

Instruction for Code Running

Sayop Kim (gtID#: 902823126)

1. Code Setup

The GridGen source code has been developed with version management tool, GIT. The git repository was built on 'github.com'. Thus, the source code as well as related document files can be cloned into user's local machine by following command:

```
$ git clone http://github.com/sayop/GridGen.git
```

If you open the git-cloned folder GridGen , you will see two different folder and README file. The CODEdev folder contains again bin folder, Python folder, and src folder. In order to run the code, user should run setup.sh script in the bin folder. Python folder contains python script that is used to postprocess RMS residual data. It may contain build folder, which might have been created in the different platform. Thus it is recommended that user should remove build folder before setting up the code. Note that the setup.sh script will run cmake command. Thus, make sure to have cmake installed on your system:

```
$ rm -rf build
$ ./setup.sh
-- The C compiler identification is GNU 4.6.3
-- The CXX compiler identification is GNU 4.6.3
-- Check for working C compiler: /usr/bin/gcc
-- Check for working C compiler: /usr/bin/gcc -- works
-- Detecting C compiler ABI info
-- Detecting C compiler ABI info - done
-- Check for working CXX compiler: /usr/bin/c++
-- Check for working CXX compiler: /usr/bin/c++ -- works
-- Detecting CXX compiler ABI info
-- Detecting CXX compiler ABI info - done
-- The Fortran compiler identification is Intel
-- Check for working Fortran compiler:
/opt/intel/composer_xe_2011_sp1.11.339/bin/intel64/ifort
-- Check for working Fortran compiler:
/opt/intel/composer_xe_2011_sp1.11.339/bin/intel64/ifort -- works
-- Detecting Fortran compiler ABI info
-- Detecting Fortran compiler ABI info - done
-- Checking whether /opt/intel/composer_xe_2011_sp1.11.339/bin/intel64/ifort
supports Fortran 90
-- Checking whether /opt/intel/composer_xe_2011_sp1.11.339/bin/intel64/ifort
supports Fortran 90 -- yes
-- Configuring done
-- Generating done
-- Build files have been written to:
/data/ksayop/GitHub.Clone/GridGen/CODEdev/bin/build
```

```

Scanning dependencies of target cfd.x
[ 12%] Building Fortran object CMakeFiles/cfd.x.dir/main/Parameters.F90.o
[ 25%] Building Fortran object CMakeFiles/cfd.x.dir/main/SimulationVars.F90.o
[ 37%] Building Fortran object CMakeFiles/cfd.x.dir/io/io.F90.o
[ 50%] Building Fortran object CMakeFiles/cfd.x.dir/main/SimulationSetup.F90.o
[ 62%] Building Fortran object CMakeFiles/cfd.x.dir/main/GridSetup.F90.o
[ 75%] Building Fortran object CMakeFiles/cfd.x.dir/main/GridTransformSetup.F90.o
[ 87%] Building Fortran object CMakeFiles/cfd.x.dir/main/GridTransform.F90.o
[100%] Building Fortran object CMakeFiles/cfd.x.dir/main/main.F90.o
Linking Fortran executable cfd.x
[100%] Built target cfd.x
$ ls
$ build  cfd.x  input.dat  setup.sh

```

If you run this, you will get executable named **cfd.x** and **input.dat** files. The input file is made by default. You can quickly change the required options.

2. Input File Setup

The GridGen code allows user to set multiple options to generate grid by reading **input.dat** file at the beginning of the computation. Followings are default setup values you can find in the input file when you run **setup.sh** script:

```

# Input file for tecplot print
Flow in a channel
imax          41
jmax           2
kmax          19
# domain input (Corner points: x,y coordinates)
p1            -0.8    0.0    0.0
p2             1.8    0.0    0.0
p3            -0.8    0.0    1.0
p4             1.8    0.0    1.0
GeoStart       0.0    0.0    0.0
GeoEnd         1.0    0.0    0.0
FESize         11
GeoSize        21
DCsize         11
width          0.1
# Grid clustering:
# cyl: stretched grid in z
# cy2: stretched Pi in z
# cy3: stretched Psi in x
# cy4: stretched grid along FE
# cy5: stretched grid along ED
# cy6: stretched grid along DC
cyl            2.0
cy2            -5.001
cy3            0.001
cy4            -1.2
cy5            1.0
cy6            0.001

```

```

# Iteration max: If nmax == 0, elliptic grid won't be calculated
nmax          500
# RMS Criterion
RMScrit       1.0E-6
# Calculate control terms: Pi, Psi
iControl      1

```

- **imax, jmax, kmax:** These three parameters set the size of grid points in **x**, **y**, and **z** direction, respectively.
- **p1, p2, p3, p4:** Define the corner points that form the front surface of the 3-dimensional computational domain.
- **GeoStart, GeoEnd:** Start and end points of airfoil geometry
- **FESize, GeoSize, DCsize:** Number of grid points along FE, airfoil shape, and DC
- **width:** Depth of 3D computational domain in **y**-direction.
- **cy1 ~ cy6:** Stretching parameters used in the stretching formula, which is inherently defined for the grid point spacing in the **z** direction. In this code, this formula is applied to control terms and bottom edge spacing to define a new grid alignment for Grid #5.
- **nmax:** Maximum number of main loop. If the residual criterion is met before this maximum number is reached, the code will be terminated. If nmax is set to 0, the code will only run for the algebraic grid.
- **RMScrit:** Minimum RMS residual value to obtain the converged Thomas method calculation.
- **iControl:** If it is 1, the code runs with pre-specified ϕ and ψ at the boundary points.