Input File Description

1 Grid File

Input Variable	Type	Default Value	Note	Description
`grid_format`	`integer`	`1`	`\`	Specifies the format of the input grid file. • `1 := GMSH` • `2 := CGNS`
`gridfile`	`character(len=150)`	`""`	`\`	File path to the input grid file relative to the current directory.

Input Variable	Type	Default Value	Note	Description
metis_option	integer	3	\	Specifies the METIS partitioning algorithm to use for partitioning the global grid. ● 1 := Create non-weighted partitions based on a multilevel recursive bisection while minimizing the edge cut. For grids with origanized all quad or hex cells, I recommend this option. ● 2 := Create non-weighted partitions based on a multilevel kway algorithm while minimizing the edge cut. ● 3 := Create non-weighted partitions based on a multilevel kway algorithm while minimizing the total communication volume. ● 4 := Create weighted partitions based on a multilevel k-way algorithm while minimizing the total communication volume (weighting is not yet implemented so this is currently the same as option 3).

2 Physics and Equations

Input Variable	Type	Default Value	Note	Description
governing_equations	integer	2	\	Specifies the governing equations to use. ● 1 := Euler equations ● 2 := Navier-Stokes equations
machref	real	0.4	\	Reference Mach number.
gam	real	1.4	\	Reference γ (ratio of specific heats).
ptotref	real	-1.0	\	Reference total pressure (units of Pa).
ttotref	real	-1.0	\	Reference total temperature (units of Kelvin).
rhoref	real	-1.0	\	Reference static density (default evaluates to $1.160833 kg/m^3$)
tref	real	300.0	\	Reference static temperature (units of Kelvin).
pref	real	1E5	\	Reference static pressure (units of Pa).
rgasref	real	287.15	\	Reference gas constant (units of $m^2/(s^2 K)$).

Input Variable	Type	Default Value	Note	Description
alpha_aoaref	real	0.0	\	Reference angle of attack in the $x-y$ plane.
beta_aoaref	real	0.0	\	Reference angle of attack in the $x-z$ plane.
Pr	real	0.72	\	Laminar Prandtl number.
suth_muref	real	1.716E-5	\	Constant μ_0 in Sutherland's law.
suth_Tref	real	273.0	\	Constant T_0 in Sutherland's law.
suth_Sref	real	110.4	\	Constant S in Sutherland's law.

3 Base FR Configuration

Input Variable	Type	Default Value	Note	Description
solution_order	integer	3	\	Specifies the degree of the polynomial space, $\mathcal{P}_{\mathcal{S}}$, to use for the solution. NOTE: The order-of-accuracy is approximately $\mathcal{P}_{\mathcal{S}}+1$.
loc_solution_pts	integer	1	\	Specifies the location of the solution points used for tensor products. ● 1 := Legendre-Gauss nodes. ● 2 := Legendre-Gauss-Lobatto nodes.
invs_flux_method	integer	1	\	Specifies the approximate Riemann solver used to compute the common inviscid fluxes at the interfaces. ● 1 := Roe flux with entropy fix ● 2 := HLLC flux ● 3 := LDFSS flux ● 4 := Rotated-Roe-HLL flux
visc_flux_method	integer	2	\	Specifies the method used to compute the common viscous fluxes at the interfaces. ● 1 := Bassi-Rebay 1 (BR1) (WARNING: high chances of stability issues) ● 2 := Bassi-Rebay 2 (BR2) ● 3 := Local DG (LDG) (smaller CFL limit than BR2)

4 Time Scheme

Input Variable	Type	Default Value	Note	Description
Runge_Kutta_Scheme	integer	3	\	Specifies which Runge-Kutta method to use. ● 1 := Classic n-stage RK method ● 2 := 2-stage/2nd-order SSP-RK method ● 3 := 3-stage/3rd-order SSP-RK method ● 4 := 5-stage/4th-order SSP-RK method ● 5 := 5-stage/4th-order Carpenter-Kennedy low storage RK method
num_timesteps	integer	25000	\	Number of time steps for the simulation.
Final_Time	real	10.0	\	Stop when the time within the simulation reaches this value.
Timestep_Type	integer	1	\	Type of time stepping to use for the simulation. ● 0 := Constant global time step. ● 1 := global time step from minimum computed cell time step. ● -1 := local time stepping within each cell from computed cell time step. ● 2 := global time step from minimum computed solution point time step. (EXPERIMENTAL) ● -2 := local time stepping at each solution point from compute solution point time step. (EXPERIMENTAL)
constant_dt	real	1E-6	\	Size of the time step that is used when Timestep Type is set to a constant global time step.
CFL	real	0.1	\	Analogous to the CFL condition, this is used to scale the time step size to get a maximum stable time step. The maximum value this can be while also keeping a simulation stable is very dependent on the simulation/flow-conditions and input settings, and experimentation is generally required to find a good value. A lower value is more likely to keep a simulation stable, but it will require more time steps to reach either convergence or a given simulation time. Typical values are $0.1 \leq CFL \leq 2$, but past experience has found simulations that were stable with values up to 7.

5 Initialization

Input Variable	Type	Default Value	Note	Description
itestcase	integer	1	\	Specifies the test-case/problem to solve, primarily affecting the initialization of the simulation. ● 1 := Generic flow, initialize to reference conditions. ● 2 := Diagonally propagating Shu version of the isentropic Euler vortex. ● 3 := Channel flow (not working)
load_restart_file	logical	.FALSE.	\	Logical flag used to indicate whether or not to read in a restart file.
restart_file	character(len=150)	out.rst	\	File path to the solution restart file that will be read if load_restart_file is true.
restart_interval	integer	0	\	Number of time steps between writing restart files. NOTE: A restart file is always written after completing the last time step.
tiem_ave_file	character(len=150)	out.ave	\	File path to the time-averaged solution restart file that will be read in if load_restart_file and output_time_averaging are both true. The code will add the accumulation of time-averaged flow variables from the new running simulation to the time-averaged flow variables of the previous simulation from which the new simulation restarted.

6 Boundary Conditions

6.1 Base

Input Variable	Type	Default Value	Note	Description

Input Variable	Type	Default Value	Note	Description
bc_input	bc_input_t	\	\	Derived type to specify boundary conditions. An iterative loop will go through the specified flow conditions and try to compute the remaining unspecified flow conditions. NOTE: A run-time error will occur if either of the following conditions are true: A) The flow conditions are over-specified and inconsistent B) The flow conditions are under-specified and it is unable to compute the remaining unspecified flow conditions. EXAMPLE: An example of a well defined boundary condition using `bc_input`: ● `bc_input(1)%name = inflow_subsonic` ● `bc_input(1)%bc_type_string = SubInflow ! Subsonic Inflow BC` ● `bc_input(1)%p_static = 101325.0` ● `bc_input(1)%p_total = 121286.025` ● `bc_input(1)%t_total = 293.15` NOTE: You must specify all BCs which should be consistent with those in the grid file.
bc_input(1)%name	character(len=32)	""	\	Character string that matches the name of a boundary group in the grid file. NOTE: Case (upper/lower-case) does NOT matter.
bc_input(1)%bc_type_string	character(len=32)	""	\	The boundary conditions are specified in the grid file and the array `bface(1,:)` is updated accordingly. BC by this `bc_type_string` must be consistent with the grid BC. Because `bc_input` is used to set up `bc_in`. One of the usages of `bc_in` is to provide the flow condition for a BC. Valid values for boundary condition types are: ● Subsonic inflow : `SubInflow` ● No-slip adiabatic wall : `AdiabaticWall` ● Slip wall : `SlipWall`
bc_input(1)%p_static	real	-1.0	\	Static pressure (units of Pa).
bc_input(1)%t_static	real	-1.0	\	Static temperature (units of Kelvin).
bc_input(1)%rho_static	real	-1.0	\	Static density (units of kg/m^3)
bc_input(1)%mach	real	0.4	\	Mach number. (dimensionless)
bc_input(1)%vx	real	0.0	\	Velocity in the x coordinate direction (units of m/s)

Input Variable	Type	Default Value	Note	Description
bc_input(1)%vy	real	0.0	\	Velocity in the y coordinate direction (units of m/s)
bc_input(1)%vz	real	0.0	\	Velocity in the z coordinate direction (units of m/s)
bc_input(1)%p_total	real	-1.0	\	Total pressure (units of Pa).
bc_input(1)%t_total	real	-1.0	\	Total temperature (units of Kelvin).
bc_input(1)%rho_total	real	-1.0	\	Total density (units of kg/m^3)
bc_input(1)%alpha_aoa	real	0.0	\	Angle of attack in the $x-y$ plane. (dimensionless)
bc_input(1)%beta_aoa	real	0.0	\	Angle of attack in the $x-z$ plane. (dimensionless)
bc_input(1)%wall_temp	real	-1.0	\	Temperature for isothermal wall (units of Kelvin).
walls_are_exact	logical	.TRUE.	\	Logical flag that determines if the solution on wall boundary conditions are treated as either the exact flow conditions or the ghost state that equals the exact flow conditions when averaged with the interior state. ● .TRUE. := The wall boundary solution is the exact wall boundary condition, e.g., $[u_{wall}; v_{wall}; w_{wall}] = 0$ for a no-slip wall. ● .FALSE. := The average of the wall boundary solution and the interior solution on the boundary face recovers the exact wall boundary condition, e.g., $[u_{wall}; v_{wall}; w_{wall}] = -[u_{wall}; v_{wall}; w_{wall}]$ for a no-slip wall. NOTE: Hartmann (2014) reports that using exact wall boundary conditions seems to be more accurate on coarse grids and for lower values of solution order, whereas using the averaging method seems to be a little more stable. Details can be found at http://elib.dlr.de/90967/1/hartmann_leicht_VKI_LS_2014-3.pdf

Input Variable	Type	Default Value	Note	Description
sub_inflow_method	integer	1	\	Specifies which subsonic inflow boundary condition algorithm to use. ● 1 := Hold the total pressure and total temperature constant at the inflow, use the outgoing characteristic from the interior to perform a Newton iteration to get the static temperature, speed of sound, and normal velocity magnitude for the exterior state. ● 2 := Use the inflow static density and velocity, and the interior static pressure. ● 3 := Hold the inflow total pressure and total temperature constant, and use the interior velocity. ● 4 := Hold the inflow total pressure and total temperature constant, and use the interior static pressure. NOTE: It is highly recommended to use option 1.

6.2 Custom Profile

Input Variable	Type	Default Value	Note	Description
cpbc_prof_input	cpbc_prof_input_t	\	\	Input file for the custom profile BC. The first line of the input file is a comment line starting with !.
cpbc_prof_input%fname	character(len=256)	""	\	File name of the input file for the custom profile BC.
cpbc_prof_input%fdelim	character	,	\	Column delimiter in the input file for the custom profile BC.
cpbc_prof_input%dat_dir	character	""	\	The input profile is a 1D curve for each primitive variable. dat_dir indicates the direction along with the input profile is distributed. Valid values are X or Y. Z is always the spanwise direction in 3D. In 2D, there is no Z.

6.3 Inflow Turbulence Generation

Input Variable	Type	Default Value	Note	Description
tgbc_bg_grid_input	tgbc_bg_grid_input_t	\	\	Background mesh parameters for turbulence generation BC.

Input Variable	Type	Default Value	Note	Description
tgbc_bg_grid_input%loc_str	character	""	\	It indicates where the background mesh is located. loc_str=X indicates that the background mesh is located at a plane of $x = const$, i.e. $y - z$ plane. Valid values are X or Z.
tgbc_bg_grid_input%nx	integer	0	\	The number of grid cells along the x direction in the background mesh for the turbulence generation BC. If the background mesh is located on the $y - z$ plane, nx does not need to be specified and will not be used in the code.
tgbc_bg_grid_input%ny	integer	0	\	The number of grid cells along the y direction in the background mesh for the turbulence generation BC.
tgbc_bg_grid_input%nz	integer	0	\	The number of grid cells along the z direction in the background mesh for the turbulence generation BC. z direction is assumed to be the spanwise direction in 3D simulation.
tgbc_bg_grid_input%xl	real	1E12	\	The lower bound of the domain of the background mesh along x direction. If loc_str=X, xl and xh will be not used and do not need to be specified in the input file.
tgbc_bg_grid_input%xh	real	0.0	\	The upper bound of the domain of the background mesh along x direction.
tgbc_bg_grid_input%yl	real	1E12	\	The lower bound of the domain of the background mesh along y direction.
tgbc_bg_grid_input%yh	real	0.0	\	The upper bound of the domain of the background mesh along y direction.
tgbc_bg_grid_input%zl	real	1E12	\	The lower bound of the domain of the background mesh along z direction. z direction is the spanwise direction. In 3D, zl and zh are always required.
tgbc_bg_grid_input%zh	real	0.0	\	The upper bound of the domain of the background mesh along z direction.

7 Solution

Input Variable	Type	Default Value	Note	Description
iter_out_interval	integer	1	\	Number of time steps between writing time-stepping and residual/convergence statistics to standard output.
results_interval	integer	0	\	Number of time steps between writing residual/convergence statistics for all conserved and primitive variables to the file results.dat.
output_interval	integer	-5	\	Number of time steps between writing CGNS solution files.
output_dir	character(len=150)	.	\	File path to the directory used for writing output files. NOTE: If one is available, it is recommended that the output directory be located on a Lustre file system in order to get the best I/O performance possible.
output_order	integer	-3	\	Specifies the degree of the polynomial space used to represent the solution in the CGNS solution files. This simply allows the ability to over-sample the solution polynomial which provides a smoother solution within each cells when visualizing the solution. NOTE: This is only used if output order > solution order.
loc_output_pts	integer	0	\	Specifies the location of the nodal points within the grid cells when writing the solution to the CGNS solution files. ● 0 := Equi-distant nodes ● 1 := Legendre-Gauss nodes ● 2 := Legendre-Gauss-Lobatto nodes NOTE: Using Legendre-Gauss nodes for the CGNS solution files will result in empty space between all grid cells because there is no connectivity information between cells.
convergence_order_abs_res	real	-16.0	\	This is the convergence goal for the orders of magnitude of the absolute residual.
convergence_order_max_res	real	-16.0	\	This is the convergence goal for the orders of magnitude reduction of the residual relative to the maximum value of the residual during the simulation.

Input Variable	Type	Default Value	Note	Description
output_time_averaging	logical	.FALSE.	\	Logical flag that enables/disables the accumulation of timeaveraged flow variables. If enabled, time-averaged restart and CGNS solution files will be written whenever the standard restart and CGNS solution files are written.