**New and Updated Features of CFL3D Version 6.8**

\* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \*

VERSION 6.7：

CFL3D is a structured-grid, cell-centered, upwind-biased, Reynolds-averaged Navier-Stokes (RANS) code. It can be run in parallel on multiple grid zones with point-matched, patched, overset, or embedded connectivities. Both multigrid and mesh sequencing are available in time-accurate or steady-state modes.

VERSION 6.8：

The version features high-fidelity hybrid RANS/LES simulations. Turbulence model is the anisotropic minimum-dissipation IDDES model (AMD-IDDES). Inviscid flux scheme is a blended central/upwind scheme. Enhanced I/O functions include flow field averaging, animating and sampling.

\* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \*

VERSION 6.7 : Computational Fluids Lab, Mail Stop 128,

NASA Langley Research Center, Hampton, VA

Release Date: February 1, 2017

VERSION 6.8 : Maochao Xiao, Yufei Zhang,

Ziyu Zhou, Shaoguang Zhang

Tsinghua University, China

Sapienza University of Rome, Italy

Contact: maochao.xiao@uniroma1.it

zhangyufei@tsinghua.edu.cn

Release Date: March 1, 2023

\* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \* \*

**NEW FEATURES**

This page lists new features in CFL3D Version 6.8. Based on the Version 5 Manual and Version 6 Manual, this page introduces the new features added in CFL3D Version 6.8, including:

[VALID KEYWORDS: Turbulence Models 3](#_Toc130045209)

[VALID KEYWORDS: Input/Output Control 4](#_Toc130045210)

[Additional Plot3d Output Options: 6](#_Toc130045211)

[Note for cases without solid wall in ALL Version: 9](#_Toc130045212)

[Blended central/upwind scheme in Version 6.8: 10](#_Toc130045213)

[IDDES and AMD-IDDES Capability in Version 6.8: 12](#_Toc130045214)

[2D and coarse movie output in Version 6.8: 14](#_Toc130045215)

[Sampling Capability in Version 6.8: 15](#_Toc130045216)

[Running-average files in Version 6.8: 17](#_Toc130045217)

[Example Cases: 19](#_Toc130045218)

# VALID KEYWORDS: Turbulence Models

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Default Value** |
| ides | (usage with 6 or 7 available after V6.5and 11 or 12 available in V6.8) 0=no DES, 1=std DES, 2=DDES (TCFD 20:181195, 2006) 3=modified DDES with turb prod OFF in LES region (2 and up available after V6.5), 11=IDDES, 12=AMD-IDDES, 13=SLA-IDDES | 0 |
| iblend | flag for central or upwind scheme: 0 for upwind scheme; 1 for blended central/upwind scheme (available in V6.8) | 0 |
| blendmin | minimum of the blending factor associated with iblend (available in V6.8) | 0.1 |
| blendmax | maximum of the blending factor associated with iblend (available in V6.8) | 1.0 |
| cdes | constant associated with DES class methods  (different in V6.8) | 0.61 |
| lfe | constant associated with IDDES  (also AMD-IDDES and SLA-IDDES)  (available in V6.8) | 0 |
| cdt1 | constant associated with IDDES  (also AMD-IDDES and SLA-IDDES)  (available in V6.8) | 20 |
| llesblnd | constant associated with IDDES  (also AMD-IDDES and SLA-IDDES)  (available in V6.8) | 2 |
| cf1 | constant associated with IDDES  (also AMD-IDDES and SLA-IDDES)  (available in V6.8) | 0.2 |
| cf2 | constant associated with IDDES  (also AMD-IDDES and SLA-IDDES)  (available in V6.8) | 6 |
| cdesamd | constant associated with AMD-IDDES  (available in V6.8) | 2.40 |
| clim | constant associated with SLA-IDDES  (available in V6.8) | -1 |
|  |  |  |

# VALID KEYWORDS: Input/Output Control

|  |  |  |
| --- | --- | --- |
|  | | |
| **Name** | **Description** | **Default Value** |
| iin\_crs | capability to skip points when using icoarsemovie in i direction  1 = do not skip any points  2 = write every other point  n = write every nth point  (available in V6.8) | 1 |
| jin\_crs | capability to skip points when using icoarsemovie in j direction similarly  (available in V6.8) | 1 |
| kin\_crs | capability to skip points when using icoarsemovie in k direction similarly  (available in V6.8) | 1 |
| jinc\_2d | capability to skip points when using i2dmovie in j direction  1 = do not skip any points  2 = write every other point  n = write every nth point  (available in V6.8) | 1 |
| kinc\_2d | capability to skip points when using i2dmovie in k direction similarly  (available in V6.8) | 1 |
| iteravg | flag to store iteration-averaged variables  0 = no averaging or storage  1 = start averaging  2 = continue averaging from previous run, or start averaging now if old files not available  (different in V6.8, see manual) | 0 |
| ioffbody | flag (0/1) for selecting the first layer grid velocity or the real wall velocity as the surface velocity(available in V6.8) | 1  (select the first layer grid velocity) |
| isample | flag for closing/opening the sampling function, when isample>0, signs the capability to skip time steps when writing sampling data(available in V6.8) | 0  (closing) |

# Additional Plot3d Output Options:

The plot3d input parameter iptype has been expanded to include the following:

(**itype=-2** is a new function added in Version 6.8. It is recommended to use **itype=-2** to output more detailed flow field files.)

iptype = 0...output q at grid points

-1...output q at grid points, but use off-body wall parallel velocity

instead of actual velocities at wall (which are zero),

in order to more easily plot wall surface-restricted streamlines...

should only be used in combination with nplot3d=-1

(available *after* V6.6)

-2...output function files at grid points (available in V6.8)

single precision, PLOT3D, unformatted output files:

field.g (gird file), field.q (function file) and

field.nam (name file).

1...output q cell centers

2...turbulence data at cell centers, output in place of the

standard plot3d q vector. (defaults in 2D are

production-term,uw, uu, ww; defaults in 3D are

production-term, uw, uu, ww Sk/eps)

3...smin at cell centers

4...eddy viscosity at cell centers

-4...eddy viscosity at grid points (available *after* V6.4)

5...cp at cell centers

-5...cp at grid points

6...p/pinf at cell centers

-6...p/pinf at grid points

7...turbulence index at walls (based on cell-center data);

only works for 3D cases (i2d=0) and

should only be used in combination with nplot3d=-1...

uses eqn 10 from Recherche Aerospatiale, Vol. 1, 1994, pp. 5-21,

based on near-wall eddy viscosity translated to

Spalart-Allmaras variable form... this formula is appropriate

for S-A, but only an approximate (crude) indicator for other models

(available *after* V6.6)

* general rule: iptype <= 0 gives grid point data, iptype > 0 gives cell-center or cell-face-center data
* If nplot3d = -1, then the code will look for any surfaces that have bc types 1004, 2004 or 1005 (i.e. solid surfaces), and output those to the specified plot3d files. This can provide an excellent means of determining if all solid surfaces are specified correctly in the boundary condition section.
* if nplot3d = -1, one line of plot3d input data MUST appear in the plot3d data section near the bottom of the input file. In this one line of plot3d data, only the value of iptype is used; all other values are ignored and may be set to zero. **Note that this is different than the procedure in Version 5, where nplot3d = -1 expected NO lines of plot3d input**.
* if cell center data is requested (iptype > 0) and nplot3d = -1, then cell FACE CENTER data is output for cell faces lying on solid surfaces
* if nprint = -1, the same comments apply as for nplot3d < 0, but note that in the print out section, only iptype = 0 or 1 are valid types for print out
* For **iptype**=-2, the current default variables for output are:

|  |  |
| --- | --- |
| **Variables** | **Physical meaning** |
| R | Density |
| U, V, W | Velocity in xyz direction |
| P | Pressure |
| T | Temperature |
| mul | Laminar viscosity |
| mut | Turbulent viscosity |
| k | Turbulent kinetic energy |
| omega | Specific dissipation rate |
| M | Mach number |
| omegax, omegay, omegaz | Vorticity in xyz direction |
| Q | Q criterion |
| gradr | Density gradient |
| gradp | Pressure gradient |
| fd | Shielding function |
| LRANS | RANS length scale |
| LLES | LES length scale |
| XBLEND | Blending factor |

* **Note that** for **iptype**=-2, **ialph**=0 will not work. If **itype**=-2 is used, make sure **ialph** is set to 1, or the resulting **field.q** file will not be readable.

# How to compile the CFL3D program:

On **Windows**, you can compile using **Visual Studio**. The specific operations are not described here.

Note that when compiling the **serial** version, the **preprocessor definitions** contains “DBLE\_PRECSN; GENERIC; P3D\_SINGLE; NOREDIRECT C”, while the **parallel** version contains “DBLE\_PRECSN; DIST\_MPI; FASTIO; GENERIC; P3D\_SINGLE; NOREDIRECT”.

Meanwhile, the compiled version is divided into **optimized version** and **unoptimized version**. The **Optimization** option is "Disable (/Od)" for the non-optimized version, and "Maximize Speed plus Higher Level Optimizations (/O3)" for the optimized version.

On **Linux**, the CFL3D project contains four folders: **build**, **header**, **misc** and **source**. The **cfl3d** folder in the **source** folder contains all the program code, which is distributed in two folders: **dist** and **libs**. In other words, if you want to change code or overwrite old code, you only need to change it in these two folders.

The build file is contained in the **build** folder. Before compiling the program, you first need to type the **./Install** command to clean up from any previous installation. Then compile with the command **make -f makefile\_mpi cfl3d\_mpi**, where **\_mpi** corresponds to the parallel version and changing it to **\_seq** corresponds to the serial version.

Optimized and unoptimized versions are selected through options in the makefile. The following statements exist in both the **makefile\_seq** and **makefile\_mpi** files, where "..." is the file directory. The optimized version is controlled by the first option, **-O0** for unoptimized and **-O2** for optimized.

FFLAG = -O0 -w -ip -fno-alias -xHost -traceback -module /.../build/$(CFLLIBSD)

FFLAG\_SPEC = -O0 -w -ip -fno-alias -xHost -traceback -module /.../build/$(CFLLIBSD)

# Note for cases without solid wall in ALL Version:

When there is **no solid wall boundary condition** in the example, i.e. **1004**, **2004** and **2005** boundary conditions, **nprint**=0 should be set in the input file **cfl3d.inp**, and the input Settings in **Print Out Specifications** are as follows:

PRINT OUT:

GRID IPTYPE ISTART IEND IINC JSTART JEND JINC KSTART KEND KINC

That is, the input at Print Out line is zero lines.

# Blended central/upwind scheme in Version 6.8:

The present simulations were carried out using an in-house finite volume code. The inviscid flux is discretized using a hybrid central/upwind scheme, where a 4th-order kinematic energy preserving (KEP) scheme is used to compute the central flux, and the Roe/3rd-order MUSCL scheme (which is the same in Version 6.7) is used to calculate the upwind flux.

The kinetic energy preserving (KEP) scheme:



Details of KEP scheme is consistent with this paper:

Antony Jameson. “Formulation of Kinetic Energy Preserving Conservative Schemes for Gas Dynamics and Direct Numerical Simulation of One-Dimensional Viscous Compressible Flow in a Shock Tube Using Entropy and Kinetic Energy Preserving Schemes,”[*Journal of Scientific Computing*](about:blank), 2008, 34(2):188-208.

DOI: [10.1007/s10915-007-9172-6](about:blank)

The viscous flux is approximated using a second-order central difference scheme. Time integration is performed using the implicit dual-time-step lower–upper symmetric Gauss–Seidel scheme:



The blending factor  is determined by using the ratio of the length scale of DDES to that of RANS:



where C3 and C4 are set as 4.0 and 0.6, respectively. In the separated regions, the LES mode is activated, and the ratio of lDDES/lRANS is commonly less than 0.6. Thus,  decreases to 0 rapidly, and the adaptive dissipation scheme degrades into the upwind scheme, which aids in resolving the turbulent structures. In the near wall and far field regions, the ratio of lDDES/lRANS is equal to 1 and the adaptive dissipation scheme returns to the blended central/upwind scheme.

Blended central/upwind scheme has been added to CFL3D in Version 6.8 through the use of the [Keyword input](about:blank#keyword) **iblend** and **blendmin**, **blendmax**. By default, **iblend** is set to be 0, indicating that the upwind scheme, Roe/3rd MUSCL scheme, is used. For blended central/upwind scheme, set **iblend**=1, where the central format is the 4th-order kinematic energy preserving scheme. The Keyword input **blendmax** and **blendmin** is the maximum and minimum of the blending factor . By default , the limiter **blendmin** is set as 0.1 and **blendmax** is set as 1.0 in order to guarantee the numerical stability.

# IDDES , AMD-IDDES and SLA-IDDES Capability in Version 6.8:

**IDDES:**

IDDES is the improved delayed detached eddy simulation approach proposed by Shur et al, which modifies the near-wall subgrid length scale to accelerate the generation of resolved turbulence in the RANS/LES switch region and introduces an elevating function to prevent the excessive reduction of the modeled Reynolds stress before the RANS/LES transition.

Original SST-IDDES model is consistent with this paper.

Gritskevich, M. S., Garbaruk, A. V., Schütze, J., and Menter, F. R. "Development of Ddes and Iddes Formulations for the K-Ω Shear Stress Transport Model\_Sst-Iddes," *Flow, Turbulence and Combustion*, Vol. 88, No. 3, 2011, pp. 431-449.

doi: 10.1007/s10494-011-9378-4

Reijasse, P., Gritskevich, M. S., Garbaruk, A. V., Menter, F. R., Knight, D., Ivanov, M., and Lipatov, I. "Fine-Tuning of Ddes and Iddes Formulations to Thek-Ωshear Stress Transport Model\_Sst-Iddes," 2013, pp. 23-42.

doi: 10.1051/eucass/201305023

IDDES capability has been added to CFL3D in Version6.8, through the use of the [Keyword input](about:blank#keyword) **ides**, **lfe**, **cdt1** and **lnwsc**. For IDDES, set **ides**=11 and use the kewords **lfe** and **cdt1** to control the specific parameters of turbulence model. The code should also be run in time-accurate mode. Running averages of flow quantities can be kept using, for example, keyword **iteravg**.

In SST-IDDES model, the IDDES length scale is defined as:



where  is the elevating function. The Keyword input **lfe** in the program is this elevating function, which is designed to prevent the excessive reduction of eddy viscosity in the vicinity of the RANS/LES interface.

And the delaying function  is defined as , which is determined by the geometry part and the flow part. The Keyword input **cdt1** is the parameter  in the expression of , which is designed to control the switchover of RANS/LES of the DDES branch.

By default, **cdt1**=20is adopted to avoid a premature switching to LES mode within a turbulent boundary layer, and the elevating function  is not activated, i.e., **lfe**=0.

**Note that**, since the turbulence model used in Version 6.7 is a DES class method based on the SA model, the default value of **cdes** is 0.65 and the default value of **cdt1** is 8.0. However, the current Version 6.8 uses the DES class method based on the SST model, so the default value of **cdes** is 0.61 and the default value of **cdt1** is 20.0.

Because the IDDES method changes the definition of subgrid length scale, the simulated vortex viscosity value is small, and it is easy to appear velocity curve. Therefore, the keyword **lnwsc** is adopted to limit the length scale. When shielding function switches between 0 and 1, the length scale will be too small in the switching process due to the limitation of near-wall scale used in the original definition. When **lnwsc=0** is set, the limitation of near-wall scale will be removed to alleviate the occurrence of velocity turning.

However, for all of the aforementioned forms of the DES, the delay of Kelvin Helmholtz (K-H) instability occurs when RANS switches to LES in the initial region of a free shear layer. The defect is called the “grey area” issue.

**AMD-IDDES:**

In order to alleviate the grey area issue, Rozema proposed anisotropic minimum-dissipation (AMD) model to improve the excessive sublattice stress. Details of AMD model is consistent with this paper:

Wybe Rozema, Hyun J. Bae, Parviz Moin, and Roel Verstappen. “Minimum-dissipation models for large-eddy simulation,” Physics of Fluids, 2015, 27.

doi: 10.1063/1.4928700

Based on AMD model, Xiao Maochao proposed a turbulence model combining AMD and IDDES, namely AMD-IDDES model: the free-of-wall subgrid length scale in IDDES model is replaced using AMD model and the near-wall subgrid length scale is untouched to avoid “log-layer mismatch”.

AMD-IDDES capability has also been added to CFL3D in Version6.8, through the use of the [Keyword input](about:blank#keyword) **ides**, **cdesamd** and **clim**. For AMD-IDDES, set **ides**=12 and other parameter settings are similar to IDDES.

The turbulence viscosity expression in AMD subgrid stress model is:



Where, , which is the keyword **cdesamd**, and the default value is 2.40. The subgrid length scale is:



However, in the IDDES method, if the local invisticity is reduced to 0, the overall vortex viscosity will be smaller, resulting in a large error in the simulation results. Therefore, in the AMD-IDDES method, the minimum vortex viscosity is not limited by 0, but by absolute value, which is the default operation when the keyword **clim=-1**.



If the keyword **clim** is set to a value **larger than 0**, other limits instead of absolute values will be used. The general form is shown in the formula. For details, see the reference program code.

**SLA-IDDES:**

The shear layer adaptive length scale (SLA) was proposed by Shur et al., which can alleviate the "grey area problem". The expression is:



For more information about and the construction of the SLA-IDDES method, see the following literature:

Shur M L, Spalart P R, Strelets M K, et al. “An enhanced version of DES with rapid transition from RANS to LES in separated flows”. *Flow, Turbulence and Combustion*, 2015, 95(4): 709737.

doi: 10.1007/s10494-015-9618-0

Guseva E K, Garbaruk A V, Strelets M K. “Assessment of delayed DES and improved delayed DES combined with a shear-layer-adapted subgrid length-scale in separated flows.” *Flow, Turbulence and Combustion*, 2016, 98(2): 481-502.

doi: 10.1007/s10494-016-9769-7

SLA-IDDES capability has also been added to CFL3D in Version6.8, through the use of the [Keyword input](about:blank#keyword) **ides**. For SLA-IDDES, set **ides**=13 and other parameter settings are similar to IDDES.

**Other *Keywords* for related settings:**

As IDDES shielding function cannot protect the boundary layer when the boundary layer mesh is dense, wall friction coefficient will decrease, velocity turning and other phenomena will occur. Therefore, the restrictions introduced in AMD-IDDES and SLA-IDDES methods are as follows:



The default value of [Keyword input](about:blank#keyword) **llesblnd** is 2, indicating the use of this restriction, that is, the hard blending mode. In contrast, when keyword **llesblnd** has a value of 1, it means no restriction is used, that is, soft blending mode.

The shielding function **fdf** used by the above two blending modes is processed in both the boundary layer and the far field, namely. When keywords **llesblnd** are 3 and 4, it means that the shielding function **fd** is only processed in the boundary layer, where 3 is soft blending mode and 4 is hard blending mode.

The other set of [Keyword input](about:blank#keyword), **cf1** and **cf2**, are used to control the exponent in the far field shielding function **ff**, where **cf1** defaults to 0.2 and **cf2** defaults to 6. Their purpose is to enable **ff** to cover a wide region of farfield and yield rapid switching.

# 2D and coarse movie output in Version 6.8:

In V6.8, the input file **coarsemovie.inp** in previous version is replaced by the Keyword inputs **iin\_crs**, **jin\_crs**, **kin\_crs** and **iinc\_2d**, **jinc\_2d**, **kinc\_2d**.

The Keyword inputs **iin\_crs**, **jin\_crs** and **kin\_crs** are used in conjunction with **icoarsemovie**, and represent the reduction factors for the three directions, the same as the parameters iinc\_coarse, jinc\_coarse, kinc\_coarse in **coarsemovie.inp**. When using **icoarsemovie**, the single precision, PLOT3D, unformatted ouput files are named:

**field\_crs\_00005.q, field\_crs\_00005.g**, **field.nam** (If **icoarsemovie** = 5)

The Keyword inputs **jinc\_2d**, **kinc\_2d** are used in conjunction with **i2dmovie**. In this case, the plane i=imax/2 will be output (iinc\_2d is not used). The Keyword inputs jinc\_2d and kinc\_2d are reduction factors for the j and k directions, respectively, the same as the parameters iinc\_2d, jinc\_2d, kinc\_2d in **coarsemovie.inp**. When using **i2dmovie**, the single precision, PLOT3D, unformatted files are named:

**field\_2d\_00005.q field\_2d\_00005.g**, **field.nam** (If **i2dmovie** = 5)

The variables of these files are all the same as the variables of output function file **field.q** for **iptype**=-2.

# Sampling Capability in Version 6.8:

The Keyword input **isample** is the flag for opening the function of sampling. By default, **isample**=0, meaning closing the sampling capability. For opening sampling capability, set **isample**>0, where the value of **isample** represents the capability to skip time steps when sampling. For example, **isample**=1 means do not skip any time steps, **isample**=2 means write sampling data every other time step, and **isample**=n means write every nth time step.

In the input file **cfl3d.inp**, the words “SAMPLING POINTS:” marks the beginning of the input file for the sampling capability. The format of the input file is as follows:

SAMPLING POINTS:

NSMP

1

GRID IPTYPE ISTART IEND IINC JSTART JEND JINC KSTART KEND KINC

1 1 1 1 1 1 1 1 1 2 1

***LT1 (in this section) - Sampling Option***

**nsmp**: the number of sampling lines

***LT2 (in this section) – Sampling Print Out Specifications***

(Data for Line Type Two should be repeated **nsmp** times.)

**grid** – designated grid number for output

**iptype** – type of flow-field variables to print out (since the sample output value is cell center value, **iptype** must be set to 1)

**istart** – starting location in direction

**iend** – ending location in direction

**iinc** – increment factor in direction

**jstart** – starting location in direction

**jend** – ending location in direction

**jinc** – increment factor in direction

**kstart** – starting location in direction

**kend** – ending location in direction

**kinc** – increment factor in direction

A folder named "sample" will be created separately for the output of the sampling function. In this folder, each specific sampling output line corresponds to a separate folder (a total of **nsmp** folders), such as "smp\_01", which contains five txt files corresponding to the output of the five original variables R,U,V,W and P. In the output txt file, the first column is the number of iteration steps, the second column is the value corresponding to the first position specified in that sample line, the second column is the second position, and so on. For example:

1 0.1210E+00 0.1210E+00

2 0.1209E+00 0.1209E+00

3 0.1208E+00 0.1208E+00

4 0.1206E+00 0.1206E+00

5 0.1204E+00 0.1204E+00

(Here, a total of one line is sampled, that is, **nsmp**=1, and the sampling line contains two points)

# Running-average files in Version 6.8:

In Version 6.8, the Keyword inputs **iteravg**, **ipertavg** and **npertavg** are all replaced by **iteravg**. This means that if **iteravg** is set to 1 or 2, then all the average quantities about the flow field and the surface will be output, including the approximation to the average of the square of the perturbation of each variable.

The field average variables are written to:

(**field.g**) **field\_avg.q** (**field.nam**)

(The files in parentheses are printed whether **iteravg** is equal to 1 or 0)

The current default variables for output are:

|  |  |
| --- | --- |
| **Variables** | **Physical meaning** |
| R\_avg, R\_rms | Density average and root mean square |
| u\_avg\_x, u\_avg\_y, u\_avg\_z | Velocity average in xyz direction |
| u\_rms\_x, u\_rms\_y, u\_rms\_z | Velocity root mean square in xyz direction |
| u\_rey\_x, u\_rey\_y, u\_rey\_z | Reynolds stress in xyz direction |
| P\_avg, P\_rms | Pressure average and root mean square |
| T\_avg, T\_rms | Temperature average and root mean square |
| mul\_avg, mul\_rms | Laminar viscosity average and root mean square |
| mut\_avg, mut\_rms | Turbulent viscosity average and root mean square |
| k\_avg, k\_rms | Turbulent kinetic energy average and root mean square |
| omega\_avg | Specific dissipation rate mean average |
| M\_avg, M\_rms | Mach number average and root mean square |
| omega\_avg\_x, omega\_avg\_y, omega\_avg\_z | Vorticity average in xyz direction |
| omega\_rms\_x, omega\_rms\_y, omega\_rms\_z | Vorticity root mean square in xyz direction |
| Q\_avg, Q\_rms | Q criterion average and root mean square |
| gradr\_avg, gradr\_rms | Density gradient average and root mean square |
| gradp\_avg, gradp\_rms | Pressure gradient average and root mean square |
| fd\_avg, fd\_rms | Shielding function average and root mean square |
| LRANS\_avg, LRANS\_rms | RANS length scale average and root mean square |
| LLES\_avg, LLES\_rms | LES length scale average and root mean square |
| XBLEND\_avg, XBLEND\_rms | Blending factor average and root mean square |

The surface average variables are written to:

(**surf.g**) **surf\_avg.q** (**surf.nam**)

(The files in parentheses are printed whether **iteravg** is equal to 1 or 0)

The current default variables for output are:

|  |  |
| --- | --- |
| **Variables** | **Physical meaning** |
| U\_AVG\_X(OB), U\_AVG\_Y(OB), U\_AVG\_Z(OB) | Velocity (of the first layer cell center) average in xyz direction |
| U\_RMS\_X(OB), U\_RMS\_Y(OB), U\_RMS\_Z(OB) | Velocity (of the first layer cell center) root mean square in xyz direction |
| P\_AVG, P\_RMS | Pressure average and root mean square |
| T\_AVG, T\_RMS | Temperature average and root mean square |
| M\_AVG, M\_RMS | Mach number average and root mean square |
| CP\_AVG, CP\_RMS | Pressure coefficient average and root mean square |
| MUT\_AVG, MUT\_RMS | Turbulent viscosity average and root mean square |
| DIS\_AVG, DIS\_RMS | First-layer cell height average and root mean square |
| CH\_AVG, CH\_RMS | Surface heat transfer coefficient average and root mean square |
| YPLUS\_AVG, YPLUS\_RMS | Yplus average and root mean square |
| CF\_AVG\_X, CF\_AVG\_Y, CF\_AVG\_Z | Friction coefficient average in xyz direction |
| CF\_RMS\_X, CF\_RMS\_Y, CF\_RMS\_Z | Friction coefficient root mean square in xyz direction |

The files are all in single precision PLOT3D format.

# Example Cases:

In the folder "**CFL3D6.8\_cases**" provided, there are two calculation examples of CFL3D Version 6.8, which are respectively examples of RANS and AMD-IDDES methods for **2D Airfoil Near-Wake**. The grid used in both examples is the same, and the airfoil is a DSMA661(MODEL A).

The grid for these cases is based on airfoil coordinates extracted from the reference:

Nakayama, A., "Characteristics of the Flow around Conventional and Supercritical Airfoils," J. Fluid Mech. (1985), Vol. 160, pp. 155-179

[https://doi.org/10.1017/S0022112085003433](about:blank).

These cases are run fully turbulent at angle of attack of zero degrees, M = 0.088, and a Reynolds number of Re = 1.2 million based on airfoil chord length. The outer boundary of the grid in this case is 500c from the airfoil, where farfield Riemann boundary conditions are imposed.

In RANS and AMD-IDDES folders, **cfl3d.inp** is the input file, and **cfl3d.x** is the grid file. Run the CFL3D program, then you can get the PLOT3D format output file: **field.g** (grid file), **field.q** (function file) and **field.nam** (name file), and get flow field distribution using these three files.

**Note that** DES needs to be calculated using restart calculation on top of the RANS calculation, so copy the **cfl3d.restart** file generated by the RANS calculation to the AMD-IDDES folder.