# TIME-ACCURATE RUNS with NSU3D

The parameter ITACC in the NSU3D input file controls the time accuracy of the simulation run.

```
ITACC = 0 !<---Steady-state run
ITACC > 0 !<---Time dependent run
     ITACC = 1 !<---First-order  accurate (BDF1) implicit time stepping scheme
     ITACC = 2 !<---Second-order accurate (BDF2) implicit time stepping scheme
```

For time accurate runs, additional time accurate parameters must be included in the NSU3D input file. These are placed after the **FORCE/MOMENT COEFFICIENT PARAMETERS** line and before the **MESH DATA FILE** line, using separation lines as shown below:

```
----------------------------------------------------------------------
FORCE/MOMENT COEFFICIENT PARAMETERS
REF_AREA  REF_LENGTH XMOMENT  YMOMENT   ZMOMENT  ISPAN   REF_MACH
858.3     16.5          0.0   0.0        0.0     3.0      0.64
----------------------------------------------------------------------
OPTIONAL TIME ACCURATE PARAMETERS
NTIME_STEP  DTACC    DT_REF_LENGTH   NTIME_STEP_OUT   MESHMOTION
2880.       0.5        -1.0            30000.              1.0
MESH MOTION INPUT FILE
input2.meshmotion.nml
--------------------------------------------------------------------
MESH DATA FILE
../grids/Hart2B_Rotor_C5_2.32Mn.part.320
----------------------------------------------------------------------
```

The key words **OPTIONAL TIME ACCURATE** or **TIME ACCURATE** are used to trigger reading of these parameters. The required parameters, in the required order, are described below:

- **NTIME_STEP**: Specifies the total number of time steps for this run. The solution parameters specified in the input file (previously for steady-state runs) such as NCYC, NMESH etc. are used to solve the implicit system at each time step.
- **DTACC** : Specifies the numerical time step size in non-dimensional units. DTACC = 1 corresponds to the time required for the freestream flow to travel a distance of DT_REF_LENGTH. (For DT_REF_LENGTH =1, DTACC corresponds to a time step during which the freestream flow travels a distance of DTACC). If DT_REF_LENGTH = -1, DTACC is given in degrees of rotation.
- **DT_REF_LENGTH** : Reference length for time step scaling. If set = -1, then the time step DTACC is given in degrees of rotation, using the rotation rate of the mesh given in the ** MESH MOTION INPUT FILE** below. Note if the rotation rate is zero, this will cause the code to stop with an output message.
- **NTIME_STEP_OUT** : Output solution file after every NTIME_STEP_OUT. Can be superseeded using IN-SITU POSTPROCESSING.
- **MESHMOTION :** *Defines static or moving mesh.*
    - **MESHMOTION = 0** denotes static (non moving) mesh. In this case, no other inputs are required.
    - **MESHMOTION = 1** denotes mesh that moves with time (rigid motion). In this case, a mesh motion input file name must be specified on the next line.
        - **MESH MOTION INPUT FILE**: The name of the mesh motion file is specified here. The mesh motion file is a Fortran namelist file that specifies rotation and translation

values for solid body motion of the mesh. An example of a mesh motion file is given below. The parameters are defined as:

- **mesh_motion_def**: Key word in namelist identifying list of mesh motion parameters.
- **mesh_id**: The meshmotion file is also used for cases with multiple overset meshes. Therefore, the motion for each mesh is defined separately. In this case, with a single mesh (overset meshing disabled), the mesh_id is always = 1.
- **rotcenterxyz(1:3)**: Center of rotation for mesh motion.
- **omegaxyz(1:3)**: Rotation rate vector components.
- **gvtransxyz(1:3)**: Grid velocity components for moving grid translation.

```
!Overset mesh motion input file
!---------------------------------------------------
&mesh_motion_def
  mesh_id           = 1
  rotcenterxyz(1:3) = 0.0,0.0,0.0
  omegaxyz(1:3)     = 0.0,0.0,0.0456389419788
  gvtransxyz(1:3)   = 0.0,0.0,0.0
/
!---------------------------------------------------
```

Note that the mesh motion input file is also copied to the ./WRK directory at run time in order to produce an archival record of all inputs for the current run.

# Non-dimensional Time Scaling:

The time step (and all temporal scales) are non-dimensionalized by the reference time scale, which corresponds to the time taken for the freestream flow to travel a unit distance in the grid. The input time step **DTACC** can be scaled by a reference length, such that the freestream flow travels the reference length distance **DT_REF_LENGTH** in time DTACC=1. In other words, the input value of **DTACC** is multiplied by **DT_REF_LENGTH** in NSU3D to set a time step corresponding to that distance. However, the internal scaling of temporal values remain nondimensionalized by the unit grid length. This means that grid velocities (translational and rotational) are nondimensionalized by the reference time scale as stated above.

To set appropriate non-dimensional numerical values for grid speed terms, either non-dimensional values can be matched, or physical quantities must be used to obtain the proper scaling. For example, if the Mach number is specified, this can be used to set the grid speed terms. Here we give a translational and a rotational example.

- **Mach Scaling Translational Example**:

The DLR-F6 test case with Mach=0.75 at 0 degrees incidence is reproduced using zero freestream Mach number. In this case, the mesh translational velocity must be set to Mach * c, where c is the speed of sound. The non-dimensionalization in NSU3D uses a freestream density and static pressure non-dimensionalized to unity. Therefore, the nondimensional freestream speed of sound is equal to sqrt(gamma). Thus, the grid velocities in this case are set as: gvtransxyz(1:3) = -0.8874,0.0,0.0 where Mach * c = 0.75 x sqrt(1.4) = 0.8874 and the minus sign indicates the mesh moves in the negative x-axis direction producing flow in the opposite direction relative to the moving body.

- **Mach Scaling Rotational Example**:

The HART-II rotor hover case is specified with a tip Mach number of 0.64. Therefore, we seek an omega value such that omega * r = 0.64 * sqrt(gamma) Our mesh is scaled by the chord of the blade (i.e. has unit chord length in grid coordinates) and a blade radius of 16.5 grid units. Thus, omega = 0.64 * sqrt(gamma) /

16.5 = 0.04589 Since this rotation is in the postive z-axis direction, the rotation vector is specified as: omegaxyz(1:3) = 0.0,0.0,0.0456389419788

- **Physical Scaling Translational Example**:

For the DLR-F6 test case with Mach=0.75 we wish to take a time step of 0.01 seconds. To obtain dimensional numbers we need to know the speed of sound and the size (or length scale) of the model. Here we assume a dimensional speed of sound of 341 m/sec and the mesh dimension is in millimeters. The physical freestream velocity is thus: Mach * speed of sound = 0.75 * 341 = 255.75 m/sec In the specified time step of 0.01, the flow travels 2.5575 meters. Therefore, the corresponding time step in the simulation is 2557.5. Note if the mesh was scaled in meters, the corresponding time step would be 2.5575. The NSU3D input time step corresponds to the grid distance travelled by the freestream flow in that time (for DT_REF_LENGH=1). The DTACC input value is divided by the freestream velocity (Mach*sqrt(gamma)) inside the code to produce the final nondimensional time step used in the calculations. This is done for convenience to provide a more intuitive time step input value for the user. The precise correspondance between the final internal time step scaling and a physical time step based in seconds is as follows:

DT* = DT * (physical speed of sound)/sqrt(gamma) * L/L*

where DT* is the internal nondimensional time step value, DT is the physical time step in seconds, and L/L* is the ratio of grid unit dimension to phyiscal dimension used in the speed of soune (i.e. meters).

- **Physical Scaling Rotational Example**:

The HART-II rotor hover case is specified with a rotational rate of 1041 RPM. The radius of the rotor is 2m and the speed of sound is 341 m/s. (One can easily verify that this produces a tip Mach number of 0.64 as previously.) First, the RPM is converted to rad/sec as: 1041 x 2 x PI/60 = 109.12 rad/sec Now the new rotational speed must be given in rad/DT* (non dimenmsional time). Thus, using the non-dimensional scaling from the previous example:

OMEGA* = OMEGA * L*/L sqrt(gamma)/speed of sound

OMEGA* = 109.12 * 0.121 * sqrt(1.4)/341

OMEGA* = 0.04589

# Example Test Cases

## Test Case 1: Hart 2 Rotor in Hover

This case simulates the 4-bladed HART2 rotor in hover using a time-dependent moving mesh approach. This case can also be run in noninertial mode, and these results can be compared with the noninertial results for the corresponding test case found in the documentation. The complete NSU3D input file for this case is reproduced below:

```
NSU3D TIME-ACCURATE CASE
RESTARTF  RESTARTT  RNTCYC
0        0       00.
RESTART FILE
none
```

```
MMESH     NTHREAD
1.0       1.0                (1st order = -1)
NCYC      NPRNT     N MESH    MESHLEVEL   CFLMIN    RAMPCYC   TURBFREEZE
25.        1000        4        1          1.0      0          0.0
CFL       CFLV      ITACC     INVBC     ITWALL    TWALL
1.0        1000.      2.0       0.0       0.0     0.0
VIS 1     VIS 2     H FACTOR  SMOOP     NCYCSM
0.0        20.0        0.0    0.00      0.0
C1        C2        C3        C4        C5        C6
0.5321    1.3711    2.7744
FIL1      FIL2      FIL3      FIL4      FIL5      FIL6
1.0       1.0       1.0
----------------------------------------------------------------------
COARSE LEVEL AND MULTIGRID PARAMETERS
CFLC      CFLVC     SMOOPC    NSMOOC
1.0       1000.     0.0       0.0
VIS0      MGCYC     SMOOMG    NSMOOMG
4.0       2.0        0.8       2
----------------------------------------------------------------------
TURBULENCE EQUATION(S)
ITURB     IWALLF    WALLDIST
4.0        0.0        1.0
CT1       CT2        CT3      CT4       CT5       CT6
 1.0      1.0        1.0      1.0       1.0
CTC1      CTC2       CTC3     CTC4      CTC5      CTC6
1.0       1.0        1.0      1.0       1.0
VIST0  TSMOOMG   NTSMOOMG
2.0        0.8        2
----------------------------------------------------------------------
MACH      Z-ANGLE   Y-ANGLE   RE       RE_LENGTH
0.0       0.0       0.0     2.147678e+06     1.0
----------------------------------------------------------------------
FORCE/MOMENT COEFFICIENT PARAMETERS
REF_AREA  REF_LENGTH XMOMENT  YMOMENT   ZMOMENT  ISPAN   REF_MACH
858.3     16.52893      0.0   0.0        0.0     2.0      0.637554652817
----------------------------------------------------------------------
OPTIONAL TIME ACCURATE PARAMETERS
NTIME_STEP  DTACC    DT_REF_LENGTH   NTIME_STEP_OUT   MESHMOTION
2880.       0.5        -1.0            30000.              1.0
MESH MOTION INPUT FILE
input.hart2.meshmotion.nml
--------------------------------------------------------------------
MESH DATA FILE
../grids/Hart2B_Rotor_C5_2.32Mn.part.320
----------------------------------------------------------------------
OPTIONAL PARAMETERS (MODIFY FROM DEFAULT VALUES SET IN set_lim_values.f)
PARAMETER NAME                           VALUE(real number with decimal)
ISAFE_MG                                  0.0
IN_SITU_POST_PROC                         1.0
```

The freestream conditions are zero flow, MACH=0.0. Note that the flow incidence angles (Y-ANGLE and Z-ANGLE) must also be zero otherwise the wind axis coordinate system will not be aligned with the body coordinate axes, even with no freestream flow. This input file specifies 25 subiterations per time step (NCYC = 25) using 4 levels of multigrid (NMESH = 4). The time step size is set to 0.5 degrees of rotation (DTACC = 0.5) and this refers to degrees rotation by setting DT_REF_LENGTH = -1.0. Note that the parameter ISAFE_MG can be set = 0.0, which accelerates multigrid convergence. ISAFE_MG = 1.0 is often used for steady-state cases to enhance stability. However, time-dependent cases with relatively small time steps are less stiff and can often converge faster with ISAFE_MG=0.0 The parmeter MESHMOTION=1.0 denotes a moving mesh case. The single block unstructured mesh rotates with the blades and the mesh motion is defined for this single mesh case with the following mesh motion input file (named: input.hart2.meshmotion.nml in the NSU3D input file):

```
!Mesh motion input file
!--------------------------------------------------
&mesh_motion_def
  mesh_id           = 1
  rotcenterxyz(1:3) = 0.0,0.0,0.0
  omegaxyz(1:3)     = 0.0,0.0,0.0456389419788
  gvtransxyz(1:3)   = 0.0,0.0,0.0
/
!--------------------------------------------------
```

The motion consists of rotation along the z-axis. The magnitude is such that the tip Mach number is approximately 0.64 (as discussed above). The simulation is run for 2880 time steps (NTIME_STEP = 2880) which corresponds to 4 rotor revolutions using the 0.5 degree time step. Figure 1.1. illustrates the convergence at each time step over the entire run as well as details of convergence for time steps near number 60000, showing that the residuals are reduced by approximately 2 orders of magnitude at each time step.

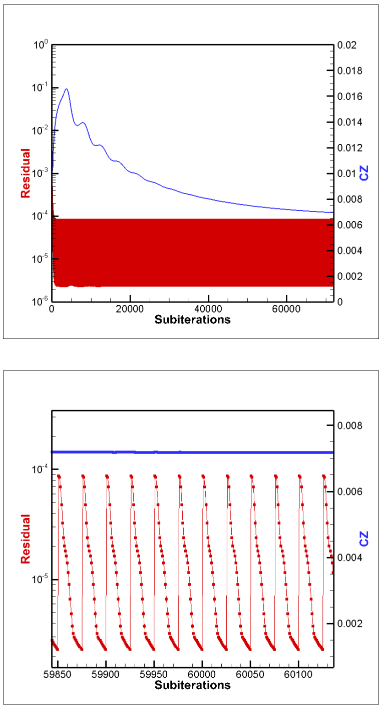

*Figure 1.1: Convergence history of time-dependent moving mesh rotor simulation in terms of subiterations. First figure provides residual and CZ (thrust) histories for entire run correspnding to 4 rotor revolutions. Second figure shows details for several time steps ( each time step corresponds to 0.5 degrees rotor rotation, using 25 subiterations per time step).*

Figure 1.2 depicts the time histories of thrust and power coefficients as a function of the number of time steps over the entire run.

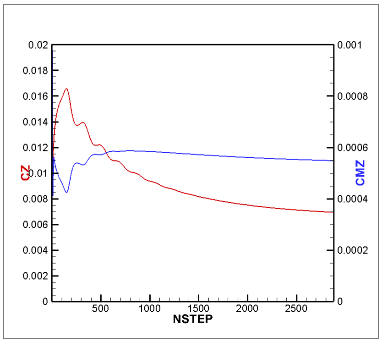

*Figure 1.2: Convergence of CZ force coefficient (thrust) and CMZ moment coefficient (power) plotted versus number of time steps for entire simulation corresponding to 4 rotor revolutions (0.5 degree time step).*

Figure 1.3 illustrates the computed flow solution at the end of four revolutions. These results can be compared with those obtained using the noninertial approach for this case. Notably, the force and moment values are still decreasing slowly after 4 rotor revolutions, suggesting a longer time simulation is required to reach the final equilibrium values for the hovering rotor.

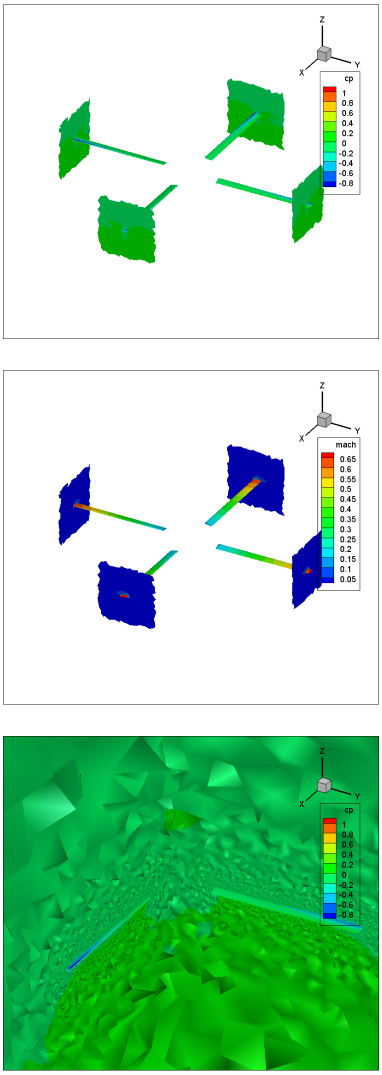

*Figure 1.3: Illustration of computed flow field after four rotor revolutions using 0.5 degree time step.*