User Guide for Compressible Flow Toolbox
Version 2.1 for Use With MATLAB® Version 7

Kevin J. Melcher
Glenn Research Center, Cleveland, Ohio
Since its founding, NASA has been dedicated to the advancement of aeronautics and space science. The NASA Scientific and Technical Information (STI) Program Office plays a key part in helping NASA maintain this important role.

The NASA STI Program Office is operated by Langley Research Center, the Lead Center for NASA’s scientific and technical information. The NASA STI Program Office provides access to the NASA STI Database, the largest collection of aeronautical and space science STI in the world. The Program Office is also NASA’s institutional mechanism for disseminating the results of its research and development activities. These results are published by NASA in the NASA STI Report Series, which includes the following report types:

- **TECHNICAL PUBLICATION.** Reports of completed research or a major significant phase of research that present the results of NASA programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA’s counterpart of peer-reviewed formal professional papers but has less stringent limitations on manuscript length and extent of graphic presentations.

- **TECHNICAL MEMORANDUM.** Scientific and technical findings that are preliminary or of specialized interest, e.g., quick release reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.

- **CONTRACTOR REPORT.** Scientific and technical findings by NASA-sponsored contractors and grantees.

- **CONFERENCE PUBLICATION.** Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or cosponsored by NASA.

- **SPECIAL PUBLICATION.** Scientific, technical, or historical information from NASA programs, projects, and missions, often concerned with subjects having substantial public interest.

- **TECHNICAL TRANSLATION.** English-language translations of foreign scientific and technical material pertinent to NASA's mission.

Specialized services that complement the STI Program Office’s diverse offerings include creating custom thesauri, building customized databases, organizing and publishing research results . . . even providing videos.

For more information about the NASA STI Program Office, see the following:


- E-mail your question via the Internet to help@sti.nasa.gov

- Fax your question to the NASA Access Help Desk at 301–621–0134

- Telephone the NASA Access Help Desk at 301–621–0390

- Write to: NASA Access Help Desk NASA Center for AeroSpace Information 7121 Standard Drive Hanover, MD 21076
User Guide for Compressible Flow Toolbox
Version 2.1 for Use With MATLAB® Version 7

Kevin J. Melcher
Glenn Research Center, Cleveland, Ohio

National Aeronautics and
Space Administration

Glenn Research Center

January 2006
Acknowledgments

The author acknowledges the significant contribution of Jonathan DeCastro, QSS Group, Inc. Mr. DeCastro conducted comprehensive testing of the algorithms comprising the Compressible Flow Toolbox, and completed the tedious task of reviewing this document in detail prior to publication.

Available from

NASA Center for Aerospace Information
7121 Standard Drive
Hanover, MD 21076

National Technical Information Service
5285 Port Royal Road
Springfield, VA 22100

Trade names or manufacturers’ names are used in this report for identification only. This usage does not constitute an official endorsement, either expressed or implied, by the National Aeronautics and Space Administration.

Available electronically at http://gltrs.grc.nasa.gov
## Contents

Abstract ....................................................................................................................................... 1  
1. Introduction ......................................................................................................................... 1  
2. Nomenclature .................................................................................................................... 3  
3. Quick Reference Tables ...................................................................................................... 5  
4. Function Reference Guide ................................................................................................... 7  
   ames ........................................................................................................................................ 7  
   ameserr ............................................................................................................................... 11  
   amesplt ............................................................................................................................... 17  
   deltamax ................................................................................................................................ 19  
   deltason .................................................................................................................................. 23  
   fanno ..................................................................................................................................... 27  
   fannoerr ............................................................................................................................. 31  
   fannoplt .................................................................................................................................. 35  
   fannotbl .................................................................................................................................. 37  
   isentbl .................................................................................................................................... 39  
   nshktbl .................................................................................................................................... 41  
   oblqshck .............................................................................................................................. 43  
   oblqw12 ............................................................................................................................... 45  
   oblqw21 .................................................................................................................................. 49  
   rayleigh ................................................................................................................................. 53  
   raylerr ..................................................................................................................................... 57  
   raylplt ..................................................................................................................................... 61  
   rayltbl ..................................................................................................................................... 63  
5. References ......................................................................................................................... 65
User Guide for Compressible Flow Toolbox
Version 2.1 for Use With MATLAB® Version 7

Kevin J. Melcher
National Aeronautics and Space Administration
Glenn Research Center
Cleveland, Ohio 44135

Abstract
This report provides a user guide for the Compressible Flow Toolbox, a collection of algorithms that solve almost 300 linear and nonlinear classical compressible flow relations. The algorithms, implemented in the popular MATLAB® programming language, are useful for analysis of one-dimensional steady flow with constant entropy, friction, heat transfer, or shock discontinuities. The solutions do not include any gas dissociative effects. The toolbox also contains functions for comparing and validating the equation-solving algorithms against solutions previously published in the open literature. The classical equations solved by the Compressible Flow Toolbox are:

• The isentropic-flow equations,
• The Fanno flow equations (pertaining to flow of an ideal gas in a pipe with friction),
• The Rayleigh flow equations (pertaining to frictionless flow of an ideal gas, with heat transfer, in a pipe of constant cross section.),
• The normal-shock equations,
• The oblique-shock equations,
• The Prandtl-Meyer expansion equations.

The user should note that the scope of this guide is limited to documenting the individual functions and providing instruction in using them to solve simple compressible flow examples. Functions in the toolbox can be used together to solve more complex compressible flow problems—that is why they were created. However, instructing the user in the broader context of compressible flow is not the intended purpose of this guide.
Background
Algorithms included in the Compressible Flow Toolbox were originally developed to support controls and dynamics research under the NASA’s High Speed Research Program. They were inspired by NACA Report 1135 “Equations Tables, and Charts for Compressible Flow” (ref. 1) which the author studied extensively as part of that research. Early implementations were first published as part of the author’s Masters Thesis in 1996. They were subsequently made publicly available via a MATLAB® third party software web site hosted by the Mathworks, Inc. After several years, the toolbox was removed from the web site for a variety of reasons, including the need to upgrade the algorithms for compatibility with newer versions of MATLAB®. Finally, to appease a number of recent requests for the software, the toolbox has been updated, expanded, and made available to the general public via the NASA Software Repository.

All of the numerical and graphical results shown in this report were generated using functions included in the Compressible Flow Toolbox version 2.1 and running MATLAB® version 7.04 on an MS Windows XP, 2.2 GHz Intel Pentium 4 processor-based personal computer. Results may vary slightly based on the precision of the floating point processor used to perform the calculations.

Organization
This User’s Guide is organized in five sections. Introduction, Nomenclature, Quick Reference Guide, Function Reference Guide, and References. Section 1. Introduction provides a general description of the User Guide along with historical information on the origin of the toolbox and availability of the software. Section 2. Nomenclature describes the symbols and special formatting conventions used throughout the text. Section 3. Quick Reference Guide provides a comprehensive list of the functions contained in the toolbox and provides a brief description of each function listed. Section 4. Function Reference Guide provides a detailed description of each function in the toolbox including its purpose, syntax, a discussion of how the algorithm works, and examples demonstrating its use. Finally, Section 5. References contains a list of references used in developing and documenting the toolbox.

Availability
At the time this report was published, the Compressible Flow Toolbox was available to the general public without cost through the NASA Software Repository.

https://technology.grc.nasa.gov/software/
2. Nomenclature

Formats and Convensions

Monospace
MATLAB® commands, functions names and screen output are
displayed in this font. For example: rayleigh.

Italics
Book titles and names of book sections, mathematical symbols
and notation, and the introduction of new terms. For example:
Introduction.

Bold Initial Caps
Key names, menu names, and items that are selected from
menus. For example: the File menu.

Symbols

This document uses the following symbols and notations:

<table>
<thead>
<tr>
<th>Roman Symbols</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A$</td>
<td>Cross-sectional area of stream tube or channel</td>
</tr>
<tr>
<td>$D_H$</td>
<td>Hydraulic diameter of the flow cross-sectional area</td>
</tr>
<tr>
<td>$I$</td>
<td>Impulse function</td>
</tr>
<tr>
<td>$M$</td>
<td>Mach number, $V/a$</td>
</tr>
<tr>
<td>$P$</td>
<td>Static Pressure</td>
</tr>
<tr>
<td>$P_t$</td>
<td>Total Pressure</td>
</tr>
<tr>
<td>$T$</td>
<td>Static Temperature</td>
</tr>
<tr>
<td>$T_t$</td>
<td>Total Temperature</td>
</tr>
<tr>
<td>$V$</td>
<td>Velocity</td>
</tr>
<tr>
<td>$f$</td>
<td>Average friction factor</td>
</tr>
<tr>
<td>$q$</td>
<td>Dynamic pressure, $\rho V^2/2$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Greek Symbols</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\beta$</td>
<td>$\sqrt{</td>
</tr>
<tr>
<td>$\gamma$</td>
<td>Ratio of specific heats of the working fluid (default = 1.4)</td>
</tr>
<tr>
<td>$\delta$</td>
<td>Turning angle (degrees)</td>
</tr>
<tr>
<td>$\theta$</td>
<td>Oblique shock angle (degrees)</td>
</tr>
<tr>
<td>$\mu$</td>
<td>Mach Angle (degrees)</td>
</tr>
<tr>
<td>$\nu$</td>
<td>Prandtl-Meyer angle (degrees)</td>
</tr>
<tr>
<td>$\rho$</td>
<td>Static mass density</td>
</tr>
<tr>
<td>$\rho$</td>
<td>Mass density</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Subscripts</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>*</td>
<td>Critical flow condition (i.e., conditions where the local fluid</td>
</tr>
<tr>
<td></td>
<td>velocity is equal to the local speed of sound)</td>
</tr>
<tr>
<td>1</td>
<td>Upstream flow property</td>
</tr>
<tr>
<td>2</td>
<td>Downstream flow property</td>
</tr>
</tbody>
</table>
3. Quick Reference Tables

### PROPERTIES OF ISENTROPIC FLOW, PRANDTL-MEYER FLOW, AND NORMAL SHOCKS

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ames</td>
<td>Solves the equations for isentropic flow, Prandtl-Meyer flow, and normal shocks to obtain flow properties.</td>
</tr>
<tr>
<td>amesplt</td>
<td>Plots the properties for isentropic flow, Prandtl-Meyer flow, and the normal shocks as a function of Mach number.</td>
</tr>
<tr>
<td>ameserr</td>
<td>Consistency check for function ames. Computes and plots, as a function of Mach number, errors in ames calculations.</td>
</tr>
<tr>
<td>isentbl</td>
<td>Generates text file containing a table of the isentropic flow properties.</td>
</tr>
<tr>
<td>nshktbl</td>
<td>Generates text file containing a table of Prandtl-Meyer flow and normal shock properties.</td>
</tr>
</tbody>
</table>

### PROPERTIES OF OBLIQUE SHOCKS

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>oblqshck</td>
<td>Solves the oblique shock equations for both weak and strong shock angles.</td>
</tr>
<tr>
<td>oblqw12</td>
<td>Solves the oblique shock equations to obtain downstream flow properties as a function of upstream flow properties.</td>
</tr>
<tr>
<td>oblqw21</td>
<td>Solves the oblique shock equations to obtain upstream flow properties as a function of downstream flow properties.</td>
</tr>
<tr>
<td>deltason</td>
<td>Computes the theoretical deflection angle that reduces supersonic flow to sonic conditions.</td>
</tr>
<tr>
<td>deltamax</td>
<td>Computes the theoretical maximum angle through which supersonic flow may be deflected or turned without separation.</td>
</tr>
</tbody>
</table>

### PROPERTIES OF FANNO-LINE FLOW

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>fanno</td>
<td>Solves the Fanno line equations to obtain properties of flow with friction.</td>
</tr>
<tr>
<td>fannoplt</td>
<td>Plots the Fanno line flow properties as a function of Mach number.</td>
</tr>
<tr>
<td>fannoerr</td>
<td>Consistency check for function fanno. Computes and plots, as a function of Mach number, errors in fanno calculations.</td>
</tr>
<tr>
<td>fannotbl</td>
<td>Generates text file containing a table of the Fanno-line flow properties.</td>
</tr>
<tr>
<td>Command</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>-------------</td>
</tr>
<tr>
<td>rayleigh</td>
<td>Solves the Rayleigh-line equations to obtain properties of flow heating or cooling.</td>
</tr>
<tr>
<td>raylplt</td>
<td>Plots the Rayleigh-line flow properties as a function of Mach number.</td>
</tr>
<tr>
<td>raylerr</td>
<td>Consistency check for function <code>rayleigh</code>. Computes and plots, as a function of Mach number, errors in <code>rayleigh</code> calculations.</td>
</tr>
<tr>
<td>rayltbl</td>
<td>Generates text file containing a table of the Rayleigh-line flow properties.</td>
</tr>
</tbody>
</table>
4. Function Reference Guide

**ames**

**Purpose**
Solve the equations for isentropic flow, both subsonic and supersonic, Prandtl-Meyer expansion, and normal shocks.

**Synopsis**

```matlab
ames
Properties=ames(VarIn,ValuesIn,VarsOut)
Properties=ames(VarIn,ValuesIn,VarsOut,Gamma)
[Properties,PltLbls]=ames(VarIn,ValuesIn,VarsOut,Gamma)
```

**Description**

*ames* by itself calls *amesplt* which plots normalized versions of the isentropic flow, Prandtl-Meyer, and normal shock functions versus Mach number.

Properties=ames(VarIn,ValuesIn,VarsOut), given a number designating one of the flow properties listed in Table 4.1 and a value or vector of values for that flow property, *ames* computes corresponding values for isentropic flow, Prandtl-Meyer flow, and normal shock functions. *VarIn* is a scalar that specifies the property used as the input (independent variable). *ValuesIn* may be a scalar or vector and contains values of the independent variable for which the other properties will be computed. *VarsOut* contains a list of Indices corresponding to the flow properties listed in Table 4.1. Indices specified in *VarsOut* may be in any order and may be repeated as desired by the user. Results are returned in the *Properties* matrix. Columns in this matrix correspond to indices specified in *VarsOut*. Rows of the *Properties* matrix contain results corresponding to the elements of *ValuesIn*.

Note that, when properties 5, 6, or 7 are used as the independent variable, the solution is double-valued. The double-valued solution is provided by making *Properties* a cell array. *Properties{1}* contains values of the solution associated with the smaller Mach number, while *Properties{2}* contains the solution associated with the larger Mach number.

Properties=ames(VarIn,ValuesIn,VarsOut,Gamma) provides a mechanism for specifying values for the ratio of specific heats of the working fluid via *Gamma*. *Gamma* is optional. If unspecified, a value of 1.4, the value of the ratio of specific heats of air at standard temperature and pressure, is used. If specified, *Gamma* may be defined as either a scalar or a vector. If it is a vector, it must have the same length as *ValuesIn*.

[Properties,PltLbls]=ames(VarIn,ValuesIn,VarsOut,Gamma), in addition to returning the properties of the fluid at user specified conditions, also returns a cell array, *PltLbls* containing text strings that may be used when plotting the results.
### Table 4.1—Description of Flow Properties Computed by Function ames

<table>
<thead>
<tr>
<th>REF. INDEX</th>
<th>PROPERTY</th>
<th>REF. 1</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISENTROPIC FLOW PROPERTIES (VALID FOR ALL M):</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.</td>
<td>$M$ or $M_1$</td>
<td></td>
<td>Mach number</td>
</tr>
<tr>
<td>2.</td>
<td>$P/P$</td>
<td>Eq. 44</td>
<td>Ratio of static to total pressure</td>
</tr>
<tr>
<td>3.</td>
<td>$\rho/\rho_s$</td>
<td>Eq. 45</td>
<td>Ratio of static to total density</td>
</tr>
<tr>
<td>4.</td>
<td>$T/T_s$</td>
<td>Eq. 43</td>
<td>Ratio of static to total temperature</td>
</tr>
<tr>
<td>5.</td>
<td>$\beta$</td>
<td>pg. 1</td>
<td>$\sqrt{\frac{M^2 - 1}{M^2}}$</td>
</tr>
<tr>
<td>6.</td>
<td>$q/P$</td>
<td>Eq. 48</td>
<td>Ratio of dynamic to total pressure</td>
</tr>
<tr>
<td>7.</td>
<td>$A/A'$</td>
<td>Eq. 80</td>
<td>Ratio of flow area to critical flow area</td>
</tr>
<tr>
<td>8.</td>
<td>$V/V'$</td>
<td>Eq. 50</td>
<td>Ratio of flow velocity to critical flow velocity</td>
</tr>
<tr>
<td>PRANDTL-MEYER FLOW (VALID FOR $M \geq 1$):</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9.</td>
<td>$\nu$</td>
<td>Eq. 171</td>
<td>Prandtl-Meyer angle (degrees)</td>
</tr>
<tr>
<td>10.</td>
<td>$\mu$</td>
<td>pg. 1</td>
<td>Mach Angle (degrees), $\sin^{-1}(1/M)$</td>
</tr>
<tr>
<td>NORMAL SHOCK PROPERTIES (VALID FOR $M \geq 1$):</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11.</td>
<td>$M_2$</td>
<td>Eq. 96</td>
<td>Mach number downstream of a normal shock</td>
</tr>
<tr>
<td>12.</td>
<td>$P_2/P_1$</td>
<td>Eq. 93</td>
<td>Static pressure ratio across a normal shock</td>
</tr>
<tr>
<td>13.</td>
<td>$\rho_2/\rho_1$</td>
<td>Eq. 94</td>
<td>Static density ratio across a normal shock</td>
</tr>
<tr>
<td>14.</td>
<td>$T_2/T_1$</td>
<td>Eq. 95</td>
<td>Static temperature ratio across a normal shock</td>
</tr>
<tr>
<td>15.</td>
<td>$P_{s,2}/P_{s,1}$</td>
<td>Eq. 99</td>
<td>Total pressure ratio across a normal shock</td>
</tr>
<tr>
<td>16.</td>
<td>$P_{s,2}/P_{s,1}$</td>
<td>Eq. 100</td>
<td>Ratio of static pressure upstream of a normal shock to total pressure downstream of the same shock</td>
</tr>
</tbody>
</table>

**Algorithm**

`ames` determines the desired flow properties by first obtaining a Mach number solution for each value, `ValuesIn`, of the user specified flow property, `VarIn`. These Mach numbers are then used to compute the other properties, `VarsOut`, specified by the user. Most of the flow equations may be manipulated analytically to obtain Mach number as a function of the other properties. However, some nonlinear relationships exist which have no simple analytical solution. In these cases, MATLAB’s `fminbnd` function is used to determine an approximate solution for Mach number from the nonlinear equations. The search is arbitrarily constrained to Mach numbers less than 100. Solutions associated with Mach numbers larger than 100 are returned as `NaN` (i.e., not a number).

**See Also**

`ameserr`, `amesplt`, `isentbl`, and `nshcktbl`

**Example 4.1:**

Determine the properties of air at Mach 2.
Example 4.2:
Given a normal shock with downstream Mach number of 0.85, determine the Mach number upstream of the shock.

```
>> ames(11,0.85,1)
ans =
1.1876
```

Example 4.3:
Determine the properties of air when $\frac{A}{A_0} = 3.007$.

```
>> properties=ames(7,3.007,1:16)
properties =
[1x16 double]    [1x16 double]
>> properties{1}
ans =
Columns 1 through 5
0.1970    0.9733    0.9809    0.9923    0.9804
Columns 6 through 10
0.0264    3.0070    0.2149       NaN       NaN
Columns 11 through 15
NaN       NaN       NaN       NaN       NaN
Column 16
NaN

>> properties{2}
ans =
Columns 1 through 6
2.6399    0.0471    0.1128    0.4177    2.4432
Columns 6 through 10
0.2299    3.0070    1.8691   42.3049   22.2597
Columns 11 through 15
0.5005    7.9638    3.4935    2.2796    0.4453
Column 16
0.1058
Example 4.4:

Plot the Mach number downstream of a normal shock as a function of the Mach number upstream of the shock.

```matlab
M1=1:0.1:10;
[M2,Lbls]=ames(1,M1,11);
plot(M1,M2);
xlabel('M_1'); ylabel(Lbls{1});
```

Figure 4.1.—Result of using function ames to compute Mach number variations across a normal shock.
ameserr

Purpose
Show the computational errors that result when using function ames to solve the equations for isentropic flow, Prandtl-Meyer expansion, and normal shocks.

Synopsis
ameserr
[error,M1]=ameserr

Description
ameserr computes the error between Mach numbers used as inputs to function ames and Mach numbers calculated from the output of function ames. The results are plotted as absolute and percent errors versus Mach number for each of the flow functions shown in Table 4.1.

[error,M1]=ameserr returns the computed error in error. If specified, M1 contains the initial vector of Mach numbers.

Algorithm
ameserr first generates a logarithmically spaced vector of 250 Mach numbers from 0.01 to 10. This vector also includes critical Mach number values where numerical stability is important, such as saddle points. ameserr then uses function ames to calculate each of the isentropic flow properties and the normal shock properties corresponding to those Mach numbers. The functions of Mach number, obtained from ames, are then used as input to the ames function in order to obtain a Mach number which corresponds to the function value. Theoretically, the initial and computed Mach numbers should be the same. In general, they are not due to round off, truncation, convergence, and/or optimization errors. The difference in the two Mach numbers is returned as the error in the calculations.

See Also
ames, amesplt, isentbl, and nshcktbl

Example 4.5:
Compute and plot the errors the errors that result from running ameserr. Plots are shown in Figure 4.2(a to g).

>> ameserr
Figure 4.2.—Output of function ameserr as computed on an Intel Pentium4 processor-based computer running MATLAB® 7.
Figure 4.2.—Output of function ameserr as computed on an Intel Pentium4 processor-based computer running MATLAB® 7 (continued).
Figure 4.2.—Output of function ameserr as computed on an Intel Pentium4 processor-based computer running MATLAB® 7 (continued).
Figure 4.2.—Output of function `ameserr` as computed on an Intel Pentium4 processor-based computer running MATLAB® 7 (continued).
amesplt

Purpose
Plots normalized properties for isentropic flow, Prandtl-Meyer expansion, and normal shocks as a function of Mach number.

Synopsis
amesplt
amesplt(MNmin,MNmax)
amesplt(MNmin,MNmax,Npts)
amesplt(MNmin,MNmax,Npts,Gamma)

Description
amesplt uses function ames to compute and plot the isentropic and normal shock flow properties at 250 points between Mach 0.01 and Mach 10 when the ratio of specific heats of the fluid is 1.4.

amesplt(MNmin,MNmax) plots results for a range of user specified Mach numbers where: MNmin is the minimum Mach number; and MNmax is the maximum Mach number.

amesplt(MNmin,MNmax,Npts) in addition to allowing the user to specify the range of Mach numbers used, this form allows the user to specify the number of data points, Npts, used to plot each curve.

amesplt(MNmin,MNmax,Npts,Gamma) in addition to allowing the user to specify Mach number and number of points per curve, this form also allows the user to specify a scalar value for the ratio of specific heats, Gamma, of the fluid.

Algorithm
amesplt first generates a logarithmically spaced vector of 250 Mach numbers from 0.01 to 10. This vector also includes critical Mach number values where numerical stability is important, such as solution saddle points. amesplt then uses this vector as inputs to function ames which is used to calculate each of the isentropic flow properties and the normal shock properties corresponding to those Mach numbers. The resulting values are normalized and plotted versus Mach number to provide the user a graphical understanding of the relationship between flow properties and Mach number.

See Also
ames, amesplt, isentbl, and nshcktbl

Example 4.6:
Plot normalized isentropic flow and normal shock properties as a function of Mach number. The resulting plots are shown in Figure 4.3 (a and b).

>> amesplt
Figure 4.3.—Normalized isentropic and normal shock functions as generated by function amesplt.
**deltamax**

**Purpose**

For steady state supersonic flow with compressive turning, **deltamax** computes the maximum flow deflection angle ($\delta$) that can occur without producing separation of the flow from the turning surface. Also, optionally calculates the angle of the oblique shock ($\theta$) that results from turning the flow. Both angles have units of degrees. See Figure 4.4 for a graphical representation of the flow situation.

**Synopsis**

deltamax

\[ \text{Delta} = \text{deltamax}(\text{M1}) \]

\[ [\Delta, \Theta] = \text{deltamax}(\text{M1}, \Gamma) \]

**Description**

**deltamax** by itself, computes and plots the maximum flow deflection and resulting oblique shock angle for a range of Mach numbers from 1.0 to 15.

\[ \Delta = \text{deltamax}(\text{M1}) \]

computes and returns the maximum flow deflection angle, \( \delta \), in degrees for user specified Mach numbers, \( \text{M1} \). \( \text{M1} \) may be a scalar, vector, or matrix.

\[ [\Delta, \Theta] = \text{deltamax}(\text{M1}, \Gamma) \]

uses optional input \( \Gamma \), the ratio of specific heats for the working fluid, to calculate the turning angle \( \Delta \) and additionally the angle, \( \Theta \), of the oblique shock that results from turning the flow. \( \Gamma \) has a default value of 1.4 and must be a scalar or have dimensions equivalent to \( \text{M1} \). The dimensions of \( \Delta \) and \( \Theta \), and the values therein, correspond to the dimensions of \( \text{M1} \).

**Algorithm**

If no input parameters are specified by the user, **deltamax** first generates a vector of upstream Mach numbers. The function then uses the Mach number(s) to calculate the maximum angle, \( \theta_{\max} \), of an oblique shock that can occur without separation. The shock angle is then used with the Mach number(s) to calculate the associated flow deflection angle, \( \delta_{\max} \).

![Oblique shock diagram](image)

**Figure 4.4.—Oblique shock diagram.**
The equation used to calculate $\theta_{\text{max}}$ is:

$$\theta_{\text{max}} = \sin^{-1} \left\{ \frac{1}{\gamma M_1^2} \left[ \frac{(\gamma + 1)}{4} M_1^2 + \sqrt{\left(\gamma + 1\right) \left[ 1 + \frac{(\gamma - 1)}{2} M_1^2 + \frac{(\gamma - 1)}{16} M_1^4 \right] - 1} \right] \right\}$$  \hspace{1cm} (4.1)

The equation used to calculate $\delta_{\text{max}}$ is:

$$\delta_{\text{max}} = \tan^{-1} \left( \frac{M_1^2 \sin^2 \theta_{\text{max}} - 1 \cot \theta_{\text{max}}}{\frac{1}{2} (\gamma + 1) M_1^2 - M_1^2 \sin^2 \theta_{\text{max}} + 1} \right)$$  \hspace{1cm} (4.2)

Similar equations may be found in reference 1, pp. 9 and 12; (ref. 2), p. 586; and (ref. 4), pp. 315 to 316.

**See Also**

deltason, oblqshck, oblqw12, and oblqw21

**Example 4.7:**

Calculate and plot the maximum compressive turning angle and oblique shock angle for airflow over a range of Mach numbers from 1 to 15.

`>> deltamax`

![Graph showing Maximum Deflection Angle vs. Mach Number](image1)

![Graph showing Shock Angle vs. Mach Number](image2)

**Figure 4.5.**—Results of function `deltamax` showing maximum turning angle and the angle of the resulting oblique shock as a function of upstream Mach number.
**Example 4.8:**

Calculate the maximum compressive turning angle and oblique shock angle for steam flowing at Mach numbers from 1.5 to 3.0. The ratio of specific heats for steam is 1.327 at standard temperature.

```
>> [Delta,Theta]=deltamax(1.5:0.1:3.0,1.327)
Delta =
    Columns 1 through 5
    12.6726   15.3598   17.8660   20.1780   22.2960
    Columns 6 through 10
    24.2282   25.9869   27.5862   29.0404   30.3634
    Columns 11 through 15
    31.5684   32.6673   33.6712   34.5896   35.4314
    Column 16
    36.2042

Theta =
    Columns 1 through 5
    66.7820   66.0774   65.6264   65.3536   65.2072
    Columns 6 through 10
    65.1509   65.1587   65.2116   65.2959   65.4013
    Columns 11 through 15
    65.5206   65.6483   65.7803   65.9137   66.0466
    Column 16
    66.1772
```


**Purpose**

For steady state supersonic flow with compressive turning, \texttt{deltamax} computes the flow deflection angle ($\delta$) that results in sonic flow downstream of the resulting oblique shock (i.e., $M_2 = 1$). Also, optionally calculates the angle of the oblique shock ($\theta$) that produces sonic flow. Both angles have units of degrees. See Figure 4.4 (pp. 19) for a graphical representation of the flow situation.

**Synopsis**

\texttt{deltason}

$\Delta = \texttt{deltason}(M_1)$

$[\Delta, \Theta] = \texttt{deltason}(M_1, \text{Gamma})$

**Description**

\texttt{deltason} by itself, computes and plots the sonic flow deflection angle and the resulting oblique shock angle for a range of Mach numbers from 1.0 to 15.

$\Delta = \texttt{deltason}(M_1)$ computes and returns the flow deflection angle, \( \Delta \), that results in sonic flow downstream. Values of \( \Delta \) are in degrees for user specified Mach numbers, \( M_1 \). \( M_1 \) may be a scalar, vector, or matrix.

$[\Delta, \Theta] = \texttt{deltason}(M_1, \text{Gamma})$ uses optional input \( \text{Gamma} \), the ratio of specific heats for the working fluid, to calculate the turning angle \( \Delta \) and additionally the angle, \( \Theta \), of the oblique shock that results from turning the flow. \( \text{Gamma} \) has a default value of 1.4 and must be a scalar or have dimensions equivalent to \( M_1 \). The dimensions of \( \Delta \) and \( \Theta \), and the values therein, correspond to the dimensions of \( M_1 \).

**Algorithm**

If no input parameters are specified by the user, \texttt{deltason} first generates a vector of upstream Mach numbers. The function then uses the Mach number(s) to calculate the angle, \( \theta_* \), of an oblique shock that produces sonic flow downstream of the shock. The shock angle is then used with the Mach number(s) to calculate the associated flow deflection angle, \( \delta_* \).

The equation used to calculate \( \theta_* \) is:

\[
\theta_* = \sin^{-1} \left( \frac{1}{\gamma M_1^2} \left( \frac{(\gamma+1)M_1^2 - (3 - \gamma)}{4} + \sqrt{\left( \frac{\gamma+1}{16} \left( 9 + \frac{\gamma}{8} - \frac{3 - \gamma}{16} M_1^2 + \frac{\gamma+1}{16} M_1^4 \right) \right)} \right) \right) \quad (4.1)
\]

The equation used to calculate \( \delta_* \) is:
\[ \delta_s = \tan^{-1} \left( \frac{M^2 \sin^2 \theta_s - 1}{\frac{1}{2} (\gamma + 1) M^2_1 - M^2_1 \sin^2 \theta_s + 1} \right) \] 

(4.2)

Similar equations may be found in reference 1, pp. 9 and 12; (ref. 2), p. 586; and (ref. 4), pp. 315 to 316.

See Also

deltamax, oblqshck, oblqw12, and oblqw21

Example 4.9:

For airflow over a range of Mach numbers from 1 to 15, calculate and plot the compressive turning angle and associated oblique shock angle that results in sonic flow downstream of the shock.

\[
\text{>> } \text{deltason}
\]

![Deflection Angle for Sonic Flow vs. Mach Number](image1)

![Shock Angle (deg) vs. Mach Number](image2)

Figure 4.6.—Results of function `deltason` showing sonic turning angle and the angle of the resulting oblique shock as a function of upstream Mach number.

Example 4.10:

Given a flow of hydrogen gas at a several Mach numbers from 1.0 to 2.0, at each Mach number, calculate the compressive turning angle that produces sonic flow and the associated oblique shock angle. The ratio of specific heats for hydrogen is 1.667 at standard temperature.

\[
\text{>> format short e }
\]

\[
\text{>> [Delta,Theta]=deltason(1.0:0.1:2.0,1.667)}
\]
Delta =
  Columns 1 through 4
    -1.4216e-022  1.2526e+000  3.2647e+000  5.5235e+000
  Columns 5 through 8
    7.8286e+000  1.0071e+001  1.2192e+001  1.4161e+001
  Columns 9 through 11
    1.5967e+001  1.7612e+001  1.9103e+001

Theta =
  Columns 1 through 4
    9.0000e+001  7.3209e+001  6.7949e+001  6.4868e+001
  Columns 5 through 8
    6.2929e+001  6.1691e+001  6.0910e+001  6.0434e+001
  Columns 9 through 11
    6.0164e+001  6.0034e+001  5.9998e+001
fanno

Purpose
Solve the equations for one-dimensional steady adiabatic flow in a constant area duct with friction.

Synopsis
Properties=fanno(VarIn,ValuesIn,VarsOut)
Properties=fanno(VarIn,ValuesIn,VarsOut,Gamma)
[Properties,PltLbls]=fanno(VarIn,ValuesIn,VarsOut,Gamma)
fanno

Description
Properties=fanno(VarIn,ValuesIn,VarsOut), given a number designating one of the flow properties listed in Table 4.2 and a value or vector of values for that flow property, fanno computes corresponding values for adiabatic frictional flow. VarIn is a scalar that specifies the property used as the input (independent variable). ValuesIn may be a scalar or vector and contains values of the independent variable for which the other properties will be computed. VarsOut contains a list of Indices corresponding to the flow properties listed in Table 4.2. Indices specified in VarsOut may be in any order and may be repeated as desired by the user. Results are returned in the Properties matrix. Columns in this matrix correspond to indices specified in VarsOut. Rows of the Properties matrix contain results corresponding to the elements of ValuesIn.

Note that, when properties 4, 6, or 7 are used as the independent variable, the solution is double-valued. The double-valued solution is provided by making Properties a cell array. Properties{1} contains values of the solution associated with the smaller Mach number, while Properties{2} contains the solution associated with the larger Mach number.

Properties=fanno(VarIn,ValuesIn,VarsOut,Gamma) provides a mechanism for specifying values for the ratio of specific heats of the working fluid via Gamma. Gamma is optional. If unspecified, a value of 1.4, the value of the ratio of specific heats of air at standard temperature and pressure, is used. If specified, Gamma may be defined as either a scalar or a vector. If it is a vector, it must have the same length as ValuesIn.

[Properties,PltLbls]=fanno(VarIn,ValuesIn,VarsOut,Gamma), in addition to returning the properties of the fluid at user specified conditions, also returns a cell array, PltLbls, containing text strings that may be used when plotting the results.

fanno by itself calls fannoplt which plots the Fanno-line flow properties versus Mach number.
Table 4.2.—Description of Flow Properties Computed by Function fanno

<table>
<thead>
<tr>
<th>REF.</th>
<th>PROPERTY</th>
<th>REF. 4</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>INDEX</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.</td>
<td>$M$ or $M_i$</td>
<td></td>
<td>Mach number</td>
</tr>
<tr>
<td>2.</td>
<td>$T/T_s$</td>
<td>Eq. 5.31</td>
<td>Ratio of static temperature at $M_i$ to static temperature at sonic conditions.</td>
</tr>
<tr>
<td>3.</td>
<td>$P/P_s$</td>
<td>Eq. 5.30</td>
<td>Ratio of static pressure at $M_i$ to static pressure at sonic conditions.</td>
</tr>
<tr>
<td>4.</td>
<td>$P_i/P_{i,s}$</td>
<td>Eq. 5.34</td>
<td>Ratio of total pressure at $M_i$ to total pressure at sonic conditions.</td>
</tr>
<tr>
<td>5.</td>
<td>$V/V_s$</td>
<td>Eq. 5.29</td>
<td>Ratio of flow velocity at $M_i$ to flow velocity at sonic conditions. Also, ratio of static density at sonic conditions to static density at $M_i$.</td>
</tr>
<tr>
<td>6.</td>
<td>$I/I_s$</td>
<td>Eq. 3.42</td>
<td>Ratio of the impulse function at $M_i$ to the impulse function at sonic conditions.</td>
</tr>
<tr>
<td>7.</td>
<td>$4\sqrt{\rho_s D_H}$</td>
<td>Eq. 5.35</td>
<td>Friction factor</td>
</tr>
</tbody>
</table>

Algorithm

fanno determines the desired flow properties by first obtaining a Mach number solution for each value, ValuesIn, of the user specified flow property, VarIn. The resulting Mach numbers are then used to compute the other properties, VarsOut. Some of the flow equations may be manipulated analytically to obtain Mach number as a function of the other properties. However, some nonlinear relationships exist which have no simple analytical solution. In these cases, MATLAB’s fminbnd function is used determine an approximate solution for Mach number from the nonlinear equations. The search is arbitrarily constrained to Mach numbers less than or equal to 100. Solutions associated with Mach numbers larger than 100 are returned as NaN (i.e., not a number).

See Also
	fannoerr, fannoplt, and fannotbl

Example 4.11:
For air flowing at Mach 3.5, determine the Fanno-line flow properties.

```matlab
>> fanno(1,3.5,1:7)
an =
 Columns 1 through 5
   3.5000   0.3478   0.1685   6.7896   2.0642
 Columns 6 through 7
   1.2743   0.5864
```

Example 4.12:
For a range of friction factors from 0.5 to 1.0, determine the static pressure ratio ($P/P_s$) and upstream Mach number of air flowing adiabatically through a constant area duct.

```matlab
>> fric=[0.5:0.1:1.0]'; properties=fanno(7,fric,[3,1])
```
properties =
    [6x2 double]    [6x2 double]

>> [fric properties{1}]
ans =
    0.5000    1.7706    0.5977
    0.6000    1.8459    0.5748
    0.7000    1.9154    0.5551
    0.8000    1.9804    0.5378
    0.9000    2.0416    0.5225
    1.0000    2.0996    0.5087

>> [fric properties{2}]
ans =
    0.5000    0.2359    2.8603
    0.6000    0.1583    3.6302
    0.7000    0.0850    5.1405
    0.8000    0.0148   12.7693
    0.9000       NaN       NaN
    1.0000       NaN       NaN

Here, properties{1} is the subsonic solution, and properties{2} is the supersonic solution. Also, column 1 of ans contains values for friction factor. Column 2 and 3 contain corresponding solutions for pressure ratio and Mach number, respectively. Note that NaN results imply solution has exceeded internal limit for intermediate Mach number calculation (i.e., M > 100).
**fannoerr**

**Purpose**
Show the computational errors that result when using function *fanno* to solve equations for one-dimensional steady adiabatic flow in a constant-area duct with friction.

**Synopsis**

```
fannoerr
[error,M1]=fannoerr
```

**Description**

*fannoerr* computes the error between Mach numbers used as inputs to function *fanno* and Mach numbers calculated from the output of function *fanno*. The results are plotted as absolute and percent errors versus Mach number for each of the flow functions shown in Table 4.2.

```
[error,M1]=fannoerr returns the computed error in error. If specified, M1 contains the initial vector of Mach numbers.
```

**Algorithm**

*fannoerr* first generates a logarithmically spaced vector of 250 Mach numbers from 0.01 to 10. This vector also includes critical Mach number values where numerical stability is important, such as saddle points. *fannoerr* then uses function *fanno* to calculate each of the Fanno-line flow properties corresponding to those Mach numbers. The functions of Mach number, obtained from *fanno*, are then used as input to the *fanno* function in order to obtain a Mach number which corresponds to the function value. Theoretically, the initial and computed Mach numbers should be the same. In general, they are not due to round off, truncation, convergence, and/or optimization errors. The difference in the two Mach numbers is returned as the error in the calculations.

**See Also**

*fanno*, *fannoplt*, and *fannotbl*

**Example 4.13:**
Compute and plot the errors the errors that result from running *fannoerr*. Plots are shown in Figure 4.7(a to d)

```
>> fannoerr
```
Figure 4.7.—Output of function fannoerr as computed on an Intel Pentium4 processor-based computer running MATLAB® 7.
Figure 4.7.—Output of function `fannoerr` as computed on an Intel Pentium4 processor-based computer running MATLAB® 7 (continued).
fannoplt

Purpose
Plot properties for Fanno-line flow, i.e., one-dimensional steady adiabatic flow in a constant-area duct with friction.

Synopsis
fannoplt
fannoplt(MNmin,MNmax)
fannoplt(MNmin,MNmax,Npts)
fannoplt(MNmin,MNmax,Npts,Gamma)

Description
fannoplt uses function fanno to compute and plot the Fanno-line flow properties at 250 points between Mach 0.05 and Mach 2.5 when the ratio of specific heats of the fluid is 1.4. This plot resembles Figure 5.4 in (ref. 4).

fannoplt(MNmin,MNmax) plots results for a range of user specified Mach numbers where: MNmin is the minimum Mach number; and MNmax is the maximum Mach number.

fannoplt(MNmin,MNmax,Npts) in addition to allowing the user to specify the range of Mach numbers used, this form allows the user to specify the number of data points, Npts, used to plot each curve.

fannoplt(MNmin,MNmax,Npts,Gamma) in addition to allowing the user to specify Mach No. and number of points per curve, this form also allows the user to specify a scalar value for the ratio of specific heats, Gamma, of the fluid.

Algorithm
fannoplt first generates a logarithmically spaced vector of 250 Mach numbers from 0.05 to 2.5. This vector also includes critical Mach number values where numerical stability is important, such as solution saddle points. fannoplt then uses this vector as inputs to function fanno which is used to calculate each of the isentropic flow properties and the normal shock properties corresponding to those Mach numbers. The resulting values are plotted versus Mach number to provide the user a graphical understanding of the relationship between flow properties and Mach number.

See Also
fanno, fannoerr, and fannotbl

Example 4.14:
Plot Fanno-line flow properties over a range of Mach numbers from 0.05 to 2.5. The resulting plot is shown in Figure 4.8.

>> fannoplt
Figure 4.8.—Fanno-line flow properties as generated by function fannopl.t.
fannotbl

Purpose
Generate a text file containing tables of Fanno-line flow properties. The tables generated by this function may be useful when computational solution of the equations is not practical.

Synopsis
fannotbl
fannotbl(Filename,Mn,Gamma)

Description
fannotbl uses function fanno to generate a table of values for Fanno-line flow properties as a function of Mach numbers from 0.01 to 10. Properties 2 through 7 of Table 4.2 are written to the text file, fannotbl.txt.

fannotbl(Filename,Mn,Gamma) computes the flow functions and writes the ASCII data to the file specified by the string variable, Filename. Functions are evaluated at Mach numbers specified in Mn. Gamma is an optional scalar variable specifying the ratio of specific heats of the working fluid. If unspecified, a value of 1.4 is used for Gamma.

See Also
fanno, fannoplt, and fannotbl

Example 4–15:
Create a table containing values for Fanno-line flow functions over a range of Mach numbers from 0.50 to 0.70 in increments of 0.01. Results are shown in Table 4.3 on the following page.

>> fannotbl('fannotbl.txt.',0.5:0.01:0.7)
Table 4.3.—Output of function fannotbl for a range of Mach numbers from 0.5 to 0.7

<table>
<thead>
<tr>
<th>M</th>
<th>T/T*</th>
<th>P/P*</th>
<th>P0/P0*</th>
<th>V/V*</th>
<th>I/I*</th>
<th>4fL*/D</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.00000e-001</td>
<td>1.14286e+000</td>
<td>2.13809e+000</td>
<td>1.33984e+000</td>
<td>5.34522e-001</td>
<td>1.20268e+000</td>
<td>1.06906e+000</td>
</tr>
<tr>
<td>5.10000e-001</td>
<td>1.14066e+000</td>
<td>2.09415e+000</td>
<td>1.32117e+000</td>
<td>5.44689e-001</td>
<td>1.19030e+000</td>
<td>9.90414e-001</td>
</tr>
<tr>
<td>5.20000e-001</td>
<td>1.13843e+000</td>
<td>2.05187e+000</td>
<td>1.30339e+000</td>
<td>5.54826e-001</td>
<td>1.17860e+000</td>
<td>9.17418e-001</td>
</tr>
<tr>
<td>5.30000e-001</td>
<td>1.13617e+000</td>
<td>2.01116e+000</td>
<td>1.28645e+000</td>
<td>5.64934e-001</td>
<td>1.16753e+000</td>
<td>8.49624e-001</td>
</tr>
<tr>
<td>5.40000e-001</td>
<td>1.13387e+000</td>
<td>1.97192e+000</td>
<td>1.27032e+000</td>
<td>5.75011e-001</td>
<td>1.15705e+000</td>
<td>7.86625e-001</td>
</tr>
<tr>
<td>5.50000e-001</td>
<td>1.13154e+000</td>
<td>1.93407e+000</td>
<td>1.25495e+000</td>
<td>5.85057e-001</td>
<td>1.14715e+000</td>
<td>7.28053e-001</td>
</tr>
<tr>
<td>5.60000e-001</td>
<td>1.12918e+000</td>
<td>1.89755e+000</td>
<td>1.24029e+000</td>
<td>5.95072e-001</td>
<td>1.13777e+000</td>
<td>6.73571e-001</td>
</tr>
<tr>
<td>5.70000e-001</td>
<td>1.12678e+000</td>
<td>1.86228e+000</td>
<td>1.22633e+000</td>
<td>6.05055e-001</td>
<td>1.12890e+000</td>
<td>6.22874e-001</td>
</tr>
<tr>
<td>5.80000e-001</td>
<td>1.12435e+000</td>
<td>1.82820e+000</td>
<td>1.21301e+000</td>
<td>6.15065e-001</td>
<td>1.12050e+000</td>
<td>5.75683e-001</td>
</tr>
<tr>
<td>5.90000e-001</td>
<td>1.12189e+000</td>
<td>1.79525e+000</td>
<td>1.20031e+000</td>
<td>6.24925e-001</td>
<td>1.11256e+000</td>
<td>5.31743e-001</td>
</tr>
<tr>
<td>6.00000e-001</td>
<td>1.11940e+000</td>
<td>1.76336e+000</td>
<td>1.18820e+000</td>
<td>6.34811e-001</td>
<td>1.10504e+000</td>
<td>4.90822e-001</td>
</tr>
<tr>
<td>6.10000e-001</td>
<td>1.11688e+000</td>
<td>1.73250e+000</td>
<td>1.17665e+000</td>
<td>6.44664e-001</td>
<td>1.09793e+000</td>
<td>4.52705e-001</td>
</tr>
<tr>
<td>6.20000e-001</td>
<td>1.11433e+000</td>
<td>1.70261e+000</td>
<td>1.16565e+000</td>
<td>6.54483e-001</td>
<td>1.09120e+000</td>
<td>4.17197e-001</td>
</tr>
<tr>
<td>6.30000e-001</td>
<td>1.11175e+000</td>
<td>1.67364e+000</td>
<td>1.15515e+000</td>
<td>6.64269e-001</td>
<td>1.08484e+000</td>
<td>3.84116e-001</td>
</tr>
<tr>
<td>6.40000e-001</td>
<td>1.10914e+000</td>
<td>1.64556e+000</td>
<td>1.14515e+000</td>
<td>6.74020e-001</td>
<td>1.07883e+000</td>
<td>3.53299e-001</td>
</tr>
<tr>
<td>6.50000e-001</td>
<td>1.10650e+000</td>
<td>1.61831e+000</td>
<td>1.13562e+000</td>
<td>6.83737e-001</td>
<td>1.07314e+000</td>
<td>3.24591e-001</td>
</tr>
<tr>
<td>6.60000e-001</td>
<td>1.10383e+000</td>
<td>1.59187e+000</td>
<td>1.12654e+000</td>
<td>6.93419e-001</td>
<td>1.06777e+000</td>
<td>2.97853e-001</td>
</tr>
<tr>
<td>6.70000e-001</td>
<td>1.10114e+000</td>
<td>1.56620e+000</td>
<td>1.11798e+000</td>
<td>7.03066e-001</td>
<td>1.06270e+000</td>
<td>2.72955e-001</td>
</tr>
<tr>
<td>6.80000e-001</td>
<td>1.09842e+000</td>
<td>1.54126e+000</td>
<td>1.10965e+000</td>
<td>7.12677e-001</td>
<td>1.05792e+000</td>
<td>2.49775e-001</td>
</tr>
<tr>
<td>6.90000e-001</td>
<td>1.09567e+000</td>
<td>1.51702e+000</td>
<td>1.10182e+000</td>
<td>7.22252e-001</td>
<td>1.05340e+000</td>
<td>2.28204e-001</td>
</tr>
<tr>
<td>7.00000e-001</td>
<td>1.09290e+000</td>
<td>1.49345e+000</td>
<td>1.09437e+000</td>
<td>7.31792e-001</td>
<td>1.04915e+000</td>
<td>2.08139e-001</td>
</tr>
</tbody>
</table>
**isentbl**

**Purpose**
Generate a text file containing tables of isentropic flow properties. The tables generated by this function may be useful when computational solution of the equations is not practical.

**Synopsis**

```
isentbl
isentbl(Filename,Mn,Gamma)
```

**Description**

`isentbl` uses function `ames` to generate a table of values for isentropic flow properties as a function of Mach numbers from 0.01 to 10. Properties 2 through 8 of **Table 4.1** are written to the text file, `isentbl.txt`.

`isentbl(Filename,Mn,Gamma)` computes the flow functions and writes the ASCII data to the file specified by the string variable, `Filename`. Functions are evaluated at Mach numbers specified in `Mn`. `Gamma` is an optional scalar variable specifying the ratio of specific heats of the working fluid. If unspecified, a value of 1.4 is used for `Gamma`.

**See Also**

`ames`, `fannotbl`, `nshcktbl`, and `rayltbl`

**Example 4–16:**
Create a table containing isentropic functions for a range of Mach numbers from 0.9 to 1.1 in increments of 0.01. Results are shown in **Table 4.4** on the following page.

```
>> isentbl('isentbl.txt.', 0.9:0.01:1.1);
```
Table 4.4.—Output of function `isentbl` for a range of Mach numbers from 0.9 to 1.1

<table>
<thead>
<tr>
<th>M or M1</th>
<th>P/Pt</th>
<th>p/pt</th>
<th>T/Tt</th>
<th>Beta</th>
<th>q/Pt</th>
<th>A/A*</th>
<th>V/V*</th>
</tr>
</thead>
<tbody>
<tr>
<td>9.30000e-001</td>
<td>5.72114e-001</td>
<td>6.71079e-001</td>
<td>8.52529e-001</td>
<td>3.67560e-001</td>
<td>3.46375e-001</td>
<td>1.00426e+000</td>
<td>9.40650e-001</td>
</tr>
<tr>
<td>9.40000e-001</td>
<td>5.65775e-001</td>
<td>6.65759e-001</td>
<td>8.49820e-001</td>
<td>3.41174e-001</td>
<td>3.49943e-001</td>
<td>1.00311e+000</td>
<td>9.49253e-001</td>
</tr>
<tr>
<td>9.60000e-001</td>
<td>5.53170e-001</td>
<td>6.55130e-001</td>
<td>8.44366e-001</td>
<td>2.80000e-001</td>
<td>3.56861e-001</td>
<td>1.00136e+000</td>
<td>9.66334e-001</td>
</tr>
<tr>
<td>9.70000e-001</td>
<td>5.46905e-001</td>
<td>6.49822e-001</td>
<td>8.41623e-001</td>
<td>2.43105e-001</td>
<td>3.60208e-001</td>
<td>1.00076e+000</td>
<td>9.74813e-001</td>
</tr>
<tr>
<td>9.80000e-001</td>
<td>5.40669e-001</td>
<td>6.44520e-001</td>
<td>8.38870e-001</td>
<td>1.98997e-001</td>
<td>3.63481e-001</td>
<td>1.00034e+000</td>
<td>9.83250e-001</td>
</tr>
<tr>
<td>9.90000e-001</td>
<td>5.34460e-001</td>
<td>6.39225e-001</td>
<td>8.36106e-001</td>
<td>1.41067e-001</td>
<td>3.66677e-001</td>
<td>1.00008e+000</td>
<td>9.91646e-001</td>
</tr>
<tr>
<td>1.00000e+000</td>
<td>5.28282e-001</td>
<td>6.33938e-001</td>
<td>8.33333e-001</td>
<td>0.00000e+000</td>
<td>3.69797e-001</td>
<td>1.00000e+000</td>
<td>1.00000e+000</td>
</tr>
<tr>
<td>1.01000e+000</td>
<td>5.22134e-001</td>
<td>6.28660e-001</td>
<td>8.30551e-001</td>
<td>1.41774e-001</td>
<td>3.72840e-001</td>
<td>1.00008e+000</td>
<td>1.00831e+000</td>
</tr>
<tr>
<td>1.02000e+000</td>
<td>5.16018e-001</td>
<td>6.23391e-001</td>
<td>8.27760e-001</td>
<td>2.00998e-001</td>
<td>3.75806e-001</td>
<td>1.00033e+000</td>
<td>1.01658e+000</td>
</tr>
<tr>
<td>1.03000e+000</td>
<td>5.09935e-001</td>
<td>6.18133e-001</td>
<td>8.24960e-001</td>
<td>2.46779e-001</td>
<td>3.78693e-001</td>
<td>1.00074e+000</td>
<td>1.02481e+000</td>
</tr>
<tr>
<td>1.04000e+000</td>
<td>5.03886e-001</td>
<td>6.12887e-001</td>
<td>8.22152e-001</td>
<td>2.85657e-001</td>
<td>3.81502e-001</td>
<td>1.00131e+000</td>
<td>1.03200e+000</td>
</tr>
<tr>
<td>1.05000e+000</td>
<td>4.97872e-001</td>
<td>6.07653e-001</td>
<td>8.19336e-001</td>
<td>3.20156e-001</td>
<td>3.84233e-001</td>
<td>1.00203e+000</td>
<td>1.04114e+000</td>
</tr>
<tr>
<td>1.06000e+000</td>
<td>4.91894e-001</td>
<td>6.02432e-001</td>
<td>8.16513e-001</td>
<td>3.51568e-001</td>
<td>3.86884e-001</td>
<td>1.00291e+000</td>
<td>1.04925e+000</td>
</tr>
<tr>
<td>1.07000e+000</td>
<td>4.85952e-001</td>
<td>5.97225e-001</td>
<td>8.13683e-001</td>
<td>3.80657e-001</td>
<td>3.89456e-001</td>
<td>1.00394e+000</td>
<td>1.05731e+000</td>
</tr>
<tr>
<td>1.08000e+000</td>
<td>4.80047e-001</td>
<td>5.92033e-001</td>
<td>8.10846e-001</td>
<td>4.07922e-001</td>
<td>3.91949e-001</td>
<td>1.00512e+000</td>
<td>1.06533e+000</td>
</tr>
<tr>
<td>1.09000e+000</td>
<td>4.74181e-001</td>
<td>5.86856e-001</td>
<td>8.08002e-001</td>
<td>4.33705e-001</td>
<td>3.94362e-001</td>
<td>1.00645e+000</td>
<td>1.07331e+000</td>
</tr>
<tr>
<td>1.10000e+000</td>
<td>4.68354e-001</td>
<td>5.81696e-001</td>
<td>8.05153e-001</td>
<td>4.58258e-001</td>
<td>3.96696e-001</td>
<td>1.00793e+000</td>
<td>1.08124e+000</td>
</tr>
</tbody>
</table>
**nshktbl**

**Purpose**
Generate a text file containing tables of supersonic flow and normal shock properties. The tables generated by this function may be useful when computational solution of the equations is not practical.

**Synopsis**

nshktbl
nshktbl(Filename,Mn,Gamma)

**Description**

nshktbl uses function ames to generate a table of values for supersonic flow and normal shock properties as a function of Mach numbers from 1 to 10. Properties 9 through 16 of Table 4.1 are written to the text file, nshktbl.txt.

nshktbl(Filename,Mn,Gamma) computes the flow functions and writes the ASCII data to the file specified by the string variable, Filename. Functions are evaluated at Mach numbers specified in Mn. Gamma is an optional scalar variable specifying the ratio of specific heats of the working fluid. If unspecified, a value of 1.4 is used for Gamma.

**See Also**
ames, isentbl, fannotbl, and rayltbl

**Example 4–17:**
Create a table containing supersonic flow and normal shock functions for a range of Mach numbers from 1.0 to 2.5 in increments of 0.1. Results are shown in Table 4.5 on the following page.

```matlab
>> nshktbl('nshktbl.txt.',1.0:0.1:2.5);
```
Table 4.5.—Output of function nshktbl1 for a range of Mach numbers from 1.0 to 2.5

<table>
<thead>
<tr>
<th>M1</th>
<th>Nu</th>
<th>Mu</th>
<th>M2</th>
<th>P2/P1</th>
<th>p2/p1</th>
<th>T2/T1</th>
<th>PT2/PT1</th>
<th>P1/PT2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.00000e+00</td>
<td>0.00000e+00</td>
<td>9.00000e+00</td>
<td>1.00000e+00</td>
<td>1.00000e+00</td>
<td>1.00000e+00</td>
<td>1.00000e+00</td>
<td>1.00000e+00</td>
<td>1.00000e+00</td>
</tr>
<tr>
<td>1.10000e+00</td>
<td>1.3362e+00</td>
<td>6.5380e+00</td>
<td>9.1177e-01</td>
<td>1.2450e+00</td>
<td>1.1690e+00</td>
<td>1.0649e+00</td>
<td>9.9828e-01</td>
<td>4.6885e-01</td>
</tr>
<tr>
<td>1.20000e+00</td>
<td>3.5582e+00</td>
<td>5.6442e+00</td>
<td>8.4217e-01</td>
<td>1.5133e+00</td>
<td>1.3416e+00</td>
<td>1.1279e+00</td>
<td>9.9279e-01</td>
<td>4.1536e-01</td>
</tr>
<tr>
<td>1.30000e+00</td>
<td>6.1702e+00</td>
<td>5.0284e+00</td>
<td>7.8595e-01</td>
<td>1.8050e+00</td>
<td>1.5157e+00</td>
<td>1.1908e+00</td>
<td>9.7937e-01</td>
<td>3.6851e-01</td>
</tr>
<tr>
<td>1.40000e+00</td>
<td>8.9870e+00</td>
<td>4.5584e+00</td>
<td>7.3970e-01</td>
<td>2.1200e+00</td>
<td>1.6896e+00</td>
<td>1.2546e+00</td>
<td>9.5819e-01</td>
<td>3.2795e-01</td>
</tr>
<tr>
<td>1.50000e+00</td>
<td>1.1905e+00</td>
<td>4.1810e+00</td>
<td>7.0108e-01</td>
<td>2.4583e+00</td>
<td>1.8620e+00</td>
<td>1.3202e+00</td>
<td>9.2978e-01</td>
<td>2.9297e-01</td>
</tr>
<tr>
<td>1.60000e+00</td>
<td>1.4864e+00</td>
<td>3.8682e+00</td>
<td>6.6843e-01</td>
<td>2.8200e+00</td>
<td>2.0317e+00</td>
<td>1.3879e+00</td>
<td>8.9520e-01</td>
<td>2.6281e-01</td>
</tr>
<tr>
<td>1.70000e+00</td>
<td>1.7809e+00</td>
<td>3.6031e+00</td>
<td>6.4054e-01</td>
<td>3.2050e+00</td>
<td>2.1972e+00</td>
<td>1.4583e+00</td>
<td>8.5572e-01</td>
<td>2.3675e-01</td>
</tr>
<tr>
<td>1.80000e+00</td>
<td>2.0729e+00</td>
<td>3.3749e+00</td>
<td>6.1650e-01</td>
<td>3.6133e+00</td>
<td>2.3592e+00</td>
<td>1.5358e+00</td>
<td>8.1268e-01</td>
<td>2.1415e-01</td>
</tr>
<tr>
<td>1.90000e+00</td>
<td>2.3586e+00</td>
<td>3.1756e+00</td>
<td>5.9561e-01</td>
<td>4.0450e+00</td>
<td>2.5156e+00</td>
<td>1.6079e+00</td>
<td>7.6735e-01</td>
<td>1.9448e-01</td>
</tr>
<tr>
<td>2.00000e+00</td>
<td>2.6379e+00</td>
<td>3.0000e+00</td>
<td>5.7735e-01</td>
<td>4.5000e+00</td>
<td>2.6666e+00</td>
<td>1.6875e+00</td>
<td>7.2087e-01</td>
<td>1.7729e-01</td>
</tr>
<tr>
<td>2.10000e+00</td>
<td>2.9071e+00</td>
<td>2.8436e+00</td>
<td>5.6127e-01</td>
<td>4.9783e+00</td>
<td>2.8119e+00</td>
<td>1.7704e+00</td>
<td>6.7420e-01</td>
<td>1.6219e-01</td>
</tr>
<tr>
<td>2.20000e+00</td>
<td>3.1735e+00</td>
<td>2.7035e+00</td>
<td>5.4705e-01</td>
<td>5.4800e+00</td>
<td>2.9512e+00</td>
<td>1.8568e+00</td>
<td>6.2813e-01</td>
<td>1.4888e-01</td>
</tr>
<tr>
<td>2.30000e+00</td>
<td>3.4282e+00</td>
<td>2.5771e+00</td>
<td>5.3411e-01</td>
<td>6.0050e+00</td>
<td>3.0845e+00</td>
<td>1.9468e+00</td>
<td>5.8329e-01</td>
<td>1.3710e-01</td>
</tr>
<tr>
<td>2.40000e+00</td>
<td>3.6746e+00</td>
<td>2.4624e+00</td>
<td>5.2311e-01</td>
<td>6.5533e+00</td>
<td>3.2139e+00</td>
<td>2.0403e+00</td>
<td>5.4014e-01</td>
<td>1.2663e-01</td>
</tr>
<tr>
<td>2.50000e+00</td>
<td>3.9123e+00</td>
<td>2.3578e+00</td>
<td>5.1298e-01</td>
<td>7.1250e+00</td>
<td>3.3333e+00</td>
<td>2.1375e+00</td>
<td>4.9901e-01</td>
<td>1.1728e-01</td>
</tr>
<tr>
<td>2.60000e+00</td>
<td>4.1417e+00</td>
<td>2.2619e+00</td>
<td>5.0387e-01</td>
<td>7.7200e+00</td>
<td>3.4899e+00</td>
<td>2.2383e+00</td>
<td>4.6012e-01</td>
<td>1.0891e-01</td>
</tr>
<tr>
<td>2.70000e+00</td>
<td>4.3621e+00</td>
<td>2.1738e+00</td>
<td>4.9563e-01</td>
<td>8.3383e+00</td>
<td>3.5589e+00</td>
<td>2.3429e+00</td>
<td>4.2359e-01</td>
<td>1.0139e-01</td>
</tr>
<tr>
<td>2.80000e+00</td>
<td>4.5745e+00</td>
<td>2.0924e+00</td>
<td>4.8816e-01</td>
<td>8.9800e+00</td>
<td>3.6635e+00</td>
<td>2.4517e+00</td>
<td>3.8946e-01</td>
<td>9.4613e-02</td>
</tr>
<tr>
<td>2.90000e+00</td>
<td>4.7793e+00</td>
<td>2.0171e+00</td>
<td>4.8138e-01</td>
<td>9.6450e+00</td>
<td>3.7628e+00</td>
<td>2.5632e+00</td>
<td>3.5773e-01</td>
<td>8.8478e-02</td>
</tr>
<tr>
<td>3.00000e+00</td>
<td>4.9757e+00</td>
<td>1.9471e+00</td>
<td>4.7519e-01</td>
<td>1.0333e+00</td>
<td>3.8571e+00</td>
<td>2.6790e+00</td>
<td>3.2834e-01</td>
<td>8.2912e-02</td>
</tr>
</tbody>
</table>
oblqshck

Purpose
For steady state supersonic flow, oblqshck computes the angle of the oblique shock that results from compressively turning the flow through angle (δ). One of two solutions will occur for each Mach number specified, a weak shock solution, or a strong shock solution. Angles have units of degrees. The flow situation is similar to that depicted in Figure 4.4 (pp. 19).

Synopsis
oblqshck
ThetaW=oblqshck(M1,delta)
[ThetaW,ThetaS]=oblqshck(M1,delta,gamma)

Description
oblqshck by itself generates a plot showing shock angle vs. deflection angle for lines of constant Mach from 1 to 20. This plot is a representation of Chart 2 in (ref. 1).

[ThetaW,ThetaS]=oblqshck(M1,delta) computes both the weak oblique shock angle, ThetaW, and the strong oblique shock angle, ThetaS, that are the result of compressively turning supersonic flow at Mach number, M1, through angle delta. M1 and delta may be either scalars or vectors. If both are vectors, they must have identical dimensions.

[ThetaW,ThetaS,DELmax,DELson]=oblqshck(M1,delta,gamma) uses optional scalar input Gamma to specify the ratio of specific heats of the working fluid. If unspecified, a value of 1.4 is used for Gamma. In addition to returning the oblique shock angles, this form also returns the maximum flow deflection angle, DELmax, and the flow deflection angle that results in sonic flow downstream of the oblique shock, DELson.

Algorithm
oblqshck uses the upstream Mach number, the flow deflection angle, and the ratio of specific heats to calculate the solution of the cubic equation for both weak and strong oblique shock angles using the method given in (ref. 5). If flow deflection angles are specified as outputs, oblqshck uses functions deltamax and deltason to compute values for those parameters at Mach number(s), M1.

See Also
deltason, deltamax, oblqw12, and oblqw21

Example 4–18:
Replicate the results in Chart 2 of (ref. 1).

>> oblqshck
Example 4–19:

Given freestream airflow at Mach 2.2, calculate the shock angle that results from turning the flow through a range of deflection angles from zero to the maximum deflection possible without separating the flow.

```
>> M1=2.2, DELmax=deltamax(M)
M1 =
   2.2000

DELmax =
   26.1028

>> delta=DELmax*[0:10]'/10; M1=M1*ones(size(delta));

>> ThetaW=oblqshck(M1,delta); [delta ThetaW]
ans =
   0   27.0357
   2.6103   29.0843
   5.2206   31.2899
   7.8308   33.6666
  10.4411   36.2338
  13.0514   39.0200
  15.6617   42.0707
  18.2719   45.4660
  20.8822   49.3676
  23.4925   54.2056
  26.1028   64.6203
```
oblqw12

Purpose
For steady state supersonic flow, oblqw12 uses the upstream fluid properties to compute properties of the flow downstream of a weak oblique shock. The flow situation is similar to that depicted in Figure 4.4 (pp. 19).

Synopsis
oblqw12 
M2=oblqw12(M1,delta)  
[M2,Theta,PTratio]=oblqw12(M1,delta,gamma)

Description
oblqw12 by itself generates a series of plots showing properties of oblique shocks for lines of constant Mach from 1 to 20. Figure 4.9(a and b) replicates the weak shock portions of Charts 2 and 4 in (ref. 1). Figure 4.9(c) shows variations in total pressure across and oblique shock as a function shock angle. In these figures, the solid lines represent lines of constant Mach number.

M2=oblqw12(M1,delta) computes the Mach number, M2, downstream of an oblique shock that results from compressively turning supersonic flow at Mach number, M1, through angle delta. M1 and delta may be either scalars or vectors. If both are vectors, they must have identical dimensions.

[M2,Theta,PTratio]=oblqw12(M1,delta,gamma) uses optional scalar input Gamma to specify the ratio of specific heats of the working fluid. If unspecified, a value of 1.4 is used. In addition to returning the downstream Mach number, this form also returns the resulting oblique shock angle as Theta, and the total pressure ratio across the shock, Pr2/Pr1, as PTratio.

Algorithm
oblqw12 uses the upstream Mach number, the flow deflection angle, and the ratio of specific heats to calculate the solution of the cubic equation for both weak and strong oblique shock angles using the method given in reference 5. It then uses equation 131 and 142 from reference 1—alternatively, equation 7.31 and 7.25 from reference 4—to calculate the downstream Mach number and total pressure ratio across the oblique shock.

See Also
deltason, deltamax, oblqshck, and oblqw21

Example 4–20:
Replicate the results in Chart 2 and 4 of reference 1. Results are shown in Figure 4.10.

>> oblqshck
Figure 4.10.—Plots generated by function \texttt{oblqw12}.
Example 4–21:
Given flow deflection of freestream airflow at Mach 2.2, calculate the shock angle that results from turning the flow through a range of deflection angles from zero to the maximum deflection possible without separating the flow.

\[
\text{>> M1}=2.2, \text{DELM}=\text{deltam}(M)\\
\text{M1} =\\
\begin{array}{c}
2.2000\\
\end{array}\\
\text{DELM} =\\
\begin{array}{c}
26.1028\\
\end{array}\\
\text{>> deltam}=\text{DELM}*[0:10]'/10; \text{M1}=\text{M1}\text{*ones(size(delta))};
\text{>> ThetaW}=\text{oblqshck}(M1,delta); [\text{delta ThetaW}]
\text{ans} =
\begin{array}{cc}
0 & 27.0357\\
2.6103 & 29.0843\\
5.2206 & 31.2899\\
7.8308 & 33.6666\\
10.4411 & 36.2338\\
13.0514 & 39.0200\\
15.6617 & 42.0707\\
18.2719 & 45.4660\\
20.8822 & 49.3676\\
23.4925 & 54.2056\\
26.1028 & 64.6203
\end{array}
\]
oblqw21

Purpose
For steady state supersonic flow, oblqw21 uses the downstream fluid properties to compute properties of the flow upstream of a weak oblique shock. The flow situation is similar to that depicted in Figure 4.4 (pp. 19).

Synopsis
oblqw21
M1=oblqw21(M2,delta)
[M1,Theta,PTratio]=oblqw21(M2,delta,gamma)

Description
oblqw21 generates a consistency check for the oblique shock equations used by functions oblqshck, oblqw12, and oblqw21. The absolute error is plotted as a function of flow deflection angle for lines of constant Mach number.

M1=oblqw21(M2,delta) computes the Mach number, M1, upstream of an oblique shock that is required to achieve downstream Mach number, M2, after compressively turning the flow through angle delta. M2 and delta may be either scalars or vectors. If both are vectors, they must have identical dimensions.

[M1,Theta,PTratio]=oblqw21(M2,delta,gamma) uses optional scalar input Gamma to specify the ratio of specific heats of the working fluid. If unspecified, a value of 1.4 is used. In addition to returning the downstream Mach number, this form also returns the resulting oblique shock angle as Theta, and the total pressure ratio across the shock, P_{t,2}/P_{t,1}, as PTratio.

Algorithm
oblqw21 uses MATLAB's fminbnd function to find a solution to the nonlinear equation for downstream Mach number as a function of upstream Mach number, the flow deflection angle, and the ratio of specific heats. fminbnd solves the nonlinear equation by employing an inline function to compute the error between the user specified Mach number, M2, and a downstream Mach number, M_{2,Guess}, that is calculated using function oblqw12 and a guess for the upstream Mach number, M_{1,Guess}. The search is arbitrarily constrained to Mach numbers less than or equal to 100. Solutions associated with Mach numbers larger than 100 are returned as NaN (i.e., not a number).

See Also
deltason, deltamax, oblqshck, and oblqw12

Example 4–22:
Generate the consistency check for the oblique shock functions.

>> oblw21
Example 4–23:
Given the values for downstream Mach number, \( M_2 \), shown below, calculate the upstream Mach number required to produce \( M_2 \) when the flow deflection angle is 0.5 degrees. Also calculate the oblique shock angle and the total pressure ratio across the oblique shock.

\[
\begin{align*}
M_2 &= [0.9000 \quad 1.0000 \quad 1.9819 \quad 2.9742 \quad 3.9624 \ldots \\
&\quad 61.9798 \quad 69.5625 \quad 76.8279 \quad 83.7768 \quad 90.4113];
\end{align*}
\]

\[
\begin{align*}
\delta &= 0.5;
\end{align*}
\]

\[
\begin{align*}
[M_1, \Theta, P_{Tratio}] &= \text{oblqw21}(M_2, \delta);
\end{align*}
\]

Warning: FANNO: Some Mach number solutions exceed internal limit (M \( > 100 \)). Solutions set to NaN

In oblqw21 at 119
\[\begin{array}{cccc}
\text{M2}(:) & \text{M1}(:) & \text{Theta}(:) & \text{PTratio}(:)
\end{array}\]
\[
\begin{array}{cccc}
0.9000 & \text{NaN} & \text{NaN} & \text{NaN} \\
1.0000 & 1.0490 & 77.7993 & 1.0000 \\
1.9819 & 2.0000 & 30.4029 & 1.0000 \\
2.9742 & 3.0000 & 19.8116 & 1.0000 \\
3.9624 & 4.0000 & 14.8009 & 1.0000 \\
61.9798 & 70.0000 & 1.1719 & 0.9500 \\
69.5625 & 80.0000 & 1.0766 & 0.9288 \\
76.8279 & 90.0000 & 1.0038 & 0.9037 \\
83.7768 & 99.9999 & 0.9468 & 0.8750 \\
90.4113 & \text{NaN} & \text{NaN} & \text{NaN}
\end{array}
\]

Note that a weak oblique shock solution does not exist for the case where \(M2 = 0.9\). Therefore, the associated solutions are defined as \text{NaN}. Also, because the search space is limited to Mach numbers less than or equal to 100, a solution for \(M2 = 90.4113, M1 = 110\), is not computed by \text{oblqw21}. 
rayleigh

Purpose
Solve the equations for one-dimensional steady flow in a constant area duct with heat transfer.

Synopsis
Properties=rayleigh(VarIn,ValuesIn,VarsOut)
Properties=rayleigh(VarIn,ValuesIn,VarsOut,Gamma)
[Properties,PltLbls]=rayleigh(VarIn,ValuesIn,VarsOut,Gamma)
rayleigh

Description
Properties=rayleigh(VarIn,ValuesIn,VarsOut), given a number designating one of the flow properties listed in Table 4.6 and a value or vector of values for that flow property, rayleigh computes corresponding values for flow though a constant area duct with heat transfer. VarIn is a scalar that specifies the property used as the input (independent variable). ValuesIn may be a scalar or vector and contains values of the independent variable for which the other properties will be computed. VarsOut contains a list of indices corresponding to the flow properties listed in Table 4.6. Indices specified in VarsOut may be in any order and may be repeated as desired by the user. Results are returned in the Properties matrix. Columns in this matrix correspond to indices specified in VarsOut. Rows of the Properties matrix contain results corresponding to the elements of ValuesIn.

Note that, when properties 2, 3, or 5 are used as the independent variable, the solution is double-valued. The double-valued solution is provided by making Properties a cell array. Properties{1} contains values of the solution associated with the smaller Mach number, while Properties{2} contains the solution associated with the larger Mach number.

Properties=rayleigh(VarIn,ValuesIn,VarsOut,Gamma) provides a mechanism for specifying values for the ratio of specific heats of the working fluid via Gamma. Gamma is optional. If unspecified, a value of 1.4, the value of the ratio of specific heats of air at standard temperature and pressure, is used. If specified, Gamma may be defined as either a scalar or a vector. If it is a vector, it must have the same length as ValuesIn.

[Properties,PltLbls]=rayleigh(VarIn,ValuesIn,VarsOut,Gamma), in addition to returning the properties of the fluid at user specified conditions, also returns a cell array, PltLbls, containing text strings that may be used when plotting the results.

rayleigh by itself calls raylplt which plots the Rayleigh-line flow properties versus Mach number. The resulting plot is similar to figure 6.5 in reference 4.
Table 4.6.—Description of Flow Properties Computed by Function `rayleigh`

<table>
<thead>
<tr>
<th>REF. INDEX</th>
<th>PROPERTY</th>
<th>REF. EQN.</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>$M_1$</td>
<td>Eq. 6.22</td>
<td>Ratio of total temperature at $M_1$ to total temperature at sonic conditions.</td>
</tr>
<tr>
<td>2.</td>
<td>$T_{t,1}/T_{t,s}$</td>
<td>Eq. 6.19</td>
<td>Ratio of static temperature at $M_1$ to static temperature at sonic conditions.</td>
</tr>
<tr>
<td>3.</td>
<td>$T_{s,1}/T_{s,s}$</td>
<td>Eq. 6.20</td>
<td>Ratio of static pressure at $M_1$ to static pressure at sonic conditions.</td>
</tr>
<tr>
<td>4.</td>
<td>$P_{t,1}/P_{t,s}$</td>
<td>Eq. 6.21</td>
<td>Ratio of total pressure at $M_1$ to total pressure at sonic conditions.</td>
</tr>
<tr>
<td>5.</td>
<td>$P_{s,1}/P_{s,s}$</td>
<td>Eq. 6.23</td>
<td>Ratio of flow velocity at $M_1$ to flow velocity at sonic conditions.</td>
</tr>
</tbody>
</table>

* Similar equations may also be found in reference 2 on p. 196.

Algorithm

`rayleigh` determines the desired flow properties by first obtaining a Mach number solution for each value, `ValuesIn`, of the user specified flow property, `VarIn`. The resulting Mach numbers are then used to compute the other properties, `VarsOut`. Most of the flow equations may be manipulated analytically to obtain Mach number as a function of the other properties. However, in the case of total pressure ratio, a nonlinear relationships exists which has no simple analytical solution. In this case, MATLAB’s `fminbnd` function is used determine an approximate solution for Mach number from the nonlinear equation. The search is arbitrarily constrained to Mach numbers less than or equal to 100. Solutions associated with Mach numbers larger than 100 are returned as NaN (i.e., not a number).

See Also

`raylerr`, `raylplt`, and `rayltbl`

Example 4.24:

For air flowing at Mach 0.72 and 2.85, determine the Rayleigh-line flow properties.

```matlab
>> rayleigh(1,[0.72 2.85],2:6)
ans =
    0.9221    1.0026    1.3907    1.0376    0.7209
    0.6685    0.3057    0.1940    3.0014    1.5757
```

Example 4.25:

Given sonic conditions at a point in a constant area duct with total temperature of 1000 K and total pressure of 300 kPa, find the Mach number and total pressure as the total temperature decreases to 800, 600, and 400 K.

First, divide duct total temperature by sonic total temperature to obtain the ratio of total temperatures.
Next, use function `rayleigh` to compute the Mach number (ref. index 1) and total pressure ratio (ref. index 5) for the specified total temperature ratios.

```
>> Properties=rayleigh(2,TTratio,[1 5])
Properties =
    [3x2 double]    [3x2 double]
```

The solution is double-valued. The subsonic solution is returned as `Properties{1}`, and the supersonic solution is returned as `Properties{2}`. Mach number values are returned in column one of each of the solutions. The subsonic and supersonic solutions for Mach number are shown below in columns one and two, respectively. Here, the rows correspond to the elements of `TTratio`, with the first row corresponding to the element one.

```
>> M1=[Properties{1}(:,1) Properties{2}(:,1)]
M1 =
    0.6884    1.5368
    0.5423    2.2361
    0.4415    3.7749
```

Values for total pressure ratio are returned in column two of each of the cells in `Properties`. The subsonic and supersonic solutions are shown below in columns one and two, respectively. Here, the rows correspond to the elements of `TTratio`, with the first row corresponding to the element one. The total pressure ratios are multiplied by the pressure ratio at sonic conditions (i.e., 300 kPa) to obtain total pressure at the specified total temperature conditions.

```
>> PT=[Properties{2}(:,2); Properties{2}(:,2)]*300
PT =
    1.0000e+003 *
    0.3139    0.3421
    0.3291    0.5379
    0.3416    2.0328
```

Note that `NaN` results would imply that the intermediate solution for Mach number exceeded the internal limit (i.e., M > 100).
raylerr

Purpose
Show the computational errors that result when using function rayleigh to solve equations for one-dimensional steady adiabatic flow in a constant-area duct with friction.

Synopsis
raylerr
[error,M1]=raylerr

Description
raylerr computes the error between Mach numbers used as inputs to function rayleigh and Mach numbers calculated from the output of function rayleigh. The results are plotted as absolute and percent errors versus Mach number for each of the flow functions shown in Table 4.6.

[error,M1]=raylerr returns the computed error in error. If specified, M1 contains the initial vector of Mach numbers.

Algorithm
raylerr first generates a logarithmically spaced vector of 250 Mach numbers from 0.01 to 10. This vector also includes critical Mach number values where numerical stability is important, such as saddle points. raylerr then uses function rayleigh to calculate each of the Rayleigh-line flow properties corresponding to those Mach numbers. The functions of Mach number, obtained from rayleigh, are then used as input to the rayleigh function in order to obtain a Mach number which corresponds to the function value. Theoretically, the initial and computed Mach numbers should be the same. In general, they are not due to round off, truncation, convergence, and/or optimization errors. The difference in the two Mach numbers is returned as the error in the calculations.

See Also
rayleigh, raylplt, and rayltbl

Example 4.26:
Perform a consistency check on the calculations in function rayleigh. Plots are shown in Figure 4.12(a to c).

>> raylerr
Figure 4.12.—Output of function ray1err as computed on an Intel Pentium4 processor-based computer running MATLAB® 7.
Figure 4.12.—Output of function raylerr as computed on an Intel Pentium4 processor-based computer running MATLAB® 7 (continued).
raylplt

Purpose
Plot properties for Rayleigh-line flow, i.e., one-dimensional steady adiabatic flow in a constant-area duct with friction.

Synopsis
raylplt
raylplt(MNmin,MNmax)
raylplt(MNmin,MNmax,Npts)
raylplt(MNmin,MNmax,Npts,Gamma)

Description
raylplt uses function rayleigh to compute and plot the Rayleigh-line flow properties at 250 points between Mach 0.05 and Mach 2.5 when the ratio of specific heats of the fluid is 1.4. This plot resembles figure 6.5 in reference 4.

raylplt(MNmin,MNmax) plots results for a range of user specified Mach numbers where: MNmin is the minimum Mach number; and MNmax is the maximum Mach number.

raylplt(MNmin,MNmax,Npts) in addition to allowing the user to specify the range of Mach numbers used, this form allows the user to specify the number of data points, Npts, used to plot each curve.

raylplt(MNmin,MNmax,Npts,Gamma) in addition to allowing the user to specify Mach No. and number of points per curve, this form also allows the user to specify a scalar value for the ratio of specific heats, Gamma, of the fluid.

Algorithm
raylplt first generates a logarithmically spaced vector of 250 Mach numbers from 0.05 to 2.5. This vector also includes critical Mach number values where numerical stability is important, such as solution saddle points. raylplt then uses this vector as inputs to function rayleigh which is used to calculate each of the isentropic flow properties and the normal shock properties corresponding to those Mach numbers. The resulting values are plotted versus Mach number to provide the user a graphical understanding of the relationship between flow properties and Mach number.

See Also
rayleigh, raylerr, and rayltbl

Example 4.27:
Plot Rayleigh-line flow properties over a range of Mach numbers from 0.5 to 2.5. The resulting plot is shown in Figure 4.13.

>> raylplt
Figure 4.13.—Rayleigh-line flow properties as generated by function raylplt.
rayltbl

Purpose
Generate a text file containing tables of Rayleigh-line flow properties. The tables generated by this function may be useful when computational solution of the equations is not practical.

Synopsis
rayltbl
rayltbl(Filename,Mn,Gamma)

Description
rayltbl uses function rayleigh to generate a table of values for Rayleigh-line flow properties as a function of Mach numbers from 0.01 to 10. Properties 2 through 6 of Table 4.2 are written to the text file, rayltbl.txt.

rayltbl(Filename,Mn,Gamma) computes the flow functions and writes the ASCII data to the file specified by the string variable, Filename. Functions are evaluated at Mach numbers specified in Mn. Gamma is an optional scalar variable specifying the ratio of specific heats of the working fluid. If unspecified, a value of 1.4 is used for Gamma.

See Also
rayleigh, raylplt, and rayltbl

Example 4–28:
Create a table containing values for Rayleigh-line flow functions over a range of Mach numbers from 0.55 to 0.59 in increments of 0.01, and from 2.0 to 2.4 in increments of 0.1. Results are shown in Table 4.7 on the following page.

>> M1=[0.55:0.01:0.59  2.0:0.1:2.4];

>> rayltbl('rayltbl.txt.',M1)
Table 4.7.—Output of function ray1tbl for a range of Mach numbers specified in Example 4.28

<table>
<thead>
<tr>
<th>M</th>
<th>TT/TT*</th>
<th>T/T*</th>
<th>P/P*</th>
<th>PT/PT*</th>
<th>V/V*</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.50000e-01</td>
<td>7.59910e-001</td>
<td>8.59870e-001</td>
<td>1.68599e+000</td>
<td>1.09397e+000</td>
<td>5.10011e-001</td>
</tr>
<tr>
<td>5.60000e-01</td>
<td>7.72486e-001</td>
<td>8.72274e-001</td>
<td>1.66778e+000</td>
<td>1.09011e+000</td>
<td>5.23015e-001</td>
</tr>
<tr>
<td>5.70000e-01</td>
<td>7.84675e-001</td>
<td>8.84158e-001</td>
<td>1.64964e+000</td>
<td>1.08630e+000</td>
<td>5.35969e-001</td>
</tr>
<tr>
<td>5.80000e-01</td>
<td>7.96478e-001</td>
<td>8.95523e-001</td>
<td>1.63159e+000</td>
<td>1.08256e+000</td>
<td>5.48866e-001</td>
</tr>
<tr>
<td>5.90000e-01</td>
<td>8.07894e-001</td>
<td>9.06371e-001</td>
<td>1.61362e+000</td>
<td>1.07887e+000</td>
<td>5.61701e-001</td>
</tr>
<tr>
<td>2.00000e+00</td>
<td>7.93388e-001</td>
<td>5.28926e-001</td>
<td>3.63636e-001</td>
<td>1.50310e+000</td>
<td>1.45455e+000</td>
</tr>
<tr>
<td>2.10000e+00</td>
<td>7.74064e-001</td>
<td>4.93558e-001</td>
<td>3.34541e-001</td>
<td>1.61616e+000</td>
<td>1.47533e+000</td>
</tr>
<tr>
<td>2.20000e+00</td>
<td>7.56135e-001</td>
<td>4.61058e-001</td>
<td>3.08642e-001</td>
<td>1.74345e+000</td>
<td>1.49383e+000</td>
</tr>
<tr>
<td>2.30000e+00</td>
<td>7.39543e-001</td>
<td>4.31220e-001</td>
<td>2.85510e-001</td>
<td>1.88602e+000</td>
<td>1.51035e+000</td>
</tr>
<tr>
<td>2.40000e+00</td>
<td>7.24213e-001</td>
<td>4.03836e-001</td>
<td>2.64784e-001</td>
<td>2.04505e+000</td>
<td>1.52515e+000</td>
</tr>
</tbody>
</table>
5. References

This report provides a user guide for the Compressible Flow Toolbox, a collection of algorithms that solve almost 300 linear and nonlinear classical compressible flow relations. The algorithms, implemented in the popular MATLAB® programming language, are useful for analysis of one-dimensional steady flow with constant entropy, friction, heat transfer, or shock discontinuities. The solutions do not include any gas dissociative effects. The toolbox also contains functions for comparing and validating the equation-solving algorithms against solutions previously published in the open literature. The classical equations solved by the Compressible Flow Toolbox are: isentropic-flow equations, Fanno flow equations (pertaining to flow of an ideal gas in a pipe with friction), Rayleigh flow equations (pertaining to frictionless flow of an ideal gas, with heat transfer, in a pipe of constant cross section.), normal-shock equations, oblique-shock equations, and Prandtl-Meyer expansion equations. At the time this report was published, the Compressible Flow Toolbox was available without cost from the NASA Software Repository.