is familiar with the contents of Volume I, the User's Manual. Discussions of earlier versions of this program are given in References 1 and 2.

This report contains descriptions of the analysis techniques used within the program. Throughout these discussions an attempt has been made

to maintain mathematical notations consistent with the appropriate reference involved. This will assist the reader in comparing the approaches with the original reference material at some slight loss in continuity within the present report. This policy has also been used in the selection of many of the program variable names.

Volume I of this report contains the input instructions for this program. Volume III contains the source language listings. The program will run on CDC types of computers using the CDC FTN compiler. The program also contains all the code necessary for operation on IBM computers, except that in the listed decks the "IBM only" cards are made inactive with a C in column 1 and identified with an I in card column 80. A small converter program is furnished to convert the program form one machine to the other.

SECTION II

PROGRAM FRAMEWORK

The major features desired in this program were:

1. Provide the ability to analyze completely arbitrary three-dimensional shapes.
2. Provide a component build-up capability where each vehicle component may be of arbitrary shape.
3. Include a number of force analysis methods so that the program would have the widest possible application to various vehicle shapes and flight conditions.
4. Provide the capability to use the best force calculation method for each vehicle component but leave the actual method selection up to the user.
5. Provide engineering methods to account for the effect of the flow field generated by one component on the characteristics of another component.
6. Provide for convenient storage of data between program components.
7. Develop a total analysis system framework that is adaptable to continued improvement and expansion.
8. Keep the program as small and as fast as possible consistent with the above goals and requirements.
9. Prepare the program decks so that they will run either on CDC or IBM computers with a minimum of effort required to convert from one to the other.
10. Keep the program input data as simple as possible consistent with the requirements of program flexibility.

It is felt that the new Mark IV program meets each of these requirements (although some new users may take exception to Item 10 above).

The Mark IV program is a modularized computer program designed to handle a variety of high speed vehicle analysis problems. Mathematically, the methods used in each program component are not what one would call complex or sophisticated. However, when all of these capabilities are tied together in one place, the result is a very large program with, in many cases, some rather complex code. The functional organization of the program is shown in Figure 1.

The detailed description of the various theoretical methods used in the program are presented on the following pages.

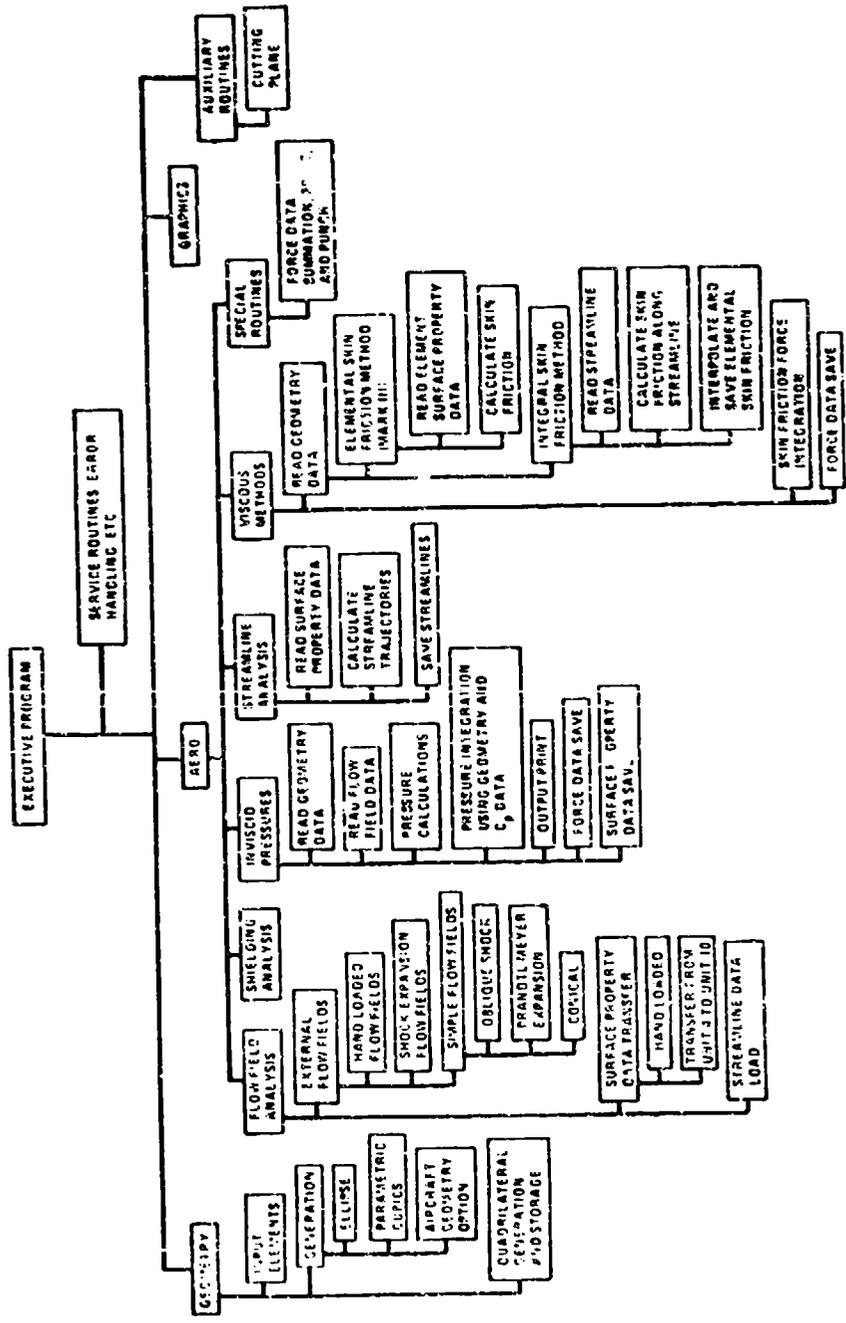


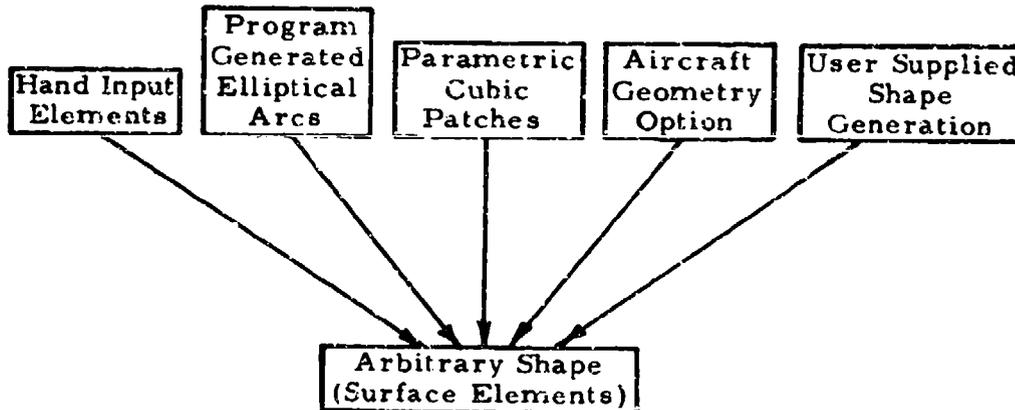
Figure 1. Functional Organization of Mark IV Program.

Reproduced from best available copy.

SECTION III

GEOMETRY

The new Mark IV Arbitrary-Body Program maintains all of the geometry capabilities of the old Mark III program. However, in the new program these capabilities are combined within a single program component that can, if required, operate as a stand-alone program. The basic capabilities of the new program include (1) input elements, (2) ellipse generation, (3) parametric cubic, and (4) the complete Aircraft Geometry Option of the Mark III Mod 3 program. These methods provide the flexibility required to analyze a variety of shapes ranging from very simple surfaces to the most complex forms. If desired, all of these methods could be used in describing a single vehicle shape. This general process is illustrated in the diagram below.



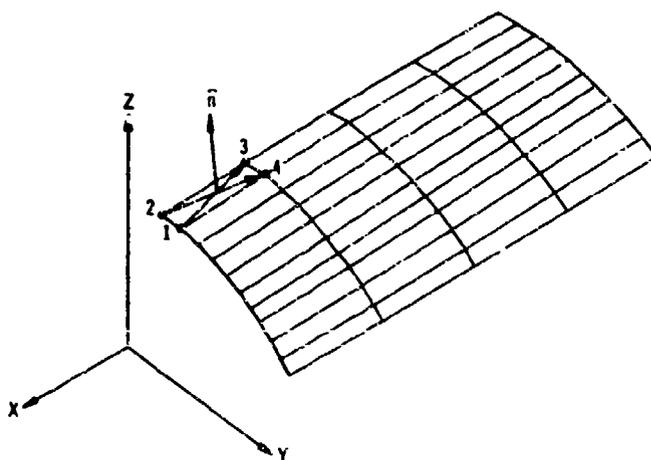
The use of a basically simple geometry representation concept has been a key feature in the development and success of the Mark III Hypersonic Arbitrary-Body Program. Many of the capabilities and options that were added to the program during its years of development would not have been possible (or very difficult to incorporate at best) if a more complicated basic geometry approach had been used originally.

The principles involved in the application of each of these geometry methods are discussed in detail in the User's Manual and need not be covered here. The principal mathematical techniques, however, are important from the programming standpoint and will be discussed on the following pages.

The Surface Element Geometry Method

The basic geometry method used by this program is the surface element or quadrilateral method. This method was developed by J. L. Hess and A. M. O. Smith for the Douglas Three-Dimensional Potential Flow Program (Reference 3). For completeness, certain parts of this report will be included in the following discussions.

The coordinate system used for this analysis is a right-handed Cartesian system as shown in the figure below.



DIAGONAL VECTORS \bar{T}_1 and \bar{T}_2

$$\begin{aligned} \bar{T}_{1x} &= X_3 - X_1 & \bar{T}_{1y} &= Y_3 - Y_1 & \bar{T}_{1z} &= Z_3 - Z_1 \\ \bar{T}_{2x} &= X_4 - X_2 & \bar{T}_{2y} &= Y_4 - Y_2 & \bar{T}_{2z} &= Z_4 - Z_2 \end{aligned}$$

UNIT NORMAL $\bar{N} = \bar{T}_2 \times \bar{T}_1$

$$\begin{aligned} N_x &= \bar{T}_{2y}\bar{T}_{1z} - \bar{T}_{1y}\bar{T}_{2z} & n_x &= N_x/N \\ N_y &= \bar{T}_{1x}\bar{T}_{2z} - \bar{T}_{2x}\bar{T}_{1z} & n_y &= N_y/N \\ N_z &= \bar{T}_{2x}\bar{T}_{1y} - \bar{T}_{1x}\bar{T}_{2y} & n_z &= N_z/N \end{aligned}$$

$$N = \sqrt{N_x^2 + N_y^2 + N_z^2}$$

AVERAGE POINT

$$\begin{aligned} \bar{x} &= \frac{1}{4}(X_1 + X_2 + X_3 + X_4) \\ \bar{y} &= \frac{1}{4}(Y_1 + Y_2 + Y_3 + Y_4) \\ \bar{z} &= \frac{1}{4}(Z_1 + Z_2 + Z_3 + Z_4) \end{aligned}$$

CORNER POINT PROJECTION DISTANCE

$$d_k = n_x(\bar{x} - X_k) + n_y(\bar{y} - Y_k) + n_z(\bar{z} - Z_k)$$

$k = 1, 2, 3, 4$

CORNER POINT COORDINATES

$$\begin{aligned} X'_k &= X_k + n_x d_k \\ Y'_k &= Y_k + n_y d_k \\ Z'_k &= Z_k + n_z d_k \end{aligned}$$

In the conventional use of this program the vehicle is usually positioned with its nose at the coordinate system origin and with the length of the body stretching in the negative X direction. The slight inconvenience of this negative sign on the body stations has been accepted so that the geometric data will be compatible with the Douglas potential flow program (Neumann Program).

The body surface is represented by a set of points in space. These points are selected on the body surface and are used by the method to obtain an approximation to this surface that is used in subsequent calculations. If the four related points of each set are connected by straight lines we may obtain a picture of how the input surface points are organized to describe a given shape. This has been done in Figure 2. The input scheme has been designed so that each point need only be input once even though it may be a member of as many as four adjacent sets of points. This is accomplished by the use of an additional parameter for each point besides the X, Y, and Z values. This parameter (known as the status flag) indicates whether a point is a continuation of a column of points (STATUS = 0), the beginning of a

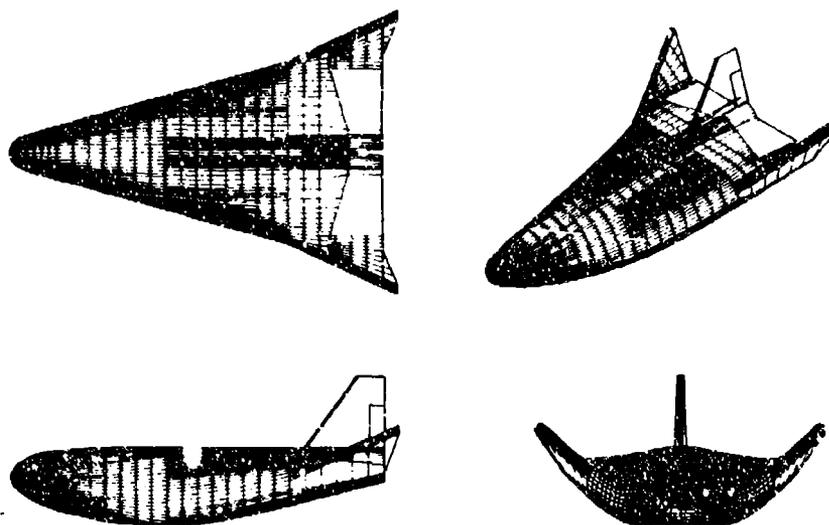


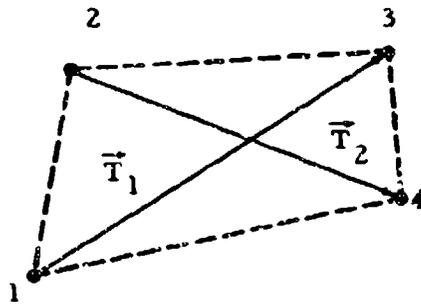
Figure 2. Output from Perspective Drawing Program

new column of points (=1), the first point of a new section of elements (=2), or the last point input for the shape (=3).

As may be seen from the drawings made by the Picture Drawing Program, the different areas of a vehicle may require a different organization and spacing of surface points for accurate representation. Each such area or organization of elements is called a section and each section is independent of all other sections. The division of a vehicle into a given set of sections may also be influenced by another consideration since the force calculation program may be made to calculate the force contributions of each section separately, using different calculation methods.

The input surface points are not sufficient in themselves for the force calculations. Each set of four related points which form an individual element must be converted into quantities useful to the program. This is accomplished by approximating each element area of the vehicle by a plane quadrilateral surface. Since we are using four surface points to form an element, no single surface will contain the points themselves. Also, adjacent plane quadrilateral edges will not necessarily be coincident. With a sufficiently small size of the surface elements this will be of no consequence in the end results.

The mathematical technique used in converting an input set of four points into a plane quadrilateral element is described below. The figure below gives a representation of the input element points with each point identified consecutively around the element by the subscripts 1, 2, 3, and 4, respectively.



The coordinates in the reference coordinate system are as follows:

1	:	x_1^i	y_1^i	z_1^i
2	:	x_2^i	y_2^i	z_2^i
3	:	x_3^i	y_3^i	z_3^i
4	:	x_4^i	y_4^i	z_4^i

The superscript i identifies the coordinates as input coordinates. We next form the two diagonal vectors \vec{T}_1 and \vec{T}_2 . The components of these vectors are

$$\begin{aligned}
 T_{1x} &= x_3^i - x_1^i & T_{1y} &= y_3^i - y_1^i & T_{1z} &= z_3^i - z_1^i \\
 T_{2x} &= x_4^i - x_2^i & T_{2y} &= y_4^i - y_2^i & T_{2z} &= z_4^i - z_2^i
 \end{aligned}$$

We may now obtain a new vector \vec{N} (and its components) by taking the cross product of the diagonal vectors.

$$\vec{N} = \vec{T}_2 \times \vec{T}_1$$

$$N_x = T_{2y} T_{1z} - T_{1y} T_{2z}$$

$$N_y = T_{1x} T_{2z} - T_{2x} T_{1z}$$

$$N_z = T_{2x} T_{1y} - T_{1x} T_{2y}$$

The unit normal vector, \vec{n} , to the plane of the element is taken as \vec{N} divided by its own length N (direction cosines of outward unit normal).

$$n_x = \frac{N_x}{N}$$

$$n_y = \frac{N_y}{N}$$

$$n_z = \frac{N_z}{N}$$

where

$$N = \sqrt{N_x^2 + N_y^2 + N_z^2}$$

The plane of the element is now completely determined if a point in this plane is specified. This point is taken as the point whose coordinates, \bar{x} , \bar{y} , \bar{z} are the averages of the coordinates of the four input points.

$$\bar{x} = \frac{1}{4} [x_1^i + x_2^i + x_3^i + x_4^i]$$

$$\bar{y} = \frac{1}{4} [y_1^i + y_2^i + y_3^i + y_4^i]$$

$$\bar{z} = \frac{1}{4} [z_1^i + z_2^i + z_3^i + z_4^i]$$

Now the input points will be projected into the plane of the element along the normal vector. The resulting points are the corner points of the quadrilateral element. The signed distance of the k -th input points ($k = 1, 2, 3, 4$) from the plane is

$$d_k = n_x(\bar{x} - x_k^i) + n_y(\bar{y} - y_k^i) + n_z(\bar{z} - z_k^i) \quad k = 1, 2, 3, 4$$

It turns out that, due to the way in which the plane was generated from the input points, all the d_k 's have the same magnitude, those for points 1 and 3 having one sign and those for points 2 and 4 having the opposite sign. Symbolically,

$$d_k = (-1)^{k-1} d_1 \quad k = 1, 2, 3, 4$$

The magnitude of the common projection distance is called d , i. e.,

$$d = |d_1|$$

The coordinates of the corner points in the reference coordinate system are given by

$$\begin{aligned}x_k^i &= x_k^j + n_x d_k \\y_k^i &= y_k^i + n_y d_k \\z_k^i &= z_k^i + n_z d_k\end{aligned} \quad k = 1, 2, 3, 4$$

Now the element coordinate system must be constructed. This requires the components of three mutually perpendicular unit vectors, one of which points along each of the coordinate axes of the system, and also the coordinates of the origin of the coordinate system. All these quantities must be given in terms of the reference coordinate system. The unit normal vector is taken as one of the unit vectors, so two perpendicular unit vectors in the plane of the element are needed. Denote these unit vectors \vec{t}_1 and \vec{t}_2 . The vector \vec{t}_1 is taken as \vec{T}_1 divided by its own length T_1 , i. e.,

$$t_{1x} = \frac{T_{1x}}{T_1}$$

$$t_{1y} = \frac{T_{1y}}{T_1}$$

$$t_{1z} = \frac{T_{1z}}{T_1}$$

where

$$T_1 = \sqrt{T_{1x}^2 + T_{1y}^2 + T_{1z}^2}$$

The vector \vec{t}_2 is defined by $\vec{t}_2 = \vec{n} \times \vec{t}_1$, so that its components are

$$t_{2x} = n_y t_{1z} - n_z t_{1y}$$

$$t_{2y} = n_z t_{1x} - n_x t_{1z}$$

$$t_{2z} = n_x t_{1y} - n_y t_{1x}$$

The vector \vec{t}_1 is the unit vector parallel to the x or ξ axis of the element coordinate system, while \vec{t}_2 is parallel to the y or η axis, and n is parallel to the z or ζ axis of this coordinate system.

To transform the coordinates of points and the components of vectors between the reference coordinate system and the element coordinate system, the transformation matrix is required. The elements of this matrix are the components of the three basic unit vectors, \vec{t}_1 , \vec{t}_2 , and \vec{n} . To make the notation uniform define

$$a_{11} = t_{1x} \quad a_{12} = t_{1y} \quad a_{13} = t_{1z}$$

$$a_{21} = t_{2x} \quad a_{22} = t_{2y} \quad a_{23} = t_{2z}$$

$$a_{31} = n_x \quad a_{32} = n_y \quad a_{33} = n_z$$

The transformation matrix is thus the array

$$\begin{array}{ccc} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{array}$$

To transform the coordinates of points from one system to the other, the coordinates of the origin of the element coordinate system in the reference coordinate system are required. Let these be denoted x_0 , y_0 , z_0 . Then if a point has coordinates x' , y' , z' in the reference coordinate system and coordinates x , y , z in the element coordinate

system, the transformation from the reference to the element system is

$$x = a_{11}(x' - x_0) + a_{12}(y' - y_0) + a_{13}(z' - z_0)$$

$$y = a_{21}(x' - x_0) + a_{22}(y' - y_0) + a_{23}(z' - z_0)$$

$$z = a_{31}(x' - x_0) + a_{32}(y' - y_0) + a_{33}(z' - z_0)$$

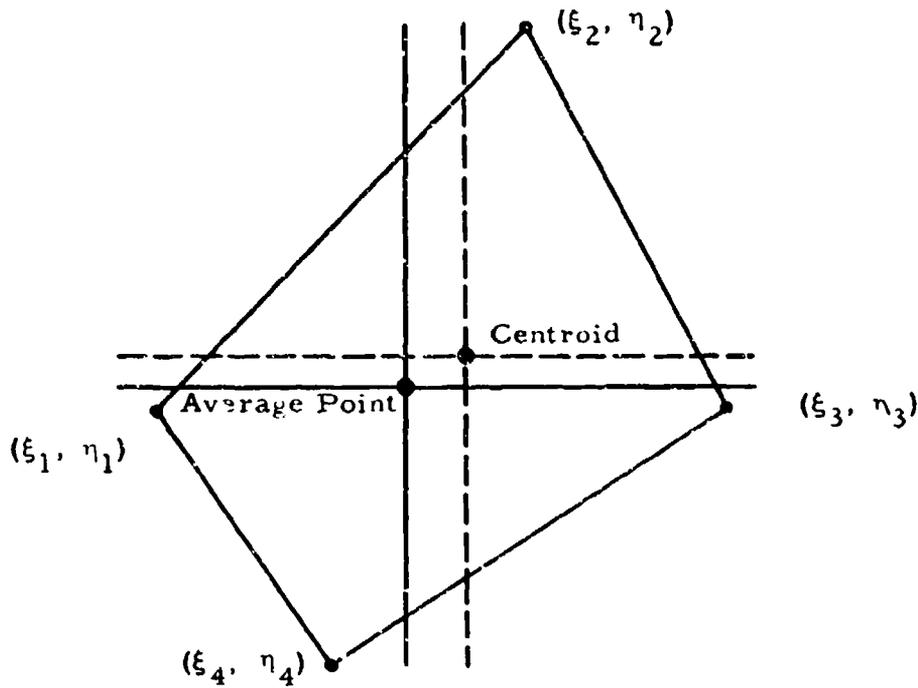
while the transformation from the element to the reference system is

$$x' = x_0 + a_{11}x + a_{21}y + a_{31}z$$

$$y' = y_0 + a_{12}x + a_{22}y + a_{32}z$$

$$z' = z_0 + a_{13}x + a_{23}y + a_{33}z$$

The corner points are now transformed into the element coordinate system based on the average point as origin. These points have coordinates x'_k, y'_k, z'_k in the reference coordinate system. Their coordinates in the element coordinate system with this origin are denoted by $\xi_k^*, \eta_k^*, 0$. Because they lie in the plane of the element, they have a zero z or ζ coordinate in the element coordinate system. Also, because the vector \vec{t}_1 , which defines the x or ξ axis of the element coordinate system, is a multiple of the "diagonal" vector from point 1 to point 3, the coordinate η_1^* and the coordinate η_3^* are equal. This is illustrated in the figure below. Using the above transformation these coordinates are explicitly



$$\xi_k^* = a_{11}(x_k' - \bar{x}) + a_{12}(y_k' - \bar{y}) + a_{13}(z_k' - \bar{z})$$

$$k = 1, 2, 3, 4$$

$$\eta_k^* = a_{21}(x_k' - \bar{x}) + a_{22}(y_k' - \bar{y}) + a_{23}(z_k' - \bar{z})$$

These corner points are taken as the corners of a plane quadrilateral.

The origin of the element coordinate system is now transferred to the centroid of the area of the quadrilateral. With the average point as origin the coordinates of the centroid in the element coordinate system are:

$$\xi_0 = \frac{1}{3} \frac{1}{\eta_2^* - \eta_4^*} \left[\xi_4^* (\eta_1^* - \eta_2^*) + \xi_2^* (\eta_4^* - \eta_1^*) \right]$$

$$\eta_0 = -\frac{1}{3} \eta_1^*$$

These are subtracted from the coordinates of the corner points in the element coordinate system based on the average point as origin to obtain the coordinates of the corner points in the element coordinate system based on the centroid as origin. Accordingly, these latter coordinates are

$$\xi_k = \xi_k^* - \xi_0$$

$$k = 1, 2, 3, 4$$

$$\eta_k = \eta_k^* - \eta_0$$

Since the centroid is to be used as the origin of the element coordinate system, its coordinates in the reference coordinate system are required for use with the transformation matrix. These coordinates are

$$x_0 = \bar{x} + a_{11} \xi_0 + a_{21} \eta_0$$

$$y_0 = \bar{y} + a_{12} \xi_0 + a_{22} \eta_0$$

$$z_0 = \bar{z} + a_{13} \xi_0 + a_{23} \eta_0$$

Since in all subsequent transformations between the reference coordinate system and the element coordinate system the centroid is used as origin of the latter, its coordinates are denoted x_0 , y_0 , z_0 . The coordinates of the average point are no longer needed. The change in origin of the element coordinate system, of course, has no effect on the coordinates of the corner points in the reference coordinate system.

The lengths of the two diagonals of the quadrilateral, t_1 and t_2 , are computed from

$$t_1^2 = (\xi_3 - \xi_1)^2$$

$$t_2^2 = (\xi_4 - \xi_2)^2 + (\eta_4 - \eta_2)^2$$

The larger of these is selected and designated the maximum diagonal t .

The body surface area and enclosed volume are determined by summing up the contributions of each element. In terms of the coordinates of the corner points, the area of the quadrilateral is

$$A = \frac{1}{2} (\xi_3 - \xi_1) (\eta_2 - \eta_4)$$

The incremental volume is given by the volume of the parallelepiped formed by the element and its projection onto the $x-z$ plane (the $x-y$ or $y-z$ planes would have served equally well).

$$V = y_0 A \eta_y$$

Summary

The foregoing procedure may be briefly summarized as follows:

Each set of four points is converted into a plane-quadrilateral element by the procedure shown in the sketch on page 7. The normal to the quadrilateral is taken as the cross product of two diagonal vectors formed between opposite element points. The order of the input points and the manner of defining the diagonal vectors is used to ensure that the cross product gives an outward normal to the body surface. The next step is to define the plane of the element by determining the averages of the coordinates of the original four corner points. These points are then projected parallel to the normal vector into the plane of the element to give the corners of the plane quadrilateral. The corner points of the quadrilateral are equidistant from the four points used to form the element. Additional parameters required for subsequent force calculations, quadrilateral area and centroid, may now be calculated.

The spacing and orientation of the elements is varied in such a way that they describe the vehicle shape accurately. Since four points are used to define the plane quadrilateral, the edges of adjacent elements are not coincident. This is not important, since the pressure is calculated only at the quadrilateral centroid. This pressure is then assumed to be constant over the surface of the element.

The plane-quadrilateral surface description method is not as elaborate as some of the other methods. It is important, however, to note that the simplicity of the method permits the use of conventional cross-sectional drawings in data preparation (no surface slopes required) and the use of semiautomatic data-reading techniques. Also, as has been illustrated in Volume I, computer-generated pictures are used in checking the geometric data for errors.

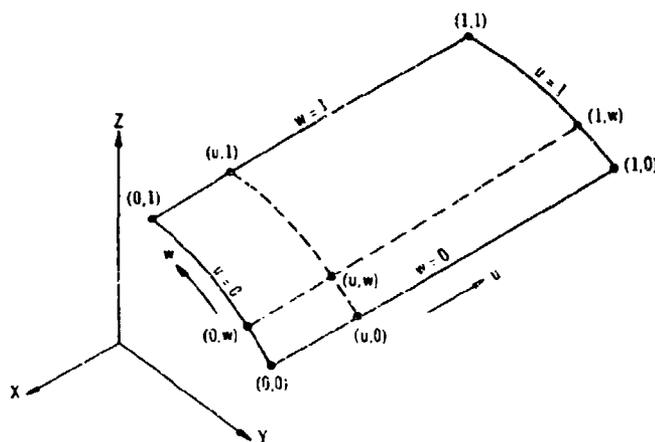
Parametric Cubic

A second technique for describing three-dimensional curved surfaces is also provided within the program. This is a mathematical surface-fit technique and is identified as the Parametric Cubic Method because of the general type of equations used.

Several different mathematical surface-fit techniques are described in the literature. The one used in this program was adopted from the formulation given by Coons of MIT (Reference 4). In this method a vehicle shape is also divided into a number of sections or patches. The size and location of each patch depends upon the shape of the surface.

The basic feature of this method is that only the surface conditions at the patch corner points are required to completely describe the surface enclosed by the boundary curves of the patch. The basic problem, however, is the determination of all the information required at these corner points, i.e., the surface equation requires corner point surface derivatives with respect to the parametric variables rather than the X, Y, Z coordinates. This has been solved by the use of additional points along the boundary curves as will be described later.

In the following discussions we will use the geometrical representation of a surface patch as illustrated in the figure below.



BOUNDARY CURVE (FOR $u = 0$)

$$X_i(0, w) = Aw^3 + Bw^2 + Cw + D$$

$$A = 2[X_i(0,0) - X_i(0,1)] + \frac{\partial X_i}{\partial w}(0,0) + \frac{\partial X_i}{\partial w}(0,1)$$

$$B = 3[X_i(0,1) - X_i(0,0)] - 2\frac{\partial X_i}{\partial w}(0,0) - \frac{\partial X_i}{\partial w}(0,1)$$

$$C = \frac{\partial X_i}{\partial w}(0,0) \quad D = X_i(0,0)$$

$$\frac{\partial X_i}{\partial w} = \frac{\partial X_i}{\partial S} \frac{\partial S}{\partial w} \quad i = 1, 2, 3 \text{ FOR } X, Y, Z$$

BLENDING FUNCTIONS

$$F_1(u) = 3u^2 - 2u^3 \quad F_3(w) = 3w^2 - 2w^3$$

$$F_2(u) = 1 - F_1(u) \quad F_4(w) = 1 - F_3(w)$$

SURFACE FORM

$$X_i(u, w) = X_i(0, w)F_2(u) + X_i(1, w)F_1(u) + X_i(u, 0)F_4(w) + X_i(u, 1)F_3(w) - X_i(0, 0)F_2(u)F_4(w) - X_i(0, 1)F_2(u)F_3(w) - X_i(1, 0)F_1(u)F_4(w) - X_i(1, 1)F_1(u)F_3(w)$$

Since the basic surface-fit equations and their derivatives are presented in Reference 4, they need be only reviewed briefly in this report.

The X, Y, Z coordinates of a point on the surface are related to the two parametric variables u and w. Thus, a surface in space is mapped into the u, w unit square. The basic problem is to find the position (X, Y, Z) of a point (u, w) in the interior of the section surface. The general procedure is to first find relationships for the four boundary curves. These are defined as third-order polynomials in terms of the parametric variables. The points on the boundary curves corresponding to u and w (0, w and u, 0, etc.) are then calculated. A general surface equation is used to calculate the properties at the point u, w. This equation uses blending or weighting functions to properly introduce the influence of each of the related boundary-curve points and the four corner points. The blending functions also ensure the continuity of the slopes across the boundaries between adjacent sections.

There are several methods for calculating the direction cosines of the tangent vectors required in the calculation of the corner-point derivatives. Most require the specification of additional surface-boundary points, some of which may lie on the extensions of the boundary curves. The derivatives must be calculated, since it would not be practical to measure them directly from drawings. The method in this program involves the use of circular arcs through three boundary-curve points, the middle one being a corner point.

The first step in the computational procedure is to determine the equations for the cubic boundary curves. The equation used is given by the following relationship for $u = 0$.

$$X_i(0, w) = Aw^3 + Bw^2 + Cw + D$$

where

$$A = 2 \left[X_i(0, 0) - X_i(0, 1) \right] + \frac{\partial X_i}{\partial w}(0, 0) + \frac{\partial X_i}{\partial w}(0, 1)$$

$$B = 3 \left[X_i(0, 1) - X_i(0, 0) \right] - 2 \frac{\partial X_i}{\partial w}(0, 0) - \frac{\partial X_i}{\partial w}(0, 1)$$

$$C = \frac{\partial X_i}{\partial w}(0, 0)$$

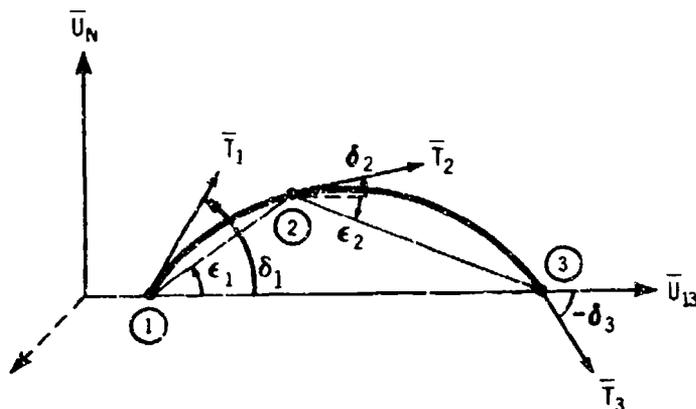
$$D = X_i(0, 0)$$

Similar equations are needed for the other three boundary curves with $v = 1$, $w = 0$, and $w = 1$.

The missing items required for the solution of the above equations are the derivatives

$$\frac{\partial X_i}{\partial w}(0, 0), \frac{\partial X_i}{\partial w}(0, 1), \text{ etc}$$

In the Arbitrary-Body Program these are determined by passing a circular arc through three points, the middle point being the corner point itself. For completeness, the development of this method is presented and the sketch below is useful in following the derivation.



This sketch is a view of the plane of the circle with \bar{U}_{13} as the base coordinate. The vectors \bar{T}_1 , \bar{T}_2 , and \bar{T}_3 are tangents to the curve at the points 1, 2, and 3.

The tangents make the angles δ_1 , δ_2 , and δ_3 with respect to \bar{U}_{13} . The chord lengths make the angles ϵ_1 and ϵ_2 with respect to the vector \bar{U}_{13} .

One of the properties of circular arcs is that the chord angle is the average of the two tangent angles.

$$\epsilon_1 = \frac{\delta_1 + \delta_2}{2} \quad \epsilon_2 = \frac{\delta_2 + \delta_3}{2} \quad \epsilon_3 = \frac{\delta_1 + \delta_3}{2}$$

For the coordinate base selected (\bar{U}_{13}), $\epsilon_3 = 0$, therefore,

$$\delta_1 = -\delta_3 \quad \text{and} \quad \delta_2 = \epsilon_1 + \epsilon_2$$

The tangent vector at point 2 is then given by

$$\bar{T}_2 = \cos \delta_2 \bar{U}_{13} + \sin \delta_2 \bar{U}_N$$

$$\bar{U}_{13} = \frac{\bar{L}_{13}}{|\bar{L}_{13}|}, \quad \bar{L}_{13} \text{ is chord vector between points 1 and 3.}$$

To determine \bar{U}_N , the binormal \bar{U}_{BN} must first be found

$$\bar{U}_{BN} = \bar{L}_{13} \times \bar{L}_{12}$$

$$\bar{U}_{BN} = \frac{\bar{U}_{BN}}{|\bar{U}_{BN}|} \quad (\text{unit vector})$$

$$\bar{U}_N = \bar{U}_{BN} \times \bar{U}_{13}$$

The radius vectors (X, Y, Z) for the three points are

$$\bar{r}_1 = X_1 \bar{i} + Y_1 \bar{j} + Z_1 \bar{k}$$

$$\bar{r}_2 = X_2 \bar{i} + Y_2 \bar{j} + Z_2 \bar{k}$$

$$\bar{r}_3 = X_3 \bar{i} + Y_3 \bar{j} + Z_3 \bar{k}$$

The chord vectors between the points are

$$\bar{L}_{12} = \bar{r}_2 - \bar{r}_1 = (X_2 - X_1)\bar{i} + (Y_2 - Y_1)\bar{j} + (Z_2 - Z_1)\bar{k}$$

$$\bar{L}_{23} = \bar{r}_3 - \bar{r}_2 = (X_3 - X_2)\bar{i} + (Y_3 - Y_2)\bar{j} + (Z_3 - Z_2)\bar{k}$$

$$\bar{L}_{13} = \bar{r}_3 - \bar{r}_1 = (X_3 - X_1)\bar{i} + (Y_3 - Y_1)\bar{j} + (Z_3 - Z_1)\bar{k}$$

and the chord angles

$$\cos \epsilon_1 = \frac{\bar{L}_{12} \cdot \bar{L}_{13}}{|\bar{L}_{12}| |\bar{L}_{13}|} \quad \cos \epsilon_2 = \frac{\bar{L}_{23} \cdot \bar{L}_{13}}{|\bar{L}_{23}| |\bar{L}_{13}|}$$

For convenience we will use the shortened notation:

$$L_{12} = |\bar{L}_{12}|, \text{ etc.}$$

$$\begin{aligned} \bar{U}_{13} &= \left(\frac{X_3 - X_1}{L_{13}} \right) \bar{i} + \left(\frac{Y_3 - Y_1}{L_{13}} \right) \bar{j} + \left(\frac{Z_3 - Z_1}{L_{13}} \right) \bar{k} \\ &= \ell_1 \bar{i} + m_1 \bar{j} + n_1 \bar{k} \end{aligned}$$

Similarly

$$\bar{U}_{12} = \ell_2 \bar{i} + m_2 \bar{j} + n_2 \bar{k}$$

$$\begin{aligned} \bar{U}_{BN} &= \frac{\bar{U}_{BN}}{|\bar{U}_{BN}|} = \bar{U}_{13} \times \bar{U}_{12} = \begin{vmatrix} \bar{i} & \bar{j} & \bar{k} \\ \ell_1 & m_1 & n_1 \\ \ell_2 & m_2 & n_2 \end{vmatrix} \\ &= (m_1 n_2 - m_2 n_1) \bar{i} - (\ell_1 n_2 - \ell_2 n_1) \bar{j} + (\ell_1 m_2 - \ell_2 m_1) \bar{k} \end{aligned}$$

$$\bar{U}_N = \bar{U}_{BN} \times \bar{U}_{13} = \begin{vmatrix} \bar{i} & \bar{j} & \bar{k} \\ () & -() & () \\ \ell_1 & m_1 & n_1 \end{vmatrix}$$

$$\begin{aligned}
&= \left[-n_1 (\ell_1 n_2 - \ell_2 n_1) - m_1 (\ell_1 m_2 - \ell_2 m_1) \right] i \\
&- \left[n_1 (m_1 n_2 - m_2 n_1) - \ell_1 (\ell_1 m_2 - \ell_2 m_1) \right] j \\
&+ \left[m_1 (m_1 n_2 - m_2 n_1) + \ell_1 (\ell_1 n_2 - \ell_2 n_1) \right] k
\end{aligned}$$

$$\bar{U}_N = \ell_N \bar{i} + m_N \bar{j} + n_N \bar{k}$$

And finally we obtain the tangent vector

$$\begin{aligned}
\bar{T}_2 &= (\ell_1 \cos \delta_2 + \ell_N \sin \delta_2) \bar{i} + (m_1 \cos \delta_2 + m_N \sin \delta_2) \bar{j} \\
&+ (n_1 \cos \delta_2 + n_N \sin \delta_2) \bar{k}
\end{aligned}$$

where

$$\ell_1 = \frac{X_3 - X_1}{L_{13}}, \quad m_1 = \frac{Y_3 - Y_1}{L_{13}}, \quad n_1 = \frac{Z_3 - Z_1}{L_{13}}$$

$$L_{13} = \left[(X_3 - X_1)^2 + (Y_3 - Y_1)^2 + (Z_3 - Z_1)^2 \right]^{1/2}$$

$$\ell_N = - \left[n_1 (\ell_1 n_2 - \ell_2 n_1) + m_1 (\ell_1 m_2 - \ell_2 m_1) \right]$$

$$m_N = - \left[n_1 (m_1 n_2 - m_2 n_1) + \ell_1 (\ell_1 m_2 - \ell_2 m_1) \right]$$

$$n_N = \left[m_1 (m_1 n_2 - m_2 n_1) + \ell_1 (\ell_1 n_2 - \ell_2 n_1) \right]$$

and

$$\ell_2 = \frac{X_2 - X_1}{L_{12}}, \quad m_2 = \frac{Y_2 - Y_1}{L_{12}}, \quad n_2 = \frac{Z_2 - Z_1}{L_{12}}$$

$$L_{12} = \left[(X_2 - X_1)^2 + (Y_2 - Y_1)^2 + (Z_2 - Z_1)^2 \right]^{1/2}$$

The final end point derivatives are then found from

$$\frac{\partial X_i}{\partial w} = \frac{\partial X_i}{\partial S} \cdot \frac{\partial S}{\partial w} = T_i \Delta S; \quad i = 1, 2, 3 \text{ for } X, Y, Z.$$

where

$\frac{\partial s}{\partial w}$ = the boundary length since $\Delta w = 1$ on the unit square patch

$$\Delta S = \sum_{I=2}^{I=NB-1} \left[(X_{I+1} - X_I)^2 + (Y_{I+1} - Y_I)^2 + (Z_{I+1} - Z_I)^2 \right]^{1/2}$$

$I = 2$ at the starting corner point

$I = NB - 1$ at the final point on the boundary curve

NB = number of points input on the boundary with one point extending off each end of the boundary curve.

Once the boundary curves are found the values required for the general surface equation can be calculated. This equation is given below.

$$\begin{aligned} X_i(u, w) = & X_i(0, w)F_0(u) + X_i(1, w)F_1(u) + X_i(u, 0)F_0(w) \\ & + X_i(u, 1)F_1(w) - X_i(0, 0)F_0(u)F_0(w) \\ & - X_i(0, 1)F_0(u)F_1(w) - X_i(1, 0)F_1(u)F_0(w) \\ & - X_i(1, 1)F_1(u)F_1(w) \end{aligned}$$

where the terms F_0 and F_1 are blending functions given by

$$\begin{aligned} F_1(u) &= 3u^2 - 2u^3 & F_1(w) &= 3w^2 - 2w^3 \\ F_0(u) &= 1 - F_1(u) & F_0(w) &= 1 - F_1(w) \end{aligned}$$

The program does not use the parametric cubic geometry data directly in the pressure calculations. Instead, the parametric cubic data are used in creating surface elements by a systematic variation of the parametric variables w , and u .

One advantage of the mathematical surface-fit technique over the plane-distributed-element method is the smaller number of surface points required to describe a shape. However, additional points are required on the boundaries to determine the required corner derivatives. This method is not as adaptable to semiautomatic data-reading techniques, since the organization of the required input data is more complex. The accuracy of this method depends upon the distribution and orientation of the surface sections, just as the plane-distributed-element method depends upon the distribution of the elements.

Aircraft Geometry Option

The primary purpose of the Aircraft Geometry Option is to provide a convenient means for generating detailed element geometry data for conventional airplane types of configurations that are made up of a fuselage, wings, horizontal tails, canards, fins, and nacelles or pods. The input to the Aircraft Geometry Option is in the form of fuselage coordinate data, airfoil ordinates and general planform shape, and element increment control data. The output consists of the standard surface element data (TYPE 3 data cards) in the format required by the components of the program. The configurations that may be generated with this option are very general in nature and include such capabilities as an arbitrarily shaped fuselage with camber, cambered wings defined by a number of airfoils, nacelles and external stores with circular cross sections, and vertical fins. The capabilities provided by the Aircraft Geometry Option may also be used in conjunction with all the other geometry generation and input features of the program to form a single vehicle shape. For example, it is possible to generate the wing and tail of a configuration using the Aircraft Geometry Option, to input a portion of the fuselage using input elements, and to complete the configuration using ellipse and parametric cubic generated data.

As a special note it should be pointed out that the Aircraft Geometry Option was originally prepared as a tool in checking out the geometry data for the NASA Harris Wave Drag Program. This capability has been maintained as a sub-set within the Aircraft Geometry Option in its present form. However, an additional aircraft surface type has been added that permits the use of arbitrarily oriented airfoils in describing wing and tail types of surfaces. Also, the Aircraft Geometry Option permits the use of arbitrarily orientated pods or nacelles.

The input requirements and capabilities of the Aircraft Geometry Option are discussed in sufficient detail in Volume I. However, there are two parts of the Aircraft Geometry generation process, that of pods or nacelles and the general airfoil surfaces, that do need a bit more on the mathematical development. This information is given in the following discussion.

Pods or Nacelles

A pod or nacelle is a body of revolution with its axis arbitrarily located with reference to the vehicle axis system. This increased capability has been added without affecting the NASA Wave Drag Program input format (the NASA program is limited to having the pod axis parallel to the vehicle X-axis). The pod is defined with respect to its own coordinate system ($X'-Y'-Z'$), the orientation of which is considered to have

been achieved through a yaw-pitch sequence of rotations. The parameters used in defining the pod and the formation of surface elements are illustrated in Figure 3.

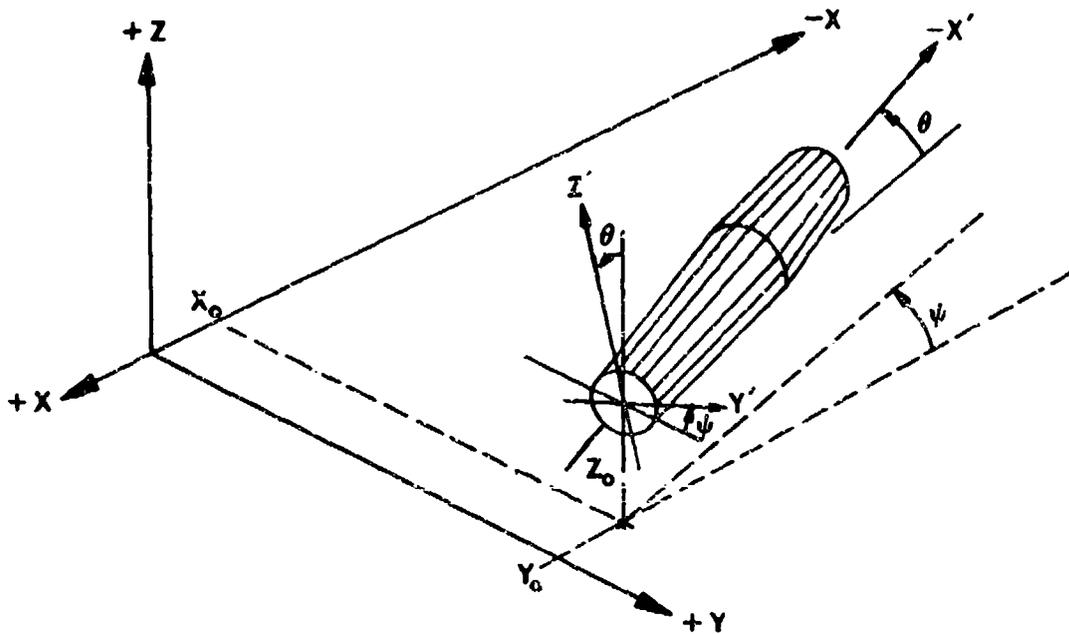


Figure 3. Pod or Nacelle Geometry.

The yaw angle ψ and the pitch angle θ are derived by the program from the coordinates of the pod origin and end point;

$$\sin \theta = (Z_E - Z_O)/L$$

$$\sin \psi = (Y_O - Y_E)/L * \cos \theta$$

where L is the length of the pod,

$$L = [(X_O - X_E)^2 + (Y_O - Y_E)^2 + (Z_O - Z_E)^2]^{1/2}$$

The surface coordinates in the vehicle axis system are given by

$$\begin{bmatrix} X - jX_O \\ Y - Y_O \\ Z - Z_O \end{bmatrix} = (E^{-1})_{\varphi=0} \begin{bmatrix} -X' \\ Y' \\ Z' \end{bmatrix}$$

where $(E^{-1})_{\varphi=0}$ is the rotation matrix E^{-1} derived in Section XII with φ set equal to zero, and

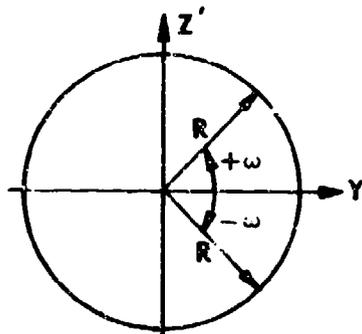
$$\begin{aligned} j &= +1 \text{ for arbitrary-body program input coordinates} \\ &= -1 \text{ for NASA Wave Drag Program input coordinates} \end{aligned}$$

Carrying out the multiplication the surface coordinates become

$$\begin{aligned} X &= jX_O - X' \cos \theta \cos \psi - Y' \sin \psi + Z' \sin \theta \cos \psi \\ Y &= Y_O - X' \cos \theta \sin \psi + Y' \cos \psi + Z' \sin \theta \sin \psi \\ Z &= Z_O + X' \sin \theta \qquad \qquad \qquad + Z' \cos \theta \end{aligned}$$

In the pod coordinate system, a radius distribution, R , is specified as a function of X' . Therefore,

$$\begin{aligned} Y' &= R \cos \omega \\ Z' &= R \sin \omega \end{aligned}$$



The meridian angle ω is taken to have zero value along the Y' -axis to automatically account for the sign of Z' .

The final expressions for the surface points in the vehicle axis system are thus given by

$$\begin{aligned} X &= jX_O - X' \cos \theta \cos \psi + R(\sin \omega \sin \theta \cos \psi - \cos \omega \sin \psi) \\ Y &= Y_O - X' \cos \theta \sin \psi + R(\sin \omega \sin \theta \sin \psi + \cos \omega \cos \psi) \\ Z &= Z_O + X' \sin \theta \qquad \qquad \qquad + R \sin \omega \cos \theta \end{aligned}$$

The input information required to define a pod or nacelle is as follows.

1. Number of pods (up to 9).
2. Number of stations to be used in the pod radii distribution input (2 to 30). This is the same for all pods.
3. The X-Y-Z coordinates of the origin and end of each pod in the vehicle coordinate system.
4. A table of X-ordinates (relative to pod origin) for the pod radii distribution.
5. Pod radii distribution for each pod.

The order of the generated surface points is from the bottom around to the top. The first point of each pod has a Status of 2, each new station starts with a Status of 1, and all other points have Status = 0. If the last point for a station fills only the left half of the Type 3 Element Data Card, a dummy point is generated to fill the right half of the card. When the pod axis lies in the X-Z origin plane, only half the pod is generated ($-90^\circ \leq \omega \leq +90^\circ$). Otherwise elements for the complete pod are determined.

In addition to specifying the axis orientation, the number of elements in 180° may also be specified. If this expanded capability is not used and the input fields are left blank, the program assumes the pod axis is parallel to the vehicle axis, and elements are generated every 15° in ω .

General Airfoil Surfaces

This geometry surface type may be used to generate surfaces that are defined by airfoil sections having arbitrary orientations in space. The airfoils are not confined to fixed planes. This more general approach permits the use of non-streamwise airfoil sections and is useful in describing intersecting components such as the wing and tail fuselage junctures. Input cards for this surface type cannot be used in input to the NASA Wave Drag Program.

The general airfoil surface is defined by connecting two or more airfoil sections with straight lines. The orientation of each airfoil is given by coordinates of the leading and trailing edges and an airfoil rotation angle. The techniques used in defining these airfoils and in performing the necessary transformation to obtain the required Z-Y-Z coordinates in the vehicle coordinate system are discussed below.

Each airfoil section is defined relative to a coordinate system fixed within the airfoil. The airfoil thickness displacements may be measured either from the mean-camber line along a line perpendicular to the airfoil axis or on a line that is normal to the mean camber line. This latter method is used in some of the early NASA airfoil documents. All airfoil section parameters are expressed as a percent of the airfoil chord. The parameters used in defining an airfoil are illustrated in Figure 4. In this illustration the airfoil lies in the η - ξ plane.

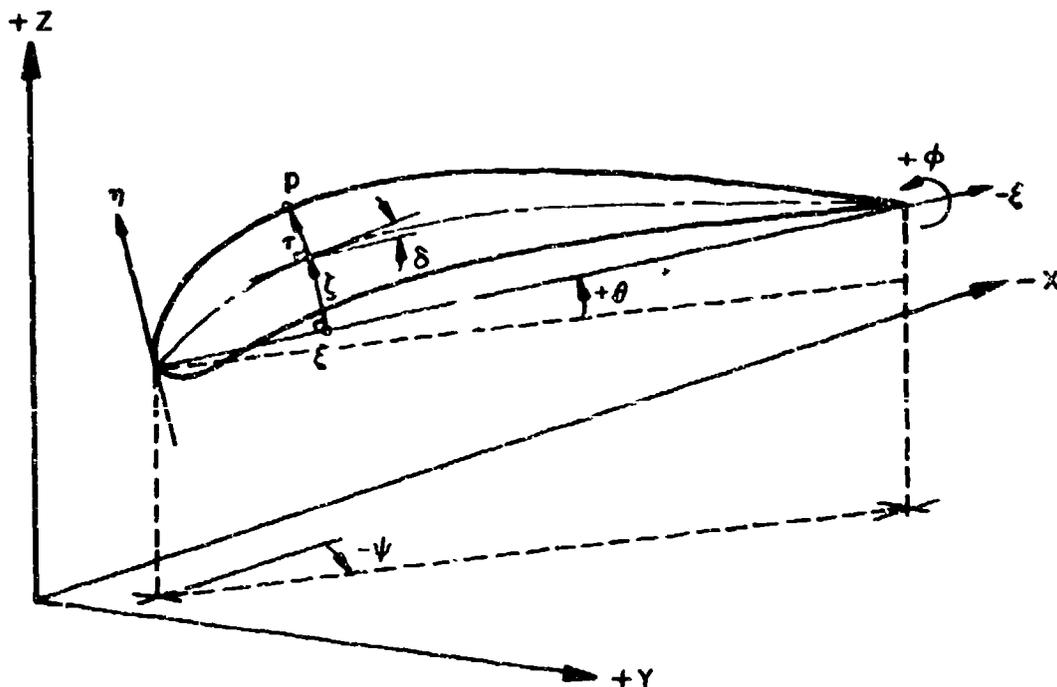


Figure 4. General Airfoil Coordinate System.

The coordinates of a point on the surface of the airfoil are given by the following relationships.

$$\eta_p = \zeta + DZ * \tau * \cos \delta$$

$$\xi_p = \xi - DZ * \tau * \sin \delta$$

Where

$$\zeta = \frac{Z_c(\xi)}{C}, \text{ the mean camber line distribution}$$

$$\tau = \frac{t(\xi)}{C}, \text{ the thickness distribution}$$

$$\tan \delta = \frac{d\zeta}{d\xi}, \text{ the slope of the mean camber line}$$

$$\begin{aligned} DZ &= +1.0 \text{ for the upper surface (thickness} \\ &\text{measured in the } +\eta \text{ direction)} \\ &= -1.0 \text{ for the lower surface (thickness} \\ &\text{measured in the } -\eta \text{ direction)} \end{aligned}$$

In the above general equations the point η_p, ξ_p on the airfoil is derived by using a thickness distribution measured along a line normal to the mean camber line. If the surface point is to be on a line normal to the airfoil chord line, the parameter δ is set equal to zero. Both options are available in the program. The upper surface of the airfoil is generated first and followed by the lower surface.

The airfoil coordinates (ξ, η) are next transformed to the vehicle axis system. The ξ - η plane orientation is considered to have been achieved through a yaw-pitch-roll sequence of rotations. The yaw angle ψ and pitch angle θ (and also the chord length C) are derived by the program from the input coordinates of the airfoil leading and trailing edges.

$$C = \left[(X_{LE} - X_{TE})^2 + (Y_{LE} - Y_{TE})^2 + (Z_{LE} - Z_{TE})^2 \right]^{1/2}$$

$$\sin \theta = (Z_{TE} - Z_{LE})/C$$

$$\sin \psi = (Y_{LE} - Y_{TE})/C * \cos \theta$$

The roll angle ϕ is input explicitly and together with ψ and θ are positive in the right-handed sense of the reference system.

Zero values for the rotation angles indicate the airfoil is orientated parallel to the X-Z plane. Zero yaw and pitch angles and a $+90^\circ$ degree roll angle gives an airfoil in the X-Y plane (such as a vertical tail root airfoil).

The surface coordinates in the vehicle-axis system are given by

$$\begin{bmatrix} X - jX_{LE} \\ Y - Y_{LE} \\ Z - Z_{LE} \end{bmatrix} = E^{-1} \begin{bmatrix} -\xi \\ 0 \\ \eta \end{bmatrix} C/100$$

The rotation matrix E^{-1} is derived in Section XII. Therefore, the desired airfoil surface coordinates are

$$X = jX_{LE} - [\xi \cos \theta \cos \psi + \eta (\sin \theta \cos \psi \cos \varphi + \sin \psi \sin \varphi)] * C/100$$

$$Y = Y_{LE} - [\xi \cos \theta \sin \psi + \eta (\sin \theta \sin \psi \cos \varphi - \cos \psi \sin \varphi)] * C/100$$

$$Z = Z_{LE} + [\xi \sin \theta + \eta \cos \theta \cos \varphi] * C/100$$

where

- j = +1 for Arbitrary-Body Program input coordinates (-X)
- = -1 for NASA Wave Drag Program input coordinates (+X)

The input information required by the Aircraft Geometry Option to define a general airfoil surface is as follows.

1. Number of airfoils.
2. Number of airfoil percent-chord points used to define the airfoils.
3. Flags to control the thickness distribution type, generation of tip and root closure elements, and repetitive use of mean camber line and thickness distributions.
4. A table of percent chord locations that are to be used for the airfoil thickness and camber distributions.
5. The X-Y-Z coordinates of the leading and trailing edge of each airfoil section.
6. The roll angle ϕ of each airfoil section.
7. The mean camber line ordinates in percent-chord at each percent chord location for each airfoil.
8. Thickness distribution in percent chord at each percent-chord position for each airfoil.

This surface type differs from those previously described in that repetitive use may be made of the arbitrary airfoil option on a single pass into the Aircraft Geometry Option. This stacking option allows wings, fins, etc., to be generated on a single pass into the Aircraft Geometry Option. A control flag also permits repetitive use of airfoil data for subsequent airfoils to save input time when all the surface airfoils are identical. Tip and root closure elements may also be generated to give a completely enclosed surface.

Control Surface Geometry *

The geometry data for a control surface flap are input to the program in the undeflected position. The methods used in transforming these data to the required deflected position are outlined in the following discussion.

The coordinate system used in these derivations is shown in Figure 5.

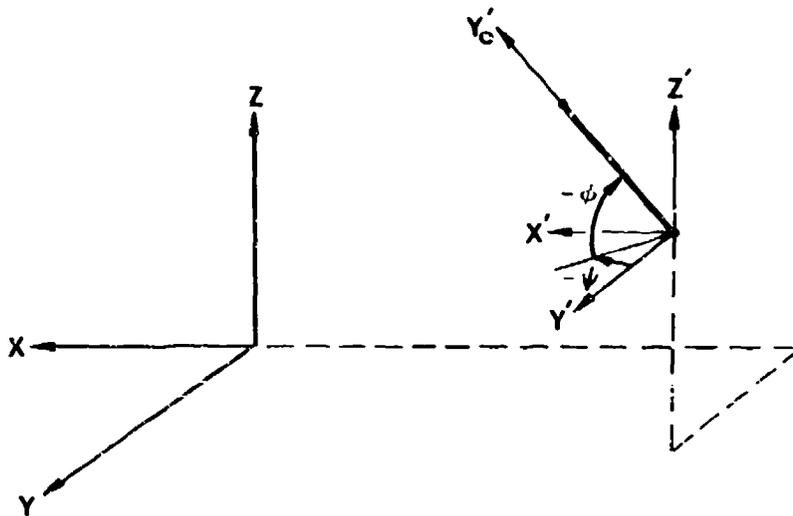


Figure 5. Control Surface Angle Definitions.

The general procedure involves a coordinate shift and an appropriate rotation to a hinge-line centered coordinate system such that the new Y-axis (Y_0) lies along the hinge line. For ψ and ϕ equal to zero and with the flap surface normal in the negative z-direction, the hinge-line centered coordinate system has the same directions as the body-axis system. The corner points, centroid, and normal vector (direction cosines) for each element of the flap are transformed into this system. Since the flap is a rigid body this information is independent of flap deflection and the hinge moment factor (moment per unit normal force) need only be determined once. However, the force magnitude is a function of the deflection angle and requires having the geometry of the deflected flap in the vehicle-centered coordinates.

* Note: Control surface deflection is not in the Mark IV Mod 0 release but will be added at a later date.

The coordinate system shift is given by

$$X' = X - X_{HL_4}$$

$$Y' = Y - Y_{HL_4}$$

$$Z' = Z - Z_{HL_4}$$

where

$()_{HL_4}$ is to point 4 on the hinge line

The new coordinates of the flap in the shifted and transformed coordinate system are given by

$$\begin{bmatrix} X'_0 \\ Y'_0 \\ Z'_0 \end{bmatrix} = [E] \begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix}$$

where

$$[E] = [\Phi] [\Psi]$$

$$[\Psi] = \begin{bmatrix} \cos \psi & \sin \psi & 0 \\ -\sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

ψ = rotation about the Z' -axis

$$[\Phi] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & \sin \phi \\ 0 & -\sin \phi & \cos \phi \end{bmatrix}$$

ϕ = rotation about the X'_0 -axis

The final rotation to the deflected position (δ_e is the control surface deflection) is given by

$$\begin{bmatrix} X'_{0\delta_e} \\ Y'_{0\delta_e} \\ Z'_{0\delta_e} \end{bmatrix} = \begin{bmatrix} \delta_e \end{bmatrix} \begin{bmatrix} X'_0 \\ Y'_0 \\ Z'_0 \end{bmatrix} = \begin{bmatrix} \delta_e \end{bmatrix} \begin{bmatrix} E \end{bmatrix} \begin{bmatrix} X' \\ Y' \\ Z \end{bmatrix}$$

where

$$\begin{bmatrix} \delta_e \end{bmatrix} = \begin{bmatrix} \cos \delta_e & 0 & -\sin \delta_e \\ 0 & 1 & 0 \\ \sin \delta_e & 0 & \cos \delta_e \end{bmatrix}$$

The coordinates of the deflected flap are then transformed back to vehicle centered coordinate system, first through the inverse rotation

$$\begin{bmatrix} X'_{\delta_e} \\ Y'_{\delta_e} \\ Z'_{\delta_e} \end{bmatrix} = \begin{bmatrix} E \end{bmatrix}^{-1} \begin{bmatrix} X'_{0\delta_e} \\ Y'_{0\delta_e} \\ Z'_{0\delta_e} \end{bmatrix}$$

and then by the coordinate shift

$$\begin{aligned} X_{\delta_e} &= X'_{\delta_e} + X_{HL4} \\ Y_{\delta_e} &= Y'_{\delta_e} + Y_{HL4} \\ Z_{\delta_e} &= Z'_{\delta_e} + Z_{HL4} \end{aligned}$$

The rotation angles are defined for a right-handed system and are found from the relationships

$$\psi = \sin^{-1} \left(\frac{X_{HL1} - X_{HL4}}{L_{XY}} \right) \text{ and } \phi = -\sin^{-1} \left(\frac{Z_{HL1} - Z_{HL4}}{L_{YZ}} \right)$$

where

$$L_{XY} = \left[(X_{HL1} - X_{HL4})^2 + (Y_{HL4})^2 \right]^{1/2}$$

and

$$L_{YZ} = \left[L_{XY}^2 + (Z_{HL1} - Z_{HL4})^2 \right]^{1/2}$$

A check is made in the program and if $Y_{HL1} < Y_{HL4}$ then the yaw rotation angle is set to $\psi = \pi - \psi$ to position the hinge line in the proper quadrant.

The third rotation angle δ_e is, of course, specified for a given problem. It should be noted in the present approach, that the coordinate system is rotated through the angle δ_e , positive in the right handed sense for the system defined. Relative to the physical problem, positive δ_e corresponds to a flap deflection into the flow.

The hinge moment factor (HMFCT) is simply a function of the element geometry and location, and is defined as follows. The total moment of an element is (considering only inviscid forces)

$$\bar{M}'_0 = -(\bar{R}'_0 \times \bar{F}) = P(\bar{R}'_0 \times \bar{N}'_0) \text{ AREA}$$

where

\bar{R}'_0 is the radius vector to the element centroid,

P is the net surface pressure,

and AREA is the element area.

The hinge line moment is just the \bar{Y}'_0 -component of the total moment;

$$M_{HL} = M_{Y'_0} = \bar{j}'_0 \cdot \bar{M}'_0 = P(\text{HMFCT})$$

where

$$\text{HMFCT} = (Z'_0 N_{X'_0} - X'_0 N_{Z'_0}) \text{ AREA}$$

Once the deflected flap is properly oriented in the vehicle centered coordinates, the force on each element and hinge moment are determined.

SECTION IV

GENERAL INTERPOLATION METHOD

The extension of the Arbitrary-Body Program to lower Mach numbers will require greater use of the Second-Order Shock-Expansion method in calculating surface pressures and flow fields. This requires that streamlines be defined prior to the start of the pressure calculations. These same streamlines may also be used in the viscous calculations and in these applications it is necessary that the number of streamlines be kept reasonably small. It would be impractical to expect that a streamline would pass through every surface element, let alone through the actual centroid of the element. It will therefore be necessary to calculate surface properties (both pressures and skin friction) along a number of streamlines, and to then use some interpolation scheme to arrive at the properties at each element centroid for use in the force integration.

Briefly, the problem may be stated as follows:

The flow properties are calculated on a grid of points defined by streamlines. The vehicle forces will be summed over a grid of points defined by the element centroids. The problem, then, is to determine the flow properties at the centroids by interpolation.

There are two general classes of interpolation. These are "interpolation in the small" or local fit and "interpolation-in-the-large" in which an entire surface or section is fit. Harder and Desmarais have presented a method, the Surface Spline, which is an ingenious resolution of the classical problem of two-dimensional interpolation. It is an "interpolation-in-the-large" scheme with all the associated convenience (irregular grids) and with accuracy rivalling the local fits.

The Surface Spline Method is the basis for the general interpolation procedure used throughout the Mark IV program. It is used for interpolating flow fields to determine interference effects, for interpolating surface velocities to calculate streamlines, and for interpolating surface properties to calculate forces (inviscid and viscous).

Surface Spline

The surface spline is based on the small deflection equation of an infinite plate that deforms in bending only. The procedure is to represent a given deflection as a symmetric deflection due to a point load at the origin. The entire surface is then taken as the sum of all the point load distributions, subject to the boundary condition that the surface becomes flat at large distances from the origin. This results in a system of linear equations which is solved for the required loads or in the present application, for the spline coefficients. The final system of equations is presented below (details of the derivation are given in Reference 5).

A function W_i is specified at n independent points (x_i, y_i) $i = 1, n$.
 A system of $n+3$ equations must be solved for the $n+3$ unknowns a_0 ,
 a_1 , a_2 and F_i ($i=1, n$).

$$\begin{aligned}
 F_1 + F_2 + \dots + F_n &= 0 \\
 x_1 F_1 + x_2 F_2 + \dots + x_n F_n &= 0 \\
 y_1 F_1 + y_2 F_2 + \dots + y_n F_n &= 0 \\
 a_0 + x_1 a_1 + y_1 a_2 + A_{11} F_1 + A_{12} F_2 + \dots + A_{1n} F_n &= W_1 \\
 a_0 + x_2 a_1 + y_2 a_2 + A_{21} F_1 + A_{22} F_2 + \dots + A_{2n} F_n &= W_2 \\
 \cdot & \\
 \cdot & \\
 \cdot & \\
 a_0 + x_j a_1 + y_j a_2 + A_{j1} F_1 + A_{j2} F_2 + \dots + A_{jn} F_n &= W_j \\
 \cdot & \\
 \cdot & \\
 a_0 + x_n a_1 + y_n a_2 + A_{n1} F_1 + A_{n2} F_2 + \dots + A_{nn} F_n &= W_n
 \end{aligned}$$

where

$$A_{ij} = r_{ij}^2 \ln r_{ij}^2$$

and

$$r_{ij}^2 = (x_i - x_j)^2 + (y_i - y_j)^2$$

It is convenient to express these equations in matrix form as follows:

$$\begin{bmatrix}
 0 & 0 & 0 & | & 1 & 1 & 1 & \dots & 1 \\
 0 & 0 & 0 & | & x_1 & x_2 & x_3 & \dots & x_n \\
 0 & 0 & 0 & | & y_1 & y_2 & y_3 & \dots & y_n \\
 \hline
 1 & x_1 & y_1 & | & A_{11} & A_{12} & A_{13} & \dots & A_{1n} \\
 1 & x_2 & y_2 & | & A_{21} & A_{22} & A_{23} & \dots & A_{2n} \\
 \cdot & & & | & \cdot & & & & \cdot \\
 \cdot & & & | & \cdot & & & & \cdot \\
 \cdot & & & | & \cdot & & & & \cdot \\
 1 & x_n & y_n & | & A_{n1} & A_{n2} & A_{n3} & \dots & A_{nn}
 \end{bmatrix}
 \begin{bmatrix}
 a_0 \\
 a_1 \\
 a_2 \\
 F_1 \\
 F_2 \\
 \cdot \\
 \cdot \\
 \cdot \\
 F_n
 \end{bmatrix}
 =
 \begin{bmatrix}
 0 \\
 0 \\
 0 \\
 W_1 \\
 W_2 \\
 \cdot \\
 \cdot \\
 \cdot \\
 W_n
 \end{bmatrix}$$

Using the partitions indicated for the system coefficient matrix and compacting the notation, this may be rewritten

$$\begin{bmatrix} B & xy^t \\ xy & A_{ij} \end{bmatrix} \begin{bmatrix} C_{kl} \end{bmatrix} = \begin{bmatrix} H_{kl} \end{bmatrix}$$

where

$$\begin{aligned} i &= 1, n \\ j &= 1, n \\ k &= 1, n+3 \\ l &= 1, M \end{aligned}$$

These equations are to be solved for the spline coefficients, $[C_{kl}]$. $[H_{kl}]$ represents the known functions at the given points:

$$H_{1l} = 0, \quad H_{2l} = 0, \quad H_{3l} = 0, \quad H_{4l} = W_1, \quad H_{5l} = W_2, \quad \text{etc.}$$

The additional parameter l refers to the number of functions to be interpolated. For example, the flow field data are interpolated for six functions; Mach number, the three-direction cosines of the velocity vector, pressure, and temperature. In this case $M=6$ and the spline coefficients are found for all six functions with one calculation of the coefficient matrix. The matrix solution is obtained using the Douglas SOLVIT Routine, details of which are given in Reference 6. The method is simply Gaussian triangularization adapted to the requirements of the computer for the case where the coefficient matrix is too large to fit into core.

Linear Spline

The same approach could be taken to define a one dimensional or linear spline. Consider a function dependent on y only. Terms involving x would be removed and the system reduced to order $n+2$. This would involve changes in the coding logic. However, the above equations are readily adapted to a function of one variable. A function independent of x is equivalent to putting x equal to a constant, say x_c . The second equation of the system becomes

$$x_c (F_1 + F_2 + F_3 + \dots + F_n) = 0$$

This is a multiple of the first equation and the system is indeterminant. Also, since the function is independent of x then a_1 should be equal to zero. This is easily accomplished by setting the term $B(2,2) = 1$. The second equation now becomes

$$a_1 + x_c (F_1 + F_2 + F_3 + \dots + F_n) = 0$$

The term in brackets is zero by the first equation and thus $a_1 = 0$. The system is no longer indeterminate and the solution of $n+3$ system proceeds as before.

The values of the B matrix are summarized as follows:

1. Surface Spline

$W_i =$ function of both x and y

$$B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

2. Linear Spline, Independent of x

$W_i =$ function of y only
 $x =$ constant $= x_c$

$$B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

3. Linear Spline, Independent of y

$W_i =$ function of x only
 $y =$ constant $= y_c$

$$B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Symmetry

If the function being interpolated has a plane of symmetry, then use can be made of images to improve the accuracy of the fit. Consider W_j specified at n points (x_j, y_j) in the range $x_1 \leq x_j \leq x_0$ and also W_j symmetrical about x_0 . The system of equations could be written including n images in the range $x_0 \leq x_j \leq (2x_0 - x_1)$. Transforming the x coordinate to

$$\xi_j = x_j - x_0$$

and using $-j$ notation to represent the images the $2n+3$ system of equations is

$$\sum_{i=-n}^n F_i = 0$$

$$\sum_{i=-n}^n \xi_i F_i = 0$$

$$\sum_{i=-n}^n y_i F_i = 0$$

$$\left[a_0 + \xi_j a_1 + y_j a_2 + \sum_{i=-n}^n A_{j,i} F_i = W_j \right] \quad j = -n, n$$

Adding symmetric pairs of equations

$$\begin{aligned} 2a_0 + a_1(\xi_j + \xi_{-j}) + a_2(y_j + y_{-j}) + (A_{j,-n} + A_{-j,-n}) F_{-n} + \dots \\ + (A_{j,-1} + A_{-j,-1}) F_{-1} + (A_{j,1} + A_{-j,1}) F_1 + \dots \\ + (A_{j,n} + A_{-j,n}) F_n = (W_j + W_{-j}) \end{aligned}$$

Using the definition of symmetry

$$\xi_{-i} = -\xi_i$$

$$y_{-i} = y_i$$

$$W_{-i} = W_i$$

and also, that $A_{ij} = r_{ij}^2 \ln r_{ij}^2$

$$\text{where } r_{i,j}^2 = (\xi_i - \xi_j)^2 + (y_i - y_j)^2$$

it is easily shown that

$$r_{-i,j}^2 = r_{i,-j}^2 = (\xi_i + \xi_j)^2 + (y_i - y_j)^2$$

and

$$r_{i,-j}^2 = r_{i,j}^2$$

Therefore,

$$A_{-i,j} = A_{i,-j}$$

and the system becomes

$$\sum_1^n F_i = 0$$

$$\sum_1^n y_i \bar{F}_i = 0$$

$$\left[a_0 + y_j a_2 + \sum_{i=1}^n (A_{j,i} + \bar{A}_{j,i}) \bar{F}_i = W_j \right] \quad j = 1, n$$

where

$$\bar{A}_{i,j} = \bar{r}_{ij}^2 \ln \bar{r}_{i,j}^2$$

$$\bar{r}_{i,j} = (x_i + x_j - 2x_0)^2 + (y_i - y_j)^2$$

The order is reduced to $n+2$ and $a_1 = 0$. As was the case for a linear spline, the $n+3$ system can be solved by changing the B matrix and setting $x = x_c$ in the $[xy]$ and $[xy^t]$ matrices.

Similar results can be obtained for y symmetry and for both x and y symmetry.

Application of the Surface Spline for Interpolation

The surface spline, or any other interpolation scheme, needs to be specified in appropriate coordinates to do the job correctly. For example, consider the flow over a swept wing. Interpolation relative to the space coordinates used to define the quadrilateral elements will produce erroneous results. The interpolation must be done in coordinates consistent with the physics of the problem and for flow on a swept wing, distance from the leading edge and distance along the span would be proper.

The surface spline, due to the nature of the basic solution (symmetric point load) works best in a one-to-one domain of the independent variables. That is, for the swept wing, coordinates of relative chord and relative span would be used defining a range of both x and y from 0.0 to 1.0.

The process of selecting and scaling the appropriate coordinates is referred to as normalization in the Mark IV program. Two different categories of data normalization are used in the Mark IV program. These are surface data and flow field data. The normalization procedures for each are described in the following sections.

Surface Data Normalization

Interpolation of surface data is involved in the Surface Streamline Option, in the Input Pressure Option, and in the Viscous Program Option. The surface geometry, input in body reference coordinates (xr, yr, zr) is first transformed to the required local reference system. The local system is defined by the orientation parameters x_0 , y_0 , z_0 , ψ_0 , θ_0 , and ϕ_0 . The local coordinates (x, y, z) are given by

$$[x, y, z] = [T] [xx, yy, zz]$$

where

$$[T] = \begin{bmatrix} (\cos\theta_0 \cos\psi_0) & (\cos\theta_0 \sin\psi_0) & (-\sin\theta_0) \\ (-\cos\phi_0 \sin\psi_0 + \sin\phi_0 \sin\theta_0 \cos\psi_0) & (\cos\phi_0 \cos\psi_0 + \sin\phi_0 \sin\theta_0 \sin\psi_0) & (\sin\phi_0 \cos\psi_0) \\ (\sin\phi_0 \sin\psi_0 + \cos\phi_0 \sin\theta_0 \cos\psi_0) & (-\sin\phi_0 \cos\psi_0 + \cos\phi_0 \sin\theta_0 \sin\psi_0) & (\cos\phi_0 \cos\theta_0) \end{bmatrix}$$

$$\begin{aligned} \text{and} \quad xx &= xr - x_0 \\ yy &= yr - y_0 \\ zz &= zr - z_0 \end{aligned}$$

Also calculated are the axial, radial, and meridian coordinates:

$$\begin{aligned} A &= x \\ \bar{R} &= (y^2 + z^2)^{1/2} \\ \phi &= \text{ARCTAN}(y/-z) \end{aligned}$$

Six coordinates (x, y, z, A, R, ϕ) are now available in the local reference system and the pair of independent variables to be used for interpolation are selected by the input flag INORM. The five options available are;

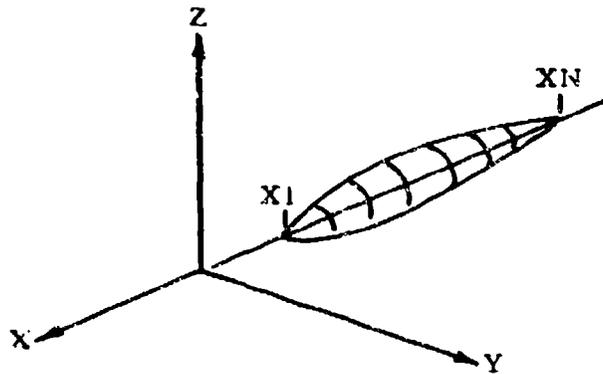
$$\begin{aligned} \text{INORM} = 0, \quad \phi &= f(A, R) \\ \text{INORM} = 1, \quad z &= f(x, y) \\ \text{INORM} = 2, \quad y &= f(x, z) \\ \text{INORM} = 3, \quad x &= f(y, z) \\ \text{INORM} = 4, \quad R &= f(A, \phi) \end{aligned}$$

To scale the data, the surfaces are grouped into two types: bodies and lifting surfaces (indicated by the input flag ISURF = 0 and 1, respectively). More complex surfaces may be composed from combinations of these two types. In addition, four boundary points must be input to scale the data. These points are input in the body reference system and transformed to the six local coordinates. The boundary data are used differently for each surface type.

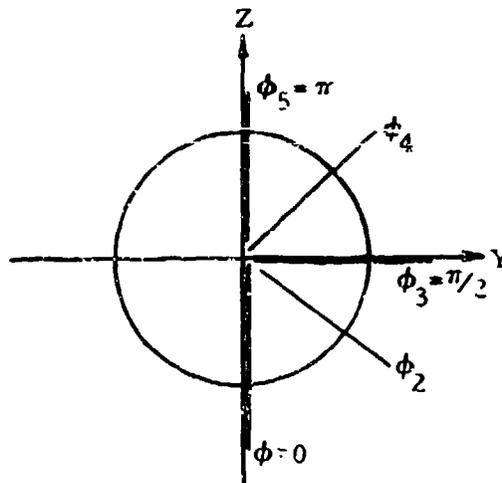
An example of the use of the boundary data for each of the two surface types is discussed in detail.

I. Bodies (ISURF = 0)

Consider a fuselage with approximately an axially-symmetric cross-section shape about the x-axis.



The flow field is calculated using the Second-Order Shock-Expansion Method in a number of meridian planes, say five.



Having done this, the flow properties at the centroids of the elements are required in order to calculate the forces or streamlines on the body. The meridian flow data has been stored on Unit 10 and will be recalled for use in the surface interpolation routine.

For this case the obvious choice of independent variables is the axial coordinate (A) and the meridian angle (ϕ) (i. e., INORM = 4).

The following boundary data are input:

$$\begin{array}{ll} \text{XB}(1) = \text{X1} & \text{XB}(2) = \text{XN} \\ \text{YB}(1) = 0.0 & \text{YB}(2) = 0.0 \\ \text{ZB}(1) = 0.0 & \text{ZB}(2) = 0.0 \end{array}$$

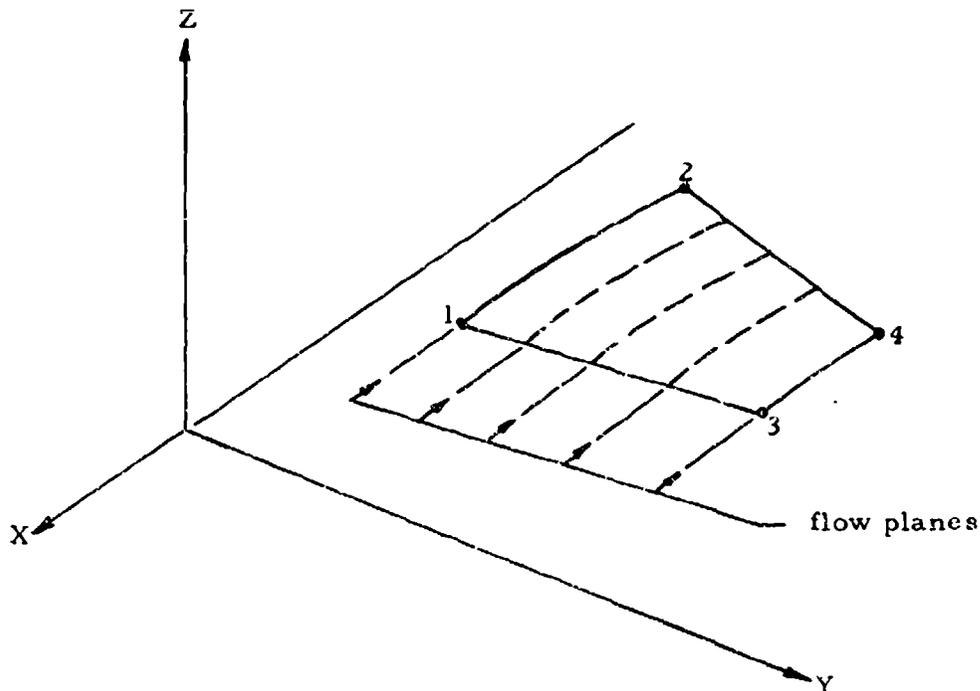
$$\begin{aligned}
 XB(3) &= 0.0 & XB(4) &= 0.0 \\
 YB(3) &= 0.0 & YB(4) &= 0.0 \\
 ZB(3) &= -1.0 & ZB(4) &= 1.0
 \end{aligned}$$

The interpolation will be performed over the length of the body ($X1 \leq X \leq XN$) and for ($0 \leq \phi \leq \pi$).

If there happen to be large variations in flow properties between $\phi = 0$ and $\phi = \pi$ (because of large α , or M_{∞}), then the interpolation could be segmented. For example, use three segments with ranges ($0.0 \leq \phi_1 \leq 60^\circ$), ($60^\circ \leq \phi_2 \leq 120^\circ$) and ($120^\circ \leq \phi_3 \leq 180^\circ$). Since the surface spline is a global fit, segmenting will relax the constraints that must be met.

II. Lifting Surface (ISURF = 1)

Consider a wing whose leading edge and trailing edge are approximately in the x, y plane. The flow properties on a lifting surface vary essentially with relative chord (x/c) and relative span (y/b) and the surface should be normalized with respect to these parameters. Therefore, use $INORM = 1$.



The four boundaries are input as indicated on the sketch. They are first transformed to local coordinates and the following parameters calculated:

$$\text{Root Chord, } CR = XB(2) - XB(1)$$

$$\text{Tip Chord, } CT = XB(4) - XB(3)$$

$$\text{Local Span, } B = YB(3) - YB(1)$$

The normalized coordinates (x/c , y/b) for a given point on the surface (x, y, z) are

$$x/c = (x - XLE)/CY$$

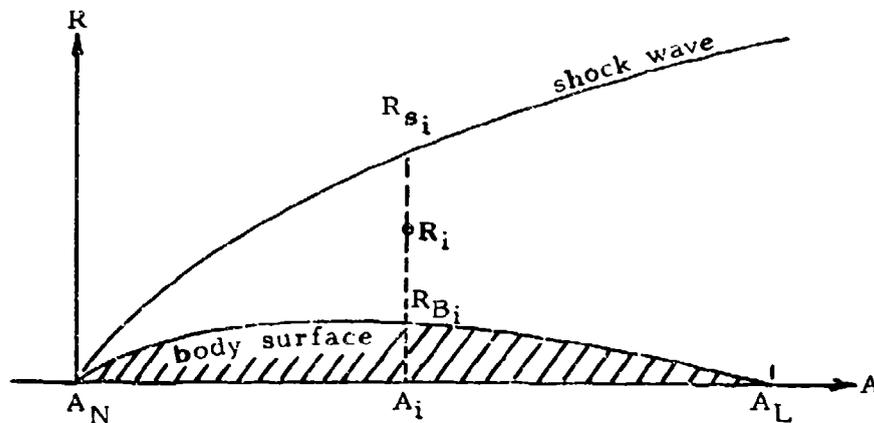
$$y/b = (y - YB(1))/B$$

where $XLE = XB(1) + (XB(3) - XB(1)) * y/c$

and $CY = CR + (CT - CR) * y/c$

Flow Field Data Normalization

The flow field about a component is made up of various flow regions. In the previous example of flow on a body, each meridian plane would be designated a flow region. Each region is specified by two boundary curves (the body surface and the shock wave) and, if desired, by points within the field.



A typical flow region is shown in the sketch in the form of radial versus axial distance. The nose station is A_N and the body length is shown at A_L . Proper normalization is obtained using a relative axial distance and the shock-layer distance.

For a point located at A_i, R_i , the normalized coordinates are

$$\bar{A}_i = \frac{A_i - A_N}{A_L - A_N}$$

$$\bar{R}_i = \frac{R_i - R_{B_i}}{R_{S_i} - R_{B_i}}$$

Curves of body radius (R_B) and shock radius (R_S) as a function of axial distance A are obtained using the linear spline.

As an example of the appropriateness of the normalization and the accuracy of the surface spline, a conical flow field is shown in Figure 6. While this is a particularly simple case, exact analytical results are available for definition and comparison. The flow region was defined by six points on each boundary curve (at stations $A = 1.0, 2.0, 4.0, 6.0, 8.0$ and 10.0) and five interior points were specified at each of three stations ($A = 1.0, 6.0$ and 10.0). Figure 6 presents results for speed of sound ratio, pressure ratio, and radial velocity component interpolated at $A = 4.0$, and clearly show the accuracy of the method.

$$M_\infty = 4.0$$

$$\theta_c = 10^\circ \quad \Lambda = 4.0$$

— Exact (AGARDOGRAPH 137)
 ○ Surface spline interpolation

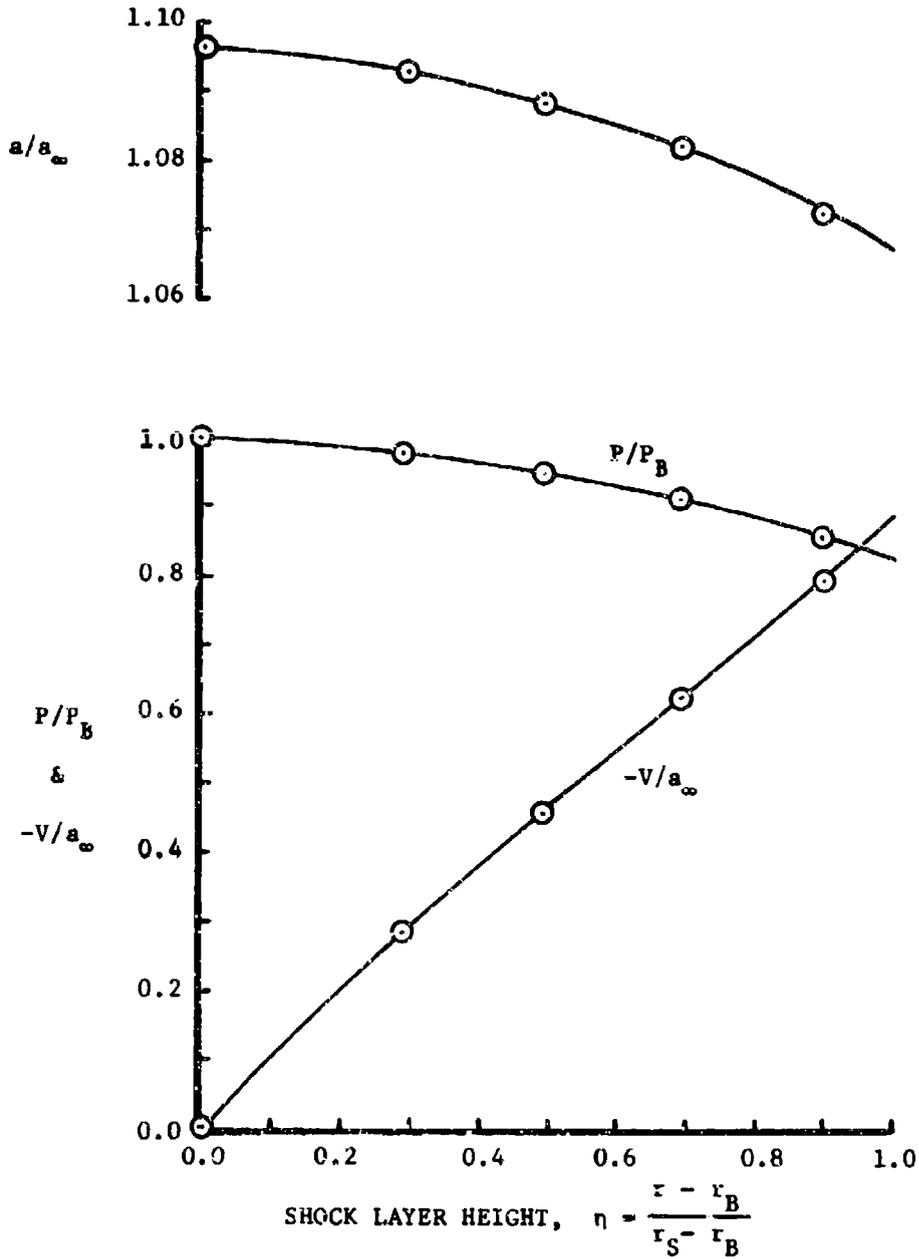


Figure 6. Conical Flow Field

SECTION V

FLOW FIELD ANALYSIS METHODS

At very high hypersonic Mach numbers the vehicle generated flow system is relatively close to each vehicle component. For these types of flow systems the importance of component interference is diminished and component build-up methods may be used with considerable success. However, as the flight Mach number is reduced down into the supersonic speed range interference effects become very significant. Interference effects also may be important even at the hypersonic Mach numbers if the vehicle is composed of discrete components such as is the case for airplane type wing-body-fin configurations. Past experience has also indicated that even the blended or all body shapes may have significant interference-type effects when analyzed in yaw.

A really accurate analysis of this problem including interference effects would require a three-dimensional method of characteristics solution. However, present mathematical and programming techniques and digital computer size and speed limitations preclude the application of the method of characteristics to typical preliminary design problems.

Prior to the advent of the large scale digital computer a number of approaches were used in the analysis of interference effects on wing-body-fin configurations at supersonic speeds. These methods, for example the work of Kaatiari in Reference 7, have since been replaced by the linear theory finite-element computer programs. However, these early hand computational methods did do a pretty good job in the low supersonic Mach number range and for the simple wing-body-fin configurations for which they were derived. The general approach in these methods was to look at each aspect of the flow and, with appropriate assumptions and simplifications of the vehicle shape, to approximate the overall effect of the flow on downstream components. These methods usually did not give detailed pressure distributions, but instead only accounted for the interference effects in a gross way on the final vehicle aerodynamic coefficients. Of course, significant changes in vehicle shape (such as body cross-section) were not always reflected in answers. Also, frequent use of slender body theory meant that the results could not be extended up into the hypersonic speed region.

The use of the digital computer has led to methods that largely replace these older hand, "engineering" methods of solution (the one notable exception being the USAF DATCOM). These computerized methods, which are usually based on linearized theory, have been summarized by Carmichael in Reference 8, and by Bradley and Miller in Reference 9. One method in wide use today is the one of Woodward (Reference 10) that uses finite elements or boxes to which potential methods are applied. However, the computer programs based on these methods are presently restricted to simple body-of-revolution and wing combinations. Even with the addition of dihedral panels, as has been recently achieved,

the representation of a shape is far from what could be called "arbitrary". This fact, coupled with the inherent limitations of linear theory, means that it has limited direct use on the more general problems involving complex shapes and a wide Mach number range.

The interference problem, therefore, resolves into one that (1) demands the ability to handle arbitrary shapes from the geometry standpoint, and (2) accounts for interference effects using engineering methods, yet retains all basic features of the true flow fields.

The fundamental approach taken in the Mark IV program is one of flexibility. It was desired that the surface pressure method used for one component not be inherently related or dependent on the flow field method used on another. The vehicle is represented by a number of components and the most appropriate flow field method is used depending upon the component shape and flight condition. This analogous approach was a large factor in the success of the Mark III program.

The Mark III program, which already does a pretty good job of predicting the vehicle characteristics, may be looked upon as the first order solution. Its weaknesses can be mainly attributed to certain regions (e. g. , vertical fin, wing carryover to fuselage) associated with particular conditions (e. g. , high angle of attack or yaw) and shielding effects. It thus seems logical to build up the vehicle flow field in a step-by-step or component-by-component fashion. For example, the vertical would be analyzed in the symmetry plane subject only to the body flow field. Next, the effect of the wing field alone, then the sum of body and wing. Finally, the combination of the wing (analyzed subject to the body field) and the body field. In the last case the complete body flow field is not required to define the wing field, but just the body flow field properties in the vicinity of the wing leading edge are sufficient.

The framework of the Mark IV program is designed to facilitate just this type of operation. The flow field data of a component can be saved and then interrogated during subsequent calculations or future runs. In the pressure calculations, a given component is identified by the user as possibly being influenced by up to four flow regions. The local properties in each flow region must have been previously generated or input and stored on the flow field direct access data unit 10. In the analysis, each element is first checked to see what flow region it is in. The appropriate flow table is then selected and the local properties determined using the surface spline method.

Component Flow Field Analysis

The Mark IV program is structured so that a variety of flow field methods can be employed. A proper perspective of the program capabilities is obtained by an examination of methods listed in the accompanying table. The solutions have been separated into four main categories with selected individual methods appropriately listed. Those enclosed in BOXES

are methods incorporated in the Mark IV program. Those marked with an asterik (*) are available and could be added at some future time. The remaining methods are considerably more involved with regard to both complexity and increased run time and far exceed the requirement of "engineer methods". In principle, however, they could be added to the basic framework of the new program for special purposes or final design point analysis.

To obtain this flexibility of choice, a common interface between the flow field methods and the rest of the program was established. The flow field data about a given component are specified in a number of planes. For example, the flow field about a body of revolution would be defined in meridian planes. This concept of flow planes was arrived at by consideration of (1) the shock-expansion method as a primary means of generating the data, and (2) the Surface Spline Method as primary user of the data. All the flow data are stored on a direct access unit (10) in a standard format and are readily accessible by other options of the program. For example, the flow field about a body is generated using the second-order shock-expansion method in the Flow Field Option of the program. This data may then be accessed by other options of the program to calculate the following:

1. Forces on the wing subject to the body field.
2. Forces on the body.
3. Streamlines on the body and the viscous forces on these streamlines.

TABLE OF FLOW FIELD ANALYSIS METHODS

I INPUT FLOW FIELD

Tabular Distributions of Quantities
Throughout the Flow Field

II EMPIRICAL APPROACHES

Shock Shape Correlations

Surface Pressure Correlations

Local Correlation Factors

III APPROXIMATE ANALYTICAL APPROACHES

Generalized Shock Expansion

Second-Order Shock Expansion

Conical Shock Expansion

*Two-Dimensional and Axially Symmetric
Method of Characteristics

*Linear Theory (Potential Solutions)

*Linear Theory (Wave Drag)

Linearized Method of Characteristics

Iterative Schemes

IV EXACT ANALYTICAL APPROACHES

*Conical Method of Characteristics

Integral Methods

Three-Dimensional Method of Characteristics

Shock-Expansion Method

The concept of shock-expansion was first introduced by Epstein in 1931 for calculating airfoil pressures and was extensively developed by the NACA in the early 1950's. Briefly, it was extended by Eggers, Syvertson, and Kraus (Reference 11) to include the determination of the shock shape and thus the entire flow field, and further by Eggers and Savin (References 12 and 13) as the "Generalized Shock Expansion Method" to include three-dimensional hypersonic flows. A so-called second-order term in surface pressure was later added by Syvertson and Dennis (Reference 14). The generalized method was derived from consideration of the full three-dimensional characteristics theory. Through an order-of-magnitude analysis based on the hypersonic similarity parameter, it was shown that disturbances associated with divergence of streamlines in planes tangent to the surface are of secondary importance compared to those associated with the curvature of streamlines in planes normal to the surface. It was further shown, consistent with the above result, that the streamlines may be taken as geodesics. For a body of revolution then, the flow may be analyzed in meridian planes; a result exactly true at zero angle of attack and only approximate if the body is inclined to the flow.

The basic premise underlying the shock expansion procedure is that only the principal characteristics in the flow need be considered, with reflections from the shock wave and from vortex lines being negligible. Development of the theory is presented in most texts on high speed flow (e. g., Hayes and Probstein, Reference 15) and the detailed equations are presented in the aforementioned NACA publications. In the following discussions on the application of the method in the Mark IV program, only those equations vital to the presentation are given. The shock-expansion methods are collectively referred to as the Shock-Expansion Method. The specific forms of two-dimensional, three-dimensional, or second-order are accessed via input flags which simply include or delete terms as required.

The starting point for the Shock-Expansion Method was a program developed in Reference 16, and was very helpful on getting a "quick jump" on the problem. The final form is highly modified, incorporating for example, the Mark IV oblique shock and cone solutions. The experimental data of NASA TN D-6480 (Reference 17) were used extensively as a guide in the exploration of various alternatives.

To use the shock-expansion method it is first necessary to define the flow line or path along which the calculations are to be made. Ideally, such a path should be a streamline but generally this is not known. The true path is approximated by a flow line defined as the intersection of the flow plane and the surface geometry. The flow plane (also referred to as the cutting plane) may be specified with arbitrary orientation and the profile shape is automatically obtained from the quadrilateral elements.

Surface Pressure

The profile shape is represented by a series of wedge or cone frustums as shown schematically in Figure 7. The flow on the first segment is given by the oblique shock or cone solution and the pressure on the downstream frustums is given by

$$P = P_c - (P_c - P_2) e^{-\eta}$$

where

P_c is the pressure on a cone of the same angle as the frustum

P_2 is the pressure resulting from a two dimensional expansion between successive frustums

η is proportional to the pressure gradient and distance downstream of the corner

If the surface is two dimensional or only first order expansion is desired, then $\eta = 0$ and the pressure is simply

$$P = P_2$$

The relationship between the first and second order pressure are also shown on Figure 7.

Calculations were made for the configuration of NASA TN D-6480 (Reference 17) which is shown in Figure 8 as loaded using the Ellipse Generation and Aircraft Geometry Options. Comparisons of the first and second order shock expansion methods with the experimental data at zero angle of attack are presented in Figures 9 and 10. Also shown are results from a Method of Characteristics Program (based on the supersonic flow field programs developed at NASA by Inouye, Rakich, and Lomax, Reference 18). The data aft of $x/L = 0.5$ are influenced by the wing and should not be considered in the present comparisons as this effect is not accounted for in the calculations. All three methods used conical flow starting conditions. The agreement between the second order expansion and the method of characteristics is good at Mach = 2.3 (Figure 9) and excellent at Mach = 4.63 (Figure 10). Both are in far better agreement with experiment than the first order shock expansion, especially at the lower Mach number. The first order method is very sensitive to the starting cone solution. The second order method does not have this deficiency as the pressure is continually adjusted by the limiting cone value, P_c .

Since these results are at zero angle of attack, P_c was obtained using the tangent-cone method.* Attention is now directed to the angle of attack

*See Section VIII, Inviscid Pressure Methods

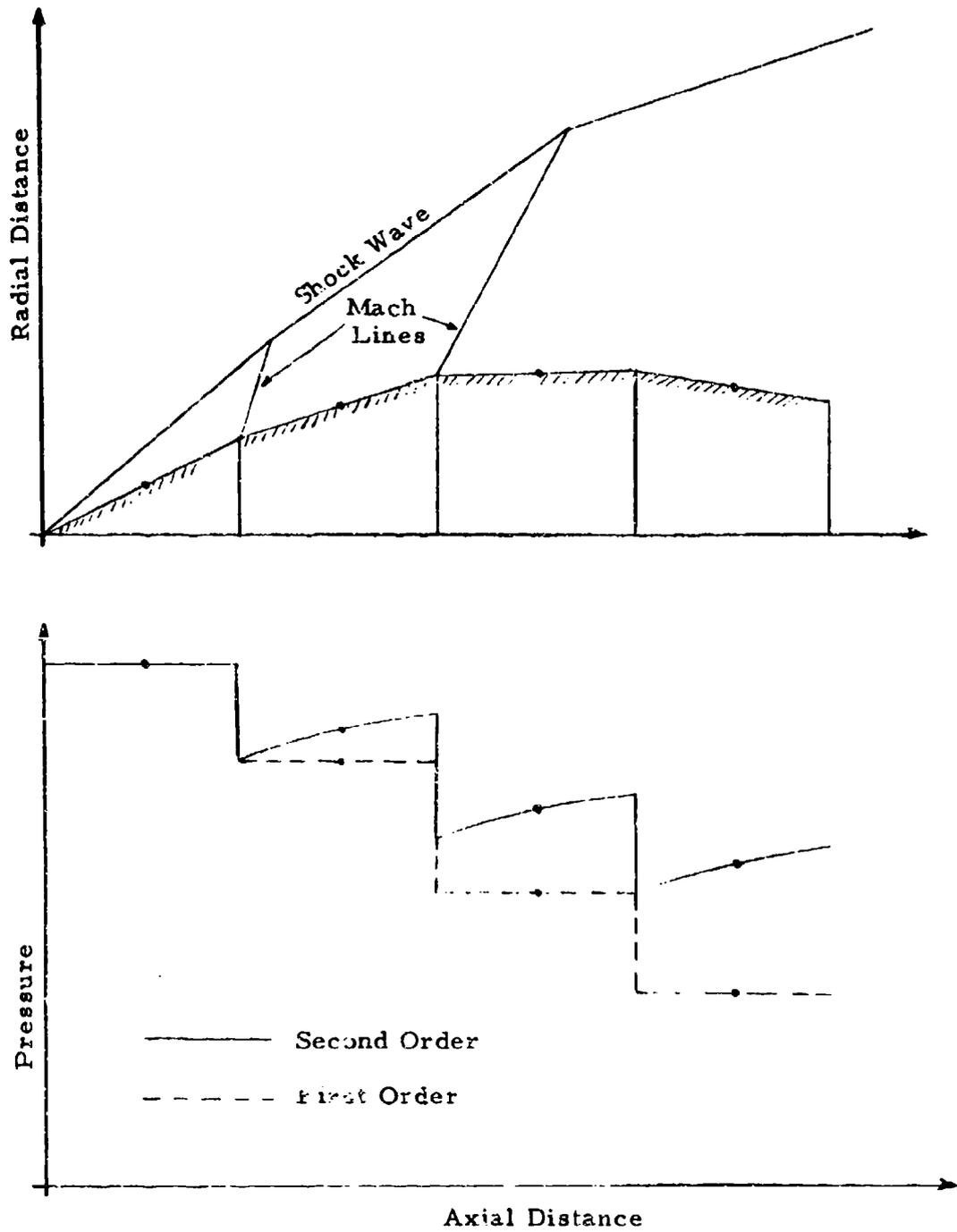


Figure 7. Schematic of Shock Expansion Method

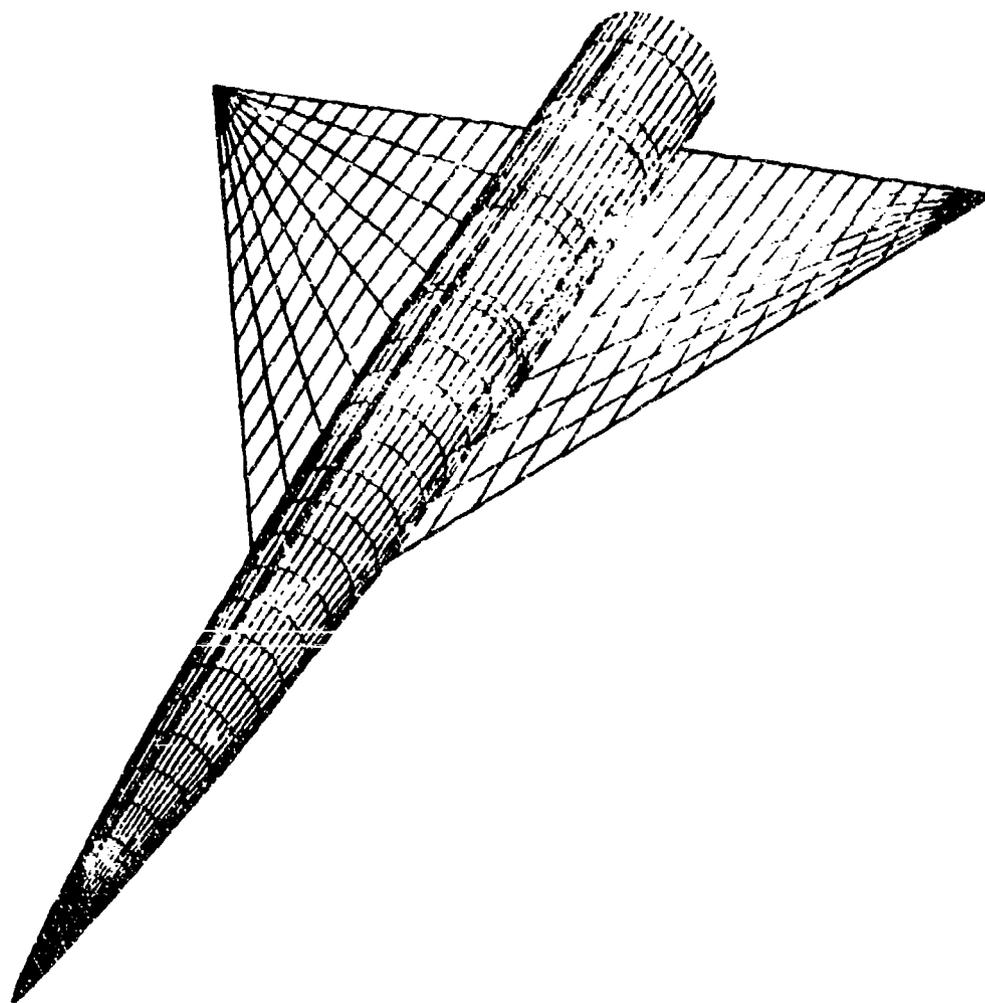


Figure 8 . Geometric Representation of NASA TN D-6480 Configuration

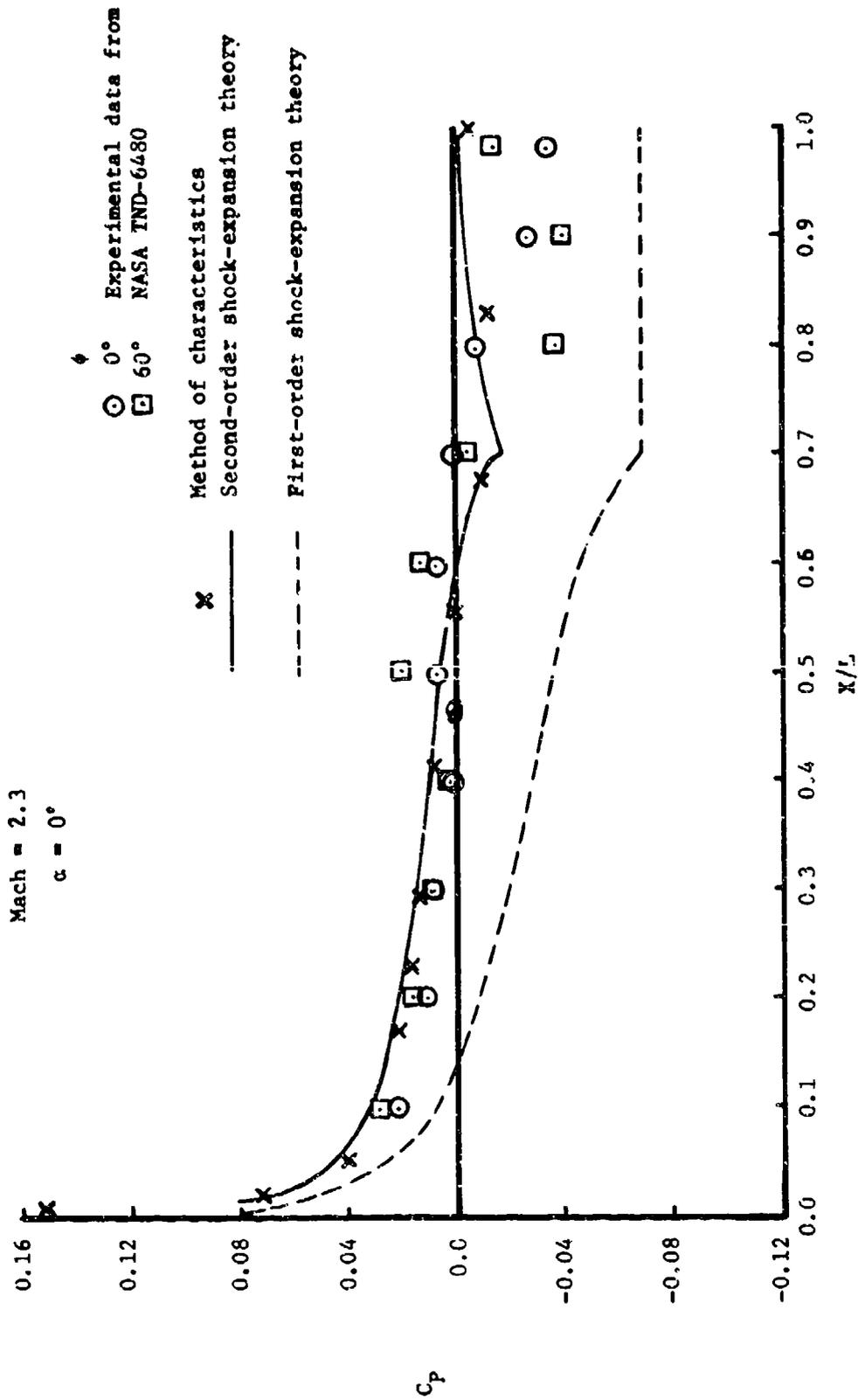


Figure 9. Comparison of Theoretical and Experimental Body Pressure Data at Mach = 2.3

Mach = 4.63

$\alpha = 0^\circ$

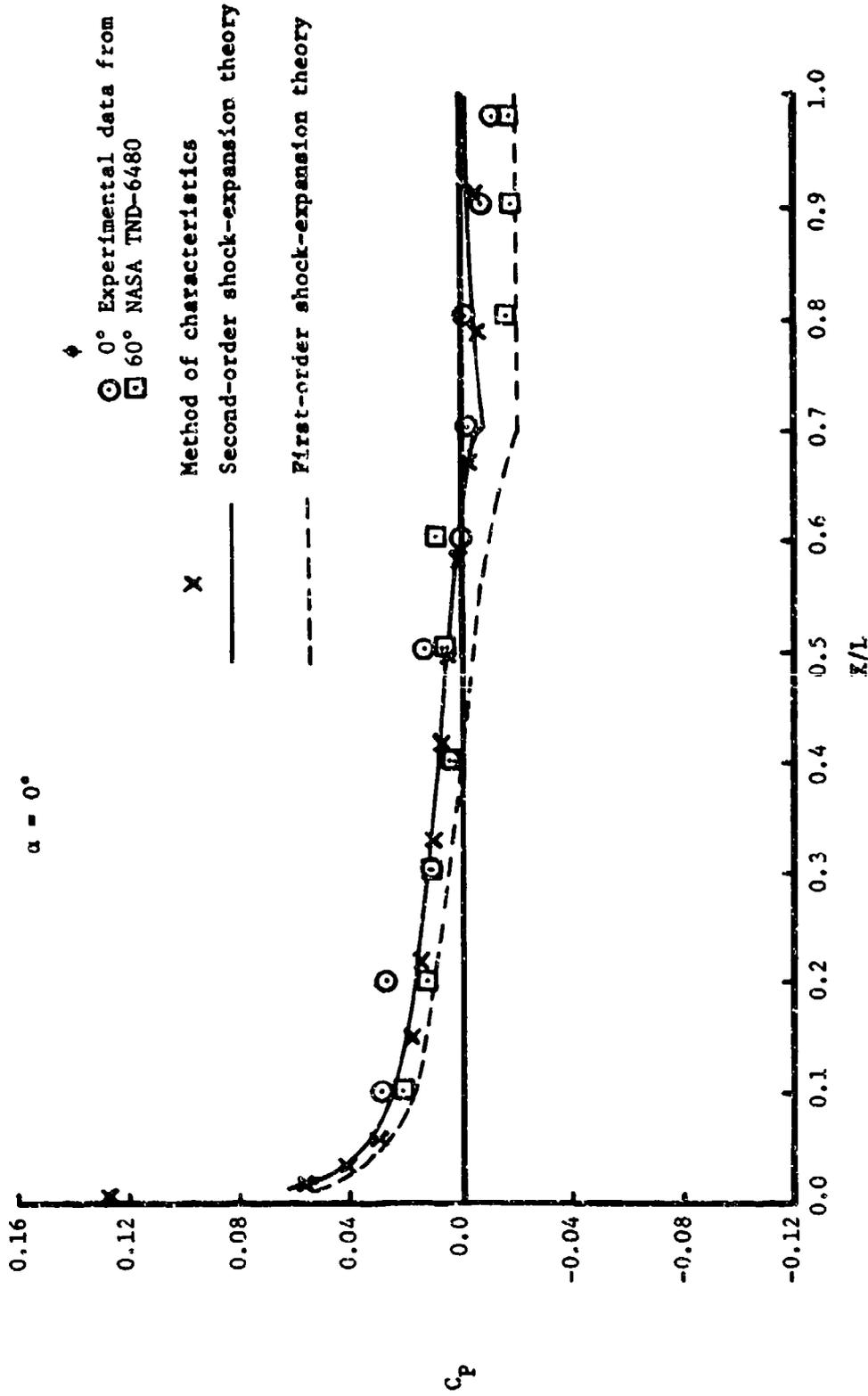


Figure 10. Comparison of Theoretical and Experimental Body Pressure Data at Mach = 4.63

cases and, in view of the results just given, only the second-order shock-expansion method was used in the analysis. The first attempt used the local impact angle to define the tangent-cone limiting condition. Typical results at Mach = 2.3 on the windward centerline ($\phi = 0$) are shown in Figure 11 and the predicted pressures were too large. It was then decided to use the inclined cone method* to define the limiting conditions and the results at the same conditions are shown in Figure 12. The agreement of the calculated and experimental pressure coefficients is good and very much better than the simple tangent-cone approach.

Comparison of the circumferential pressure distributions are given for the Mach = 4.63 case in Figures 13 through 17. The $\phi = 0$ results are in very good agreement whereas the $\phi = 60$ and $\phi = 120$ results are only in fair agreement. The data along the leeward centerline ($\phi = 180$) showed higher C_p 's than the $\phi = 120$ meridian data. This was felt to be the result of a viscous induced recompression and these data have been left off the figures for clarity. Also, only those data not influenced by the presence of the wing have been included on the plots. To check the method at stations aft of the maximum diameter, the body-alone configuration of Reference 19 was also analyzed. This is a body of revolution symmetrical about $x/L = 0.57$ and truncated with a finite base diameter. Pressure distributions are compared at Mach = 2.5 at zero angle of attack (Figure 18), 2° angle of attack (Figure 19), and 4° angle of attack (Figure 20). Both the windward ($\phi = 0$) and leeward ($\phi = 180$) centerline data are shown for the angle of attack cases. The windward data are in good agreement over the length of the body except at the very aft locations. These discrepancies are probably a result of sting interference. The leeward data show the recompression effect previously mentioned, being more pronounced on the aft portions of the body where the viscous effects become dominant.

Shock Wave Shapes

At zero angle-of-attack, the Douglas developed tangent-cone method provides excellent results for both the surface pressure and shock wave angle. At angle of attack, the inclined cone method provides two means of predicting surface pressure. However, no similar method for predicting shock wave angle is available. What is needed is a relationship for shock angle analogous to Jones' pressure coefficient formula. Lacking this, an empirical solution has been devised which follows the trend of exact results. In summary, the tangent-cone impact method is used to calculate the Mach number normal to the shock wave, which in turn is used to calculate the pressure ratio across the shock. This pressure ratio is then modified by a factor to provide agreement with the zero angle of attack results.

*See Section VIII, Inviscid Pressure Methods

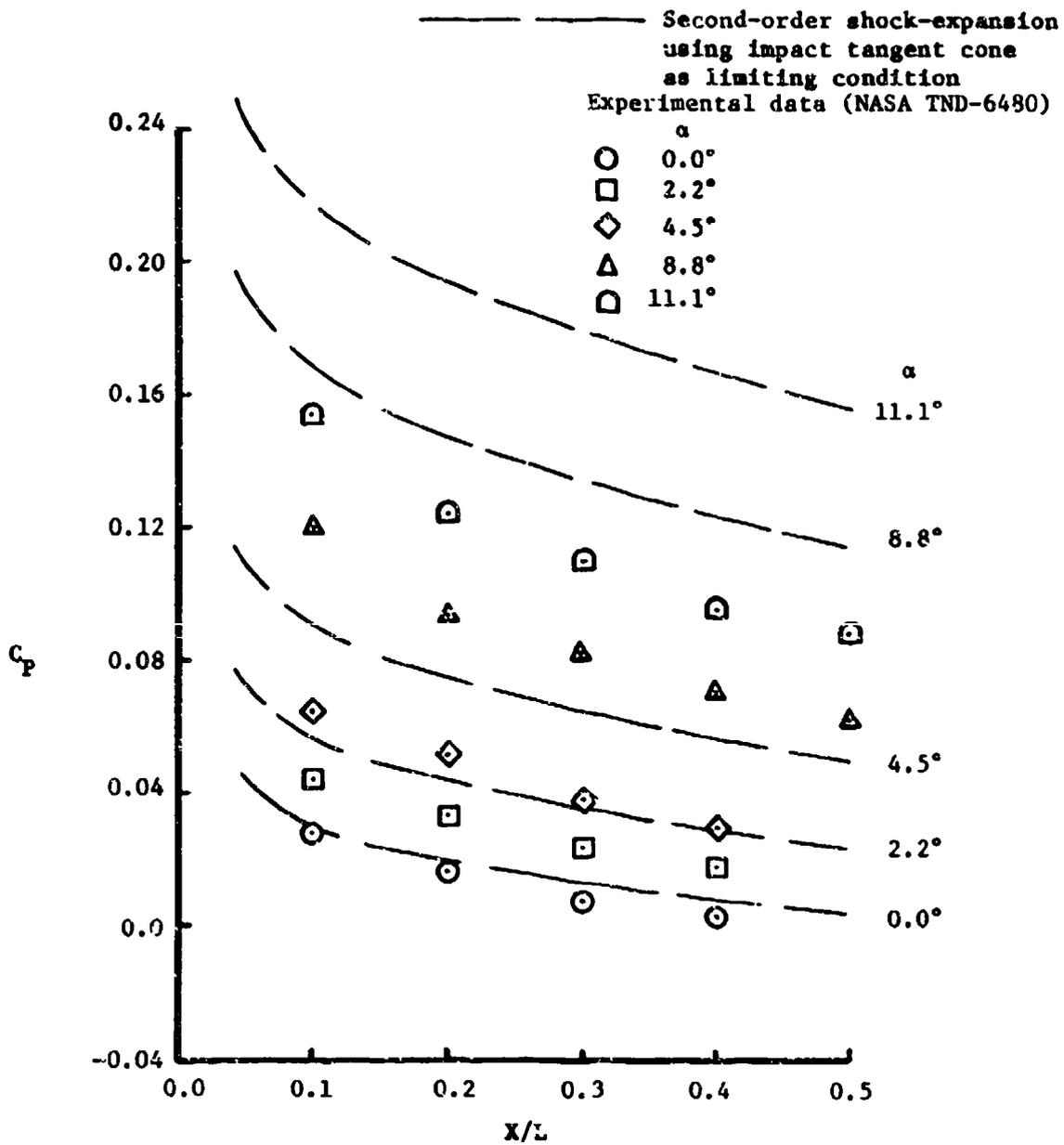


Figure 11. Comparison of Experimental and Theoretical Body Pressure Data Using Impact Tangent Cone as Limiting Condition; $M_\infty = 2.3$, $\phi = 0^\circ$

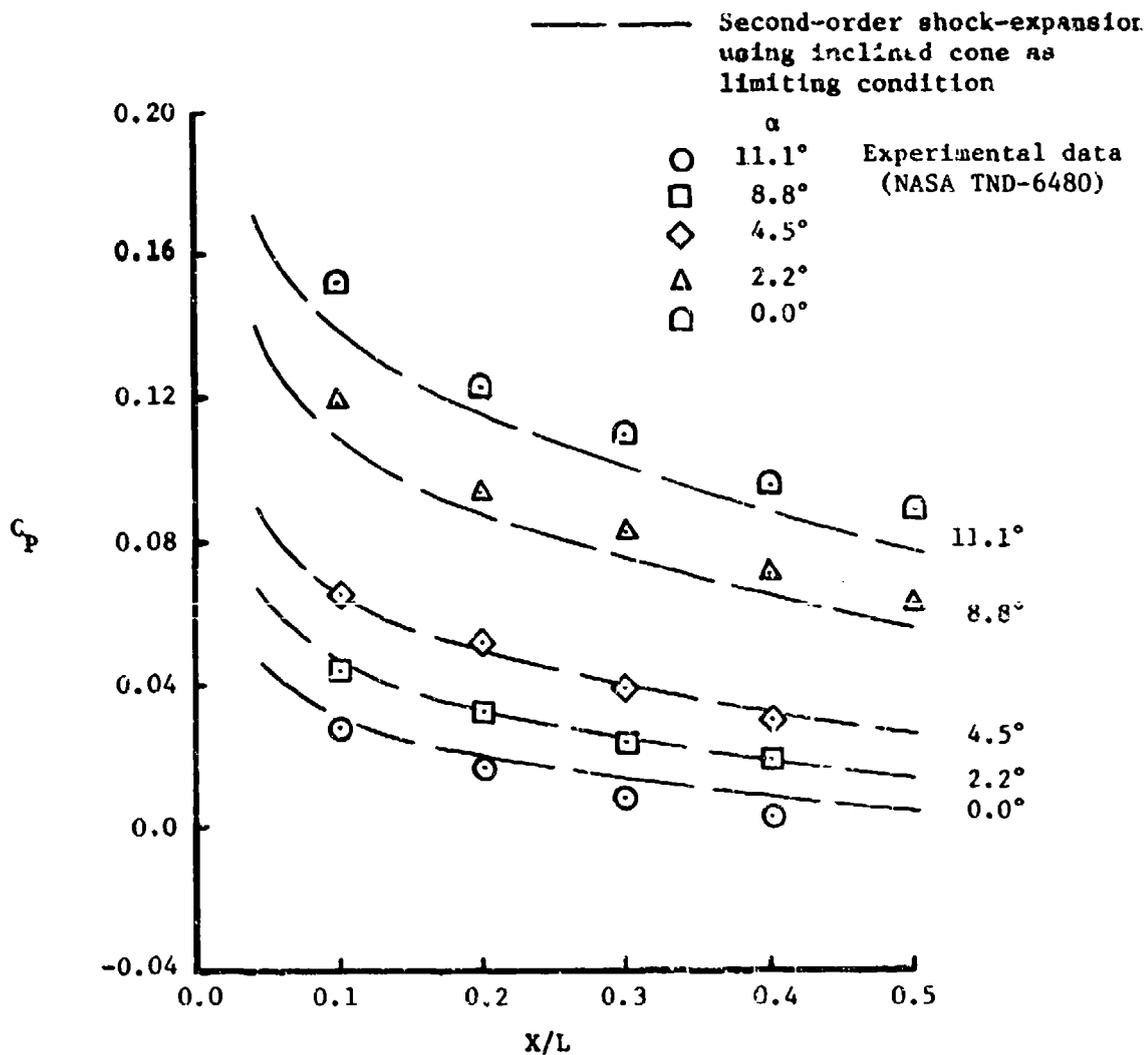


Figure 12. Comparison of Experimental and Theoretical Body Pressure Data, Using Inclined Cone as Limiting Condition; $M_\infty = 2.3$, $\phi = 0^\circ$

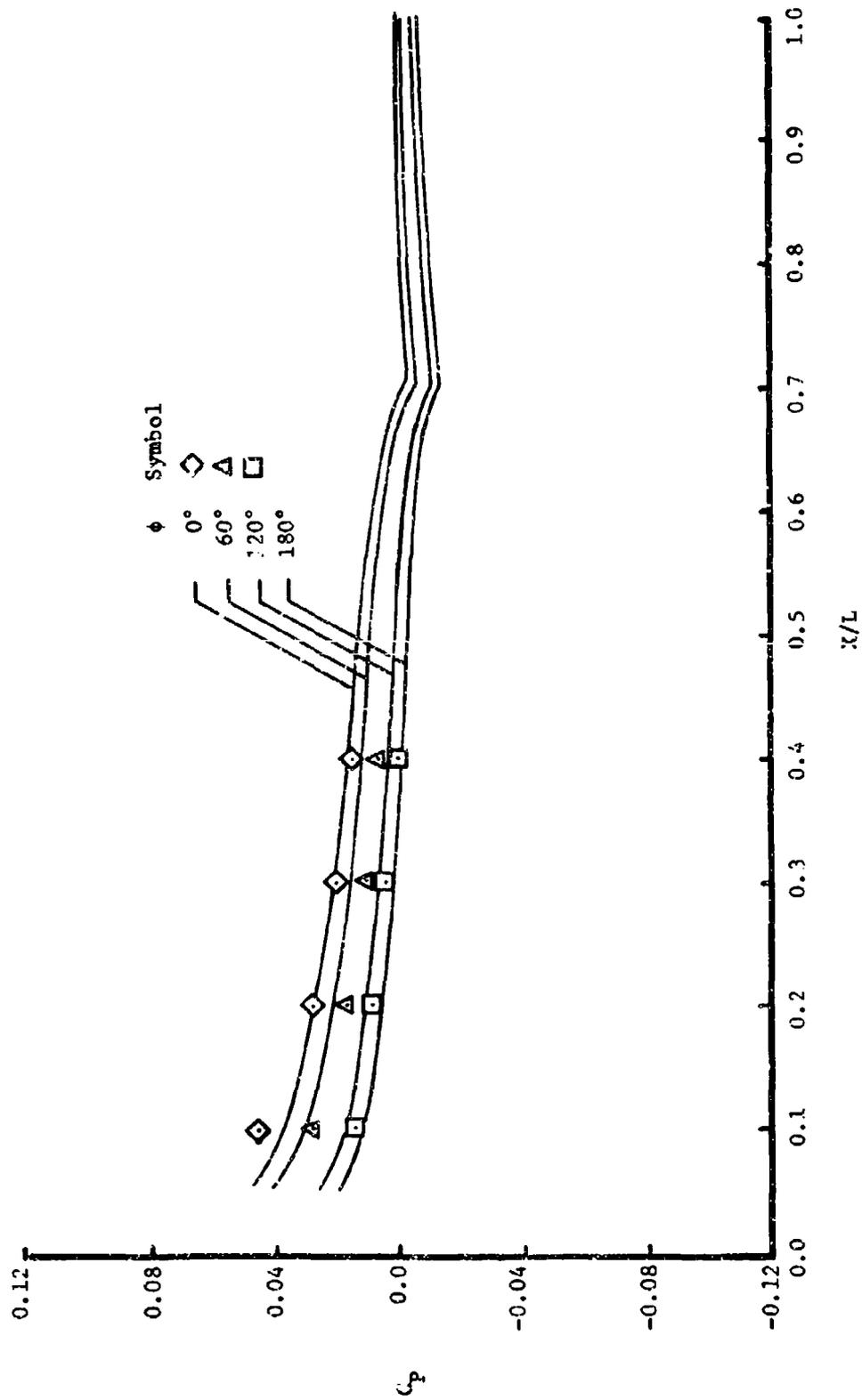


Figure 13. Comparison of Experimental Pressure Data with Theory on the Body: $M_\infty = 4.63$, $\alpha = 2^\circ$

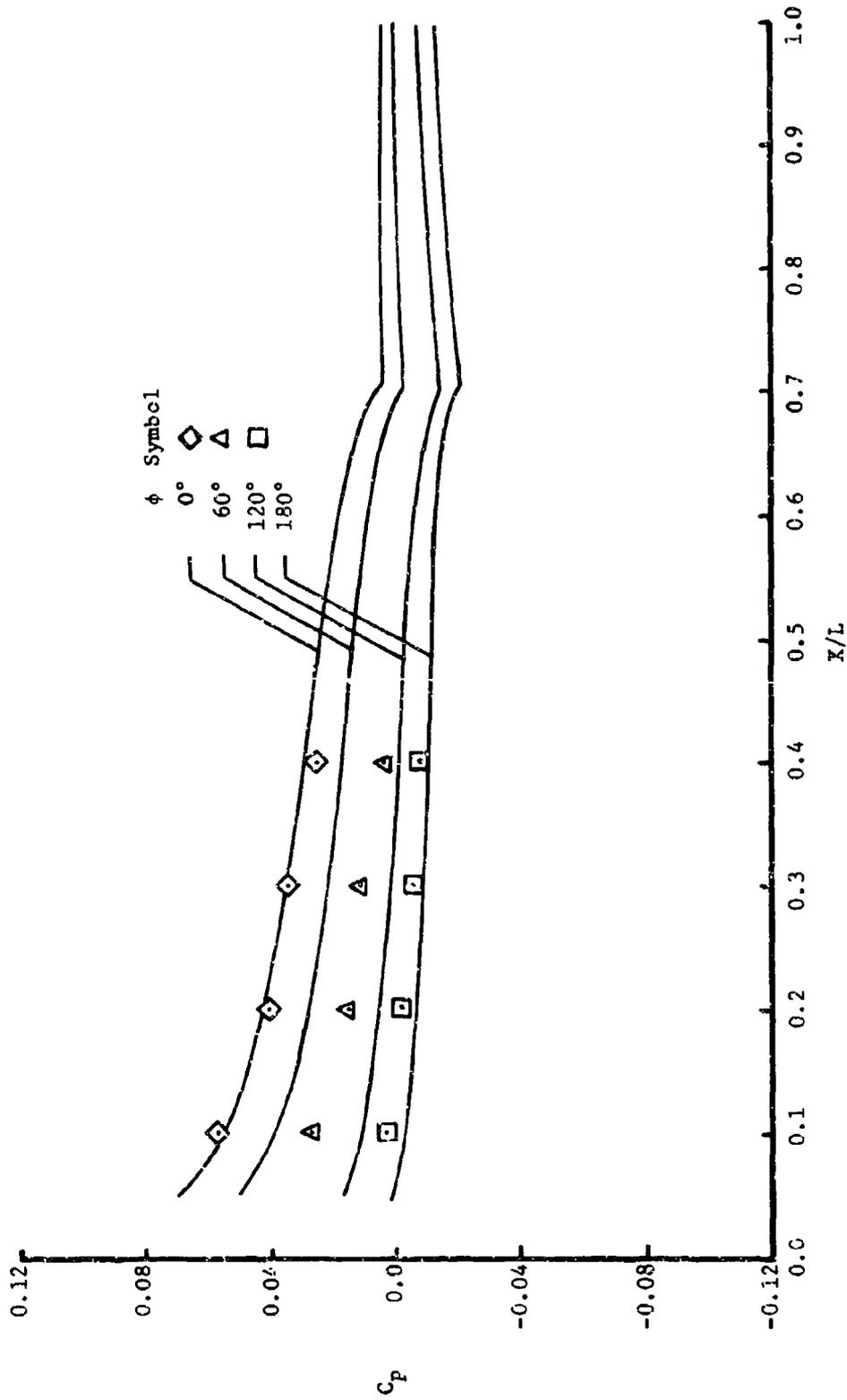


Figure 14. Comparison of Experimental Pressure Data with Theory on the Body; $M_\infty = 4.63$, $\alpha = 4.2^\circ$

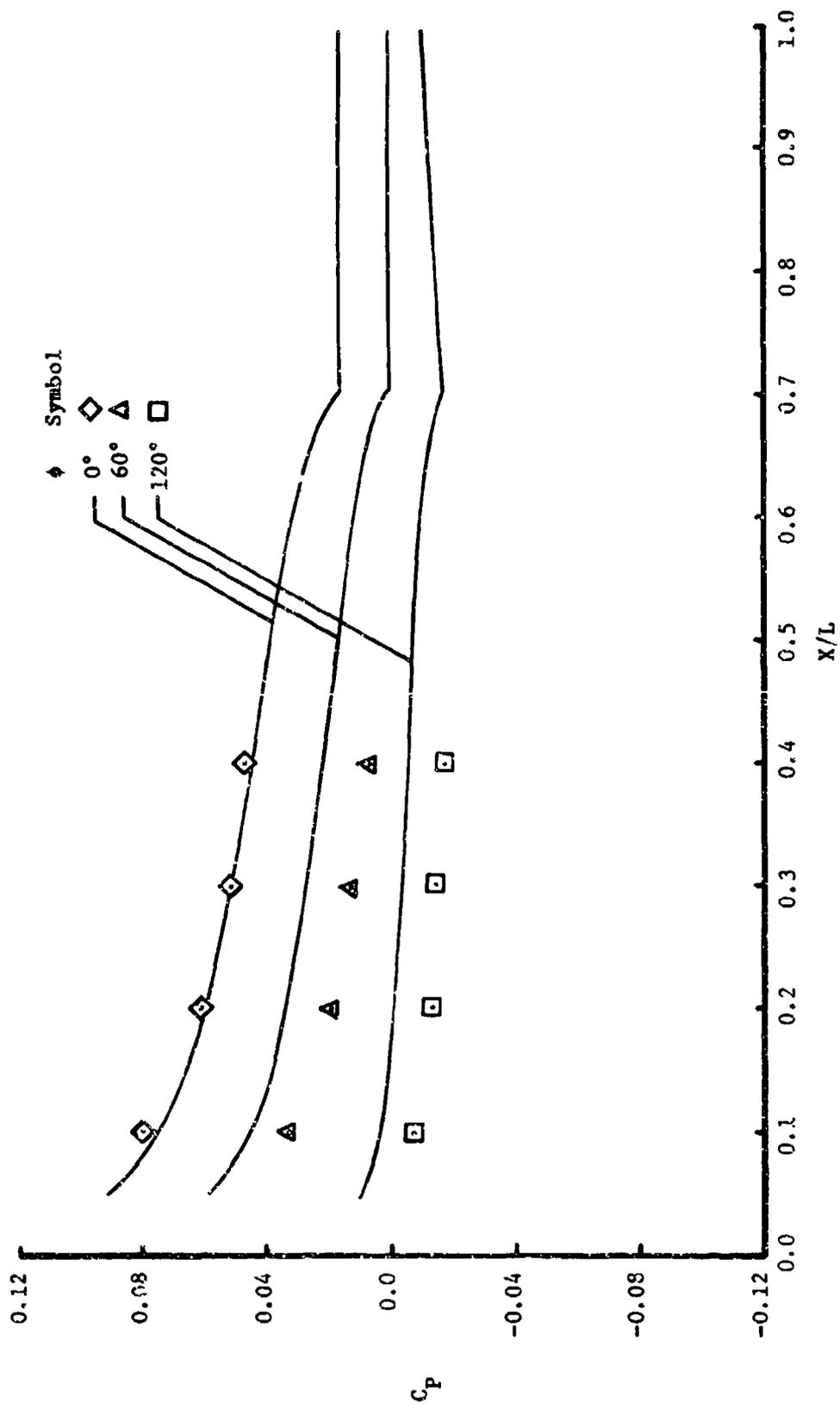


Figure 15. Comparison of Experimental Pressure Data with Theory on the Body; $M_{\infty} = 4.63$, $\alpha = 6.2^\circ$

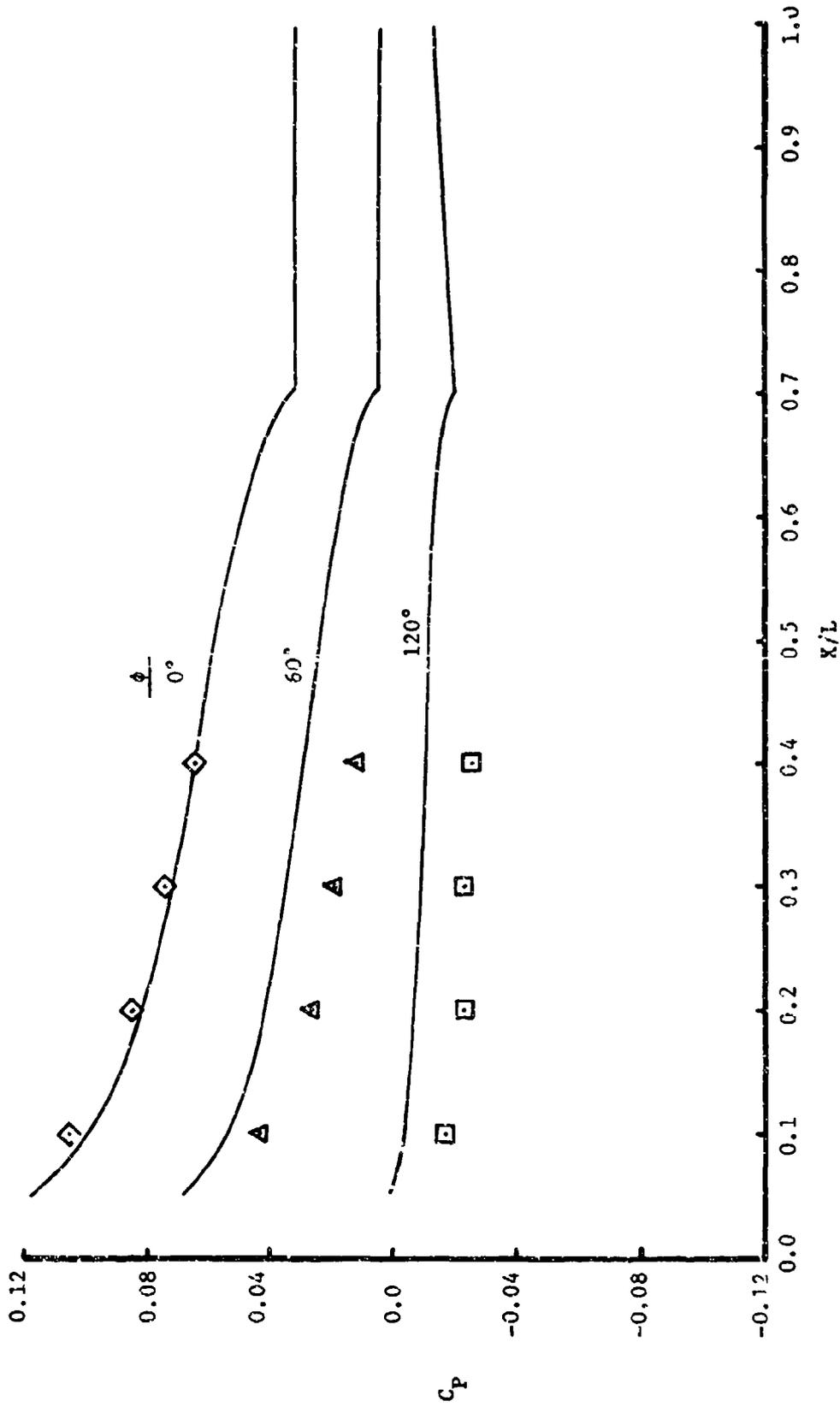


Figure 16. Comparison of Experimental Pressure Data with Theory on the Body; $M_\infty = 4.63$, $\alpha = 8.3^\circ$

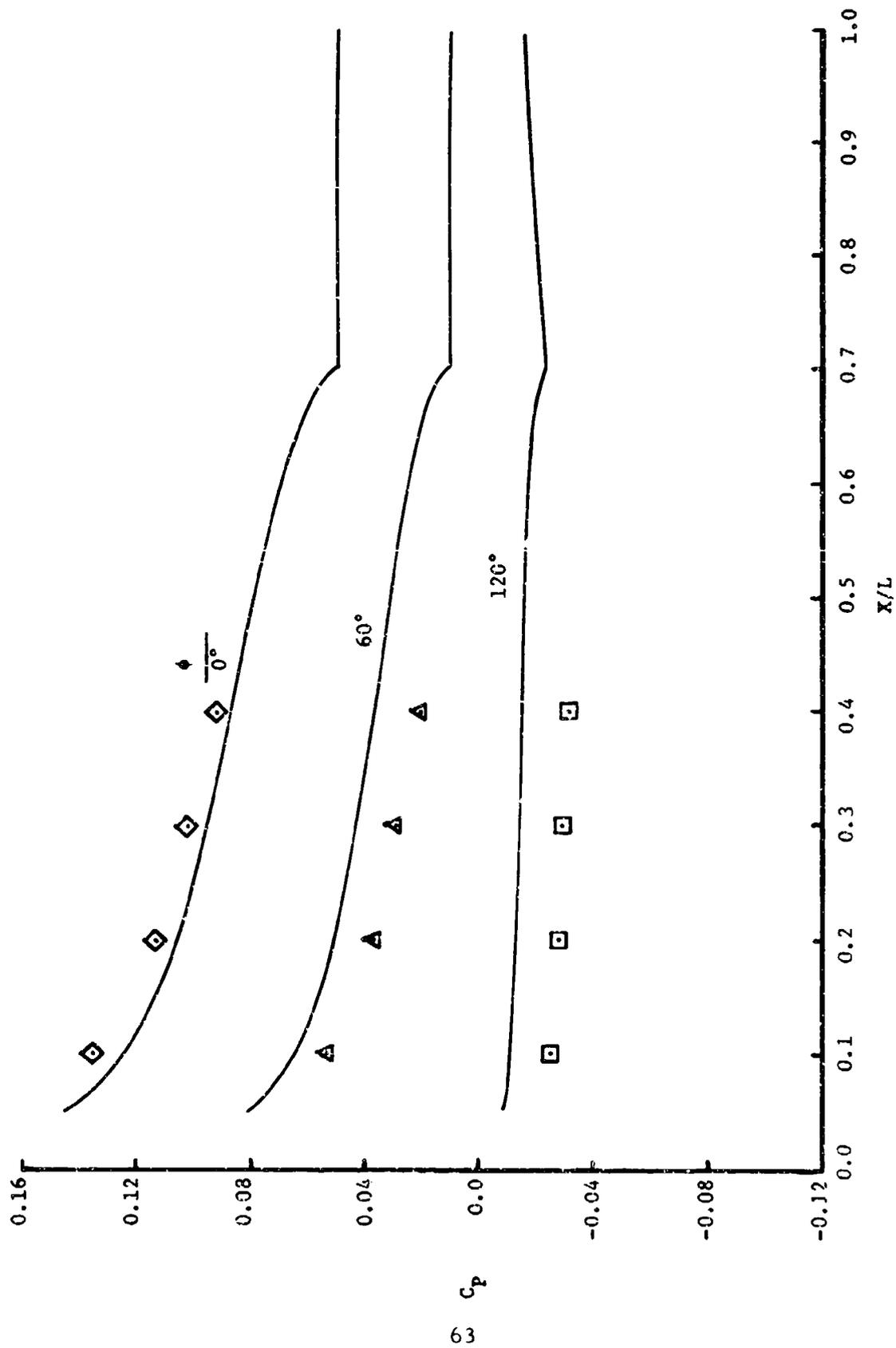


Figure 17. Comparison of Experimental Pressure Data with Theory on the Body; $M_\infty = 4.63$, $\alpha = 10.4^\circ$

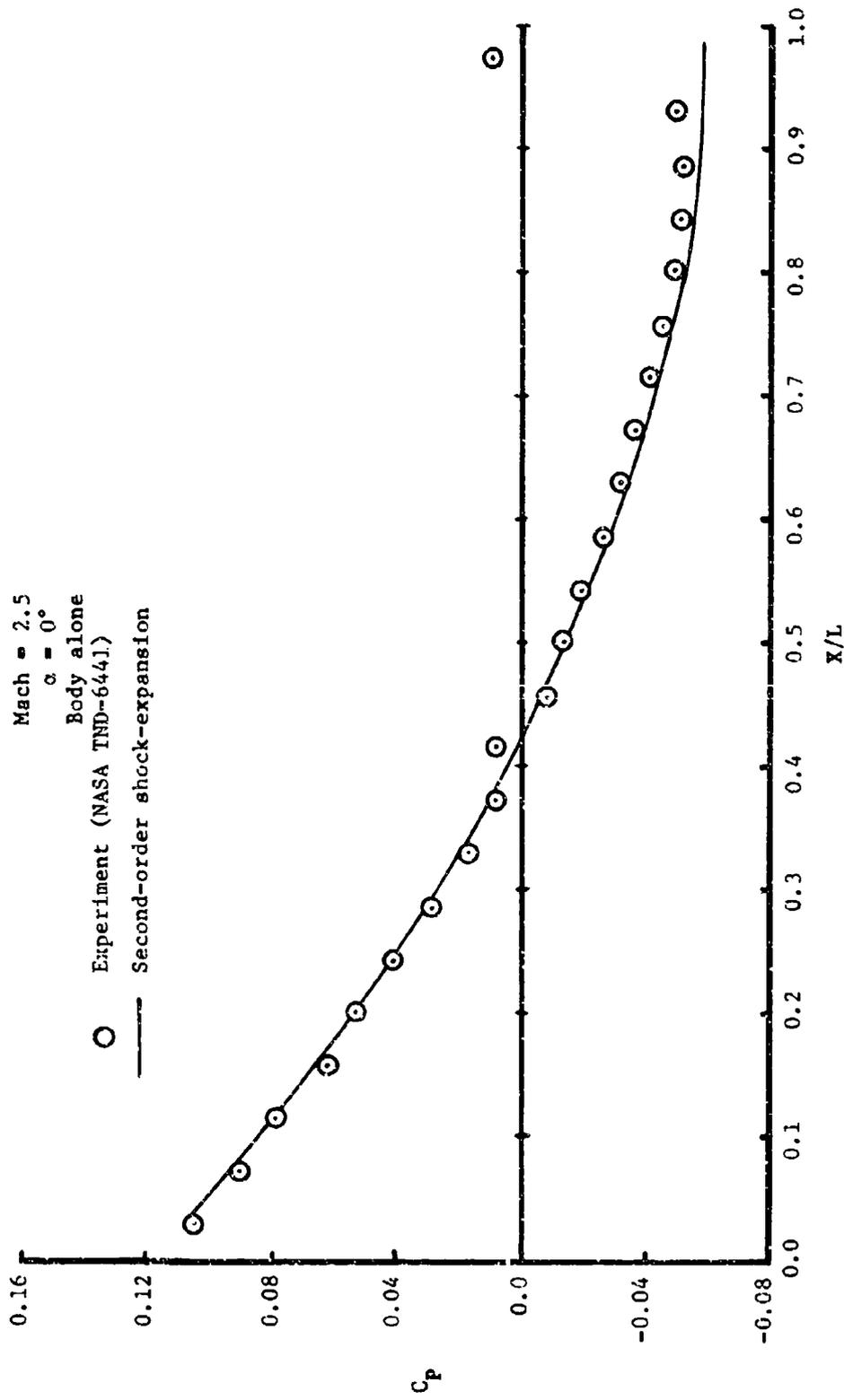


Figure 18. Body Pressure Distribution Comparison; $M_\infty = 2.5$, $\alpha = 0^\circ$

Mach = 2.5
 $\alpha = 2^\circ$
 Body alone

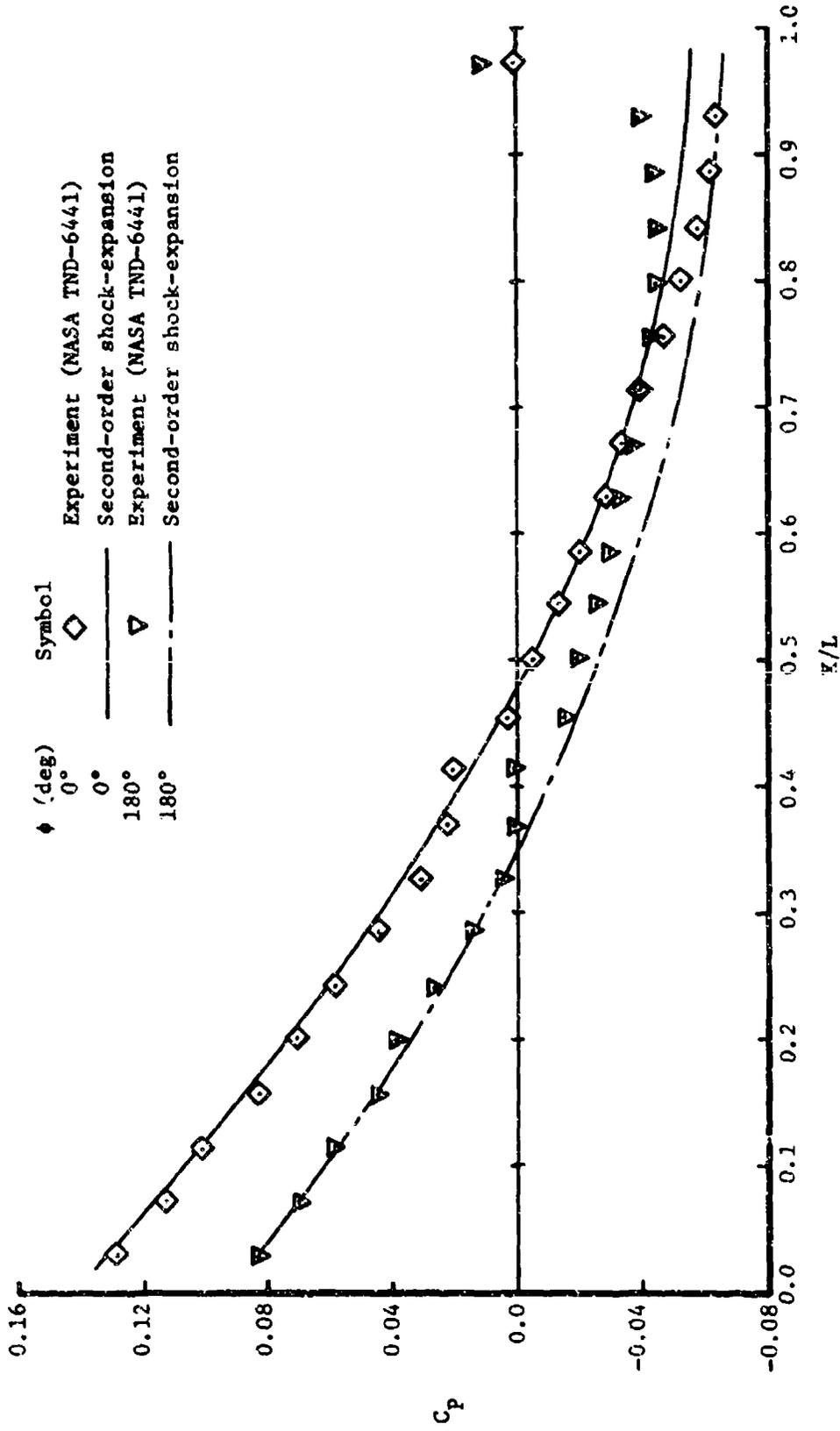


Figure 19. Body Pressure Distribution Comparison; $M_\infty = 2.5$, $\alpha = 2^\circ$

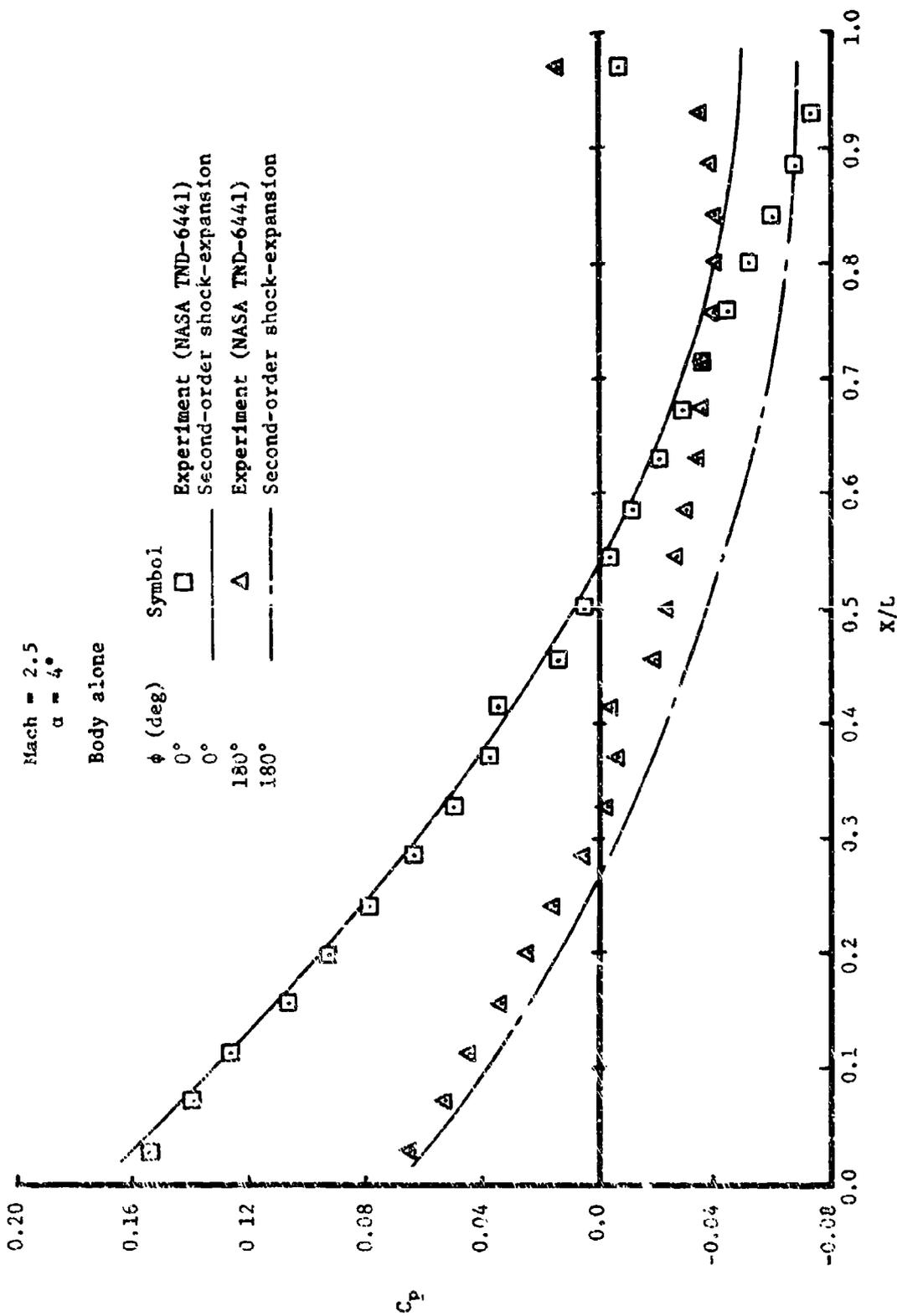


Figure 20. Body Pressure Distribution Comparison; $M_\infty = 2.5$, $\alpha = 4^\circ$

The shock normal Mach number, M_{ns} , as given by the tangent-cone impact method is

$$M_{ns} = K_c M_p \sin \delta_i + \text{EXP}(-K_c M_p \sin \delta_i)$$

where

M_p is the Mach number in a particular ϕ -plane,

δ_i is the impact angle M_p makes with the surface,

and

$$K_c = 2(\gamma + 1)/(\gamma + 3)$$

The pressure ratio across the shock wave, P_s , is then calculated as

$$P_s = P_{si} \left(\frac{P_s}{P_{si}} \right)_{\alpha=0}$$

where P_{si} is the impact pressure ratio across the shock

$$= \left[2\gamma M_{ns}^2 - (\gamma + 1) \right] / (\gamma + 1),$$

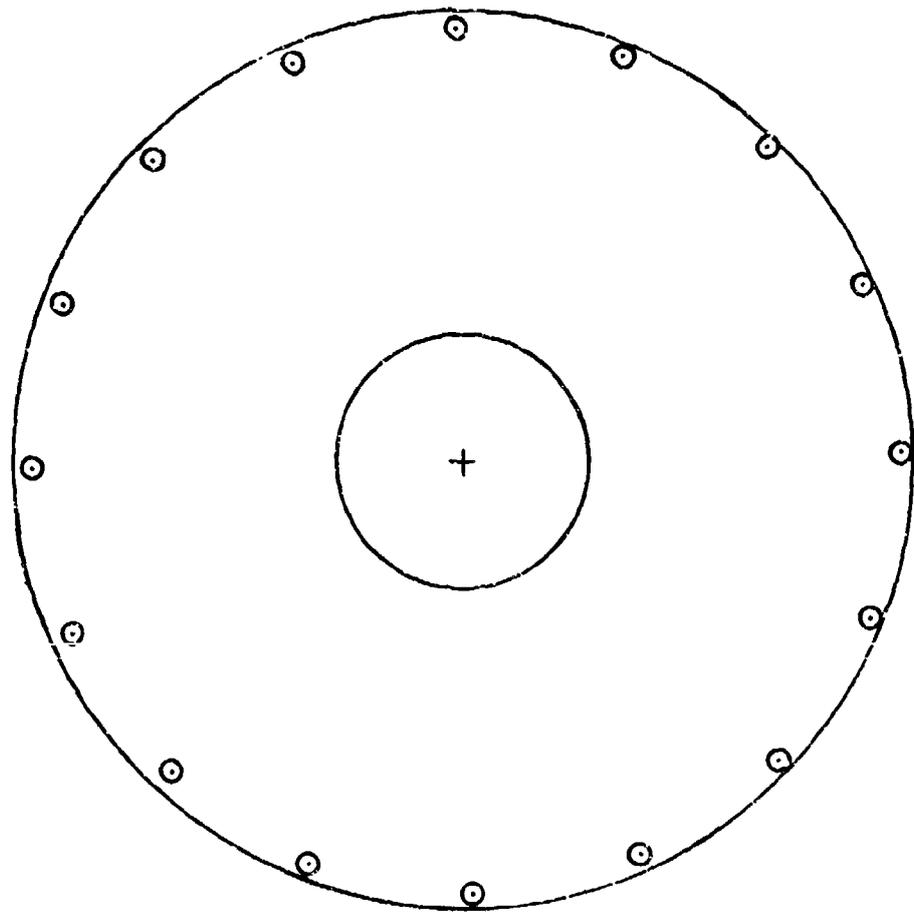
$(P_s)_{\alpha=0}$ is obtained from the zero angle of attack cone results

and

$(P_{si})_{\alpha=0}$ is the impact pressure ratio across the shock at zero angle of attack

Calculations using this method have been compared with the exact solutions tabulated in Reference 20. The cases selected were for a 10-degree semi-apex cone at angles of attack of 0, 5, 10, and 11 degrees. Results for freestream Mach numbers equal to 2, 5, and 10 are shown in Figures 21, 22, and 23, respectively. The method has also been compared with the experimental data given in Reference 13. These data are for Mach = 5.05 at angles of attack of 0, 5, 10, and 15 degrees. The results for a cone semi-apex angle of 11.42 degrees are presented in Figure 24 and for 18.92 degrees in Figure 25.

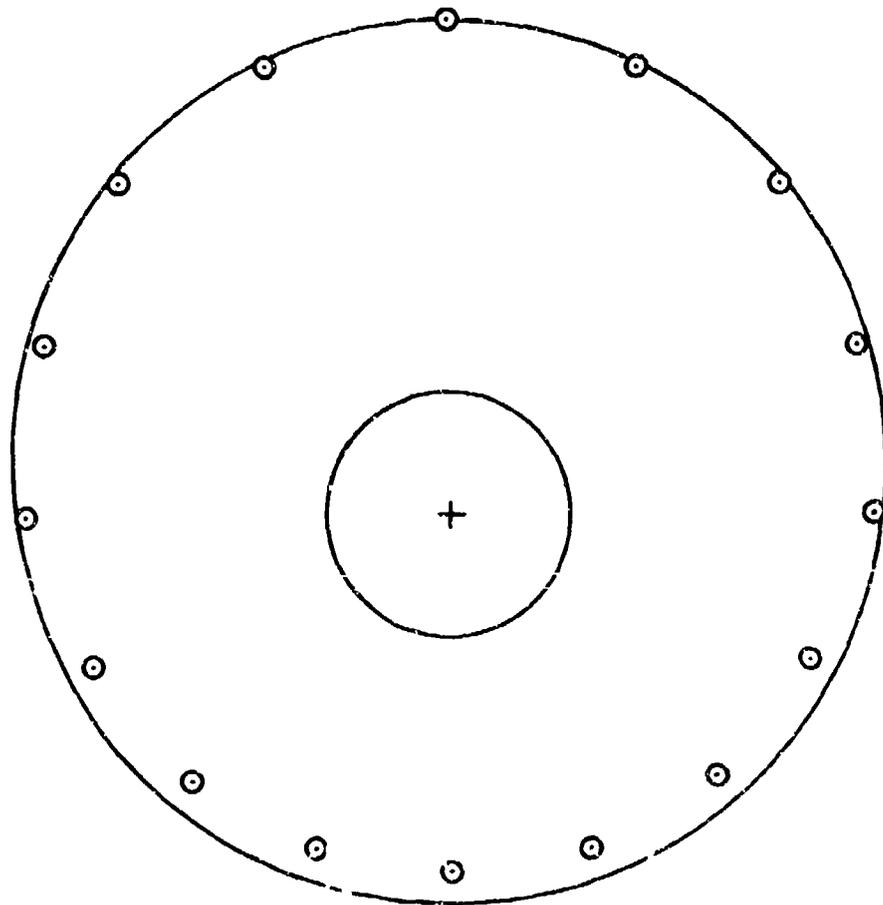
In summarizing the data comparison presented, the method devised for calculating shock shapes does closely follow the exact and experimental results. In view of the approximate solution used, the results are in fact remarkable. Noteworthy in this respect are the 10-degree cone results at Mach = 5.0 and the 11.42-degree cone results at Mach = 5.05.



○ Cone tables (AGARDOGRAPH 137)
— Mark IV

(a) $\alpha = 0^\circ$

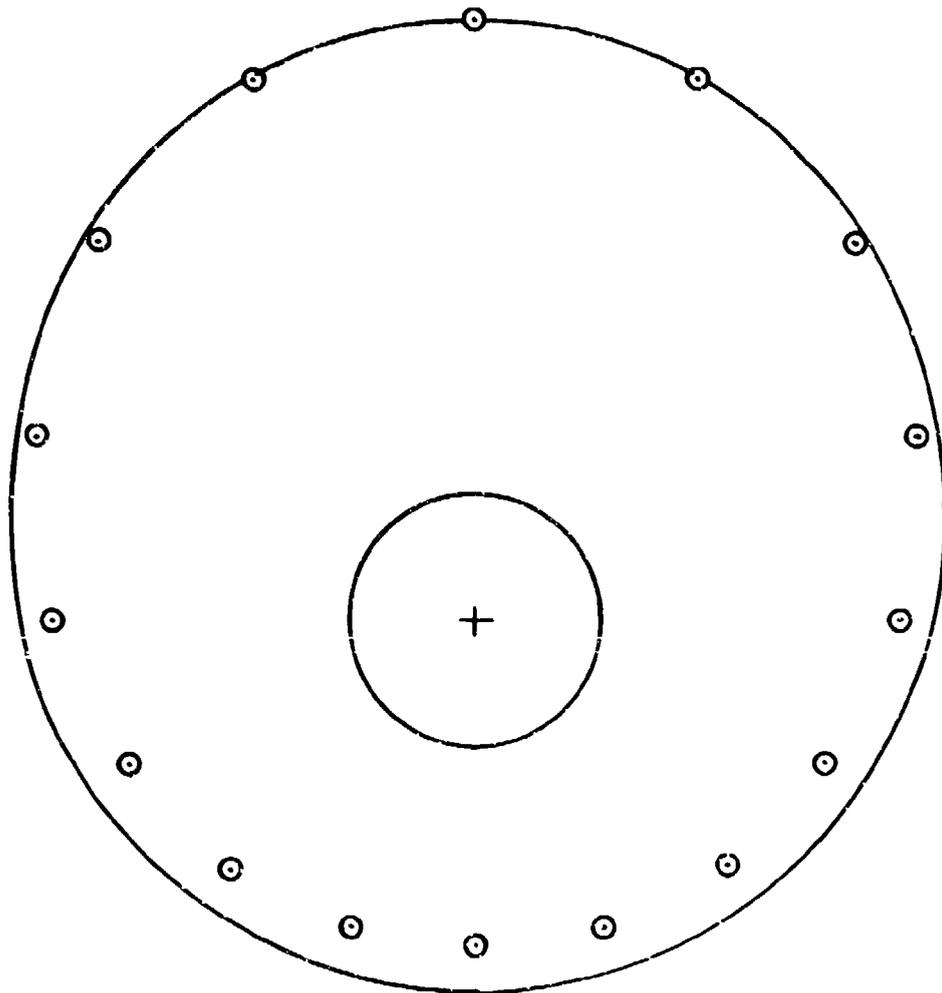
Figure 21. Shock Wave Shape Comparison for a Cone; $M_\infty = 2.0$, $\theta_c = 10^\circ$



○ Cone tables (AGARDOGRAPH 137)
— Mark IV

(b) $\alpha = 5^\circ$

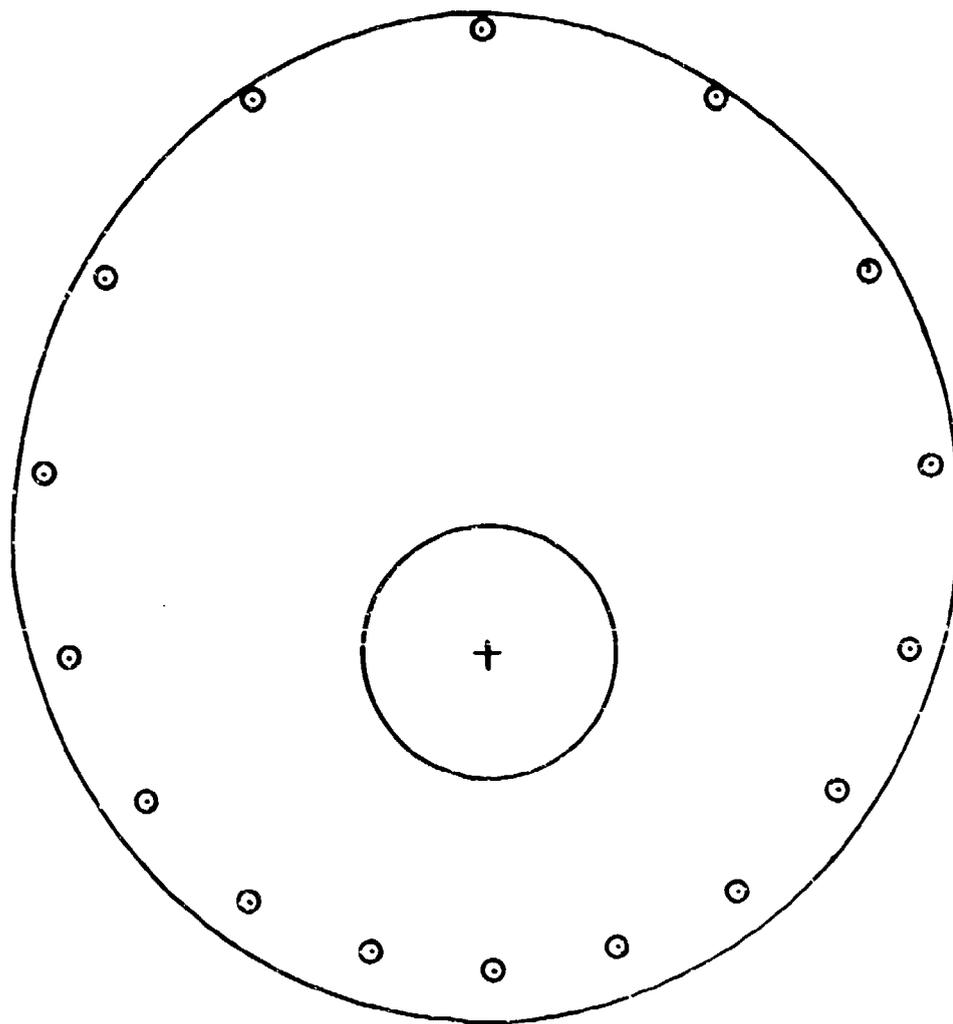
Figure 21. - Continued



○ Cone tables (AGARDOGRAPH 137)
— Mark IV

(c) $\alpha = 10^\circ$

Figure 21. - Continued



○ Cone tables (AGARDOGRAPH 137)

— Mark IV

(d) $\alpha = 11^\circ$

Figure 21. - Concluded

○ Cone tables (AGARDOGRAPH 137)

— Mark IV

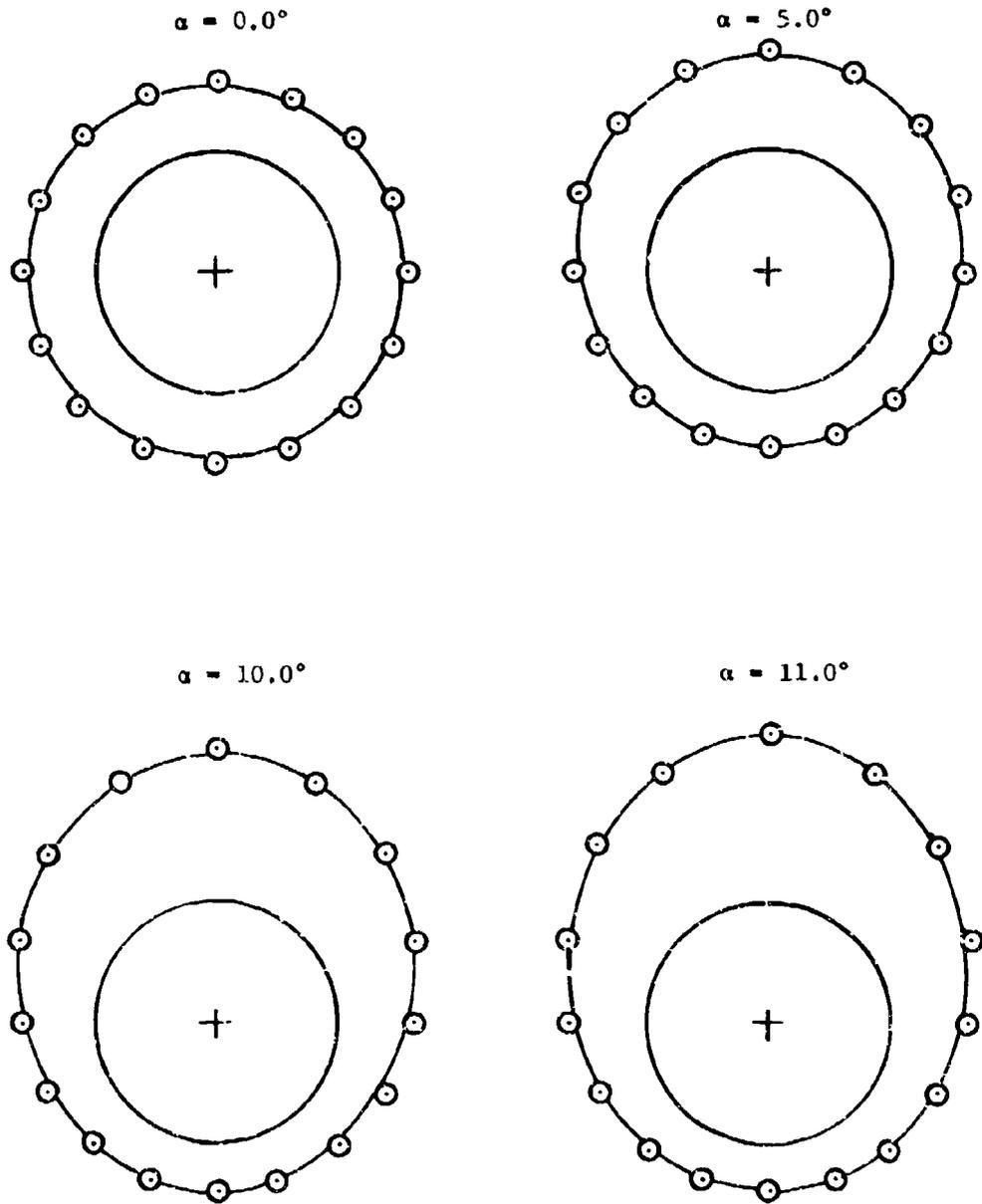


Figure 22. Shock Wave Shape Comparison for a Cone; $M_\infty = 5.0$, $\theta_c = 10^\circ$

○ Cone tables (AGARDOGRAPH 137)
— Mark IV

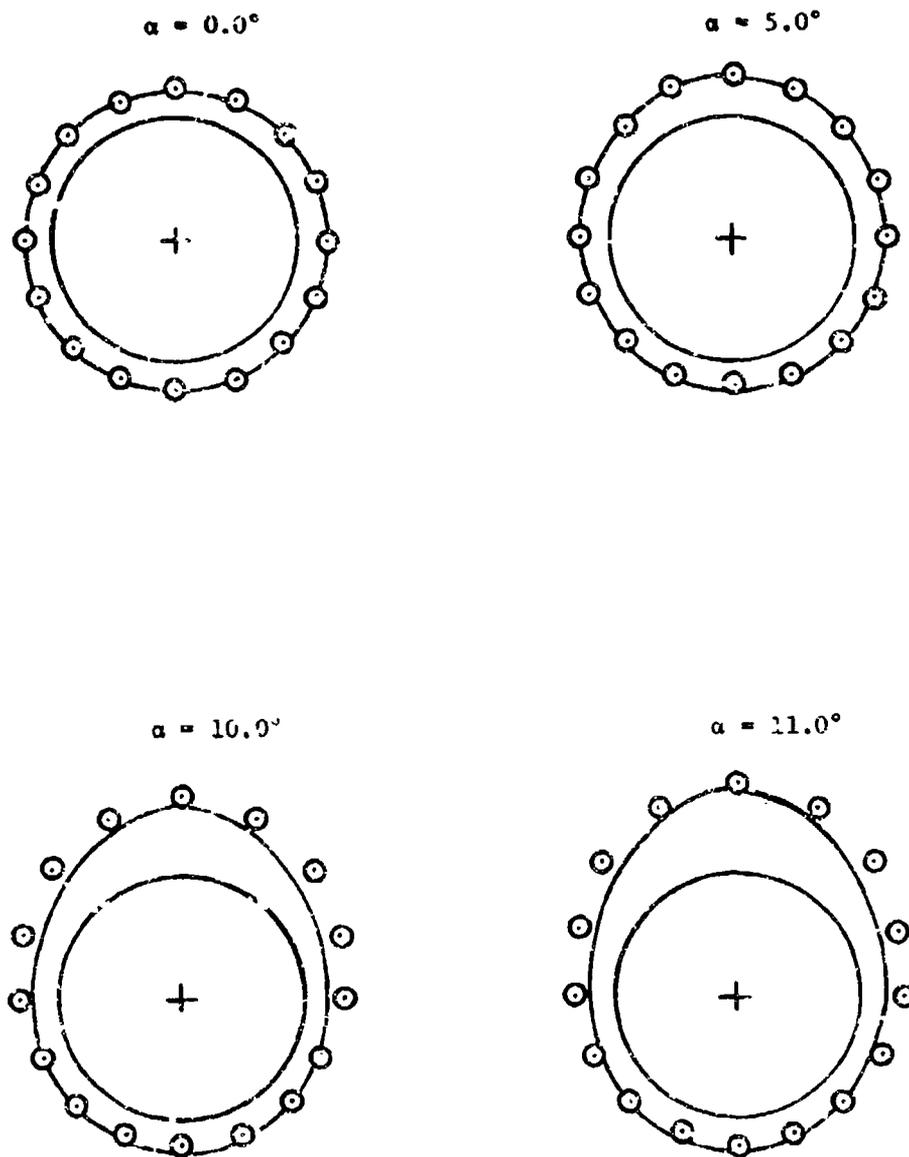


Figure 23. Shock Wave Shape Comparison for
a Cone; $M_\infty = 10$, $\theta_c = 10^\circ$

- Experiment (NACA TN-3349)
- Mark IV
- - - - Generalized shock expansion ($\alpha = 15^\circ$ only)
(NACA TN-3349)

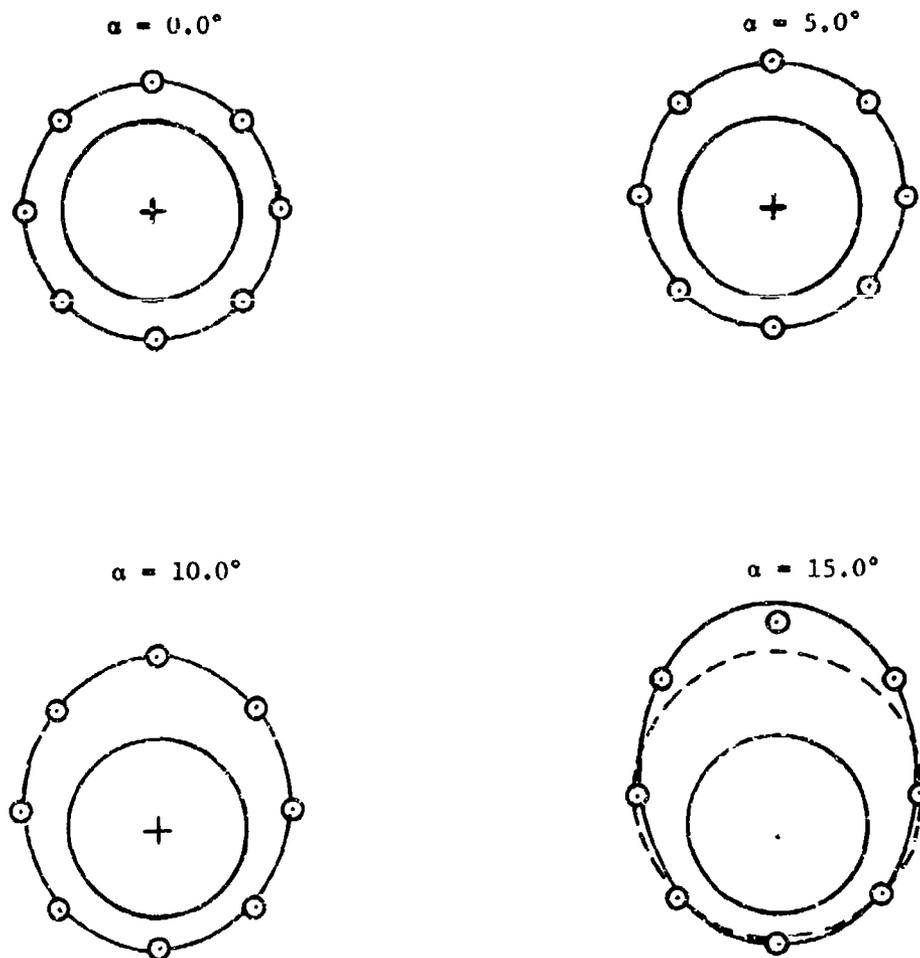


Figure 24. Shock Wave Shape Comparison for
a Cone; $M_\infty = 5.05$, $\theta_c = 11.42^\circ$

- ⊙ Experiment (NACA TN-3349)
- Mark IV
- - - Generalized shock expansion ($\alpha = 15^\circ$ only)
(NACA TN-3349)

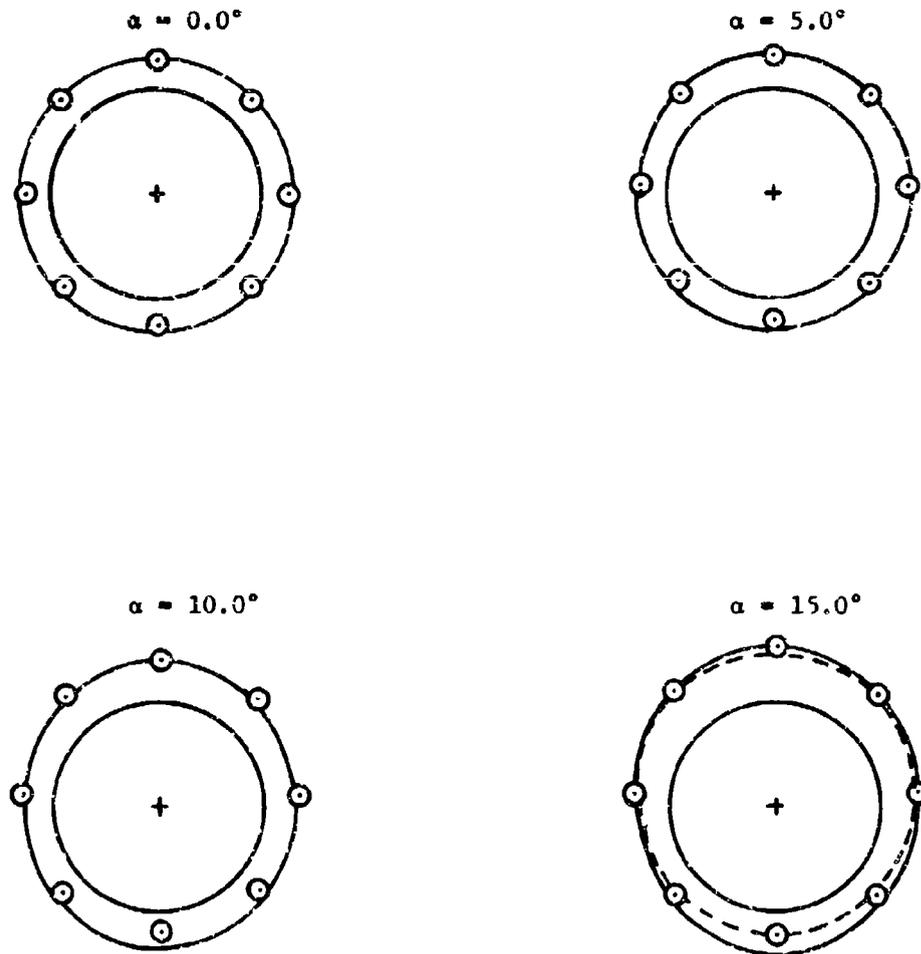


Figure 25. Shock Wave Shape Comparison for a Cone; $M_\infty = 5.05$, $\theta_c = 18.92^\circ$

A basic condition in constructing the flow field about a two-dimensional body using the shock expansion method is that the pressure is constant along Mach lines emanating from the surface. In the case of flow about a three-dimensional body, this condition is modified to account for the conical flow at the nose. In the conical region there is a pressure difference, ΔP_n , between the surface and the shock wave

$$\Delta P_n = P_{cn} - P_{sn}$$

where P_{cn} is the cone surface pressure ratio at the nose

and P_{sn} is the cone shock pressure ratio at the nose

It was suggested in Reference 12 that this ΔP be used to represent the net pressure change between the body surface and the shock along each Mach line emanating from the surface downstream of the nose.

That is,

$$P_s = P_b - \Delta P_n$$

where P_s is the shock pressure ratio

and P_b is the body pressure ratio

It has been found that this expression permits too fast a decay in the shock pressure. To compensate for this, a damping factor, f , is introduced;

$$P_s = P_b - \Delta P_n \cdot f$$

The form used for f is simply the ratio of the local surface deflection angle to the nose cone angle and the value of P_s is limited to 1.0 as a minimum value.

Comparison of the shock shape calculated by this procedure and the method of characteristics is shown in Figure 26 for the body of NASA TN D-6480. Calculations are also compared with the experimental data of Reference 13 in Figure 27. The body is a fineness ratio 3 ogive and test conditions are 10-degrees angle of attack at Mach = 5.05. The results are very good at the nose but tend to deteriorate downstream. This points out an additional problem which will be encountered in calculating shock shapes on bodies. Namely, the errors are accumulative. Thus it will be difficult to accurately predict the extent or breadth of the shock field. This is compensated for somewhat in that the pressure progressively weakens downstream.

$M_\infty = 2.3$ $\alpha = 0^\circ$
NASA TND-6480 body shape

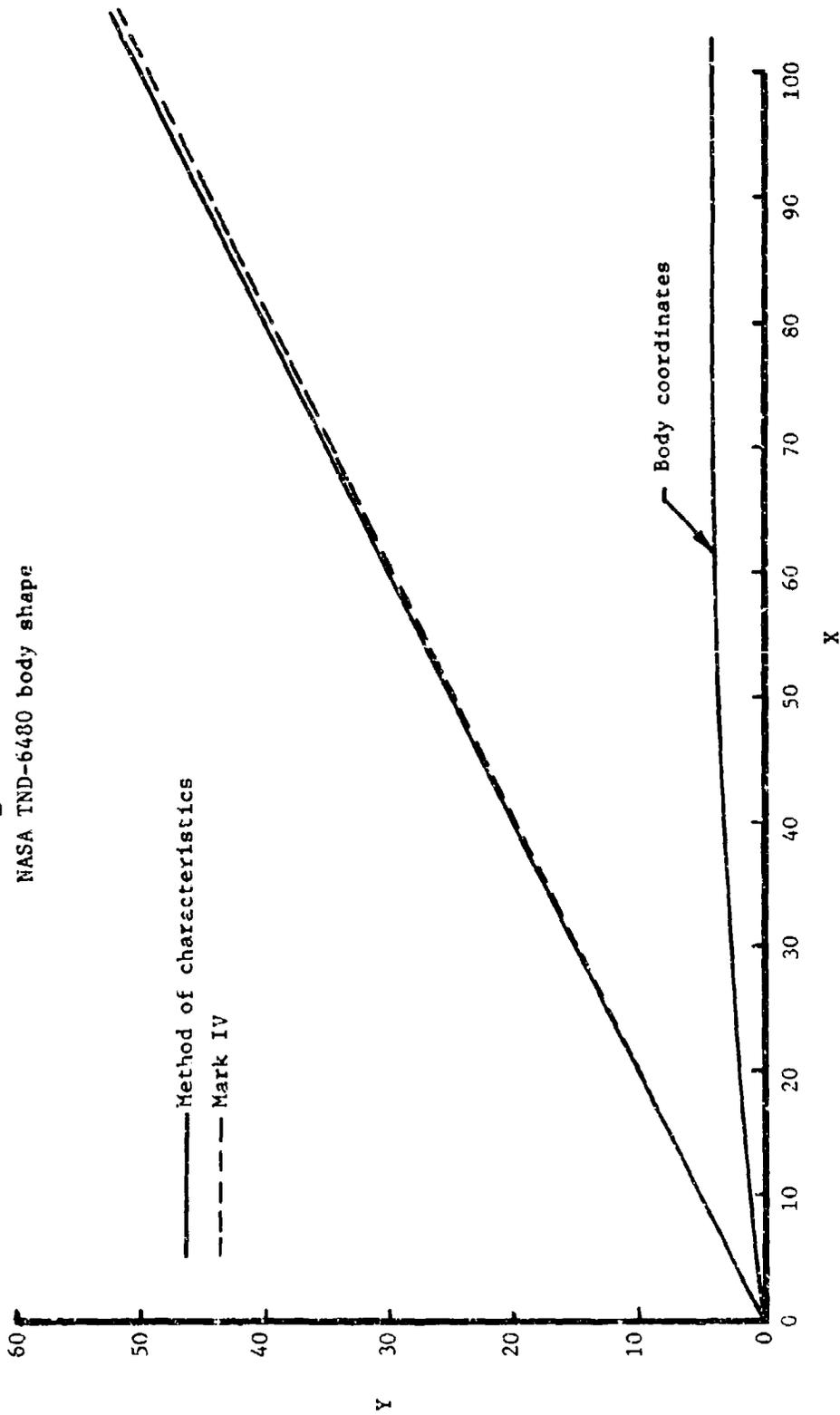


Figure 26. Shock Wave Shape Comparison for NASA TND-6480 Body

$M_\infty = 5.05$ $\alpha = 10^\circ$
 Ogive nose angle = 18.92°

○ Experiment (NACA TN-3349)
 — Mark 1V

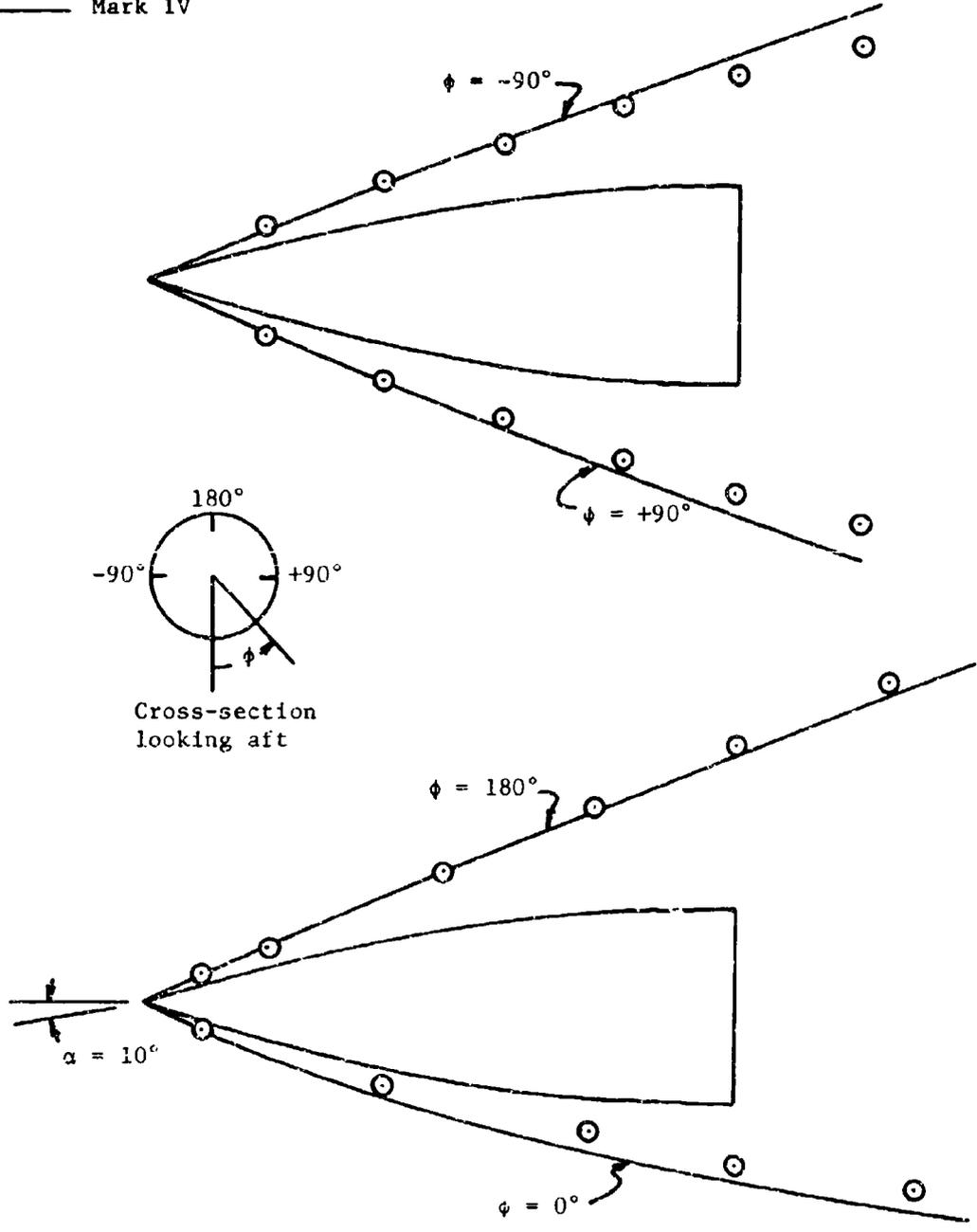


Figure 27. Shock Wave Shape Comparison for Fineness Ratio 3 Ogive; $M_\infty = 5.05$, $\alpha = 10^\circ$

Flow Field Calculation

The complete shock expansion flow field analysis method is demonstrated on the configuration of NASA TND-6480. The methods for surface pressure, shock wave shape, and the Surface Spline interpolation are combined to calculate the wing pressure distribution subject to the body flow field. The Mach = 4.63 data at zero angle of attack are used for comparison. While this angle of attack is not the most representative from the standpoint of force calculations, it permits parallel calculations using the method of characteristics to further assess the method.

The configuration planform and flow region are shown in Figure 28. Figures 29, 30, and 31 present the Mach number, pressure ratio, and flow angle, respectively, behind the body shock wave. Figure 32 gives the Mach number along the body surface. (The pressure distribution on the body was previously presented in Figure 10). The shock wave and the body comprise the two boundary curves used in the Surface Spline. The locations at which the data were stored on the flow field unit (10) are indicated on the x-scales of the plots. Figure 33 shows the location of a right-running characteristics along which the flow was interrogated. Results of the Surface Spline interpolation for local Mach number, pressure ratio, and flow angle are compared with the method of characteristics calculation in Figures 34, 35, and 36, respectively. The differences due to the different boundary conditions at the shock wave are to be expected. The general character of the flow is fairly well maintained. Most remarkable in this respect is the β curve (Figure 36). It should be reemphasized that only values along the boundary curves have been used - that is along curves through the points labeled "shock" and "body".

Finally, the pressure distributions on the wing at four span locations are shown in Figure 37 and 38. Both flow fields have been used in conjunction with the tangent-wedge and tangent-cone pressure methods. These two pressure methods were also run without the body flow and all are compared to the experimental data. A first observation of the figures show the tangent-wedge method more appropriate for the condition run. A second and more subtle observation is that the body flow field causes a concavity to the pressure distribution on the forward half of the wing. This is best seen on the inboard station ($y/(b/2) = 0.258$) and is consistent with the experimental data. Thirdly, the outboard station ($y/(b/2) = 0.815$) clearly shows the effect of the body shock crossing the wing section (at $\sim x/c = 0.25$).

$M_\infty = 4.63$ $\alpha = 0^\circ$ $\beta = 0^\circ$

MASA TND-6430 configuration

Method of characteristics

Second-order shock-expansion

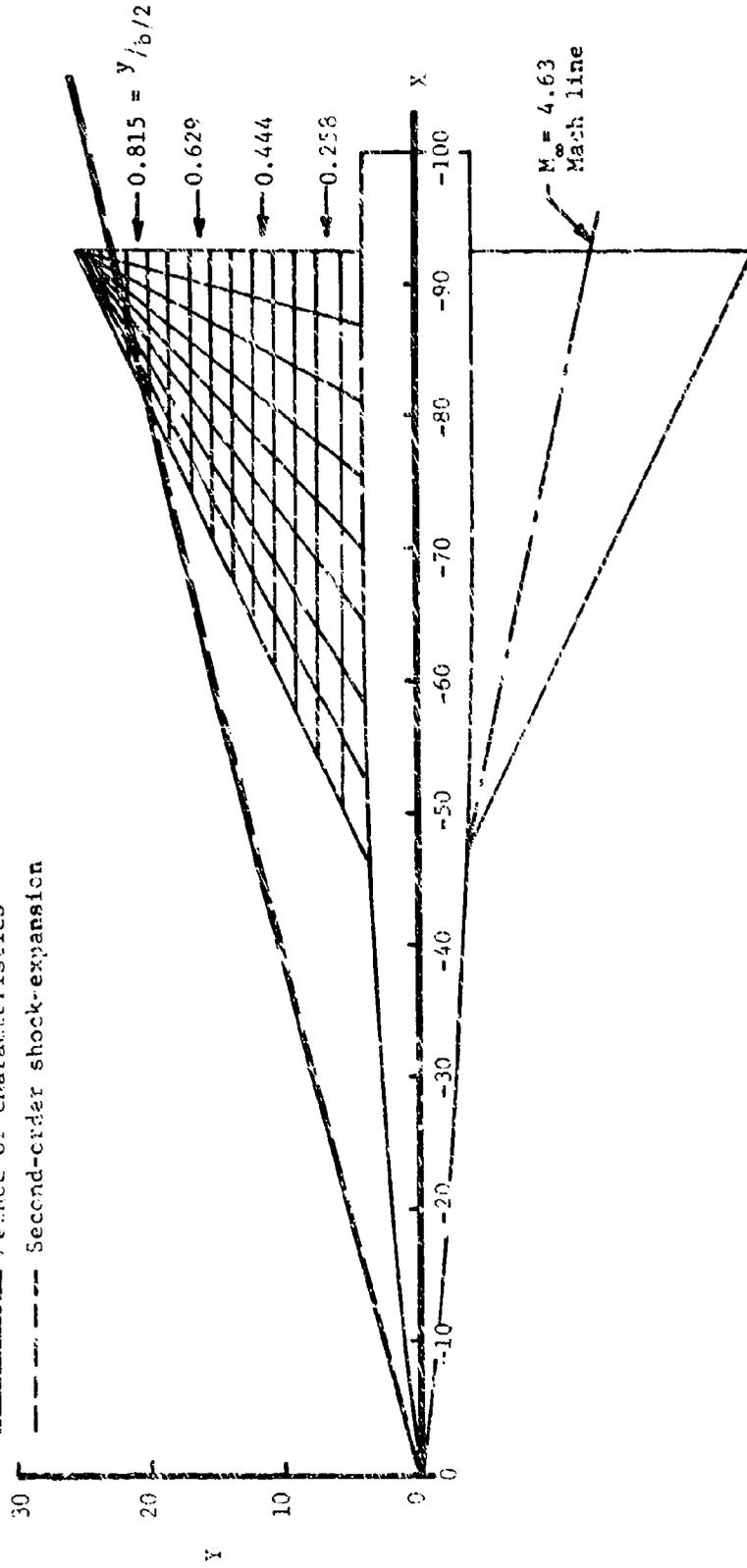


Figure 28. Shock Wave Shape Comparison and Body Flowfield Interference Region on Wing

$M_\infty = 4.63$ $\alpha = 0^\circ$ $\delta = 0^\circ$

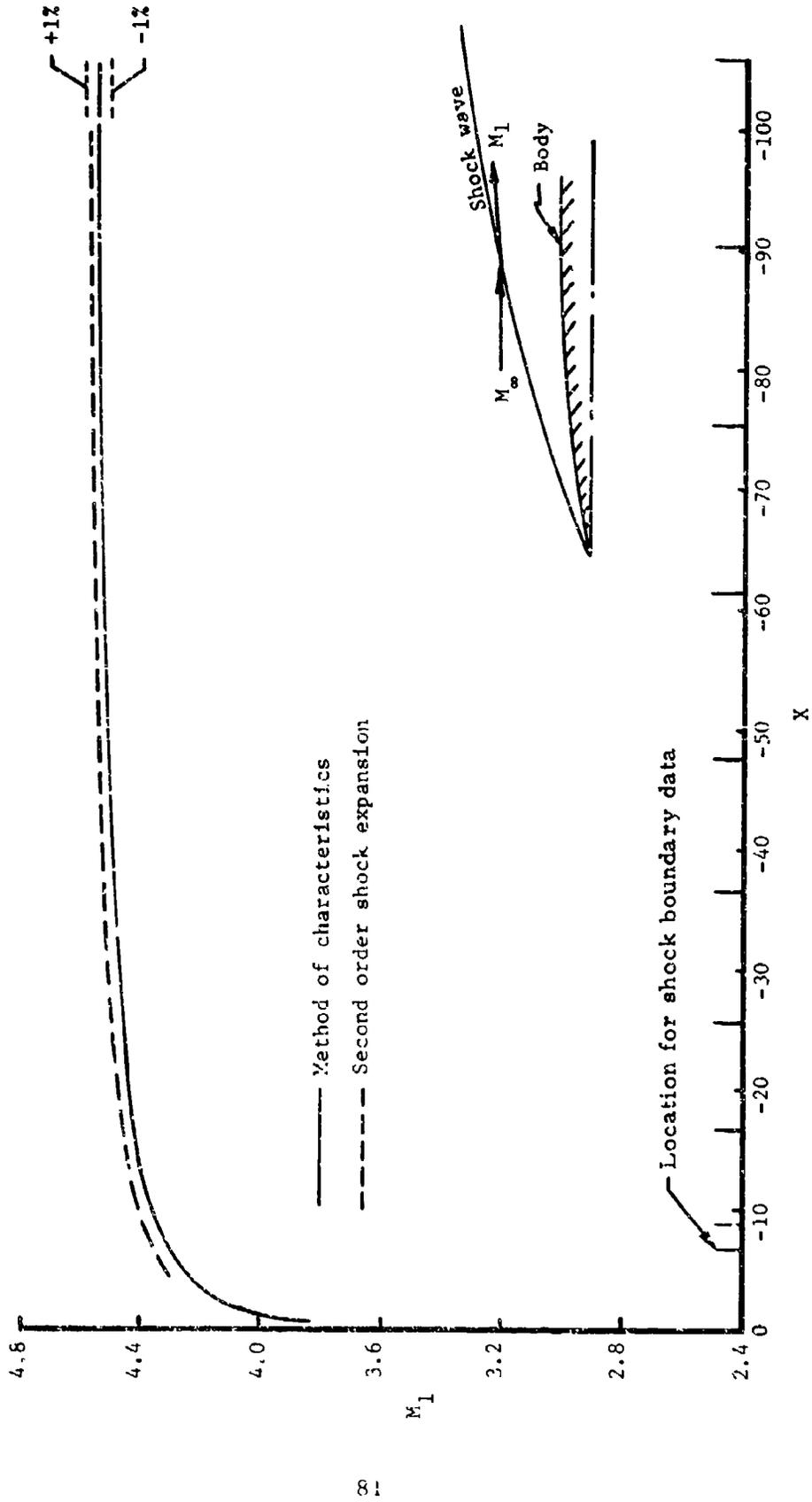


Figure 29. Comparison of Mach Number Behind Body Generated Shock Wave

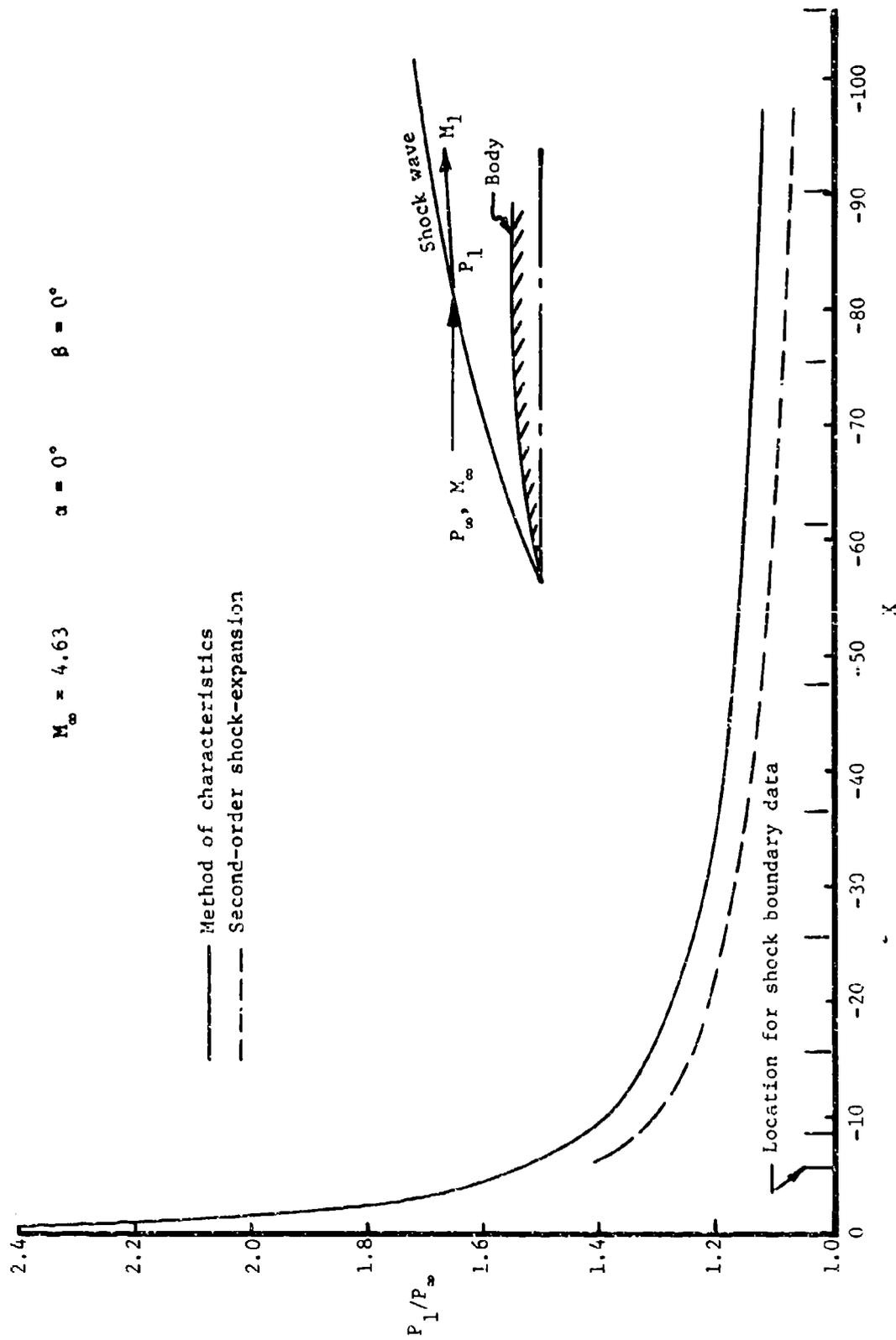


Figure 30. Comparison of Pressure Ratio Behind Body Generated Shock Wave

$M_\infty = 4.63$ $\alpha = 0^\circ$ $\beta = 0^\circ$

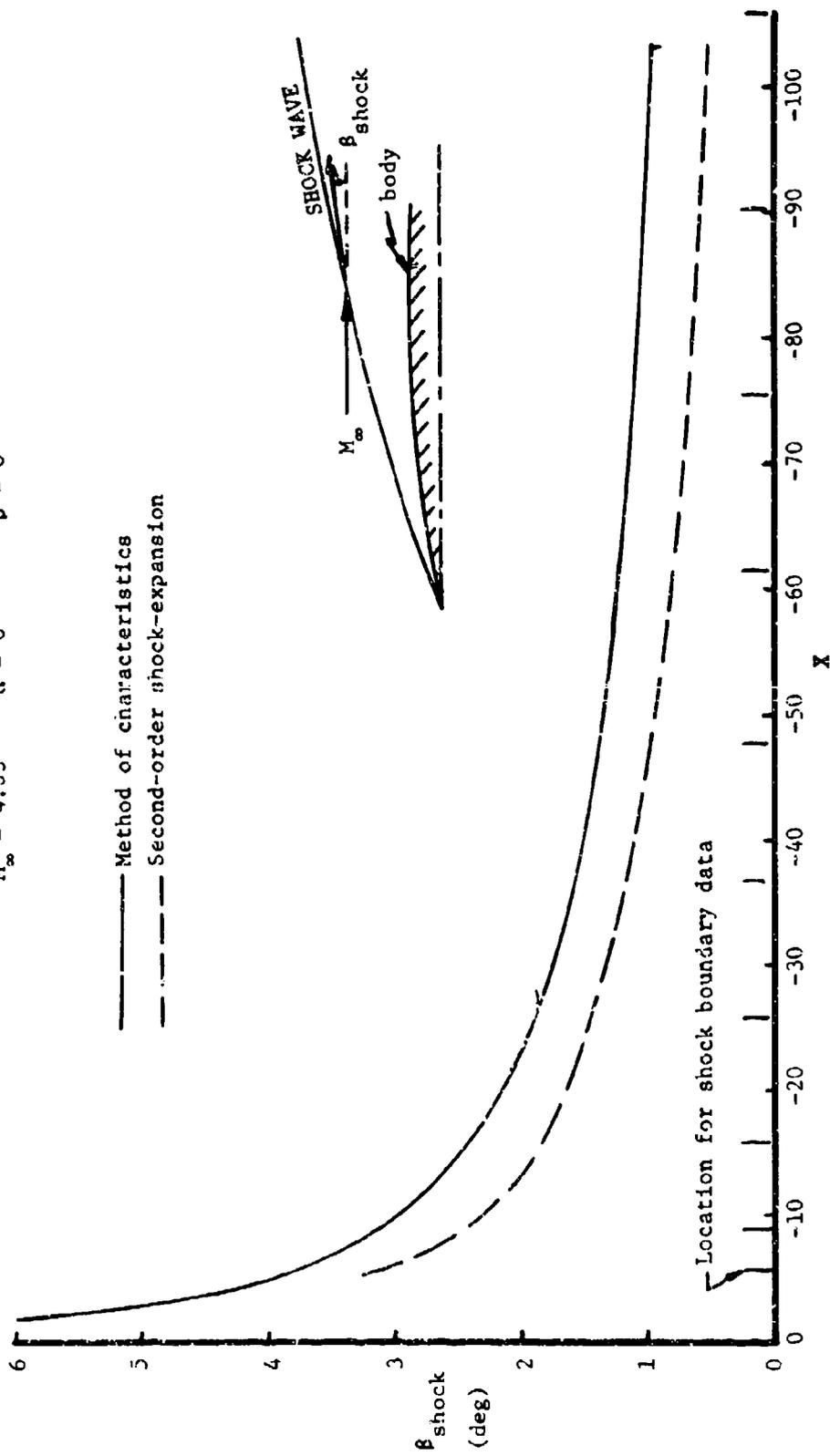


Figure 31. Comparison of Flow Angle Behind Body Generated Shock Wave

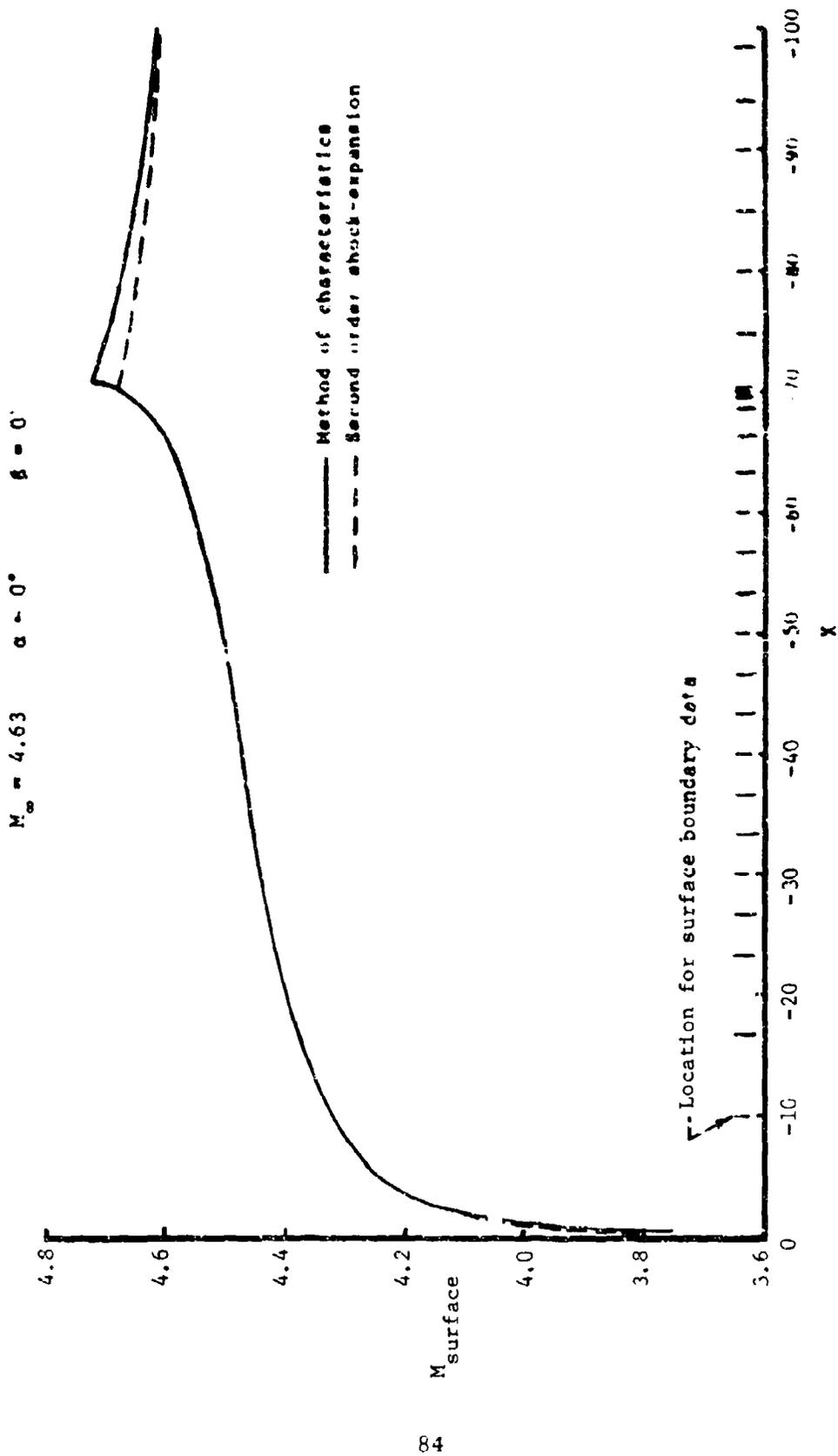


Figure 32. Comparison of Surface Mach Number

$M_\infty = 4.63$ $\alpha = 0^\circ$ $\beta = 0^\circ$

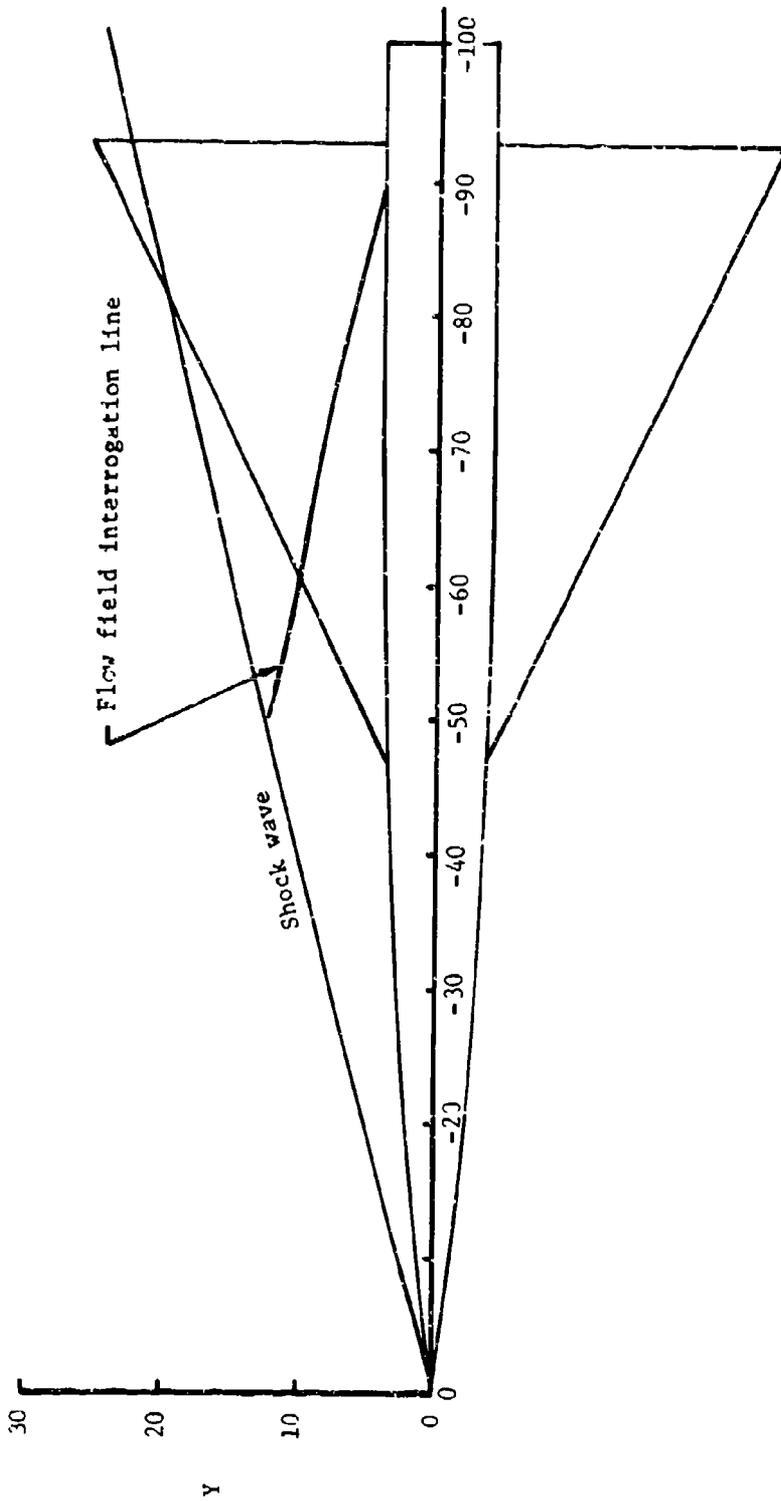


Figure 33. Flow Field Interrogation Line

$M_\infty = 4.63$ $\alpha = 0^\circ$ $\beta = 0^\circ$

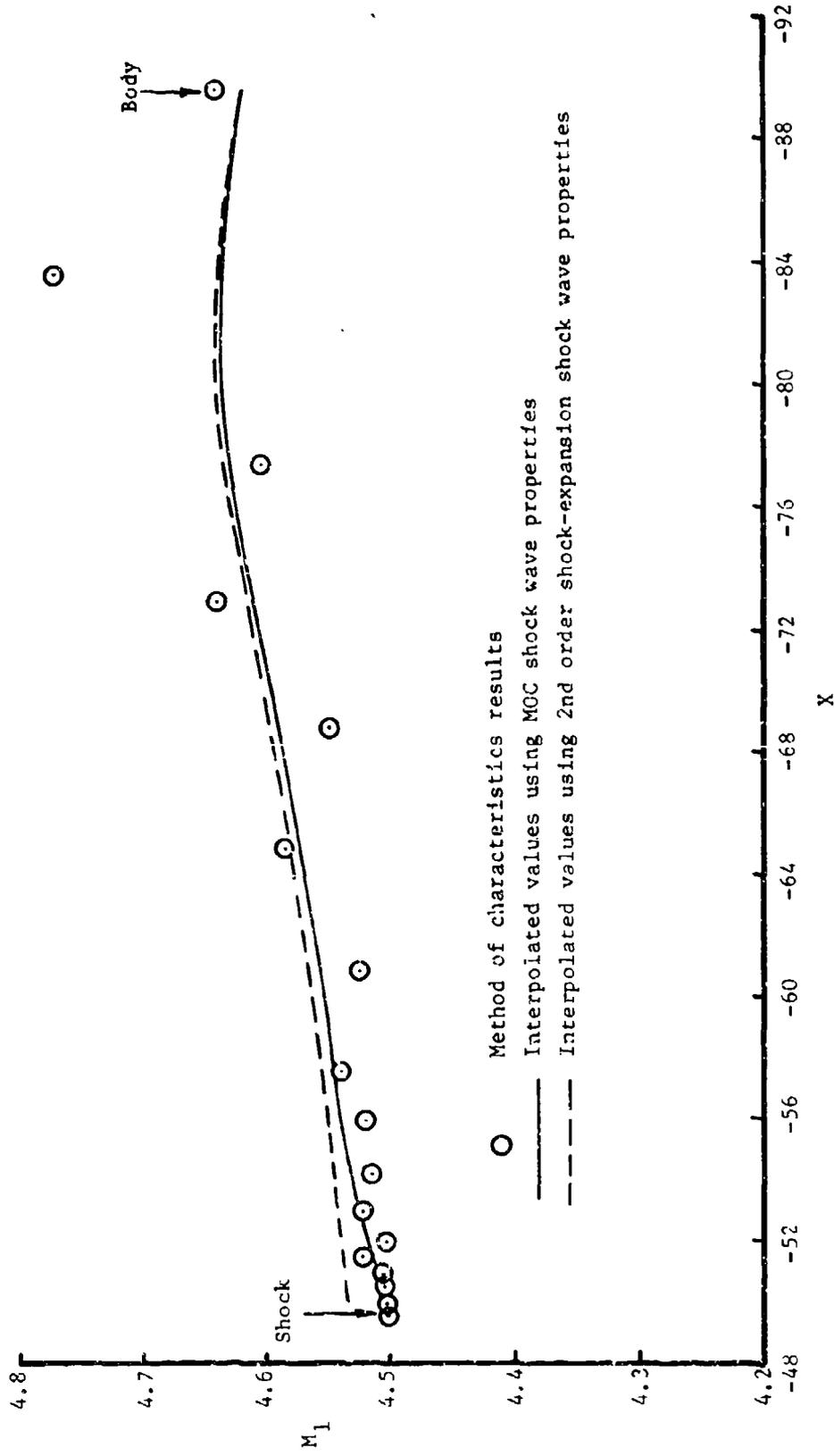


Figure 34. Comparison of Mach Number Along the Flow Field Interrogation Line

$$M_\infty = 4.63 \quad \alpha = 0^\circ \quad \Lambda = 0^\circ$$

- Method of characteristics results
- Interpolated values using MOC shock wave properties
- - - Interpolated values using 2nd order shock-expansion shock wave properties

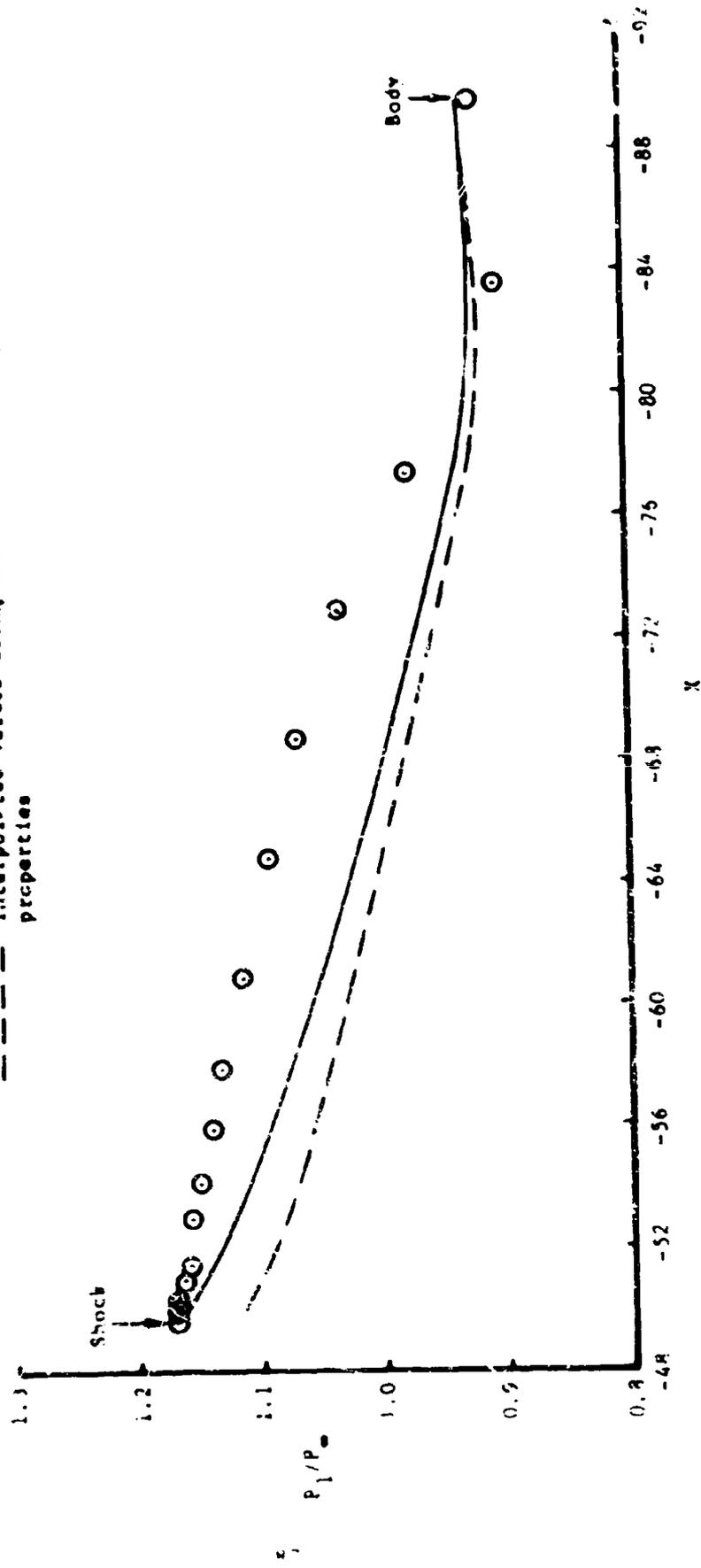


Figure 35. Comparison of Pressure Ratio Along the Flow Interrogation Line

$M_\infty = 4.63$ $\alpha = 0^\circ$ $\beta = 0^\circ$

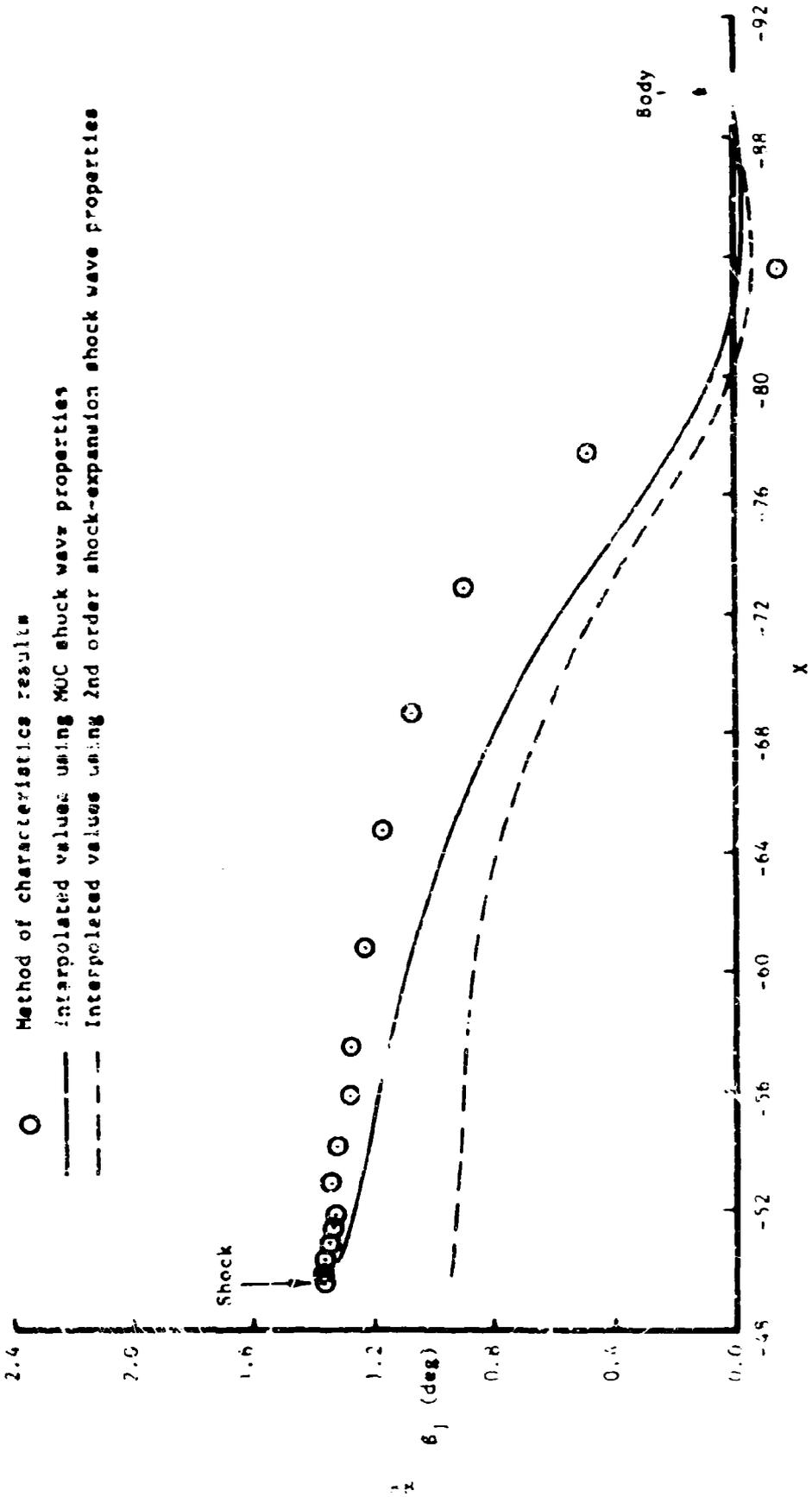


Figure 16. Comparison of Flow Angle Along the Flow Interrogation Line

$M_\infty = 4.63$ $\alpha = 0^\circ$ $\beta = 0^\circ$

— Tangent wedge (no interference)

— Tangent wedge with interference (MOC shock properties)

— Tangent wedge with interference (2nd order shock expansion shock properties)

— Tangent cone with interference (MOC shock properties)

— Tangent cone (no interference)

○ □ Experimental data (NASA TN D-6480)

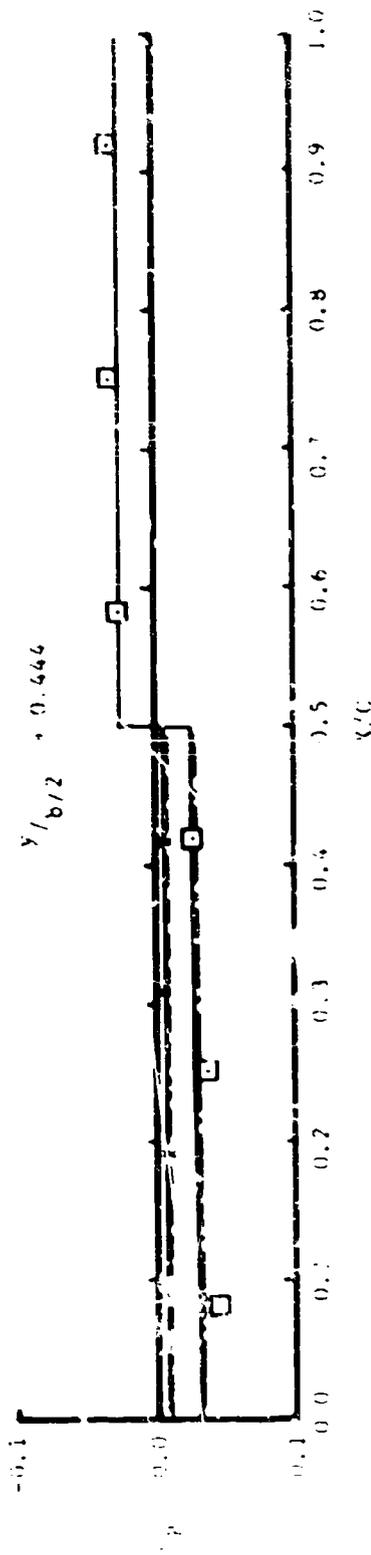
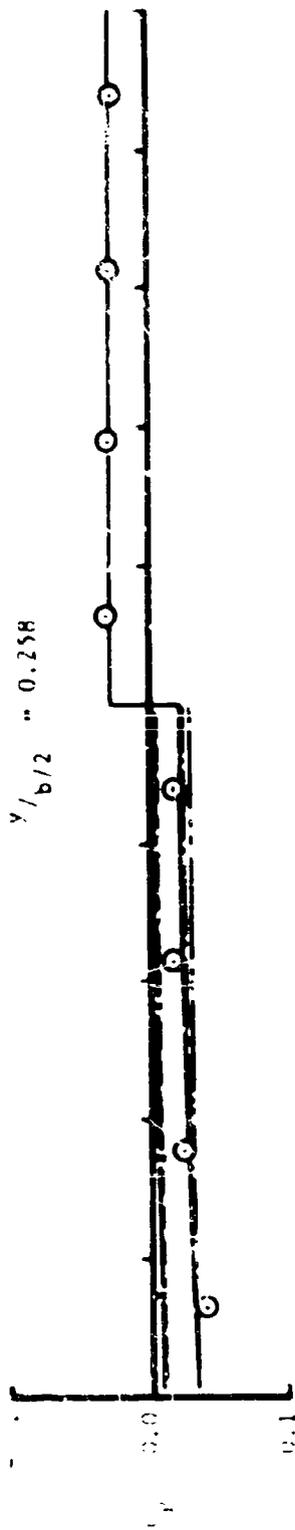


Figure 31. Comparison of Experimental and Theoretical Chordwise Pressure Distribution ($y/b/2 = 0.258$ and 0.446)

$$M_\infty = 4.63 \quad \alpha = 0^\circ \quad \beta = 0^\circ$$

- Tangent wedge (no interference)
- - - Tangent wedge with interference (MOC shock properties)
- Tangent wedge with interference (2nd order shock-expansion shock properties)
- - - Tangent cone with interference (MOC shock properties)
- Tangent cone (no interference)

△ ◇ Experimental data (NASA FN D-6480)

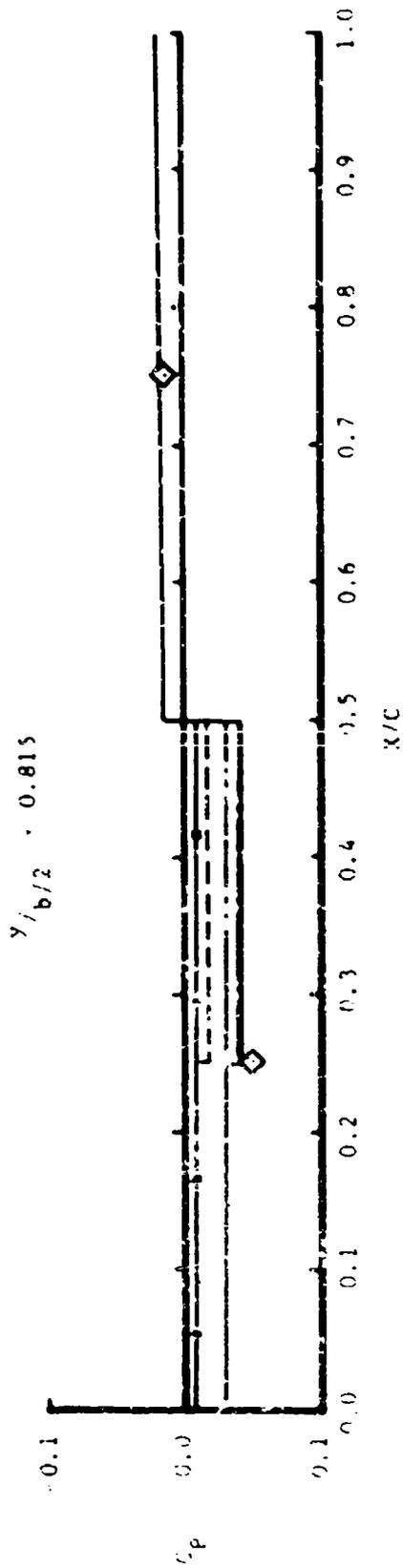
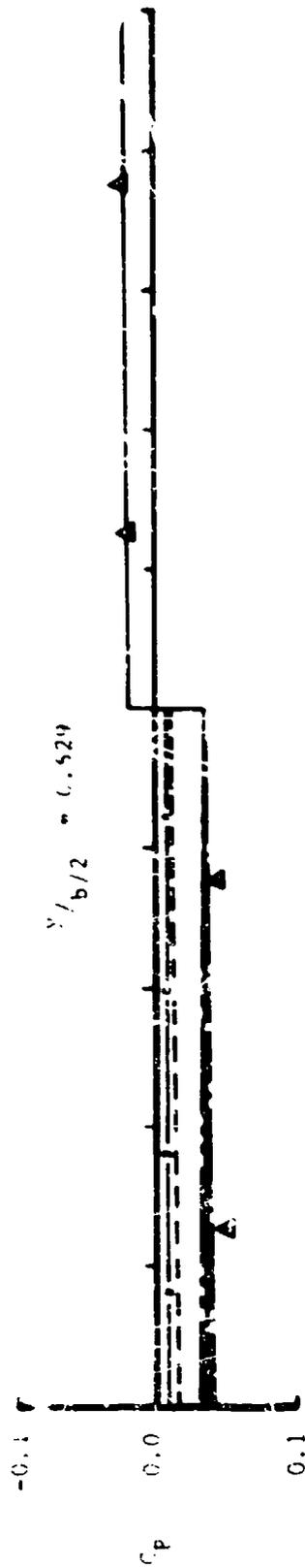


Figure 28. Comparison of Experimental and Theoretical Chordwise Pressure Distribution ($\gamma/b/2 = 0.629$ and 0.815)

SECTION VI

SHIELDING EFFECTS

In the conventional Newtonian formulation of hypersonic flow the pressure coefficient is zero on those portions of the body that are invisible to a distant observer who views the body from the direction of the oncoming freestream. That is, the pressure coefficient is zero on portions of the body that are hidden or shielded by upstream portions of the body surface. To obtain accurate force and moment calculations, such shielded surfaces must be identified and eliminated from the computation. A procedure for accomplishing this is presented in the following discussion*. A general summary of the approach will be presented first followed by a more detailed description of the theory.

With respect to any given direction of the freestream every portion of the body may be classified as either forward-facing or rear-facing. On a forward-facing portion of the surface the dot product of the local outer normal vector with the freestream velocity vector is negative. On a rear-facing portion of the surface the corresponding dot product is positive. Rear-facing portions are always shielded from the freestream direction, and accordingly they do not contribute to the force or moment integrals. The identification of rear-facing surfaces may be performed easily in terms of the above-mentioned dot product. On a convex body, such as an ellipsoid, rear-facing portions of the surface are the only portions that are shielded, and no problems arise. Nontrivial identification problems arise for partially concave bodies or for multiple bodies, where some forward-facing surfaces may be shielded by upstream forward-facing surfaces.

A typical example of shielding on a vehicle in both pitch and yaw is shown in Figure 39. Note that the lower part of the tail and the aft side of the fuselage is shielded from the freestream as is a part of the canopy. Since the basic pressure and force calculations are very rapid, the procedure for identifying shielded quadrilateral elements must also be fast to avoid substantial increases in overall computing time. For some applications flat portions of the body are represented by very large elements, having dimensions that are not small compared to the body dimensions. Thus, it is not sufficient to consider elements as either completely shielded or not shielded at all, but the case of a partially shielded element must be accounted for. Also, the procedure must handle the situation where an element is shielded by an element that is itself shielded by a third element. Finally, the direction of the freestream velocity must be arbitrary. Considerations of a general freestream and a small computing time eliminate a procedure like that of Reference 39. This latter procedure cannot handle a freestream normal to the body axis and is very time consuming because it must calculate a very large number of incremental angles, which presumably must be obtained by means of inverse trigonometric functions.

The low computing time of the Mark IV Program in cases of true hypersonic flow are due to the fact that flow conditions on each surface element are independent of conditions on the other elements. Thus, the flow

* The shielding analysis method described here was derived by J. L. Foss.

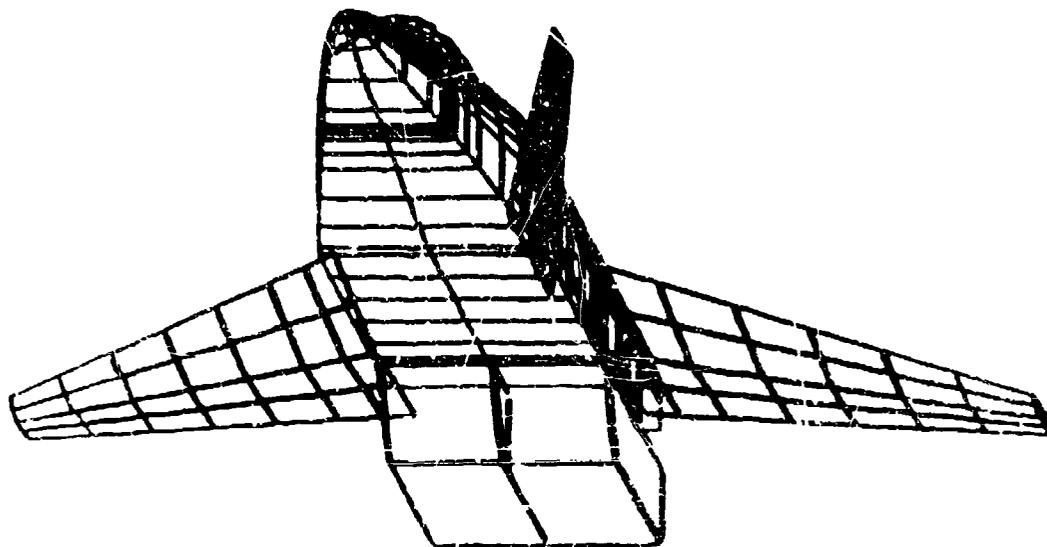


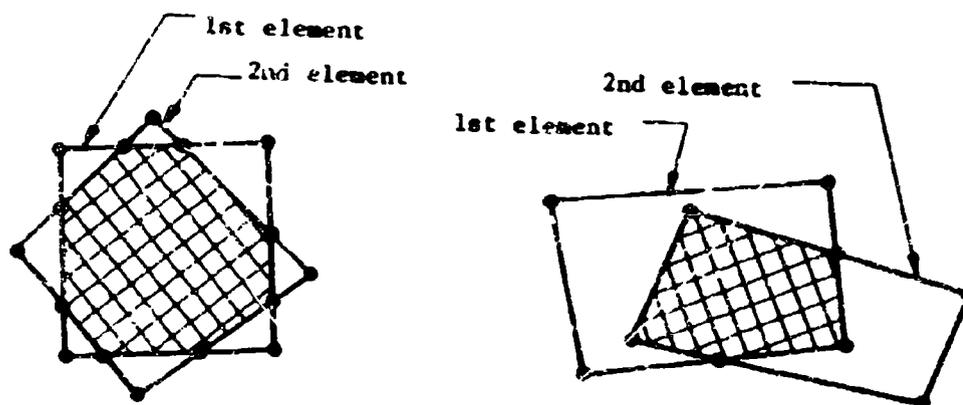
Figure 39. Picture of Vehicle in Pitch and Yaw Illustrating the Shielding Problem.

calculation need be done only N times, where N is the total number of elements on the body. A general routine for identifying shielded elements must allow the possibility that any element may shield any other element. Thus, the test for the shielding of one element by another must be performed a number of times of the order of N^2 . (The actual number of tests is approximately $1/2 N^2$. If the elements are ordered in a sequence, each element need be tested for shielding only with succeeding elements of the sequence, because preceding elements will already have been tested.) Thus, for N of the order of 1000 the shielding test may require more time than the flow calculation. On the other hand, each element is shielded by at most a few elements, so the total number of shieldings is of the order of N . (There would be N^2 shieldings only if every element shielded all other elements.) Thus, the key to a rapid procedure is a very simple test that can be applied to two elements and that will quickly indicate the impossibility of shielding for most pairs of elements. Then the cases of near or actual shielding can be treated more elaborately, because their total number is of order N . Any geometric quantities connected with an element that can aid the computation should be calculated once and for all at the outset and stored, because again only N such calculations are required. Accordingly, the projections of the elements in a plane normal to the freestream velocity are obtained. For each projected element the maximum and the minimum values of the coordinates of the four corner points of the element in this plane are recorded once and for all. Now for the large majority of element pairs the maximum value of a coordinate for one element is less than the minimum value of that coordinate for the other element, and thus no shielding is possible. This is the required test, which could hardly be simpler.

Additional simplification and computing-time reduction are obtained by having the user of the program input the elements grouped into "simple sections", such that no forward-facing element of a section shields any other. This eliminates the need for testing within a section and simplifies the handling of the case when an element is shielded by an element that is itself shielded by a third element. Moreover, the grouping into sections should be easy for the user to accomplish. For example, any convex portion of a configuration is input as one section. The organization of surface elements into panels exists in the Mark IV Program. The shielding procedure simply utilizes this feature.

If a pair of elements fail the simple shielding test, one element may or may not shield the other. As mentioned above, the projections of the elements into a plane normal to the freestream velocity are obtained once and for all. The required calculation determines whether or not the two projected elements overlap and determines the common region, if they do overlap. The projected elements overlap if and only if at least one side of the first element intersects at least one side of the second. It is not sufficient to test whether or not the corner points of one projected element lie inside the other element, because the projected elements may overlap even if all corners of each element lie outside the

other, as illustrated in the sketch below. For convex elements a side of one can intersect at most two sides of the other. The region common to two quadrilateral elements is a polygon with at most eight sides (see sketch). The computational task is to determine the vertices of this



Two Possible Intersections

polygon. As shown in the sketch, each vertex is either an intersection of two sides, one belonging to each projected element, or a corner of one projected element that lies inside the other. Once the polygon is known, it is divided into from one to three quadrilaterals, one of which may have three sides (a special case of a quadrilateral with one side of zero length). This is done so that in subsequent operations all elements are quadrilaterals.

If the projections of two elements overlap, the more downstream element of the two is determined by considering the distances of the two elements from the plane normal to the freestream velocity. The more downstream of the two is the shielded element. It is a simple matter to project the above-described region of overlap (a polygon divided into quadrilaterals) onto the shielded element. This projection is denoted a negative element (or elements).

Finally, all negative elements are known, together with their projections on the plane normal to the freestream velocity. The next stage of the calculation determines the effects of multiple shielding where an element is shielded by an upstream element, which is in turn shielded by a third element. This situation arises when an element is shielded by two (or more) elements, and the two shielded portions overlap. In the present framework the condition is that two negative elements overlap, which is a special case of shielding. Thus, the negative elements are examined for shielding in a manner roughly similar to that outlined above. However, there are some simplifying conditions that ensure that the process of accounting for multiple shielding requires very little computing

time compared to the basic shielding procedure described above. The principal simplification arises from the fact that the only negative elements that can overlap are those that correspond to the same shielded element. Since all negative elements for each shielded element are calculated sequentially and stored together, the searching procedure is very short. A frequently occurring situation consists of two adjacent elements of a section both shielding the same element of another section. In this case it is known in advance that the two shielded portions cannot overlap. The area common to two negative elements is again a polygon that is determined in the above-mentioned way and divided into quadrilaterals. The resulting element is a positive element like the original elements.

The multiple shielding process can be carried on indefinitely to account for an element that is shielded by many others. However, the first application described above appears to cover all cases of practical interest. Simple shielding corresponds to the case where two forward-facing elements lie on a line parallel to the freestream velocity. The first application of multiple shielding covers the case where two forward-facing elements are on a line parallel to the freestream. It is planned to restrict attention to this case initially. This does not restrict the number of sections into which the body may be divided.

After the above elements have been generated, forces and moments on the body are calculated in the usual way by summing the contributions of all the elements. The contributions of the negative elements are multiplied by minus one before summing. The positive elements arising from the overlap of two negative elements are summed as they stand. Thus, for example, in the case of simple shielding the contributions of all the original elements are first added, and the contributions of the shielded portions of the elements are later subtracted to give the desired net forces and moments.

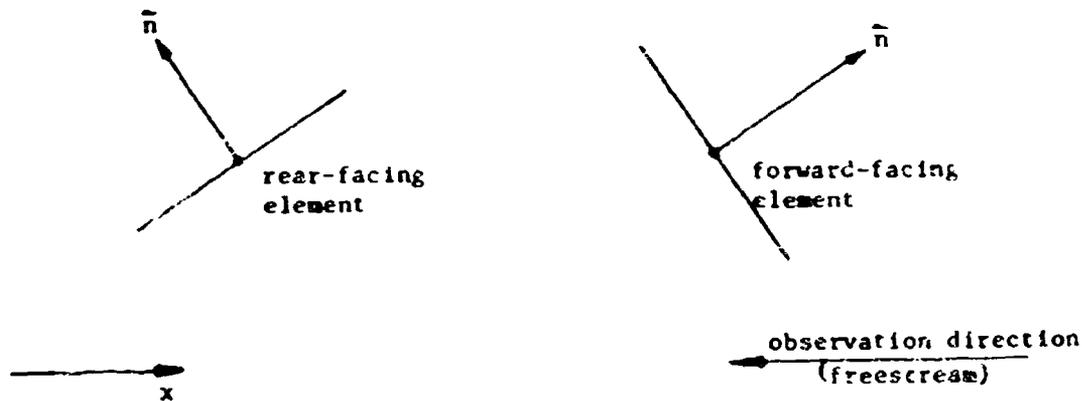
With this general approach description as background, the following discussion provides the detailed procedures used in the shielding computations.

Given a body represented by plane quadrilateral surface elements and given a direction, determine what elements and/or parts of elements are visible to a far-distant observer in the given direction. For a completely general procedure all combinations of shielding elements must be considered. Thus, every element has a potential effect on every other and the "calculation" involves an effort of order N^2 , where N is the number of elements. For the large majority of element combinations the "calculation" consists of a test whose result is negative. Thus the basic "computational operation" is very fast. However, a calculation of the order N^2 can be expected to be time-consuming compared to a much more complicated calculation of order N , such as is accomplished in other parts of the Mark IV program. The principal criterion for formulating a calculation procedure is computation speed.

Division of the Body in Simple Sections

To reduce computing time and simplify the programming logic an assumption is made that somewhat restricts generality of the method and relies on the user to furnish some judgment in inputting a body. However, the scheme adopted appears to apply to all bodies of practical interest for all observation (freestream) directions. Moreover, the judgment required of the user appears reasonable.

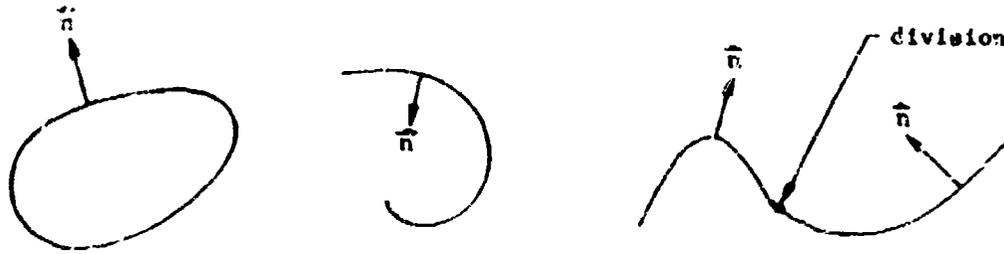
First forward-facing and rear-facing elements must be defined. Suppose there is a vector along the observer's line of sight. Take the dot product of this vector with the unit normal to the element. The dot product is negative for a forward-facing element and positive for a rear-facing element. (See sketch.)



Rear-Facing and Forward-Facing Elements

The input elements are organized into sections as described in Section III. However, it is assumed for the present that the user divides the body into sections in a way that aids the program. Specifically, it is assumed that the body is divided into simple sections. A section is defined as simple if and only if any line parallel to the observation direction intersects no more than one forward-facing element of the section. For example, any entirely convex or entirely concave section is a simple section for all observation directions. See parts (a) and (b) of the sketch below.

A mixed concave-convex section is not a simple section for all observation directions, but it may be for some directions. Usually it is possible to divide the concave-convex section into two simple sections along an inflection line as shown in (c) in the sketch. At worst it might be necessary to run two cases, one divided one way for certain directions and one divided another way for other directions.



(a) Convex Section (b) Concave Section (c) Convex-Concave Section

Element Formation

The procedure for forming elements is a slight addition to the present procedure, and several additional quantities are stored. The present procedure first calculated the coordinates of the four corner points of the quadrilateral element in the reference coordinate system in which the body is input. These are transformed into coordinates based on the element, and the reference coordinates are presently discarded. In the new scheme the reference coordinates of the corner points must also be stored with the geometric quantities that define an element.

Body Rotation

The body is rotated to make the observation (freestream) direction lie along the negative x-axis. Standard rotation formulas are applied.

Rear-Facing Elements

Each element whose normal vector has a negative x-component is eliminated from consideration. This can be done either at this stage or as the element occurs in the procedure below.

Max-Min Coordinate Computation for Each Forward-Facing Element

Let the zy-coordinates of the corner points of an element be denoted y_k, z_k where $k = 1, 2, 3, 4$. Determine

$$y_{\max} = \max (y_1, y_2, y_3, y_4)$$

$$y_{\min} = \min (y_1, y_2, y_3, y_4)$$

$$z_{\max} = \max (z_1, z_2, z_3, z_4)$$

$$z_{\min} = \min (z_1, z_2, z_3, z_4)$$

(A-1)

These must be recorded either logically by integer designation or physically by storing the selected max-min coordinates an additional time. This last would add four additional storage quantities for each forward-facing element.

Section Ordering

While the computation (A-1) above is proceeding, the maximum value of x for each element is computed but not stored. It is compared with the largest maximum of previously considered elements. The final result is the largest value of x of any forward-facing element in the section. Call this $x(\text{MAX})$. The sections are ordered in increasing order of $x(\text{MAX})$ starting with the smallest. With this definition, each section is behind all subsequent sections as viewed from the observation direction.

Basic Problem. Element Overlap

The basic problem of this computation consists of determining what elements are blocked or shielded by others as seen from the observation direction and of determining the geometry of the shielded region. All calculation up to here has been preparatory and has been done essentially once for each element or N times - not N^2 .

Since the observation direction is parallel to the x -axis, the question of whether or not one element shields another is equivalent to whether or not their projections in the yz -plane intersect or overlap. This is the test that must be made an order of N^2 times. Some information is already available before any testing:

- a. Elements in the same simple section cannot shield the other.
- b. If two or more elements of one section shield one element of another section, the various shielded portions cannot themselves overlap.
- c. If the yz -projections of two elements overlap, it is the element of the higher-ordered section that shields the element of lower-ordered section - not the opposite.

Projected Elements

An element is projected into the yz -plane by simply ignoring the x -coordinates and considering only the yz -reference coordinates of the corner points. The element in the yz -plane that is obtained this way is denoted a projected element. It is the projected elements that are tested for overlap.

Direct Coordinate Test

The sections are considered in order beginning with the lowest ordered. Each projected element of a section is tested for overlap with all elements of all subsequent sections. This test is done in more than one way to minimize computing time in comparing two elements: one is

called the first element and the other the second element. It is natural to denote the element that is being tested for overlap with all others (of subsequent sections) as the first element. Thus, if there is overlap, it is the first element that is shielded by the second.

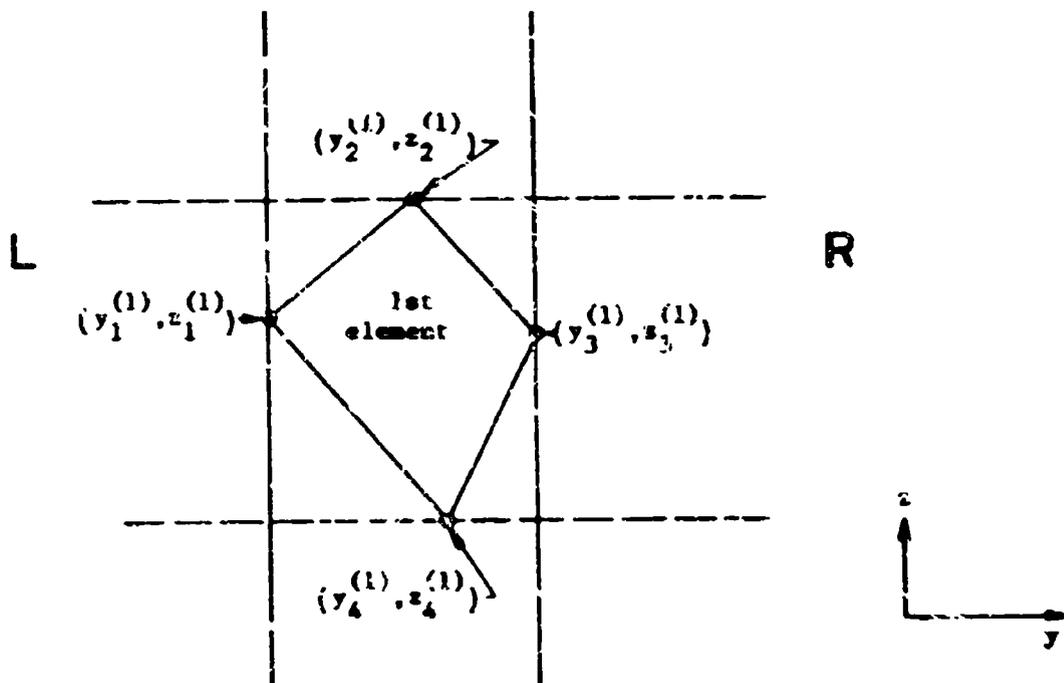
Most element pairs are 'sufficiently disjoint' so that their nonoverlap can be revealed by the very simple direct coordinate test. Suppose the first projected element has corner point coordinates $y_k^{(1)}$, $z_k^{(1)}$, and the second has corner point coordinates $y_k^{(2)}$, $z_k^{(2)}$, where in both cases $k = 1, 2, 3, 4$. Maximum and minimum y and z are known for each element from (A-1)

A sufficient condition for nonoverlap is that all $y_k^{(2)}$ are greater (or less) than all $y_k^{(1)}$. A similar statement holds for the z 's. These conditions are equivalent to the following inequalities

$$\begin{aligned} (y_{\max}^{(2)} - y_{\min}^{(1)}) (y_{\min}^{(2)} - y_{\max}^{(1)}) &> 0 \\ (z_{\max}^{(2)} - z_{\min}^{(1)}) (z_{\min}^{(2)} - z_{\max}^{(1)}) &> 0 \end{aligned} \tag{A-2}$$

If the y inequality is satisfied, the two projected elements do not overlap, and the z inequality need not be performed.

To see the meaning of the inequalities (A-2), let the first projected element be as shown in the sketch below. The y inequality is satisfied for



all second elements that lie entirely in region R or entirely in region L. Probably most elements do so and thus, one test is required. The inequality is not satisfied for second elements that intersect or lie between the solid vertical lines. However, the z-inequality is satisfied for all such elements that lie entirely above or below both dotted horizontal lines. Probably most second elements that do not satisfy the y-inequality do satisfy the z-inequality.

The very simple inequalities (A-2) reveal nonoverlap for most element pairs and the N^2 part of the calculation consists mainly of these two tests (only one in a majority of cases).

Left-Right Test

This is applied to a pair of elements that fail to satisfy inequalities (A-2).

The "side vectors" of the first projected quadrilateral element are needed and probably those of the second. These are to be computed ahead of time for all elements and stored. The side vectors are

$$\begin{aligned}\vec{s}_{12} &= (y_2 - y_1)\vec{j} + (z_2 - z_1)\vec{k} \\ \vec{s}_{23} &= (y_3 - y_2)\vec{j} + (z_3 - z_2)\vec{k} \\ \vec{s}_{34} &= (y_4 - y_3)\vec{j} + (z_4 - z_3)\vec{k} \\ \vec{s}_{41} &= (y_1 - y_4)\vec{j} + (z_1 - z_4)\vec{k}\end{aligned}\tag{A-3}$$

Superscripts 1 and 2 will be used to denote quantities associated with the first and the second projected elements, respectively. Now two projected elements overlap if and only if one of the following conditions is satisfied. Either: (1) at least one side of the first intersects at least one side of the second, or (2) one element completely contains the other. This last occurs infrequently and is handled separately. Thus, the basic operation here is to determine if a particular side of the first element and a particular side of the second element intersect. Consider the side 12 of the first element (the procedure is identical for the other sides). A point (y, z) is said to lie to the left of this side if it is left with respect to the side vector $\vec{s}_{12}^{(1)}$. Form the quantity

$$\vec{r}_{12}(y, z) = (y - y_1^{(1)})\vec{j} + (z - z_1^{(1)})\vec{k}\tag{A-4}$$

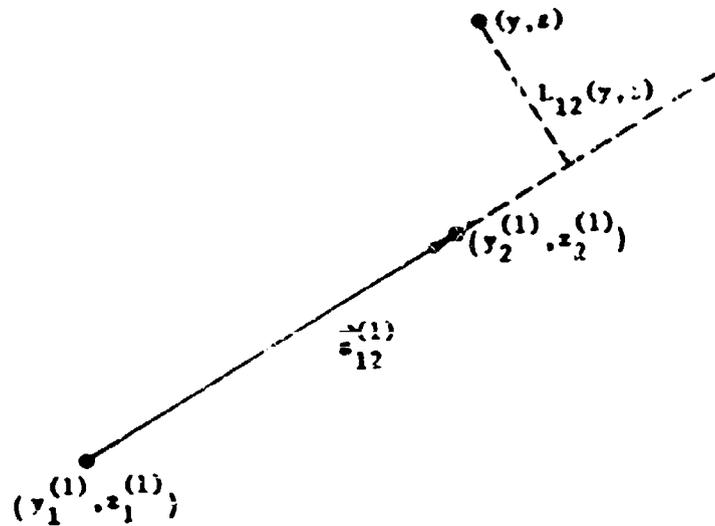
Now

$$\vec{s}_{12}^{(1)} \times \vec{r}_{12}(y, z) = [L_{12}(y, z)]\vec{i}\tag{A-5}$$

where

$$L_{12}(y, z) = (z - z_1^{(1)})(y_2^{(1)} - y_1^{(1)}) - (y - y_1^{(1)})(z_2^{(1)} - z_1^{(1)})\tag{A-6}$$

and four differences in L_{12} have already been computed. $L_{12}(y, z)$ is the perpendicular distance that the point (y, z) lies to the left of the extension of side 12 (see sketch below). In particular, $L_{12}(y, z)$ is positive if (y, z) is to the left of 12 and negative if it is to the right. Now side 12 of the first projected element and a side of the second projected element intersect if and only if the end points of each are in opposite directions with respect to the other. This requires that one endpoint of the side of the second element lies to the left of side 12 of the first element and the other endpoint of the side of the second element lies to the right of side 12 of the first element. A similar statement must hold for the end points of side 12 of the first element with respect to the side of the second element.



Thus, first consider side 12 of the first element and compute left distances for all four corner points of the second element

$$L_{12}(y_1^{(2)}, z_1^{(2)})$$

$$L_{12}(y_2^{(2)}, z_2^{(2)})$$

$$L_{12}(y_3^{(2)}, z_3^{(2)})$$

$$L_{12}(y_4^{(2)}, z_4^{(2)})$$

(A-7)

If all four are of the same sign, no intersections with side 12 are possible. If two consecutive L_{12} are of opposite sign (counting the first and last as consecutive), an intersection of side 12 with the side between the two points in question is possible. For example, if

$L_{12}(y_1^{(2)}, z_1^{(2)})$ and $L_{12}(y_2^{(2)}, z_2^{(2)})$ are of opposite sign, an intersection of side 12 of the first element and side 12 of the second element is possible. To verify the possible intersection, take the pertinent side of the second element (12 in the example) and compute L's for the two end points of side 12 of the first element. There is an intersection if and only if these are of opposite sign. Such a check is necessary only for the "sign changes" of the sequence (A-7). If two consecutive L_{12} of (A-7) have the same sign, the side of the second element between the points in question cannot intersect side 12 of the first element. For convex elements, a side of the first element can intersect no more than two sides of the second element.

The above procedure is repeated for all four sides of the first element. The results determine which sides of the first intersect which sides of the second. Also calculated are the sixteen L's of the four corner points of the second element with respect to all four sides of the first element and the L's with respect to the sides of the second for cases of possible or actual intersection.

Final Element Classification

The procedure above determines whether any of the sides of the two projected elements intersect. If there are intersections, the two elements overlap. If there are not, there are three possibilities. Either: (1) the elements do not overlap, or (2) the first element completely contains the second, or (3) the second element completely contains the first. If there is no overlap, this phase of calculation is complete and logic proceeds to the next element pair.

The first element completely contains the second if and only if all sixteen of the L's of the corner points of the second element with respect to the sides of the first element are negative.

The first element is completely contained in the second if and only if the L's test showed that each side of the first element had exactly two possible intersections with sides of the second element and all eight turned out not to be intersections.

General Handling of Overlapping Projected Elements

The basic calculational task for overlapping projected elements consists of determining the polygonal area common to the two elements: projecting the polygon onto the shielded element (the first element in this scheme), and treating the result as a negative element in force and moment calculations. That is: (1) generate all necessary geometric quantities describing the shielded portion, and (2) put them aside for later use as a negative element.

Completely Contained Projected Elements

If the projection into the yz-plane of one element completely contains the other, there are two possibilities which are treated as described below.

The first possibility is that the contained element is shielded. This means the first element is contained. In this case a duplicate of the first element is added to the negative elements and no additional overlap comparisons are made with that particular element as first element. However, the element is maintained among the normal positive elements and any previous intersections are left undisturbed. This last consideration is the reason for this unusual treatment. If the first element were to be simply eliminated, a search for previous intersections would have to be made, and they would have to be eliminated also. However, there is no point in looking for subsequent overlaps.

The second possibility is that the contained element shields the other, i.e., second element is contained. In this case the negative element is the projection of the contained (second) element onto the containing (first) element.

Overlapping Projected Elements with Intersecting Sides

The usual case of overlap is that for which one or more sides of the projected elements intersect. For convex elements, a side of one element intersects: (1) no sides of the other element, (2) one side of the other element, or (3) two sides of the other element. The first element is the one that is shielded. The negative element is the projection on the first element of the common area of the projected elements in the yz-plane. The principal task is to determine the common area of the elements in the yz-plane and to divide this area into quadrilaterals and/or triangles. Projection of the area onto the first element is then rather easy. The common area is a polygon. All vertices of the polygon are determined in clockwise order about the perimeter. (Recall that the four corner points of any forward-facing quadrilateral element are in clockwise order in the yz-plane.) The vertices of the desired polygon consist of: (1) points of intersection of the sides of the two quadrilaterals, (2) corner points of the first element that lie inside the second.

The yz-coordinates of the intersections can be written down easily in terms of the L's calculated as in equation (A-6). Basically, a 2x2 set of linear equations is solved, but some of the work has already been done in calculating L's. Thus, for each intersection point its yz-coordinates and a designation of which sides of the two elements intersect at that point are available. A corner point of the second element lies inside the first if and only if all four of the L's that apply to that point are negative. Thus, the information for tagging each corner point of the second as lying inside or outside the first is already available. Generally, the same information is not available for all four corners of the first element, because all sixteen L's of the corners of the first element with respect to the second have not been calculated by the procedure.

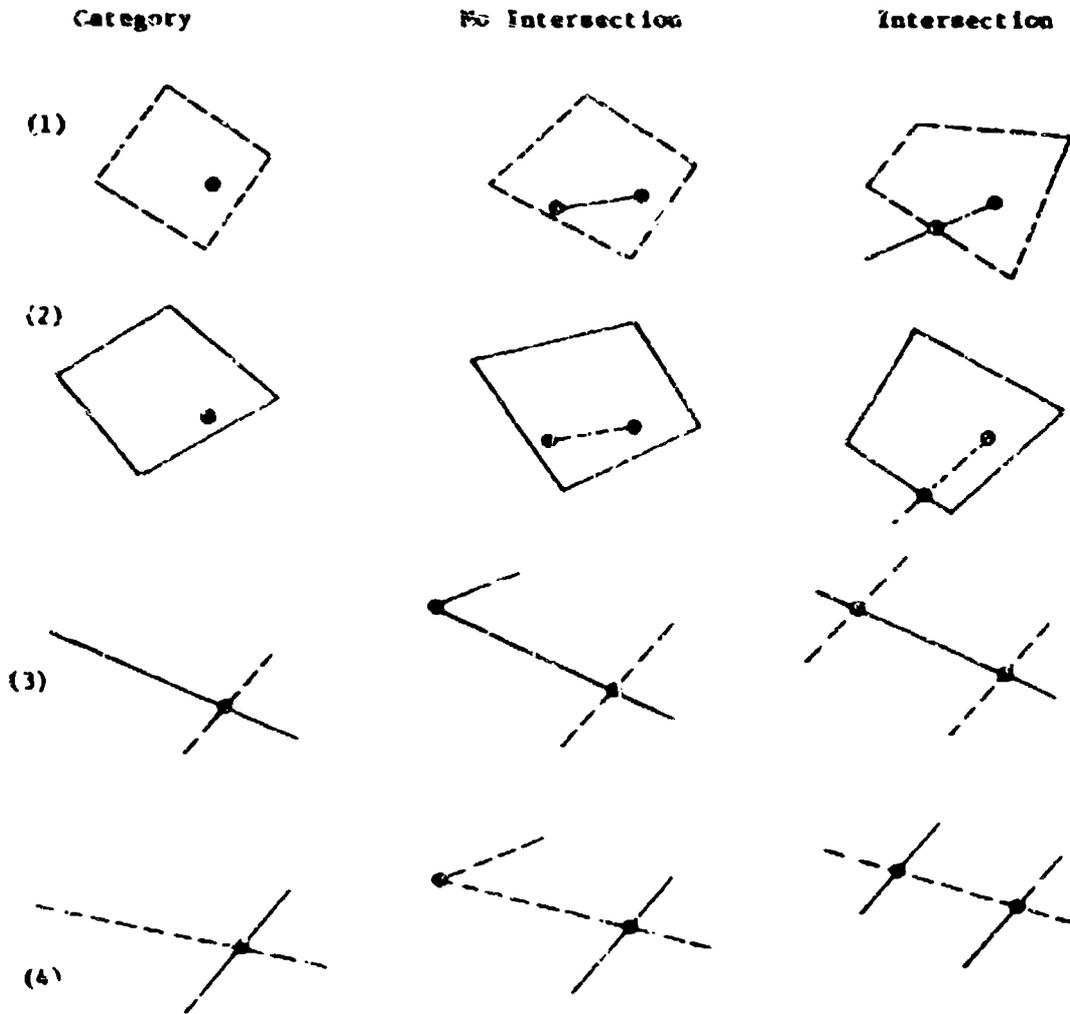
Two cases will be considered separately: (1) at least one corner of the second projected element is inside the first, and (2) no corner of the second is inside the first. The difference between these two cases lies solely in the rule for initiating the vertex search. Once the process has begun, it is identical in both cases.

If a nonzer - number of corners of the second lie inside the first case 1), start with the lowest numbered corner (1, 2, 3, or 4). The first vertex of the common area polygon is simply that corner point. Now consider the side of which the given corner is the initial point. For example, if the first vertex is corner point 3, consider side 34. Now check the information on intersections to determine how many times this side intersects sides of the first element. There are only two possibilities: zero or one. (If a side has two intersections, both of its end points must be outside the other element). If there are zero intersections, the other end point of the side (corner point 4 in the example) is also inside, and it is the next vertex of the common area polygon. In this case the calculation proceeds to the next side (41 in the example). If there is one intersection, it is the next vertex of the common area polygon. In this case, determine which side of the first element has been intersected; then determine whether or not there is another intersection on this side of the first element. If there is, it is the next vertex. If not, the terminal point of the side, e.g., corner point 3 on side 23, is the next vertex. In the latter case the next stage of the calculation is like the original stage. In the former case, the next stage is like that following the first intersection with a side of the element. It can be seen that there are only four truly distinct operations in the above scheme corresponding to the fact that there are only four essentially different starting points. The operations of the procedure for determining vertices of the common area polygon may be divided into four categories, associated with the four kinds of starting points, as follows.

1. Interior corner point, first element.
2. Interior corner point, second element.
3. Side intersection - continue first element.
4. Side intersection - continue second element.

The third category refers to the fact that if previously an intersection with a side of the first element has been determined by extending a side of the second element from within, then the next vertex is to be sought on that same side of the first element. Category 4 refers to an analogous situation for a side of the second element. Obviously, categories (1) and (2) and categories (3) and (4) are symmetric and only two algorithms are needed. For each category there are exactly two possibilities for the next vertex depending on whether or not there is an (additional) intersection on the side in question. The possibilities are illustrated in the sketch below, where solid lines are used to denote the first element and dotted lines are used for the second element.

Determination of Vertices of the Common Area Polygon



The basic procedure can be summarized by the table below:

Category	Category (1)		Category (2)		Category (3)		Category (4)	
Starting Point:	Interior Corner First Element		Interior Corner Second Element		Intersection Continue First Element		Intersection Continue Second Element	
Does side have an (additional) intersection?	Yes	No	Yes	No	Yes	No	Yes	No
Next vertex	I	T(1)	I	T(2)	I	T(1)	I	T(2)
Next category	(4)	(1)	(3)	(2)	(4)	(1)	(3)	(2)

I = the intersection point

T(1) = terminal point of the side of the first element

T(2) = terminal point of the side of the second element

As mentioned above, there are two cases as far as starting the above procedure: (1) start at a corner of the second element that is inside the first (category (2)), or (2) there are no such corners. The second case is further divided into two parts so that altogether there are three starting conditions.

If no corner of the second element lies inside the first, consider first the lowest ranking side of the first element that has two intersections (if there is one). Determine which intersection is nearer the initial point of the side. Take that intersection as the first vertex of the common area polygon and the other as second vertex. Then continue the above procedure commencing with the second intersection as starting point with the category (4) procedure.

If no side has two intersection points, take the lowest ranking side of the first element that has an intersection and test the two endpoints of the side one by one to determine which one is interior to the second element. (One must be because there is just one intersection.) This involves examining the signs of some L 's, some of which may already be computed from equation (A-6). If one end point is interior, the other cannot be, and there is no need to test it. There are two possibilities. If the initial point of the side is interior, take it as the first vertex, the intersection as the second vertex, and initiate the above procedure starting with the intersection and category (4). If the terminal point is interior, take the intersection as first vertex, the terminal point as second vertex, and initiate the above procedure with the terminal point as starting point and category (1).

In all cases the above procedure is continued until the next vertex determined is the first vertex. That is, continue until it "comes back where it started". Now all vertices of the common area polygon are known in clockwise order in the yz -plane.

Division of the Common Area Polygon Into Quadrilaterals

It would be possible to deal with the common area polygon directly as a polygon, but it appears more efficient to subdivide it. The polygon has either 3, 4, 5, 6, 7 or 8 sides. Associate the vertices in consecutive groups of four with possibly three in the last group. The groupings are as follows (other groupings are possible):

No. of Vertices	Groupings of Vertices		
3	1 2 3		
4	1 2 3 4		
5	1 2 3 4	4 5 1	
6	1 2 3 4	4 5 6 1	
7	1 2 3 4	4 5 6 7	4 7 1
8	1 2 3 4	4 5 6 7	4 7 8 1

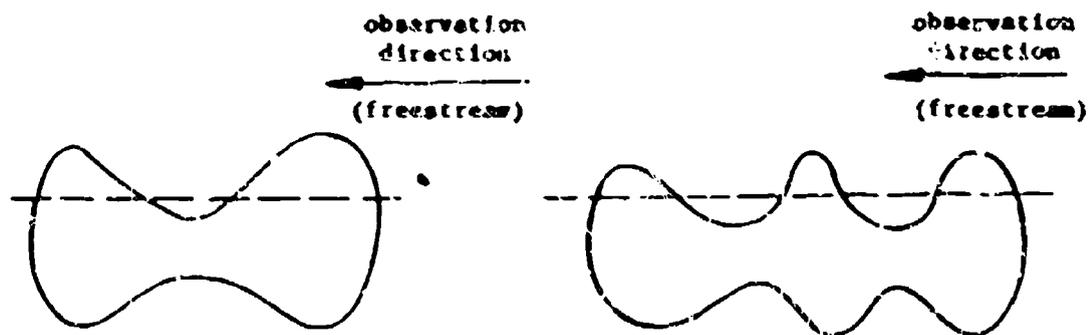
Thus the polygon is divided into either one, two or three elements. Either all of these elements are quadrilaterals or all but one are. The remaining element (if any) is a triangle, which is a special case of a quadrilateral. These are the projections of the negative elements associated with the intersection of the two original elements.

Formation of the Negative Elements

The yz -projections of the negative elements that are obtained above must be projected onto the shielded element, which is the first element of the two original elements. The most efficient way to do this is to compute geometric quantities for the negative elements in the yz -plane and then adjust these quantities by means of the x -component of the normal vector of the shielded element. This is a well-known procedure. In typical cases the number of element shieldings should be of the same order or less than the number of elements. Thus, the computing time for forming the negative elements should be no greater than that for the original element formation. Also recorded are the shielded and the shielding element for each negative element.

Multiple Intersections

The procedure is now finished unless there are multiple intersections. These occur when the yz -projections of two or more negative elements overlap. This situation arises only when at least three forward-facing elements are intersected by a line parallel to the observation direction. For a closed body, this means that some line parallel to the observation direction must intersect the body at least six times. These considerations are illustrated in the sketch below. The case when some line parallel to the observation direction intersects exactly three forward-facing elements and no line intersects more than three is denoted simple multiple intersection, because only two (not more) negative elements overlap in their yz -projections. It appears that all bodies of practical interest are included in the case of simple multiple intersection, and attention will be restricted to that case.



(a) No multiple intersection

(b) Simple multiple intersection

Pressure and Moment Calculation in the Case of No Multiple Intersections

These are done exactly as they are in the existing Hypersonic Arbitrary-Body Aerodynamic Computer Program. Pressures and moments are computed for all elements. When they are added together, the contributions of the negative elements are reversed in sign. Thus, the contribution of a shielded portion of an element is added positively with the element itself and then cancelled by adding the negative effect of the corresponding negative element.

Determination of Simple Multiple Intersection

Negative elements are stored according to the first or shielded element and are labeled with what element did the shielding and to what section the latter belongs. It is easy then to search for cases where one element is shielded by more than one other element, because the corresponding negative elements are stored together. If this occurs, the area common to the negative elements must be determined so that it will not be subtracted more than once.

If an element is shielded by two or more elements of the same simple section, the resulting negative elements cannot overlap. Thus, the condition for multiple intersection is that an element is shielded by elements from two or more different sections.

Determination of the overlap of negative elements is essentially the same as the original element-overlap calculation. First, notice that all elements are in the same plane, namely the plane of the shielded element. Thus, all calculations are carried out in this plane and no projection is subsequently necessary.

The logic of the calculation is as before. The negative elements for each section are tested for overlap with those of each succeeding section and common area polygons computed. Now, however, the "unit calculation" consists of several comparisons because each negative element may consist of several quadrilateral elements. The resulting common area polygons are associated with the lower (shielded) section. The resulting "negative elements" are positive elements and are treated as such in the force and moment calculations. Thus, the fact that two negative elements overlap means that by subtracting the contributions of both of them in the force and moment computations, too much has been subtracted. This is corrected by addition of a positive element identical to the common area of the two negative elements.

SECTION VII

COMPUTATION OF VEHICLE FORCES

Calculation of Local Flow Conditions

In the geometry part of the program the input element is converted into a plane quadrilateral element. The quadrilateral is described by its area, the coordinates of the centroid of the element and by the direction cosines of the surface unit normal. In the force calculation methods we must also know the angle that the element makes with the freestream velocity vector (the impact angle). This angle changes as the vehicle attitude (angle of attack, yaw, and roll angle) changes. The impact angle may be found from the following relationship:

$$\delta = \pi/2 - \theta$$

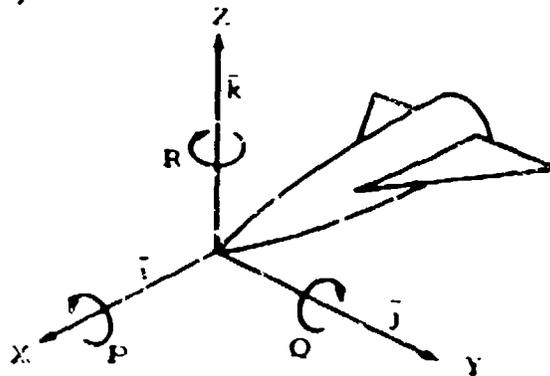
$$\cos \theta = \frac{\bar{n} \cdot \bar{V}}{|\bar{n}| |\bar{V}|}$$

where

\bar{n} is the unit normal outward from the surface with direction cosines n_x, n_y, n_z

\bar{V} is the local velocity vector with direction cosines in the vehicle coordinate system given by V_x, V_y, V_z

The direction cosines of the unit surface normal are given by the quadrilateral calculations. The value of the local velocity vector V depends upon the vehicle attitude with respect to the freestream direction and its angular rotation rates, and is derived in the discussion below. The rotation directions are consistent with the conventional stability body-axis system. The coordinate system, however, is changed to be consistent with the geometric description system discussed previously.



where

- P = rolling velocity
- Q = pitching velocity
- R = yawing velocity
- Ω = total angular velocity

The movement of a given element of the vehicle with respect to the freestream depends upon the vehicle rotation rate and the position of the element relative to the rotation center. The radius vector from an arbitrary reference point on the vehicle to a point on the surface is given by

$$\vec{r} = (x - x_0) \vec{i} + (y - y_0) \vec{j} + (z - z_0) \vec{k}$$

x_0, y_0, z_0 is the moment reference point (center of gravity).

total angular velocity is given by

$$\vec{\Omega} = P \vec{i} - Q \vec{j} - R \vec{k}$$

The freestream velocity vector is given by

$$\vec{V}_\infty = V_{\infty x} \vec{i} + V_{\infty y} \vec{j} + V_{\infty z} \vec{k}$$

The total velocity vector relative to the surface element obtained by combining the above relationships as follows:

$$\vec{V} = \vec{V}_\infty - \vec{\Omega} \times \vec{r}$$

The local velocity vector therefore becomes

$$\begin{aligned} \vec{V} &= \left\{ V_{\infty x} + [Q(z-z_0) - R(y-y_0)] \right\} \vec{i} \\ &+ \left\{ V_{\infty y} + [R(x-x_0) + P(z-z_0)] \right\} \vec{j} \\ &+ \left\{ V_{\infty z} - [P(y-y_0) + Q(x-x_0)] \right\} \vec{k} \end{aligned}$$

$$\text{or } \vec{V} = V_x \vec{i} + V_y \vec{j} + V_z \vec{k}$$

where

$$\begin{aligned} V_x &= V_{\infty x} + [Q(z-z_0) - R(y-y_0)] \\ V_y &= V_{\infty y} + [R(x-x_0) + P(z-z_0)] \\ V_z &= V_{\infty z} - [P(y-y_0) + Q(x-x_0)] \end{aligned}$$

The total local velocity is given by

$$V_{\text{local}} = \sqrt{V_x^2 + V_y^2 + V_z^2}$$

The conventional surface impact angle is then given by

$$\delta = \pi/2 - \cos^{-1} \left(\frac{-n_x V_x - n_y V_y - n_z V_z}{V_{\text{local}}} \right)$$

where n_x , n_y , n_z , are the outward surface unit normal direction cosines

To complete the preceding computations we must obtain the freestream velocity components; $V_{\infty x}$, $V_{\infty y}$, and $V_{\infty z}$. These equations are derived below by using the conventional right-handed coordinate system and applying the necessary rotation matrices using a yaw-pitch-roll sequence.

Rotation about x-axis, yaw

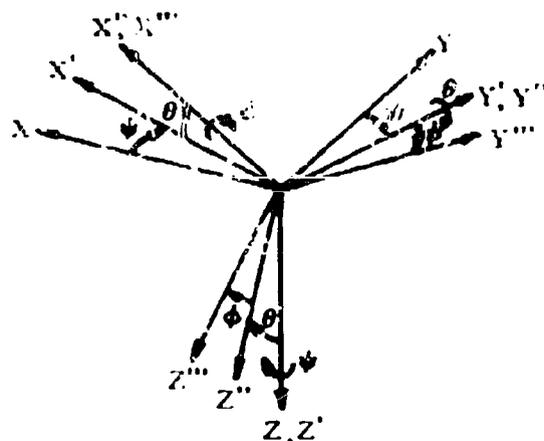
$$[\psi] = \begin{bmatrix} \cos \psi & \sin \psi & 0 \\ -\sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Rotation about y-axis, pitch

$$[\theta] = \begin{bmatrix} \cos \theta & 0 & -\sin \theta \\ 0 & 1 & 0 \\ \sin \theta & 0 & \cos \theta \end{bmatrix}$$

Rotation about z-axis, roll

$$[\phi] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & \sin \phi \\ 0 & -\sin \phi & \cos \phi \end{bmatrix}$$



The complete rotation matrix is as follows

$$[A] = [\phi][\theta][\psi]$$

$$[A] = \begin{bmatrix} \cos \theta \cos \psi & \cos \theta \sin \psi & -\sin \theta \\ \sin \phi \sin \theta \cos \psi - \cos \phi \sin \psi & \sin \phi \sin \theta \sin \psi + \cos \phi \cos \psi & \sin \phi \cos \theta \\ \cos \phi \sin \theta \cos \psi + \sin \phi \sin \psi & \cos \phi \sin \theta \sin \psi - \sin \phi \cos \psi & \cos \phi \cos \theta \end{bmatrix}$$

$$\begin{bmatrix} V_{\infty x}''' \\ V_{\infty y}''' \\ V_{\infty z}''' \end{bmatrix} = [A] \begin{bmatrix} -V_{\infty} \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} -V_{\infty} \cos\theta \cos\psi \\ -V_{\infty} \sin\phi \sin\theta \cos\psi + V_{\infty} \cos\phi \sin\psi \\ -V_{\infty} \cos\phi \sin\theta \cos\psi - V_{\infty} \sin\phi \sin\psi \end{bmatrix}$$

To be consistent with the coordinate and sign conventions used in the program we must now apply the following relationships

$$V_{\infty x} = V_{\infty x}''', \quad V_{\infty y} = -V_{\infty y}''', \quad V_{\infty z} = -V_{\infty z}'''$$

$$\alpha = \theta, \quad \beta = -\psi, \quad \phi = \phi$$

where

α = angle of attack (+ with vehicle nose up)

β = sideslip angle (+ with vehicle nose left)

ϕ = roll angle (+ with right wing down)

The freestream velocity components are, therefore, given by

$$V_{\infty x} = -V_{\infty} \cos\alpha \cos\beta$$

$$V_{\infty y} = V_{\infty} \sin\phi \sin\alpha \cos\beta + V_{\infty} \cos\phi \sin\beta$$

$$V_{\infty z} = V_{\infty} \cos\phi \sin\alpha \cos\beta - V_{\infty} \sin\phi \sin\beta$$

Coefficient Transformations

The conversion of the axial force and normal force coefficients to lift and drag coefficients requires the following matrix operation.

$$\begin{bmatrix} C_D' \\ C_Y' \\ C_L' \end{bmatrix} = [A]^{-1} \begin{bmatrix} -C_A \\ C_Y \\ -C_N \end{bmatrix}$$

where

$$[A]^{-1} = \begin{bmatrix} \cos\theta \cos\psi & \sin\phi \sin\theta \cos\psi - \cos\phi \sin\psi & \cos\phi \sin\theta \cos\psi + \sin\phi \sin\psi \\ \cos\theta \sin\psi & \sin\phi \sin\theta \sin\psi + \cos\phi \cos\psi & \cos\phi \sin\theta \sin\psi - \sin\phi \cos\psi \\ -\sin\theta & \sin\phi \cos\theta & \cos\phi \cos\theta \end{bmatrix}$$

Since

$$C_D = -C_D', \quad C_L = -C_L', \quad C_Y' = C_Y'$$

$$\alpha = \theta, \quad \beta = -\phi, \quad \phi = \phi$$

the final force coefficient transformation equations become

$$C_D = C_A \cos \alpha \cos \beta - C_Y \sin \phi \sin \alpha \cos \beta - C_Y \cos \phi \sin \beta$$

$$+ C_N \cos \phi \sin \alpha \cos \beta - C_N \sin \phi \sin \beta$$

$$C_Y' = C_A \cos \alpha \sin \beta - C_Y \sin \phi \sin \alpha \sin \beta + C_Y \cos \phi \cos \beta$$

$$+ C_N \cos \phi \sin \alpha \sin \beta + C_N \sin \phi \cos \beta$$

$$C_L = -C_A \sin \alpha - C_Y \sin \phi \cos \alpha + C_N \cos \phi \cos \alpha$$

In some parts of the program it is necessary to know the direction in which an element shear force is acting (i.e., free molecular flow, skin friction). This direction is assumed to be in the plane of the surface outward normal and the local incident velocity vector, and is determined by taking successive vector products as follows.

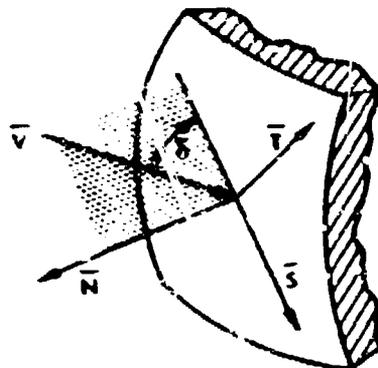
Surface Velocity Vector

The procedure is illustrated in the accompanying sketch where the incident velocity vector is defined as

$$\bar{V} = V_x \bar{i} + V_y \bar{j} + V_z \bar{k}$$

and the surface normal as

$$\bar{N} = n_x \bar{i} + n_y \bar{j} + n_z \bar{k}$$



First, a surface tangent vector (\bar{T}) is defined by the cross product of the normal and velocity vectors:

$$\bar{T} = T_x \bar{i} + T_y \bar{j} + T_z \bar{k}$$

where

$$T_x = n_y V_z - n_z V_y$$

$$T_y = n_z V_x - n_x V_z$$

$$T_z = n_x V_y - n_y V_x$$

Then the direction of the shear force (\vec{S}) is given by the cross product of the surface tangent and normal vectors:

$$\vec{S} = S_x \vec{i} + S_y \vec{j} + S_z \vec{k}$$

where

$$S_x = T_y n_z - T_z n_y$$

$$S_y = T_z n_x - T_x n_z$$

$$S_z = T_x n_y - T_y n_x$$

Pressure Coefficient Corrections

In the program force calculations the pressure on each element is calculated completely independent of all other elements (except the shock-expansion method). If the vehicle is rotating the local pressure coefficient must be corrected back to freestream conditions. This is accomplished by the following relationship.

$$C_{p_l} = C'_{p_{local}} \left(\frac{V_{local}}{V_{\infty}} \right)^2$$

where

C_{p_l} = pressure coefficient based on local conditions and including vehicle rotation rate correction

$C'_{p_{local}}$ = local pressure coefficient without local rotation rate velocity correction

When interference effects are being accounted for, the pressure coefficient is determined on the basis of a "local" freestream condition as interpolated from the flow field data. This coefficient is corrected to the real freestream conditions by the equation below.

$$C_p = \frac{\left(C_{p_l} \frac{\gamma}{2} M_l^2 + 1.0 \right) \frac{p_l}{p_{\infty}} - 1}{\frac{\gamma}{2} M_{\infty}^2}$$

where

C_p = pressure coefficient based on freestream conditions

C_{p_l} = pressure coefficient based on local interference flow field

M_l = Mach number based on local interference flow field, FS(0)

$\frac{P_l}{P_\infty}$ = ratio of local interference flow field pressure to freestream pressure, DINFL(5)

M_∞ = freestream Mach number

Vehicle Force Coefficients

In the arbitrary-body program, the contribution of each element to three force coefficients and three moment coefficients are calculated. The basic relationships to accomplish this are as follows:

$$\text{axial force} \quad \Delta C_A = (C_p n_x - C_f S_x) \frac{\Delta A}{S_{ref}}$$

$$\text{side force} \quad \Delta C_Y = (C_p n_y - C_f S_y) \frac{\Delta A}{S_{ref}}$$

$$\text{normal force} \quad \Delta C_N = -(C_p n_z + C_f S_z) \frac{\Delta A}{S_{ref}}$$

$$\text{rolling force} \quad \Delta C_l = \Delta C_Y \frac{z}{b} + \Delta C_N \frac{y}{b}$$

$$\text{pitching moment} \quad \Delta C_m = \Delta C_N \frac{x}{c} + \Delta C_A \frac{z}{c}$$

$$\text{yawing moment} \quad \Delta C_n = \Delta C_Y \frac{x}{b} - \Delta C_A \frac{y}{b}$$

where

ΔA = element area

C_f = surface skin friction shear force coefficient

S_x, S_y, S_z = direction cosine components of surface velocity vector

b = reference span (lateral and directional moment coefficient reference length)

\bar{c} = mean aerodynamic chord (for longitudinal moment reference)

x, y, z = distances from the center of gravity

= $x_{centroid} - x_{cg}$, etc.

The minus signs in the above equations are required because of the sign conventions on x and z in the body coordinate system (x positive forward, and z positive upward).

The total force and moment coefficients are obtained by summing the contributions of all the elements:

$$C_A = \sum \Delta C_A$$

$$C_Y = \sum \Delta C_Y$$

$$C_N = \sum \Delta C_N$$

$$C_l = \sum \Delta C_l$$

$$C_m = \sum \Delta C_m$$

$$C_D = \sum \Delta C_D$$

SECTION VIII

INVISCID PRESSURE METHODS

Many of the pressure calculation methods used in the analysis of high-speed shapes are listed in Figure 40. An attempt has been made in the preparation of this figure to indicate the interrelationships of the methods (the information can, of course, be organized in many different ways). Some of these methods are more applicable to the arbitrary-body problem than others.

The method of characteristics is the eventual ideal approach for the calculation of forces on three-dimensional shapes at high speeds. It will require starting solutions for three-dimensional blunt bodies of arbitrary shape. The development of a method of calculating three-dimensional boundary layers would permit the use of an iterative process to account for the viscous-inviscid interaction. Although this approach has been used for some very simple shapes, the complete solution for arbitrary shapes is some time away. Significant progress is also being made in the solution of the inviscid flow field by finite difference methods. However, present mathematical techniques and digital-computer size and speed capability must be reserved for simple shapes or important detail design applications where very large computer times might be acceptable.

Many of the other methods shown in Figure 40 would be useful force-calculation methods for inclusion in an arbitrary-body system. The selection of the proper method in a given application depends upon the vehicle-component shape and flight condition and must be selected by the engineer on the basis of his knowledge and experience in the use of each method.

Three basic paths of obtaining the inviscid pressure in the Mark IV program exist. The first path is the most frequently used and is the calculation of pressures by one of the simple impact or expansion pressure methods. These methods require impact angle, or a change

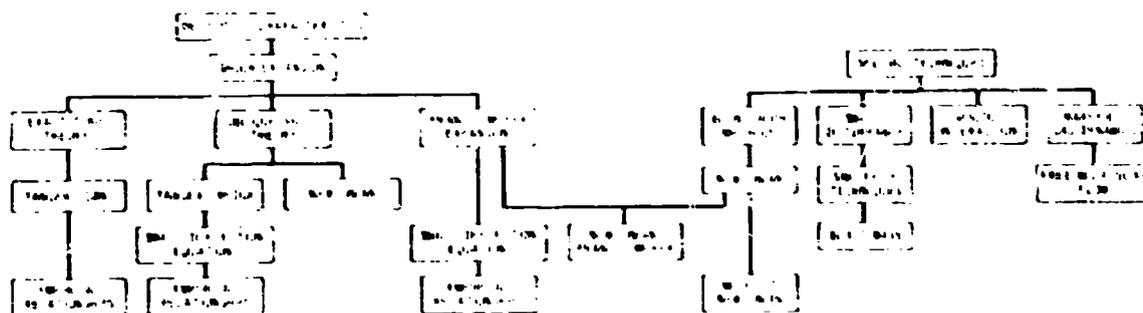


Figure 40. Pressure Calculation Methods

in angle of an element from a previous point and in some cases the free-stream Mach number (M_{∞}). These methods, in this mode, are used without any interference effects as in previous HABS programs. The next section presents a discussion of these simple impact type pressure calculation techniques.

The second path to obtaining pressures includes the calculation of interference effects from one component on another. This capability uses some of the simple impact methods but with local conditions determined from the flow field of another component. This method of obtaining inviscid pressures with corrections for these interference effects is discussed on page 114.

The final path for determining the inviscid pressures on the quadrilaterals of a shape is by means of interrogation of previously stored pressure data. This stored data may be either calculated or experimental results. The stored pressure information is not required at the quadrilateral centroids since interpolation of pressures can be accomplished. Pressures, once obtained at the centroid of each quadrilateral, are summed in the same manner as the previous two approaches to obtain final vehicle forces. A discussion of this method is presented at the last section.

BASIC PRESSURE CALCULATION METHODS

The arbitrary body force computer program contains a number of optional methods for calculating the pressure coefficient. In each method the only geometric parameter required is the element impact angle, δ , or the change in the angle from a previous element.

Before the program calculates the pressure on each surface element, it checks to see if the element is facing the flow (in an impact region) or facing away from the flow (in a shadow region). The methods to be used in calculating the pressure in impact and shadow regions may be specified independently. A summary of the program pressure options is presented below.

Basic Pressure Calculation Methods

Mark IV Mod 0 Program

<u>Impact Flow</u>	<u>Shadow Flow</u>
1. Modified Newtonian	1. Newtonian ($C_p = 0$)
2. Modified Newtonian+Prandtl-Meyer	2. Modified Newtonian+Prandtl-Meyer
3. Tangent wedge	3. Prandtl-Meyer from free-stream
4. Tangent-wedge empirical	4. Inclined cone
5. Tangent-cone	5. Van Dyke Unified
6. Inclined cone	6. High Mach base pressure
7. Van Dyke Unified	7. Shock-expansion
8. Blunt-body shear force	8. Input pressure coefficient
9. Shock-expansion	9. Free molecular flow
10. Free molecular flow	
11. Input pressure coefficient	
12. Hankey flat-surface empirical	
13. Delta wing empirical	
14. Modified Dahlem-Buck	
15. Blast wave	

Since most of these methods are adequately discussed in the literature they will be reviewed only briefly in this document. The blunt-body shear force and the boundary-layer induced pressure methods are discussed in detail in the section describing Viscous Force Methods.

Modified Newtonian

This method is probably the most widely used of all the hypersonic force analysis techniques. The major reason for this is its simplicity. Like all the force calculation methods, however, its validity in any particular application depends upon the flight condition and the shape of the vehicle or component being considered. Its most general application is for blunt shapes at high hypersonic speed. The usual form of the modified Newtonian pressure coefficient is

$$C_p = K \sin^2 \delta$$

In true Newtonian flow ($M = \infty$, $\gamma = 1$) the parameter K is taken as 2. In the various forms of modified Newtonian theory, K is given values other than 2 depending on the type of modified Newtonian theory used. K is frequently taken as being equal to the stagnation pressure coefficient. In other forms it is determined by the following relationship (Reference 36).

$$K = \frac{C_{p_{nose}}}{\sin^2 \delta_{nose}}$$

where

$C_{p_{nose}}$ = the exact value of the pressure coefficient at the nose or leading edge

δ_{nose} = impact angle at the nose or leading edge

In other work K is determined purely on an empirical basis.

$$K = f_n(M, \alpha, \text{shape})$$

When modified Newtonian theory is used, the pressure coefficient in shadow regions (δ is negative) is usually set equal to zero.

Modified Newtonian Plus Prandtl-Meyer

This method, described as the blunt body Newtonian + Prandtl-Meyer technique, is based on the analysis presented by Kaufman in Reference 21. The flow model used in this method assumes a blunt body with a detached shock, followed by an expansion around the body to supersonic conditions. This method uses a combination of modified Newtonian and Prandtl-Meyer expansion theory. Modified Newtonian theory is used along the body until a point is reached where both the pressure and the pressure gradients match those that would be calculated by a continuing Prandtl-Meyer expansion.

The calculation procedure derived for determining the pressure coefficient using the blunt body Newtonian + Prandtl-Meyer technique is outlined below.

1. Calculate freestream static to stagnation pressure ratio

$$P = \frac{P_{\infty}}{P_o} = \left[\frac{2}{(\gamma + 1) M_{\infty}^2} \right]^{\frac{\gamma}{\gamma - 1}} \left[\frac{2 + M_{\infty}^2 - (\gamma - 1)}{\gamma + 1} \right]^{\frac{1}{\gamma - 1}}$$

2. Assume a starting value of the matching Mach number, M_q (for $\gamma = 1.4$ assume $M_q = 1.35$)

3. Calculate matching point to free-stream static pressure ratio

$$Q = \frac{P_q}{P_o} = \left[\frac{2}{2 + (\gamma - 1) M_q^2} \right]^{\frac{\gamma}{\gamma - 1}}$$

4. Calculate new free-stream static to stagnation pressure ratio

$$P_c = Q \left[1 - \frac{\gamma^2 M_q^4 Q}{4(M_q^2 - 1)(1 - Q)} \right]$$

5. Assume a new matching point Mach number (1.75) and repeat the above steps to obtain a second set of data.

6. With the above two tries use a linear interpolation equation to estimate a new matching point Mach number. This process is repeated until the solution converges.

7. Calculate the surface slope at the matching point

$$\sin^2 \delta_q = \frac{Q - P}{T - P}$$

8. Use the Prandtl-Meyer expansion equations to find the Mach number on the surface element, M_δ

9. Calculate the surface pressure ratio

$$\frac{P_\delta}{P_o} = \eta_c \left[1 + \frac{\gamma - 1}{2} M_\delta^2 \right]^{-\frac{\gamma}{\gamma - 1}}$$

where

η_c is provided as an empirical correction factor

P_δ is the pressure on the element of interest

10. Calculate the surface to freestream pressure ratio

$$\frac{P_\delta}{P_\infty} = \left(\frac{1}{P} \right) \left(\frac{P_A}{P_o} \right)$$

11. Calculate the surface pressure coefficient

$$C_{P_\delta} = \frac{2}{\gamma M_\infty^2} \left(\frac{P_\delta}{P_\infty} - 1 \right)$$

The results of typical calculations using the above procedure are shown in Figure 41. Note that the calculations give a positive pressure coefficient at a zero impact angle. As pointed out in several references these results correlate well with test data for blunt shapes. However, if the surface curvature changes gradually to zero slope some distance from the blunt stagnation point the pressure calculated by this method will be too high. This is caused by characteristics near the nose intersecting the curved shock system and being reflected back onto the body. If the zero slope is reached near the nose (such as in a hemisphere or a cylinder) this effect has not had time to occur.

Tangent-wedge

The tangent-wedge and tangent-cone theories are frequently used to calculate the pressures on two-dimensional bodies and bodies of

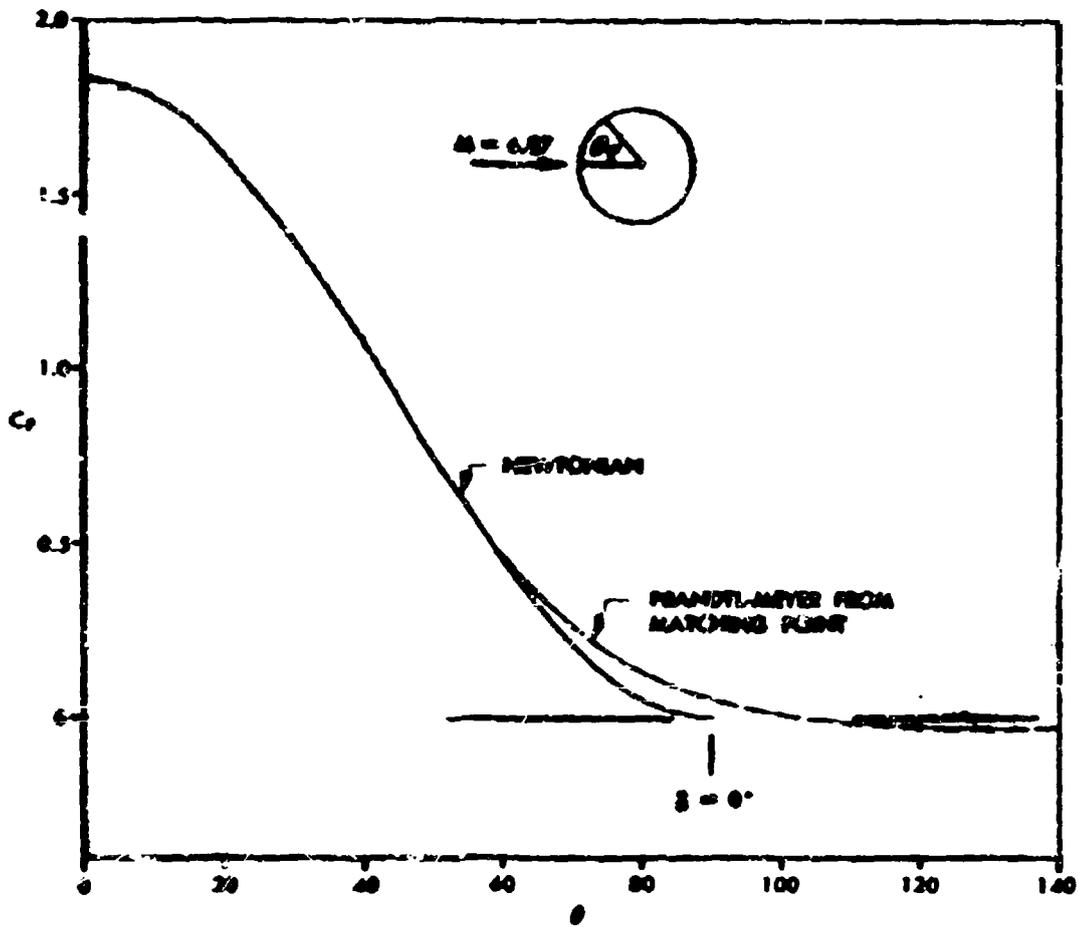


Figure 41. Blunt Body Newtonian + Prandtl-Meyer Pressure Results

revolution, respectively. These methods are really empirical in nature since they have no firm theoretical basis. They are suggested, however, by the results of more exact theories that show that the pressure on a surface in impact flow is primarily a function of the local impact angle. In this program the tangent-wedge pressures are calculated using the oblique shock relationships of NACA TR-1135 (Reference 22). The basic equation used is the cubic given by

$$(\sin^2 \theta_s)^3 + b(\sin^2 \theta_s)^2 + c(\sin^2 \theta_s) + d = 0 \quad \text{or}$$

$$R^3 + b R^2 + c R + d = 0$$

where

$$\theta_s = \text{shock angle}$$

$$\delta = \text{wedge angle}$$

$$b = -\frac{M^2 + 2}{M^2} - \gamma \sin^2 \delta$$

$$c = \frac{2M^2 + 1}{M^4} + \left[\frac{(\gamma + 1)^2}{4} + \frac{\gamma - 1}{M^2} \right] \sin^2 \delta$$

$$d = -\frac{\cos^2 \delta}{M^4}$$

The roots of the above cubic equation may be obtained by using the trigonometric solution procedure (see Reference 64) as indicated below.

$$y_1 = 2 \sqrt{-p/3} \cos(-/3) - b/3$$

$$y_2 = -2 \sqrt{-p/3} \cos(-/3 + 60^\circ) - b/3$$

$$y_3 = -2 \sqrt{-p/3} \cos(-/3 - 60^\circ) - b/3$$

$$R_1 = y_1 - b/3$$

$$R_2 = y_2 - b/3$$

$$R_3 = y_3 - b/3$$

where

y_i = roots of the reduced cubic equation

$$p = -\frac{b^2}{3} + c$$

$$q = 2\left(\frac{b}{3}\right)^3 - \frac{bc}{3} + d$$

$$\cos \omega = -\frac{q}{2\sqrt{-(p/3)^3}}$$

R_i = $\sin^2 \theta_s$ = roots of the cubic equation

The smallest of the three roots corresponds to a decrease in entropy and is disregarded. The largest root is also disregarded since it never appears in physical actuality.

For small deflections, the cubic solution becomes very sensitive to numerical accuracy; that is, to the number of significant digits carried. Since this is dependent on the particular machine employed, an alternate procedure is used.

When the flow deflection angle is equal to or less than 2.0 degrees, the following equation is used instead of the above cubic relationships (Reference 23):

$$\sin^2 \theta_s = \frac{L}{M^2} + \frac{\gamma+1}{2} \frac{\gamma-1}{\gamma M^2 - 1} \delta$$

Once the shock angle is obtained the remaining flow properties may be found from the relationships of Reference 22.

$$\text{density} = \rho_2 = \rho \left[\frac{(\gamma+1) M^2 \sin^2 \theta_s}{(\gamma-1) M^2 \sin^2 \theta_s + 2} \right]$$

$$\text{temperature} = T_2 = T \left[\frac{[2\gamma M^2 \sin^2 \theta_s - (\gamma-1)] [(\gamma-1) M^2 \sin^2 \theta_s + 2]}{(\gamma+1)^2 M^2 \sin^2 \theta_s} \right]$$

$$\text{pressure coefficient} = C_p = \frac{4(M^2 \sin^2 \theta_s - 1)}{(\gamma + 1) M^2}$$

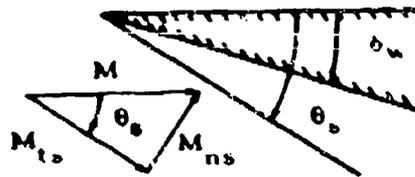
where

()₂ = conditions behind the shock

Oblique shock detachment conditions are reached when no solution may be found to the above cubic relationships. Under these conditions the program uses the Newtonian + Prandtl-Meyer method for continued calculations.

Tangent-Wedge and Delta Wing Newtonian Empirical Method

The tangent wedge Newtonian empirical method and the tangent-cone method used in the Delta Wing empirical method are based on the empirical relationships below.



For wedge flow

$$\sin \theta_s = \frac{\sin \delta_w}{(1 - \epsilon) \cos (\theta_s - \delta_w)}$$

where

$$\epsilon = \frac{\rho}{\rho_2} = \frac{\gamma - 1}{\gamma + 1} \left[1 + \frac{2}{(\gamma - 1) M_{ns}^2} \right]$$

For cone flow (thin shock layer assumption)

$$\sin \theta_s \approx \frac{\sin \delta_c}{\left(1 - \frac{\epsilon}{2}\right) \cos (\theta_s - \delta_c)}$$

In the limit as $M \rightarrow \infty$, $\epsilon = \epsilon_{\text{lim}} = \frac{\gamma - 1}{\gamma + 1}$ and $\cos (\theta_s - \delta) = 1$

Therefore

$$\begin{array}{cc} \text{wedge} & \text{cone} \\ \sin \theta_s = \frac{\gamma + 1}{2} \sin \delta_w & \sin \theta_s = \frac{2(\gamma + 1)}{\gamma + 3} \sin \delta_c \end{array}$$

These limiting expressions for θ may now be compared with the data of TR-1135 (Reference 22) at $\gamma = 7/5$ using the following similarity parameters. The exact equations contain three variables - θ_s , δ , and ϵ . Noting that for $\gamma = \text{constant}$, $\epsilon = \text{fn}(M_{ns})$ only, the preceding equations may be rewritten in the following form:

$$\begin{array}{cc} \text{wedge} & \text{cone} \\ M_{ns} = \frac{M \sin \delta_w}{\left(1 - \epsilon\right) \cos (\theta_s - \delta_w)} & M_{ns} = \frac{M \sin \delta_c}{\left(1 - \frac{\epsilon}{2}\right) \cos (\theta_s - \delta_c)} \end{array}$$

The parameter $(\theta - \delta)$ is approximately constant and independent of M except near the shock detachment condition. The equations essentially contain only two variables, M_{ns} and $M \sin \delta$. These are used as coordinates to plot the data for wedge flow shown in Figure 42. A similar plot could be obtained for cone flow. From the figure it is seen that the data are nearly normalized with the use of these coordinates.

For rapid calculations we need relationships for M_{ns} as a function of $M \sin \delta$ that satisfy the following requirements:

1. The effect of shock detachment is neglected
2. At $M \sin \delta = 0$, $M_{ns} = 1$
3. The solution asymptotically approaches the $M = \infty$ line
4. Have the correct slope, $\frac{d M_{ns}}{d M \sin \delta}$ at $M \sin \delta = 0$

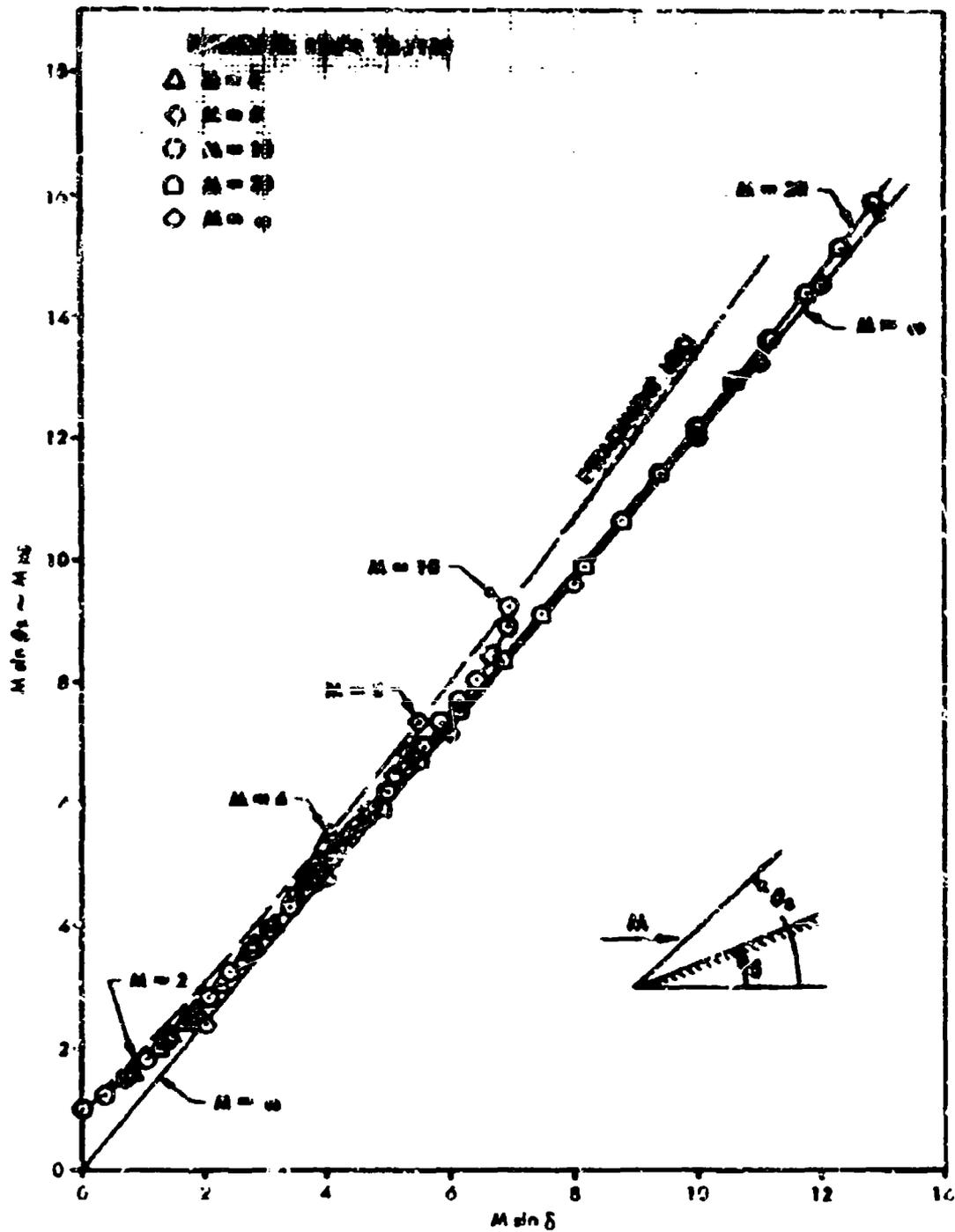


Figure 42. Wedge Flow Shock Angle

These conditions lead to equations of the following form

$$\text{wedge } M_{ns} = K_w M' + e^{-\frac{K_w}{2} M'}$$

$$K_w = \frac{\gamma + 1}{2}$$

$$\text{cone } M_{ns} = K_c M' + e^{-K_c M'}$$

where

$$M' = M \sin \delta$$

$$K_c = 2(\gamma + 1)/(\gamma + 3)$$

These expressions are compared with the data of TR-1135 in Figures 43 and 44. The cone data are also shown in Figure 45 with the same scales as in Figure 42.

The pressure coefficient may now be obtained by the following relationships for a wedge and cone respectively.

$$C_p = \left(\frac{4}{\gamma + 1} \right) (M_{ns}^2 - 1) / M^2$$

$$C_p = 2 \sin^2 \delta \left[1 - \frac{(\gamma - 1) M_{ns}^2 + 2}{4(\gamma + 1) M_{ns}^2} \right]$$

Experimental results have shown the pressure on the centerline of a delta wing to be in agreement with two-dimensional theory at small values of the similarity parameter ($M' < 3.0$) and the conical flow theory at higher values. The previous expressions derived for wedge and cone flows have been combined to give these features. The resulting relationships are given below.

$$M_{ns} = K_c M' + e^{-\left(K_c - \frac{K_w}{2} \right) M'}$$

For $\gamma = 7/5$

$$M_{ns} = 1.09 M \sin \delta + e^{-0.49 M \sin \delta}$$

The similarity parameter relationship for pressure is

$$M^2 C_p = \left(\frac{4}{\gamma + 1} \right) (M_{ns}^2 - 1)$$

The shock angle and pressure coefficient calculated from the above equations are compared with the experimental results (Reference 24) in Figures 46 and 47, respectively.

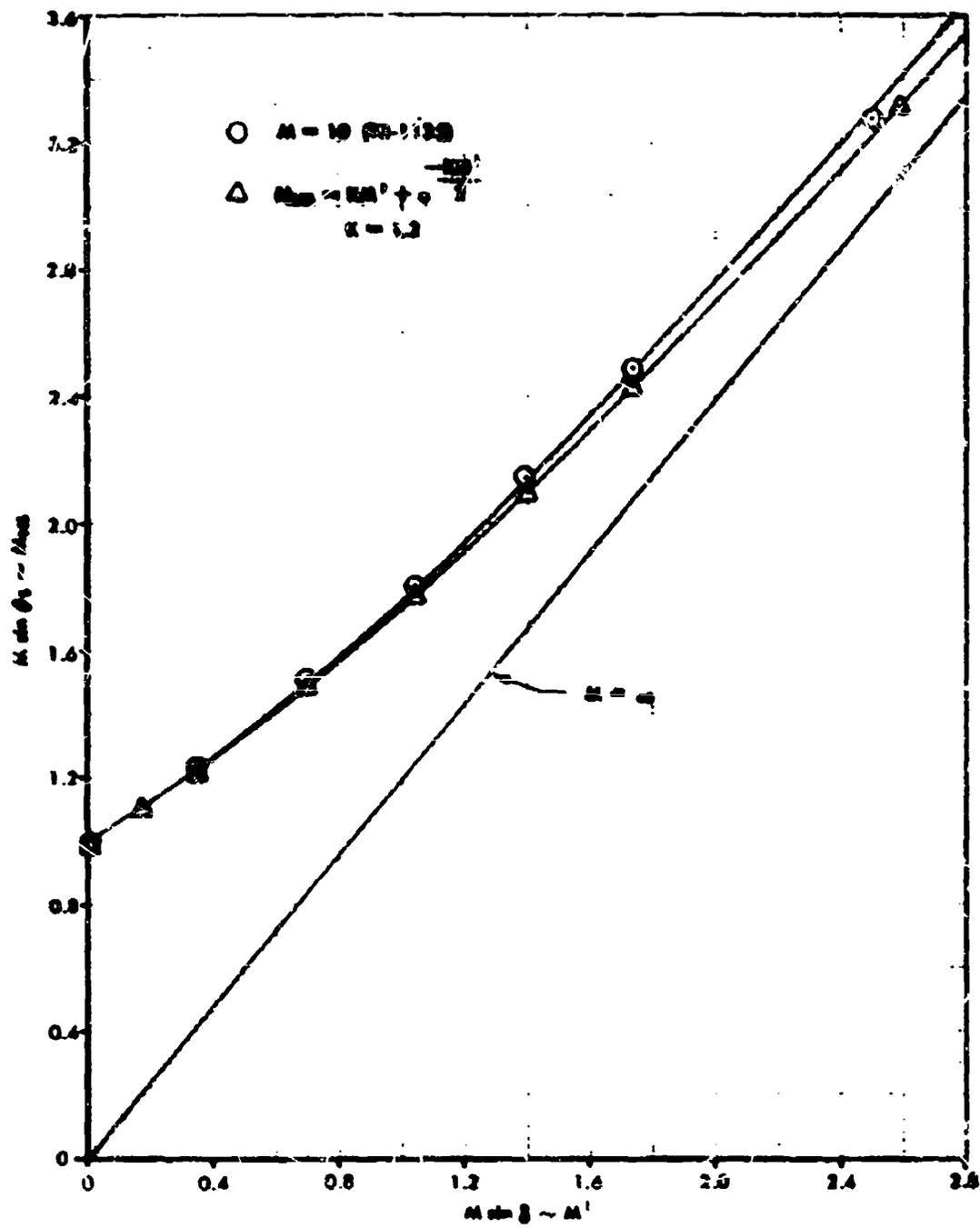


Figure 43. Wedge Flow Shock Angle Empirical Correlation

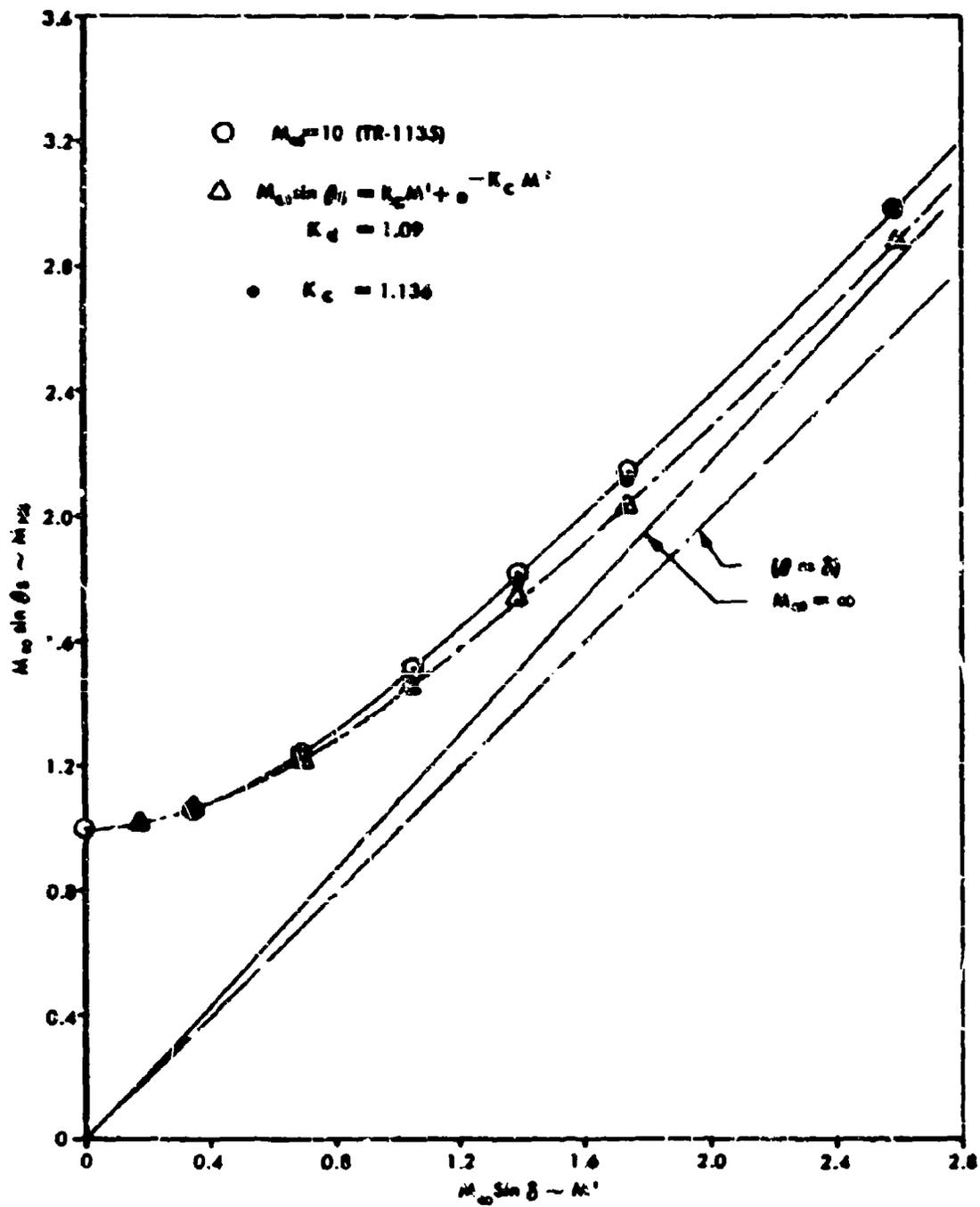


Figure 44. Conical Flow Shock Angle Empirical Correlation

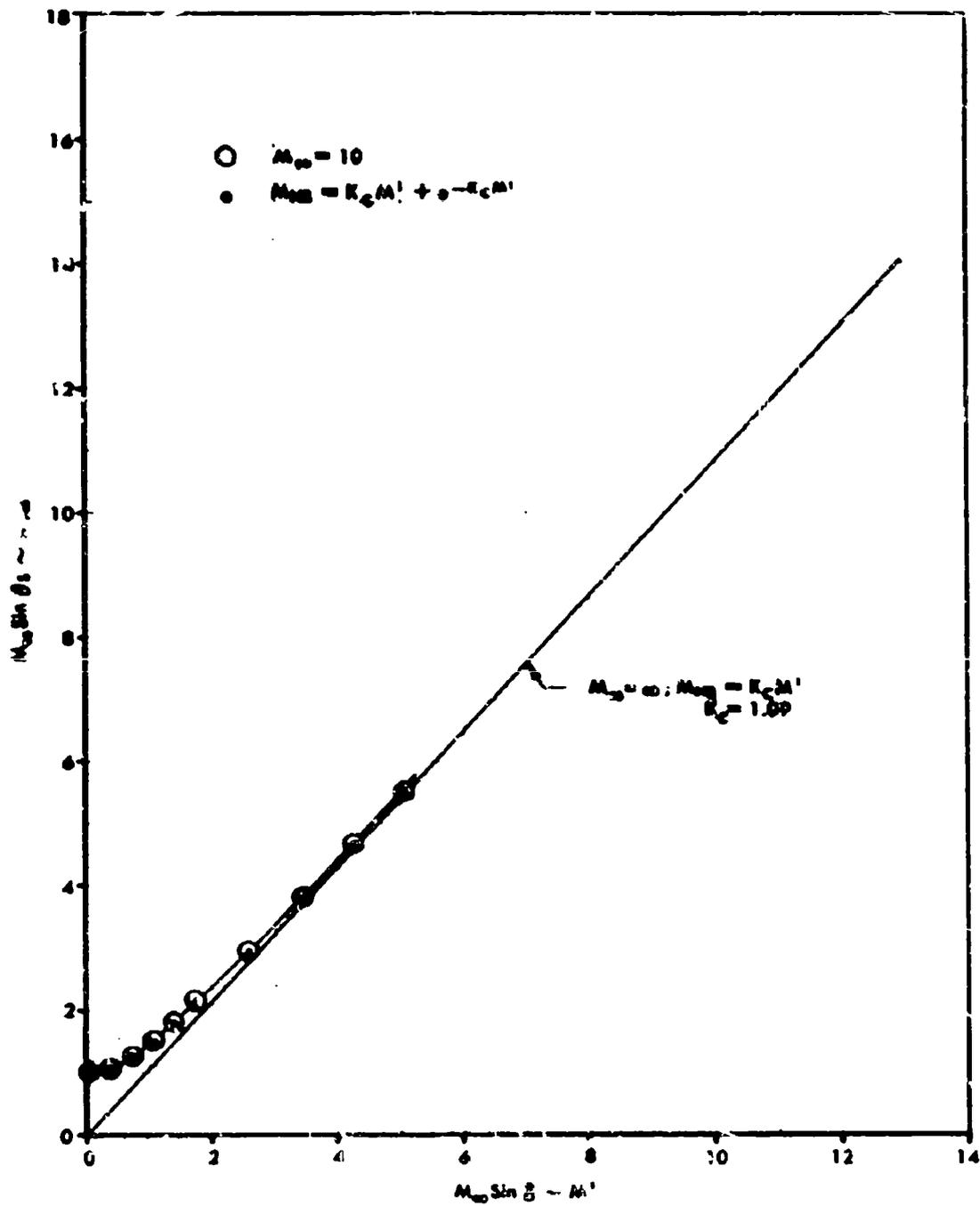


Figure 45. Conical Flow Shock Angle Empirical Correlation

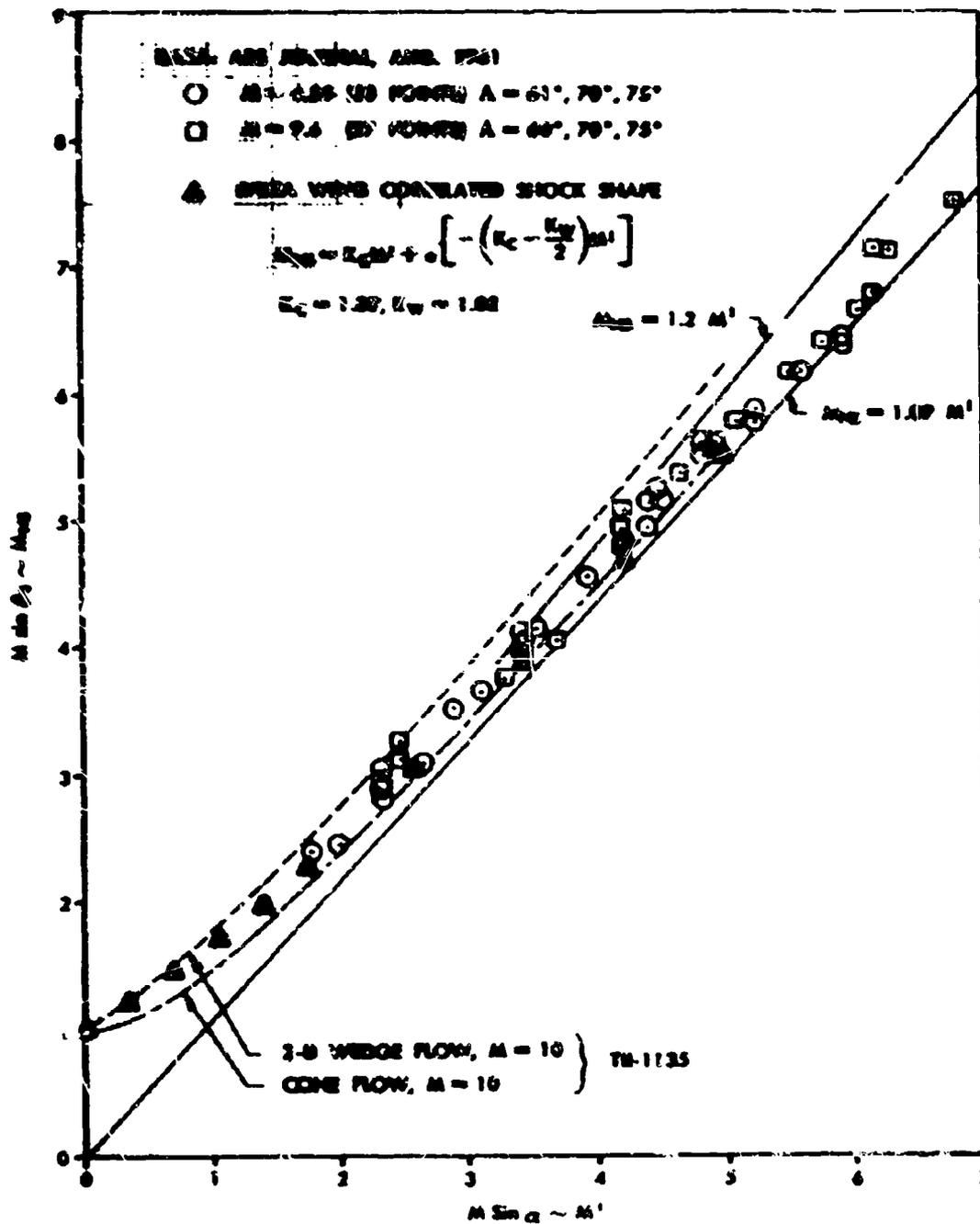


Figure 46. Delta Wing Centerline Shock Angle Correlation

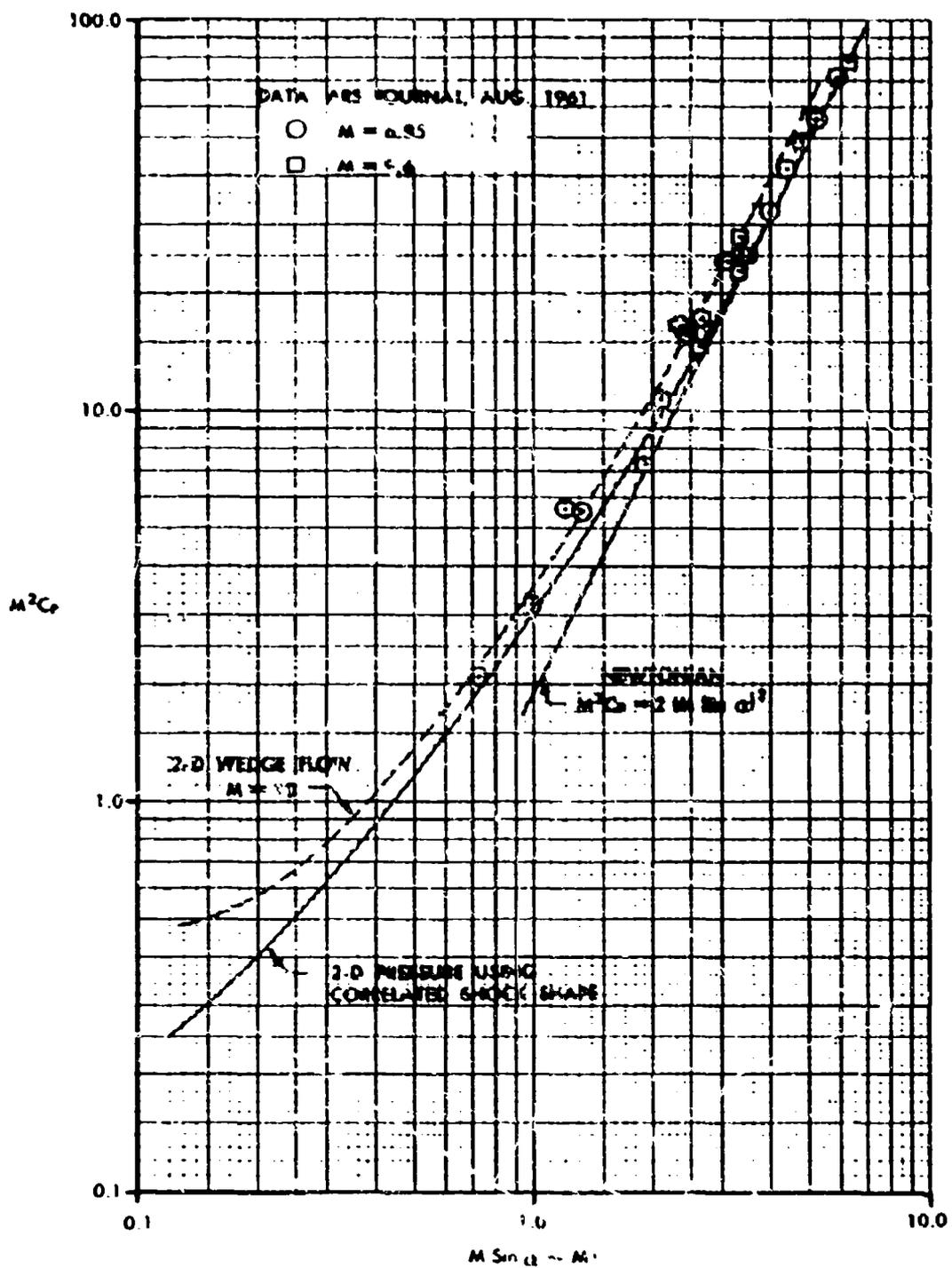


Figure 47. Delta Wing Centerline Pressure Coefficient Correlation

Tangent Cone

An approximate solution for predicting the surface flow conditions on a cone in supersonic flow has been devised applicable to the entire Mach number - cone angle regime for an attached bow-wave. Detailed comparison with exact results show improved accuracy over currently available approximations. This is a new cone method and should not be confused with the empirical tangent cone method just discussed on the previous pages and which was used in the old Mark III program.

Basis for this new method is the combining of two approximate techniques, one yielding accurate results in the low supersonic range and the other in the high supersonic range, by the use of transition functions defined in terms of the appropriate similarity variables to provide uniformly valid solutions over the entire speed range. Specifically, second-order slender-body theory (Reference 25) is used for small values of the unified similarity parameter and the approximate solution of Hammit and Murthy (Reference 26) for large values.

The surface pressure and conical shock-wave angle are determined which, together with the assumption of an ideal gas with constant ratio of specific heats, are sufficient to calculate all the surface flow variables. It should also be mentioned that the present solution does not require multiple integration of the differential equations across the flow field, but is obtained by direct algebraic solution providing results rapidly.

The quantities of direct importance to the Supersonic-Hypersonic Arbitrary-Body Aerodynamic Computer Program are the surface pressure coefficient and Mach number. The calculated pressure coefficients have been compared to exact results and, for Mach numbers greater than 2, the maximum error is less than 1 percent and in the hypersonic speed range the average error is of the order of 0.25%. The accuracy of the predicted surface Mach numbers is extremely good (the order of 0.30 percent maximum error) throughout the speed range, except as bow-wave detachment is reached. For the present purposes, the extreme conditions correspond to surface Mach number equal to 1.0 and a comparison with exact results showed good agreement.

For reference, the present method has been compared with Schwartz's formula for pressure coefficient (Reference 27). The percent relative error in C_p is given in Table 1 using Schwartz's formula and in Table 2 using the Douglas method (the tabulated values have been rounded to the nearest integer percent). For completeness, the relative error in surface Mach number and cone shock angle using the Douglas method are shown in Tables 3 and 4, respectively (the Schwartz formula is only for C_p). The exact values (subscript ex) were obtained from Reference 20.

Table 1

Percent Relative Error in C_p (Schwartz)

$$100 \times \left[\frac{C_{p_{ex}} - C_p}{C_{p_{ex}}} \right]$$

Cone Angle	Mach Number					
	1.5	2	4	6	10	20
5	0	0	0	0	0	0
10	2	1	0	0	1	0
15	3	1	0	1	1	1
25	8	3	1	1	0	0
35	-	6	2	1	0	4
40	-	-	3	1	0	7

Table 2

Percent Relative Error in C_p (Douglas)

$$100 \times \left[\frac{C_{p_{ex}} - C_p}{C_{p_{ex}}} \right]$$

Cone Angle	Mach Number					
	1.5	2	4	6	10	20
5	0	0	0	0	-1	0
10	0	0	-1	-1	0	0
15	-3	0	-1	0	0	0
20	-6	0	0	0	0	0
25	-4	-2	0	0	0	0
30	-	-2	0	0	0	0

Table 3

Percent Relative Error in Surface Mach Number

$$100 \times \left[\frac{M_{ex} - M}{M_{ex}} \right]$$

Cone Angle	Freestream Mach Number					
	1.5	2	4	6	10	20
5	0	0	0	0	0	0
10	0	0	0	0	0	0
15	1	0	0	0	0	0
20	2	0	0	0	0	0
25	1	1	0	0	0	0
30	-	1	0	0	0	0

Table 4

Percent Relative Error in Shock Angle

$$100 \times \left[\frac{\theta_{sex} - \theta_s}{\theta_{sex}} \right]$$

Cone Angle	Freestream Mach Number			
	2	4	10	20
5	-2	-3	-3	-1
10	-1	-3	-1	0
20	-2	-1	0	0
30	-1	-1	0	0
40	-	0	0	0
50	-	-1	0	0

Inclined Cone

A method for predicting the pressure distributions on circular cones at angle-of-attack has been put together based on the British work of Reference 28. For convenience, this will be referred to as the CP#792 method. The cited reference presents the development of this method in ample detail and only the essential features will be described herein, along with several modifications that have been made. The original method was compared with experimental results for ten different cases and have been rerun using the Douglas version. In addition, Jones' pressure formula (Reference 29) has been compared with these cases. Both methods are in good agreement with the data.

Method of CP #792

Briefly, the CP #792 method is an ingenious extension of simple impact theory:

$$C_p = K \cdot \sin^2 \delta$$

where K is a suitable impact coefficient and δ is the impact angle. The impact angle for a cone is easily expressed in terms of the cone angle θ , angle of attack α , and circumferential angle ϕ (measured from most windward generator):

$$\sin \delta = \sin \theta \cos \alpha + \cos \theta \sin \alpha \cos \phi$$

Substituting this into the C_p equation, the pressure coefficient may be expressed as the sum of three terms:

$$C_p = C_{pA} + C_{pN} + C_{pX}$$

where

$$C_{pA} = K \sin^2 \theta \cos^2 \alpha$$

$$C_{pN} = K \cos^2 \theta \sin^2 \alpha \cos^2 \phi$$

$$C_{pX} = K (2 \sin \theta \cos \theta \sin \alpha \cos \alpha \cos \phi)$$

These three terms lend themselves to the following physical interpretation. C_{pA} is that part of the total which can be regarded as being generated by the axial flow component, $M_{\infty} \cos \alpha$. C_{pN} is that part of the total which can be considered as being generated by the normal or transverse flow component, $M_{\infty} \sin \alpha$. Finally, C_{pX} is a cross-product term which can be regarded as arising from the interaction between the axial and normal flow components.

The authors of CP #792 make the assumption that the three components can at all times be treated independently of each other and then proceed to develop coefficients K_A , K_N , and K_X suitable for each flow. This is the crux of the whole method. The Mark IV application retains only the K_X term and calculates the axial and normal components directly.

Axial Component C_{pA}

At $\alpha = 0$, $C_{pA} = K \sin^2 \theta$ which is simply the pressure coefficient for a cone (C_{pC}) at $M_A = M_\infty \cos \alpha$. Since the total C_p is based on free-stream dynamic pressure, the cone value must be adjusted by the q -ratio:

$$C_{pA} = C_{pC} q_A / q_\infty = C_{pC} \cos^2 \alpha$$

Where $C_{pC} = f(M_A)$ and is obtained by use of the previously discussed tangent cone method.

Normal Component C_{pN}

The development of C_{pN} is analogous to C_{pA} . Namely, when the flow is all normal ($\alpha = 90^\circ$) the pressure coefficient on the windward generator is simply the stagnation value C_{pS} at $M_N = M_\infty \sin \alpha$. The circumferential variation is simply taken as $\cos^2 \phi$ up to $\phi = 90^\circ$ and zero thereafter.

$$C_{pN} = C_{pS} \cos^2 \phi q_N / q_\infty = C_{pS} \cos^2 \phi \sin^2 \alpha$$

where

$$C_{pS} = f(M_N)$$

Cross-Product Component C_{pX}

This term is used in the original form:

$$C_{pX} = K_X \cdot 2 \sin \theta \cos \theta \sin \alpha \cos \phi$$

A correlation curve for K_X was derived in CP #792 by consideration of the results obtained from small incidence theory. This curve is approximated in the Mark IV application as

$$K_X = 1.95 + 0.07 \cdot \cos (X)$$

where

$$X = \frac{\pi}{2 \beta \sin \theta \cos \theta}$$

and

$$\beta = \sqrt{M_\infty^2 - 1}$$

Jones' Formula

An alternate method for obtaining C_p on circular cones at angle-of-attack has been presented by Jones (Reference 29). It is based on a least squares curve fit to the extensive tabulation of exact numerical solutions which were also compiled by Jones (Reference 20). This method is easy to use and has been shown to have acceptable accuracy relative to the exact solutions.

$$C_p = C_{p_{\alpha=0}} + \left[A_1 T + \frac{A_2 T}{M_{\infty}^2} + \frac{A_3}{M_{\infty}^2} \right] (\alpha/\theta) \\ + \left[A_4 T + \frac{A_5 T}{M_{\infty}^2} + \frac{A_6}{M_{\infty}^2} \right] (\alpha/\theta)^2$$

where

$$T = (\sin B\theta) (\cos \theta)$$

and

$$A_i = a_{0i} + a_{1i} \cos \phi + a_{2i} \cos 2\phi$$

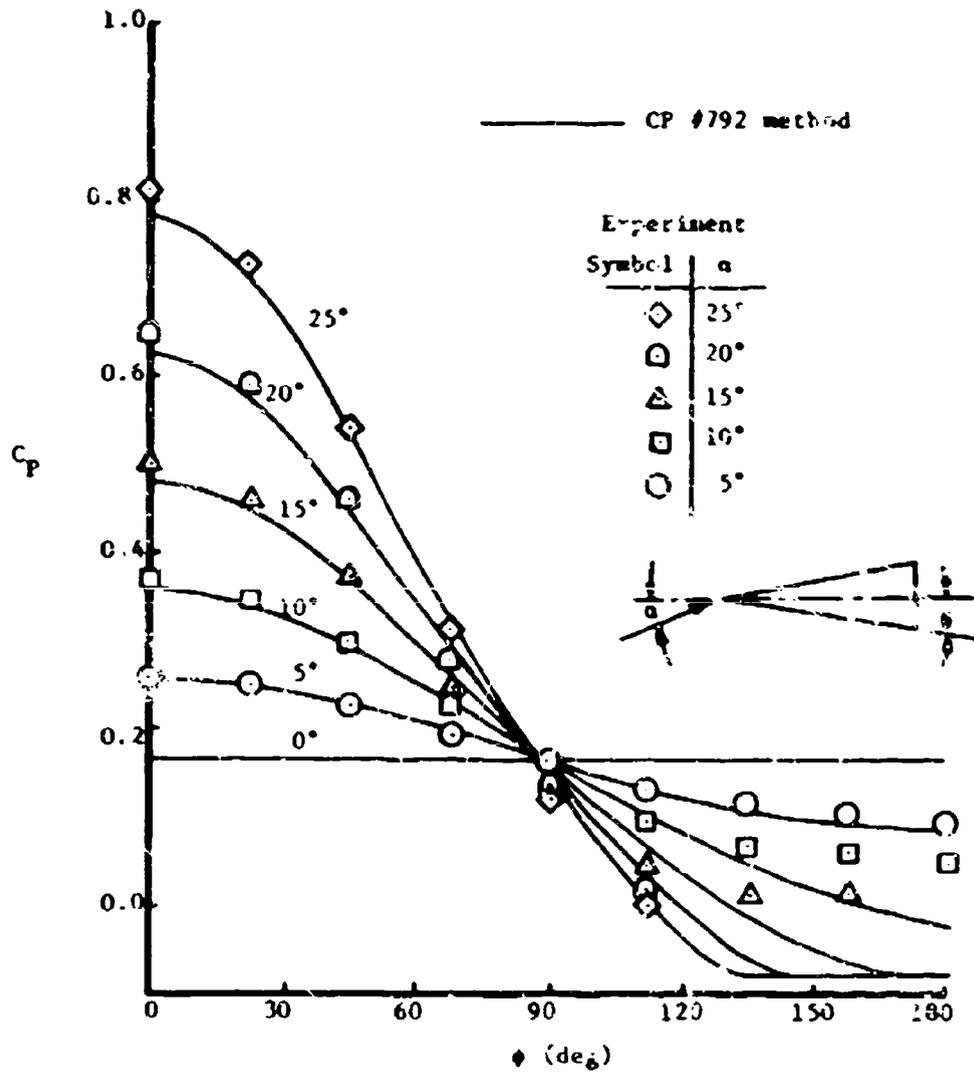
The coefficients B and a_{ji} have been determined by a parametric least squares fit.

$C_{p_{\alpha=0}}$ is obtained using the tangent cone method (inpaint method No. 5).

Comparison with Experiment

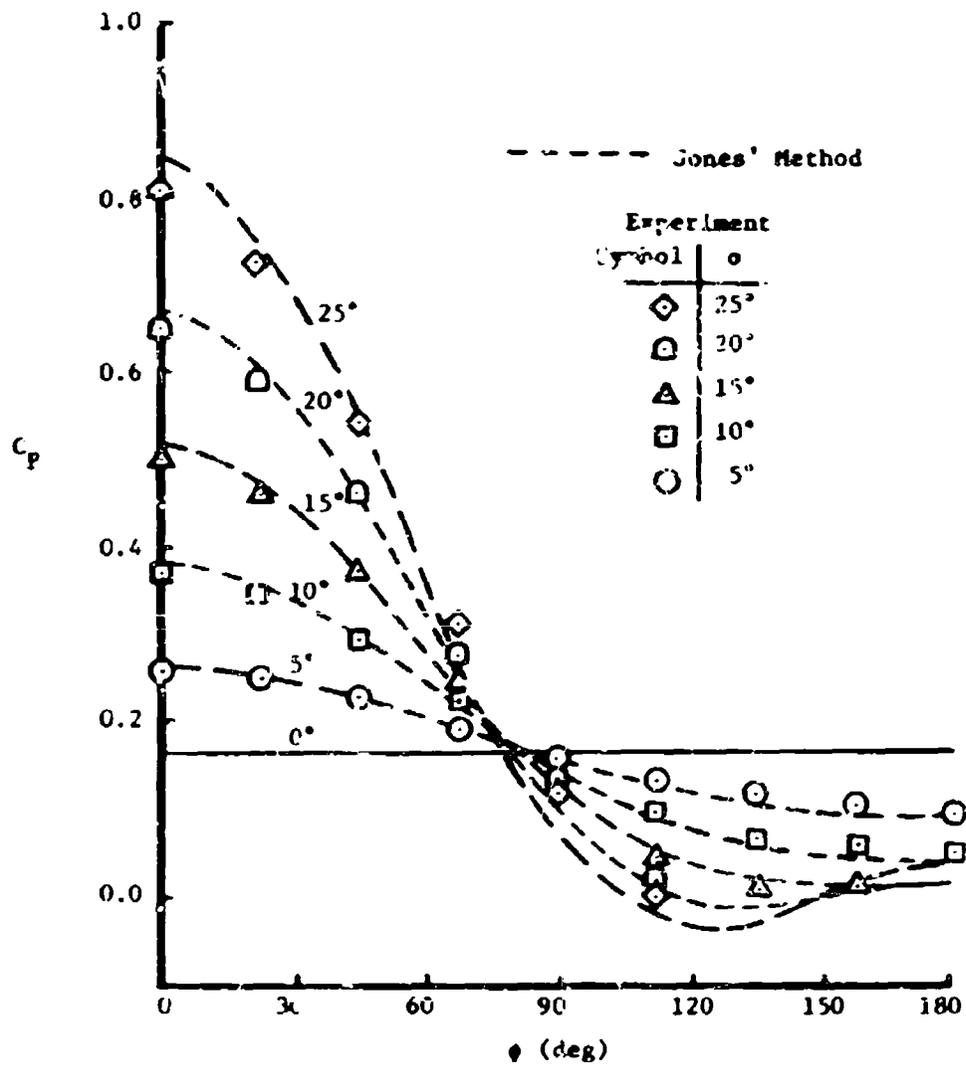
Both the Douglas version of the CP #792 method and Jones' formula have been compared with the ten experimental cases given in Reference 28. Four of these cases (two plots each) are included as representative of the results obtained. On Figures 43 to 51, a and b each, comparison between predicted and experimental pressure coefficients are given. Each figure shows, for particular values of cone angle and Mach number, pressure coefficient versus circumference angle for a range of angles of attack. The predicted results of CP #792 are indicated by full lines on the a-set of figures. Jones' results are given by broken lines on the b-set of figures.

Both methods are in good agreement with the data. Jones' formula is much better on the leeward side (e. g., Figure 49 (b)) and CP #792 is better on the windward side. Jones' formula was derived for $\theta \leq 25^\circ$ and relative incidence $\alpha/\theta \leq 1.0$ and in the present comparisons appears to extrapolate reasonably well for $\alpha/\theta > 1.0$. The CP #792 method was not intended to work on the leeward surface, but the results are not that bad.



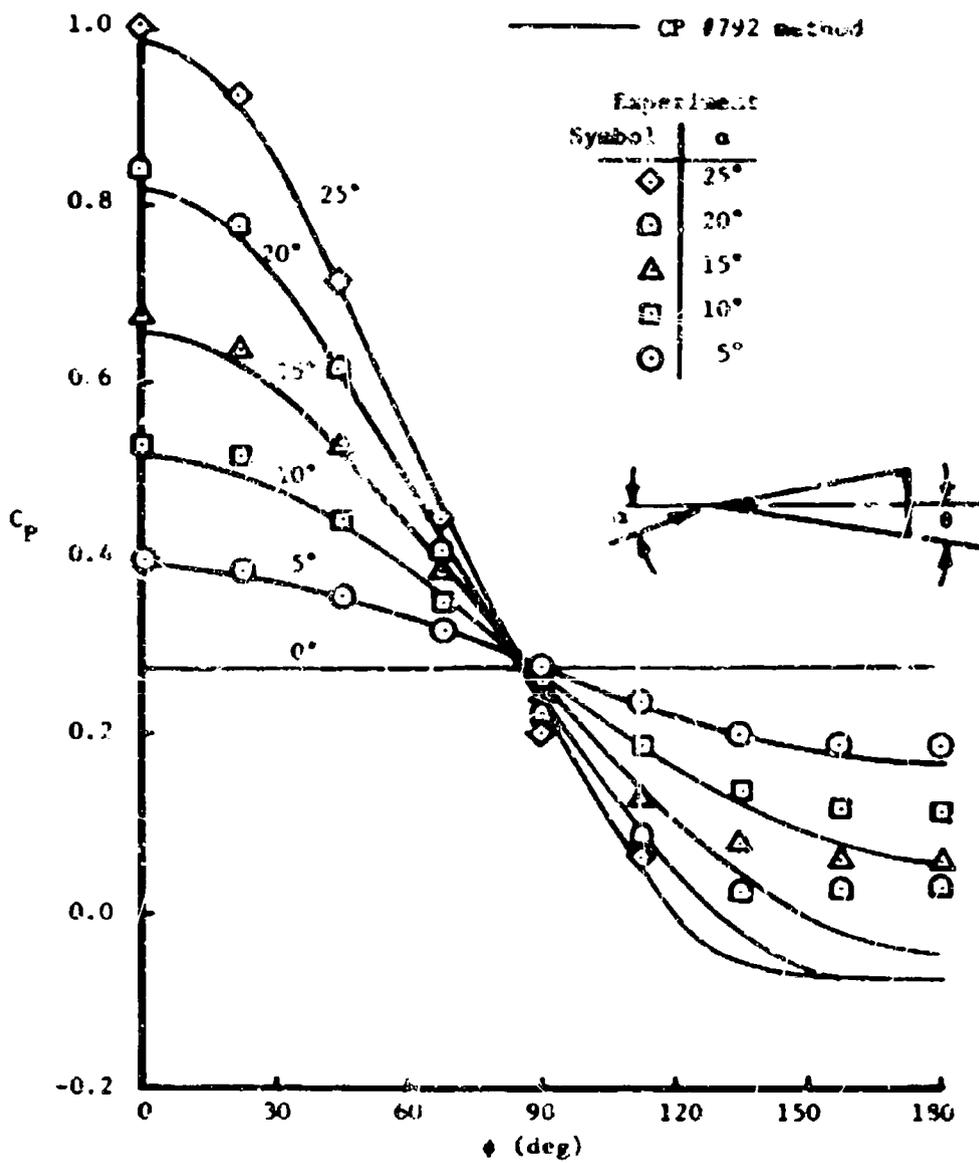
(a) CP #792 Method

Figure 48. Comparison Between Measured and Predicted Cone Pressure, $M_\infty = 3.53$, $\theta = 15^\circ$



(b) Jones' Method

Figure 48. Continued



(a) CP 792 Method

Figure 49. Comparison Between Measured and Predicted Cone Pressures; $M_\infty = 3.53$, $\theta = 20^\circ$

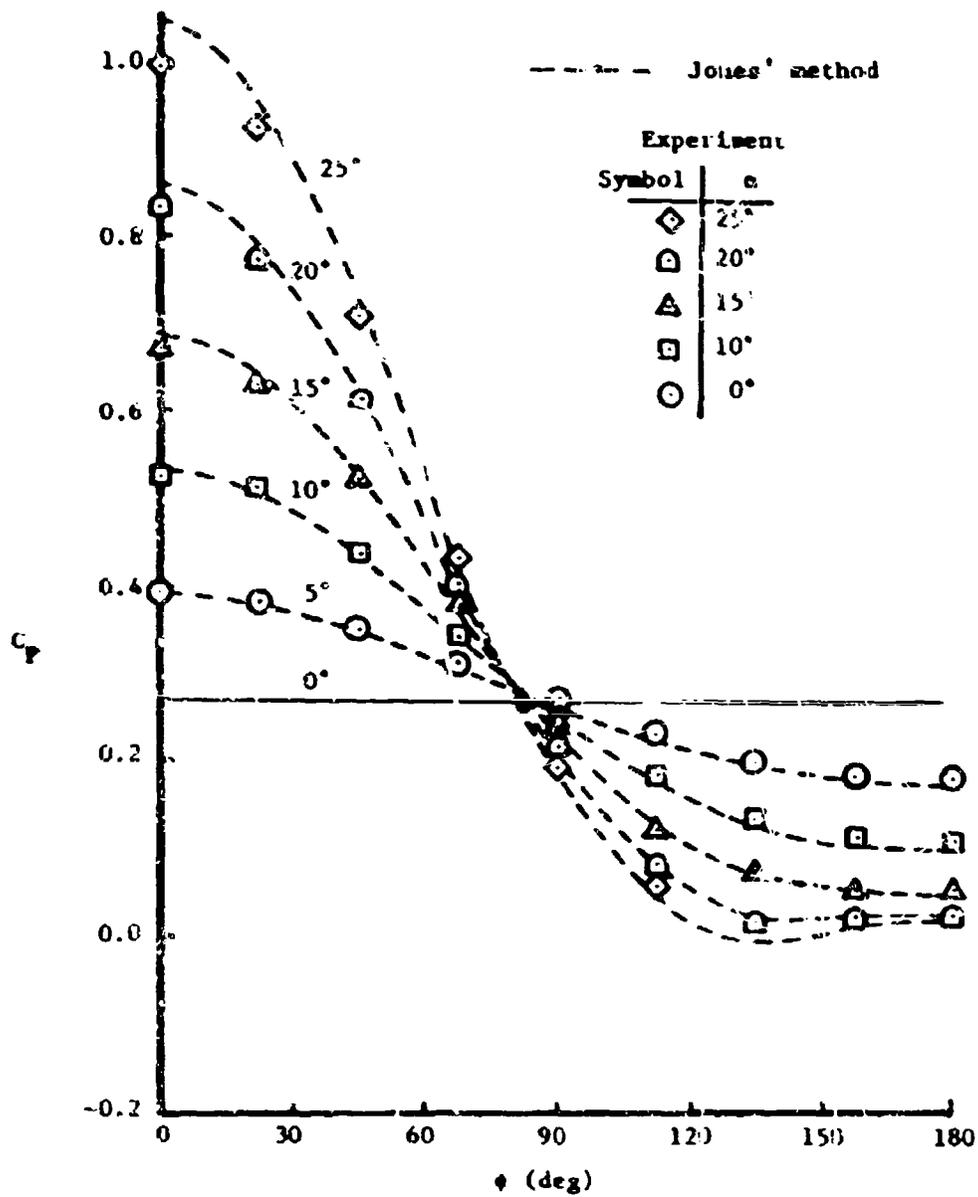
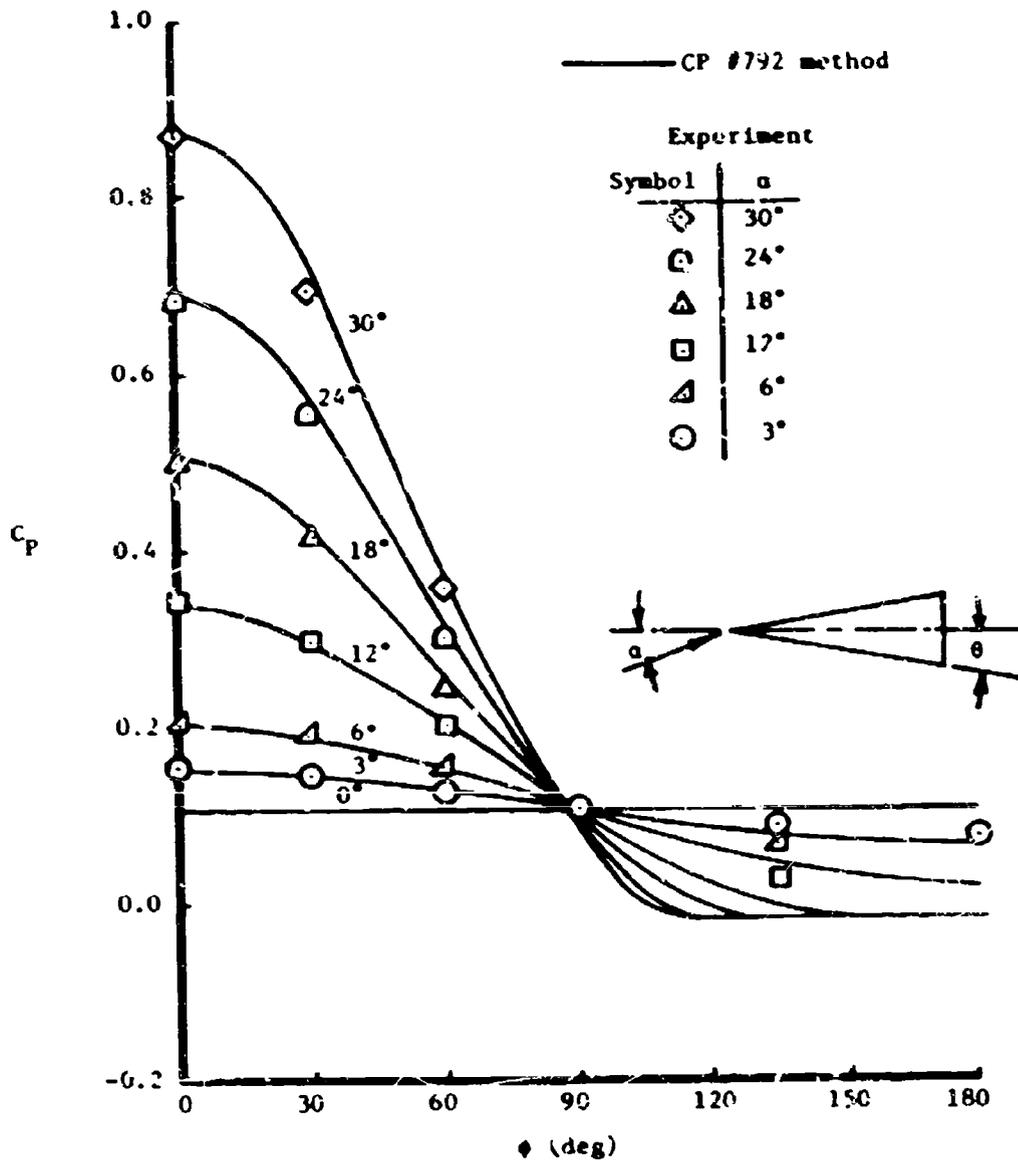
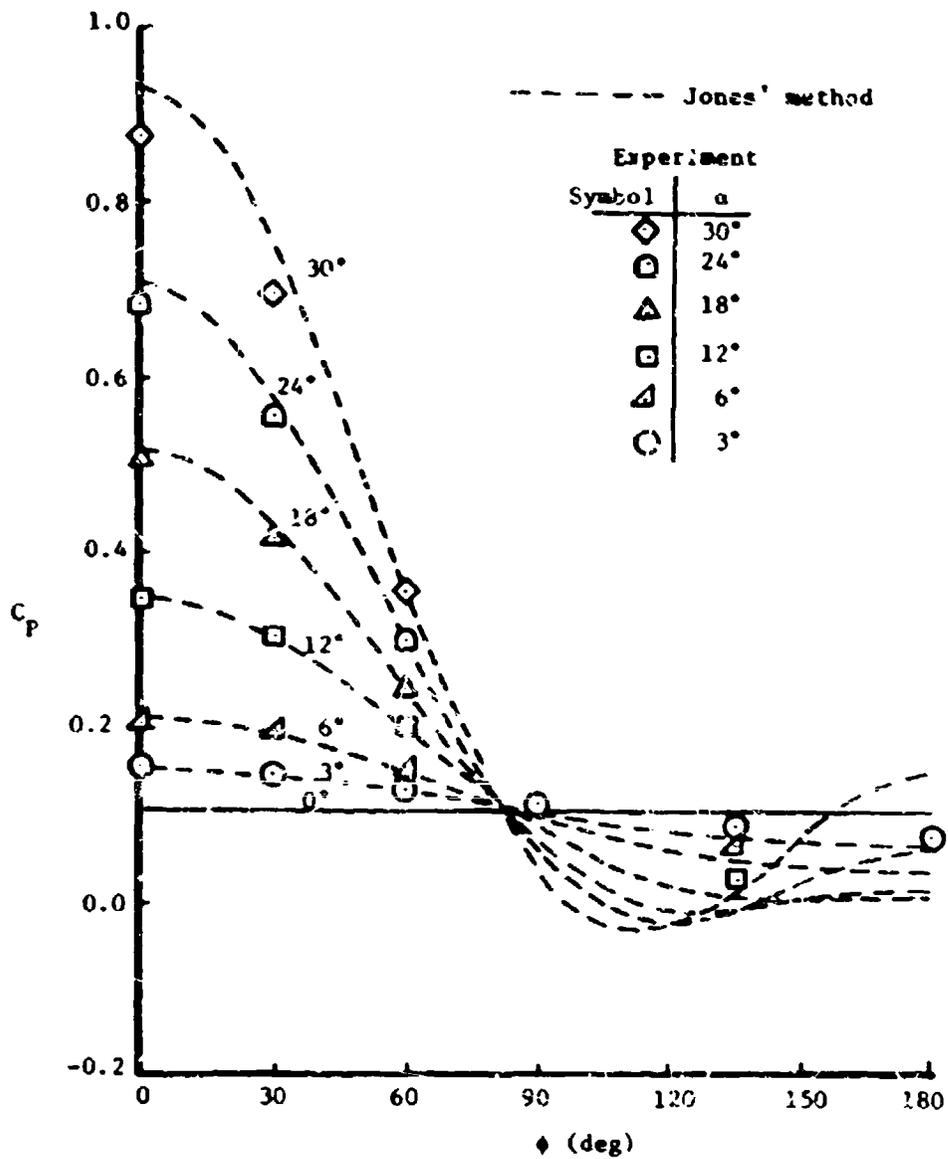


Figure 49. - Continued.



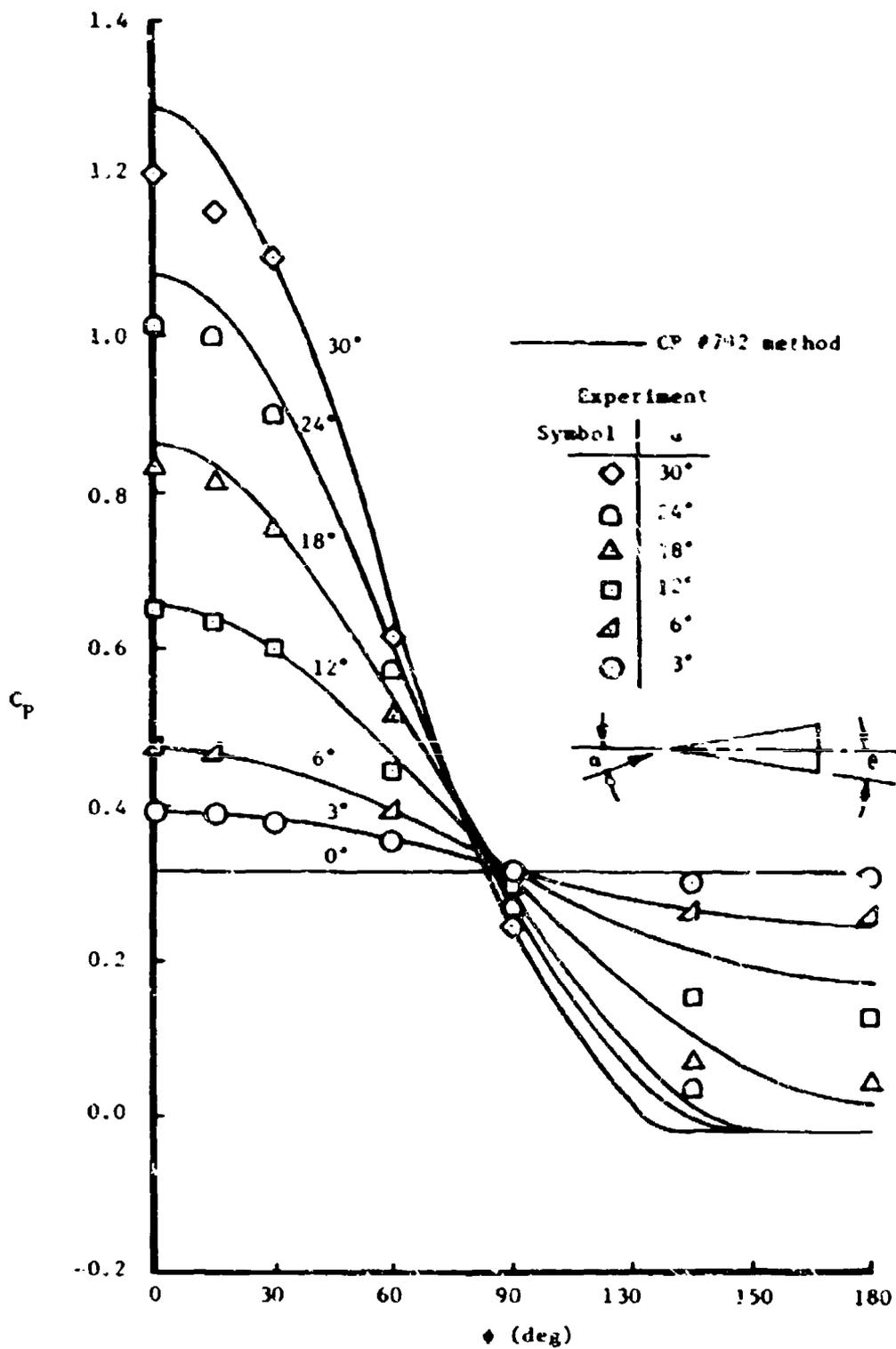
(a) CP #792 Method

Figure 50. Comparison Between Measured and Predicted Conc Pressures; $M_\alpha = 6.85$, $\theta = 12.5^\circ$



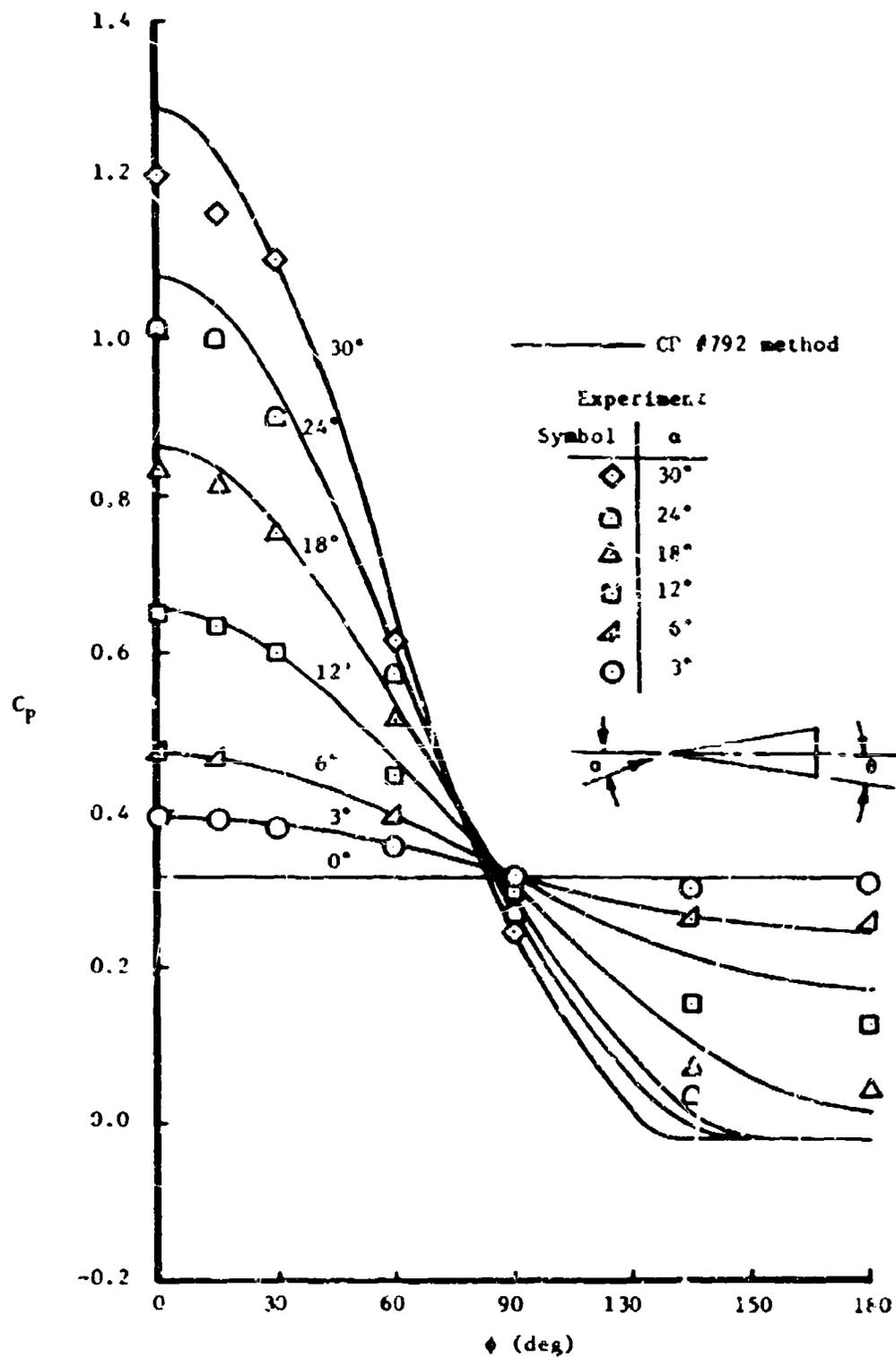
(b) Jones' Method

Figure 50. - Continued.



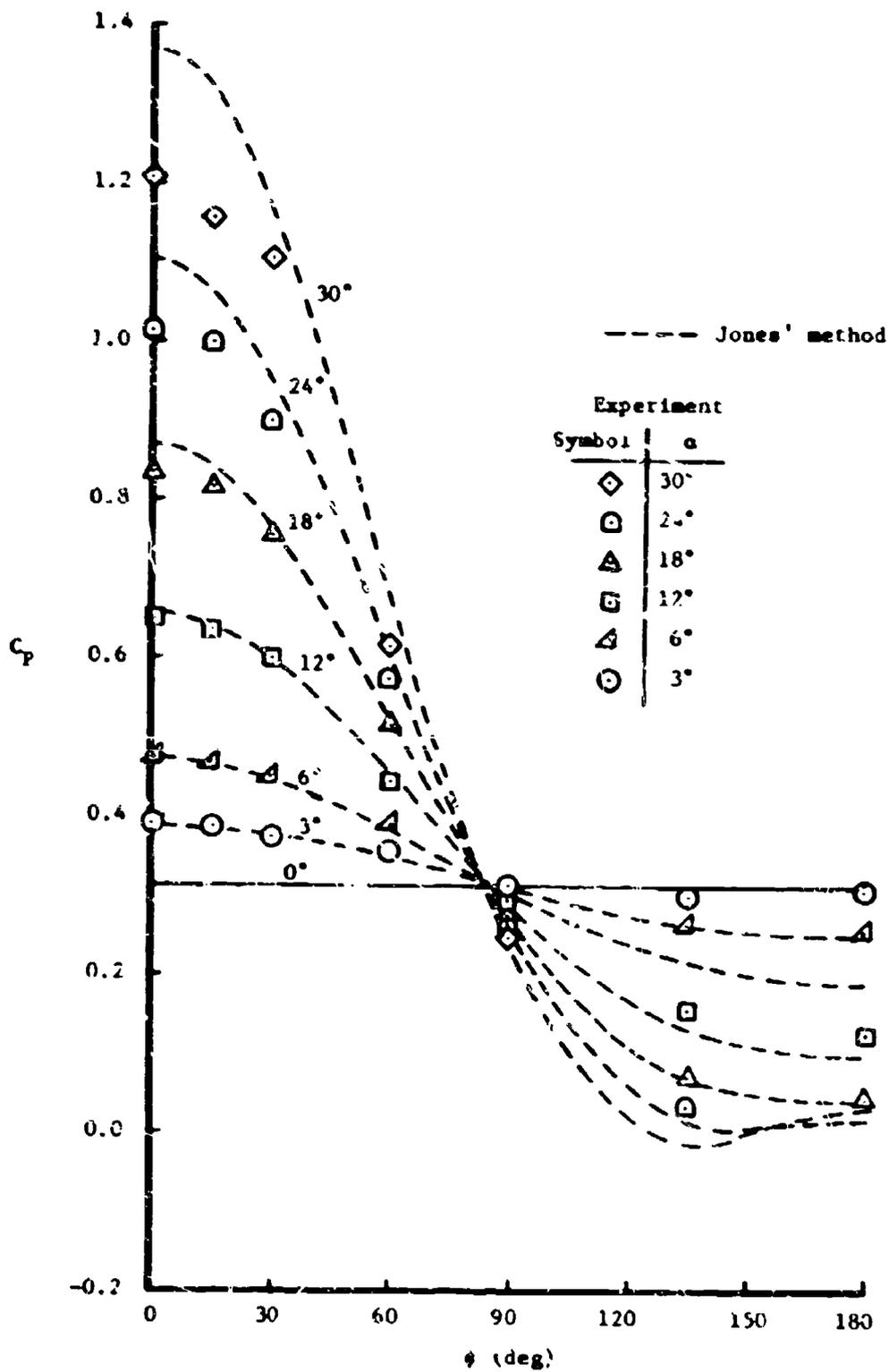
(a) CP #792 Method

Figure 51. Comparison Between Measured and Predicted Cone Pressures: $M_{\infty} = 6.85$, $\theta = 22.5^\circ$



(a) CP #792 Method

Figure 51. Comparison Between Measured and Predicted Cone Pressures; $M_{\infty} = 6.85$, $\theta = 22.5^\circ$

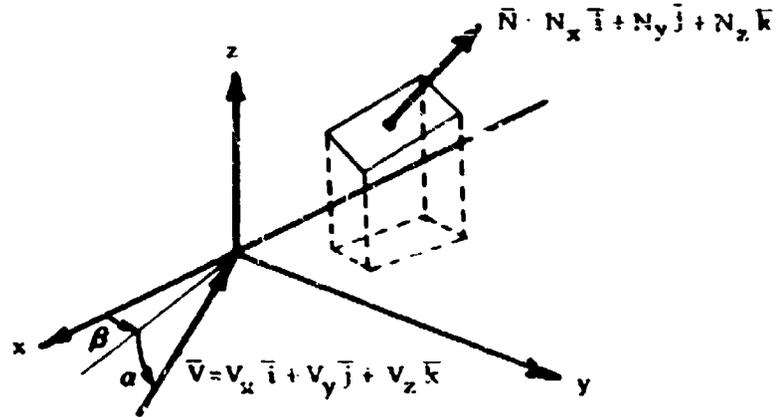


(b) Jones' Method

Figure 51. - Continued.

Use as an Element Impact Method

As an element impact method, the cone angle, meridian angle, and angle of attack are defined by the element normal and the velocity vector as follows (see sketch).

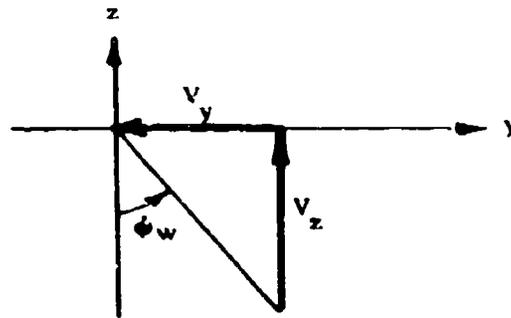


The cone semi-vertex angle, θ , is defined as the angle between the surface and the x axis.

$$\theta = \arcsin (\bar{N} \cdot \bar{i}) = \arcsin (N_x)$$

The meridian angle and angle of attack have to be defined relative to the windward plane. The meridian angle location of the windward plane is

$$\phi_w = \arctan (-V_y/V_z)$$



The meridian angle of the element is

$$\phi_e = \arctan (-N_y/N_z)$$

The meridian angle of the cone relative to the windward plane is then

$$\phi = \phi_e - \phi_w$$

The angle of attack of the cone in the windward plane is

$$\alpha = \arccos (-V_x)$$

The Mark IV program uses Jones' Formula as the inclined cone method for both impact and shadow flows. For the Second-Order Shock, Expansion flow field and pressure option, either the CP #792 or Jones method can be used by use of an input flag.

Van Dyke Unified Method

This force calculation method is based on the unified supersonic-hypersonic small disturbance theory proposed by Van Dyke in Reference 30 as applied to basic hypersonic similarity results. The method is useful for thin profile shapes and as the name implies extends down to the supersonic speed region.

The similarity equations that form the basis of this method are derived by manipulating the oblique shock relations for hypersonic flow. The basic derivations are shown on pages 753 and 754 of Reference 31. The result obtained for a compression surface under the assumption of a small deflection angle and large Mach number is (hypersonic similarity equation),

$$C_p = \delta^2 \left[\frac{\gamma + 1}{2} + \sqrt{\left(\frac{\gamma + 1}{2}\right)^2 + \frac{4}{H^2}} \right]$$

where δ is the hypersonic similarity parameter given by $M\delta$. The contribution by Van Dyke in Reference 30 suggests that this relationship will also be valid in the realm of supersonic linear theory if the hypersonic similarity parameter $M\delta$ is replaced by the unified supersonic-hypersonic parameter $(\sqrt{M^2 - 1})\delta$. This latter parameter is used in the calculations for this force option in the arbitrary body program.

A similar method may also be obtained for a surface in expansion flow with no leading edge shock such as on the upper side of an airfoil. The resulting equation is

$$C_p = \delta^2 \frac{2}{\gamma H^2} \left[\left(1 - \frac{\gamma - 1}{2} H\right)^{\frac{2\gamma}{\gamma - 1}} - 1 \right]$$

where again H is taken to be $(\sqrt{M^2 - 1})\delta$ in the unified theory approach.

Shock-Expansion Method

This force calculation method is based on classical shock-expansion theory (see Reference 31). In this method the surface elements are handled in a "strip-theory" manner. The characteristics of the first element of each longitudinal strip of elements may be calculated by oblique shock theory, by conical flow theory, the delta wing empirical method or by a Prandtl-Meyer expansion. Downstream of this initial element the forces are calculated by a Prandtl-Meyer expansion.

By a proper selection of the element orientation the method may be used for both wing-like shapes and for more complex body shapes. In this latter case the method operates in a hypersonic shock-expansion theory mode.

Free Molecular Flow Method

At very high altitudes conventional continuum flow theories fail and one must begin to consider the general macroscopic mass, force, and energy transfer problem at the body surface. This condition occurs when the air is sufficiently rarefied so that the mean free path of the molecules is much greater than a characteristic body dimension. This condition is known as free molecular flow and the method of analysis selected for this program is described in Reference 32. This method was also used in Reference 33. The equations used were taken from these references and are presented below.

Pressure Coefficient

$$C_p = \frac{1}{S^2} \left\{ \left[\frac{2 - f_n}{\sqrt{\pi}} S \sin \delta + \frac{f_n}{2} \sqrt{\frac{T_b}{T_\infty}} \right] e^{-(S \sin \delta)^2} + \left[(2 - f_n) (S^2 \sin^2 \delta + \frac{1}{2}) + \frac{f_n}{2} \sqrt{\pi \frac{T_b}{T_\infty}} S \sin \delta \right] [1 + \operatorname{erf}(S \sin \delta)] \right\}$$

Shear Force Coefficient

$$C_f = \frac{(\cos \delta) f_t}{\sqrt{\pi} S} \left\{ e^{-(S \sin \delta)^2} + \sqrt{\pi} S \sin \delta [1 + \operatorname{erf}(S \sin \delta)] \right\}$$

where

$$S = \text{speed ratio} = \sqrt{\gamma/2} M_\infty$$

$$f_n = \text{normal momentum accommodation coefficient (=0.0 for Newtonian and =1.0 for completely diffuse reflection)}$$

$$\delta = \text{impact angle}$$

$$T_b = \text{body temperature, } ^\circ\text{K}$$

$$T_\infty = \text{freestream temperature, } ^\circ\text{K}$$

$$\operatorname{erf} = \text{error function } \operatorname{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-x^2} dx$$

$$f_t = \text{tangential momentum accommodation coefficient (=0.0 for Newtonian flow and 1.0 for completely diffuse reflection)}$$

The analysis to determine the direction of the shear force is discussed in Section VII.

The final components of the shear force in the vehicle axis system are given by

$$\text{SHEAR}_X = (\text{SHEAR})(S_X) / \text{STOTAL}$$

$$\text{SHEAR}_Y = (\text{SHEAR})(S_Y) / \text{STOTAL}$$

$$\text{SHEAR}_Z = (\text{SHEAR})(S_Z) / \text{STOTAL}$$

where

SHEAR is the shear force as calculated by the free molecular flow equations.

$$\text{STOTAL} = (S_X^2 + S_Y^2 + S_Z^2)^{1/2}$$

In using the free molecular flow method the above analysis must be carried out over the entire surface of the shape including the base, shadow regions, etc. When the free molecular flow method is selected, it is used for both impact and shadow region.

This method of determining the shear direction is also used for the continuum viscous forces discussed in Section X. The plane formed by the velocity vector and the surface normal is referred to as the velocity plane, since both the incident and surface velocity are in this plane. This definition is correct for two-dimensional flow, however, it is only an approximation to the shear direction in the general arbitrary-body case.

Hankey Flat-Surface Empirical Method

This method uses an empirical correlation for lower surface pressures on blunted flat plates. The method, derived in Reference 34, approximates tangent-wedge at low impact angles and approaches Newtonian at high impact angles. The pressure coefficient is given by

$$C_p = 1.95 \sin^2 \delta + 0.21 \cos \delta \sin \delta$$

Modified Dahlem-Buck

This is an extended form of the Dahlem-Buck method derived in Reference 35. The original method uses an empirical relationship which approximates tangent cone pressures at low impact angles and approaches Newtonian values at the large impact angles. The original equations are

$$\text{for } \delta \leq 22.5^\circ \quad C_{PDB} = \left[\frac{1.0}{\sin(4\delta)^{3/4}} + 1.0 \right] \sin^2 \delta$$

$$\text{for } \delta > 22.5^\circ \quad C_{PDB} = 2 \sin^2 \delta$$

If, at small values of δ , the bracketed term exceeds 5.0 it is set at 5.0

The original Dahlem-Buck method has been shown to yield good agreement for highly swept shapes at large hypersonic Mach numbers.

The modified Dahlem-Buck method is an extension by its originators to lower Mach numbers (see Reference 37). It was assumed that the empirical pressure coefficient would be affected by a change in Mach number in the same way as the pressure coefficient on the surface of a right circular cone is affected by Mach number. Thus,

$$C_{PMDB} = C_{PDB} \cdot \frac{C_{p_{cone}}(M < 20)}{C_{p_{cone}}(M = 20)}$$

where C_{PMDB} is the modified Dahlem-Buck pressure coefficient.

The data of Reference 37 for cone half angles from 10 to 30 degrees was analyzed and it was found that the quantity $(C_{p_{cone}}(M < 20)/C_{p_{cone}}(M = 20))^{-1.0}$ could be graphed as a straight line on a logarithmic scale for the mentioned cone angles. A curve fit allowed the cone pressure coefficient fraction to be analytically defined such that,

$$\frac{C_{p_{cone}}(M < 20)}{C_{p_{cone}}(M = 20)} - 1.0 = a\delta^n$$

where δ is impact angle in degrees and

$$a = (6.0 - 0.3M_\infty) + \sin\left(\frac{\ln(M_\infty) - 0.588}{1.20} \pi\right)$$

$$-n = 1.15 + 0.5 \sin\left(\frac{\ln(M_\infty) - 0.916}{3.29} \pi\right)$$

Blast Wave Pressure Increments

This method uses conventional blast-wave parameters to calculate the over-pressure due to bluntness effects. Contributions determined by this procedure must be added to the regular inviscid pressure forces (tangent-wedge, tangent-cone, Newtonian, etc.) calculated over the same vehicle geometry. The specific blast wave solutions used in the Program were derived by Lukasiwicz in Reference 38:

$$\frac{P}{P_\infty} = A M_\infty^2 \left\{ \frac{1}{(C_D)^{1+j}} \right\}^{\frac{2+j}{3}} \cdot B$$

where

- C_D is the nose drag coefficient
- d is the nose diameter or thickness
- X_0 is a coordinate reference point

and the coefficients A, B are

Flow	j	A	B
Two-Dimensional	0	0.121	0.56
Axisymmetric	1	0.067	0.44

THE STORED PRESSURE OPTION

The Stored Pressure Option Cards are used when the vehicle component forces are to be calculated using pressure data previously stored on the flow field data unit 10. This option may be used in several ways. For example, the forces on a particular component may be calculated using experimental results which have been previously stored on unit 10 by use of the Flow Field Data Hand-Load Option. More directly, forces may be determined using the data generated by the Shock-Expansion Flow Field Option and stored on unit 10.

In either case, pressure data is available at a limited number of discrete locations on the component. The function of the Input Pressure Option is to obtain the value of pressure at the centroid of each element. This is accomplished by interpolation using the Surface Spline Method and, as in the other applications of this method, proper normalization of the coordinates is required to obtain meaningful results. The forces are calculated by summing the contributions of all the elements that make up the component. A detailed discussion of the Surface Spline and the related normalization procedure can be found in Section IV.

When the Stored Pressure Option is used in connection with experimental pressure data any incorrect or spurious data should be removed beforehand. Data of this type, when the Surface Spline is used, can yield faulty pressure interpolations. The resulting errors in the curve fit may not be confined to the local region of the questionable data points. Another important point to be considered when utilizing this option is that of embedded shocks. Interpolation across strong shock boundaries by use of the Surface Spline is not recommended. Each major flow field contained on a surface should be split into primary and secondary (embedded flows). Output from the Shock Expansion Flow Field Option is in the required form. Any experimental data inputs must also be of the same form if meaningful interpolations of surface data are to be obtained.

SECTION IX

STREAMLINE CALCULATIONS

In steady flow, a streamline is a path or trajectory of a fluid particle. The calculation of a streamline is then a trajectory problem and the classical Runge-Kutta method may be used to solve the defining differential equations. What is needed are definitions of the body and its velocity field and in the present application to arbitrary bodies no convenient analytical forms exist.

The approach taken in the Mark IV program is to use the Surface Spline to define the body and the velocity field. This is the key to the whole method and in effect provides the necessary analytical forms. The Surface Spline is an "interpolation-in-the-large" scheme and should not be applied indiscriminantly to an arbitrary body. Rather, it should be used in separate applications to the various components and panels that make up the vehicle. This is analogous with the basic philosophy of using the best pressure method suitable to a particular panel. In addition, meaningful results from the Surface Spline can only be obtained by using appropriate coordinates with suitable normalization. These statements, while distinctly restrictive in tone, are intended only to convey a realistic perspective of the program capabilities.

In practice the calculations are not nearly so restrictive, as several modes of coordinate selection and normalization are automatically available through input flags. The following paragraphs discuss the basic equations and present results verifying the general approach taken in the solution.

Mathematically, the streamline may be defined as follows:

$$\frac{dx}{dt} = V_x, \quad \frac{dy}{dt} = V_y, \quad \frac{dz}{dt} = V_z$$

where V_i is the surface velocity component in the i -direction. These are normally calculated in the FORCE routine and assume a Newtonian impact type of surface velocity (see the Surface Velocity Vector discussion on page 11). Using the definition of the total velocity, $V_T = ds/dt$, the above equations may be re-written in terms of the particle path:

$$\frac{dx}{ds} = C_x, \quad \frac{dy}{ds} = C_y, \quad \frac{dz}{ds} = C_z$$

where C_i is the direction cosine of the i -th velocity component.

The streamlines are calculated by specifying an increment in path, Δs , and marching along using a fourth-order Runge-Kutta scheme to solve the differential equations. At this stage, several options are available which are more or less dependent on the form of both the body definition and the

C_i field. For example, all three coordinates could be solved for independently. While a streamline will be found, there is no guarantee that it will be on the surface. This problem has been experienced by other investigators and may be avoided by solving the equations in parametric form. That is, two of the streamline coordinates (x, y) are found using Runge-Kutta and the third is obtained as a function of these two from the surface definition: $z = F(x, y)$. This is equivalent to solving for the projection of the streamline path in the x, y plane, Figure 52. The choice of independent coordinates (x, y) is not arbitrary, but must be suitable for the particular panel geometry being considered. For example, the reference coordinates of the sketch would not be appropriate for a body of revolution. The obvious choice in this case would be cylindrical-polar coordinates (x, ϕ, r) . This also satisfies the requirements on coordinates necessary to use the Surface Spline and therefore, fits in very nicely.

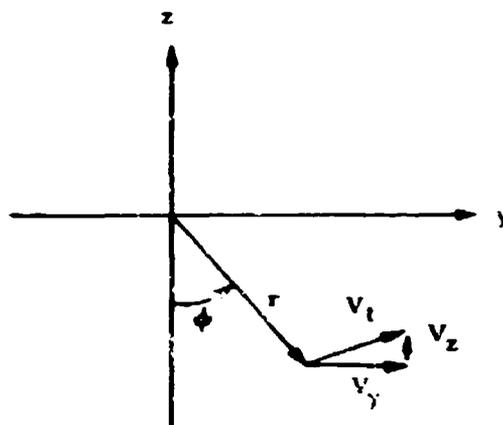
In general, the coordinates used in obtaining the solution will not correspond to the x, y, z reference and it is appropriate to express the differential equations in a slightly more general form:

$$\frac{dx_i}{ds} = C_i; \quad i = 1, 2, 3$$

The C_i are now interpreted to mean the rate of change of i -th coordinate along the path.

This distinction can be made clear by digressing for a moment and explicitly considering the aforementioned cylindrical-polar coordinates. The coordinates and components of the velocity vector are related as follows (see sketch):

$$\begin{aligned} x &= x \\ y &= r \sin \phi \\ z &= -r \cos \phi \\ V_x &= V_x \\ V_r &= V_y \sin \phi - V_z \cos \phi \\ V_\phi &= V_y \cos \phi + V_z \sin \phi \end{aligned}$$



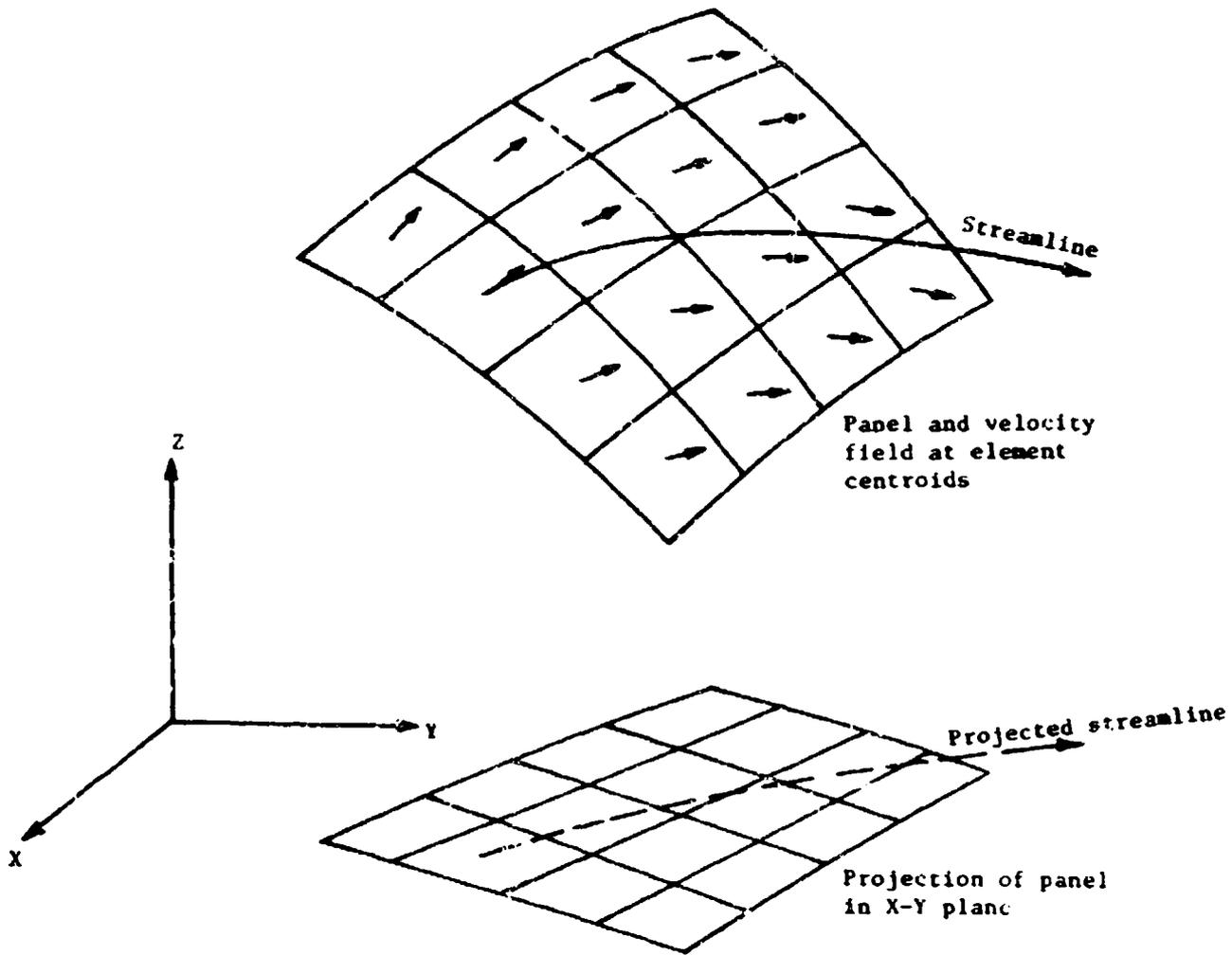


Figure 52. Schematic of Streamline Geometry.

Differentiating the coordinate equations, substituting into the streamline equations and solving in terms of the new coordinates gives:

$$\frac{dx}{ds} = \frac{V_x}{V_T} \equiv C_x$$

$$\frac{d\phi}{ds} = \frac{V_y \cos \phi + V_z \sin \phi}{V_T r} = \frac{V_\phi}{V_T} \cdot \frac{1}{r} \equiv C_\phi$$

$$\frac{dr}{ds} = \frac{V_y \sin \phi - V_z \cos \phi}{V_T} = \frac{V_r}{V_T} \equiv C_r$$

The C_x and C_r represent the direction cosines of the x and r velocity components but the C_ϕ is the direction cosine of the ϕ velocity component multiplied by $1/r$.

Similarly, any scaling required in normalizing the coordinates can be incorporated in the definition of C_i . For example, let the x coordinate be scaled by length L :

$$x_n = \frac{x}{L}$$

then

$$\frac{dx_n}{ds} = \frac{1}{L} \frac{V_x}{V_T}$$

and simply define

$$C_x \equiv \frac{1}{L} \frac{V_x}{V_T}$$

The purpose of all this is to keep the form of the Runge-Kutta (i.e., the streamline solution) independent of coordinate choice. The importance of this statement cannot be over stressed. It allows the flexibility necessary to define streamlines on an arbitrary body.

The Mark IV program streamline calculation procedure may be summarized as follows:

First, the velocity field on the body is calculated (or input) at discrete points. Next, the Surface Spline is passed through these points to provide a numerical definition of the body and velocity field. Then, the streamline is calculated using a fourth-order Runge-Kutta integration.

This procedure has been compared to an analytical solution to check its accuracy and the results are shown in Figure 53. The body is a prolate

Differentiating the coordinate equations, substituting into the streamline equations and solving in terms of the new coordinates gives:

$$\frac{dx}{ds} = \frac{V_x}{V_T} \equiv C_x$$

$$\frac{d\phi}{ds} = \frac{V_y \cos \phi + V_z \sin \phi}{V_T r} = \frac{V_\phi}{V_T} \cdot \frac{1}{r} \equiv C_\phi$$

$$\frac{dr}{ds} = \frac{V_y \sin \phi - V_z \cos \phi}{V_T} = \frac{V_r}{V_T} \equiv C_r$$

The C_x and C_r represent the direction cosines of the x and r velocity components; but the C_ϕ is the direction cosine of the ϕ velocity component multiplied by $1/r$.

Similarly, any scaling required in normalizing the coordinates can be incorporated in the definition of C_x . For example, let the x coordinate be scaled by length L :

$$x_n = \frac{x}{L}$$

then

$$\frac{dx_n}{ds} = \frac{1}{L} \frac{V_x}{V_T}$$

and simply define

$$C_x \equiv \frac{1}{L} \frac{V_x}{V_T}$$

The purpose of all this is to keep the form of the Runge-Kutta (i.e., the streamline solution) independent of coordinate choice. The importance of this statement cannot be over stressed. It allows the flexibility necessary to define streamlines on an arbitrary body.

The Mark IV program streamline calculation procedure may be summarized as follows:

First, the velocity field on the body is calculated (or input) at discrete points. Next, the Surface Spline is passed through these points to provide a numerical definition of the body and velocity field. Then, the streamline is calculated using a fourth-order Runge-Kutta integration.

This procedure has been compared to an analytical solution to check its accuracy and the results are shown in Figure 53. The body is a prolate

PROLATE SPHERIOD

$b/a = 0.2$
 $\alpha = 30^\circ$

Starting point for Mark IV calculations
 as taken from analytical results -

$X = 0.9611$
 $\phi = 11.16^\circ$

- Analytically generated streamline
- Mark IV $\Delta S = 0.002$ (every other calculated point is plotted)
- △ Mark IV $\Delta S = 0.02$ (every calculated point is plotted)

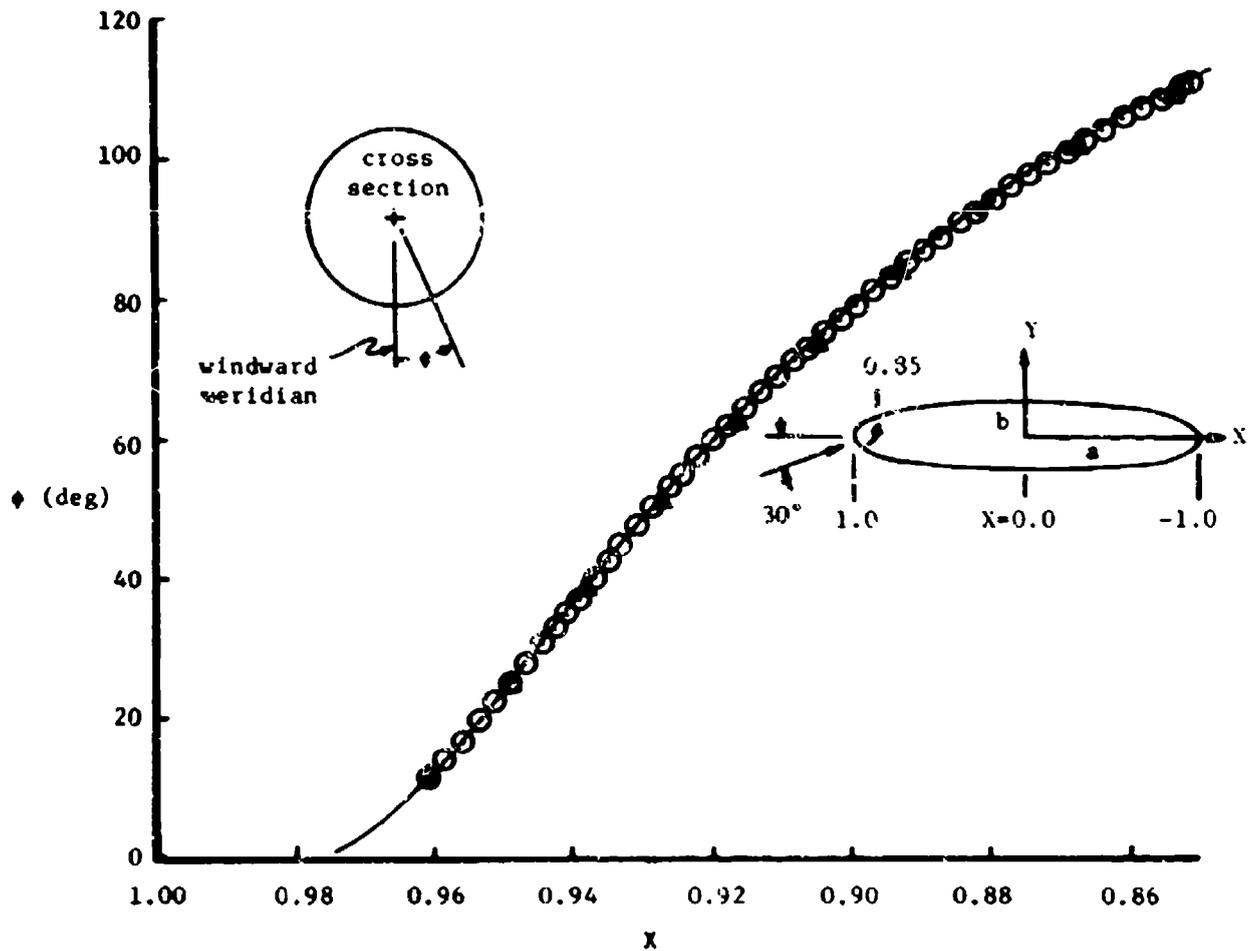


Figure 53. Comparison of Calculated Surface Streamline Paths

spheroid and the flow is incompressible, non-lifting at 30 degrees angle-of-attack. Both the body and the velocity can be defined analytically. The Mark IV calculations are shown for two ΔS increments and are in excellent agreement with the analytical solution. The Mark IV calculations have been made both forward (+ ΔS) and backwards (- ΔS) and trace the same streamline.

In addition, the calculations were done both in the three-variable (x, y, z) form and in the two-variable parametric form. There is no perceptible difference in the results of ϕ versus x given in Figure 53, however, the three-variable streamline did leave the surface as anticipated. For the present case this deviation from the surface was very small. It should be noted that all the various Mark IV calculations were made through input flags.

SECTION X

VISCOUS FORCE CALCULATION METHODS

The most difficult part in the analysis of an arbitrary shape is the calculation of viscous forces. A detailed knowledge of the local properties and the flow history along surface streamlines is required. This combined with the natural complexity of the boundary-layer equations necessitates considerable simplification of the problem before solutions can be obtained. In the Mark III Hypersonic Arbitrary Body program an engineering approach was used in calculating viscous forces that was simple yet retained the essential characteristics of the boundary layer problem. No attempt was made to calculate the detailed skin friction over the exact arbitrary shape used for the inviscid pressures. Instead, for skin friction purposes the vehicle was represented by a simplified geometry model composed of a small number of flat surfaces on each of which the shear force was determined.

This approach was considered by the authors to be quite consistent with the state of boundary layer theories. However, some users objected to this simple approach because it required the loading of another geometry deck in addition to the inviscid pressure geometry model. It is not clear just why there should be an objection to loading an additional 20 or 30 elements after as many as 1500 to 2000 have been loaded for the inviscid pressures.

However, the simplified skin friction geometry model approach is still recommended as being the most economical use of the machine resources. The Mark III skin friction capabilities have, therefore, been retained in the new Mark IV program, although they have been expanded slightly to allow a larger number of elements to be analyzed on one pass into the skin friction option. Also, the local properties on the skin friction elements are now calculated in the inviscid portion of the program just as though they were inviscid geometry elements.

However, in spite of the above comments it is recognized that some types of viscous studies will require a more detailed analysis than is possible with the simplified geometry model approach. To cover these situations the Mark IV program also has a new alternate viscous analysis method that works with the same geometry as is used for the inviscid pressures. In this method the viscous flow is calculated using an integral boundary layer method. The computations are made along streamlines calculated over the detailed inviscid pressure geometry model. Once the skin friction is calculated along these streamlines it is fit with the surface spline routines and the skin friction coefficient on each element of the detailed geometry model determined by interpolation.

The integral boundary layer method does not calculate the surface wall temperature. However, this information is required by the integral method so it must be furnished as input data or it can be calculated by calling the same temperature calculation routine used by the simplified skin friction model option (old Mark III). The temperature calculation routine itself calculates a skin friction coefficient value along with the wall temperature based on the old Mark III methods (i.e., reference temperature, Spalding-Chu, etc.). If the user wishes, this skin friction coefficient

can be used instead of that calculated by the integral boundary layer method. Under this mode of operation everything is handled just as though the integral method was being used (i. e., the use of streamlines on the detailed geometry model, element skin friction from the surface spline interpolation, etc.). The only difference is that after the temperature and Mar_{III} skin friction are calculated, no call is made to the integral program, and instead, the computations proceed with the skin friction coefficients out of the temperature program.

The user should be cautioned that the methods used in the new viscous capabilities of the Mark IV program are still not what we would call a "three-dimensional" boundary layer solution. Although computations are made along streamlines, the integral boundary layer method used at the present time is still basically a two-dimensional method (cross flow pressure gradients are not accounted for). Later versions of the program may include streamline divergence effects but the initially released program does not. Also, in the first release of the Mark IV program there is no capability for continuing a streamline calculation across one geometry component and on to an adjacent component. Therefore, the new integral boundary layer method should be restricted to relatively well behaved parts of a vehicle shape.

The Integral Boundary Layer Method

The integral boundary layer method contained in the viscous part of the Mark IV program is essentially the same program as presented by W. D. McNally in NASA TN D-5681 (Reference 40). A major modification to the McNally program was required to remove the assumption of isentropic flow implicitly used throughout the boundary layer equations. Minor modifications in the coding have also been made to improve efficiency and to reduce storage requirements. This integral boundary layer program is well documented in McNally's report so the development of the equations, etc., will not be duplicated in this report. Any user of the integral method in the Mark IV program should obtain a copy of NASA TN D-5681 as supplementary documentation.

The integral boundary layer method uses Cohen and Reshotko's method for the laminar boundary layer calculations (Reference 41), and Sasman and Cresci's method (Reference 42) for the turbulent boundary layer. The Schlichting-Wrick-Granville method (Reference 43-45) is used to predict the transition point. The present Mark IV program does not calculate transitional flow data between the wholly laminar and turbulent conditions. However, some of the code in the skin friction data storage part of the program provides facilities for inclusion of transitional data in future program additions.

The application of the integral boundary layer method involves the use of several other parts of the program. The process starts with the analysis of the vehicle component in the inviscid pressure part of the program. Subroutine FORCE calculates the local flow properties on each element. These properties (direction cosines of the surface velocity vector, Mach number and local pressure and temperature ratios to freestream conditions) are saved on the Unit 4 geometry data storage set right along with

the component geometry data. It is then necessary to enter the flow-field option and to have the surface property data copied onto the flow field storage Unit 10 by the Surface Data Transfer option. This step is necessary so that the data will be in the proper format for use by the surface spline routines.

The streamline option is then called. It is, in general, not possible or desirable to try to calculate streamlines down each row of geometry elements. Instead, streamlines are only calculated so as to properly cover the inboard and outboard extremes of the component together with one or more interior lines. The streamline surface trajectories together with the associated interpolated local surface properties along them are stored back on the flow field data Unit 10. Only after all of the above steps are accomplished can we call the viscous part of the program.

In the viscous part of the program input parameters are used to retrieve selected streamline data sets. The integral boundary layer method is applied along each of these streamlines and the resulting skin friction coefficients stored back on the flow field storage unit right with the stream-

We now have skin friction data along the streamline data points but not on the actual geometry quadrilateral centroids where we need them for the force integrations. To solve this problem the streamline skin friction data are fit with the surface spline method and the skin friction coefficients found at each of the element centroids by interpolation.

However, to accomplish the spline surface fit and interpolation it is necessary that the data be well behaved. That is, laminar, transitional, turbulent, and separated flows must be prepared and interpolated as separate sets of data. The bookkeeping (pointers, counters, etc.) necessary to provide this is automatically accomplished within the main integral boundary layer routine. Because of these features, the integral boundary layer method is able to analyze a component that has mixed boundary layer types, i.e., laminar and turbulent flows. The simplified skin friction model using the old Mark III methods is not capable of this.

However, there are certain things in the Mark III skin friction option that are not available in the integral boundary layer method. For example, the Mark III skin friction includes viscous induced pressure effects on the skin friction coefficient and the integral program does not.

Mark III Skin Friction Option

In the Mark III skin friction option of the Mark IV program an engineering approach has been selected that retains the essential characteristics of the hypersonic boundary-layer problem. No attempt is made to calculate the detailed skin friction distribution on the exact arbitrary shape, but rather, the vehicle is represented by a number of flat surfaces on each of which the shear force is determined.

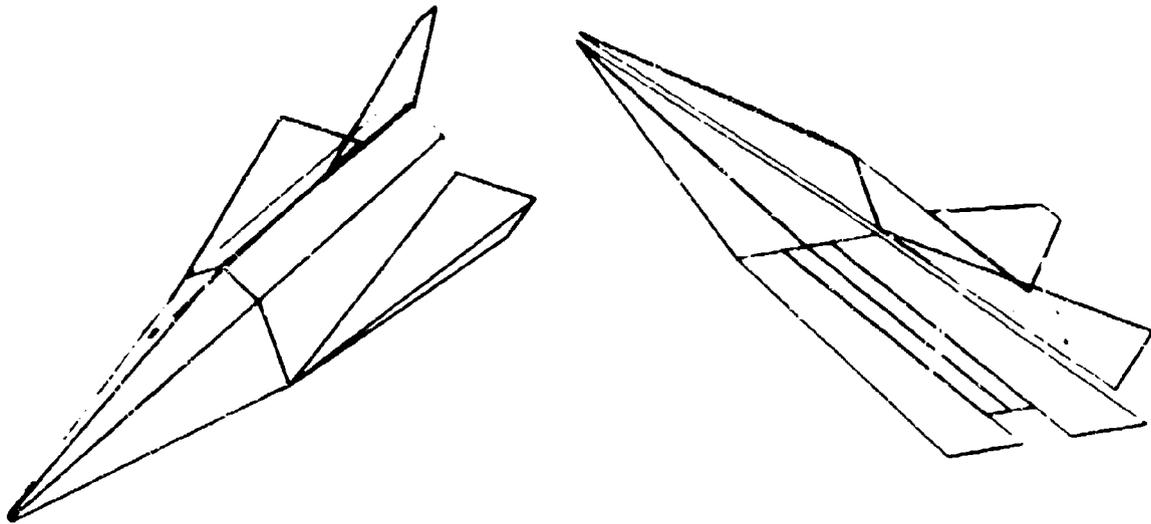
The surface streamlines are assumed in the velocity plane and the flow history is approximated by the inclusion of an initial surface. The shear force is determined for both laminar and turbulent flow and may be summed over the vehicle for either type.

Reference temperature and reference enthalpy methods are available for both laminar and turbulent flows and, in addition, the Spalding-Chi method with either temperature or enthalpy ratios may be selected for turbulent calculations. The surface temperature may be either input or the radiation equilibrium value determined. The effect of planform shape, leading edge viscous-interaction, and the viscous forces on blunt bodies are also considered.

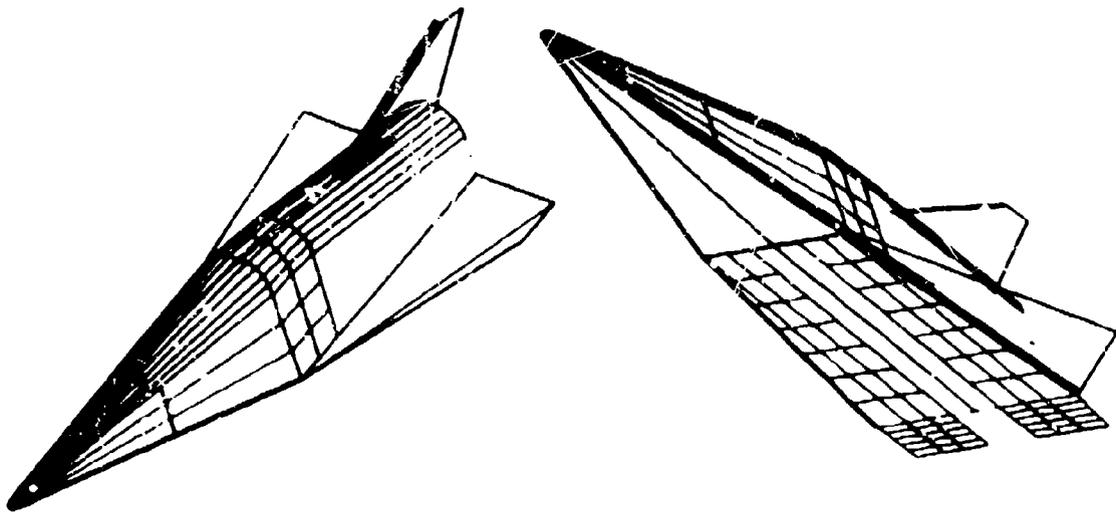
Skin Friction Geometry Model

For the skin friction calculations a geometrically complex vehicle is divided into a number of plane surfaces in a manner which adequately approximates the true shape. Leading-edge surfaces and local curvatures are omitted. Regions of relatively large curvatures can be represented by using a greater number of plane surfaces. The degree to which this is done will depend upon the complexity of the actual shape and experience of the designer. The geometry data for the skin-friction geometry model is prepared in the same way as the surface element data used for the inviscid pressure calculations and retain their relative location to each other and to the flight path. This skin friction modeling technique is best described by viewing, for example, a typical high L/D vehicle shown in Figure 54. The upper half presents the skin-friction representation of the vehicle which is to be contrasted with the detailed inviscid geometry given in the lower half of the figure. As used in the Supersonic-Hypersonic Arbitrary-Body Program, the skin friction surfaces are referred to as an approximate representation of the vehicle. This is in contrast to the inviscid geometry which for all practical purposes exactly represents the vehicle.

From the input element data, the surface normal, area, and area centroid coordinates are calculated. In addition, maximum chord length, taper ratio, and true area are input for each surface. The latter may be different from the calculated areas since curvatures have been neglected. The initial surface, specified by its maximum chord length and taper ratio, is assumed to be in the plane of the skin-friction surface and, therefore, the flow history is only approximated. The element planform effect on the average skin friction is included, however, and is discussed separately for laminar and turbulent flows in later sections. The shear force on each surface is assumed to act through its centroid in a direction on the surface parallel to plane containing the surface normal and the free-stream velocity vector, as described in Section VII.



a) Representation for Viscous Calculations



b) Representation for Inviscid Calculations

Figure 54. Geometry Modeling for a Typical High L/D Vehicle

Local Flow Conditions

The required local properties (pressure, temperature, density, and velocity) for use in calculating the viscous flows for both the integral boundary layer method and the Mark III method are calculated within the inviscid pressure part of the program. That is, the geometry over which the viscous flow is to be calculated must first be analyzed just as though it were regular inviscid geometry data. A flag in the pressure part of the program controls the storage of the required local property data so that it will be available for the viscous options.

The skin-friction surfaces and local properties, thus, have been defined in a way that reduces the problem of calculating the viscous forces on a complex shape to one of solving for the skin friction on a number of constant-property flat plates.

Incompressible Flow

The basic philosophy behind both the Spalding-Chi and the reference condition methods is the same. Namely, that the suitably transformed skin-friction coefficient is given by the constant-property or incompressible formulas based on a Reynolds number also suitably transformed. To emphasize the point, this may be stated another way: The compressible skin-friction is given by the incompressible form with appropriate correction factors to account for compressibility effects. That is,

$$C_{f\delta} = C_{f_i} / F_c$$

$$C_{f_i} = f(Rx_i), \quad Rx_i = F_{Rx} \quad Rx$$

where

$$C_f = \text{skin friction coefficient}$$

$$Rx = \text{Reynolds number}$$

$$(\)_i = \text{indicates incompressible}$$

$$(\)_\delta = \text{indicates compressible}$$

The incompressible formulas used in the Hypersonic Arbitrary-Body Program are given in Table 5 and the compressibility factors, F_c and F_{Rx} are discussed below.

Flow	Skin Friction Coefficient, $f(Rx_1)$		Source
	Local	Average	
Laminar	$0.664/\sqrt{Rx_1}$	$1.328/\sqrt{Rx_1}$	Blasius
Turbulent ($Rx_1 > R_{Min}$)	$0.088 (\log Rx_1 - 2.3686)$ $[\log Rx_1 - 1.5]^3$	0.088 $[\log Rx_1 - 1.5]^2$	Sivells & Payne (Ref. 65)
R_{Min}	2540	6570	

Table 5. Incompressible Skin-Friction Coefficient Formulas

The Sivells and Payne formulas have singularities occurring at low Reynolds numbers. However, both occur below the point at which the turbulent values cross the respective Blasius laminar curves. Thus, the turbulent incompressible skin-friction coefficients for Reynolds numbers equal to or less than R_{Min} are given by the corresponding laminar values.

Compressible Flow

Reference Temperature and Reference Enthalpy Method

$$F_c = \rho_\delta / \rho^\circ$$

$$F_{Rx} = (\mu_\delta / \mu^\circ) \frac{1}{F_c}$$

where ρ is the density, μ the viscosity, and the superscript " \circ " means evaluated at the reference temperature, T° , or reference enthalpy, H° ;

$$\frac{T^\circ}{T_\delta} = (A1) \frac{T_W}{T_\delta} + (A2) \frac{T_{AW}}{T_\delta} + (1 - A1 - A2)$$

$$\frac{H^\circ}{H_\delta} = (A1) \frac{H_W}{H_\delta} + (A2) \frac{H_{AW}}{H_\delta} + (1 - A1 - A2)$$

The value of the coefficients used are due to Monaghan (Reference 46) for Prandtl number equal to 0.71,

$$A1 = 0.5825$$

$$A2 = 0.1675$$

The subscript "W" indicates the wall value and subscript "AW" refers to adiabatic wall conditions given by

$$\frac{T_{AW}}{T_A} = \frac{H_{AW}}{H_A} = 1 + \left(\frac{\gamma-1}{2}\right) \gamma M_A^2$$

where

γ = ratio of specific heats

M = Mach number

r = recovery factor = $(P_r)^{1/n}$

$n = 2$ for lamirar flow

$n = 3$ for turbulent flow

P_r = Prandtl number, ≈ 0.71)

Spaiding-Chi Method (Reference 38):

$$F_c = A / \left\{ \text{ARSIN} \left(\frac{A-B}{C} \right) + \text{ARSIN} \left(\frac{A+B}{C} \right) \right\}^2$$

where

$$A = \frac{H_f W}{H_A} - 1$$

$$B = \frac{H_W}{H_A} - 1$$

$$C = \left[(A+B)^2 + 4A \right]^{1/2}$$

$$F_{Rx} = \left(\frac{H_{AW}}{H_A} \right)^q / \left[F_c \left(\frac{H_W}{H_A} \right)^{p+q} \right], \quad q = 0.772, \quad p = 0.702$$

Surface Equilibrium Temperature

In the Arbitrary-Body Program the surface equilibrium temperature is defined as the temperature satisfying the steady-state heat balance between the boundary-layer convection to the surface and the surface radiation to space.

convective heating: $QC(T_c) = C_h (H_{AW} - H_W)$

radiation heating: $QR(T_r) = R_K T_R^4$

where: C_h is the heat transfer coefficient

and $R_K = \epsilon \sigma$, ϵ = emissivity, (≈ 0.8)

σ = Stefan-Boltzman constant

The surface equilibrium temperature is defined when $QC(T_C) = QR(T_R)$ for $T_C = T_R$. The solution is obtained by a simple linear intercept technique illustrated in the sketch and explained briefly as follows.

Linear relations are assumed for both heating rates

$$QC = AC + (BC)T$$

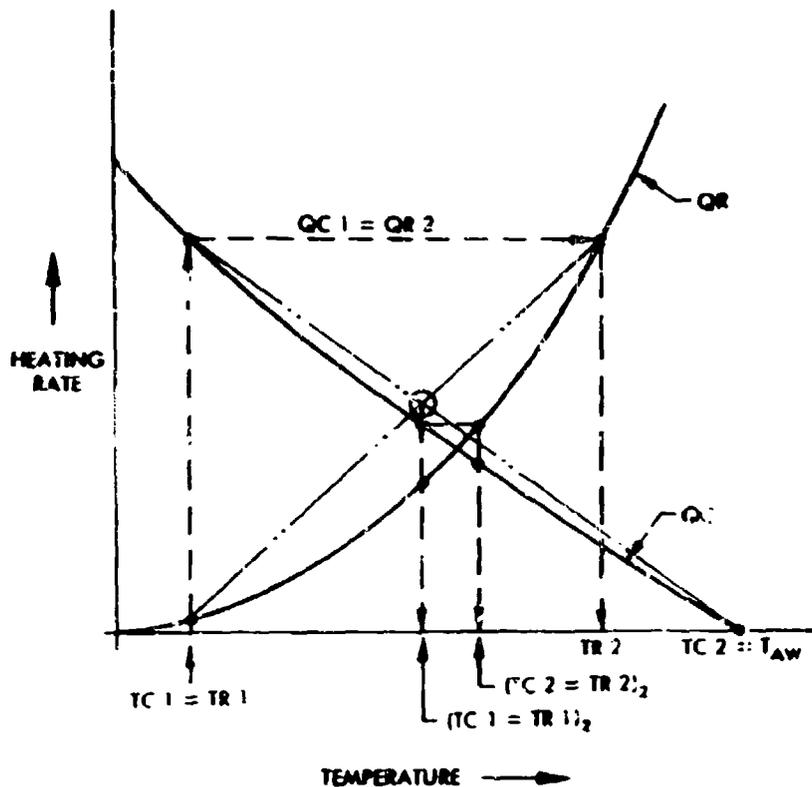
$$QR = AR + (BR)T$$

The four coefficients are initialized as follows.

1. Assume $TC1 = TR1 = 100^\circ R$
2. Calculate $QC1$ and $QR1$
3. Let $QR2 = QC1$ and calculate

$$TR2 = \left(\frac{QR2}{R_K} \right)^{1/4}$$

4. If $TR2 > TC2 = T_{AW}$, then set $TR2 = TC2$ and calculate new $QR2$



The coefficients may now be readily determined and the result of the linear solution of the heat balance equation is simply

$$T = (AC - AR)/(BR - BC)$$

The convective and radiation heating rates are then calculated at this temperature and checked for convergence:

$$|1 - QC1/QR1| \leq EPST, \text{ where } EPST = 5.0 E-4$$

If the criteria is not satisfied the cycle is repeated with $TC1 = TR1 = T$, $QR2 = QC1$, and $TC2 = TR2$. The present technique, while lacking sophistication, is accurate and quite rapid. Normally, two or three cycles are required for ideal gas solutions and one additional cycle for real gas cases.

Real Gas Effects

It is felt that some comments are in order with regard to the overall procedure. Specifically, what is the correctness or justification in using real gas reference enthalpy viscous solutions when the local inviscid flow has been determined only for a calorically perfect or ideal gas? To answer this question, an extensive comparison of laminar boundary-layer methods was undertaken in support of an earlier study and the details are reported in Reference 47. Briefly, the skin friction was determined for the flight conditions of the matrix given in Table 6 corresponding to the surface equilibrium temperatures (emissivity = 0.8) at the one-foot station of a flat plate.

Altitude (1000 Ft)	Velocity (1000 fps)					
	8	12	16	20	24	28
100	x	x	x	x	-	-
150	-	x	x	x	x	-
200	-	-	x	x	x	x
250	-	-	x	x	x	x

Table 6. Flight Matrix for Skin Friction Calculations

Angle-of-attack variation from 0° to 40° in 10° increments and five boundary-layer calculations were made at each condition. The latter correspond to the combination of three boundary-layer solutions and two shock wave solutions for local properties as shown in Table 7.

Boundary Layer Solution	Local Properties	
	Real	Ideal
Exact	1	-
Reference Enthalpy	2	3
Reference Temperature	4	5

Table 7. Boundary Layer Calculations

Also, additional calculations were made at the flight condition of 20,000 fps, 200,000 feet altitude, and wall temperature equal to 2000°R .

Methods 1, 2, and 5 are self-consistent with respect to the assumptions made and are regarded as normal calculation modes. Methods 3 and 4 are inconsistent in the assumptions made between the inviscid and viscous solutions and are termed mixed calculation modes. The free-stream properties were specified by the 1962 U. S. Standard Atmosphere and Sutherland's viscosity formula. The oblique shock-wave solutions are accurate to 5-significant digits in the inverse density ratio. For the real gas solution, the thermodynamic properties for equilibrium dissociating and ionizing air were obtained by the method in Reference 48. The assumed ideal gas is calorically perfect with ratio of specific heats equal to 1.40.

The real gas variation for the density-viscosity product in the viscous solutions was obtained as a function of enthalpy and pressure using the polynomial equations given in Reference 49. This product is based on the most recent thermodynamic data of Hilsenrath (Reference 50) and the viscosity calculations of Hansen (Reference 51). The Prandtl number was assumed equal to 0.71 for all the methods.

Typical results of the comparison are shown in Figure 55. The exact solutions were obtained using the Douglas General Laminar Compressible Boundary-Layer Program as described in Reference 49. The reference method calculations shown are based on the coefficient values of Monaghan. These were selected since the skin friction calculated consistently gave the best agreement with the exact results. Comparison of the three formulations considered - Monaghan (Reference 46), Michel (Reference 52) and Eckert (Reference 53) are shown in Table 8 for the same flight conditions as Figure 55. Major conclusions of the comparison are:

1. With the exception of possibly zero angle-of-attack the reference temperature method, using existing values for the coefficients A1 and A2, is inadequate for predicting skin friction for the complete range of hypersonic flight conditions considered.
2. The real gas, reference enthalpy method using Monaghan's formulation adequately predicts the laminar skin friction over the complete flight range considered. The results, however, are consistently about 3 to 5 percent lower than the exact calculations.
3. The mixed calculation mode, ideal gas inviscid - real gas reference enthalpy is in substantial agreement with the real gas reference enthalpy calculation up to 30° angle-of-attack.

Reference Enthalpy Due to	Angle of Attack in Degrees									
	0	5	10	15	20	25	30	35	45	50
Monaghan	0.247	0.623	1.056	1.445	1.753	1.967	2.070	2.121	1.853	1.590
Michel	0.247	0.628	1.062	1.447	1.747	1.953	2.067	2.075	1.788	1.529
Eckert	0.243	0.613	1.038	1.418	1.717	1.926	2.042	2.056	1.788	1.534

Table 8. Comparison of Reference Methods. Values of $C_f \times 10^2$. (Altitude = 200,000 Ft., Velocity = 20,000 ips, $T_w = 20600^\circ R$)

On the basis of the results of this study, the mixed-mode ideal gas inviscid-real gas reference enthalpy calculation has been included in the Hypersonic Arbitrary-Body Program. The real gas fluid properties of air are determined by the procedures described in detail in Reference 50. Three different formulas are used to specify the viscosity. At very low temperatures such as might be experienced in a high speed wind tunnel the viscosity is found from the

SYMBOL	SOLUTION
—————	REAL GAS, EXACT
-----	REAL GAS, REF. ENTHALPY*
- - - - -	IDEAL GAS, REF. TEMPERATURE*
▲	IDEAL GAS, REF. ENTHALPY*
■	REAL GAS, REF. TEMPERATURE*

*MONAGHAN'S REFERENCE
AT $P_r = 0.71$

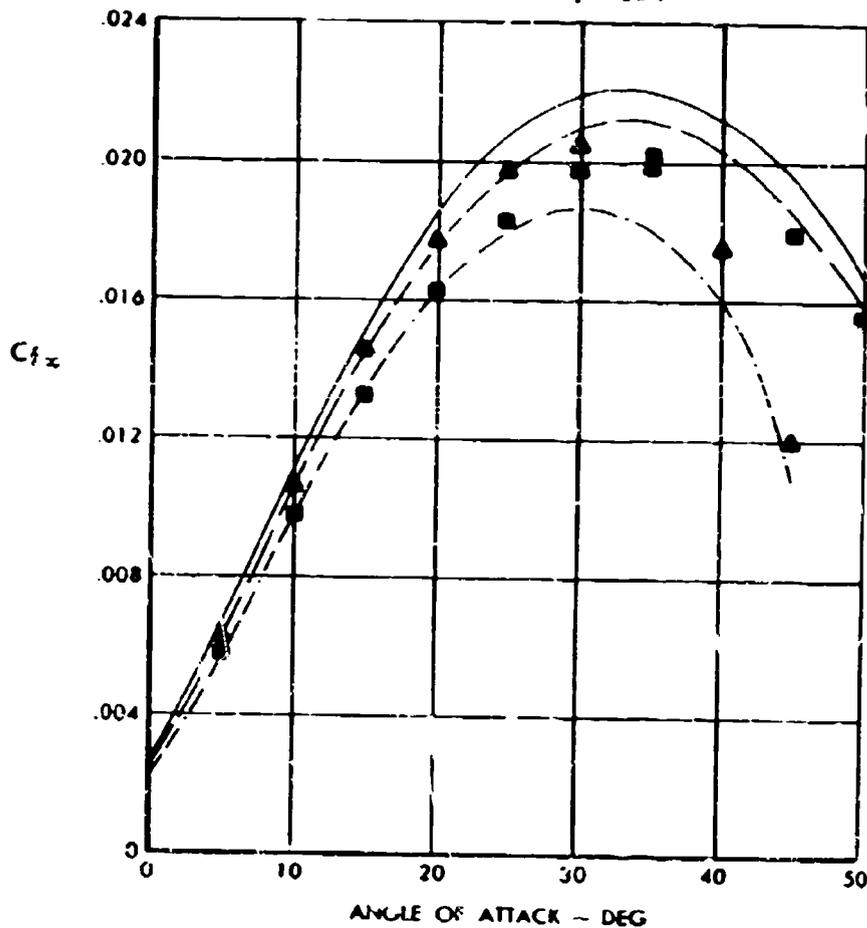


Figure 55. Laminar Skin-Friction Coefficient Comparison
(Altitude = 200,000 Ft., Velocity = 20,000 fps., $T_w = 2000^{\circ}R$)

Bromley-Wiike results (Reference 54). In the Arbitrary-Body Program these are approximated by the following linear relationship:

For $T \leq 225^\circ\text{R}$

$$\mu = 0.80383436 T \times 10^{-9} \frac{\text{lb sec}}{\text{ft}^2}$$

At higher temperatures and for an ideal gas the Sutherland viscosity formula is used (Reference 25):

For $T > 225^\circ\text{R}$

$$\mu = 2.270 \frac{T^{5/2}}{T + 198.6} \times 10^{-8} \frac{\text{lb sec}}{\text{ft}^2}$$

For real gas and temperatures greater than about 6000°R Hansen's viscosity values are used (Reference 51).

Viscous - Inviscid Interaction

Under conditions of low Reynolds number and high Mach number, the mutual interaction of the boundary layer and the inviscid flow field can have a large effect on both the laminar skin friction and surface pressure. Boundary-layer displacement effects in hypersonic flow over flat plates have been studied at length (e.g., Reference 55) and the present approach is limited to consideration of these methods. Basically, a pressure is induced from the relatively large outward streamline deflection caused by the thick hypersonic boundary layer. The classical approach is to consider an effective body, made up of the actual body plus the boundary-layer displacement thickness, in an iterative solution with the inviscid flow. This in itself is an approximation and, in addition, the simplifying assumptions of hypersonic viscous similarity are usually employed. This procedure has been adopted for use in the Arbitrary-Body Program and a brief background and development of the final equations follow.

Bertram and Blackstock (Reference 56) presented some simple procedures for estimating the boundary layer induced effects on pressure and skin friction. These involved the use of hypersonic-similarity-boundary-layer theory solutions in an iterative technique with the hypersonic small-disturbance tangent-wedge pressure equation. The analysis showed good correlation with experimental data for surfaces at nearly zero degrees incidence to the free-stream. White (Reference 57) extended the theory of Bertram and Blackstock to include the effect of angle of attack and presented a direct method for solving the problem without requiring iterations. White used hypersonic small disturbance expressions for both compression and expansion flows and introduced a new interaction parameter to correlate the wall temperature effect. Recently, Bertram (Reference 58) has presented more elaborate solutions for the problem employing the techniques of White. Implicit to all these solutions is the assumption of a calorically perfect gas and a Prandtl number of unity.

White's solution has been used in the present analysis because of the relative simplicity in its application. His numerical results showed the local pressure to be nearly a linear function of the iteration parameter, λ :

$$P = P_0 (1 + B)$$

where

$$B = m \frac{\lambda}{P_0}$$

and

$$\lambda = \frac{G M_c^3}{\sqrt{1+2j}} \left(\frac{C}{R_x} \right)^{1/2}$$

The quantity G is a simple function of wall temperature and specific heat, C is the Chapman-Rubesin viscosity coefficient, and j is the Mangler transformation parameter: two-dimensional flow, $j=0$; axially-symmetric flow, $j=1$.

In the above equations, P is the local pressure to free-stream pressure ratio, and the subscript "0" refers to the inviscid value obtained from the hypersonic small-disturbance relations.

Bertram's (Reference 56) correlation for local skin friction coefficient is

$$C_f = 0.664 K_1 \left(\frac{PC}{R_x} \right)^{1/2}$$

where K_1 is a pressure gradient and wall temperature correction factor. The shear on the surface is

$$\tau_w = \int q_s C_f dA$$

In the present analysis, the approach taken is to determine the effect of factor due to viscous-interaction using White's method and then to modify the previous result without interaction accordingly. This viscous-interaction factor, K_{VI} , is obtained by carrying out the integration of the preceding equation and is defined as follows,

$$K_{VI} = \frac{(\tau_w)_{VI}}{\tau_w} = \sqrt{1 + B_{Cr}} + B_{Cr} \log_e \left[\frac{\sqrt{1 + B_{Cr}} + 1}{\sqrt{B_{Cr}}} \right]$$

where B_{C_T} is based on the root-chord and K_1 has been assumed equal to one. This expression is for a plate with taper ratio one, but the integration could have been done for an arbitrary value (e.g., Reference 59). In the present application the planform effects are included in the shear force without interaction, τ_w . This application results in a slightly lower factor but has the advantage of permitting a step-by-step build up and comparison of the overall viscous forces. The magnitude of the skin-friction correction factor using the above techniques is shown in Figure 56.

The induced pressure on a surface is determined as an increment in pressure coefficient.

$$\Delta C_p = C_p - C_{p_0} = \frac{\bar{P} - P_0}{\frac{\gamma}{2} M^2}$$

The average pressure increment, $\bar{P} - P_0$, is found by summing the local pressure distribution over the surface.

$$\bar{P} - P_0 = \frac{1}{A} \int (P - P_0) dA$$

Substituting the expression for local pressure and integrating gives

$$\bar{P} - P_0 = 2m\lambda c_T$$

The ΔC_p due to induced pressure is determined for the skin-friction geometry representation of the vehicle shape and effects due to the planform shape and due to the initial surface are discussed in the next section.

The basic hypersonic small-disturbance relations for calculating pressure are:

For compression flow ($K \geq 0$)

$$P = 1 + \gamma \left(\frac{\gamma+1}{4} \right) K^2 + \gamma K \left\{ 1 + \left(\frac{\gamma+1}{4} K \right)^2 \right\}^{1/2}$$

For expansion flow ($-2/(\gamma-1) \leq K \leq 0$)

$$P = \left[1 + \frac{\gamma-1}{2} K \right]^{2\gamma/(\gamma-1)}$$

The similarity parameter, K , is given by:

$$K = K_0 + \frac{\lambda K_0}{\sqrt{P}} \left[1 + \frac{\lambda}{2P} \frac{dP}{d\lambda} \right]$$

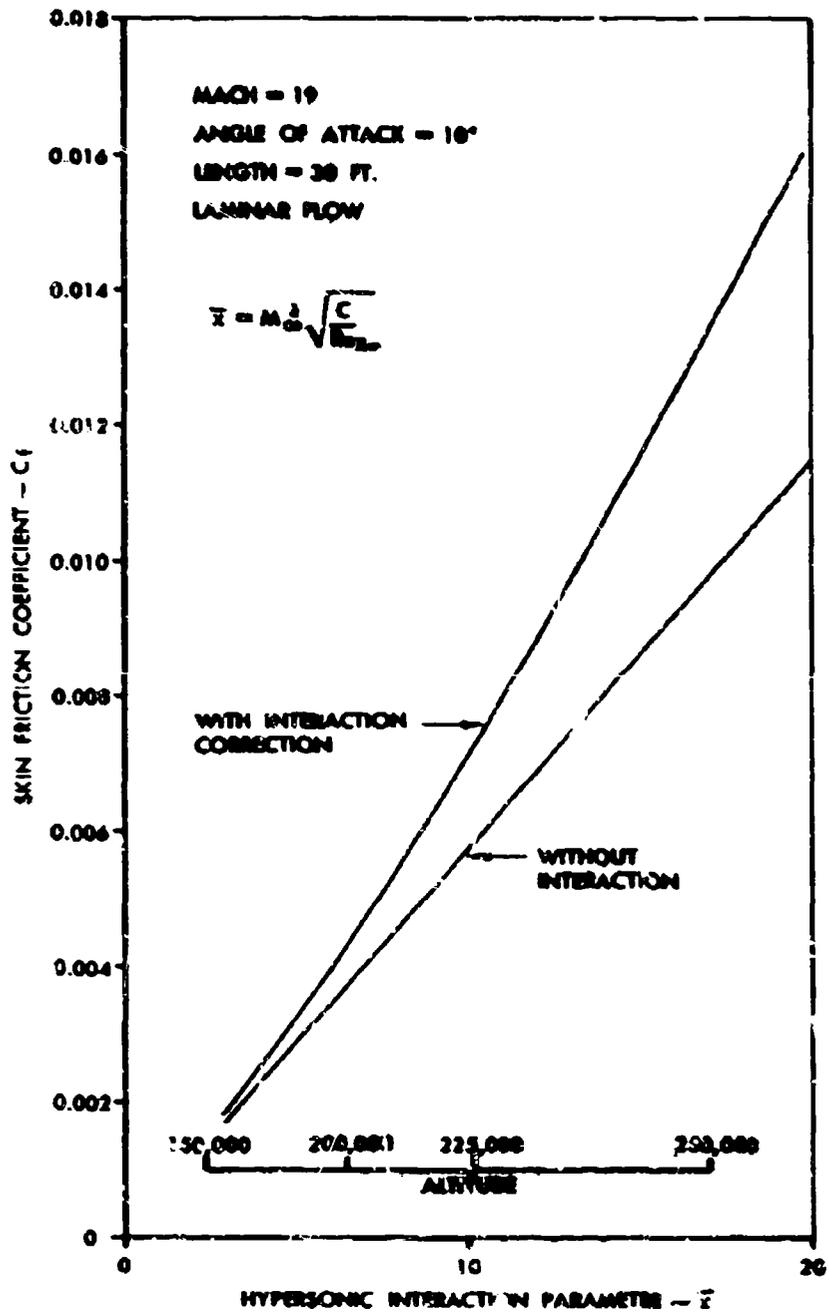


Figure 56. Effect of Viscous Interaction on Skin Friction Coefficient

where $K_0 = M_\infty \sin \delta$ (δ is the surface impact angle) and K_4 , a boundary-layer growth parameter, is taken equal to 1.0.

White (Reference 57) observed that the pressure equation (either compression or expansion) and the expression for K constituted a first-order nonlinear differential equation in $P(\lambda)$ and obtained numerical solutions directly without iteration. The results are shown in Table 9 from which White also observed that the pressure could be approximated by the linear relationship

$$P = P_0 + m\lambda$$

where P_0 and the slope parameter, m , are just functions of K_0 . P_0 is given by the hypersonic similarity relations as a function of K_0 and, in the Arbitrary-Body Program, m is approximated to the data of Table 9 by the following analytical curves:

For $-2/(\gamma - 1) \leq K_0 < -3.0$,

$$m = 1.424 + 0.215 K_0$$

For $K_0 \geq -3.0$,

$$m = 1.4156 + 0.41727 K_0 - 0.0419101 K_0^2 - 0.010427 K_0^3 + 0.00214381 K_0^4 - 0.000103217 K_0^5$$

For $K_0 \geq 10.0$,

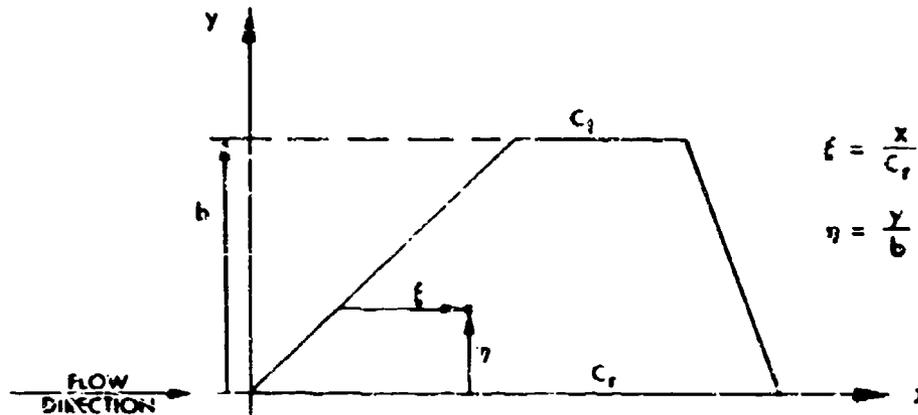
$$m = [2\gamma / (\gamma + 1)]^{1/2}$$

λ	Similarity Parameter K_0							
	-3	-2	-1	0	+1	+2	+5	+10
0.0	0.002	0.028	0.210	1.000	3.473	8.734	44.14	170.2
0.5	0.172	0.339	0.748	1.835	4.555	9.930	45.41	171.4
1.0	0.428	0.736	1.379	2.777	5.722	11.18	46.70	172.7
1.5	0.738	1.192	2.059	3.740	6.914	12.47	48.01	174.0
2.0	1.092	1.695	2.770	4.709	8.108	13.76	49.33	175.3
2.5	1.485	2.234	3.506	5.679	9.294	15.07	50.66	176.6
3.0	1.908	2.801	4.260	6.651	10.47	16.37	51.99	177.9
3.5	2.359	3.392	5.029	7.622	11.64	17.67	53.34	179.3
4.0	2.833	4.004	5.810	8.593	12.80	18.96	54.70	180.6
4.5	3.328	4.632	6.601	9.505	13.95	20.25	56.06	181.9
5.0	3.840	5.275	7.400	10.54	15.09	21.52	57.42	183.2

Table 9. Numerical Solutions for Pressure Ratio P ($\gamma = 1.4$)

Planform Effects

The previous sections have dealt with the determination of the local skin-friction coefficient or the average skin-friction coefficient per unit span. In this section, the determination of the viscous force contribution of a surface element having a planform shape of the type shown in the sketch below is considered. In the derivations that follow it is implicitly assumed that the root and tip chords are parallel to the oncoming flow.



The product of local skin-friction coefficient (C_{f_s}) and dynamic pressure (q_∞) is integrated over the surface area (A) to obtain the shear force.

$$\tau_w = \int q_\infty C_{f_s} dA$$

(The symbol τ is customarily used to define shear stress, however in the present text it is used consistently as a force. This is done to avoid the unnecessary use of area ratios in the defining equations and at the same time retain the significant connotation associated with the symbol.)

The shear force on each surface is then written as a coefficient with respect to the free-stream dynamic pressure (q_∞) and a specified reference area (S),

$$C_{F_w} = \frac{\tau_w}{\frac{1}{2} q_\infty S}$$

and summed over all surfaces to obtain the vehicle characteristics due to viscous forces.

Laminar Shear Force

The local properties are constant on each surface and the above expression becomes

$$\tau_W = q_\delta (C_{f_\delta})_{c_r} c_r \int_0^b \left\{ \int_0^c (x)^{-\frac{1}{2}} dx \right\} dy$$

where the surface has root chord c_r , span b , and $(C_{f_\delta})_{c_r}$ is evaluated at the root chord. The local chord length may be expressed as

$$c = c_r [1 - (1 - TR) \eta]$$

where TR is the taper ratio ($= c_t/c_r$) and η is the normalized span dimension ($= y/b$). Substituting this expression and completing the integration gives the shear force on the surface as

$$\tau_W = q_\delta A (C_{F_\delta})_{c_r} \frac{4}{3} \left[\frac{1 + TR + \sqrt{TR}}{(1 + TR)(1 + \sqrt{TR})} \right]$$

where $(C_{F_\delta})_{c_r}$ is the local, length-averaged skin-friction coefficient evaluated at the root-chord.

In the Arbitrary-Body Program the shear force is expressed in terms of an average chord length, \bar{c} ;

$$\tau_W = q_\delta A (C_{F_\delta})_{\bar{c}}$$

where

$$\bar{c} = c_r \left\{ \frac{4}{3} \left[\frac{1 + TR + \sqrt{TR}}{(1 + TR)(1 + \sqrt{TR})} \right] \right\}^2$$

Viscous-Interaction

As was explained in the previous section, the effect of planform on the shear force is not determined directly for flows with viscous-interaction but is included in the calculation of shear force without interaction. This procedure results in a slightly lower force but has the advantage of permitting a step-by-step build-up and comparison of the overall viscous forces. There is, however, an additional effect on the induced pressure due to planform shape which is accounted for.

The average pressure is obtained by integrating the local pressure over the surface:

$$\begin{aligned} \bar{P}_A &= \frac{1}{A} \int P \, dA = \frac{1}{A} \int_0^b \left\{ \int_0^c P \, dx \right\} dy \\ &= \frac{c_r b}{A} \int_0^1 \left\{ \int_0^{c/c_r} (P_0 + m \lambda_{c_r} \xi^{-\frac{1}{2}}) d\xi \right\} d\eta \end{aligned}$$

where $\xi = x/c_r$, the normalized streamwise coordinate.

Substituting the expressions for

$$A = \frac{c_r b}{2} (1 + TR)$$

and

$$c/c_r = 1 - (1 - TR)\eta$$

the integration is easily completed. The result is

$$\bar{P}_A = P_0 \left\{ 1 + \frac{\delta}{3} B_{c_r} \left[\frac{1 + TR + \sqrt{TR}}{(1 + TR)(1 + \sqrt{TR})} \right] \right\}$$

where

$$B_{c_r} = \frac{m}{P_0} \lambda_{c_r}$$

The average pressure increment for the surface is then

$$\bar{P}_A - P_0 = \frac{\delta}{3} m \lambda_{c_r} \left[\frac{1 + TR + \sqrt{TR}}{(1 + TR)(1 + \sqrt{TR})} \right]$$

which for $TR = 0$ reduces to the value previously given.

Turbulent Shear Force

Because of the nature of the assumed skin-friction formulas, a different approach than used for laminar flow is taken to obtain the turbulent shear force. The end result, however, is an approximate solution which is very similar to the laminar result. The shear force equation is derived as follows.

$$\begin{aligned}\tau_W &= \int q_\delta C_{f_\delta} dA = q_\delta \int_0^b \left\{ \int_0^c C_{f_\delta} dx \right\} dy \\ &= q_\delta b \int_0^1 c C_{F_\delta} d\eta\end{aligned}$$

The variable of integration is transformed to the local chord-length Reynolds number in two steps. First in terms of the chord length c ,

$$\tau_W = q_\delta b \int_{c_t}^{c_r} \frac{c C_{F_\delta}}{c_r (1 - TR)} dc$$

Next, the variable of integration is transformed to the incompressible Reynolds number, $Re_i = Fr_x \left(\frac{\rho U c}{\mu} \right)_\delta$, and normalized with respect to root-chord values;

$$\tau_W = \frac{q_\delta b c_r (C_{F_\delta})_{c_r}}{(1 - TR)} \int_{TR}^1 \left(\frac{Rc}{Rc_r} \right)_i \left(\frac{C_F}{C_{F_{c_r}} \delta} \right) d \left(\frac{Rc}{Rc_r} \right)_i$$

Noting that the surface area is $A = \frac{c_r b}{2} (1 + TR)$, and also that $\left(\frac{C_F}{C_{F_{c_r}} \delta} \right) = \left(\frac{C_F}{C_{F_{c_r}} \delta} \right)_i$, the shear equation becomes

$$\tau_W = q_\delta A (C_{F_\delta})_{c_r} \left(\frac{2}{1 - TR^2} \right) \int_{TR}^1 \left(\frac{Rc}{Rc_r} \right)_i \left(\frac{C_F}{C_{F_{c_r}} \delta} \right)_i d \left(\frac{Rc}{Rc_r} \right)_i$$

With a simple power-law skin-friction formula this equation is easily evaluated;

$$\left(\frac{C_F}{C_{F_{c_r}} \delta} \right)_i = \left(\frac{Rc}{Rc_r} \right)_i^{-\frac{1}{N}}, \text{ where } N \text{ is positive}$$

and

$$\begin{aligned}\tau_W &= q_\delta A (C_{F_\delta})_{c_r} \left(\frac{2}{1 - TR^2} \right) \int_{TR}^1 \left(\frac{Rc}{Rc_r} \right)_i^{1 - \frac{1}{N}} d \left(\frac{Rc}{Rc_r} \right)_i \\ &= q_\delta A (C_{F_\delta})_{c_r} \left(\frac{2}{2 - \frac{1}{N}} \right) \left(\frac{1 - TR^{2 - \frac{1}{N}}}{1 - TR^2} \right)\end{aligned}$$

For laminar flow $N = 2$ and it is easily verified that this expression is identical to the one previously presented.

In general, the skin-friction coefficient is not given by a simple power-law relationship and this is the reason for deriving the turbulent shear with the Reynolds number as the independent variable.

The use of the Sivells and Payne formula in the shear equation introduces a singularity in the integrand and the function is nonintegrable. However, this singularity occurs at a Reynolds number much below the laminar cutoff and the shear equation may be integrated numerically. Several examples for the numerically determined integrand are shown in Figure 57. The upper-bound represented by laminar flow and a lower-bound represented by constant skin-friction are also shown. The curves are smooth and the area under each curve times the quantity $2/(1 - TR^2)$ is the factor by which the shear increases due to a tapered planform.

It may be observed from Figure 57, that even with a large variation of Reynolds number on the planform (for example, $Rc_r = 10^9$ to zero at the tip), the major contribution to the integral is obtained over the first decade ($Rc/Rc_r = 1.0$ to 0.1). In the case of the upper-bound (laminar flow) and the lower-bound ($N = \infty$) this contribution is 97 and 99 percent, respectively. This then, suggested the approximate approach of representing the Sivells and Payne formula in the integrand over the entire Reynolds number range by a local power-law fit obtained as the average over the first decade.

Thus, the shear on the surface is obtained from the power-law solution with the exponent parameter, N , given as (for Sivells and Payne);

$$N = \frac{\log Rc_r - 2}{0.8686}$$

Alternately, as was done for laminar flow, the shear force may be expressed in terms of an average chord, \bar{c} ;

$$\tau_w = q_\delta A(C_{F_\delta})\bar{c}$$

where

$$\bar{c} = c_r \left(\frac{Rc_r}{10^{3/2}} \right)^{Q-1}$$

and

$$Q = \left\{ \left(\frac{1 - TR^2}{1 - TR^2 - \frac{1}{N}} \right)^{2 - \frac{1}{N}} \right\}^{\frac{1}{2}}$$

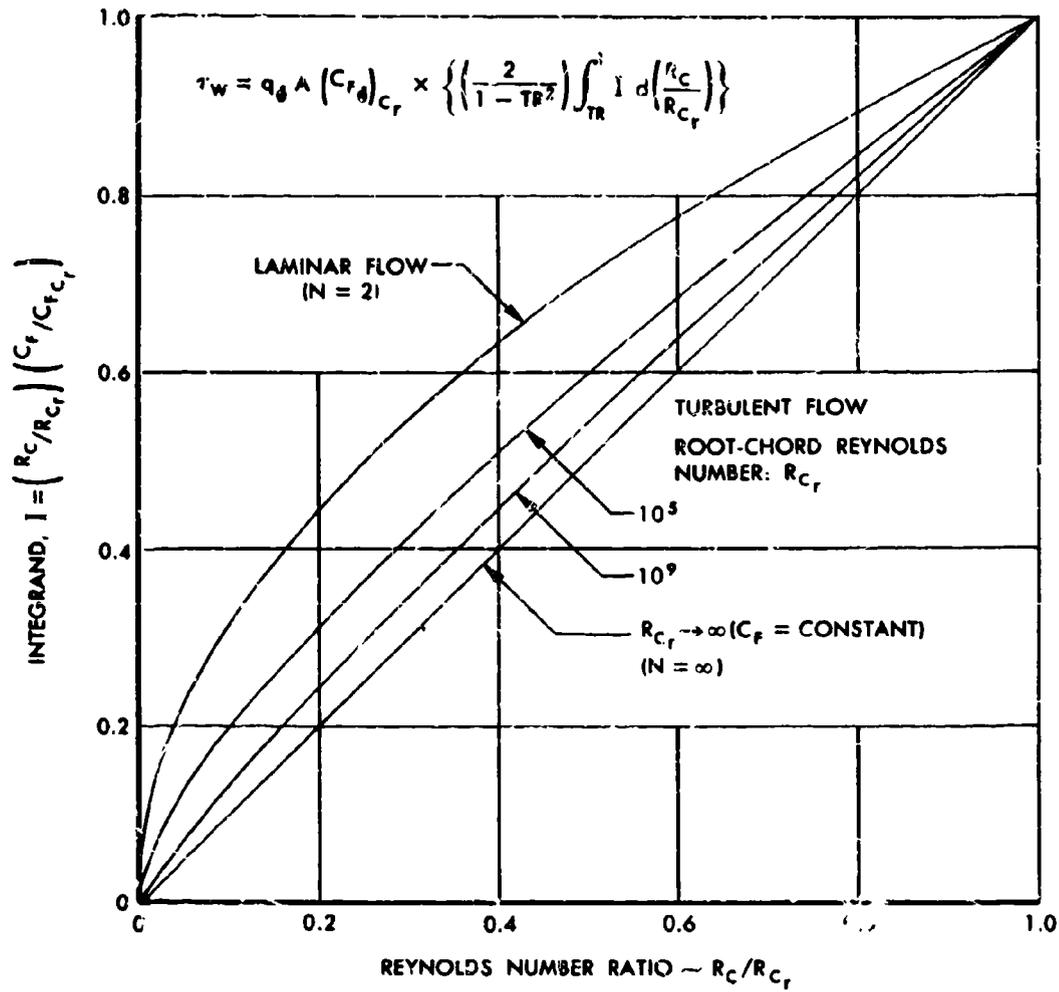


Figure 57. Planform Effect on Shear Force

Initial Surface Correction to Shear Force

When an initial surface is specified, the shear force is determined for the combined surface geometry, for the initial surface, and the difference obtained as the value for the surface of interest. This in effect is dealing with three surfaces which have the following characteristics (see sketch below):

1. Initial surface; Area A_1 , maximum chord length L_1 , taper ratio TR_1 , and shear force τ_{W1} .
2. Surface of interest; Area A_2 , maximum chord length L_2 , taper ratio TR_2 , and shear force τ_{W2} .
3. Combined surface; Area $A_3 = A_1 + A_2$, maximum chord length L_3 , taper ratio TR_3 , and shear force τ_{W3} .

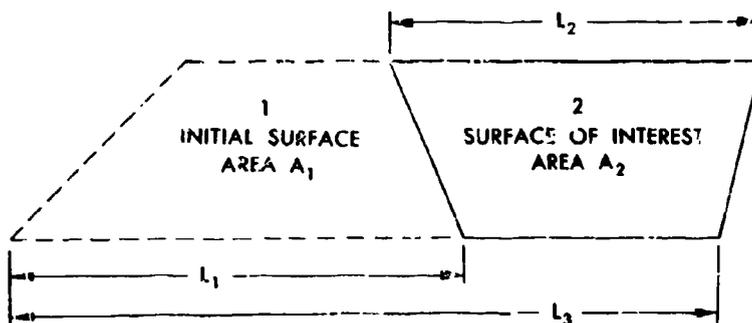
The shear force on surface 2 is

$$\begin{aligned} \tau_{W2} &= \tau_{W3} - \tau_{W1} \\ &= q_{\delta} A_2 (C_{F\delta} K_{VI})_3 \left\{ 1 - \frac{A_1}{A_2} \left[\frac{(C_F K_{VI})_1}{(C_F K_{VI})_3} - 1 \right] \right\} \end{aligned}$$

In the Arbitrary-Body Program this is compacted to the form

$$\tau_{W2} = q_{\delta} A_2 (C_{F\delta} K_{VI})_3 (1 - FF)$$

where FF has the mnemonic form factor or friction factor. Three possibilities are considered in determining the friction factor: (1) both surfaces laminar, (2) first surface laminar and second surface turbulent, and (3) both surfaces turbulent.



Initial Surface Correction to Induced Pressure

The average pressure on surface 2 is defined as follows:

$$P_2 = \frac{F_2}{A_2} = \frac{F_3 - F_1}{A_2} = \frac{\bar{P}_3 A_3 - P_1 A_1}{A_2}$$

where F_i is the force on surface i . The average pressures on the initial surface and on the combined surface are given by

$$P_1 = P_0 \left\{ 1 + \frac{8}{3} B_1 \left[\frac{1 + TR_1 + \sqrt{TR_1}}{(1 + TR_1)(1 + \sqrt{TR_1})} \right] \right\}$$

$$P_3 = P_0 \left\{ 1 + \frac{8}{3} B_3 \left[\frac{1 + TR_3 + \sqrt{TR_3}}{(1 + TR_3)(1 + \sqrt{TR_3})} \right] \right\}$$

and the areas by

$$A_1 = bL_1(1 + TR_1)/2$$

$$A_2 = bL_2(1 + TR_2)/2$$

$$A_3 = bL_3(1 + TR_3)/2$$

Substituting these expressions into the above definition and after some algebraic manipulation the result may be written as

$$\bar{P}_2 - P_0 = \frac{8}{3} m\lambda_3 \left(\frac{L_3}{L_2} \right) \left[\frac{1 + TR_3 + \sqrt{TR_3}}{(1 + TR_2)(1 + \sqrt{TR_3})} \right] \left\{ 1 - \left(\frac{L_1}{L_3} \right)^{\frac{1}{2}} \left(\frac{1 + TR_1 + \sqrt{TR_1}}{1 + TR_3 + \sqrt{TR_3}} \right) \left(\frac{1 + \sqrt{TR_3}}{1 + \sqrt{TR_1}} \right) \right\}$$

The length L_3 is defined as the maximum chord length of the combined surface, so as $L_1 \rightarrow 0$ it is readily verified that the pressure reduces to the same expression previously given for a single, tapered plate.

Viscous Force on Blunt Bodies

The earliest space capsules were designed with large spherical nose caps and flew ballistically at zero degrees angle of attack. For such vehicles, it was found that inviscid flow field calculations were adequate to predict the splash point. The later generation capsules were designed to fly at angle of attack to provide lift and it has been shown that viscous forces can have a significant effect on predicting the splash point. The theoretical solution, then, must provide some means for estimating the viscous effect.

The procedure included in the Arbitrary-Body Program is that developed by Goldberg of the General Electric Company (References 60 and 61). This method is given in the form of relatively simple correlation formulas in terms of the shock-layer Reynolds number and inverse density ratio. The method is applicable to the low density conditions associated with high altitude entry and is equally suited to real gas or ideal gas analysis.

The shear force in the stagnation region of a blunt-faced body is given as

$$\tau_w = \tau_{w_0} K_{VI}$$

where the shear without low density or viscous-interaction effects is

$$\tau_{w_0} = \frac{q_\infty A 2 \cos \delta}{(1 - 0.475 \sqrt{\epsilon}) \sqrt{Re_s}}$$

and δ is the surface impact angle,

ϵ is the inverse density ratio, $= (\rho_2/\rho_\infty)^{-1}$

Re_s is the shock Reynolds number

$$= \rho_2 U_2 R_B / \mu_2$$

R_B is the body nose radius.

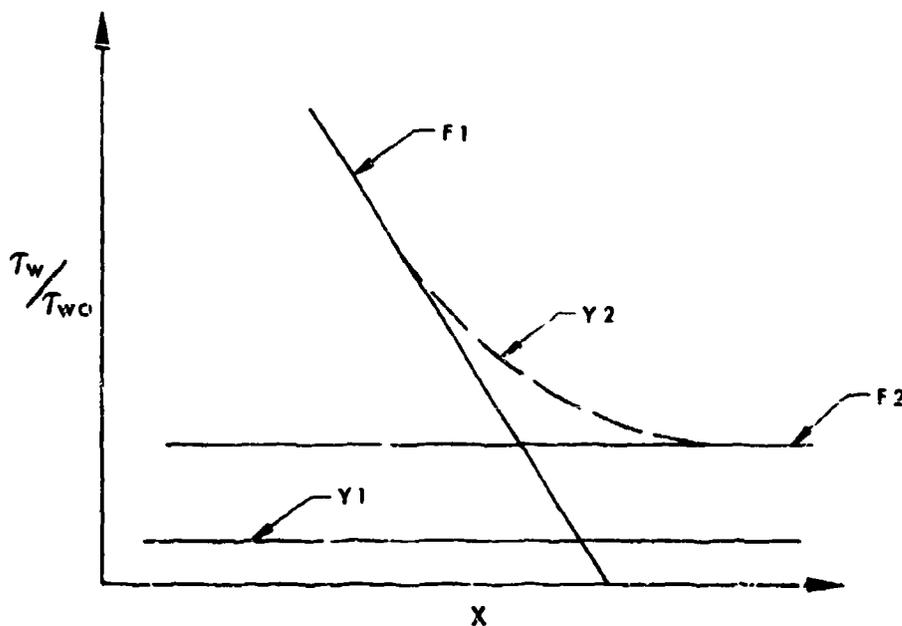
The viscous-interaction correction factor, $K_{VI} = \tau_w / \tau_{w_0}$, was obtained from higher-order analysis of the boundary-layer flow (Reference 60). The present authors have developed a correlation formula to represent these solutions in the Arbitrary-Body Program. This factor, a complicated function of both shock Reynolds number and density ratio, has been approximated by a combination of exponential transition functions of the type described by Grabau (Reference 62). These are

$$\text{even transition: } y = \frac{1}{1 - \exp K(X - X_0)}$$

$$\text{odd transition: } y = \frac{1}{1 + \exp K(X - X_0)}$$

These functions are essentially the kernels for the Bose-Einstein and for the Fermi-Dirac distribution functions, respectively, for the even and the odd transitions. The notation of transition is used since these functions represent the smooth transition from one asymptote to another; the even case does not have a point of inflection and the odd transition has a point of inflection.

In the present application, a correlation formula for the viscous-interaction parameter has been obtained by a combination of an even and odd transition function. The curve is considered to have three asymptotes (see the sketch below); Y1, F1, and F2. First an even transition is determined for the curve between F1 and F2 and this is designated Y2. Next, an odd transition is established between Y1 and Y2. The curves are adjusted through the values specified for the exponential constants, K, and the origin coordinates, X₀. Details of this procedure are given in Reference 62.



The correlation formulas developed for the present case are as follows.

Independent variable $X = \log(\epsilon^3 R_{0g})$

$$F1 = A1 + B1(X)$$

$$A1 = 0.667$$

$$B1 = 1.1111$$

$$F2 = 1.0$$

$$Y1 = 0.0$$

$$Y_2 = F_1 + \frac{(1.0 - F_1)}{1.0 - \exp \{EVK (X - X_{OEV})\}}$$

$$EVK = -1.80$$

$$X_{OEV} = -0.3$$

$$\frac{\tau_w}{\tau_{w_0}} = \frac{Y_2}{1.0 + \exp \{ODK (X - X_{OOD})\}}$$

$$ODK = -2.0$$

$$X_{OOD} = AOD + BOD(\log \epsilon)$$

$$AOD = 1.0$$

$$BOD = 3.2907$$

Comparison of this correlation and the boundary-layer solutions are shown in Figure 58. The general shape of the curves is well represented by the correlation, although some accuracy is lost, particularly at the peak of the $\epsilon = 0.04$ curve. It would be possible to tailor-fit each of the ϵ -curves through further variation in F_1 , the exponential constants and origin coordinates. However, since only three solutions were available, the determination of more accurate fits was not deemed justified. Three additional ϵ -curves are given on the figure to demonstrate the behavior of the correlation formula.

An example of this technique is shown in Figure 59 where the predicted values of lift coefficient for the Gemini space capsule are compared with experimental results (Reference 63). The modified Newtonian calculation has been performed for the entire shape and the viscous calculations (broken lines) made only for the blunt face. The present comparison, due to the limited data used, may not completely justify the method, but it does show the significance of the viscous contributions.

The blunt-body viscous calculations are not limited to entry capsules but may be applied to any blunt portions of a vehicle (e.g., leading edges). The method is primarily dependent on impact angle and, therefore, the detailed inviscid geometry is used. It is for this reason that the method has been included as one of the inviscid force options. Zero contribution is assumed for shadow flow.

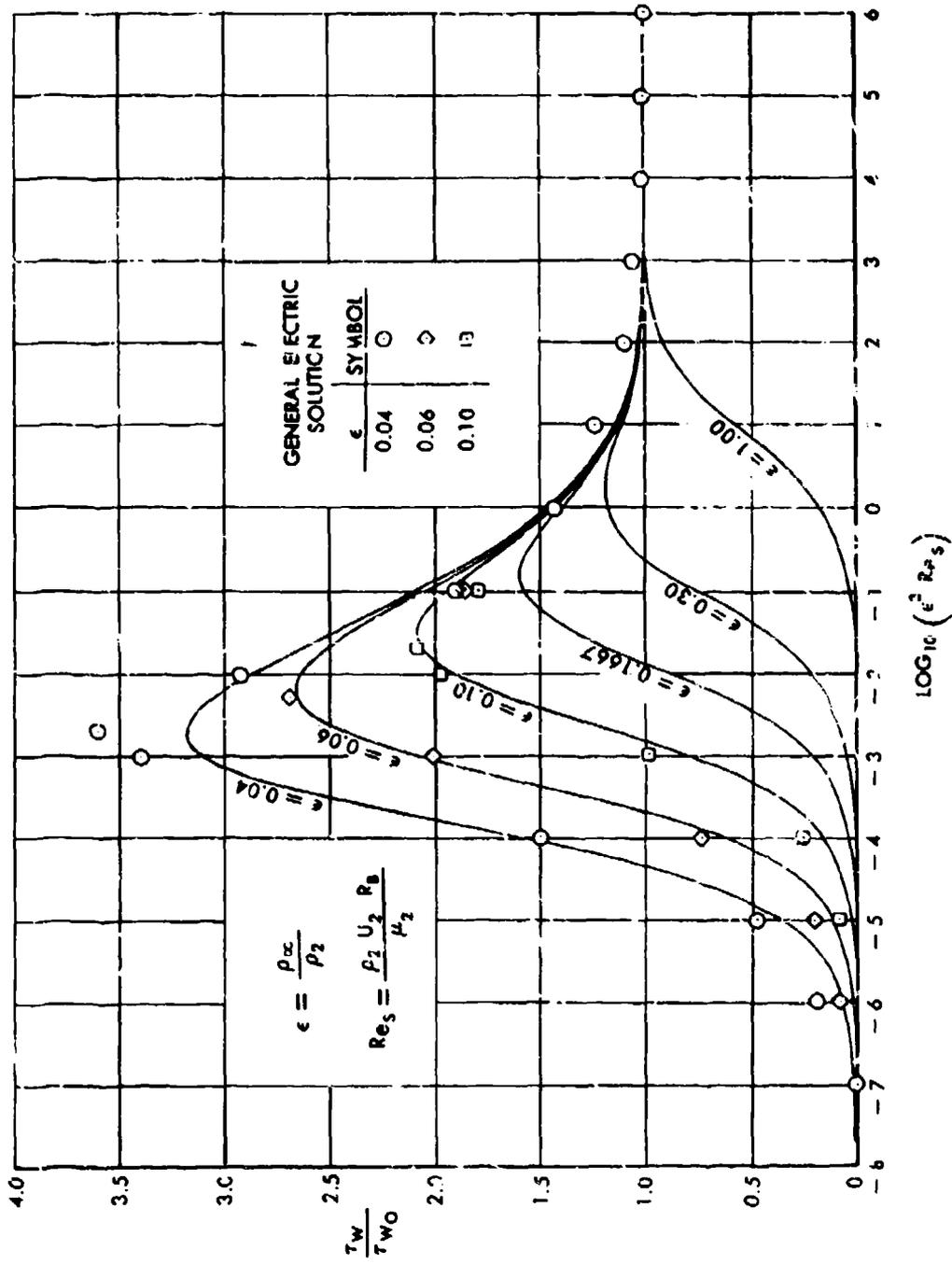


Figure 58. Low Density Correction to Blunt-Body Viscous Forces

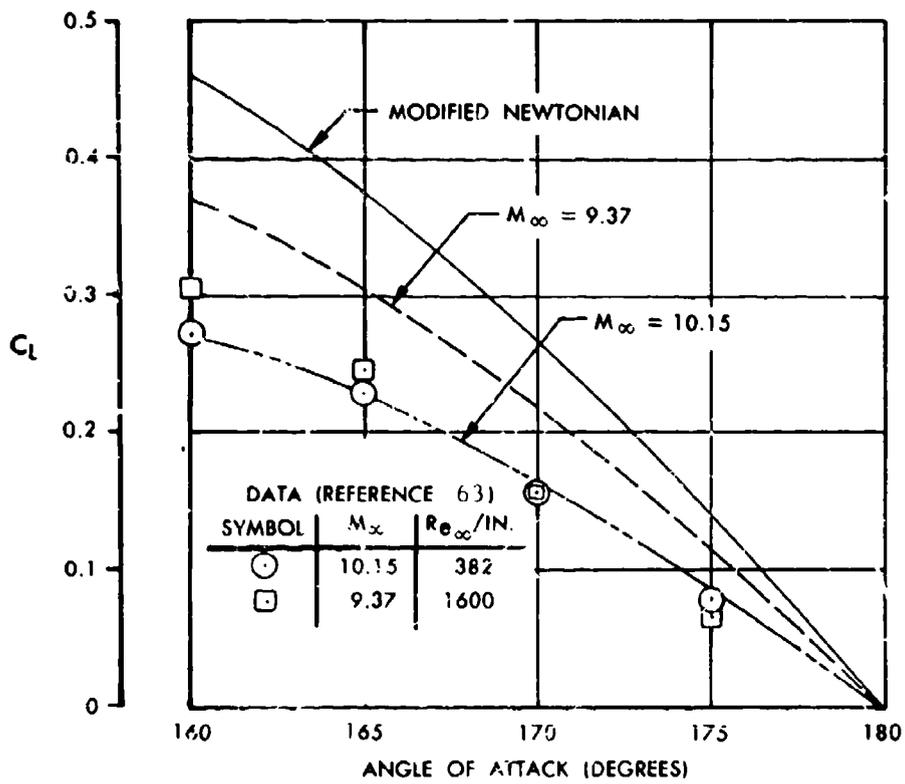
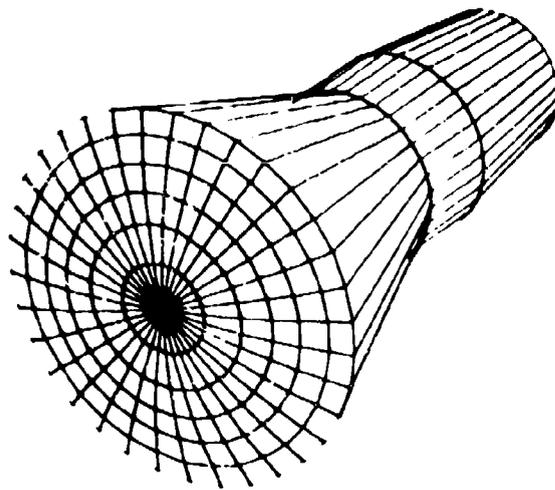


Figure 59. Gemini Lift Coefficient Comparison

SECTION XI
AUXILIARY ROUTINES

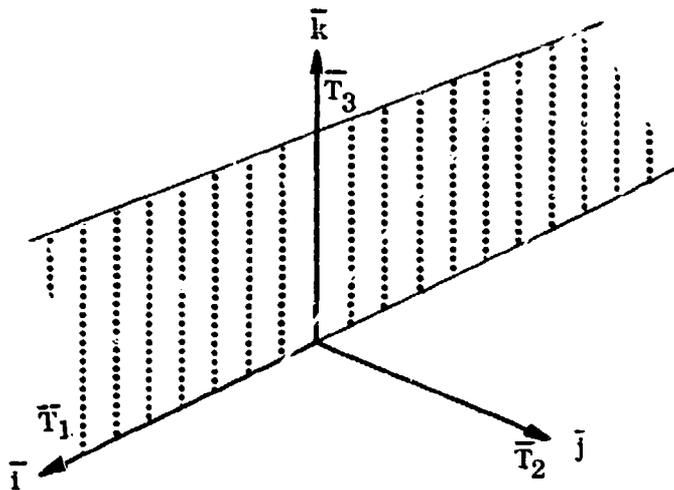
The Auxiliary Routines option of the program is provided to house features that are not directly related to one of the other major program components. In the initial release of the Mark IV program it contains only the General Cutting Plane option.

General Cutting Plane Option

The General Cutting Plane option may be used to determine the section shape of an arbitrary body in any desired plane. This capability is sometimes useful in the geometry preparation stage of a problem. In this application it can be used to help define the intersection line between intersecting vehicle panels. The cutting plane is orientated so as to represent one of the panels (or elements). Its intersection with the other panel is then determined with the General Cutting Plane option. This information can then be used to assist in preparation of the geometry data to be input in the geometry part of the program.

Orientation of the Cutting Plane

Initially the cutting plane is assumed to be in the x-z plane with its position specified by three mutually perpendicular orientation vectors; \bar{T}_1 , \bar{T}_2 , \bar{T}_3 coincident with \bar{i} , \bar{j} , \bar{k} , respectively.



The orientation of the plane is given by three rotations in a yaw-pitch-roll sequence (ψ , θ , ϕ) and by a final offset rotation β . Angle ψ is a rotation about \bar{T}_3 , angle θ is a rotation about \bar{T}_2 , angle ϕ is a rotation about \bar{T}_1 , and angle β is a rotation about \bar{T}_3 . The orientation vectors are given by

$$\begin{bmatrix} \bar{T}_1 \\ \bar{T}_2 \\ \bar{T}_3 \end{bmatrix} = [R] \begin{bmatrix} \bar{i} \\ \bar{j} \\ \bar{k} \end{bmatrix}$$

where the rotation matrix is defined as

$$[R] = [\beta] [\phi] [\theta] [\psi]$$

$[R]$ is given in complete form at the end of this section.

In most applications $\psi = \theta = 0$ and

$$[R] = \begin{bmatrix} \cos \beta & \sin \beta \cos \phi & -\sin \beta \sin \phi \\ -\sin \beta & \cos \beta \cos \phi & -\cos \beta \sin \phi \\ 0 & \sin \phi & \cos \phi \end{bmatrix}$$

This whole business seems somewhat awkward, but has been selected with the applications in mind. The cross-section areas needed for wave drag calculations are determined by cutting planes having $\beta = \text{Mach angle}$. The cutting planes are distributed along the body and taken over the necessary viewing angles (ϕ).

Meridian sections for shock expansion calculations are determined with $\beta = 0$ and varying the roll angle ϕ .

Intersection of Cutting Planes and Configuration Geometry

It is expected that analyses will be required of complex configurations consisting of a great many elements and therefore any geometry handling, such as section cuts, must be done efficiently. The Douglas Arbitrary Wave Drag Program is an example of this. Earlier programs select a cross section plane and search the geometry for possible intersections. As the geometry descriptions become more complex, it is apparent that large amounts of time are simply wasted in searching. The Douglas Program, however, selects an element and determines any and all intersections involving that element. The configuration elements are cycled and the intersections collected according to cross-section - the same result as earlier programs. The big difference is that each element is "called" only once.

The intersections are found by projecting the element into a plane normal to the cutting plane. For cross section cuts a plane containing \bar{T}_1, \bar{T}_2 is convenient, while for meridian cuts a plane containing \bar{T}_2, \bar{T}_3 is more suitable. The procedure for meridian cuts is described below.

A plane is completely described by its normal vector and a point in the plane. The sketches shown have implicitly assumed this point on the x-axis and, in particular, at the coordinate origin for meridian cuts. An offset for the plane location (x_0, y_0, z_0) is easily accounted for and will so

be assumed in the following discussion. For the definition established, the three vectors associated with the cutting plane are ($\beta = 0$);

$$\bar{T}_1 = \cos \theta \cos \psi \bar{i} + \cos \theta \sin \psi \bar{j} - \sin \theta \bar{k}$$

$$\bar{T}_2 = -(\cos \phi \sin \psi + \sin \phi \sin \theta \cos \psi) \bar{i} + (\cos \phi \cos \psi - \sin \phi \sin \theta \sin \psi) \bar{j} - \sin \phi \cos \theta \bar{k}$$

$$\bar{T}_3 = -(\sin \phi \sin \psi - \cos \phi \sin \theta \cos \psi) \bar{i} + (\sin \phi \cos \psi + \cos \phi \sin \theta \sin \psi) \bar{j} + \cos \phi \cos \theta \bar{k}$$

Once the plane axis (\bar{T}_1) is set (ψ, θ values given), the projection plane is fixed and for simplicity the \bar{T}_2, \bar{T}_3 plane at $\phi = 0$ is used. The coordinate axis of the projection plane, designated $\bar{Y}\bar{P}$ and $\bar{Z}\bar{P}$ are

$$\bar{Y}\bar{P} = (\bar{T}_2)_{\phi=0} = -\sin \psi \bar{i} + \cos \psi \bar{j}$$

$$\bar{Z}\bar{P} = (\bar{T}_3)_{\phi=0} = \sin \theta \cos \psi \bar{i} + \sin \theta \sin \psi \bar{j} + \cos \theta \bar{k}$$

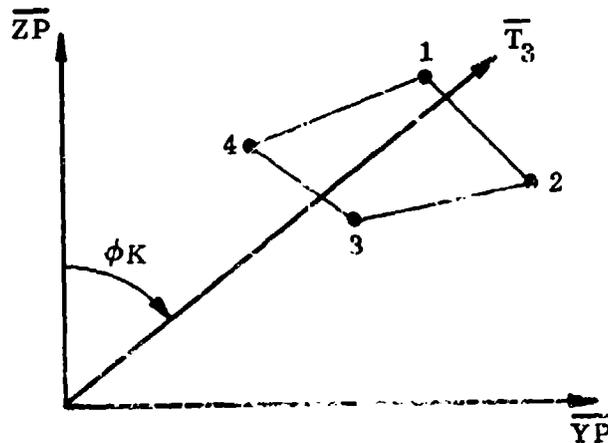
The corner points of the quadrilateral are projected into the $Y\bar{P}, Z\bar{P}$ plane. The radius vector to a corner point is

$$R(N) = X(N)\bar{i} + Y(N)\bar{j} + Z(N)\bar{k}$$

and the projected components are

$$Y\bar{P}(N) = \bar{R}(N) \cdot \bar{Y}\bar{P} = -X(N) \sin \psi + Y(N) \cos \psi$$

$$Z\bar{P}(N) = \bar{R}(N) \cdot \bar{Z}\bar{P} = X(N) \sin \theta \cos \psi + Y(N) \sin \theta \sin \psi + Z(N) \cos \theta$$



be assumed in the following discussion. For the definition established, the three vectors associated with the cutting plane are ($\beta = 0$);

$$\bar{T}_1 = \cos \theta \cos \psi \bar{i} + \cos \theta \sin \psi \bar{j} - \sin \theta \bar{k}$$

$$\bar{T}_2 = -(\cos \phi \sin \psi + \sin \phi \sin \theta \cos \psi) \bar{i} + (\cos \phi \cos \psi - \sin \phi \sin \theta \sin \psi) \bar{j} - \sin \phi \cos \theta \bar{k}$$

$$\bar{T}_3 = -(\sin \phi \sin \psi - \cos \phi \sin \theta \cos \psi) \bar{i} + (\sin \phi \cos \psi + \cos \phi \sin \theta \sin \psi) \bar{j} + \cos \phi \cos \theta \bar{k}$$

Once the plane axis (\bar{T}_1) is set (ψ, θ values given), the projection plane is fixed and for simplicity the \bar{T}_2, \bar{T}_3 plane at $\phi = 0$ is used. The coordinate axis of the projection plane, designated $\bar{Y}\bar{P}$ and $\bar{Z}\bar{P}$ are

$$\bar{Y}\bar{P} = (\bar{T}_2)_{\phi=0} = -\sin \psi \bar{i} + \cos \psi \bar{j}$$

$$\bar{Z}\bar{P} = (\bar{T}_3)_{\phi=0} = \sin \theta \cos \psi \bar{i} + \sin \theta \sin \psi \bar{j} + \cos \theta \bar{k}$$

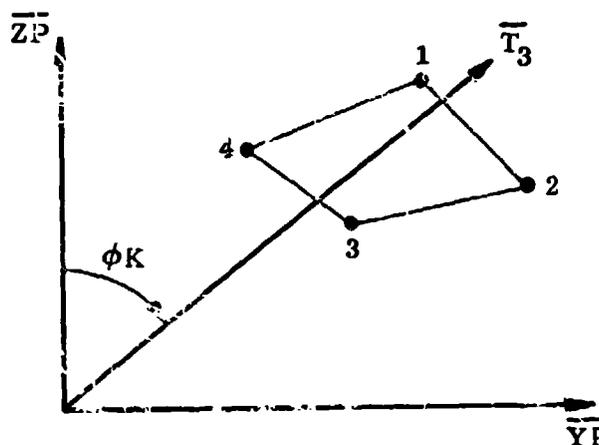
The corner points of the quadrilateral are projected into the $\bar{Y}\bar{P}, \bar{Z}\bar{P}$ plane. The radius vector to a corner point is

$$\bar{R}(N) = X(N)\bar{i} + Y(N)\bar{j} + Z(N)\bar{k}$$

and the projected components are

$$\bar{Y}\bar{P}(N) = \bar{R}(N) \cdot \bar{Y}\bar{P} = -X(N) \sin \psi + Y(N) \cos \psi$$

$$\bar{Z}\bar{P}(N) = \bar{R}(N) \cdot \bar{Z}\bar{P} = X(N) \sin \theta \cos \psi + Y(N) \sin \theta \sin \psi + Z(N) \cos \theta$$



The four corner points (N = 1, 4) are projected into this plane and the desired intersections are with the \bar{T}_3 vectors, which in this plane is simply

$$\bar{T}_3 = \sin \phi_K \bar{Y}P + \cos \phi_K \bar{Z}P$$

The subscript K is used to indicate a number of planes, say 1 to M.

The angular position of each corner point is determined

$$A(N) = \tan^{-1} (ZP(N)/YP(N))$$

and the ϕ_K are interrogated to find the condition

$$\phi_K < A_n < \phi_{K+1}$$

each corner point is assigned a plane number equal to K:

$$MP(N) = K$$

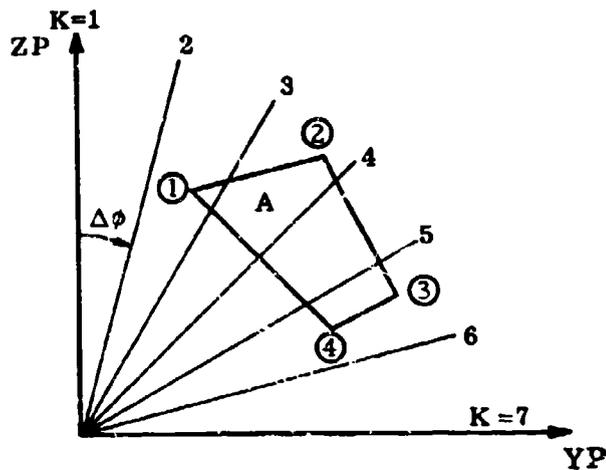
The plane numbers of successive corner points are tested for possible intersections and if indicated the intersections are found. The number of intersections on the line segment between two points is given by the difference in plane numbers:

$$NI_{N, N+1} = | MP(N) - MP(N+1) |$$

An example will help to demonstrate the procedure. First, an additional simplification is presented. If the meridian planes are desired for equal increments in $\Delta\phi$, the corner point plane number (MP) is given directly by

$$MP(N) = [\tan^{-1} (ZP(N)/YP(N)) / \Delta\phi]$$

where integer arithmetic is assumed (i. e., the value of MP is truncated to next lower integer). For the example then, consider equally spaced meridian planes as shown in the sketch and calculate intersections with element A.



Corner Point Number N	Plane Number MP(N)	Line Segment	Number of Intersections MP(N) - MP(N+1)
1	2		
2	3	1-2	1
3	5	2-3	2
4	5	3-4	0
		4-1	3

The results are presented in the table and are easily verified from the sketch. It should be noted that an element line coincident with a cutting plane does not produce an intersection. Rather, this is recorded by the "cross-member".

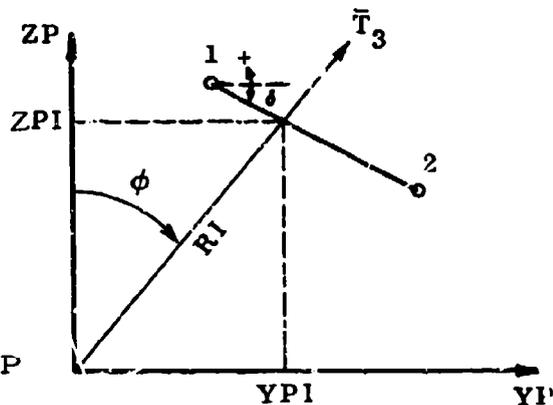
Calculation of Intersection, Point I.

The equation of the cutting plane projection is

a) $YP = \tan \phi \cdot ZP$

and the equation of projected line segment is

b) $ZP = (ZP_1 - \tan \delta \cdot YP_1) + \tan \delta \cdot YP$

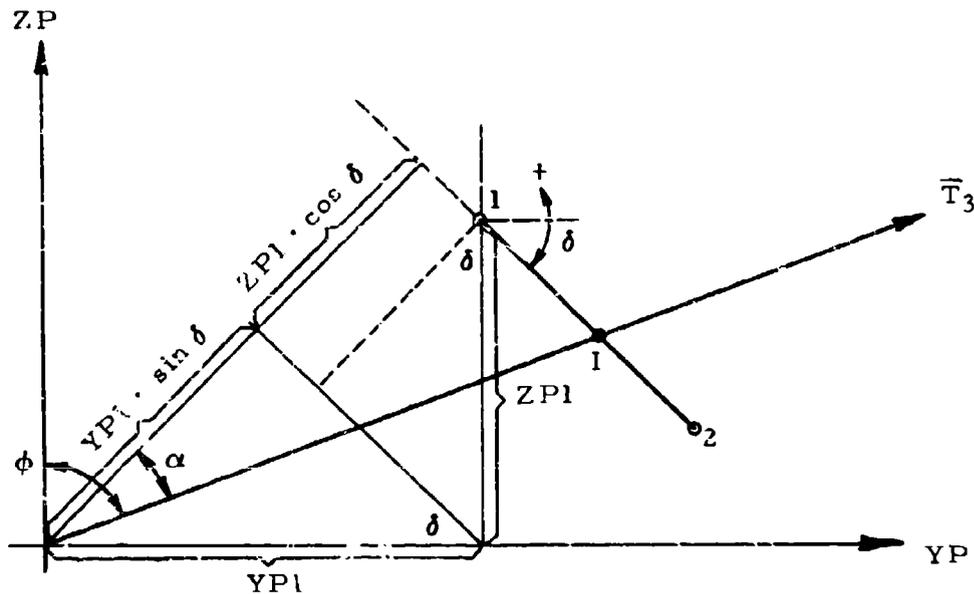


The solution for the intersection of two linear equations is easily found by straightforward equalities (i. e., $YP_{1a} = YP_{1b}$ and $ZP_{1a} = ZP_{1b}$). To avoid certain singularities however, the resulting equations may be reworked algebraically. Therefore, a geometric interpretation of the solution which

accounts for the singularities is presented. The coordinates of the intersection (YPI, ZPI) are given in terms of radius length of the intersection, RI:

$$YPI = RI * \sin \phi$$

$$ZPI = RI * \cos \phi$$



Observing the construction in the above sketch and noting that $\alpha = \phi + \delta$, the radius length of the intersection is

$$RI = (ZP1 \cdot \cos \delta \cdot YP1 \cdot \sin \delta) / \cos (\phi + \delta)$$

where δ has the sign convention given by positive slope in YP, ZP coordinates (i.e., sign of dZP/dYP). This may be clarified by rewriting in terms of the point coordinates YP1, ZP1 and YP2, ZP2. First expand $\cos(\phi + \delta) = \cos \phi \cos \delta - \sin \phi \sin \delta$ and then multiply through by the length of the line segment

$$RI = \frac{ZP1 \cdot (YP2 - YP1) - YP2 \cdot (ZP2 - ZP1)}{\cos \phi \cdot (YP2 - YP1) - \sin \phi \cdot (ZP2 - ZP1)}$$

This equation is well bel. ved since the line segment and the cutting plane cannot be coincident. The intersection in body coordinates is calculated directly using the principle of proportional parts. The length ratio is called as

$$LR = \left[\frac{(Y_{P1} - Y_{P1})^2 + (Z_{P1} - Z_{P1})^2}{(Y_{P2} - Y_{P1})^2 + (Z_{P2} - Y_{P1})^2} \right]^{1/2}$$

$$= \frac{Y_{P1} - Y_{P1}}{Y_{P2} - Y_{P1}} = \frac{Z_{P1} - Z_{P1}}{Z_{P2} - Z_{P1}}$$

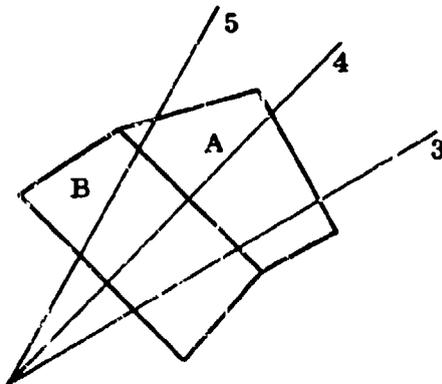
then

$$X_I = X_1 + (X_2 - X_1) * LR$$

$$Y_I = Y_1 + (Y_2 - Y_1) * LR$$

$$Z_I = Z_1 + (Z_2 - Z_1) * LR$$

Since many elements are involved the question naturally arises as to whether the same intersection is calculated twice. That is, in the preceding example, since line segment 4-1 has three intersections would these same intersections be recalculated for the element B which also contains this segment?



This duplication is avoided by selecting the line segments used for any given element. The selection is based on the way the element points are ordered. Briefly, recalling from Section III, elements are loaded

by section according to rows and columns. Each element has a number and a status flag and all this information is available. A sample section of elements is illustrated below.

3	②	③	②	③		
	ⓐ		ⓑ			
2	①	④	①	④		
	ⓐ		ⓑ			
Row J = 1	②	③	②	③	etc.	
	ⓐ		ⓑ			
Column I =	1	2	3	4		
Status Flag	2					

Corner Point Number ①

Element Number ⓐ

A scheme to project the line segments is as follows:

I ≠ 1 project lines 2-3 and 3-4

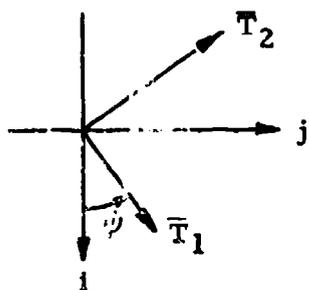
J = 1 also project line 4-1

I = 1 also project line 1-2

With this scheme, duplication is limited to section boundaries.

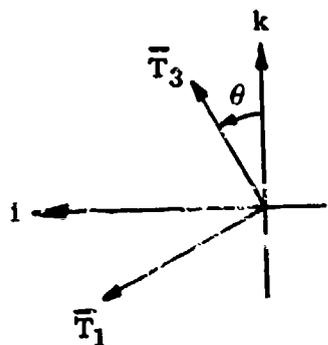
Rotation Matrix [R]

ψ rotation about \bar{T}_3



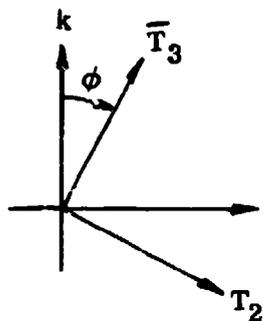
$$[\psi] = \begin{bmatrix} \cos \psi & \sin \psi & 0 \\ -\sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

θ rotation about \bar{T}_2



$$[\theta] = \begin{bmatrix} \cos \theta & 0 & -\sin \theta \\ 0 & 1 & 0 \\ \sin \theta & 0 & \cos \theta \end{bmatrix}$$

ϕ rotation about \bar{T}_1



$$[\phi] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & -\sin \phi \\ 0 & \sin \phi & \cos \phi \end{bmatrix}$$

β rotation about T_3

$$[\beta] = \begin{bmatrix} \cos \beta & \sin \beta & 0 \\ -\sin \beta & \cos \beta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

The vectors $\bar{T}_1, \bar{T}_2, \bar{T}_3$ are given by

$$\begin{bmatrix} \bar{T}_1 \\ \bar{T}_2 \\ \bar{T}_3 \end{bmatrix} = [R] \begin{bmatrix} \bar{i} \\ \bar{j} \\ \bar{k} \end{bmatrix}$$

where $[R] = [\beta][\phi][\theta][\psi]$

$$[R] = \left\{ \begin{array}{l} [\cos\beta\cos\theta\cos\psi - \sin\beta(\cos\phi\sin\psi + \sin\phi\sin\theta\cos\psi)] \\ [\cos\beta\cos\theta\sin\psi + \sin\beta(\cos\phi\cos\psi - \sin\phi\sin\theta\sin\psi)] \\ -[\cos\beta\sin\theta + \sin\beta\sin\phi\cos\theta] \\ \\ -[\sin\beta\cos\theta\cos\psi + \cos\beta(\cos\phi\sin\psi + \sin\phi\sin\theta\cos\psi)] \\ [\sin\beta\cos\theta\sin\psi + \cos\beta(\cos\phi\cos\psi - \sin\phi\sin\theta\sin\psi)] \\ [\sin\beta\sin\theta - \cos\beta\sin\phi\cos\theta] \\ \\ -[\sin\phi\sin\psi - \cos\phi\sin\theta\cos\psi][\sin\phi\cos\psi + \cos\phi\sin\theta\sin\psi][\cos\phi\cos\theta] \end{array} \right\}$$

SECTION XII

COMPUTER GRAPHICS

The use of computer graphics has been a key feature of the Arbitrary-Body Program since its inception. The first picture drawing routine was prepared as early as 1965 (for the Mark II program). The first on-line graphics program using the IBM 2250 was prepared in 1966. The use and importance of graphics capabilities in checking out geometry data has been adequately documented in a number of references by users of the Mark III version of the program. The techniques used in the old Mark III program and its supporting on-line graphics programs have been adapted and used by a number of organizations in the development of their own graphics programs. Versions of the Mark III Picture Drawing Program have been prepared for several different types of hardcopy devices including the SC-4020, SD-4060, CALCOMP, and the Gerber Plotter. Since most users of the new Mark IV program already have picture drawing programs developed for the support of the old Mark III program it has not been necessary to include a graphics program within the new program on its initial release.

The use of interactive graphics, such as is possible with the IBM 2250 and the CDC 274 cathode ray tube equipment in combination with suitable hardcopy or camera equipment, is by far the most efficient type of computer graphics operation. The term "interactive graphics" implies that the engineer has direct and real time control of the operation of a graphics program (the selection and manipulation of input data, program options, viewing angles, output data, etc.). However, the use of interactive graphics in an engineering application should be carefully weighed against the cost of program development and the very high cost of program operation.

The use of computer graphics to checkout arbitrary-body geometry data is well understood. However, the use of graphics in presenting flow field data is a relatively new development. As was the case with the geometry picture drawing in the beginning we should ask the question why? Why do we need flow field graphics? For the geometry problem it was to check the input shape data. For the flow field data problem computer graphics can be used to monitor intermediate program output data. The objective here would be to show in a graphical form the vehicle generated shock waves, embedded flow boundaries, and surface streamlines.

A very simple scheme has been used in the latest version of the Douglas IBM 2250 Graphics Program to allow the easy addition of flow field plotting capability. The trick used is to convert the flow field information into standard geometry data format (complete with status flags, etc.). It is then possible to submit the flow field data to the picture drawing program just as though it were geometry data. The addition of a dotted line routine under the control of the program operator helps to separate the flow field data from the actual geometry data.

The series of pictures shown in Figures 60 through 62 illustrate the operation of this program. Figure 60 shows the body generated flow field and Figure 61 the wing field. A side view of the vehicle flow field at 5° angle of attack is shown in Figure 62.

The flow field data necessary to construct these types of pictures are generated in the flow field option of the Mark IV program and stored on the flow field storage unit 10. The manner in which these data are stored is described in Section II of Volume III. Users wishing to modify their own graphics programs to produce flow field pictures should study this section carefully.

The following discussion and derivations related to computer graphics are presented in the interest of completeness and to aid new users of the Arbitrary Body Program system. Some new suggested features for existing graphics programs are also discussed.

Picture Drawing Methods

As explained in Section III, the geometric shape of a vehicle is defined by input sets of points in three-dimensional space. A grouping of four surface points is used to describe a surface element. An organization of a large number of related elements forms a body panel and a number of panels describe a vehicle component. Several components are usually used to make up the complete vehicle. The equations required to produce perspective drawings of the geometry data are derived in the following paragraphs.

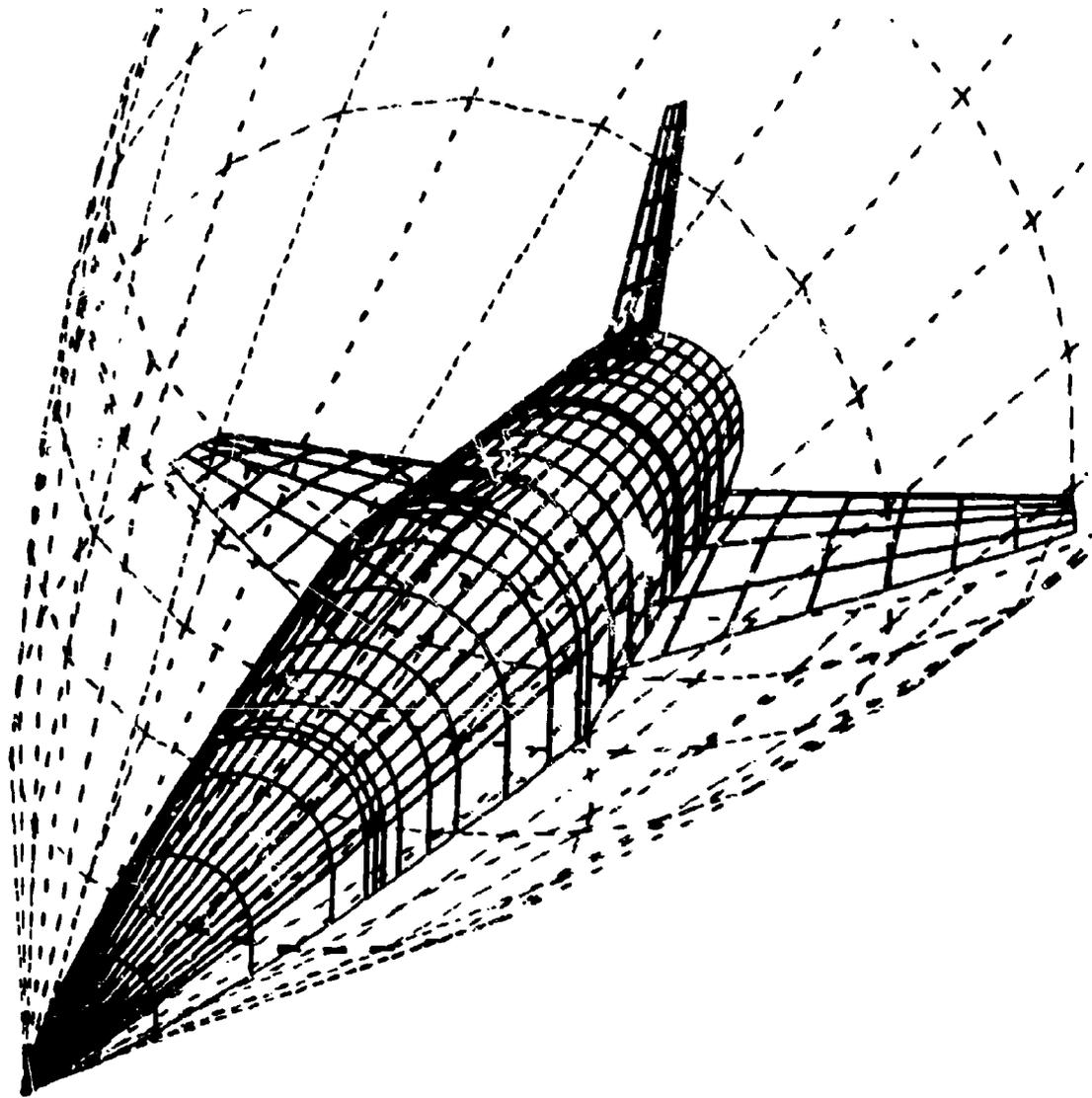


Figure 60. Body Generated Flow Field

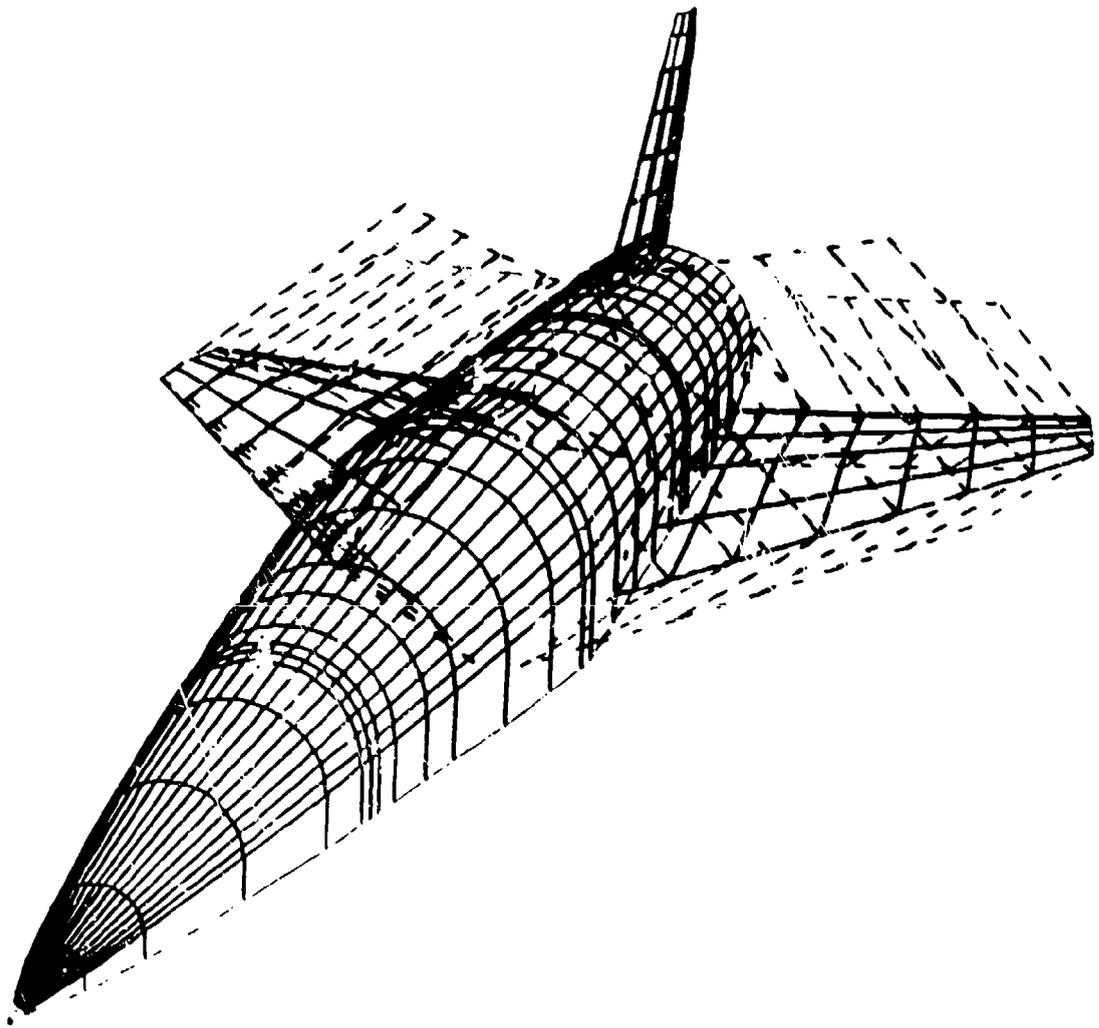


Figure 01. Wing Generated Flow Field.

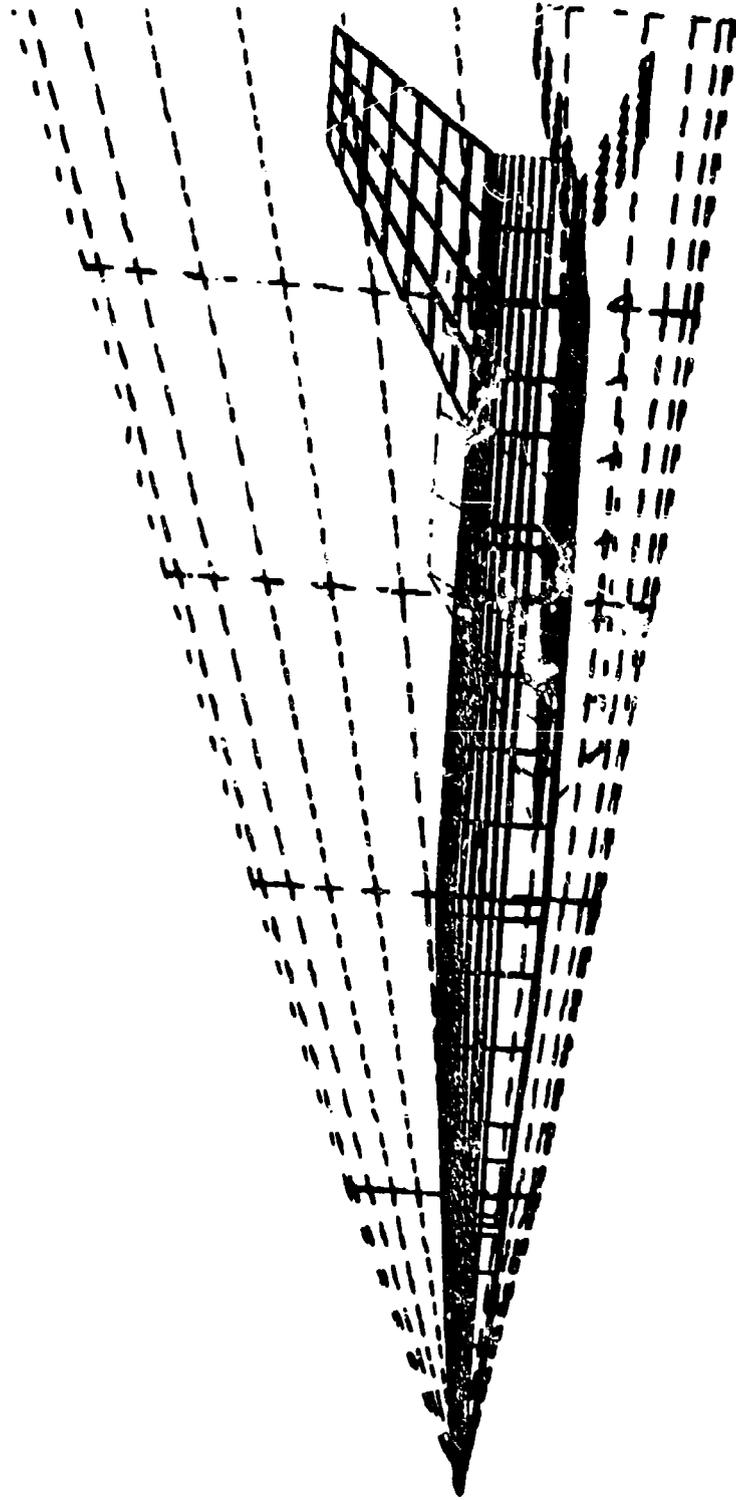
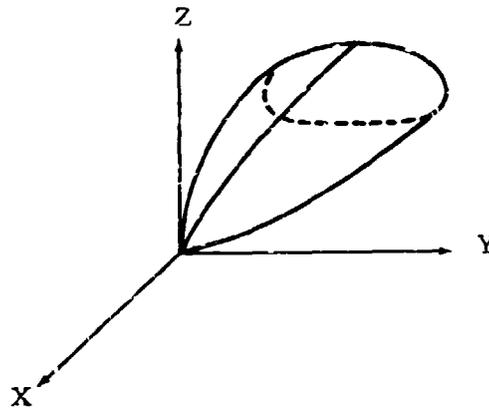


Figure 62. Side View of Vehicle Flow Field at 5° Angle of Attack.

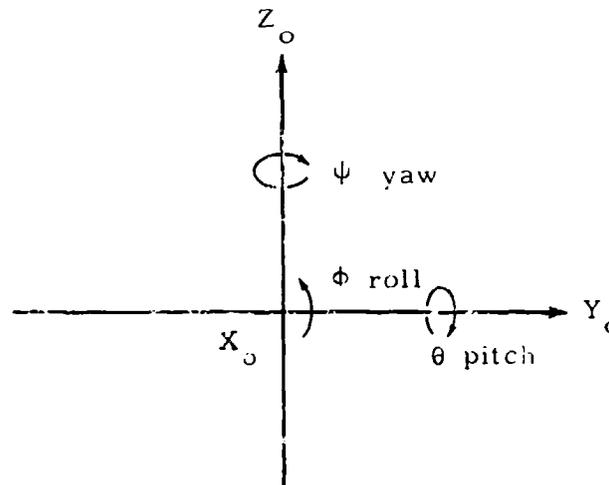
Each point on the surface is described by its coordinates in the body reference coordinate system.

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

The body reference coordinate system is assumed to be a conventional right-handed Cartesian system as illustrated below.



To create the perspective drawings illustrated in this report each surface point on the body must be rotated to the desired viewing angle and then transformed into a coordinate system in the plane of the paper. With zero rotation angles the body coordinate system is coincident with the fixed system in the plane of the paper.



The rotations of the body and its coordinate system to give a desired viewing angle are specified by a yaw-pitch-roll sequence (ψ, θ, ϕ) . This rotation is given by the following relationship:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} \phi \end{bmatrix} \begin{bmatrix} \theta \end{bmatrix} \begin{bmatrix} \psi \end{bmatrix} \begin{bmatrix} X_0 \\ Y_0 \\ Z_0 \end{bmatrix}$$

Where the rotation matrices are

$$\begin{bmatrix} \psi \end{bmatrix} = \begin{bmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} \theta \end{bmatrix} = \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix}$$

$$\begin{bmatrix} \phi \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{bmatrix}$$

or

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} E \end{bmatrix} \begin{bmatrix} X_0 \\ Y_0 \\ Z_0 \end{bmatrix}$$

where $\begin{bmatrix} E \end{bmatrix} = \begin{bmatrix} \phi \end{bmatrix} \begin{bmatrix} \theta \end{bmatrix} \begin{bmatrix} \psi \end{bmatrix}$

Since each point on the surface is given by its coordinates in the X, Y, Z system, its position in the fixed coordinate system (X_0, Y_0, Z_0) may be found by reversing the above process.

$$\begin{bmatrix} X_0 \\ Y_0 \\ Z_0 \end{bmatrix} = \begin{bmatrix} E \end{bmatrix}^{-1} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

If we carry out this operation we obtain

$$\begin{bmatrix} X_0 \\ Y_0 \\ Z_0 \end{bmatrix} = \begin{bmatrix} \cos\theta\cos\psi & -\sin\psi & \cos\phi + \sin\theta\cos\psi & \sin\phi & \sin\psi & \sin\phi + \sin\theta\cos\psi & \cos\phi \\ \cos\theta\sin\psi & \cos\psi & \cos\phi + \sin\theta\sin\psi & \sin\phi & -\cos\psi & \sin\phi + \sin\theta\sin\psi & \cos\phi \\ -\sin\theta & & \cos\theta\sin\phi & & & \cos\theta\cos\phi & \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

$$X_0 = X(\cos\theta\cos\psi) + Y(-\sin\psi\cos\phi + \sin\theta\cos\psi\sin\phi) + Z(\sin\psi\sin\phi + \sin\theta\cos\psi\cos\phi)$$

$$Y_0 = X(\cos\theta\sin\psi) + Y(\cos\psi\cos\phi + \sin\theta\sin\psi\sin\phi) + Z(-\cos\psi\sin\phi + \sin\theta\sin\psi\cos\phi)$$

$$Z_0 = X(-\sin\theta) + Y(\cos\theta\sin\phi) + Z(\cos\theta\cos\phi)$$

We may now use these last two equations to transform a given point on the body (X, Y, Z) with a specified set of rotation angles (ψ, ϕ, θ) into the plane of the paper (the Y_0, Z_0 system). With the available graphics subroutines it now becomes a simple matter to plot these data and to connect the related points with straight lines.

In the surface fit technique used in this program and described in Reference 3, each input element is replaced by a plane quadrilateral surface element whose characteristics are used for all subsequent calculations. These characteristics include the area, centroid, and the direction cosines of the surface unit normal. The surface unit normals may be transformed through the required rotation angles just as was done for the individual points. The resulting value of the component of the unit normal in the X_0 direction (out of the plane of the paper) may be found from the following equation.

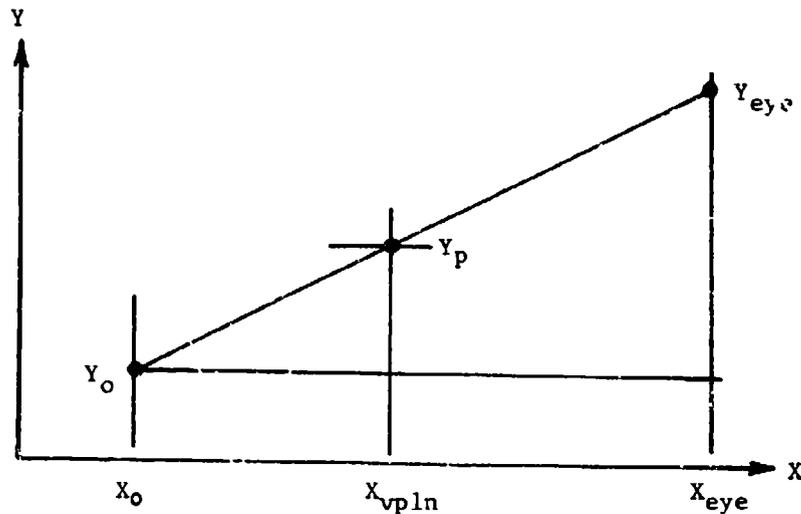
$$n_{x_0} = n_x(\cos\theta\cos\psi) + n_y(-\sin\psi\cos\phi + \sin\theta\cos\psi\sin\phi) + n_z(\sin\psi\sin\phi + \sin\theta\cos\psi\cos\phi)$$

where n_x, n_y, n_z are the components of the surface unit normal in the vehicle reference system.

If n_{x_0} is positive then the surface element is facing the viewer. If n_{x_0} is negative the element faces away from the plane of the paper. This result is used in the program to provide the capability of deleting most of those elements on a vehicle that normally could not be seen by a viewer. The resulting picture is thus made more realistic and confusing elements which are on the back side of the vehicle do not appear. No criterion is provided, however, for the deletion of those elements that face the viewer but are blocked by other body components. This may be accomplished by a proper selection of viewing angle or by a physical deletion of the offending section from the input data.

The pictures generated by the above procedures are not true perspective pictures but represent the limiting condition where the viewing "eyeball" is positioned at infinity. That is, the picture does not exhibit any foreshortening that exists when a three-dimensional object is viewed close at hand. These types of pictures are certainly acceptable for the purpose for which they were intended - to check out geometry data. However, for some applications it may be desirable to have the computer produce true perspective pictures. The procedures necessary to accomplish this are discussed below.

In past arbitrary-body picture drawing programs the picture has been drawn in the Y-Z plane, with the X-axis projecting out of the picture screen and, therefore, not affecting the resulting image. To obtain a true perspective image it is necessary to know the position of the imaginary eyeball (or camera lens) relative to the rotated position of the shape. A viewing ray is assumed to exist between each point on the shape and the eyeball. The true perspective image is then formed by determining where these rays pass through a viewing plane placed parallel to the Y-Z plane and between the shape and the eyeball. The position of the viewing plane is not important as long as it is outside the rotated shape. The closer the viewing plane is to the eyeball, the smaller the perspective image. This process is illustrated below.



The resulting equations for the image position of a point are as follows.

$$Y_p = Y_0 + ((Y_0 - Y_{eye}) / (X_0 - X_{eye})) * (X_{vpln} - X_0)$$

$$Z_p = Z_0 + ((Z_0 - Z_{eye}) / (X_0 - X_{eye})) * (X_{vpln} - X_0)$$

A FORTRAN subroutine to accomplish the perspective conversion is shown below. A sample picture produced by this process is shown in Figure 63.

```

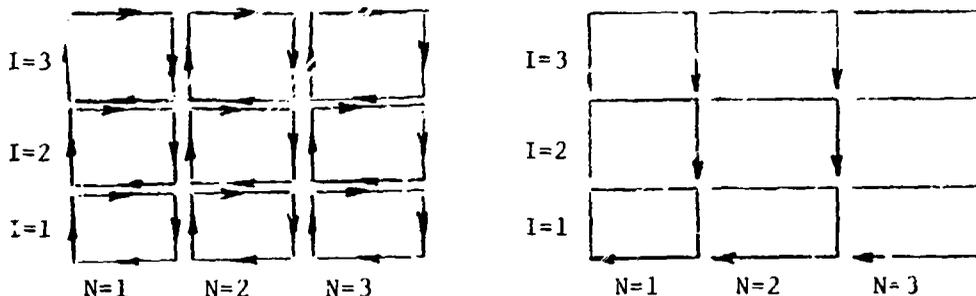
SUBROUTINE PERSPC
  DIMENSION XO (4), YO(4),ZO(4)
  COMMON /PER/XO, YO, ZO, XEYE, YEYE, ZEYE, XVPLN
  C DO LOOP TO CONVERT FOUR ELEMENT CORNER POINTS
    TO PERSPECTIVE
  DO 10 I=1, 4
    CORR = (XVPLN-XO(I)) / (XO(I)-XEYE)
  20 Y(I) = YO(I) + (YO(I)-YEYE) * CORR
  20 Z(I) = ZO(I) + (ZO(I)-ZEYE) * CORR
  10 CONTINUE
  RETURN
  END

```

Past arbitrary-body graphics programs have drawn each element separately. That is, each element was drawn independently by the graphics device. This meant that the common line between adjacent elements was drawn twice. This did not present a problem on hard copy devices such as the SD-4060 or CALCOMP. However, on some machines such as the IBM 2250 it may not be possible to get all elements of a vehicle shape drawn on the screen at one time because of machine vector storage limitations. However, a method has been worked out to avoid this duplicate drawing of lines between adjacent elements and some users may wish to modify their existing programs accordingly. The procedure used is quite simple and involves the use of information already available in the geometry analysis part of most arbitrary-body programs. The basic method of identifying the points of an element (and also of drawing the element in the graphics program) involves a clockwise numbering of the points as shown below.



A single panel containing several elements is described below. The left figure shows the elements as they were originally drawn by the program. The right figure illustrates the lines that are used in drawing the same elements in the new version of the program.



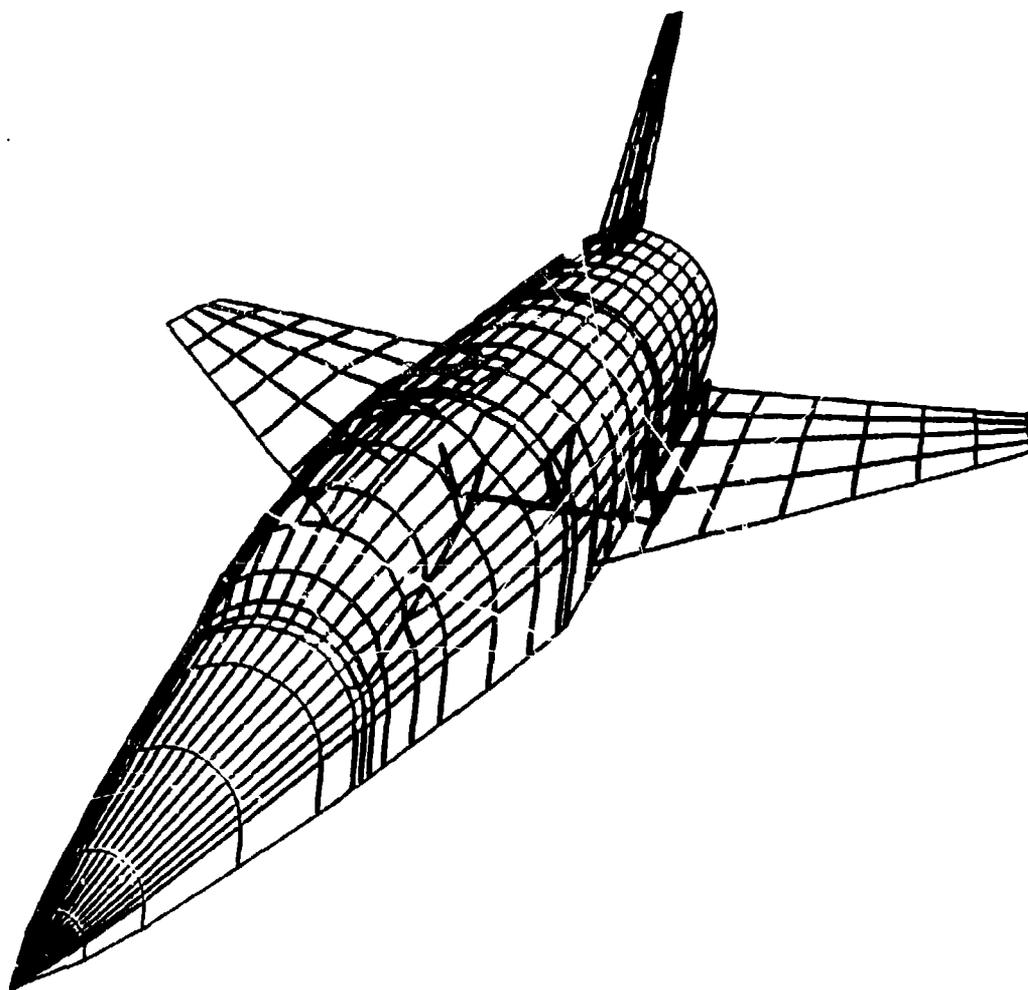


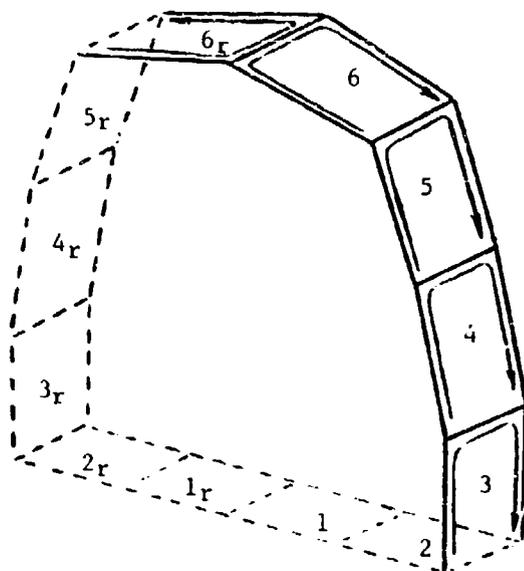
Figure 63. True Perspective with Arbitrary-Body Picture Drawing Program.

Each vertical column of elements is identified by the parameter N. The element number in a given column is identified by the parameter I. Both of these parameters are available in the part of the program that generates the quadrilaterals from the input data.

In the diagram at the right note that line 1-2 for each element is drawn only when $N=1$. Line 2-3 is drawn for all values of N and I, as is line 3-4. Line 4-1 is drawn for $I=1$ only. It would be a simple matter to store the N and I parameters in the bead along with the Y-Z values, and to check these values in the DISPLAY routine to determine which lines are to be drawn.

The above procedure will avoid the duplication of line drawing within a given section of a vehicle (until the next STATUS=2 is reached). Adjacent vehicle sections will still have some duplication of lines at the section edges. However, it is not worth the effort to try to develop a scheme to avoid this.

Application of the above scheme of checking on the N and I parameters will also, at times, leave a line undrawn that we actually would like to have in place. This occurs at the edges of pictures as is illustrated for a single cross-section of elements in the diagram below.



In the above drawing the subscript r is used to indicate a reflected element due to shape symmetry. Elements 1, 2 and 1_r through 5_r are not drawn because they do not face the viewer. Note that one side of element 3 will not be drawn (using the N and I parameter checks alone) since side 4-1 would normally be drawn if element 2 faced the viewer. Side 4-1 of element 6_r is not drawn for the same reason.

The above situation can be corrected by introducing two new flags to supplement the N-I parameter checks. These two flags (IN and INR in our program) are used to indicate when the preceding element on that same side of the vehicle was not drawn because it did not face the viewer. Each time an input element is drawn the IN parameter is set to zero. If it is not drawn because it does not face the viewer (NXO.LE. 0.0) then it is set to 1. Elements on the reflected side of the symmetry plane are handled in the same manner with the INR flag. A third flag, IFLAG is used to indicate when we are on the input element side (IFLAG=0) and when we are on the reflected side (IFLAG=1). The complete checking procedure for the above situations is as follows.

Draw line 1-2 when

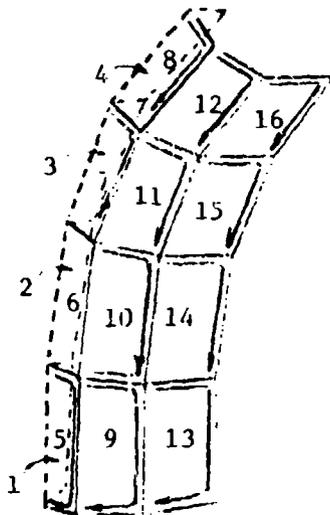
N = 1 or
IFLAG = 0 and IN = 1 or
IFLAG = 1 and INR = 1

Draw line 4-1 when

I = 1 or
IFLAG = 0 and IN = 1 or
IFLAG = 1 and INR = 1

Draw lines 2-3 and 3-4 for all conditions

The preceding checking procedure will produce complete pictures with a minimum amount of line duplication for most vehicle shapes. However, there is still another situation that has not been handled by these checks. This duplication is shown in the drawing below.



This situation has not been corrected as yet in the Douglas program. The solution involves the addition of yet another set of flags similar in concept to the IN and INR flags. Only this time, the flags will have to be subscripted arrays. The procedure essentially involves a check to see if the last element that had the same value if I was drawn or not. For example, in the drawing above, if element 1 was not drawn then side 1-2 of element 5 must be drawn. The same applies to elements 2-6, 3-7, and 4-8.

REFERENCES

1. Gentry, A. E., and Smyth, D. N., Hypersonic Arbitrary-Body Aerodynamic Computer Program, Mark III Version, Volume I User's Manual, Volume II Program Formulation and Listing, Douglas Report DAC 61552, April 1968.
2. Gentry, A. E., Hypersonic Arbitrary-Body Aerodynamic Computer Program (Mark II), Volume I - User's Manual and Volume II - Program Formulation and Listings, Douglas Report No. DAC 56080, March 1967.
3. Hess, J. L. and Smith, A. M. O., Calculation of Non-Lifting Potential Flow About Arbitrary Three-Dimensional Bodies, Douglas Report No. ES 40622, March 1962.
4. Coons, S. A., Surfaces for Computer-Aided Design of Space Figures, Massachusetts Institute of Technology Preprint No. 299.
5. Harder, R. L., and Desmarais, R. N., Interpolation Using Surface Splines, Journal of Aircraft, Volume 9, No. 2, February 1972, pp. 189-191.
6. Hess, J. L. and Riddell, T. M., Direct Solutions of a Square Matrix Whose Size Greatly Exceeds High-Speed Storage, McDonnell Douglas Report DAC 70000, July 1969.
7. Kaattari, G. E., Estimation of Directional Stability Derivatives at Moderate Angles and Supersonic Speeds, NASA Memo 12-1-58A.
8. Carmichael, R. L., A Critical Evaluation of Methods for Computing Wing-Body Interference at Supersonic Speeds, ICAS Paper No. 68-08, Sixth Congress of the ICAS, Muchen German, September 9-13, 1968.
9. Bradley, R. G., and Miller, B. D., Lifting Surface Theory - Advances and Applications, AIAA Paper No. 70-192, 8th Aerospace Sciences Meeting, New York, January 19-21, 1970.
10. Woodward, F. A., A Unified Approach to the Analysis and Design of Wing-Body Combinations at Subsonic and Supersonic Speeds, AIAA Paper No. 68-55, January 22-24, 1968.
11. Eggers, A. J., Jr., Syvertson, C. A., and Kraus, S., A Study of Inviscid Flow About Airfoils at High Supersonic Speeds, NACA Report 1123, 1953.
12. Eggers, A. J., Jr., and Savin, R. C., A Unified Two-Dimensional Approach to the Calculation of Three-Dimensional Hypersonic Flows, With Application to Bodies of Revolution, NACA Report 1249, 1955.

13. Savin, R. C., Application of the Generalized Shock-Expansion Method to Inclined Bodies of Revolution Traveling at High Supersonic Speeds, NACA TN 3349, April 1955.
14. Syvertson, C. A., and Dennis, D. H., A Second-Order Shock-Expansion Method Applicable to Bodies of Revolution Near Zero Lift, NACA Report 1328, 1957.
15. Hayes, W. D., and Probstein, R. F., Hypersonic Flow Theory, Volume I, Inviscid Flows, Second Edition (1966), Academic Press, New York.
16. Jackson, C. M., and Smith, R. S., A Method for Determining the Total Drag of a Pointed Body of Revolution in Supersonic Flow With Turbulent Boundary Layer, NASA TN D-5046, March 1969.
17. Jernell, L. S., Comparisons of Theoretical and Experimental Pressure Distributions Over a Wing-Body Model at High Supersonic Speeds, NASA TN D-6480, September 1971.
18. Inouye, M., Rakich, J. V., and Lomax, H., A Description of Numerical Methods and Computer Programs for Two-Dimensional and Axisymmetric Supersonic Flow Over Blunt-Nosed and Flared Bodies, NASA TN D-2970, August 1965.
19. Jackson, C. M., and Sawyer, W. C., A Method for Calculating the Aerodynamic Loading on Wing-Body Combinations at Small Angles of Attack in Supersonic Flow, NASA TN D-6441, October 1971.
20. Jones, D. J., Tables of Inviscid Supersonic Flow About Circular Cones at Incidence, $\gamma = 1.4$, AGARDograph 137, November 1969.
21. Kaufman, L. G., II., Pressure Estimation Techniques for Hypersonic Flows Over Blunt Bodies, Journal of Astronautical Sciences, Volume X, No. 2, Summer 1963.
22. Ames Research Staff, Equations, Tables, and Charts for Compressible Flow, NACA TR 1135, 1953.
23. Liepmann, H. W., and Roshko, A., Elements of Gasdynamics, John Wiley and Sons, Inc., 1957.
24. Bertram, M. H. and Henderson, A., Jr., Recent Hypersonic Studies of Wings and Bodies, ARS Journal, Volume 31, No. 8, August 1961.
25. Van Dyke, M. D., Second-Order Slender-Body Theory - Axisymmetric Flow, NASA TR R-47, 1959.
26. Hammitt, A. G., and Murthy, K. R., Approximate Solutions for Supersonic Flow Over Wedges and Cones, JAS, January 1960, pp. 71-73.

27. Schwartz, L. W., Comment on 'An Empirical Expression for Drag Coefficients of Cones at Supersonic Speeds', AIAA Journal, March 1969, pp. 572-573.
28. Collingborne, J. R., Crabtree, L. F., and Bartlett, W. J., A Semi-Empirical Prediction Method for Pressures on the Windward Surface of Circular Cones at Incidence at High Supersonic and Hypersonic Speeds ($M \geq 3$), Aeronautical Research Council, C. P. No. 792, January 1964.
29. Jones, D. J., A Pressure Formula for an Inclined Circular Cone in Supersonic Flow, $\gamma = 1.4$, AIAA Journal, February 1972, pp. 234-236.
30. Van Dyke, M. D., A Study of Hypersonic Small-Disturbance Theory, NACA Report 1194, 1954.
31. Shapiro, A. H., The Dynamics and Thermodynamics of Compressible Fluid Flow, The Ronald Press, 1953.
32. Hayes, W. D., and Probstein, R. F., Hypersonic Flow Theory, Academic Press, 1959.
33. Van Tassell, W., Free-Molecular and Newtonian Coefficients for Arbitrary Bodies, RAD-TM-63-63, August 1963.
34. Hankey, W. L., Jr., Optimization of Lifting Re-Entry Vehicles, ASD-TDR-62-1102, March 1963.
35. This reference has been deleted. For further information contact AFFDL/FXG/A.C. 513/255-2630.
36. Love, E. S., Henderson, A., Jr., and Bertram, M. H., Some Aspects of the Air-Helium Simulation and Hypersonic Approximations, NASA TN D-49, October 1959.
37. This reference has been deleted. For further information contact AFFDL/FXG/A.C. 513/255-2630.
38. Lukasiewicz, J., Hypersonic Flow-Blast Analogy, AEDC-TR-61-4, June 1961.
39. Roberts, M. L., Newtonian Aerodynamic Coefficients for and Arbitrary -Body, Including "All Shadowed" Areas, J. Spacecraft, Volume 7, No. 9, September 1970.
40. McNally, W. D., FORTRAN Program for Calculating Compressible Laminar and Turbulent Boundary Layers in Arbitrary Pressure Gradients, NASA TN D-5681, May 1970.

41. Cohen, C. B., and Reshotko, E., The Compressible Laminar Boundary Layer with Heat Transfer and Arbitrary Pressure Gradient, NACA TR 1294, 1956.
42. Sasman, P.K., and Cresci, R.J., Compressible Turbulent Boundary Layer with Pressure Gradient and Heat Transfer, AIAA J., Volume 4, No. 1, January 1966, pp. 19-25.
43. Schlichting, H., Boundary Layer Theory, Sixth Ed., McGraw-Hill Book Company, Inc., 1968.
44. Schlichting, H., and Ulrich, A., Zur Berechnung des Umschlages Laminar-Turbulent, Jahrbuch d. dt. Luftfahrtforschung, No. 1, 1942, --. 8-35.
45. Granville, F.S., The Calculation of the Viscous Drag of Bodies of Revolution, Report 849, David Taylor Model Basin, July 1953.
46. Monaghan, R.J., An Approximate Solution of the Compressible Laminar Boundary Layer on a Flat Plate, British R&M No 2760, 1953.
47. This reference has been deleted. For further information contact AFFDL/FXG/A.C. 513/255-2630.
48. Lewis, C.H. and Burgess, E.G., III, Empirical Equations for Thermodynamic Properties of Air and Nitrogen to 15,000° K, AEDC-TDR-63-138, July 1963.
49. Kaups, K., and Keltner, G., Laminar Compressible Boundary Layer on a Yawed Infinite Wing, Douglas Report No. LB 32706, March 1967.
50. Hilsenrath, J. and Klein, M., Tables of Thermodynamic Properties of Air in Chemical Equilibrium Including Second Virial Corrections from 1500°K to 15,000°K, AEDC-TR-65-58, March 1965.
51. Hansen, C.F., Approximations for the Thermodynamic and Transport Properties of High Temperature Air, NACA TN-4150, 1959.
52. Michel, R. and Kretzschmar, G., Determination Theorique D'une Enthalpie de Reference Pour la Couche Limite Laminaire de la Plaque Plane, Influence due Nombre de Prandtl, O. N. E. R. A. Report T.P. No. 41, 1963.
53. Eckert, E.R.G., Survey on Heat Transfer at High Speeds, WADC TR 54-70, April 1954.
54. Bromley, L.A., and Wilke, C.R., Viscosity at Low Temperatures, University of California TR HE-150-157.

55. Sieron, T.R., and Martinez, C., Jr., Effects and Analysis of Mach Number and Reynolds Number on Laminar Skin Friction at Hypersonic Speeds, AFFDL-TR-65-5, April 1965.
56. Bertram, M.H., and Blackstock, T.A., Some Simple Solutions to the Problem of Predicting Boundary-Layer Self-Induced Pressures, NASA TN D-798, April 1961.
57. White, F.M., Jr., Hypersonic Laminar Viscous Interactions of Inclined Flat Plates, ARS Journal, May 1962.
58. Bertram, M.H., Hypersonic Laminar Viscous Interaction Effects on the Aerodynamics of Two-Dimensional Wedge and Triangular Planform Wings, NASA TN D-3523, August 1966.
59. Neal, L., Jr., Average Pressure and Skin Friction on Tapered Plates Hypersonic Flow, J. Spacecraft, Volume 3, No. 6, June 1966, pp. 945-946.
60. Goldberg, L., The Structure of the Viscous Hypersonic Shock Layer, General Electric Report R65SD50, December 1965.
61. Goldberg, L., Forces and Moments on the Front Face of a Blunt Lifting Reentry Vehicle, General Electric Report R66SD21, April 1966.
62. Grabau, M., A Method of Forming Continuous Empirical Equations for the Thermodynamic Properties of Air From Ambient Temperatures to 15,000°K, With Applications, AEDC-TN-59-102, August 1959.
63. Griffith, B.J., Comparison of Data from the Gemini Flights and AEDC-VKF Wind Tunnels, AEDC-TR-66-178, October 1966.
64. Korn, G.A., and Korn, T.M., Mathematical Handbook for Scientists and Engineers, McGraw-Hill, April 1961.
65. Sivells, J.C., and Payne, R.G., A Method of Calculating Turbulent-Boundary-Layer Growth at Hypersonic Mach Numbers, AEDC-TR-59-3, March 1959.
66. Spalding, D.B., and Chi, S.W., The Drag of a Compressible Turbulent Boundary Layer on a Smooth Flat Plate With and Without Heat Transfer, J. Fluid Mechanics, Volume 18, Part 1, January 1964, pp. 117-143.