Session 5:
Turbulent Flow Simulations

Eric Nielsen

(Jan-Renee Carlson and Chris Rumsey)

Session Overview

- Turbulent flow simulations
  - Available turbulence models
  - Namelist parameters
  - Typical simulations
  - Default values in namelists are green
  - Optional namelist arguments are in blue

Turbulent Flow Simulations

```
&governing_equations
   eqn_type      = "cal_per_compress"
   viscous_terms = "inviscid"
   viscous_terms = "laminar"
   viscous_terms = "turbulent"
/
```

Turbulent Flow Simulations

```
&turbulent_diffusion_models
   turb_model                  ="sa"          (Spalart-Allmaras)
   turb_model                  ="menter-sst"  (k-ω—SST)
   turb_model                  ="des"         (Detached Eddy simulation )
   turb_model                  ="hrles"       (Hybrid RANS-LES)
   re_stress_model             ="linear"
   turb_compress_model         ="off"
   prandtlnumber_turbulent = 0.9
/
```

Turbulent Flow Simulations
S-A

```
&turbulent_diffusion_models
  turb_model           ="sa"
  turb_compress_model ="none"
  turb_compress_model ="ssz"
/
&spalart
  turbinf = 1.3
/
```

`turb_compress_model ="ssz"` (AIAA-95-0863, Shur et al.)

`turbinf = 1.3` (default makes $\nu_{t,\infty} = 0.009$)

Turbulent Flow Simulations
Menter-SST

```
&turbulent_diffusion_models
   turb_model            ="menter-sst"
   turb_compress_model ="none"
   turb_compress_model ="wilcox"   (Wilcox compressibility)
/
&kw_sst
   strain_production =.false.  (Default uses vorticity)
   strain_production =.true.  (strain rate tensor)
   k_inf             = 9.0e-9
   w_inf             = 1.0e-6
/
```

Turbulent Flow Simulations

```
&turbulent_diffusion_models
  turb_model              ="des"
  turb_model              ="hrles"
/
```

- Defer to after-session discussion as this topic is too extensive for the time allocated

Typical Usages

- RANS wall bounded flow simulation using Spalart-Allmaras (part 1)

```
&version_number
    input_version             = 2.0
/
&project
    project_rootname          = "bump_3levelsdown_177x81"
/
&reference_physical_properties
    mach_number               = 0.2
    reynolds_number           = 3000000.0
    temperature               = 540.0
    temperature_units         = "Rankine"
/
&turbulent_diffusion_models
    turb_model                = "sa"
/
&spalart
    turbinf                   = 3.0
/
&nonlinear_solver_parameters
    schedule_iteration        = 1      250
    schedule_cfl              = 10.    250.
    schedule_cflturb          = 10.    250.
/
&code_run_control
    steps                     = 20000
    restart_write_freq        = 1000
    restart_read              = "off"
/
```

Typical Usages

- RANS wall bounded flow simulation using Spalart-Allmaras (part 2)

```
&boundary_conditions
    total_pressure_ratio(3)    = 1.02828
    total_temperature_ratio(3) = 1.008
    static_pressure_ratio(4)   = 1.0
/
&boundary_output_variables
    mu_t  = .true.
/
```

Typical Usages

- RANS shear/Jet flow simulation using k-ω-SST (part 1)

```
&raw_grid
   grid_format = "vgrid"
   data_format = "unformatted"
 /
&project
   project_rootname = "arn2"
/
&inviscid_flux_method
   flux_limiter            = "none"
   flux_construction       = "roe"
/
&turbulent_diffusion_models
   turb_model           = "menter-sst"
/
&code_run_control
   steps              = 500
   restart_write_freq = 250
   restart_read       = "on"
/
&reference_physical_properties
  mach_number         =  0.50
  reynolds_number     =  0.5e+6
  temperature         =  300.0
/
```

Typical Usages

- RANS shear/Jet flow simulation using k-ω-SST (part 2)

```
&boundary_conditions
   total_pressure_ratio(1)    = 1.197
   total_temperature_ratio(1) = 0.950
   static_pressure_ratio(4)   = 1.0
/
 &nozzle_parameters
   inflow_pt_ramp=100,
/
 &sampling_parameters
   number_of_geometries = 1
   type_of_geometry(1)  = 'circle'
   circle_center(1,:)   = 0.,0.,0.
   circle_normal(1,:)   = 1.,0.,0.
   circle_radius(1)     = 5.
/
 &sampling_output_variables
   mach                 = .true.
   turb1                = .true.
   turb2                = .true.
   mu_t                 = .true.
/
```

Typical Usages

- RANS shear/Jet flow simulation using k-ω-SST (part 3)

```
&boundary_output_variables
   number_of_boundaries = 3
   boundary_list        = '1, 7, 8'
   mach                 = .true.
   turb1                = .true.
   turb2                = .true.
   mu_t                 = .true.
   yplus                = .true.
 /
 &nonlinear_solver_parameters
    schedule_iteration =  1 500
    schedule_cfl       =  1.0   50.0
    schedule_cflturb   =  1.0   50.0
 /
 &version_number
    input_version = 2.2
 /
```

- Execute statement

```
mpirun nodet_mpi                  \
   --animation_freq 100           \
   --sampling_freq 100            \
   --alternate_freestream 0.05    \
   > fun3d_output
```

Session 8 (cont.)
Boundary Conditions
(WE ARE THE PROBLEM!)

Eric Nielsen

(Jan-Renee Carlson)

(WE'RE PART OF THE SOLUTION TOO!)

Session Overview

- Boundary conditions
  - Available boundaries (slightly abridged)
  - Namelist parameters (optional/needed) (slightly abridged)
  - A typical(?) simulation

Boundary Conditions

	Grid	FUN3D	Type
`tangency`	5	3000	slip wall
`viscous_solid`	4	4000	no-slip wall
`symmetry_x,y,z`		6661, 6662, 6663	x-,y-,z-symmetry plane
`farfield_riem`	3	5000	Riemann
`farfield_roe`		5050	farfield
`back_pressure`		5051	outflow
`subsonic_outflow_p0`		7012	outflow
`subsonic_outflow_mach`		5052	outflow
`massflow_out`		7031	outflow
`massflow_in`		7036	inflow

Boundary Conditions

Grid	FUN3D	Type
subsonic_inflow_pt	7011	inflow
subsonic_inflow_vel	7010	inflow
extrapolate	5026	outflow
fixed_inflow	7100	inflow
fixed_outflow	7105	outflow

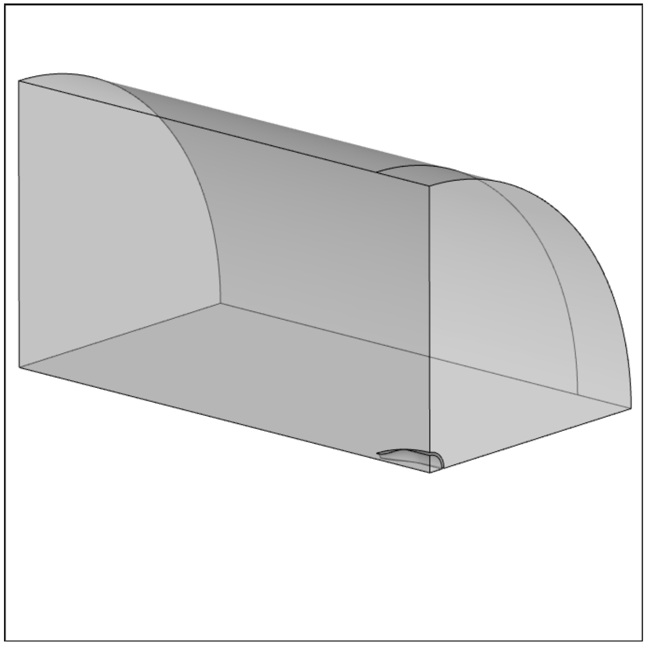
The Usual Suspects

- Farfield (typ. inflow) – **`farfield_riem`** (5000)
- Farfield (typ. outer boundary) – **`farfield_roe`** (5050)
- Wing/Body/Tail/Flate plate – **`viscous_solid`** (4000)
- Subsonic plenum / wind tunnel inflow – **`subsonic_inflow_pt`** (7011)
- Channel / wind tunnel outflow (allows supersonic flow)– **`back_pressure`** (5051)
- Supersonic inflow (e.g. nozzle exit face) – **`fixed_inflow`** (7100)
- Supersonic outflow – **`extrapolate`** (5026)
- Subsonic inlet (restricted to subsonic flow)– **`subsonic_outflow_p0`** (7012)

Sample Problem

Static test of an Acoustic Research Nozzle (ARN)

- Namelist settings
- Execution
- Output



FIGURES

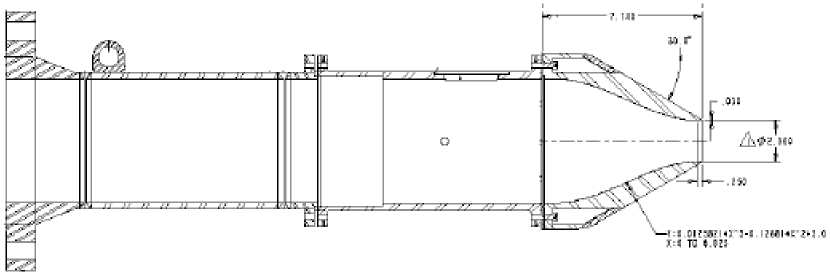


Figure 1 NASA Acoustic Reference Nozzle system, with ARN2 (51mm diameter) nozzle.

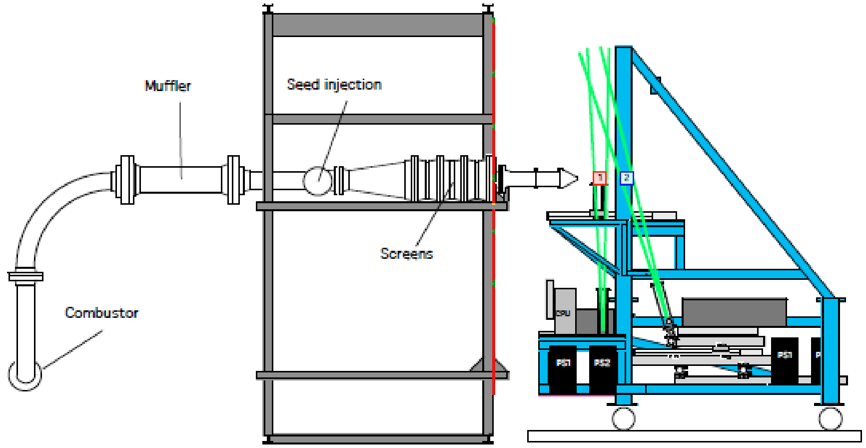


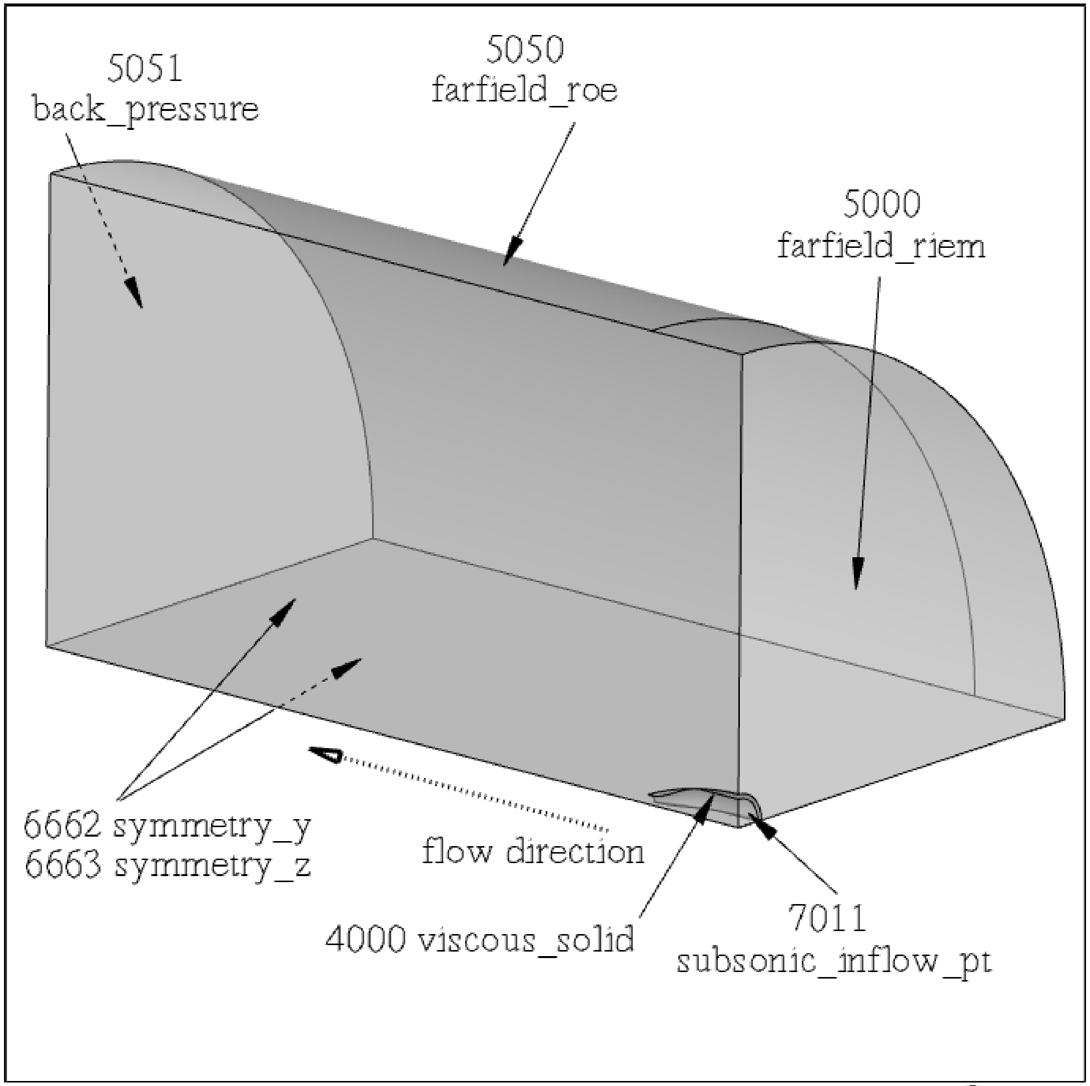
Figure 2 SHJAR with Dual PIV setup.

AIAA-2003-3130, Bridges

Sample Problem

- Boundaries

Acoustic Research Nozzle



Acoustic Research Nozzle

```
#Wed Jan 27 13:28:40 2010
#bc.map
Patch #      BC        Family    #surf    surfIDs       Family
#------------------------------------------------------------
-
1          7011          5           0        0     nozzleinflow
2          5050          3           0        0     farfield
3          5050          3           0        0     farfield
4          5051          3           0        0     outflow
5          4000          4           1       20     boattail
6          4000          4           1       19     nozzle
7          6663          1           0        0     z-symmetry
8          6662          1           0        0     y-symmetry
9          5000          3           0        0     inflow

 &boundary_conditions
   total_pressure_ratio(1)     = 1.197
   total_temperature_ratio(1) = 0.950
   static_pressure_ratio(4)    = 1.0
 /
```

Acoustic Research Nozzle

- fun3d.nml

```
&raw_grid
   grid_format = "vgrid"
   data_format = "unformatted"
  /
 &version_number
    input_version = 2.0
 /
 &project
    project_rootname = "arn2"
 /
 &governing_equations
    viscous_terms    = "turbulent"
 /
 &inviscid_flux_method
    flux_limiter            = "none"
    first_order_iterations =  1000
    flux_construction       = "roe"
 /
 &turbulent_diffusion_models
    turb_model            = "menter-sst"
 /
&code_run_control
    steps              = 5000
    restart_write_freq =  250
    restart_read       = "off"
 /
```

Acoustic Research Nozzle

- fun3d.nml (cont.)

```
&sampling_parameters
  number_of_geometries = 1
  type_of_geometry(1)  = 'circle'
  circle_center(1,:)   = 0.,0.,0.
  circle_normal(1,:)   = 1.,0.,0.
  circle_radius(1)     = 5.
/
&sampling_output_variables
   mach   = .true.
   turb1  = .true.
   turb2  = .true.
   mu_t   = .true.
/
&boundary_output_variables
   number_of_boundaries=1
   boundary_list="1"
   mach   = .true.
   turb1  = .true.
   turb2  = .true.
   mu_t   = .true.
/
```

Acoustic Research Nozzle

- fun3d.nml (cont.)

```
&reference_physical_properties
   temperature_units = "Kelvin"
   mach_number       =  0.50
   reynolds_number   =  0.5e+6
   temperature       =  300.0
/
&boundary_conditions
  total_pressure_ratio(1)    = 1.197
  total_temperature_ratio(1) = 0.950
  static_pressure_ratio(4)   = 1.0
/
&nozzle_parameters
  inflow_pt_ramp=100,
/
&nonlinear_solver_parameters
   time_accuracy      = "steady"
   schedule_number    =  2
   schedule_iteration =  1 500
   schedule_cfl       =  1.0  50.0
   schedule_cflturb   =  1.0  50.0
/
```

(Here Mach number and Reynolds number serve as the viscous scaling for the simulation since the jet is exhausting in to quiescent flow.)

Acoustic Research Nozzle

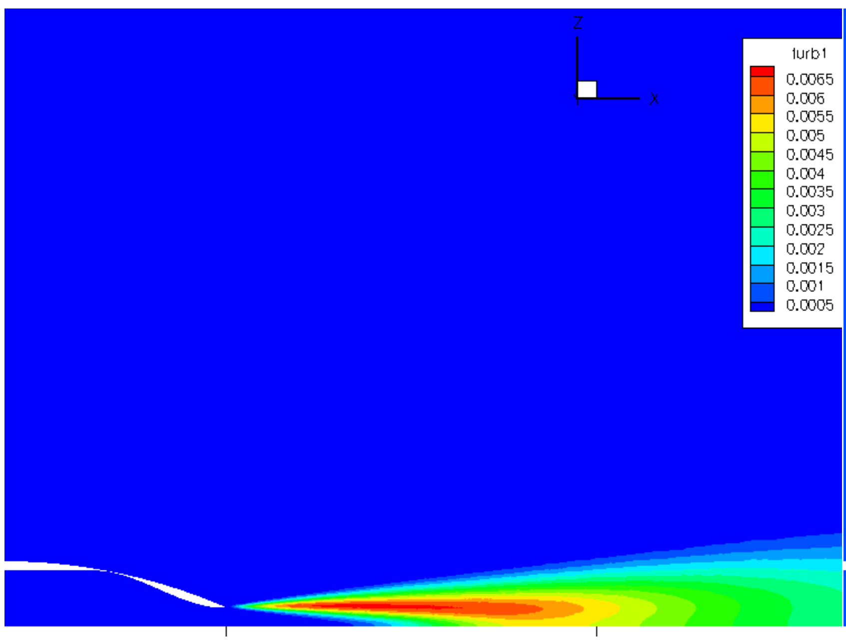
- Execution

```
~/mpirun nodet_mpi \
  --animation_freq 100          \
  --sampling_freq 100           \
  --alternate_freestream 0.05   \
  > fun3d_output
```

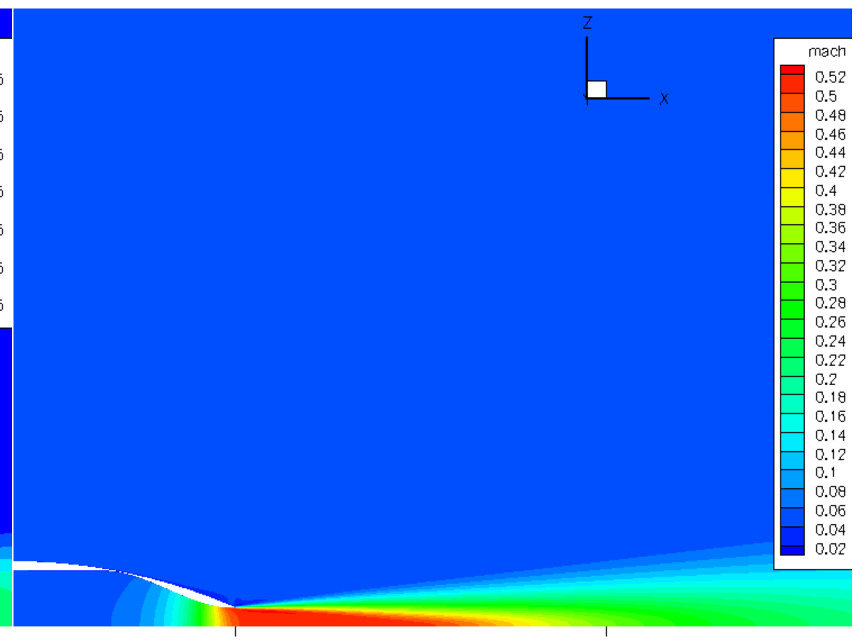
- Files output
  - history file: **arn2_hist.dat**
  - sampled data: **arn2_tec_sampling_geom1_timestepxxxx.dat**
  - flowfield data: **arn2_tec_boundary_timestepxxxx.dat**

Acoustic Research Nozzle

Turbulent kinetic energy



Mach number



Appendix: Boundary condition list

Boundary Conditions

&boundary_conditions namelist

Boundary	Input	Namelist parameter(s)
tangency	none	
viscous_solid	temperature (optional)	**wall_temperature(ib), wall_temp_flag(ib)**
symmetry_x,y,z	none	
farfield_riem, farfield_roe	none	
extrapolate	none	

Boundary Conditions

&boundary_conditions namelist

Boundary	Input	Namelist parameter(s)
back_pressure	static pressure (Mach > 0)	**static_pressure_ratio(ib)**
subsonic_outflow_p0	static pressure (Mach < 1)	**static_pressure_ratio(ib)**
subsonic_outflow_mach	Mach # (0 < Mach < 1)	**mach_bc(ib)**

Boundary Conditions

&boundary_conditions namelist

Boundary	Input	Namelist parameter(s)
massflow_out	massflow	**massflow(ib)**
massflow_in	mass flow, total temperature	**massflow(ib),** **total_temperature_ratio(ib)**

Boundary Conditions

&boundary_conditions namelist

Boundary	Input	Namelist parameter(s)
subsonic_inflow_pt	total pressure, total temperature, flow angle	**total_pressure_ratio(ib), total_temperature_ratio(ib), subsonic_inflow_velocity(ib) ="normal"**
subsonic_inflow_vel	density, velocity	**q_set(ib,1), q_set(ib,2:4)**

Boundary Conditions

`&boundary_conditions` namelist

Boundary	Input	Namelist parameter(s)
`fixed_inflow`	density, velocity & pressure (M > 1)	`q_set(ib,1:5)`
`fixed_outflow`	density, velocity & pressure (?)	`q_set(ib,1:5)`