Appendix B: AVL Tutorial

AVL has a pretty good user manual, but it is long and has few pictures. In this tutorial we will cover the basics needed to understand the input files and how to use AVL to do the most common tasks with multiple pictures. After this tutorial the reader should be able to:

- Understand the input file format and be able to create and modify new ones
- Know how to run different flight conditions
- Be able to evaluate trim conditions
- Understand how to use the lift distribution to evaluate max lift coefficient
- Estimate hingemoments for a control surface

The first thing that we will cover is the input file format. The format is not unique; we will cover one example, but there are many possible variations. The first line of the input file is the configuration name. For the example:

```
DBF 08-09 Configuration 11-13-08
```

The next line defines the Mach number, in this case zero. Note that the pound sign (#) denotes a comment. The AVL calculations are corrected using the Prandtl-Glauert compressibility factor for non-zero Mach numbers. This approximation is only valid for Mach numbers of up to about 0.6-0.8, with the higher sweep angles enabling higher Mach numbers. For our purposes zero is enough.

```
#Mach
 0.0
```

The next set of inputs defines any planes of symmetry. If any asymmetric flight condition will ever be used (such as nonzero sideslip) then no planes of symmetry should be used. The example below is for no planes of symmetry. Note that we will define the airplane to be symmetric, but the flight condition will not necessarily be.

```
#IYsym   IZsym   Zsym
 0       0       0.0
```

All of the reference parameters are defined next. The units used here, and everywhere else in the input file, are completely arbitrary. It could be feet, inches, meters or even leagues. All that is important is that the units be consistent. The area, chord and span reference lengths are defined first. They are followed by the reference point. This point should be the center of gravity of the configuration, or else the trim solutions presented later will be invalid.

```
#Sref     Cref     Bref
1233.8    11.864   104.0
#Xref     Yref     Zref
11.0      0.0      0.0
```

Now we get to meat of the input file, where we define the planform, the airfoil and the control surfaces. The first line in this case lets AVL know that the following will be part of the same surface and gives that surface a name.

```
#==============================================
SURFACE
Wing
```

Now we define how many chordwise and spanwise sections the surface will be divided into. In this case we are using 10 chordwise sections. Typically no more than 10 chordwise sections should be used to minimize rounding error. The Cspace input defines how these will be distributed. The 1.0 denotes a cosine distribution; this should be used unless the user has some specific reason not to, as it tends to be the most computationally efficient technique. The spanwise parameters are left blank because they will be defined differently for the different sections of the wing. They can be defined here, in which case they will distribute the spanwise cuts without regard to any planform discontinuities, but it is recommended to use the individualized method for better control of the discretization.

```
#Nchordwise  Cspace   Nspanwise   Sspace
10           1.0
```

The next two parameters are used to tell AVL to duplicate the geometry along the y-axis, and to rotate the whole surface a given angle.

```
YDUPLICATE
0.0
ANGLE
0.0
```

The wing (and any other lifting surface) is defined by the x,y,z coordinates of the leading edge of the planform, defined section-by-section. The chord then defines the length. The Ainc variable defines the local incidence angle, and is used to define the twist along the span. The Nspanwise is used here to define the number of spanwise sections over the course of the current section. The Sspace being equal to 1 defines the cosine distribution that is recommended for use. The AFILE then gives the airfoil coordinates filename. Alternatively, it is possible to use NACA and then a 4 or 5-series airfoil digits. The airfoil is used to define the camber line, where the panels will be distributed.

```
#--------------------------------------------------------------
SECTION
#Xle    Yle    Zle     Chord   Ainc  Nspanwise  Sspace
2.5     0.     0.      22.5    0.0   6          1
AFILE
sd7062.dat
```

The next section is defined below. There are still no control surfaces on this section. This and the previous section define one full section with no sweep and a constant 22.5 chord.

```
SECTION
2.5     4.5    0       22.5    0.0   10          1
AFILE
sd7062.dat
```

The next two sections define the sections with control surfaces. The CONTROL lets AVL know that control surface data is forthcoming. The first input defines the control surface name. The next (also the last) should be -1.0 for ailerons and rudders, and 1.0 for flaps and elevators. This input defines whether the control surface is deflected symmetrically or asymmetrically. The next input is equal to one minus the flap-to-chord ratio. The example is a 30% chord control surface. The next three inputs define the hinge line, in this case 0,0,0, or along the span. Note that it is possible to define more than one control surface over the same section. The example has a flaperon that is a combine aileron and flap. The flaperon deflection will then be equal to the sum of these two in practice, but AVL analyzes them separately.

```
SECTION
#Xle    Yle    Zle     Chord   Ainc  Nspanwise  Sspace
8.5     11.    0.      10.     0.0   20         1
CONTROL
aileron  -1.0  0.7   0. 0. 0.  -1.
CONTROL
flap  1.0  0.70  0. 0. 0.   1.
AFILE
sd7062.dat

SECTION
8.5     52.    0       10.     0.0   0          1
CONTROL
aileron  -1.0  0.7   0. 0. 0.  -1.
CONTROL
flap  1.0  0.70  0. 0. 0.   1.
AFILE
```

```
sd7062.dat
```

The next set of inputs defines the horizontal tail. There is only one section so that it is significantly simpler than the wing. No airfoil is specified, so the default is symmetric.

```
#==============================================
SURFACE
HTail
#Nchordwise  Cspace   Nspanwise   Sspace
10           1.0       10         1.0
YDUPLICATE
0.0
ANGLE
0.0
#---------------------------------------------------------------
SECTION
#Xle    Yle    Zle     Chord   Ainc  Nspanwise  Sspace
39.     0.     0.      7.      0.0   0          0
CONTROL
elevator  1.0  0.70  0. 0. 0.   1.
SECTION
#Xle    Yle    Zle     Chord   Ainc  Nspanwise  Sspace
39.     11.    0.      7.      0.0   0          0
CONTROL
elevator  1.0  0.70  0. 0. 0.   1.
```

The final surface is the vertical tail. Because the defined surface is not along the centerline (y=0) there are actually two vertical tails. The second is created as a duplicate surface.

```
#==============================================
SURFACE
VTail
#Nchordwise  Cspace   Nspanwise   Sspace
10           1.0       10         1.0
YDUPLICATE
0.0
ANGLE
0.0
#---------------------------------------------------------------
SECTION
#Xle    Yle    Zle     Chord   Ainc  Nspanwise  Sspace
39.     8.     0.      7.      0.0   0          0
CONTROL
rudder   -1.0  0.70  0. 0. 0.  -1.
SECTION
#Xle    Yle    Zle     Chord   Ainc  Nspanwise  Sspace
39.     8.     9.5     7.      0.0   0          0
CONTROL
rudder   -1.0  0.70  0. 0. 0.  -1.
```

The AVL interface is very similar to the XFOIL one. The opening screen is shown below. The period before an option indicates that it is a higher level menu. A "?" will bring up the menu options for any given menu. The load command reads in the input file defined above. The duplicate surfaces are created and some statistics are shown. In this case there are six surfaces because the duplicates are counted separately. The Nstrp is 112; this is the number of strips, which are the spanwise sections. The total number of sections is Nvor, 1408 in the example. This is close to the maximum that AVL will allow.

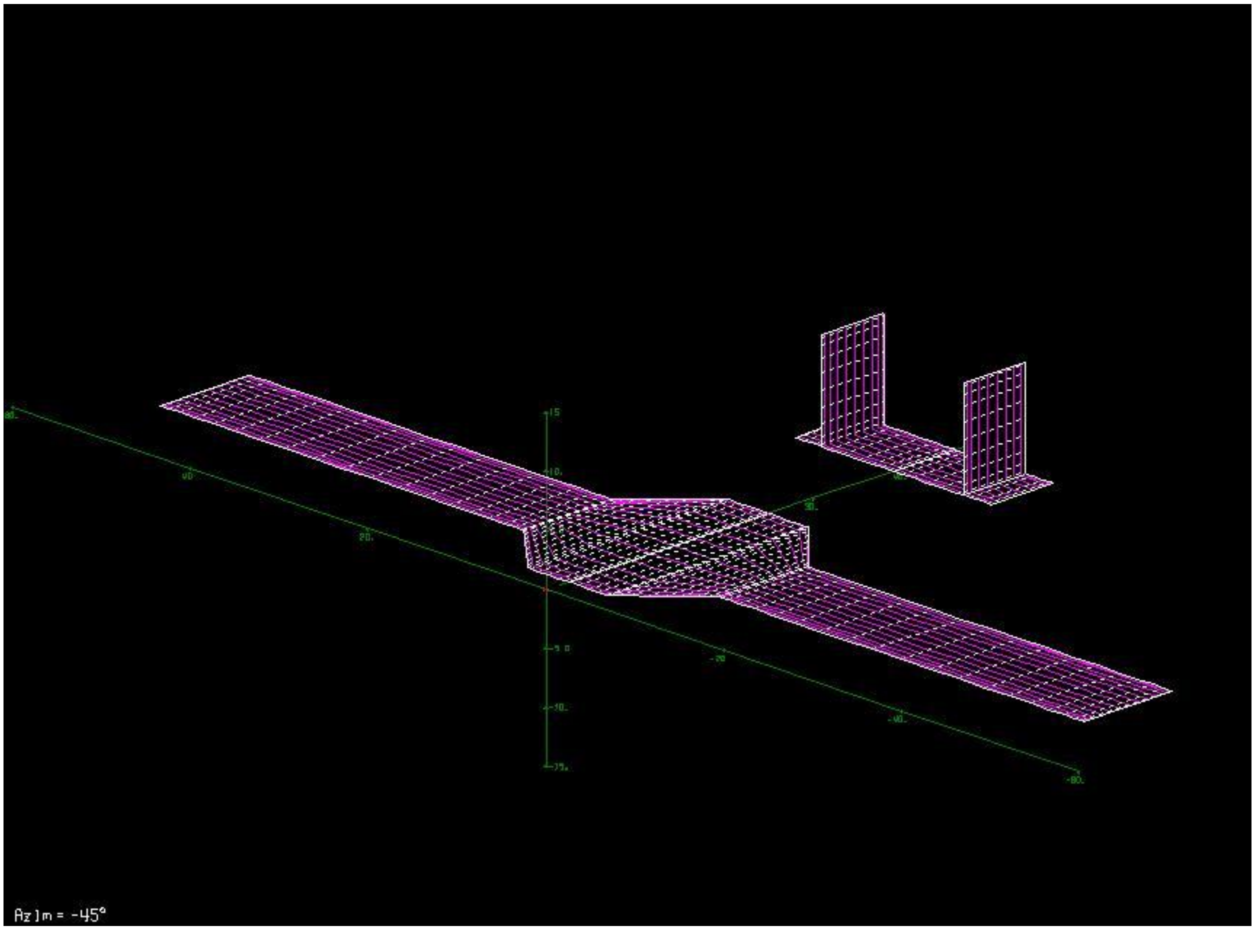
```
 This software comes with ABSOLUTELY NO WARRANTY,
   subject to the GNU General Public License.

 Caveat computor
 ==================================================

 ==========================================================
   Quit    Exit program

  .OPER    Compute operating-point run cases
  .MODE    Eigenvalue analysis of run cases
  .TIME    Time-domain calculations

   LOAD f  Read configuration input file
   MASS f  Read mass distribution file
   CASE f  Read run case file

   CINI    Clear and initialize run cases
   MSET i  Apply mass file data to stored run case(s)

  .PLOP    Plotting options
   NAME s  Specify new configuration name

AVL   c>  load buzzed.avl

Reading file: buzzed.avl  ...

Configuration: DBF 08-09 Configuration 11-13-08

  Building surface: Wing
    Reading airfoil from file: sd7062.dat
    Reading airfoil from file: sd7062.dat
    Reading airfoil from file: sd7062.dat
    Reading airfoil from file: sd7062.dat

  Building duplicate image-surface: Wing (YDUP)

  Building surface: HTail

  Building duplicate image-surface: HTail (YDUP)

  Building surface: VTail

  Building duplicate image-surface: VTail (YDUP)

Mach =    0.0000  (default)
Nbody =   0      Nsurf =   6      Nstrp = 112      Nvor  =1408

Initializing run cases...

AVL   c> _
```

The main menu in which we will operate is the OPER menu, and it is shown below. We will explore most of the options here to some degree.

```
AVL   c>  oper

Operation of run case 1/1:   -unnamed-
==========================================================

 variable          constraint
 ------------      ------------------------
 A lpha        ->  alpha       =   0.000
 B eta         ->  beta        =   0.000
 R oll  rate   ->  pb/2V       =   0.000
 P itch rate   ->  qc/2V       =   0.000
 Y aw   rate   ->  rb/2V       =   0.000
 D1  aileron   ->  aileron     =   0.000
 D2  flap      ->  flap        =   0.000
 D3  elevator  ->  elevator    =   0.000
 D4  rudder    ->  rudder      =   0.000
 ------------      ------------------------

 C1  set level or banked  horizontal flight constraints
 C2  set steady pitch rate (looping) flight constraints
 M odify parameters

"#" select  run case          L ist defined run cases
 +  add new run case          S ave run cases to file
 -  delete  run case          F etch run cases from file
 N ame current run case       W rite forces to file

eX ecute run case             I nitialize variables

 G eometry plot               T refftz Plane plot

 ST  stability derivatives    FT  total   forces
 SB  body-axis derivatives    FN  surface forces
 RE  reference quantities     FS  strip   forces
                              FE  element forces
 DE  design changes           VM  strip shear,moment
 O ptions                     HM  hinge moments

.OPER (case 1/1)   c>  _
```

The first thing we will do is bring up the geometry plot. This allows us to verify that the geometry was read in and input correctly. The chordwise and spanwise sections can also be seen. The view can be controlled with the “v” command, where the angle of the camera view can be controlled. The only other command of interest here is the “lo” which turns on the loading. This will allow us to view the load on each panel when we start doing calculations.

```
.OPER (case 1/1)   c>  g

========================================
 K eystroke mode       V iewpoint
 A nnotate plot        O ptions
 H ardcopy plot        S elect surfaces
 Z oom                 U nzoom

 CH ordline     T      CA amber         F
 CN tlpoint     F      TR ailing legs   F
 BO ound leg    T      NO rmal vector   F
 LO ading       F      AX es, xyz ref.  T

Geometry plot command:
```


Az]m = -45°

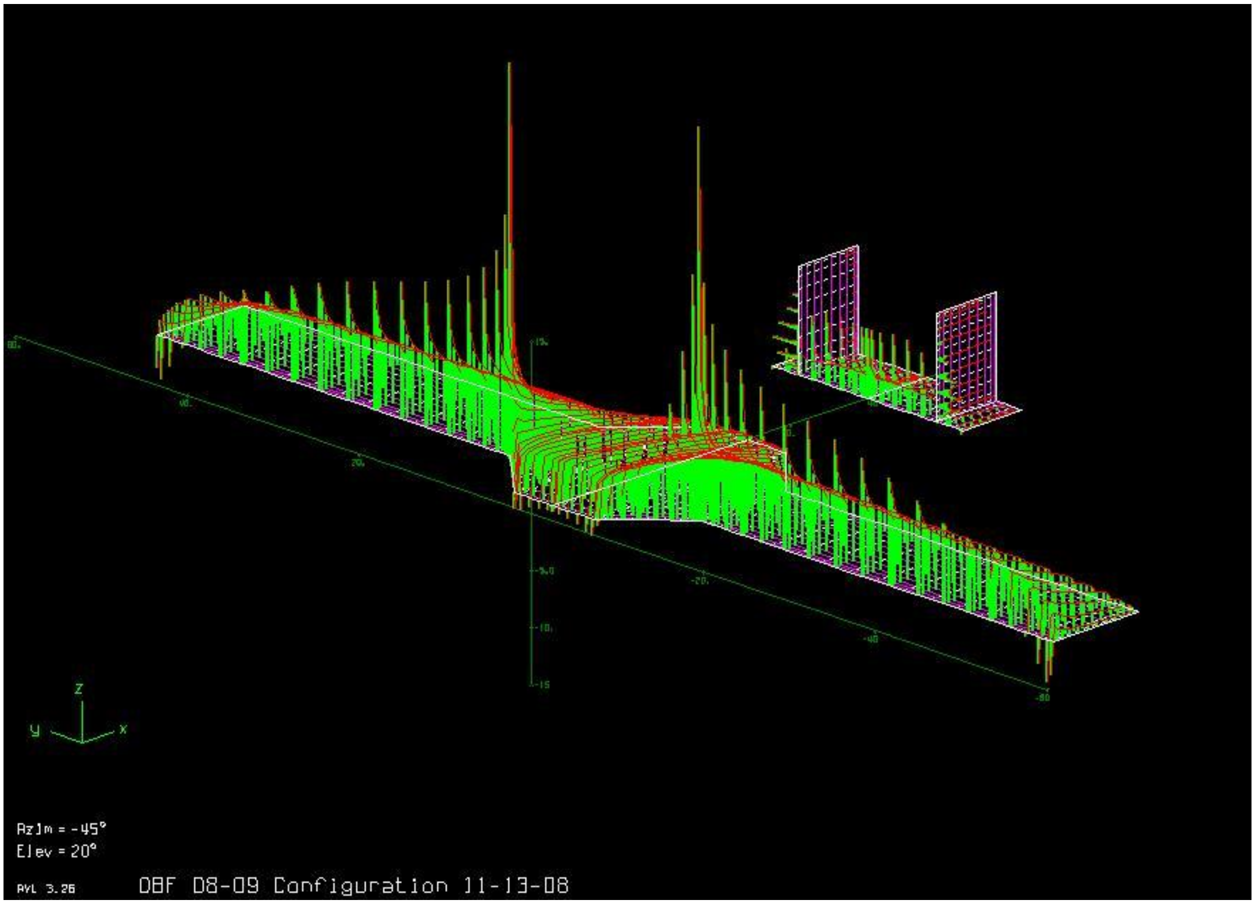
If we hit enter on the geometry menu it will take us back to the OPER menu. From here we can define the angle of attack. The command to do so is “a a 5” for five degrees angle of attack. The first a defines the variable; the second a says that angle of attack will be defined directly. It is also possible to define the lift coefficient and have AVL find the angle of attack. An example command to do this would be “a c 1.0”. The commands do not all have to be in a row; AVL will prompt the user after the first input if it needs more information. It is shorter and easier to use the combined format when the user knows what they are doing.

```
.OPER (case 1/1)   c>  a a 5

Operation of run case 1/1:   -unnamed-
==========================================================

 variable          constraint
 ------------      ------------------------
 A lpha        ->  alpha       =   5.000
 B eta         ->  beta        =   0.000
 R oll  rate   ->  pb/2V       =   0.000
 P itch rate   ->  qc/2V       =   0.000
 Y aw   rate   ->  rb/2V       =   0.000
 D1  aileron   ->  aileron     =   0.000
 D2  flap      ->  flap        =   0.000
 D3  elevator  ->  elevator    =   0.000
 D4  rudder    ->  rudder      =   0.000
 ------------      ------------------------

 C1  set level or banked  horizontal flight constraints
 C2  set steady pitch rate (looping) flight constraints
 M odify parameters

"#" select  run case          L ist defined run cases
 +  add new run case          S ave run cases to file
 -  delete  run case          F etch run cases from file
 N ame current run case       W rite forces to file

eX ecute run case             I nitialize variables

 G eometry plot               T refftz Plane plot

 ST  stability derivatives    FT  total   forces
 SB  body-axis derivatives    FN  surface forces
 RE  reference quantities     FS  strip   forces
                              FE  element forces
 DE  design changes           VM  strip shear,moment
 O ptions                     HM  hinge moments

.OPER (case 1/1)   c>
```

The command "x" then executes the actual solution. The main results are then displayed. The drag is shown as two different terms. The first, CDind, is the integrated drag solution looking at all the vortices on the lifting surfaces. The second, CDff, is the Trefftz Plane drag. This drag is taken by looking at the loss in momentum in the velocity far aft of the surfaces; the Trefftz Plane drag is typically more accurate and should be used in preference to the other.

```
.OPER (case 1/1)   c>  x
 Building normalwash AIC matrix...
 Factoring normalwash AIC matrix...
 Building source+doublet strength AIC matrix...
 Building source+doublet velocity AIC matrix...
 Building bound-vortex velocity matrix...

iter d(alpha)   d(beta)    d(pb/2V)   d(qc/2V)   d(rb/2V)   aileron    flap
  elevator   rudder
  1  0.107E-06  0.000E+00  0.000E+00  0.000E+00  0.000E+00  0.000E+00  0.000E+0
0  0.000E+00  0.000E+00
 ---------------------------------------------------------------
Vortex Lattice Output -- Total Forces

Configuration: DBF 08-09 Configuration 11-13-08
    # Surfaces =   6
    # Strips   = 112
    # Vortices =1408

 Sref =  1233.8       Cref =  11.864       Bref =  104.00
 Xref =  11.000       Yref =  0.0000       Zref =  0.0000

Standard axis orientation,  X fwd, Z down

Run case:  -unnamed-

 Alpha =   5.00000     pb/2V =   0.00000     p'b/2V =   0.00000
 Beta  =   0.00000     qc/2V =   0.00000
 Mach  =     0.000     rb/2V =   0.00000     r'b/2V =   0.00000

 CXtot =   0.04422     Cltot =   0.00000     Cl'tot =   0.00000
 CYtot =   0.00000     Cmtot =  -0.11042
 CZtot =  -0.74772     Cntot =   0.00000     Cn'tot =   0.00000

 CLtot =   0.74873
 CDtot =   0.02111
 CDvis =   0.00000     CDind =   0.02111
 CLff  =   0.75062     CDff  =   0.01993    | Trefftz
 CYff  =   0.00000         e =    1.0265    | Plane

  aileron         =   0.00000
  flap            =   0.00000
  elevator        =   0.00000
  rudder          =   0.00000

 ---------------------------------------------------------------
```

If we go back to the geometry we can see the force on each section. The rough lift distribution can be seen as well as the chordwise pressure distribution. The spikes below are due to the abrupt change in chord causing a discontinuity in lift.



Back to the OPER menu and we can look at the full set of stability derivatives using the “st” command. The resulting output is displayed below. The results can also be written to a file. It is important to note that the units for the control surface derivatives are in per degree and not per radian. All the other derivatives are given in per radian.

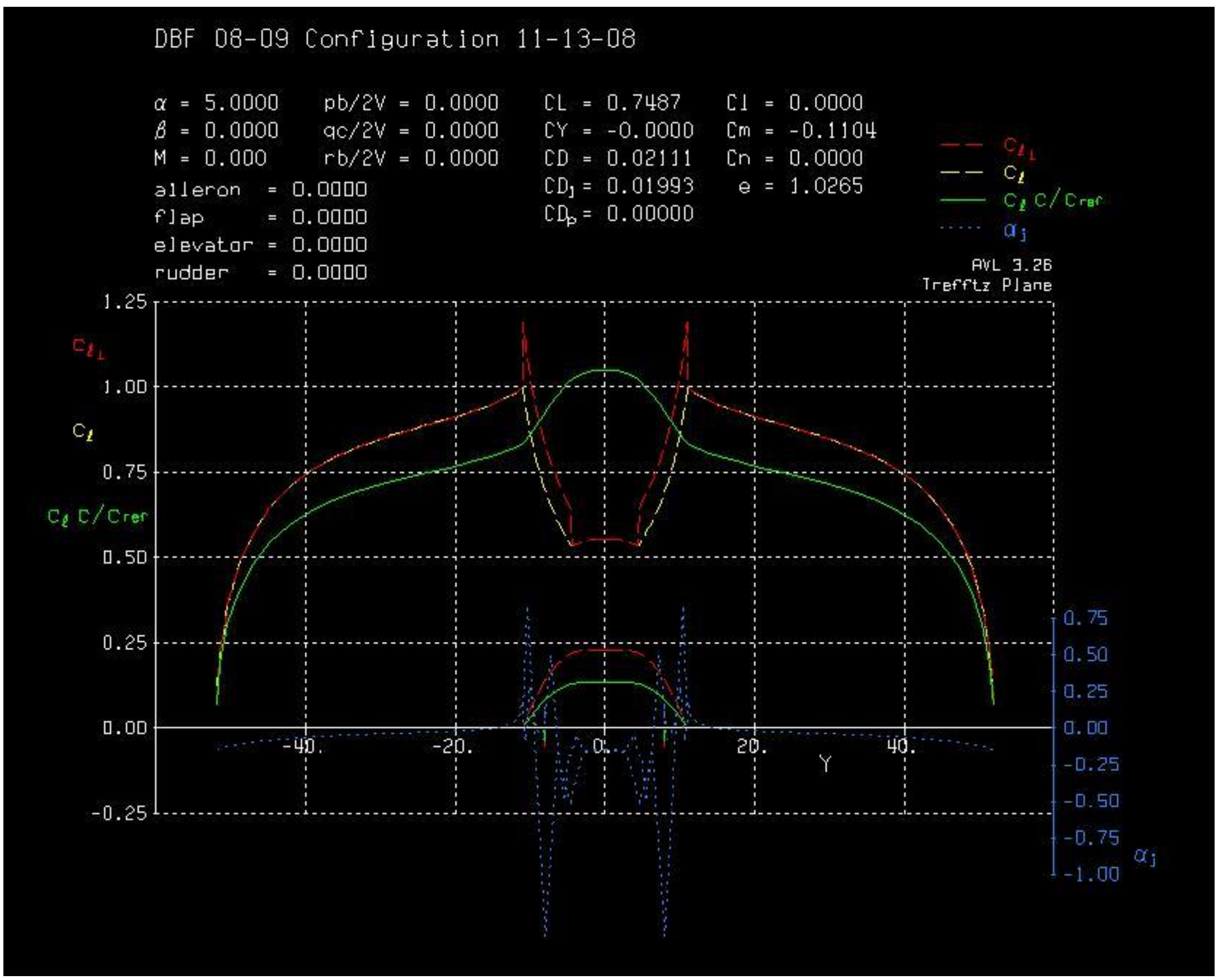
```
 ---------------------------------------------------------------

 Stability-axis derivatives...

                             alpha                beta
                      ----------------     ----------------
 z' force CL |     CLa =   4.871125     CLb =   0.000000
 y  force CY |     CYa =   0.000000     CYb =  -0.203220
 x' mom.  Cl'|     Cla =   0.000000     Clb =  -0.048339
 y  mom.  Cm |     Cma =  -0.581589     Cmb =   0.000000
 z' mom.  Cn'|     Cna =   0.000000     Cnb =   0.062223

                     roll rate  p'     pitch rate  q'     yaw rate  r'
                  ----------------   ----------------   ----------------
 z' force CL |     CLp =   0.000000     CLq =   6.801833     CLr =   0.000000
 y  force CY |     CYp =   0.119693     CYq =   0.000000     CYr =   0.118425
 x' mom.  Cl'|     Clp =  -0.487251     Clq =   0.000000     Clr =   0.179998
 y  mom.  Cm |     Cmp =   0.000000     Cmq =  -6.024068     Cmr =   0.000000
 z' mom.  Cn'|     Cnp =  -0.062103     Cnq =   0.000000     Cnr =  -0.043589

                   aileron      d1     flap         d2     elevator     d3     ru
dder       d4
                  ----------------   ----------------   ----------------   --
--------------
 z' force CL |   CLd1 =   0.000000   CLd2 =   0.037942   CLd3 =   0.005256   CLd
4 =   0.000000
 y  force CY |   CYd1 =  -0.000772   CYd2 =   0.000000   CYd3 =   0.000000   CYd
4 =   0.002560
 x' mom.  Cl'|   Cld1 =   0.008629   Cld2 =   0.000000   Cld3 =   0.000000   Cld
4 =   0.000107
 y  mom.  Cm |   Cmd1 =   0.000000   Cmd2 =  -0.008028   Cmd3 =  -0.012454   Cmd
4 =   0.000000
 z' mom.  Cn'|   Cnd1 =   0.000489   Cnd2 =   0.000000   Cnd3 =   0.000000   Cnd
4 =  -0.000790
 Trefftz drag| CDffd1 =   0.000000 CDffd2 =   0.001911 CDffd3 =   0.000338 CDffd
4 =   0.000000
 span eff.   |    ed1 =   0.000000    ed2 =   0.005684    ed3 =  -0.002932    ed
4 =   0.000000



 Neutral point  Xnp =  12.416505

 Clb Cnr / Clr Cnb  =    0.188128    (  > 1 if spirally stable )

 Enter output filename (or <Return>): temp.out
 File  temp.out  exists.  Overwrite?  Y
```

The hingemoments can also be shown using the “hm” command from the OPER menu. The hingemoment coefficients for the current condition are displayed for each control surface. The hingemoment coefficient is equal to the following, where S_{cs} is the planform area of the control surface and c_{cs} is the mean chord of the control surface.

$$C_{hm} = \frac{Moment}{\bar{q} S_{cs} c_{cs}}$$

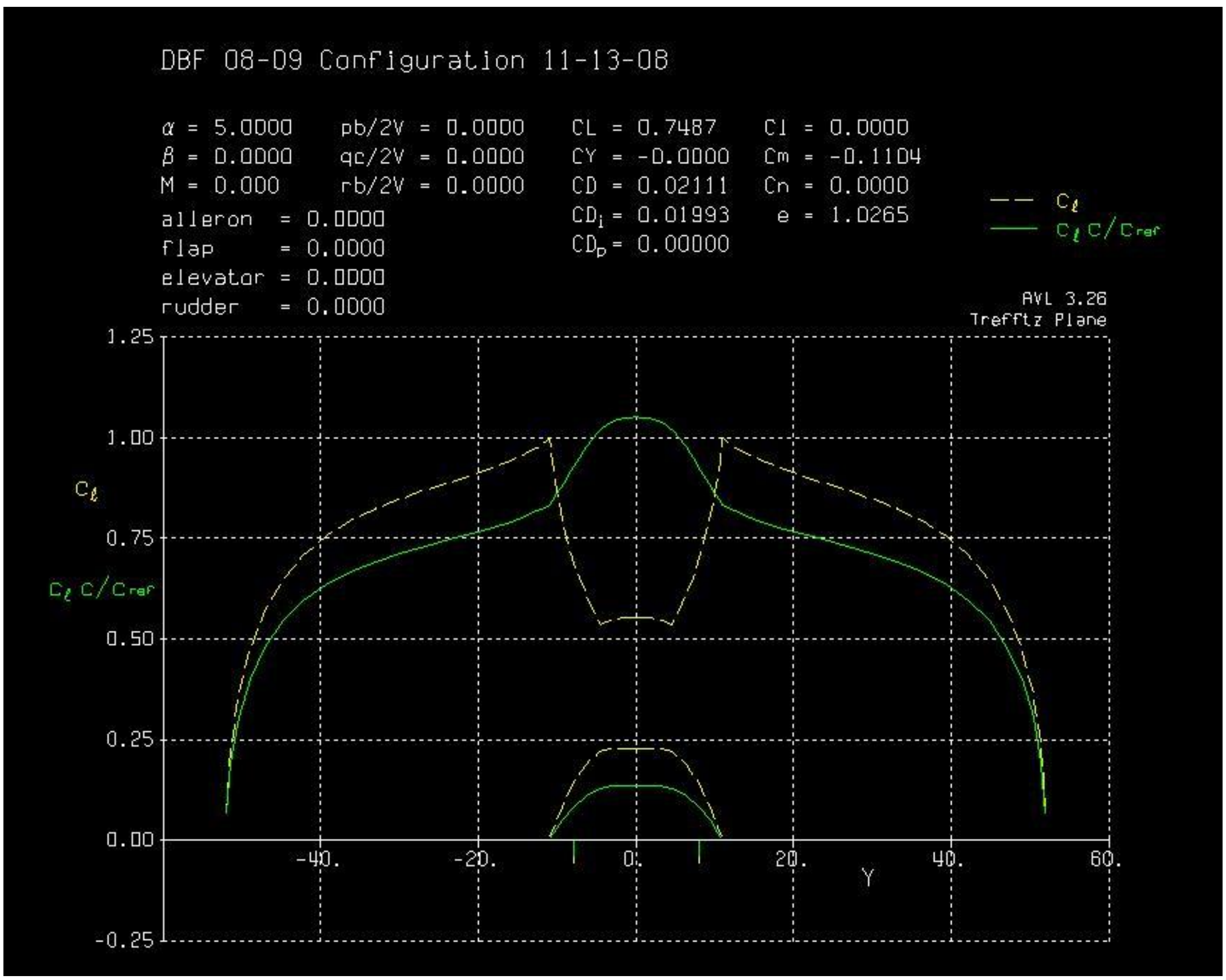
```
.OPER (case 1/1)   c>  hm
------------------------------------------------------------------
Control Hinge Moments
(referred to    Sref =    1234.       Cref =    11.8640)

Control           Chinge
----------------- ----------
aileron           -0.8619E-09
flap              -0.7413E-02
elevator          -0.9407E-04
rudder            -0.3787E-11
------------------------------------------------------------------
```

From the geometry menu there is also the option to view the lift distribution. The initial screen is shown below. The lift coefficient perpendicular to the surface, the lift coefficient, the lift coefficient normalized by local chord (this one is proportional to the actual lift generated), and the induced angle of attack are shown for all horizontal surfaces (no vertical tail, but horizontal tail is there). The plot is a mess, and it is difficult to figure out what is happening.



If we input “p” and “w” then we get rid of the perpendicular lift and the downwash and we have a useful plot. The lift distribution is shown here, and that can be used to assess how close to elliptical the lift is; in this case the lifting body in the middle keeps it far from elliptical. The lift coefficient shows the local section lift across the span. The center section has a lower lift coefficient even though the actual lift is higher; that is the result of the much higher chord.

This plot can be used to assess vehicle maximum lift coefficient in an iterative fashion. The maximum section lift here is 1.00. If the airfoil maximum section lift coefficient was 1.00 then this would be the vehicle max lift. Otherwise the angle of attack can be increased until some section is at the section max lift coefficient. The vortex lattice solution method is not accurate past the linear region, so the results cannot be trusted any further than when any part is at the local section max lift coefficient. DO NOT FORGET THIS!



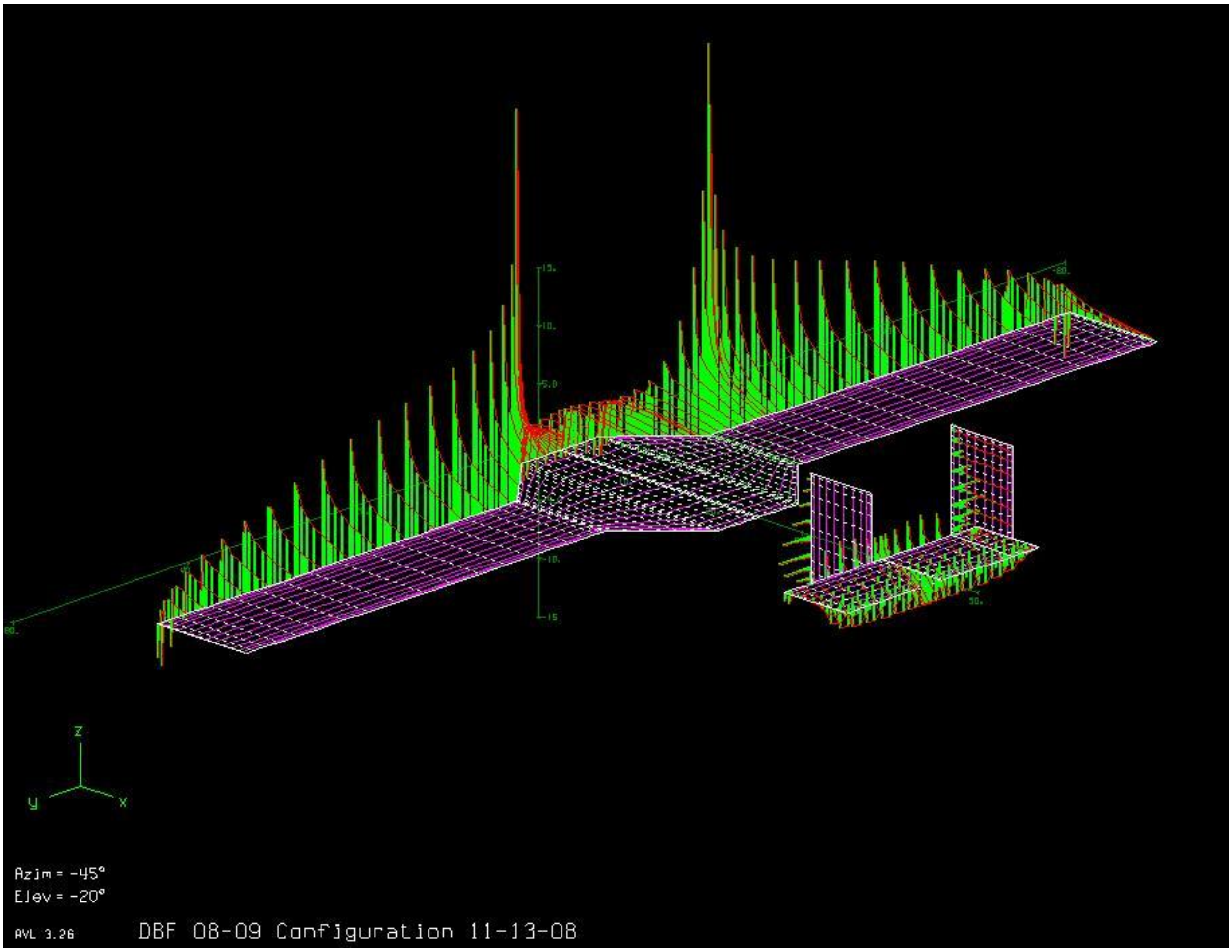
Trim can be evaluated from the OPER menu. The procedure to do this is to first select the control surface, in this case “d3”, which is the elevator. Then define the moment that surface is to zero out, in this case the pitching moment, or “pm”. Then what level it should be set to, zero in the example. The combined command is then “d3 pm 0”. Similar procedures can be used to zero out the yawing and rolling moments with the rudder and aileron, respectively.

```
.OPER (case 1/1)   c>  d3 pm 0

Operation of run case 1/1:   -unnamed-
==========================================================

 variable          constraint
 ------------      ------------------------
 A lpha       ->   alpha       =   5.000
 B eta        ->   beta        =   0.000
 R oll  rate  ->   pb/2V       =   0.000
 P itch rate  ->   qc/2V       =   0.000
 Y aw   rate  ->   rb/2V       =   0.000
 D1  aileron  ->   aileron     =   0.000
 D2  flap     ->   flap        =   0.000
 D3  elevator ->   Cm pitchmom =   0.000
 D4  rudder   ->   rudder      =   0.000
 ------------      ------------------------
```

Once we execute the new case we get the elevator required to trim the airplane. In this case it is 8.87 degrees. The lift coefficient has also been reduced from 0.75 to 0.7 from the elevator down force.

```
.OPER (case 1/1)   c>  x

 iter d(alpha)   d(beta)    d(pb/2V)   d(qc/2V)   d(rb/2V)   aileron    flap
   elevator   rudder
   1  0.107E-06  0.000E+00  0.000E+00  0.387E-09  0.000E+00  0.000E+00  0.000E+0
0 -0.887E+01  0.000E+00
   2  0.107E-06  0.000E+00  0.000E+00 -0.387E-09  0.000E+00  0.000E+00  0.000E+0
0 -0.585E-02  0.000E+00
   3  0.107E-06  0.000E+00  0.000E+00  0.255E-15  0.000E+00  0.000E+00  0.000E+0
0 -0.418E-05  0.000E+00
 ---------------------------------------------------------------
 Vortex Lattice Output -- Total Forces

 Configuration: DBF 08-09 Configuration 11-13-08
     # Surfaces =   6
     # Strips   = 112
     # Vortices =1408

  Sref =  1233.8       Cref =  11.864       Bref =  104.00
  Xref =  11.000       Yref =  0.0000       Zref =  0.0000

 Standard axis orientation,  X fwd, Z down

 Run case:  -unnamed-

  Alpha =   5.00000     pb/2V =    0.00000     p'b/2V =   0.00000
  Beta  =   0.00000     qc/2V =    0.00000
  Mach  =     0.000     rb/2V =    0.00000     r'b/2V =   0.00000

  CXtot =   0.04139     Cltot =    0.00000     Cl'tot =   0.00000
  CYtot =   0.00000     Cmtot =    0.00000
  CZtot =  -0.70102     Cntot =    0.00000     Cn'tot =   0.00000

  CLtot =   0.70196
  CDtot =   0.01987
  CDvis =   0.00000     CDind =    0.01987
  CLff  =   0.70368     CDff  =    0.01513    | Trefftz
  CYff  =   0.00000         e =     1.1881    | Plane

   aileron         =   0.00000
   flap            =   0.00000
   elevator        =  -8.87223
   rudder          =   0.00000

 ---------------------------------------------------------------
```

If we go back to the geometry we can see the effect of the elevator deflection to trim. This view is from below the airplane so that the downward force on the elevator panels can be clearly seen.



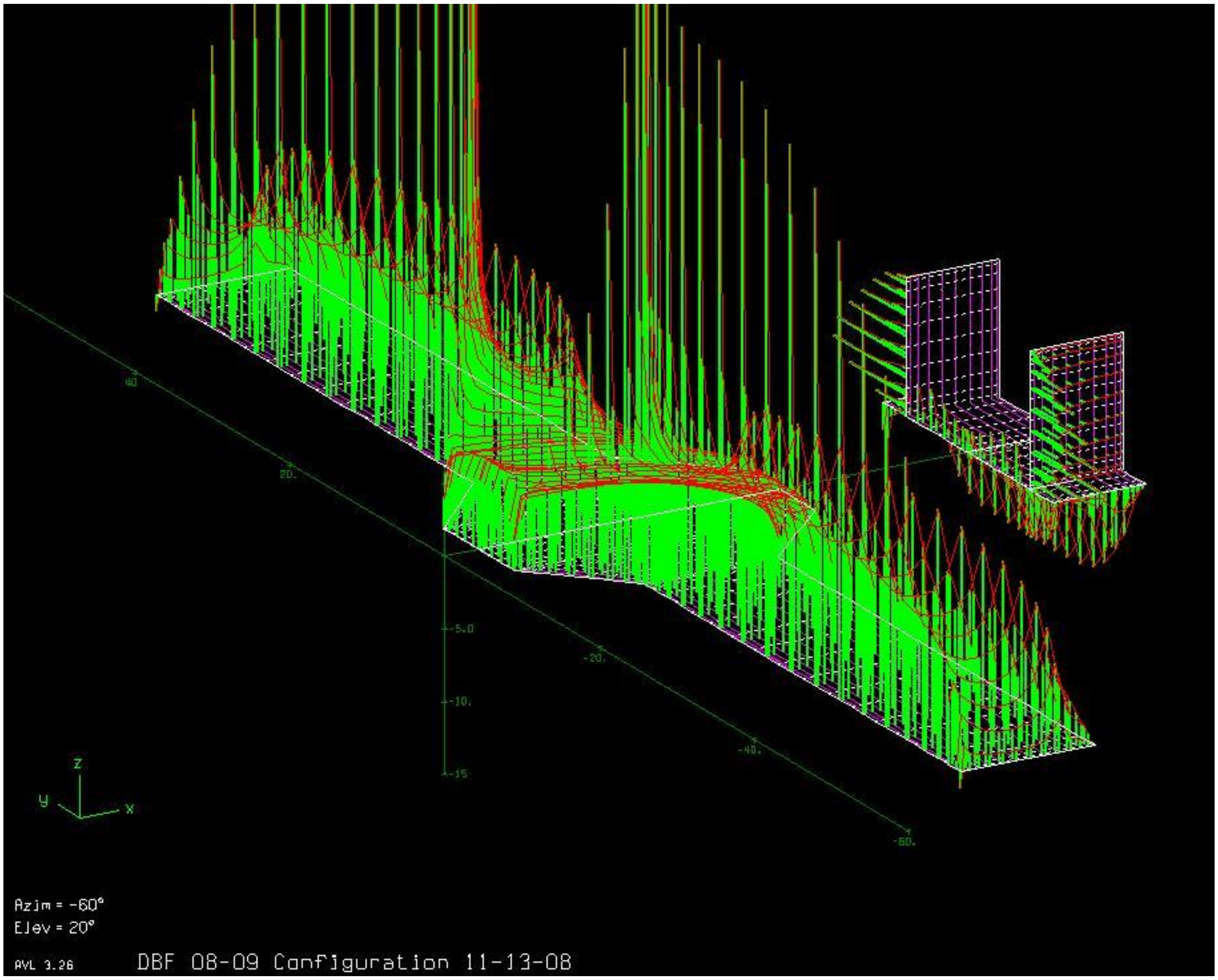
A control surface deflection does not have to trim a moment out; they can also be directly specified. An example below is shown for the flap deflection. The command is “d2 d2 20” to deflect 20 degrees. The first “d2” selects the control surface, the second “d2” says that we will define it directly, and the last part is the actual deflection.

```
.OPER (case 1/1)   c>  d2 d2 20

Operation of run case 1/1:   -unnamed-
==========================================================

  variable          constraint
  ------------      ------------------------
  A lpha       ->   alpha        =   5.000
  B eta        ->   beta         =   0.000
  R oll  rate  ->   pb/2V        =   0.000
  P itch rate  ->   qc/2V        =   0.000
  Y aw   rate  ->   rb/2V        =   0.000
  D1  aileron  ->   aileron      =   0.000
  D2  flap     ->   flap         =   20.00
  D3  elevator ->   Cm pitchmom  =   0.000
  D4  rudder   ->   rudder       =   0.000
  ------------      ------------------------
```

When we execute the case the results are below. The lift coefficient is up from 0.7 to 1.38, and the required elevator deflection is 21.86 degrees, which is about as high as is reasonable to expect before having the horizontal tail stall.

```
.OPER (case 1/1)   c>  x

iter d(alpha)   d(beta)    d(pb/2V)   d(qc/2V)   d(rb/2V)   aileron    flap
  elevator   rudder
   1  0.107E-06  0.000E+00  0.000E+00 -0.208E-11  0.000E+00  0.000E+00  0.200E+0
2 -0.129E+02  0.000E+00
   2  0.107E-06  0.000E+00  0.000E+00  0.208E-11  0.000E+00  0.000E+00  0.000E+0
0 -0.812E-01  0.000E+00
   3  0.107E-06  0.000E+00  0.000E+00 -0.290E-14  0.000E+00  0.000E+00  0.000E+0
0 -0.183E-03  0.000E+00
   4  0.107E-06  0.000E+00  0.000E+00  0.282E-14  0.000E+00  0.000E+00  0.000E+0
0  0.762E-06  0.000E+00
 ---------------------------------------------------------------
 Vortex Lattice Output -- Total Forces

 Configuration: DBF 08-09 Configuration 11-13-08
     # Surfaces =   6
     # Strips   = 112
     # Vortices =1408

  Sref =  1233.8       Cref =  11.864       Bref =  104.00
  Xref =  11.000       Yref =  0.0000       Zref =  0.0000

 Standard axis orientation,  X fwd, Z down

 Run case:  -unnamed-

  Alpha =   5.00000     pb/2V =   0.00000     p'b/2V =   0.00000
  Beta  =   0.00000     qc/2V =   0.00000
  Mach  =     0.000     rb/2V =   0.00000     r'b/2V =   0.00000

  CXtot =   0.03051     Cltot =   0.00000     Cl'tot =   0.00000
  CYtot =   0.00000     Cmtot =   0.00000
  CZtot =  -1.39141     Cntot =   0.00000     Cn'tot =   0.00000

  CLtot =   1.38877
  CDtot =   0.09088
  CDvis =   0.00000     CDind =   0.09088
  CLff  =   1.39640     CDff  =   0.06291    | Trefftz
  CYff  =   0.00000         e =    1.1255    | Plane

   aileron         =   0.00000
   flap            =  20.00000
   elevator        = -21.85912
   rudder          =   0.00000

 ---------------------------------------------------------------
```

We can go back to the geometry plot to see the effect of the flap deflection. The peaks are all much higher and a close look reveals that the chordwise distribution has a big hump in the back due to the flaps. The down force on the elevator has also massively increased.


40
20.
-5.0
-10.
-15
-20.
-40.
-60.
z
y
x
Azim = -60°
Elev = 20°
AVL 3.26
DBF 08-09 Configuration 11-13-08

The forces on each surface can be displayed with “fn”. The result is shown below. The surface names have been partially cut-off because they do not quite fit on the page, but they can be deciphered to figure out the contributions from each.

```
.OPER (case 1/1)   c>  fn
 ---------------------------------------------------------------
 Surface Forces (referred to Sref,Cref,Bref about Xref,Yref,Zref)
 Standard axis orientation,  X fwd, Z down

     Sref =   1234.        Cref =   11.8640   Bref =  104.0000
     Xref =     11.0000    Yref =    0.0000   Zref =    0.0000

 n     Area      CL       CD       Cm       CY       Cn       Cl      CDi      CDv
 1   616.875  0.7251  0.0442 -0.0840 -0.0018 -0.0010 -0.1726  0.0442  0.0000   W
ing
 2   616.875  0.7251  0.0442 -0.0840  0.0018  0.0010  0.1726  0.0442  0.0000   W
ing (YDUP)
 3    77.000 -0.0308  0.0019  0.0842  0.0000  0.0002  0.0015  0.0019  0.0000   H
Tail
 4    77.000 -0.0308  0.0019  0.0842  0.0000 -0.0002 -0.0015  0.0019  0.0000   H
Tail (YDUP)
 5    66.500  0.0001 -0.0006 -0.0002  0.0116 -0.0033  0.0004 -0.0006  0.0000   V
Tail
 6    66.500  0.0001 -0.0006 -0.0002 -0.0116  0.0033 -0.0004 -0.0006  0.0000   V
Tail (YDUP)

 Surface Forces (referred to Ssurf, Cave about root LE on hinge axis)

    n     Ssurf      Cave       cl       cd      cdv     cm_LE
    1    616.875    11.863   1.4503   0.0883   0.0000   0.0000  Wing
    2    616.875    11.863   1.4503   0.0883   0.0000   0.0000  Wing (YDUP)
    3     77.000     7.000  -0.4936   0.0305   0.0000   0.0000  HTail
    4     77.000     7.000  -0.4936   0.0305   0.0000   0.0000  HTail (YDUP)
    5     66.500     7.000  -0.2148  -0.0117   0.0000   0.0000  VTail
    6     66.500     7.000  -0.2148  -0.0117   0.0000   0.0000  VTail (YDUP)
 ---------------------------------------------------------------
```

From the OPER menu we can also look at the stripwise forces. These are equivalent to the section lift generated over each strip. The command to look at these is “fs”. An example for just the horizontal tail is below. The wing has enough strips that it does not fit well on a single page.

```
 Surface # 3      HTail
    # Chordwise = 10   # Spanwise = 10     First strip = 73
    Surface area =   77.000000       Ave. chord =    7.000000
    CLsurf  =  -0.03080     Clsurf  =   0.00148
    CYsurf  =   0.00000     Cmsurf  =   0.08423
    CDsurf  =   0.00190     Cnsurf  =   0.00022
    CDisurf =   0.00190     CDvsurf =   0.00000

 Forces referred to Ssurf, Cave about hinge axis thru LE
    CLsurf  =  -0.49357     CDsurf  =   0.03051
    Deflect =

Strip Forces referred to Strip Area, Chord
   j      Yle     Chord    Area     c cl      ai      cl_norm  cl        cd
 cdv    cm_c/4    cm_LE  C.P.x/c
   73   0.0677   7.0000   1.8843  -3.9176  -0.0082  -0.5647  -0.5647   0.0331
0.0000   0.2323   0.3722    0.661
   74   0.5995   7.0000   5.4685  -3.9122  -0.0074  -0.5639  -0.5639   0.0331
0.0000   0.2323   0.3721    0.662
   75   1.6109   7.0000   8.5174  -3.8735  -0.0014  -0.5584  -0.5584   0.0333
0.0000   0.2328   0.3712    0.667
   76   3.0031   7.0000  10.7326  -3.7782   0.0191  -0.5447  -0.5447   0.0336
0.0000   0.2337   0.3686    0.679
   77   4.6396   7.0000  11.8972  -3.6624   0.2661  -0.5281  -0.5281   0.0336
0.0000   0.2330   0.3638    0.691
   78   6.3604   7.0000  11.8972  -3.5635   0.0067  -0.5139  -0.5139   0.0329
0.0000   0.2266   0.3538    0.691
   79   7.9969   7.0000  10.7326  -3.3277   6.6461  -0.4800  -0.4800   0.0319
0.0000   0.2094   0.3283    0.686
   80   9.3891   7.0000   8.5174  -2.9941  -0.3711  -0.4317  -0.4317   0.0270
0.0000   0.1738   0.2808    0.653
   81  10.4005   7.0000   5.4685  -2.2784   0.4249  -0.3281  -0.3281   0.0160
0.0000   0.1163   0.1977    0.604
   82  10.9323   7.0000   1.8843  -0.8639  -0.4545  -0.1243  -0.1243   0.0052
0.0000   0.0412   0.0720    0.581
```