FUN3D v13.4 Training

Session 5: Visualization and Component Force and Moment Tracking

Jan Carlson

Learning Goals

What we will cover

- Visualization overview (&sampling_parameters)
- Component force and moment tracking (&component_parameters)

Visualization Learning Goals

- What this will teach you
  - Run-time flow visualization output
    - Output on boundary surfaces
    - Output on user-specified “sampling” surfaces within the volume
    - Output of full volume data
    - Output generated by “slicing” boundary data - “sectional” output
- What you will not learn
  - The plethora of output options available for visualization
  - Tecplot usage
- What should you already know
  - Basic flow solver operation and control

User Inputs for FUN3D

Input deck `fun3d.nml`

- The user is required to supply an input deck for FUN3D named `fun3d.nml` (fixed name)
- **&sampling_parameters**
- **&sampling_output**
- &component_parameters

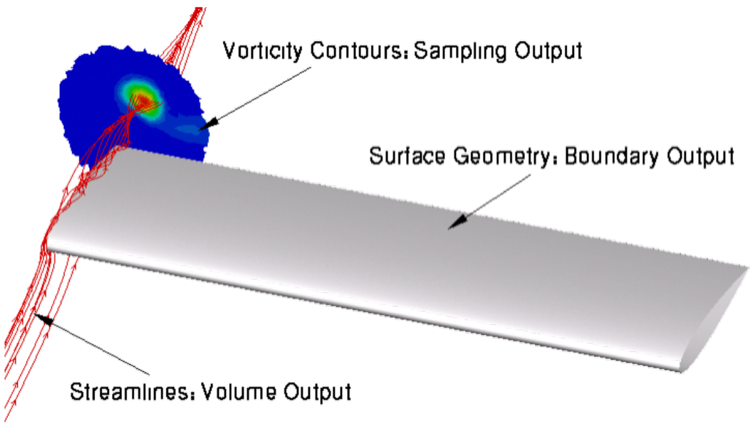
List of Key Input/Output Files

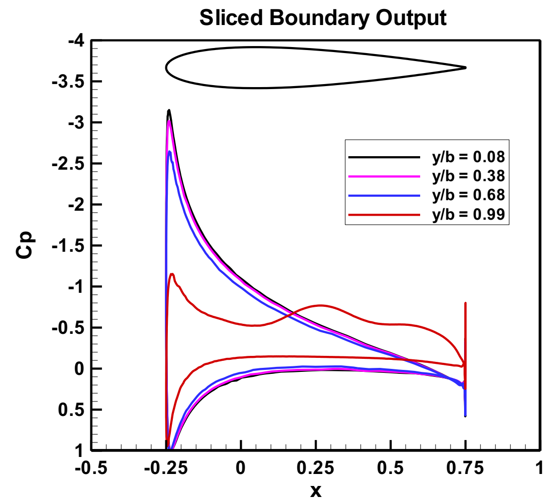
- Input
  - Grid files (prefixed with project name, suffixes depend on grid format)
  - **`fun3d.nml`**
- Output
  - **`[project_name].grid_info`**
  - **`[project_name].forces`**
  - **`[project_name]_hist.dat`**
  - **`[project_name].flow`**
  - **`[project_name]_tec_sampling_geom[n].dat`**
  - **`[project_name]_fm_[component_name].dat`**
  - **`[project_name]_stream_info.dat`**

Background

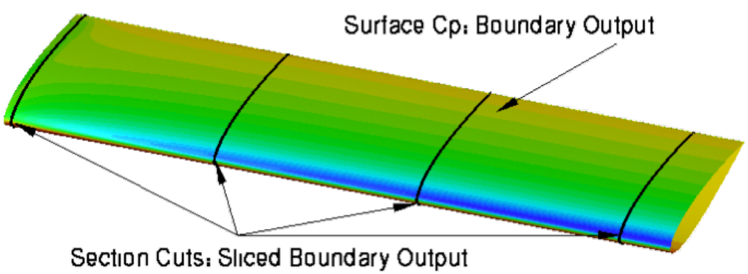
- Datasets are getting simply too large to post-process in a traditional manner
- FUN3D allows visualization data to be generated as the solver is running
    - User specified frequency and output type
    - User specified output variables from a fairly extensive list
- Majority of output options are Tecplot-based
    - Volume output may also be generated in Fieldview, CGNS formats
- Note FUN3D also supports true insitu visualization at scale using the DoE VisIt package; however, this is not covered here
    - Intelligent Light is currently integrating VisIt's insitu capabilities with Fieldview

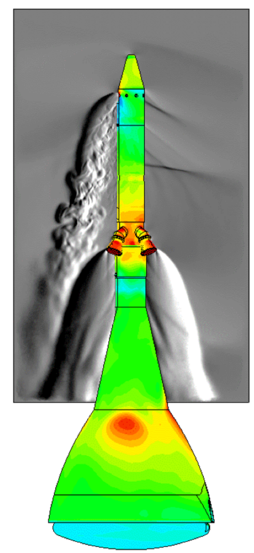
Selected Visualization Output Examples





Isosurfaces





Schlieren,
boundary output

Visualization Overview

- All of the visualization outputs require similar namelist-specified "frequency" N to activate:
  - In all cases, N = 0, 1, 2, 3, …
    - N = 0 generates no output
    - N < 0 generates output only at the ***end*** of the run - typically used for steady-state cases. The actual value of N is ignored
    - N > 0 generates output every N^{th} time step - typically used to generate animation for unsteady flows; can also be used to observe how a steady flow converges

Visualization Overview

- Customizable output variables (except sliced boundary data):
  - Most variables are the same between the boundary surface, sampling and volume output options; boundary surface has a few extra
  - See manual for lists of all available variables
  - Default variables always include x, y, z, and the “primitive” flow variables u, v, w, and p (plus density if compressible)
  - Several “shortcut” variables: e.g.,

    `primitive_variables = rho, u, v, w, p`
  - Must explicitly turn off the default variables if you don’t want them (e.g., **`primitive_variables = .false.`**)
  - Variable selection for each co-processing option done with a different namelist to allow “mix and match”

Visualization Overview

- For boundary surface output, default is all solid boundaries in 3D and one y=const plane in 2D; alternate output boundaries selected with, e.g.:

```
&boundary_output_variables
   number_of_boundaries = 3
   boundary_list = '3,5,9'   ! blanks OK as
                                 delimiter too: '3 5 9'
                             ! dashes OK as delimiter
                                 too: '3-9'
 /
```

- If you already have a converged solution and don't want to advance the solution any further, can do a "pass through" run:
  - set **`steps = 0`** in **`&code_run_control`**
  - You must have a restart file (**`[project_name].flow`**)
  - Run the solver with the appropriate namelist input to get desired output
  - **`[project_name].flow`** will remain unaltered after completion

Visualization Overview

- Sampling output requires additional data to describe the desired sampling surface(s)
  - Specified in namelist **`&sampling_parameters`**
  - Surfaces may be planes, quadrilaterals or circles of arbitrary orientation, or may be spheres or boxes
  - Isosurfaces and schlierens also available
  - Points may also be sampled (special format for time histories, acoustics)
  - See manual for complete info
- Sliced boundary surface output requires additional data to describe the desired slice section(s)
  - Specified in namelist **`&slice_data`**
  - Always / only outputs x, y, z, C_p, C_{fx}, C_{fy}, C_{fz}
  - User specifies which (solid) boundaries to slice, and where
  - See manual for complete info
- Output files will be ASCII unless you have built FUN3D against the Tecplot library (exception: sliced boundary data is always ASCII)

Visualization Overview

- ASCII files have .dat extension
- Binary files have (.plt | .szplt) extension - smaller files; load into Tecplot faster
- Boundary output file naming convention (T = time step counter):
  - **`[project_name]_tec_boundary_timestep[T](.dat|.plt|.szplt)`** if N > 0
  - **`[project_name]_tec_boundary(.dat|.plt|.szplt)`** if N < 0
- Volume output file naming convention (note: 1 file *per processor* P; for a single file, Fun3D needs to be linked with TecIO-MPI, a parallel version of tecio).
  - **`[project_name]_part[P]_tec_volume_timestep[T](.dat|.plt|.szplt)`** if N > 0
  - **`[project_name]_part[P]_tec_volume(.dat|.plt|.szplt)`** if N < 0
- Sampling output file naming convention (one file per sampling geometry G):
  - **`[project_name]_tec_sampling_geom[G]_timestep[T](.dat|.plt|.szplt)`** if N > 0
  - **`[project_name]_tec_sampling_geom[G](.dat|.plt|.szplt)`** if N < 0

Boundary Output Visualization Example

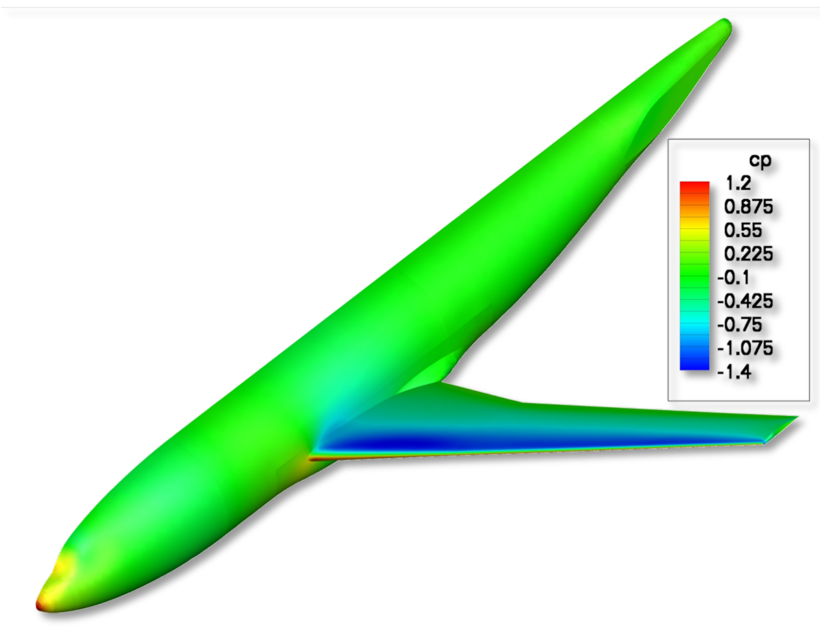
```
&global
  boundary_animation_freq = -1
/
&boundary_output_variables
  primitive_variables = .false.
  cp                  = .true.
  yplus               = .true.
/
```

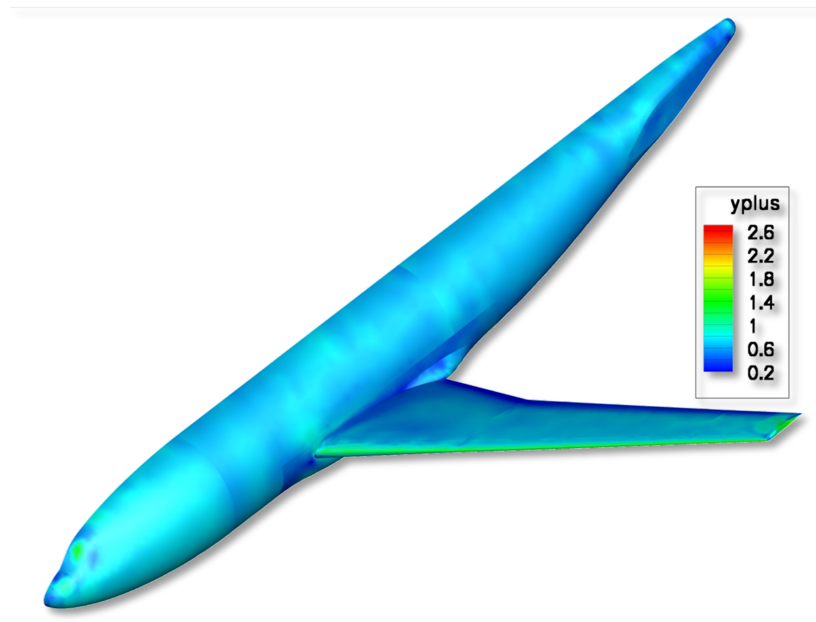
Dump boundary vis at end of run

Turn off rho, u, v, w, p

Turn on C_p

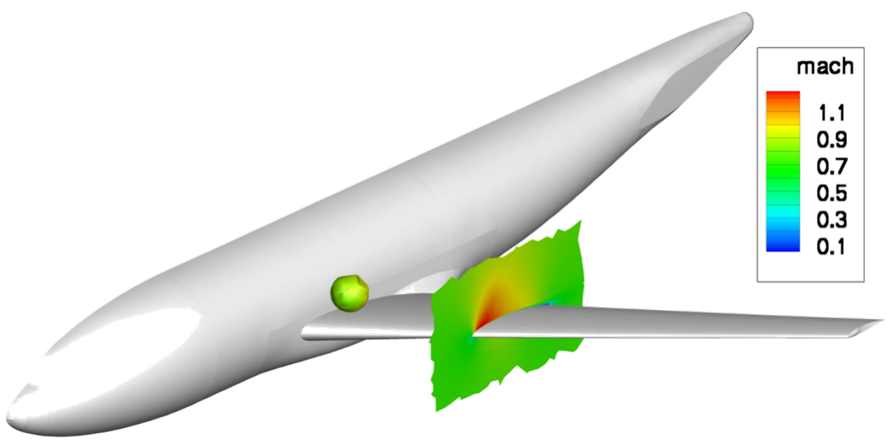
Turn on y^+

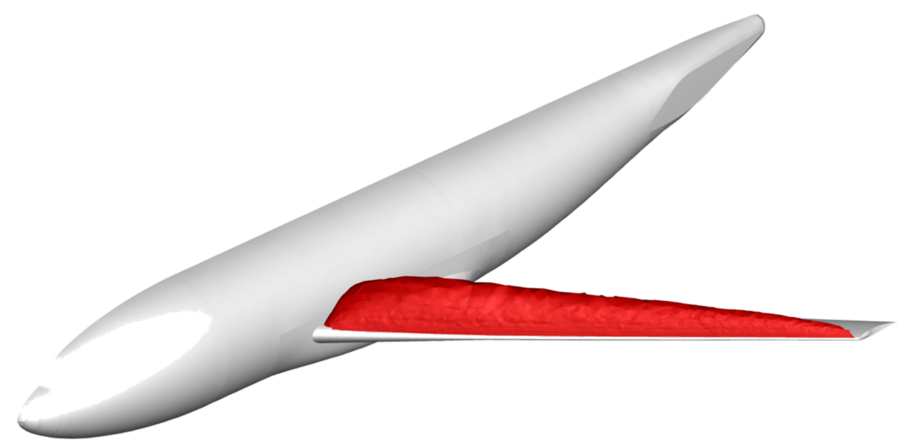




Sampling Visualization Example

`&sampling_parameters`	
`number_of_geometries   = 3`	Want 3 sampling geometries
`type_of_geometry(1)    = 'plane'`	First geometry is a plane
`plane_center(2,1)      = -234.243`	Plane y-coordinate
`plane_normal(2,1)      = 1.0`	Plane y-normal
`sampling_frequency(1) = -1`	Write at end of run
`type_of_geometry(2)    = 'sphere'`	Second geometry is a sphere
`sphere_center(:,2)     = 74.9,–107.7,50.`	Center x,y-z-coordinates
`sphere_radius(2)       = 20.0`	Sphere radius
`sampling_frequency(2) = -1`	Write at end of run
`type_of_geometry(3)    = 'isosurface'`	Third geometry is an isosurface
`isosurf_variable(3)    = 'mach'`	Isosurface will be based on Mach number
`isosurf_value(3)       = 1.00`	Isosurface defined by Mach=1
`sampling_frequency(3) = -1`	Write at end of run
`/`	
`&sampling_output_variables`	
`primitive_variables = .false.`	Turn off rho, u, v, w, p
`mach                = .true.`	Turn on Mach number
`/`	





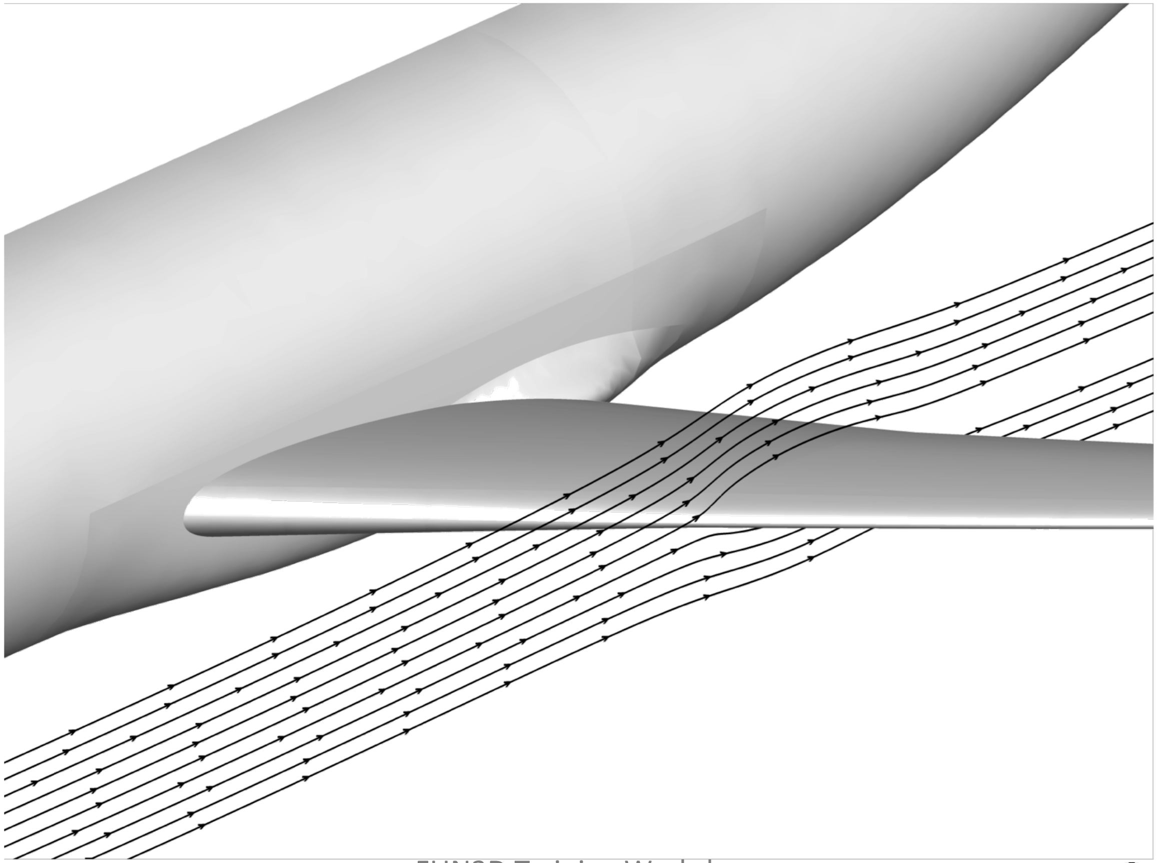
Volume Visualization Example

```
&global
  volume_animation_freq = -1
/
&volume_output_variables
  export_to='tecplot'
          ! 'tec'
/
```

Dump output at end of run

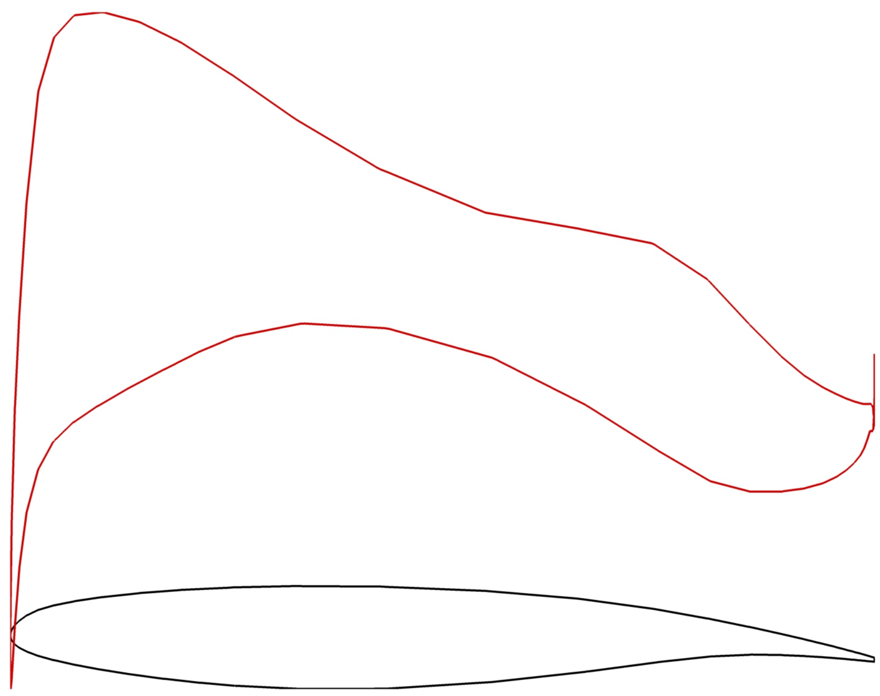
Writes results to Tecplot file

Writes results to a single ASCII Tecplot file



Slicing Visualization Example

```
&global
  slice_freq = -1                    Dump output at end of run
/
&slice_data
  nslices = 1                        Perform one slice
  slice_location(1) = -234.243       Coordinate of slice
/
```



Troubleshooting/FAQ

- I can see what look like ragged dark lines on sampling surfaces and volume data – what is that?
  - Duplicate information at partition boundaries is not removed; if surface is not completely opaque, double plotting locally doubles the opaqueness (duplicate info ***is*** removed from boundary surface output)
  - Turn off transparency in Tecplot
- When I dump out volume plot files in Tecplot format, I get a file for every processor – is there a way around this?
  - Not currently. However, Tecplot can be easily told to load all of the files at once without having to individually select them all.
  - The team is working with Tecplot to develop their next generation of I/O API's, with special focus on massively parallel needs
  - Alternative: switch to Fieldview or CGNS output, which uses a single file

Component Force Tracking

Component Tracking Learning Goals

- What this will teach you
  - Namelist setup for grouping boundaries to get run-time forces and moments (F&M)
    - Solid boundary: skin friction and pressure force
    - Flow-through boundary/surface: momentum flux, pressure force, mass flow, average density, velocity, static and total pressure, static and total temperature, Mach number...
  - Output (Tecplot) and auxiliary diagnostic output
- What you will not learn
  - Tecplot usage
- What should you already know
  - Basic flow solver operation and control

Background

- **[project_name]_hist.dat** tracks the summation of all viscous solid walls
- **[project_name].forces** provides ***final*** force and moment for each boundary
- Need for calculating a time history of flow-through surface characteristics, such as average pressure, density, mass-flow, momentum flux, etc
- Need for mixing and matching different combinations of solid boundaries and flow-through surfaces for installation performance and propulsion-airframe-integration (PAI) studies.
- Need for extra diagnostic and analysis information for propulsion simulations: discharge coefficient, thrust ratio, ARP 1420, etc.

Files

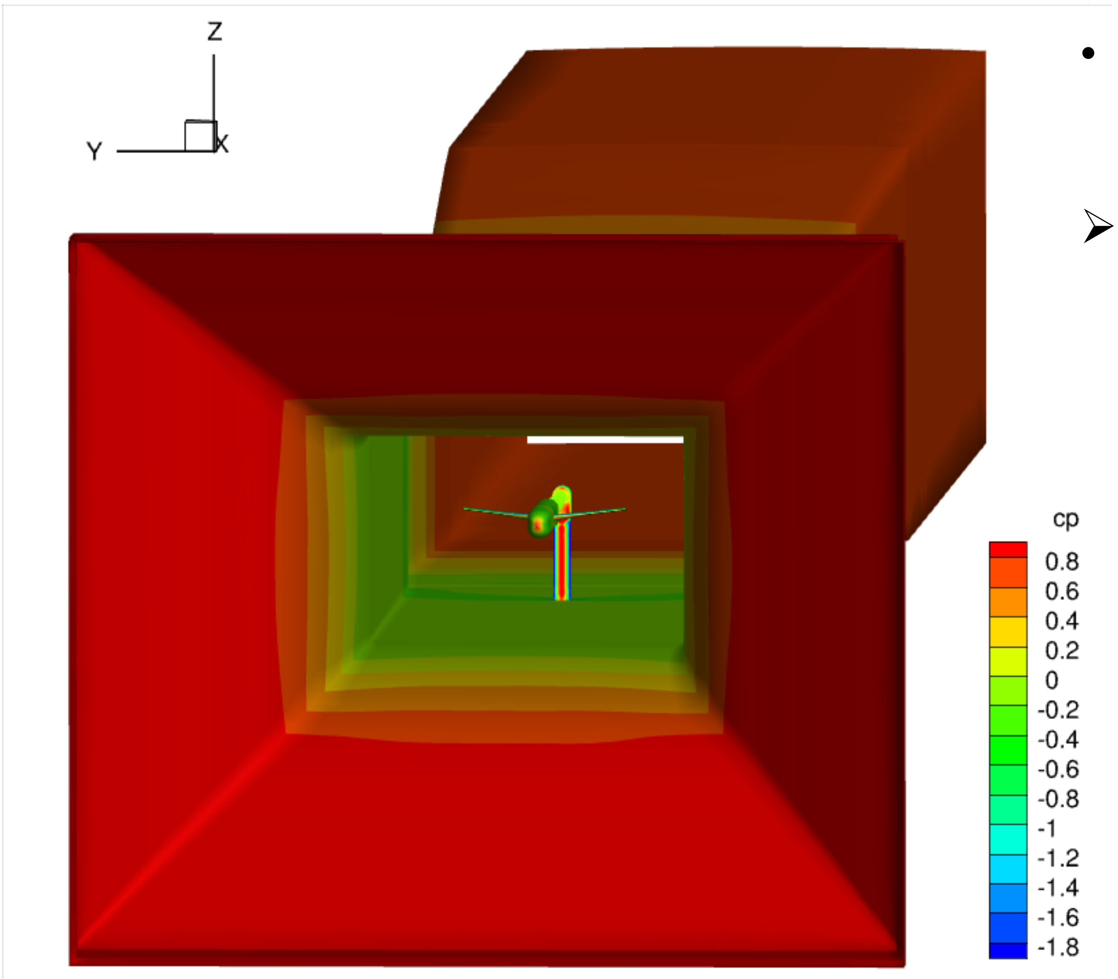
Input deck: fun3d.nml

- The user is required to supply an input deck for FUN3D named `fun3d.nml` (fixed name)
- &sampling_parameters
- &sampling_output
- **&component_parameters**

Output

- **[project_name]_fm_[component_name].dat**
- **[project_name]_stream_info.dat**

Example 1 – Model in Tunnel



- Multiple length scales
- Multiple time scales
- Monitor several aspects of the tunnel and model F&M

➢ Say I want to monitor mass flow, mass flow conservation and model lift and drag to evaluate solution convergence.

Example 1 – Model in Tunnel

```
$ cat mit.mapbc
  13 aflr3/ugrid sio fun3d bc types
   1         4000 Inlet_Front
   2         4000 Inlet_Contraction
   3         4000 Test_Section
   4         4000 Diffuser_Part1
   5         4000 Diffuser_Part2
   6         4000 Diffuser_Extension
   7         4000 Fuselage
   8         4000 PortWing
   9         4000 StarboardWing
  10         4000 Sting
  11         4000 Mast
  12         7011 BC_Inlet
  13         5051 Outflow

&boundary_conditions
    total_pressure_ratio(12) = 1.0252288
 total_temperature_ratio(12) = 1.0071442
   static_pressure_ratio(13) = 1.0181197
                  grid_units = 'feet'
/
```

```
&component_parameters
    allow_flow_through_forces = .true.
    list_forces               = .true.
    number_of_components = 4

    component_count(1) = 1
    component_input(1) = '12'
     component_name(1) = 'Inflow'

    component_count(2) = 1
    component_input(2) = '13'
     component_name(2) = 'Outflow'

    component_count(3) = 2
    component_input(3) = '12,13'
     component_name(3) = 'Total'

    component_count(4) = -1
    component_input(4) = '7,8,9'
     component_name(4) = 'model'
 /
```

This will result in 4 ASCII Tecplot force and moment history files:

```
mit_fm_inflow.dat
mit_fm_outflow.dat
mit_fm_total.dat
mit_fm_model.dat
```

Example 1 – Model in Tunnel

```
$ cat mit.mapbc
  13 aflr3/ugrid sio fun3d bc types
   1         4000 Inlet_Front
   2         4000 Inlet_Contraction
   3         4000 Test_Section
   4         4000 Diffuser_Part1
   5         4000 Diffuser_Part2
   6         4000 Diffuser_Extension
   7         4000 Fuselage
   8         4000 PortWing
   9         4000 StarboardWing
  10         4000 Sting
  11         4000 Mast
  12         7011 BC_Inlet
  13         5051 Outflow

&boundary_conditions
    total_pressure_ratio(12) = 1.0252288
 total_temperature_ratio(12) = 1.0071442
   static_pressure_ratio(13) = 1.0181197
                  grid_units = 'feet'
/
```

```
&component_parameters
    allow_flow_through_forces = .true.
    list_forces               = .true.
    number_of_components = 4

    component_count(1) = 1
    component_input(1) = '12'
     component_name(1) = 'Inflow'

    component_count(2) = 1
    component_input(2) = '13'
     component_name(2) = 'Outflow'

    component_count(3) = 2
    component_input(3) = '12,13'
     component_name(3) = 'Total'

    component_count(4) = -1
    component_input(4) = '7,8,9'
     component_name(4) = 'model'
 /
```

This will result in 4 ASCII Tecplot force and moment history files:

```
mit_fm_inflow.dat
mit_fm_outflow.dat
mit_fm_total.dat
mit_fm_model.dat
```

Example 1 – Model in Tunnel

```
$ cat mit.mapbc
  13 aflr3/ugrid sio fun3d bc types
   1         4000 Inlet_Front
   2         4000 Inlet_Contraction
   3         4000 Test_Section
   4         4000 Diffuser_Part1
   5         4000 Diffuser_Part2
   6         4000 Diffuser_Extension
   7         4000 Fuselage
   8         4000 PortWing
   9         4000 StarboardWing
  10         4000 Sting
  11         4000 Mast
  12         7011 BC_Inlet
  13         5051 Outflow

&boundary_conditions
    total_pressure_ratio(12) = 1.0252288
 total_temperature_ratio(12) = 1.0071442
   static_pressure_ratio(13) = 1.0181197
                  grid_units = 'feet'
/
```

```
&component_parameters
    allow_flow_through_forces = .true.
    list_forces               = .true.
    number_of_components = 4

    component_count(1) = 1
    component_input(1) = '12'
     component_name(1) = 'Inflow'

    component_count(2) = 1
    component_input(2) = '13'
     component_name(2) = 'Outflow'

    component_count(3) = 2
    component_input(3) = '12,13'
     component_name(3) = 'Total'

    component_count(4) = -1
    component_input(4) = '7,8,9'
     component_name(4) = 'model'
/
```

This will result in 4 ASCII Tecplot force and moment history files:

mit_fm_inflow.dat
mit_fm_outflow.dat
mit_fm_total.dat
mit_fm_model.dat

Example 1 – Model in Tunnel

```
$ cat mit.mapbc
  13 aflr3/ugrid sio fun3d bc types
   1       4000 Inlet_Front
   2       4000 Inlet_Contraction
   3       4000 Test_Section
   4       4000 Diffuser_Part1
   5       4000 Diffuser_Part2
   6       4000 Diffuser_Extension
   7       4000 Fuselage
   8       4000 PortWing
   9       4000 StarboardWing
  10       4000 Sting
  11       4000 Mast
  12       7011 BC_Inlet
  13       5051 Outflow
```

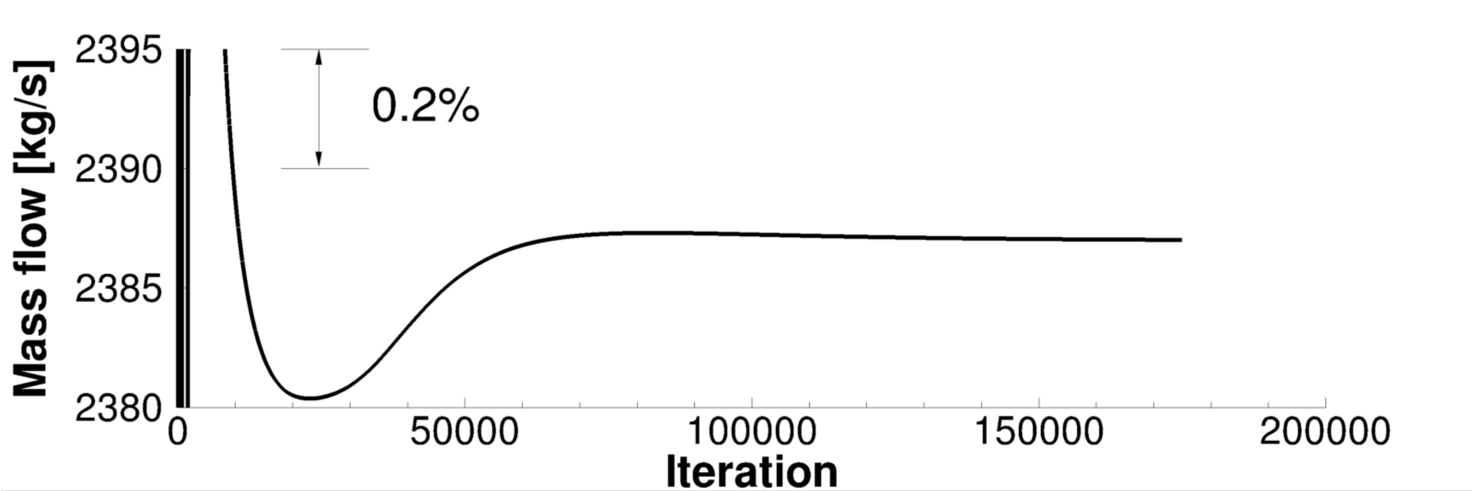
```
&boundary_conditions
    total_pressure_ratio(12) = 1.0252288
 total_temperature_ratio(12) = 1.0071442
   static_pressure_ratio(13) = 1.0181197
                  grid_units = 'feet'
/
```

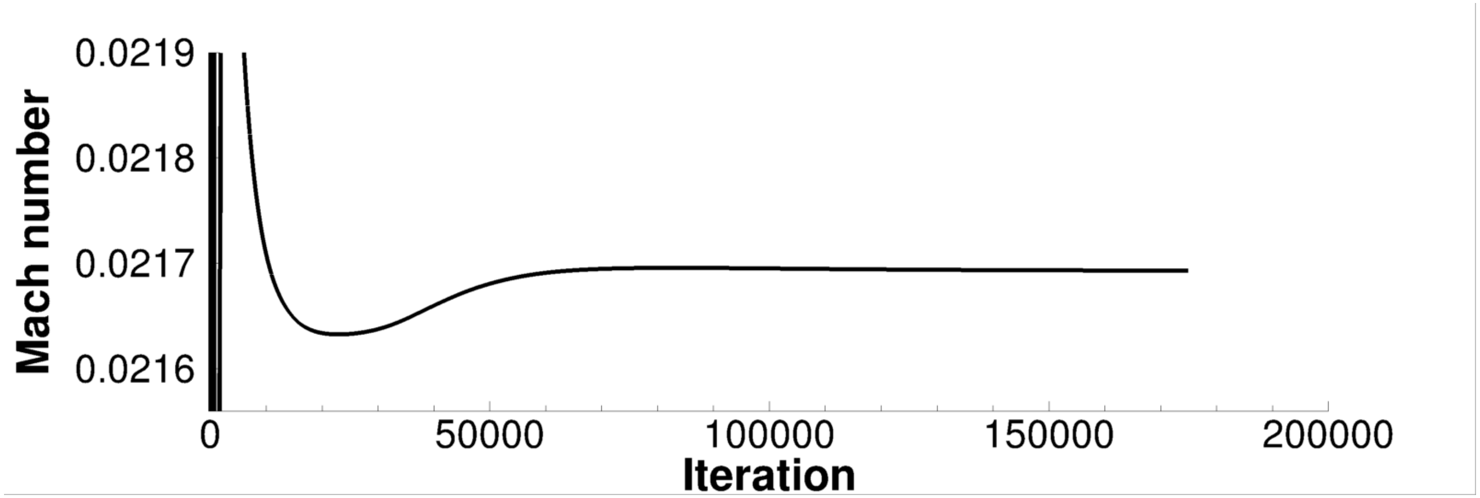
```
&component_parameters
    allow_flow_through_forces = .true.
    list_forces               = .true.
    number_of_components = 4

    component_count(1) = 1
    component_input(1) = '12'
     component_name(1) = 'Inflow'

    component_count(2) = 1
    component_input(2) = '13'
     component_name(2) = 'Outflow'

    component_count(3) = 2
    component_input(3) = '12,13'
     component_name(3) = 'Total'

    component_count(4) = -1
    component_input(4) = '7,8,9'
     component_name(4) = 'model'
 /
```

This will result in 4 ASCII Tecplot force and moment history files:

```
mit_fm_inflow.dat
mit_fm_outflow.dat
mit_fm_total.dat
mit_fm_model.dat
```

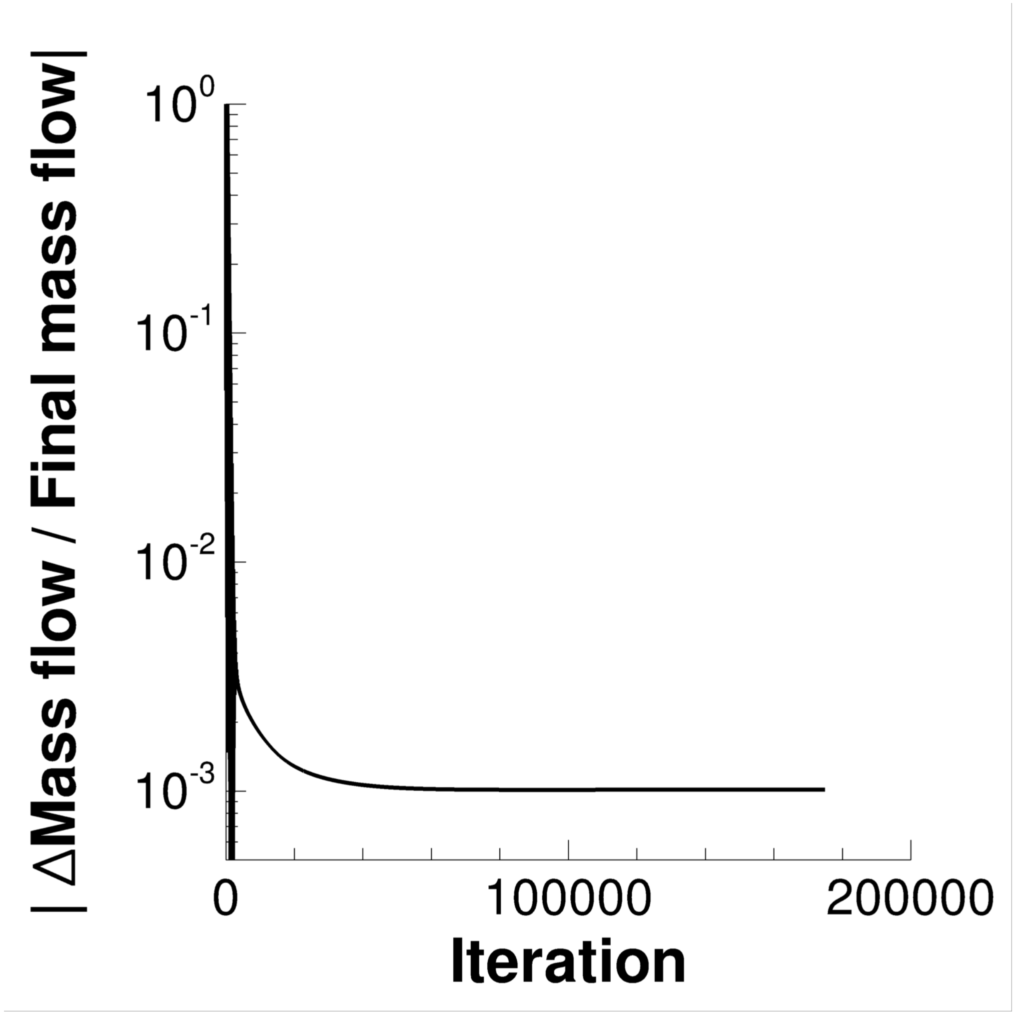
Example 1 – Model in Tunnel
`mit_fm_inflow.dat`



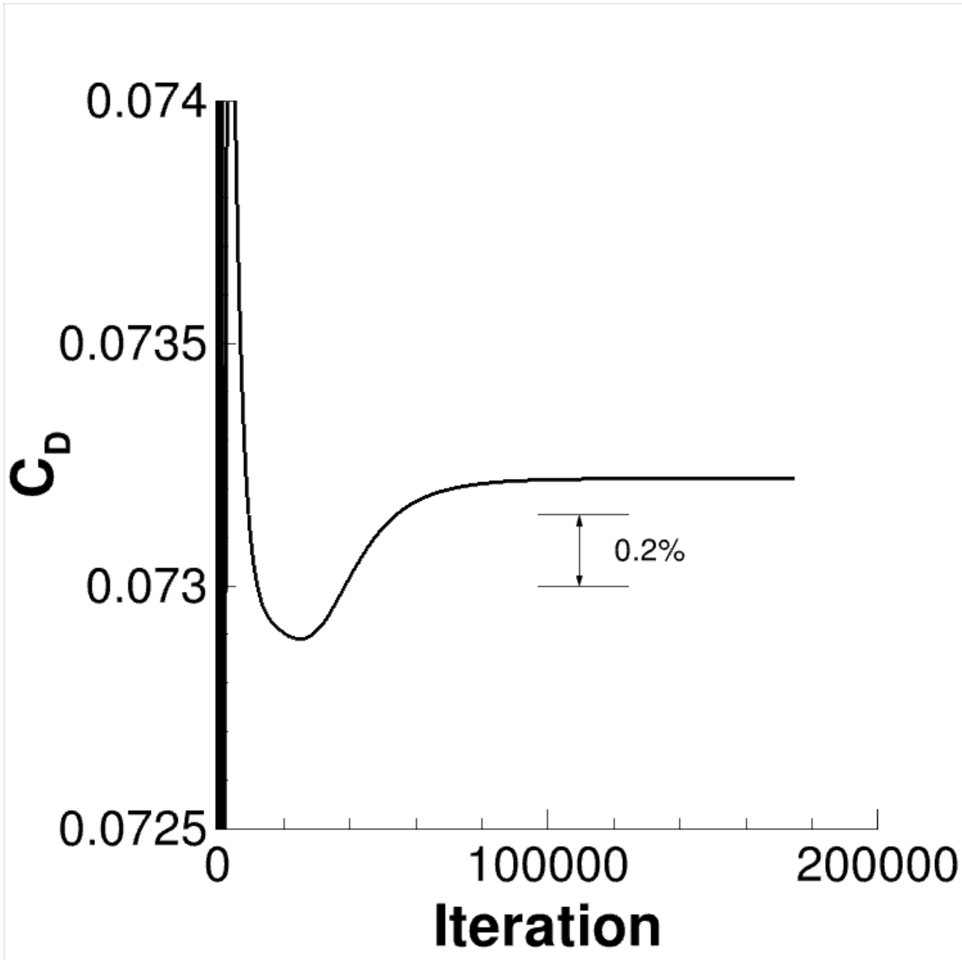


Example 1 – Model in Tunnel

mit_fm_total.dat



Example 1 – Model in Tunnel
`mit_fm_model.dat`



Example 1 – Model in Tunnel
mit_stream_info.dat

Contains averaged flow data and forces and moments in mesh, MKS and Imperial units.

```
Component/stream information for mit

Reynolds #   =  1.3129E+06
Mach         =        0.19
Iteration    =  175000
grid_units   = feet
Tref         =       519.9 [R]
T_infinity   =       288.8 [K]
             =       519.9 [R]
r_infinity   =      1.1992 [kg/m^3]
             =      0.0749 [lbm/ft^3]
             =    0.002327 [slug/ft^3]
p_infinity   =     99431.9 [Pa]
             =      14.421 [lbf/sq.in.]
             =    2076.678 [lbf/sq.ft.]
a_infinity   =      340.70 [m/s]
             =     1117.79 [ft/s]
u_infinity   =       64.39 [m/s]
             =      211.26 [ft/s]
q_infinity   =      2486.3 [Pa]
             =         0.4 [lbf/sq.in.]
             =        51.9 [lbf/sq.ft.]
pt ref       =    101940.4 [Pa]
Tt ref       =       290.9 [K]
```

Example 1 – Model in Tunnel

mit_stream_info.dat

Component 1 information:

```
              Inflow

component_type       (   1)    = boundary
component_symmetry (   1)    =      1.0
shape direction      (   1)    =     0.000     0.000     0.000
shape area                     =     2838.48925 mesh^2,      263.70428 [m^2],      2838.48925 [ft^2]
shape area * symmetry          =     2838.48925 mesh^2,      263.70428 [m^2],      2838.48925 [ft^2]
average density                =          1.018 [Fun3d],        1.2204 [kg/m^3],      0.07619 [lbm/ft^3]
average velocity               =          0.022 [Fun3d],        7.4169 [m/s],          24.334 [ft/s]
average u velocity             =          0.022 [Fun3d],        7.4170 [m/s],          24.334 [ft/s]
average v velocity             =         -0.000 [Fun3d],       -0.0177 [m/s],          -0.058 [ft/s]
average w velocity             =          0.000 [Fun3d],        0.0160 [m/s],           0.052 [ft/s]
average static pressure        =          0.732 [Fun3d],    101902.920 [Pa],           14.780 [psi]
average total pressure         =          0.732 [Fun3d],    101937.303 [Pa],           14.785 [psi]
minimum total pressure         =          0.732 [Fun3d],    101865.816 [Pa],           14.774 [psi]
maximum total pressure         =          0.733 [Fun3d],    101997.086 [Pa],           14.793 [psi]
average total enthalpy         =          0.000 [Fun3d],         0.000 [J]
average entropy                =          0.000 [Fun3d],         0.000 [J/K]
average static temperature =          1.007 [Fun3d],       290.877 [K],           523.579 [R]
average total temperature  =          1.007 [Fun3d],       290.905 [K],           523.629 [R]
 -------------------------------------------
mass flow                      =       62.88518 [Fun3d],    2387.01562 [kg/s],     5262.46863 [lbm/s]
```

Example 2 – Nozzle Performance

ASME calibration nozzle

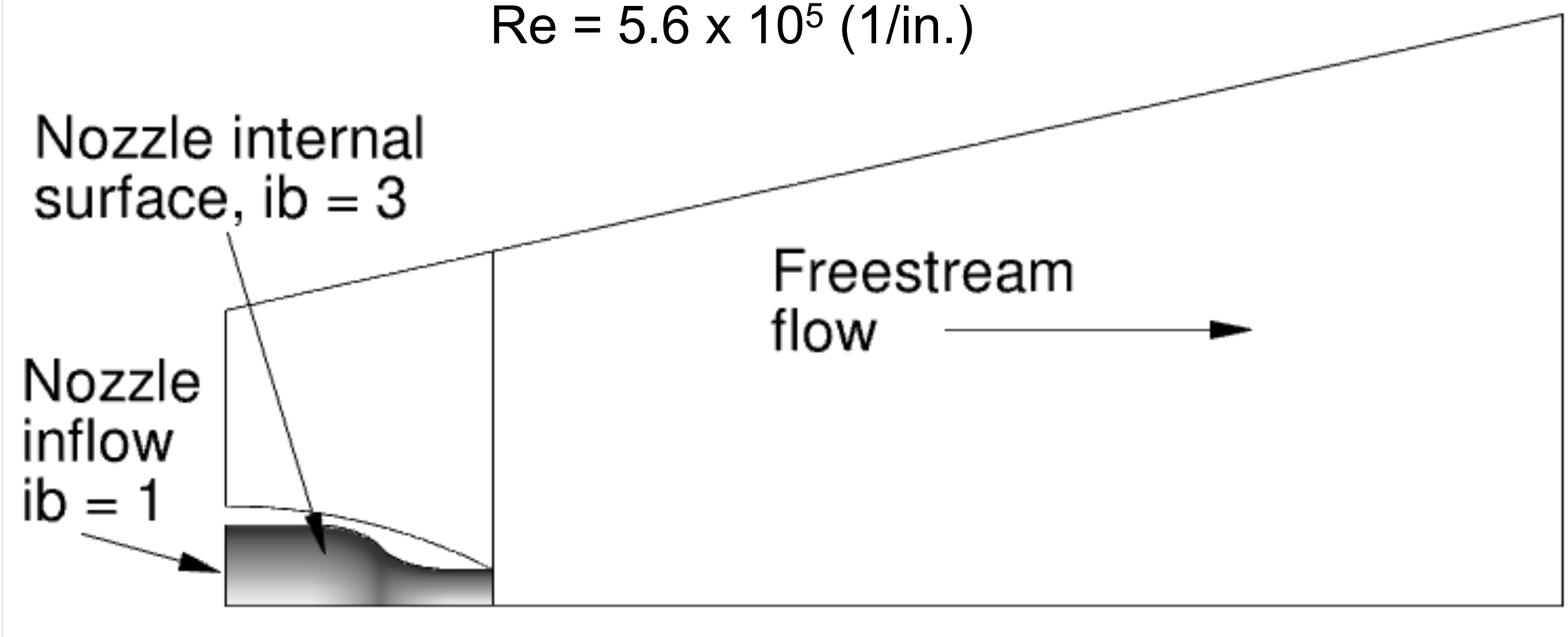
A_{throat} = 19.6349 sq.in.

NPR = $p_{total,jet}/p_{infinity}$ = 2.0

NTR = $T_{total,jet}/T_{infinity}$ = 1.0

$Mach_{reference}$ = 0.5

Re = 5.6 x 10^5 (1/in.)



NPR – nozzle pressure ratio
NTR – nozzle temperature ratio

Example 2 – Nozzle Performance

```
&component_parameters
     number_of_components = 4
        area_reference_ct = 864. ! Aerodynamic units

         component_count(1) = 1
         component_input(1) = '3'
          component_name(1) = 'core'
      component_symmetry(1) = 4.

         component_count(2) = 1
         component_input(2) = '2'
          component_name(2) = 'shell'
      component_symmetry(2) = 4.

         component_count(3) = 2
         component_input(3) = '2,3'
          component_name(3) = 'total'
      component_symmetry(3) = 4.       Quarter plane symmetry
            calculate_cd(3) = T        Calculate discharge coefficient
             throat_area(3) = 19.5087  Throat area for Cd ( actual grid area )
                     npr(3) = 2.00     The set nozzle pressure ratio
                     ntr(3) = 1.0      The set nozzle temperature ratio
  calculate_thrust_ratio(3) = T        Calculate thrust ratio

         component_input(4) = '0'
          component_type(4) = 'circle'
          component_name(4) = 'survey'
     circle_center(1:3,4) = 7.999, 0.0, 0.0
     circle_normal(1:3,4) = 1.0, 0.0, 0.0
         circle_radius(4) = 2.5
    component_symmetry(4) = 4.
          calculate_cd(4) = T
           throat_area(4) = 19.5087
                   npr(4) = 2.00
                   ntr(4) = 1.0
calculate_thrust_ratio(4) = T

  allow_flow_through_forces = T
  list_forces               = T
/
```

Same format as sampling.

Example 2 – Nozzle Performance

asme_stream_info.dat

component = 3

```
                 total

component_type      (   3)   = boundary
component_symmetry  (   3)   =      4.0
shape direction     (   3)   =     0.000     0.000     0.000
shape area                   =       24.60585 mesh^2,        0.01587 [m^2],        0.17087 [ft^2]
shape area * symmetry        =       98.42340 mesh^2,        0.06350 [m^2],        0.68350 [ft^2]
average density              =          2.839 [Fun3d],        3.3276 [kg/m^3],     0.20774 [lbm/ft^3]
average velocity             =          0.080 [Fun3d],       27.7654 [m/s],         91.094 [ft/s]
average u velocity           =          0.119 [Fun3d],       41.3750 [m/s],        135.745 [ft/s]
average v velocity           =         -0.000 [Fun3d],       -0.0111 [m/s],         -0.036 [ft/s]
average w velocity           =         -0.000 [Fun3d],       -0.0111 [m/s],         -0.036 [ft/s]
average static pressure      =          1.414 [Fun3d],    199882.299 [Pa],          28.990 [psi]
average total pressure       =          1.429 [Fun3d],    201881.394 [Pa],          29.280 [psi]
minimum total pressure       =          1.415 [Fun3d],    199932.300 [Pa],          28.998 [psi]
maximum total pressure       =          1.430 [Fun3d],    202112.696 [Pa],          29.314 [psi]
average total enthalpy       =          0.000 [Fun3d],         0.000 [J]
average entropy              =          0.000 [Fun3d],         0.000 [J/K]
average static temperature   =          0.698 [Fun3d],       209.255 [K],          376.658 [R]
average total temperature    =          1.000 [Fun3d],       300.005 [K],          540.009 [R]
```

Example 2 – Nozzle Performance

asme_stream_info.dat

component = 3 (continued)

```
 -----------------------------------------
mass flow                  =      22.34347 [Fun3d],      5.86688 [kg/s],      12.93425 [lbm/s]
Throat area                =      19.50870 [Fun3d],      0.01259 [m^2],        0.13548 [ft^2]
Ideal mass flow            =      22.57958 [Fun3d],      5.92887 [kg/s]       13.07093 [lbm/s]
Discharge coefficient      =         0.990
 -----------------------------------------
Axial

X-momentum flux           =       2.66329 [Fun3D],      242.818 [N],          54.58759 [lbf]
Y-momentum flux           =      -0.00071 [Fun3D],       -0.065 [N],          -0.01456 [lbf]
Z-momentum flux           =      -0.00071 [Fun3D],       -0.065 [N],          -0.01458 [lbf]
X-pressure flux           =      66.10880 [Fun3D],     6027.279 [N],        1354.98631 [lbf]
Y-pressure flux           =       0.00000 [Fun3D],        0.000 [N],           0.00000 [lbf]
Z-pressure flux           =       0.00000 [Fun3D],        0.000 [N],           0.00000 [lbf]
Viscous forces            =       2.49887 [Fun3D],      227.827 [N],          51.21752 [psf]
Pressure forces           =      18.59011 [Fun3D],     1694.900 [N],         381.02873 [psf]
Total forces              =      21.08898 [Fun3D],     1922.727 [N],         432.24625 [psf]

Ideal thrust              =      21.17706 [Fun3D],     1930.757 [N],         434.05147 [psf]
Thrust ratio              =         0.996
coefficient area          =     864.00000 mesh^2,       0.55742 [m^2]
p_t A*                    =      1234.286 [Fun3D],   112532.450 [N],   25298.30114 [psf],
Thrust coefficient        =         0.017
```

Example 2 – Nozzle Performance

asme_stream_info.dat

component = 4

```
               survey

component_type     (   4)   = circle
component_symmetry (   4)   =      4.0
shape direction    (   4)   =     1.000     0.000     0.000
shape area                  =        4.87718 mesh^2,        0.00315 [m^2],        0.03387 [ft^2]
shape area * symmetry       =       19.50872 mesh^2,        0.01259 [m^2],        0.13548 [ft^2]
average density             =          1.237 [Fun3d],        1.4495 [kg/m^3],     0.09049
[lbm/ft^3]
average velocity            =          0.924 [Fun3d],      320.8319 [m/s],       1052.598 [ft/s]
average u velocity          =          0.923 [Fun3d],      320.4375 [m/s],       1051.304 [ft/s]
average v velocity          =          0.000 [Fun3d],        0.0000 [m/s],          0.000 [ft/s]
average w velocity          =          0.000 [Fun3d],        0.0000 [m/s],          0.000 [ft/s]
average static pressure     =          0.733 [Fun3d],    103561.733 [Pa],          15.020 [psi]
average total pressure      =          1.415 [Fun3d],    199969.692 [Pa],          29.003 [psi]
minimum total pressure      =          0.725 [Fun3d],    102399.840 [Pa],          14.852 [psi]
maximum total pressure      =          1.430 [Fun3d],    202026.637 [Pa],          29.301 [psi]
average total enthalpy      =          2.503 [Fun3d],    301727.286 [J]
average entropy             =         -0.271 [Fun3d],      -109.027 [J/K]
average static temperature  =          0.830 [Fun3d],       248.905 [K],          448.029 [R]
average total temperature   =          1.001 [Fun3d],       300.320 [K],          540.576 [R]
```

Example 2 – Nozzle Performance

asme_stream_info.dat

component = 4 (continued)

```
 ----------------------------------------
mass flow                =       22.29055 [Fun3d],       5.85298 [kg/s],       12.90361 [lbm/s]
Throat area              =       19.50870 [Fun3d],       0.01259 [m^2],         0.13548 [ft^2]
Ideal mass flow          =       22.57958 [Fun3d],       5.92887 [kg/s]        13.07093 [lbm/s]
Discharge coefficient    =          0.987
 ----------------------------------------
Axial

X-momentum flux         =       20.66137 [Fun3D],      1883.741 [N],         423.48180 [lbf]
Y-momentum flux         =        0.02509 [Fun3D],         2.288 [N],           0.51429 [lbf]
Z-momentum flux         =        0.02508 [Fun3D],         2.287 [N],           0.51414 [lbf]
X-pressure flux         =        0.36182 [Fun3D],        32.988 [N],           7.41602 [lbf]
Y-pressure flux         =        0.00000 [Fun3D],         0.000 [N],           0.00000 [lbf]
Z-pressure flux         =        0.00000 [Fun3D],         0.000 [N],           0.00000 [lbf]
Viscous forces          =       20.66137 [Fun3D],      1883.741 [N],         423.48180 [psf]
Pressure forces         =        0.36182 [Fun3D],        32.988 [N],           7.41602 [psf]
Total forces            =       21.02319 [Fun3D],      1916.729 [N],         430.89783 [psf]

Ideal thrust            =       21.12690 [Fun3D],      1926.184 [N],         433.02346 [psf]
Thrust ratio            =          0.995
coefficient area        =      864.00000 mesh^2,        0.55742 [m^2]
p_t A*                  =       1234.286 [Fun3D],    112532.450 [N],    25298.30114 [psf],
Thrust coefficient      =          0.017
```

ARP 1420

Volume rake

Input for an inlet ARP 1420 analysis along with integrated flow data of circle.

```
             component_count(2) = 1
             component_input(2) = '0' ! No boundary associated with this input
              component_type(2) = 'circle' ! Add sampling information to AIP raking
          circle_center(1:3,2) = 2.995, 0.0, 0.0
          circle_normal(1:3,2) = 1.0, 0.0, 0.0
              circle_radius(2) = 1.0
              component_name(2) = 'Inlet-2'
         component_symmetry(2) = 1.0
calculate_arp1420_distortion(2) = .true.
   inlet_distortion_boundary(2) = 0
            number_of_rakes(2) = 8
            number_of_rings(2) = 5

rake_points(2,1:3,1,1)=2.995000,0.000000,0.316228
rake_points(2,1:3,2,1)=2.995000,0.000000,0.547723
rake_points(2,1:3,3,1)=2.995000,0.000000,0.707107
rake_points(2,1:3,4,1)=2.995000,0.000000,0.836660
rake_points(2,1:3,5,1)=2.995000,0.000000,0.948683

etc.
```

ARP 1420
Boundary rake

Input for an inlet ARP 1420 analysis on an outflow boundary.

```
                  component_count(1) = 1
                  component_input(1) = '1' ! Print out flowfield information for boundary
                   component_name(1) = 'Inlet'
               component_symmetry(1) = 1.0
     calculate_arp1420_distortion(1) = .true.
        inlet_distortion_boundary(1) = 1    ! Find rake data on this boundary
                  number_of_rakes(1) = 8
                  number_of_rings(1) = 5

     rake_points(1,1:3,1,1)=3.000000,0.000000,0.316228
     rake_points(1,1:3,2,1)=3.000000,0.000000,0.547723
     rake_points(1,1:3,3,1)=3.000000,0.000000,0.707107
     rake_points(1,1:3,4,1)=3.000000,0.000000,0.836660
     rake_points(1,1:3,5,1)=3.000000,0.000000,0.948683

     etc…
```

What Could Possibly Go Wrong?

In General…

- Do not hesitate to send questions to *fun3d-support@lists.nasa.gov* ; we are happy to try to diagnose problems
  - Please send as much information about the problem/inputs/environment that you can, as well as all screen output and any error output
  - In extreme cases, we may request your grid and attempt to run a case for you to track down the problem
  - If you cannot send us a case due to restrictions, size, etc., a generic/smaller representative case that behaves similarly can be useful
  - Check the manual for guidance
- Ask the FUN3D user community, *fun3d-users@lists.nasa.gov*

What We Learned

- Overview of visualization output options and examples
- Overview of component F&M tracking and example

Don’t hesitate to send questions our way!

fun3d-support@lists.nasa.gov