**XFOIL 6.5 User Primer** last update 5 May 1995

Mark Drela, MIT Aero & Astro

## General Description

XFOIL is an interactive program for the design and analysis of subsonic isolated airfoils. It consists of a collection of menu-driven routines which perform various useful functions such as:

- Viscous (or inviscid) analysis of an existing airfoil, allowing
    * forced or free transition
    * transitional separation bubble(s)
    * limited trailing edge separation
    * lift and drag predictions just beyond CLmax
    * Karman-Tsien compressibility correction

- Airfoil design and redesign by interactive specification of a surface speed distribution via CRT cursor or mouse. Two such facilities are implemented.
    * Full-Inverse, based on a complex-mapping formulation
    * Mixed-Inverse, an extension of XFOIL's basic panel method

  Full-inverse allows multi-point design, while Mixed-inverse allows relatively strict geometry control over parts of the airfoil.

- Airfoil redesign by interactive specification of new geometric parameters such as
    * new max thickness and/or camber
    * new LE radius
    * new TE thickness
    * new camber line via geometry specification
    * new camber line via loading change specification
    * flap deflection
    * explicit contour geometry (via CRT cursor)

- Drag polar calculation with fixed or varying Reynolds and/or Mach numbers.

- Writing and reading of airfoil geometry and polar save files

- Plotting of geometry, pressure distributions, and polars
  (Versaplot-derivative plot package used)

XFOIL will comfortably run on an old MicroVAX II, but it is much more effective on a modern RISC Workstation. A high-end PC would also be suitable, although graphics routines for PCs are not available at present. The source code of XFOIL is Fortran 77.

*Theory Reference*

Drela, M.,
XFOIL: An Analysis and Design System for Low Reynolds Number Airfoils, Conference on Low Reynolds Number Airfoil Aerodynamics, University of Notre Dame, June 1989.

The above also appears in:
*Low Reynolds Number Aerodynamics*. T.J. Mueller (Editor). Lecture Notes in Engineering #54. Springer Verlag. 1989.
ISBN 3-540-51884-3/ISBN 0-387-51884-3

## Inviscid Formulation

The inviscid formulation of XFOIL is a simple linear-vorticity stream function panel method. A finite trailing edge base thickness is modeled with a source panel. The equations are closed with an explicit Kutta condition. A high-resolution inviscid calculation with the default 140 panels requires seconds to execute on a RISC workstation. Subsequent operating points for the same airfoil but different angles of attack are obtained nearly instantly.

A Karman-Tsien compressibility correction is incorporated, allowing good compressible predictions all the way to sonic conditions. The theoretical foundation of the Karman-Tsien correction breaks down in supersonic flow, and as a result accuracy rapidly degrades as the transonic regime is entered. Of course, shocked flows cannot be predicted with any certainty.

## Inverse Formulation

There are two types of inverse methods incorporated in XFOIL:Full-Inverse and Mixed-Inverse. The Full-Inverse formulation is essentially Lighthill's and van Ingen's complex mapping method, which is also used in the Eppler code. It calculates the entire airfoil geometry from the entire surface speed distribution. The Mixed-Inverse formulation is simply the inviscid panel formulation (the discrete governing equations are identical) except that instead of the panel vortex strengths being the unknowns, the panel node coordinates are treated as unknowns wherever the surface speed is prescribed. Only a part of the airfoil is altered at any one time, as will be described later. Allowing the panel geometry to be a variable results in a non-linear problem, but this is solved in a straightforward manner with a full-Newton method.

## Viscous Formulation

The boundary layers and wake are described with a two-equation lagged dissipation integral BL formulation and an envelope e^n transition criterion, both taken from the transonic analysis/design ISES code.The entire viscous solution (boundary layers and wake) is strongly interacted with the incompressible potential flow via the surface transpiration model (the alternative displacement body model is used in ISES). This permits proper calculation of limited separation regions. The drag is determined from the wake momentum thickness far downstream. A special treatment is used for a blunt trailing edge which fairly accurately accounts for base drag. In fact, it is generally preferable to set a very small trailing edge thickness (0.0005 x chord , say) even if the airfoil has a "sharp" trailing edge. The inviscid panel method uses an ad-hoc vorticity extrapolation scheme for a sharp trailing edge, which occasionally leads to difficulties in viscous cases for arcane reasons -- the overall viscous-case Newton matrix is much better conditioned with the blunt trailing edge than with a sharp trailing edge, even if the blunt trailingedge is very thin and "practically sharp".

The total velocity at each point on the airfoil surface and wake, with contributions from the freestream, the airfoil surface vorticity, and the equivalent viscous source distribution, is obtained from the panel solution with the Karman-Tsien correction added. This is incorporated into the viscous equations, yielding a nonlinear elliptic system which is readily solved by a full-Newton method as in the ISES code. Execution times are quite rapid, requiring about 10 seconds on a RISC workstation for a high-resolution calculation with 140 panels.  For a  sequence of closely spaced angles of attack (as in a polar), the calculation time per point can be substantially smaller.

If lift is specified, then the wake trajectory for a viscous calculation  is taken from an inviscid solution at the specified lift.  If alpha is specified, then the wake trajectory is taken from an inviscid solution at that alpha.  This is not strictly correct, since viscous effects will in general decrease lift and change the trajectory.  This secondary correction is not performed, since a new source influence matrix would have to be calculated each time the wake trajectory is changed.  This would result in unreasonably long calculation times.  The effect of this approximation on the overall accuracy is small, and will be felt mainly near or past stall, where accuracy tends to degrade anyway.  In attached cases, the effect of the incorrect wake trajectory is imperceptible.

**Program Execution**

At any one time, XFOIL has one current airfoil geometry and operating  state stored in its arrays.  There is also a temporary buffer airfoil which is edited by the geometry modification routines.  When the  program is fired up, the following top level menu and prompt appear:

```
      QUIT  Exit program
     .OPER  Direct operating point(s)
     .MDES  Complex mapping design routine
     .QDES  Surface speed design routine
     .GDES  Geometry design routine
      SAVE  Write airfoil to generic disk file
      ISAV  Write airfoil to ISES coordinate file
      MSAV  Write airfoil to MSES coordinate file
      LOAD  Read buffer airfoil from disk file
      NORM  Buffer airfoil normalization toggle
      NACA  Set NACA 4,5-digit airfoil and buffer airfoil
      PANE  Generate paneled airfoil from buffer airfoil
     .PPAR  Show/change paneling
      WDEF  Write  xfoil.def  file with current settings
      RDEF  Read   xfoil.def  file
      SIZE  Change plot size
      NAME  Change airfoil name

         XFOIL  c>
```

The commands preceded by a period place the user in another lower-level menu.  The other commands are executed immediately  and the user is prompted for another top level command.  The LOAD command, for instance, is used to read in a formatted airfoil coordinate file with the coordinates running from thetrailing edge, round the leading edge, back to the trailing edge in either direction:

```
X(1)  Y(1)
X(2)  Y(2)
 .     .
 .     .
X(N)  Y(N)
```

XFOIL will normally perform all operations on an airfoil with the same shape and location in cartesian space as the input airfoil. However, if the normalization flag is set (toggled with the NORM command), the airfoil coordinates will be immediately normalized to unit chord and the leading edge will be placed at the origin. A message is printed to remind the user.

Whether the coordinates are normalized or not, they are stored as the "buffer" airfoil, and the program then generates the "current" working airfoil by setting up its own panel node distribution on the splined buffer airfoil shape. Panel density is increased in areas of high curvature (i.e. the leading edge) and at the trailing edge to a degree specified by the user. The user can also increase panel density over one additional interval on each airfoil side, perhaps near transition. The airfoil paneling can be displayed and/or modified with PPAR.

In some cases it is desirable to use the input airfoil points as the panel nodes, bypassing the paneling routine. This is done with the command sequence

```
 LOAD
 GDES
 EXEC
```

which simply copies the input buffer airfoil into the current working airfoil.

The LOAD command can also read an ISES- or MSES-format coordinate file.An ISES coordinate file has the same format as the generic file above,except that the first line specifies the airfoil name, and the second line specifies ISES computational domain parameters (not used in XFOIL). An MSES coordinate file is like the ISES file except that it can contain multiple elements, each one separated by the line

```
999.0 999.0
```

The user is asked which of these elements is to be read in. When issuing the MSAV command, the user is also asked which element in the file is to be overwritten. XFOIL can thus be used to easily "edit" individual elements in MSES multielement configurations. Of course, normalization should not be performed on an element if it is to be written back to the same multielement file.

## Units

Most XFOIL operations are performed on the airfoil's cartesian coordinates x,y , which do not necessarily have a unit chord c. Since the chord is difficult to define for odd shapes, the XFOIL force coefficients CL, CD, CM are obtained by normalizing the forces and moment with only the freestream dynamic pressure (the reference chord is assumed to be unity). Likewise, the XFOIL Reynolds number RE is defined with the freestream velocity and viscosity, and an implied unit chord:

```
 CL = L / q          |     V = freestream speed
 CD = D / q          |     v = freestream kinematic viscosity
 CM = M / q          |     r = freestream density
 RE = V / v          |     q = 0.5 r V^2
```

The conventional definitions are

```
 Cl = L / q c
 Cd = D / q c
 Cm = M / q c^2
 Rc = V c / v
```

so that the conventional and XFOIL definitions differ only by the chord factor  c.

For example, a NACA 4412 airfoil is operated in the OPER menu at

```
 RE   = 500000
 ALFA = 3
```

first with chord=1.0, and then with chord=0.5 (changed with SCAL  command in the GDES menu, say).  The results produced by XFOIL are:

```
 c = 1.0 :    CL = 0.80    CD = 0.0082    (RE = 500000, Rc = 500000)
 c = 0.5 :    CL = 0.40    CD = 0.0053    (RE = 500000, Rc = 250000)
```

Since CL is not normalized with the chord, it is nearly proportional to the airfoil size.  It is not exactly proportional, since the true chord Reynolds number Rc is different, and there is always a weak  Reynolds number effect on lift.  In contrast,  the CD for the smaller  airfoil is significantly greater than 1/2 times the larger-airfoil CD,  since chord Reynolds number has a significant impact on profile drag.   Repeating the c = 0.5 case at RE = 1000000, produces the expected  result that CL and CD are exactly 1/2 times their c = 1.0 values.

```
 c = 0.5 :    CL = 0.40    CD = 0.0041    (RE = 1000000, Rc = 500000)
```

Although XFOIL performs its operations with no regard to the size  of the airfoil, some quantities are nevertheless defined in terms  of the chord length.  Examples are the camber line shape and BL trip  locations, which are specified in terms of the relative x/c,y/c along  and normal to the airfoil chord line.  This is done only for the user's convenience.  In the input and output labeling, "x,y" always refer  based coordinates which are shifted, rotated, and scaled so that  the airfoil's leading  edge is at (x/c,y/c) = (0,0), and  the airfoil's trailing edge is at (x/c,y/c) = (1,0).  The two  systems cooincide only if the airfoil is normalized.

## Analysis

Most of the commands in the top level XFOIL menu merely put the user into some lower command level with its own menu and prompt. Issuing The OPER command, for instance, will produce the prompt

```
    .OPERi  c>
```

Typing a " ? " will result in the OPER analysis menu being displayed:

```
     <cr>  Return to top level
     VISC  Viscous/Inviscid toggle
     HARD  Hardcopy enable/disable toggle
     SIZE  Change plot size
     PACC  Auto polar accumulation enable/disable toggle
     PADD  Add point to polar save file and/or dump file
     PFIL  Specify new polar save and/or dump filenames
    .VPAR  Change BL parameter(s)
     RE    Change Reynolds number
     MACH  Change Mach number
     TYPE  Change type of Mach,Re variation with CL
     ALFA  Prescribe alpha
     CLI   Prescribe inviscid CL
     CL    Prescribe CL
     ASEQ  Prescribe a sequence of alphas
     CSEQ  Prescribe a sequence of CLs
     PREF  Reference Cp overlay enable/disable toggle
     FREF  Reference CL,CD... display enable/disable toggle
     CPX   Plot Cp vs x
     CPXG  Plot Cp vs x with grid overlay
     CPV   Plot airfoil with pressure vectors (gee wiz)
     FMOM  Calculate flap hinge moment and forces
     VELS  Calculate velocity components at a point
     DUMP  Output Ue,Dstar,Theta,Cf vs s,x,y to file
    .VPLO  BL variable plots
    .ANNO  Annotate plot
```

The VPAR, VPLO, and ANNO commands will put the user into yet another menu level, while all the other commands are executed immediately. Hopefully, most of them are self-explanatory. For inviscid cases, the CLI and CL commands are identical. For viscous cases, CLI is equivalent to specifying alpha, this being determined a priori from the specified lift coefficient via an inviscid solution. CL will return a viscous solution with the specified true viscous lift coefficient at an alpha which is determined as part of the solution (prescribing a CL above CLmax will cause serious problems, however!). The user is always prompted for any required input. When in doubt, typing a " ? " will always produce a menu.

After an ALFA, CL, or CLI command is executed, the Cp vs x distribution is displayed, and can be displayed again at any time with CPX or CPXG. If the viscous mode is active, the true viscous Cp is shown as a solid line, and the inviscid Cp at that same alpha is shown as a dashed line. Each dash covers one panel, so the local dashed line density is also a useful visual indicator of panel resolution quality. If the inviscid mode is active, only the inviscid Cp is shown as a solid line.

If the Cp reference data overlay option is enabled with PREF, initiating a Cp vs x plot will first result in the user being prompted for a formatted data file with the following format:

```
x(1)  Cp(1)
x(2)  Cp(2)
 .       .
.    .
```

The Cp vs x plot is then displayed as usual but with the data overlaid. If FREF has been issued previously, then numerical reference values for CL, CD, etc. will be requested and added to the plot next to the computed values.

As mentioned earlier, all forces are normalized with freestream dynamic pressure only. CL, CD, CM are the usual chord-based definitions only if the airfoil has a unit chord -- in general, they will scale with the airfoil's chord. Also, CM is defined about the cartesian point (x,y) = (0.25,0.0), which is not necessarily the airfoil's 1/4 chord point.

-- Force calculation --

The lift and moment coefficients CL, CM, are calculated by direct surface pressure integration:

$$CL = L/q = \int Cp \; d\bar{x} \qquad\qquad CM = M/q = \int -Cp \; [(x-1/4) \; dx + y \; dy]$$

where $\bar{x} = x \cos(a) + y \sin(a)$ ; a = angle of attack

The integrals performed in the counterclockwise direction around the airfoil contour. The pressure coefficient Cp is calculated using the Karman-Tsien compressibility correction.

The drag coefficient CD is obtained by applying the Squire-Young formula at the last point in the wake --- NOT at the trailing edge.

$$CD = D/q = 2 \; Theta\_i = 2 \; Theta \; (u/V)^{(H+5)/2}$$

where Theta = momentum thickness |
u = edge velocity | at end of wake
H = shape parameter |

V = freestream velocity
Theta_i = momentum thickness at "downstream infinity"

The Squire-Young formula in effect extrapolates the momentum thickness to downstream infinity. It assumes that the wake behaves in a asymptotic manner downstream of the point of application. This assumption is strongly violated in the near-wake behind an airfoil with trailing edge separation, but is always reasonable some distance behind the airfoil. Hence, the usual application of Squire-Young at the trailing edge is questionable with separation present, but its application at the last wake point (typically 1 chord downstream) is always reasonable. Also, application at the last wake point also results in the formula having a smaller effect in any case, since there u ~ V, and hence Theta_i ~ Theta.

In most 2-D airfoil experiments, drag is measured indirectly by measuring 2 Theta/c in the wake, often within one chord of the airfoil's trailing edge. For consistency, this should be compared to the Theta value predicted by XFOIL at the same wake location, rather than the "true" Cd = 2 Theta_i/c value which is effectively at downstream infinity. In general, Theta_i will be smaller than Theta. In most airfoil drag measurement experiments, this difference may amount to the drag measurement being several percent too large, unless some correction is performed.

In addition to calculating the total viscous CD from the wake momentum thickness, XFOIL also determines the friction and pressure drag components CDf,CDp of this total CD. These are calculated by

$$CDf = \int Cf \, d\bar{x} \qquad\qquad CDp = CD - CDf$$

Here, Cf is the skin friction coefficient defined with the freestream dynamic pressure, not the BL edge dynamic pressure commonly used in BL theory. Note that CDp is deduced from CD and CDf instead of being calculated via surface pressure integration. This conventional definition

$$CDp = \int Cp \, d\bar{y} \quad ; \quad \bar{y} = y \cos(a) - x \sin(a) \ , \quad \bar{x} = \text{as above}$$

is NOT used, since it is typically swamped by numerical noise.

## -- Transition criterion --

Transition in an XFOIL solution is triggered by one of two ways:

```
  free  transition:  e^n criterion is met
 forced transition:  a trip or the trailing edge is encountered
```

The e^n method is always active, and free transition can occur upstream of the trip. The e^n method has the user-specified parameter "Ncrit", which is the log of the amplification factor of the most-amplified frequency which triggers transition. A suitable value of this parameter depends on the ambient disturbance level in which the airfoil operates, and mimics the effect of such disturbances on transition.  Below are typical values of Ncrit for various situations.

| situation | Ncrit | |
|---|---|---|
| sailplane | 12-14 | |
| motorglider | 11-13 | |
| clean wind tunnel | 10-12 | |
| average wind tunnel | 9 | <= standard "e^9 method" |
| dirty wind tunnel | 4-8 | |

Note: The e^n method in XFOIL is actually the simplified envelope version, which is the same as the full e^n method only for flows with constant H(x). If H is not constant, the two methods differ somewhat, but this difference is typically within the uncertainty in choosing Ncrit.

The e^n method is only appropriate for predicting transition in  situations where the growth of 2-D Tollmien-Schlichting waves via linear instability is the dominant transition-initiating mechanism.   Fortunately, this happens to be the case in a vast majority of airfoil applications.  Other possible mechanisms are:

* Crossflow instabilities. These occur on swept wings with significant favorable chordwise pressure gradients.

* Attachment-line transition.  This requires large sweep, large LE radius, and a large Reynolds number.  Occurs primarily on big jets.

* Bypass transition.  This occurs in cases with sufficient wall roughness and/or large freestream turbulence or vibration levels. The linear-instability phase predicted by the e^n method is  "bypassed", giving relatively early transition.  Usually occurs in favorable pressure gradients, while the linear-instability mechanism usually dominates in adverse pressure gradients.

If any of these alternative transition mechanisms are present, the trips must be set to mimick their effect.  The bypass transition mechanism can be mimicked to some extent by the e^n method by  setting Ncrit to a small value --- Ncrit=1 or less.  This will  cause transition just after linear instability begins.  For very large freestream turbulence or roughness in favorable pressure gradients, bypass transition can occur before the  linear instability threshold, and in this case trips will have  to be set as well.

### -- Numerical accuracy --

If strong separation bubbles are present in a viscous solution, then it is very important to have good resolution in the region of the bubble(s).  The large gradients at a bubble tend to cause significant numerical errors even if a large number of panels is used. If a separation bubble appears to be poorly resolved, it is a good idea to re-panel the  airfoil with more points, and/or with points bunched around the bubble  region.  The paneling is controlled from the PPAR menu. A good rule of  thumb is that the shape parameter Hk just after transition in the bubble  should not decrease by more than 1.0 per point.

Likewise, the surface velocity Ue/Vinf should not change by more than 0.05 per point past transition, otherwise there may be significant numerical errors in the drag. The point values can be observed by issuing SYMB from the VPLO menu. Moderate chord Reynolds numbers (1-3 million, say) usually require the finest paneling, since the bubbles are still important, but very small. On many airfoils, especially those with small leading edge radii, the development of the small bubble which froms just behind the leading edge can have a significant effect on CLmax. For such cases, the default paneling density at the bubble may not be adequate. In all cases, inadequate bubble resolution results in a "ragged" or "scalloped" CL vs CD drag polar curve, so fortunately this is easy to spot.

The BL equations are normally discretized with two-point central differencing (i.e. the Trapezoidal Scheme), which is second-order accurate, but only marginally stable. In particular, it has problems with the relatively stiff shape parameter and lag equations at transition, where at high Reynolds number the shape parameter must change very rapidly. Oscillations and overshoots in the shape parameter will occur with the Trapezoidal Scheme if the grid cannot resolve this rapid change. To avoid this nasty behavior, upwinding must be introduced, resulting in the Backward Euler Scheme, which is very stable, but has only first-order accuracy. Previous versions of XFOIL allowed a specific constant amount of upwinding to be user-specified. Currently, XFOIL automatically introduces upwinding into the equations only in regions of rapid change (typically transition). This ensures that the overall scheme is stable and as accurate as possible.

### -- Viscous solution acceleration --

The execution of a viscous case requires the solution of a large linear system every Newton iteration. The coefficient matrix of this system is 1/3 full, although most of its entries are very small. Substantial savings in CPU time (factor of 4 or more) result when these small entries are neglected. SUBROUTINE BLSOLV which solves the large Newton system ignores any off-diagonal element whose magnitude is smaller than the variable VACCEL, which is initialized in SUBROUTINE INIT, and which can be changed at runtime from the VPAR menu. This should in principle affect the convergence rate of the viscous solution and thus result in more Newton iterations, although the effect is usually too small to notice. For very low Reynolds number cases (less than 100000), it MAY adversely affect the convergence rate or stability, and one should try reducing VACCEL or even setting it to zero if all other efforts at convergence are unsuccessful. The value of VACCEL has absolutely no effect on the final converged viscous solution (if attained).

### -- Polar calculations --

The polar calculation facility driven from the OPER menu deserves a detailed description. Anytime an operating point is calculated, the results can be saved in either a small "save file" containing alpha, CL, CD, etc., and/or an unformatted "dump file" containing in addition the surface Cp, Theta, Dstar, etc. distributions. A collection of these points written to the save and dump files then constitutes a polar. The simplest way to create one is to first issue the PACC command which sets the auto polar accumulation toggle and asks for the save and dump filenames, and then calculate the operating points with ASEQ or CSEQ. The points will be written automatically as they are calculated. Alternatively, each point can be calculated individually with ALFA or CLI or CL, and written out with PADD.

When the PACC or PADD command is first issued, the save and dump filenames will be requested. If these files already exist in the current directory AND represent the same airfoil and freestream conditions, then subsequent points will be appended to them. If the airfoil and conditions are different, then save and dump file writing will not be enabled so that the old polar isn't clobbered. Messages are always produced informing the user of what's going on and the general status of things.

Polar save file(s) can be plotted with the separate program PPLOT. It also permits reference data to be overlaid if it is available. A reference data file has the following form:

```
CD(1)  CL(1)
CD(2)  CL(2)
 .      .
 .      .
999.0  999.0
alpha(1)  CL(1)
alpha(2)  CL(2)
  .       .
  .       .
999.0   999.0
alpha(1)  Cm(1)
alpha(2)  Cm(2)
  .       .
  .       .
999.0   999.0
Xtr/c(1)  CL(1)
Xtr/c(2)  CL(2)
  .        .
  .        .
999.0   999.0
```

The number of points in each set (CD-CL, alpha-CL, etc.) is arbitrary, and can be zero. The file pplot.def contains plotting parameters read in by PPLOT.

The contents of a polar dump file can be selectively plotted with the separate menu-driven program PXPLOT. It is executed with:

```
% pxplot <dump filename>
```

This allows surface plots of Cp vs x, H vs x, etc. for any or all of the saved operating points. Of course, these plots can be generated in XFOIL for any individual operating point, so PXPLOT and the dump file itself are somewhat redundant in this respect.

## -- Re, Mach dependence --

A few comments are in order on the TYPE command, which allows the user to set the dependence of the Mach and Reynolds numbers on CL. Any CL-CD polar can be of the following three types:

```
Type   parameters held constant       varying      fixed
----   ------------------------       -------   -----------
  1    M          , Re            ..   lift     chord, vel.
  2    M sqrt(CL) , Re sqrt(CL)   ..   vel.     chord, lift
  3    M          , Re CL         ..   chord    lift , vel.
```

* Type 1 corresponds to a given wing at a fixed velocity going over an angle of attack range, as in a wind tunnel test alpha sweep or a sudden aircraft pullup. This is also the common form for an airfoil polar.

* Type 2 corresponds to an aircraft in level flight at a given altitude undergoing trim speed changes. This is the most useful airfoil polar form for determining a drag polar for an aircraft at 1-g. For this case, The "Mach number" input with the MACH command is actually interpreted as the product M sqrt(CL), and the "Reynolds number" input with the VISC or RE commands is actually interpreted as RE sqrt(CL). For a wing in level flight, these products can be computed from the following exact relations, with Re based on the mean chord:

```
                        1/2                                  1/2
             |  2  W/S |                     1  | 2 rho W |
M sqrt(CL) = | --- --- |       RE sqrt(CL) = -- | ------- |
             | 1.4  p  |                     mu |    AR   |

W  = weight         p   = ambient pressure
S  = wing area      mu  = dynamic viscosity
AR = aspect ratio   rho = ambient air density
```

* Type 3 corresponds to a wing of "rubber chord" with a given lift at a given speed. This is best used for selecting an optimum CL for an airfoil while taking Reynolds number changes into account. The product RE CL can be computed from the following:

```
           2 W
 RE CL = ------       b = span = sqrt(S*AR)
         mu V b
```

Caution must be used with Types 2 and 3 so as to not allow the CL to go negative. In addition, with non-zero Mach and Type 2, the CL must not fall below that value which makes Mach exceed unity. Warning messages are printed when these problems occur.

## Output
------

All output goes directly to the terminal screen. The MIT plot package libPlt.a (based on the ancient Versaplot package) used by XFOIL is aimed at X-Windows graphics, although it supports many plotting terminals (including Tektronix emulators), and generates vector files for PostScript laser printers. The variable IDEV (initialized in SUBROUTINE INIT) can take on values between 1 and 12, and tells libPlt which type of graphics terminal is being used. File libPlt.doc contains a list which gives the IDEV values associated with various terminal types. If a particular IDEV value "almost" works, it may be a good idea to sift through the other device codes by trial and error. If nothing works, then custom plotting routines will have to be written. XFOIL uses only a few low-level routines, so this shouldn't be too painful. File libPlt.doc contains the descriptions of these routines.

It should be possible to link XFOIL with a standard Versaplot package. The only non-standard routine included in libPlt.a is SUBROUTINE WHEREC, which simply returns the plot coordinates of the screen plot cursor. It is crucial to the interactive design facilities of XFOIL, so some equivalent version will have to be written if libPlt is not used.

The standard libPlt should compile with no trouble on DEC and Iris workstations, and reportedly also works on Sun workstations. Versions for the RS/6000 and HP-9000 are also provided. The libPlt source code for all these machines is nearly the same. The main differences are actually in the makefile syntax, and in the manner in which C and Fortran code is linked.

A VAX version of libPlt which drives VAX-UIS and also Tektronix-type graphics is also available. This version makes VAX-VMS system calls, and cannot be used with other systems.

## Plot Hardcopy
-------------

For hardcopy, every screen plot will be echoed to the PostScript file plot.ps whenever the hardcopy flag is set (toggle with HARD command). The size of the hardcopy plot can be changed with the SIZE command from most menus. The number requested is the width of the plot in inches. This file can then be printed after the XFOIL session is completed.

*** NOTE ***
XFOIL must be exited with QUIT in order for the hardcopy file to be properly terminated and closed. Printing plot.ps before exiting XFOIL, or after exiting with ^C, is not likely to work.

*** NOTE ***
For the geometry plot in GDES, and the Qspec(s) plots in QDES and MDES (described below), the hardcopy plot width will also be affected if the graphics window is resized with the cursor at the window manager level. This is because the plot is always scaled so that it fills up as much of the window as possible. If the window size is left at its start-up size, the hardcopy plot widths will come out with the specified size in inches.

## Surface Speed Design Routines

XFOIL's Full-Inverse complex-mapping facility (MDES) takes as input a speed distribution "Qspec" specified over the entire airfoil surface, modifies it somewhat to satisfy the Lighthill constraints, and generates a new overall geometry. It is intended as a redesign method in which the surface speed distribution of an existing airfoil is used as a starting point to generate a new speed distribution. A ``pure" design code which requests the entire surface speed distribution every time is often less natural to use, since airfoil design is invariably an iterative process involving repeated analyze/fix cycles. The MDES menu is shown below.

```
   <cr>   Return to Top Level
   INIT   Re-initialize mapping
   QSET   Reset Qspec <== Gamma
   AQ     Show/select alpha(s) for Qspec
   CQ     Show/select  CL(s)   for Qspec
   SYMM   Toggle symmetry flag
   TGAP   Set new TE gap
   TANG   Set new TE angle
   VISC   Qvis overlay toggle
   REFL   Reflected Qspec overlay toggle
   MODI   Modify Qspec
   MARK   Mark off target segment for smoothing
   SMOO   Smooth Qspec inside target segment
   FILT   Change inverse-mapping filter level
   SLOP   Toggle modified-Qspec slope matching flag
   PLOT   Plot Qspec (line) and Gamma (symbols)
   SPEC   Plot mapping coefficient spectrum
   HARD   Hardcopy enable/disable toggle
   BLOW   Blowup
   RESE   Reset to original scale
   SIZE   Change absolute plot size
   EXEC   Execute  full-inverse calculation
```

The initial Qspec distribution is taken from "Gamma", the speed distribution corresponding to the current geometry at the last angle of attack employed in OPER. Qspec can be set back to this Gamma with QSET anytime.

Qspec can be modified to whatever is desired with the MODI command by specifying points with the screen cursor which are then splined. The points can be entered in any order, and any pair of identical points is ignored, so that a point can be "erased" by clicking on it again anytime later (this will work only if the terminal driver rounds off all cursor inputs to the nearest pixel). The input sequence is terminated with three identical points.

Normally, the modified piece of Qspec(s) is blended into the current Qspec(s) with matching slopes at the piece endpoints. This slope matching can be turned on/off with the SLOP toggle command. Qspec can also be smoothed with the SMOO command, which normally operates on the entire distribution, but can be confined to a target segment whose endpoints are selected with the MARK command. The smoothing acts to alleviate second derivatives in Qspec(s), so that with many consecutive SMOO commands Qspec(s) will approach a straight line over the target segment. If the slope-matching flag is set, the endpoint slopes are preserved. The BLOW command can be used to enlarge regions of interest at any time by specifying opposite corners of the blowup region.

The FILT command is an alternative smoothing procedure which acts on the Fourier coefficients of Qspec directly, and is global in its effect. It is useful for "cleaning up" the entire Qspec(s) distribution if noise is present from some geometric glitch on the airfoil surface. Also, unintended noise might be introduced into Qspec from a poor modification via the cursor.

FILT acts by multiplying the Fourier coefficients by a Hanning window filter function raised to the power of a filter parameter "F". This tapers off the high frequencies of Qspec to varying degrees. A value of F = 0.0 gives no filtering, F = 1.0 gives the standard Hanning filter, F = 2.0 applies the Hanning filter twice, etc. The standard Hanning filter appears to be a bit too drastic, so a filter parameter of F = 0.2 is currently used. Hence, issuing FILT five times corresponds to the standard Hanning filter. The SPEC command displays the mapping

The symmetry-forcing option (SYMM toggle) is useful when a symmetric airfoil is being designed. If active, this option zeroes out all antisymmetric (camber) Qspec changes, and doubles all symmetric (thickness) changes. This unfortunately has the annoying side effect of also doubling the numerical roundoff noise in Qspec every time a MODI operation is performed. This noise sooner or later becomes visible as high-frequency wiggles which double with each MODI command. Issuing FILT occasionally keeps this parasitic noise growth under control.

The MODI, BLOW, MARK, SMOO, SLOP, FILT commands can be issued repeatedly in any order until Qspec is modified to have the desired distribution. Once Qspec has the desired shape, EXEC generates the corresponding new buffer airfoil. In general, the resulting surface speed distributions will not exactly match Qspec, since corrections are automatically added to maintain the specified trailing edge gap and to enforce consistency with the freestream speed. The trailing edge gap is initialized from the initial airfoil and can be changed with TGAP. To reduce the "corrupting" effect of the constraint-driven corrections, a good rule of thumb is that the Qspec distribution should be modified so as to preserve the total CL. The CL is simply twice the area under the Qspec(s) curve (= 2 x circulation), so that this area should be preserved.

MDES normally starts with only one Qspec distribution (same as Gamma) as the default. A very useful feature of the MDES facility is the ability to display and modify a number of Qspec distributions corresponding to different alpha or inviscid CL values. These values are displayed and/or selected via the AQ or CQ commands. When any one Qspec distribution is modified, the result of modification is also displayed on all the other distributions. This allows rapid design at multiple operating points. When the Qspec curves correspond to specified CL values, the alpha for each curve will be adjusted after each Qspec modification so as to preserve that curve's CL. The resulting Qspec will therefore not match the input cursor points exactly because of this alpha correction.

The EXEC command generates a new buffer airfoil corresponding to the current Qspec distribution. If subsequent operations on this airfoil are to be performed (SAVE, OPER, etc.), it is necessary to first generate a current airfoil from this buffer airfoil using PANE at the top level menu.  This seemingly  complicated sequence is necessary because the airfoil points  generated by EXEC are uniformly spaced in the circle plane, which gives a rather poor point (panel node) spacing distribution on the physical airfoil.  This sequence also prevents the current airfoil from being overwritten immediately when EXEC is issued.   Once the new current airfoil is generated with PANE, it can  then be analyzed in OPER, modified in GDES, or whatever.

The Full-Inverse facility is very fast, after an initialization  calculation of several seconds (on a RISC workstation), it requires  only a fraction of a second to generate the new buffer airfoil.

* * * * *

XFOIL's Mixed-Inverse facility (QDES) is useful in certain redesign  problems where parts of the airfoil cannot be altered under any  circumstances.  The Mixed-Inverse menu is shown below.

```
     <cr>   Return to top level
     QSET   Reset Qspec <== Gamma
     VISC   Qvis overlay toggle
     REFL   Reflected Qspec overlay toggle
     MARK   Mark off target segment
     MODI   Modify Qspec
     SMOO   Smooth Qspec inside target segment
     SLOP   Toggle modified-Qspec slope matching flag
     PLOT   Plot Qspec (line) and Gamma (symbols)
     HARD   Hardcopy enable/disable toggle
     BLOW   Blowup
     RESE   Reset to original size
     SIZE   Change absolute plot size
     CPXX   CPxx endpoint constraint toggle
     EXEC   Execute mixed-inverse calculation
     REST   Restore geometry from buffer airfoil
```

The QDES menu above is intentionally geared for the redesign of a segment of an existing airfoil (with its surface speed distribution calculated  previously in OPER) rather than the generation of a totally new airfoil.  When QDES is entered, the specified speed distribution Qspec is  initialized to the current speed distribution Gamma last set in OPER. If a direct solution for the current airfoil hasn't been calculated yet,  QDES goes ahead and calculates it, using the last-set angle of attack. If this isn't the desired angle, it can be set in OPER using ALFA. QSET can then be used to set Qspec from the current Gamma distribution.

Qspec can be repeatedly modified with the screen cursor and the MODI command, exactly as in Full-Inverse.  It is also necessary to mark off the target segment where the geometry is to be modified with the MARK command.  EXEC then modifies the airfoil over the target segment to match Qspec there as closely as possible.  The remainder of the  airfoil geometry is left unaltered.  EXEC requests the number of Newton iterations to be performed in the inverse calculation.  Although as many as six iterations may be required for convergence to machine zero, it is _not_ necessary to fully converge a Mixed-Inverse case.  Two  iterations are usually sufficient to get very close to the new geometry.  In any case, the new surface speed distribution Gamma  which actually results from the inverse calculation will typically differ somewhat from the specified distribution Qspec by function modes which are added to Qspec. At least two modes are added, with their magnitudes determined by geometric closure requirements at the inverse segment endpoints. As with the MDES complex-mapping routine, the necessary modifications to Qspec will be smallest if Qspec is modified so that CL (the area under the Qspec(s) curve) is roughly preserved.

Issuing PLOT after the EXEC command finishes will compare the specified (Qspec) and resulting (Gamma) speed distributions. If extra smoothness in the surface speed is required, the CPXX command just before EXEC will enable the addition of two additional modes which allow the second derivative in the pressure at the endpoints to be unchanged from the starting airfoil. The disadvantage of this option is that the resulting surface speed Gamma will now deviate more from the specified speed Qspec. It is allowable to repeatedly modify Qspec, set or reset the CPXX option, and issue the EXEC command in any order.

The Mixed-Inverse modification is performed on the current airfoil directly, in contrast to Full-Inverse which generates the buffer airfoil as its output. In fact, it is important _not_ to issue the PANE command at top level after doing work in the QDES menu, as the new current airfoil will be overwritten with the old buffer airfoil.

**Geometry Design Routine**

Executing the GDES command from the top level menu will put the user into the GDES routine. Its menu is

```
     <cr>   Return to Top Level
     GSET   Set buffer  airfoil <== current airfoil
     EXEC   Set current airfoil <== buffer  airfoil
     SYMM   Toggle y-symmetry flag
     ADEG   Change absolute angle of attack (degrees)
     ARAD   Change absolute angle of attack (radians)
     TRAN   Translate
     SCAL   Scale about origin
     TGAP   Change trailing edge gap
     LERA   Change leading edge radius
     CANG   Display panel node corner angles
     TCAM   Find or change thickness, camber
     CAMP   Add to camber line from added delta(Cp)
     CAMS   Add to camber line from splined input points
     FLAP   Deflect trailing edge flap
     MODI   Modify contour via cursor
     SLOP   Toggle modified-contour slope matching flag
     CORN   Double point with cursor (set corner)
     ADDP   Add point with cursor
     DELP   Delete point with cursor
     MOVP   Move point with cursor
     UNIT   Normalize buffer airfoil to unit chord
     DIST   Determine distance between 2 cursor points
     PLOT   Plot buffer airfoil
     BLOW   Blowup plot
     RESE   Reset to original plot size
     SIZE   Change absolute plot size
     OVER   Overlay disk file airfoil
    .ANNO   Annotate plot
```

The first command typically executed is GSET, which sets the temporary buffer airfoil from the current airfoil. Sometimes it might be desired to operate directly on the coordinates of an already existing buffer airfoil. It typically contains coordinates read in from a disk file by LOAD at Top Level, or coordinates generated by EXEC from the MDES menu, depending on what was done last. In either of these cases, GSET is skipped. Most of the above commands can then be used to modify this buffer airfoil, and the results displayed using PLOT. Sometimes a sequence of commands is necessary to achieve the desired effect. For instance, suppose an airfoil with the current thickness envelope is to be given an entirely new camber line. Issuing TCAM and inputting a new thickness of 999 (keep same thickness) and a new camber of 0 will result in the current thickness envelope unchanged and the current camber eliminated, so that a symmetrical airfoil remains. The new camber line can then be added with either CAMS or CAMP. CAMS takes the new camber line as a sequence of x/c,y/c coordinate pairs which are splined. CAMP takes a sequence of x/c,delta(Cp) pairs instead. This delta(Cp) (i.e. loading) distribution, defined as

delta(Cp) = (Cp)lower - (Cp)upper

is then used in Glauert's thin-airfoil relations to define the x/c,y/c camber line.

With CAMS and CAMP, a slope discontinuity in y(x) or Cp(x) can be specified with two identical consecutive x/c values, which prevents splining across this point. CAMP can thus easily generate a camber line with a piecewise-linear delta(Cp) loading distribution, as for example an a=0.8 NACA 6-digit airfoil:

| x/c | delta(Cp) |
|---|---|
| 0.0 | 0.5 |
| 0.8 | 0.5 |
| 0.8 | 0.5 |
| 1.0 | 0.0 |

This results in a constant delta(Cp) = 0.5 for 0.0 < x/c < 0.8 ,<br>
then decreasing linearly to delta(Cp) = 0.0 for 0.8 < x/c < 1.0

The various commands for modifying camber-line, thickness, leading edge radius, etc. should suffice for most geometry modification tasks. If truly frustrated, the user can "draw" the new contour with the MODI command, which accepts cursor inputs in the same manner as the MDES and QDES procedures. Slope matching at the modified-piece endpoints can likewise be enabled/disabled with the SLOP toggle command. The only important difference is that here the points must be entered in consecutive order along the new contour, although one can "erase" a previously-entered point by clicking over it at any time.

A point can be doubled with the CORN command. A doubled point is useful wherever a sharp corner is required, such as at a flap break. Normally, the spline routine enforces slope continuity at all points, effectively preventing sharp corners. A doubled point, marked by a small diamond symbol on the plot, causes separate splines to be generated on each side of the corner, thus allowing the slope break. The doubled point is eliminated by clicking on it after issuing the DELE command. Using DELE on a normal single point will delete that point entirely.

Once the desired buffer airfoil is created, a new current airfoil is set directly from the buffer airfoil with the EXEC command. Alternatively, the new current airfoil can be generated from the buffer airfoil with the PANE command at top level.  The new current airfoil can then be  analyzed in OPER.  If the buffer airfoil has any doubled corner points, the doubled points will be eliminated, but a current-airfoil node will  fall exactly on each buffer-airfoil corner.

The TGAP command sets the thickness (or gap) of the blunt trailing edge.The gap "ds" is defined as the distance between the upper and lower coordinate endpoints:  ds^2 = dx^2 + dy^2.  If the gap is already nonzero, then the new TE base vector (dx,dy) will have the same orientation as the old one, i.e.

```
  dy|       dy|
  --|    =  --|
  dx|       dx|
     new       old
```

If the gap is zero to begin with, then the new base vector will be perpendicular to the trailing edge bisector.  If the base orientation comes out in an unexpected way, it probably means that the gap was not *exactly* zero.  The fix is to first set the gap to zero, and then set it again to the desired value.

After a new gap size is input, a "blending distance/c" will also be requested.  This controls how rapidly the new TE blends into the original airfoil, and is essentially the length scale for the blending  function, which is exponential in x/c.  The limiting values are:

```
distance/c = 0:  Only the upper- and lower-surface TE points are changed
           = 1:  A linear "wedge" is added or subtracted from the airfoil
```

**Start-up Defaults**

XFOIL has hardwired parameters (in subr. INIT) controlling the paneling, plotting, and viscous execution.  Most of these can be changed at runtime in the various menus.  To avoid the need to change the parameters everytime XFOIL is executed, they can be saved to the default file  xfoil.def  with the WDEF  command at TOP LEVEL.  This file has the format:

```
120       1.0000   0.3000   0.2000 | Npan    PPanel  TErat  REFrat
    1.0000   1.0000   1.0000   1.0000 | XrefS1  XrefS2  XrefP1 XrefP2
   10.0000   0.7000   0.0180          | Size    PlotAR  CHsize
    1.0000  -2.0000  -0.5000          | CPmax   CPmin   CPdel
    0.1200   0.7000                   | XoffAir ScalAir
   1         0.0000   0.0500          | MAtype  Mach    Vaccel
   1         0.0000   9.0000          | REtype  Re/10^6 Ncrit
    1.0000   1.0000                   | XtripT  XtripB
```

Line 1: Paneling parameters from the PPAR menu.
Line 2: Paneling refinement locations
Line 3: Specifies the absolute plot size, the plot aspect ratio,
and scales the character, number, and symbol heights.
Line 4: Defines the Cp axis annotations.
Line 5: x-offset and scale factor for airfoil on Cp vs x plot
Line 6: Mach-CL dependence type, Mach number, solution acceleration param.
Line 7: Re-CL dependence type, Reynolds number, transition parameter
Line 8: Forced transition x/c locations on top,bottom sides

This file will be read at any time with the RDEF command, thus avoiding the manual entry of all the information.

**Cavea**ts

The XFOIL code is not foolproof, and requires some level of aerodynamic expertise and common sense on the part of the user. Although the inviscid analysis (OPERi), geometry design (GDES), and Full-Inverse (MDES) routines are nearly invulnerable to failure, the Mixed-Inverse (QDES) design routines and especially the viscous analysis (OPERv) routines will fail if a "reasonable" problem is not specified. Typical failure scenarios are:

- Viscous Analysis (OPERv)
    - * Massive separation from excessive airfoil thickness, flap deflection, or angle of attack
    - * Poor resolution of leading edge pressure spike
    - * Poor resolution of small viscous features (e.g. separation bubbles)
    - * Reynolds number too low

- Mixed-Inverse Surface Speed Design (QDES)
    - * Re-entrant airfoil shape (negative thickness)

Although unlikely, a possible consequence of these occurences is an arithmetic fault causing program failure. It is therefore a good idea to save any previous work before an ambitious calculation is attempted.

When performing viscous analysis calculations, it is always a good idea to sequence runs so that alpha does not change too drastically from one case to another. The Newton solution method always uses the last available solution as a starting guess for a new solution, and works best if the change from the old to the new solutions is reasonably small. For this reason, it is best to perform difficult calculations (such as past CLmax) by gradually increasing alpha. The ASEQ command in OPER is convenient for this. If the user insists on a large change from one point to another, it is best to force a re-initialization of the boundary layers with the INIT command from the VPAR menu in OPER before the radical calculation is performed. INIT should always be executed whenever the viscous solution blows up but the program doesn't crash.

The viscous analysis will execute at most ~10 Newton iterations each time an ALFA, CL, etc. command is issued. If convergence is not achieved within this limit, ALFA or CL can be issued as often as needed, with another ~10 Newton iterations being performed each time. This iteration limit is passed as a parameter to SUBROUTINE VISCAL, e.g.

```
    CALL VISCAL(10)
```

The call statements occur in several places in SUBROUTINE OPER (in xoper.f), and can be easily changed if deemed appropriate.

One should always be wary of trusting solutions which show regions of supersonic flow. Such flows can be reliably predicted only with a truly nonlinear field method (such as the ISES code). As a rule of thumb, if the maximum Mach number doesn't exceed 1.05 anywhere, shock losses will be very small, the Cp distributions will be reasonably accurate, and the drag predicted by XFOIL is likely to be accurate.

### VAX VMS Batch Execution

It is often inconvenient to run a long XFOIL calculation such as a large polar interactively on a microVAX.  A simple way to perform the calculation in batch mode is to type the appropriate XFOIL commands into a file, and specify it as the input file for a batch request.  For example, if one wants to calculate a polar for the  Eppler 387 airfoil whose coordinates are in file E387.DAT, a suitable command sequence (comments optional) would be:

```
LOAD          load airfoil
E387.DAT      coordinate filename
EPPLER 387    airfoil name
PPAR          paneling parameter change menu
P             request new panel bunching parameter
1.4           input panel bunching parameter
T             request new TE bunching parameter
.2            input TE bunching parameter
              XFOIL requests a carriage return here
              <cr> to get back to top level
OPER          get into OPER menu
VISC          request viscous option
200000.       input Reynolds number
PACC          request automatic polar writing to disk
E387_09.200   polar save filename  [Ncrit=9 (default), Re=200 thousand]
XE387_09.200  polar dump filename
ASEQ          request alpha sequence
3.0           start alpha
9.0           end alpha
0.5           alpha increment
VPAR          get into VPAR menu
INIT          force BL re-initialization (since we'll specify a big
alpha jump)
              get back to OPER menu
ASEQ          request alpha sequence
2.5           start alpha
-3.0          end alpha
-.5           alpha increment
              XFOIL requests a carriage return here
              <cr> to get back to top level
QUIT          terminate program
```

The blank lines are significant as commented above (the Ultrix version of XFOIL does not require them, however).  Note that the polar is calculated in two pieces, always starting from an intermediate alpha which will very likely converge.  Starting from CLmax or CLmin to perform one big sweep risks convergence failure of the starting point and possibly a failure of the entire sweep if the BL blows up badly enough.

Although placing of the above commands into a file requires knowledge of what prompts XFOIL will put out, this is easily determined  interactively.  Once the user is familiar with the program, a  "dry run" is rarely necessary.

After the command file (E387.RUN, say) above is created, it is submitted with the VMS command

```
$ @[USER.XFOIL]XBATCH E387.RUN
```

Where the command file XBATCH.COM contains the following:

```
$ CURDIR := 'F$DIRECTORY()'
$ SUBMIT/NOPRINT/NOTIFY/NAME=XFOIL/PARAM=('P1','CURDIR')
[USER.XFOIL]BXFOIL
```

And the command file BXFOIL.COM contains the following:

```
$ SET DEFAULT 'P2'
$ ASSIGN/NOLOG 'P1' FOR005
$ ASSIGN/NOLOG SYS$OUTPUT FOR006
$ RUN [USER.XFOIL]XFOIL
```

These command files assume that XBATCH.COM, BXFOIL.COM, and XFOIL.EXE all reside in directory [USER.XFOIL]. The output XFOIL normally dumps to the screen will go to file XFOIL.LOG in the user's home directory, and the polar save file E387_09.200 and dump file XE387_09.200 will appear in the directory from which the batch job was submitted, and can be plotted by executing PPLOT and PXPLOT.

**UNIX Background Execution**

In UNIX, a "batch" XFOIL run can be executed simply with

```
% xfoil < e387.run > e387.log &
```

which will read the terminal commands from file e387.run as in the VMS example above (the extra blank lines must be omitted, though), and the screen text output will be sent to file e387.log. Any polar files generated by the commands in e387.run will be created separately.

On RISC machines, a polar calculation is fast enough so that batch execution will rarely be necessary.