# CACTUS User's Manual
## Version 1.0

Jonathan C. Murray
Aerosciences Department
Sandia National Laboratories
Albuquerque, NM 87185-0825

Matthew Barone
Wind Energy Technologies
Sandia National Laboratories
Albuquerque, NM 87185-0825

# Contents

# 1 Introduction

This document serves as a user's manual for CACTUS (Code for Axial and Crossflow TUrbine Simulation). A brief overview of the code is provided below, however, more detailed information on the methods used can be found in [1].

# 2 CACTUS Overview

CACTUS is a turbine performance simulation code based on the blade element method and using a free vortex line description of the turbine wake flow. The code was originally based on VDART3, a free vortex wake simulation of the Darrieus wind turbine, developed by Strickland [2]. The codebase has been largely upgraded to the Fortran 9x standard, and a number of modifications have been made to the original VDART3 methods including updates to the blade loads models, and new models to handle generic device geometry and marine turbine specific physics.

## 2.1 Blade Loads and Wake Models

CACTUS simulates a turbine device consisting of an arbitrary configuration of blade element sections. Each section can be assigned arbitrary load coefficient vs. angle of attack characteristics, which typically correspond to two-dimensional lift and drag coefficient data for a particular foil section. Since data from two-dimensional wind tunnel tests or foil performance calculations are used to represent element loads, it is generally assumed that these elements are in locally two-dimensional flow.

A rotor blade consisting of an arbitrary planform shape and foil sections can be modeled by the synthesis of a number of blade elements. The blade loads and wake of the turbine rotor are evolved in time over a certain number of rotor revolutions, until the revolution-averaged rotor power is converged. The code output includes the blade aerodynamic forces, wake vortex trajectories, and performance metrics such as torque and power.

CACTUS uses a potential flow model comprised of free vortex line elements to represent the turbine wake flow field. The vortex line structure attached to a single blade element is shown in Figure 1. At each point in time, the bound vorticity ($\Gamma_B$) on each blade element is related to the element lift coefficient through the Kutta-Joukowski theorem, and the spanwise ($\Gamma_S$) and trailing vorticity ($\Gamma_T$) are recovered through the application of the Helmholz theorem of conservation of circulation along a vortex line [3].

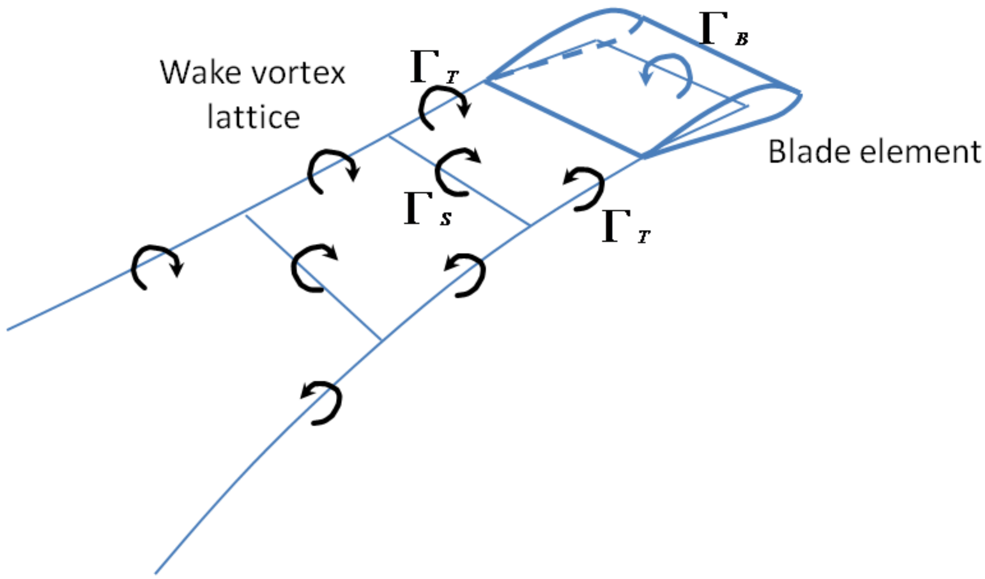


Figure 1. Blade element with associated vortex lattice system.

### 2.1.1 Dynamic Blade Loads

The operational cycle of some turbines, most notably cross-flow turbines or axial flow turbines in yawed flow conditions, cause the turbine blades to operate in dynamically variable flow conditions. The effects of blade rotation with respect to the surrounding fluid and the effects of dynamically variable flow angle of attack are captured with additional models.

The effects of blade section pitch rate (rotation around an axis normal to the section plane) are captured by analogy to an analytical solution for a pitching flat plate. Improvements have been made to the original methodology used in VDART3, and modifications have been made to handle non-zero section pitching moment due to cambered foil sections.

Under certain operational conditions, the turbine blades may operate at angles of attack beyond their steady-state stall limits for significant lengths of time. The transient behavior of the blade section loads during this “dynamic stalling” process must be modeled as it is not captured by the steady load coefficient data input for each foil section. The primary effect of dynamic stalling is a delay in the appearance of stalled flow effects on blade loads to higher angles of attack than would be expected in steady flow.

Two models for dynamic stall effects on blade section loads are included in CACTUS. The modified Boeing-Vertol method of Gormont is the default. This algebraic method approximates dynamic stall effects with a “lagged” angle of attack, where the magnitude of the lag is empirically correlated to the angle of attack rate. The Leishman-Beddoes model incorporates more physical models and attempts to model the temporal evolution of dynamic stall flow phenomena and associated effects on blade loads. This model may provide more accurate results than the algebraic Boeing-Vertol method, but requires many more simulation time steps to be taken per turbine revolution to achieve converged results.

## 2.2 Wall Boundaries

CACTUS can simulate the effects of proximity to a ground plane or free (water) surface on turbine performance. The boundary conditions, either zero normal flow for a ground plane or constant surface pressure for a free surface, are applied using rectangular source panel elements. The free surface boundary condition is currently implemented as a quasi-static boundary, allowing it to respond only to the average flow created by the turbine and wake over a full revolution.

The user is allowed to specify the time step interval between updates to the wall panel system. For the free surface model, the wall update interval specifies the number of time steps between updates to the revolution averaged quantities. It is often not necessary to update these quantities on every time step. If it's possible to reduce the frequency of wall panel system updates and still obtain convergence of the simulation output of interest, this can reduce simulation run time considerably.

# 3 Normalization Parameters

Some definitions of parameters used in normalize CACTUS input and output parameters are given below.

| | |
|---|---|
| $\rho$ | Fluid density |
| $U_\infty$ | Freestream fluid flow speed |
| $A_T$ | Turbine reference area. Typically this reference area is chosen to be the projected frontal area of the volume swept by the rotor. |
| R | Turbine reference radius |
| $\omega$ | Turbine rotation rate |
| $U_{Loc}$ | Fluid flow speed local to an element. |
| $U_{Tip}$ | Turbine "tip" speed. Defined as $U_{Tip} = \omega R$. |
| c | Element chord length |
| $A_E$ | Element planform area |

In general, most parameters have been normalized at the machine scale. Unless otherwise noted in the input and output descriptions below, the force, torque, and power coefficients are normalized as

$$C_F = F \Big/ \tfrac{1}{2}\rho {U_\infty}^2 A_T$$

$$C_T = T \Big/ \tfrac{1}{2}\rho {U_\infty}^2 A_T R$$

$$C_P = P \Big/ \tfrac{1}{2}\rho {U_\infty}^3 A_T \quad (1)$$

# 4 Input Description

This section describes the namelist, geometry, and foil data input files required by CACTUS. CACTUS is run from the command line with the path to the namelist input file passed as the only argument. The geometry input file and the foil data table files are referenced in the namelist input file.

## 4.1 Geometry Input

This section describes the CACTUS turbine geometry input file format, and the tools provided for generating an input file for a generic turbine configuration. CACTUS considers the turbine rotor to be comprised of lifting-line blades and optional support struts. Both blades and struts are decomposed into elements. The primary difference between the two is that the strut elements do not shed a free vortex wake and use simplified empirical models for element loads.

When generating geometry, it should be noted that the nominal freestream flow direction in CACTUS is the +x direction. If a ground plane or free surface calculation is being performed, these surfaces are oriented with their normal vectors in the +y direction.

The CACTUS geometry input file is comprised of a header section and individual sections for each of the blades and struts. Each blade and strut section contains the details of the element decomposition. The parameters and file format are described below. Note that each parameter occupies one line in the geometry file and array valued parameters are input as component values separated by spaces.

### 4.1.1 File Parameters

#### 4.1.1.1 Header

| | |
|---|---|
| *NBlade* | Number of blades. |
| *NStrut* | Number of struts. |
| *RotN* | Turbine rotation axis normal vector (x y z values). |
| *RotP* | Turbine rotation origin point (x y z values). |
| *RefAR* | Turbine reference area (for force/torque/power normalization) divided by reference radius squared. |
| *RefR* | Turbine reference radius (reference length dimension) for scaling dimensional output values (ft). Corresponds to the reference radius used to normalize all geometry inputs below. |
| *Type* | String indicating the turbine geometry generation function used to create this file. This line is for reference only, and is not used internally in CACTUS. |

#### 4.1.1.2 Blade

| | |
|---|---|
| *Blade i* | Header indicating the blade number index. This line is for reference only, and is not used internally in CACTUS. |
| *NElem* | Number of elements. |
| *QCx* | Blade quarter-chord line x coordinates at element ends divided by reference radius (NElem + 1 values). |
| *QCy* | Blade quarter-chord line y coordinates at element ends divided by reference radius (NElem + 1 values). |
| *QCz* | Blade quarter-chord line z coordinates at element ends divided by reference radius (NElem + 1 values). |
| *nx* | Element unit normal vector x component at element ends (NElem + 1 values). |
| *ny* | Element unit normal vector y component at element ends (NElem + 1 values). |
| *nz* | Element unit normal vector z component at element ends (NElem + 1 values). |
| *tx* | Element unit tangent vector (rearward chord line direction) x component at element ends (NElem + 1 values). |
| *ty* | Element unit tangent vector (rearward chord line direction) y component at element ends (NElem + 1 values). |
| *tz* | Element unit tangent vector (rearward chord line direction) z component at element ends (NElem + 1 values). |
| *CtoR* | Element chord to turbine reference radius ratio (NElem + 1 values). |
| *AreaR* | Element area divided by turbine reference radius squared (NElem values). |
| *iSect* | Index of the foil section data to be applied to each element (NElem values). This index corresponds to the order in which the foil section data table files are supplied in the namelist input file. |

There should be a separate blade data block for each blade in the turbine rotor.

Note that the direction of the element normal vector defines the sign of the flow angle of attack calculated on that element and used to interpolate the foil data tables. Positive angle of attack will be calculated when the relative flow velocity component in the element normal direction is positive. Flow angle of attack is zero when the relative flow velocity is aligned with the element tangent vector.

#### 4.1.1.3 Strut

| | |
|---|---|
| *Strut i* | Header indicating the strut number index. This line is for reference only, and is not used internally in CACTUS. |
| *NElem* | Number of elements. |
| *SEx* | Strut mid-chord x coordinates at element ends divided by reference radius (NElem + 1 values). |
| *SEy* | Strut mid-chord x coordinates y coordinates at element ends divided by reference radius (NElem + 1 values). |

| | |
|---|---|
| *SEz* | Strut mid-chord x coordinates z coordinates at element ends divided by reference radius (NElem + 1 values). |
| *CtoR* | Element chord to turbine reference radius ratio (NElem + 1 values). |
| *AreaR* | Element area divided by turbine reference radius squared (NElem values). |
| *TtoC* | Strut thickness to chord ratio (single value, assumed constant over strut). |
| *BInd* | Index of the blade to which the strut connects. Used for blade-strut interference drag calculation. |
| *EInd* | Index of the element on the above blade at which the strut connects. Used for blade-strut interference drag calculation. |

There should be a separate strut data block for each strut in the turbine rotor.

Note that the strut model is currently fairly simplistic and assumes a radial strut attached to the turbine rotation axis at geometry start point, and to a particular blade at the geometry endpoint (typical of vertical axis wind turbines).

### 4.1.2 File Format Example

A file format example is given below.

---

```
NBlade:   2
NStrut:   2
RotN:   0.00000e+00   1.00000e+00   0.00000e+00
RotP:   0.00000e+00   0.00000e+00   0.00000e+00
RefAR:   3.52000e+00
RefR:   3.15000e+01
Type: VAWT
Blade 1:
        NElem:   4
        QCx:  -1.25936e-02  -1.25936e-02  -1.25936e-02  -1.25936e-02  -1.25936e-02
        QCy:   0.00000e+00   6.60000e-01   1.32000e+00   1.98000e+00   2.64000e+00
        QCz:  -0.00000e+00  -7.50000e-01  -1.00000e+00  -7.50000e-01  -0.00000e+00
        nx:   0.00000e+00   0.00000e+00   0.00000e+00   0.00000e+00   0.00000e+00
        ny:   8.34609e-01   6.03858e-01  -0.00000e+00  -6.03858e-01  -8.34609e-01
        nz:   5.50842e-01   7.97092e-01   1.00000e+00   7.97092e-01   5.50842e-01
        tx:   1.00000e+00   1.00000e+00   1.00000e+00   1.00000e+00   1.00000e+00
        ty:   0.00000e+00   0.00000e+00   0.00000e+00   0.00000e+00   0.00000e+00
        tz:   0.00000e+00   0.00000e+00   0.00000e+00   0.00000e+00   0.00000e+00
        CtoR:   7.40800e-02   7.40800e-02   7.40800e-02   7.40800e-02   7.40800e-02
        AreaR:   7.40096e-02   5.22828e-02   5.22828e-02   7.40096e-02
        iSect:   1   1   1   1
Blade 2:
        NElem:   4
        QCx:   1.25936e-02   1.25936e-02   1.25936e-02   1.25936e-02   1.25936e-02
        QCy:   0.00000e+00   6.60000e-01   1.32000e+00   1.98000e+00   2.64000e+00
```

```
        QCz:   1.54227e-18   7.50000e-01   1.00000e+00   7.50000e-01   1.54227e-18
        nx:   6.74587e-17   9.76156e-17   1.22465e-16   9.76156e-17   6.74587e-17
        ny:   8.34609e-01   6.03858e-01   0.00000e+00  -6.03858e-01  -8.34609e-01
        nz:  -5.50842e-01  -7.97092e-01  -1.00000e+00  -7.97092e-01  -5.50842e-01
        tx:  -1.00000e+00  -1.00000e+00  -1.00000e+00  -1.00000e+00  -1.00000e+00
        ty:   0.00000e+00   0.00000e+00   0.00000e+00   0.00000e+00   0.00000e+00
        tz:  -1.22465e-16  -1.22465e-16  -1.22465e-16  -1.22465e-16  -1.22465e-16
        CtoR:   7.40800e-02   7.40800e-02   7.40800e-02   7.40800e-02   7.40800e-02
        AreaR:   7.40096e-02   5.22828e-02   5.22828e-02   7.40096e-02
        iSect:   1   1   1   1
Strut 1:
        NElem:   4
        SEx:   0.00000e+00   0.00000e+00   0.00000e+00   0.00000e+00   0.00000e+00
        SEy:   1.32000e+00   1.32000e+00   1.32000e+00   1.32000e+00   1.32000e+00
        SEz:  -0.00000e+00  -2.50000e-01  -5.00000e-01  -7.50000e-01  -1.00000e+00
        CtoR:   7.40800e-02   7.40800e-02   7.40800e-02   7.40800e-02   7.40800e-02
        AreaR:   1.85200e-02   1.85200e-02   1.85200e-02   1.85200e-02
        TtoC:   1.50000e-01
        BInd:   1
        EInd:   3
Strut 2:
        NElem:   4
        SEx:   0.00000e+00  -3.06162e-17  -6.12323e-17  -9.18485e-17  -1.22465e-16
        SEy:   1.32000e+00   1.32000e+00   1.32000e+00   1.32000e+00   1.32000e+00
        SEz:   0.00000e+00   2.50000e-01   5.00000e-01   7.50000e-01   1.00000e+00
        CtoR:   7.40800e-02   7.40800e-02   7.40800e-02   7.40800e-02   7.40800e-02
        AreaR:   1.85200e-02   1.85200e-02   1.85200e-02   1.85200e-02
        TtoC:   1.50000e-01
        BInd:   2
        EInd:   3
```

### 4.1.3 Geometry Creation Tools

A set of MATLAB tools have been created to allow the user to generate a CACTUS geometry input file for an arbitrary turbine rotor. This set of MATLAB scripts is located in the *CreateGeom* folder in the CACTUS source code directory. Note that these scripts should also run without modification under GNU Octave (free software package that mimics much of the basic MATLAB syntax), with the possible exception of the plotting functions in the script *PlotTurbineGeom.m*.

The script *CreateTurbine.m* generates an empty turbine geometry structure and optionally fills it out with data for a parameterized generic turbine type. See the comments at the top of the file for details. The user can then modify the data in the created geometry structure as necessary to represent their particular problem.

Once the turbine geometry structure has been finalized, the script *WriteTurbineGeom.m* will create the CACTUS geometry input file. The script *ReadTurbineGeom.m* will read an existing CACTUS geometry input file and create a corresponding geometry structure in MATLAB for further modification.

The script *PlotTurbineGeom.m* will plot the geometry contained in a turbine geometry structure at a particular phase angle of rotation around the turbine rotation axis. The plotting functions in this script have not been verified to work in GNU Octave.

An example script that creates and plots turbine rotor geometry for a vertical axis wind turbine is provided in Appendix A.

## 4.2 Foil Table Data

Foil data table files contain the force coefficient data and dynamic stall parameters for a foil to be used in the calculation. The user can specify multiple foil data table files in the CACTUS namelist input file and they are indexed in the order they appear in the namelist input file. The *iSect* parameter in the geometry input file identifies the index of the foil data table to be applied to each blade element section.

The foil coefficients are input as lift, drag, and pitching moment (about the 25% chord point) coefficients (per span) as a function of angle of attack (AOA) and Reynolds number. A maximum of 20 Reynolds number values are allowed. A maximum of 1000 AOA values for each Reynolds number are allowed. The AOA data must go from -180 to 180 deg. The sign convention for AOA is such that positive AOA on a blade element is generated by positive relative flow velocity in the positive element normal direction. Note that the cross-flow turbine geometry generator provided with CACTUS creates element normal vectors in the machine inward direction. The axial flow turbine geometry generator creates element normal vectors in the machine rearward (downwind) direction at zero blade incidence angle.

Parameter definitions and a file format example are below. Note that a number of foil data table files are provided in the *Airfoil_Section_Data* folder in the CACTUS source code directory.

### 4.2.1 File Parameters

#### 4.2.1.1 Header

| | |
|---|---|
| *Title* | Title for this foil data table. For reference only; not used internally in CACTUS. |
| *Thickness to Chord Ratio* | Thickness to chord ratio for this foil section. |
| *Zero Lift AOA* | Angle of attack (deg) at zero lift for this section. |
| *Reverse Camber Direction* | This flag allows the user to reverse the orientation of a non-symmetric foil with respect to the normal vector of the blade element section to which it is applied (set the flag to 1). |

### 4.2.1.2 Reynolds Number Data Block

| | |
|---|---|
| *BV Dyn. Stall Model* | The Boeing-Vertol dynamic stall model uses reference angle of attack values to switch from a steady attached flow state to a dynamic stalled state. While these are denoted as "stall" AOA values, they should be set back from the foil stall AOA such that the lift coefficient is still fairly linear with AOA at this point. Generally, a value 50 - 75% of the way between the zero lift AOA and the stall AOA works well. |
| *LB Dyn. Stall Model* | The Leishman-Beddoes dynamic stall model uses a reference lift slope and a critical lift coefficient value to indicate the onset of leading edge stall. The critical lift coefficient value should be approximately equal to the value of lift coefficient that would have been obtained at the foil stall AOA, had the lift coefficient remained linear with AOA (with slope given by the reference lift coefficient slope). |
| *Force and Moment Coefficients* | Foil force and moment coefficient data at angle of attack from -180 to 180 deg. (AOA, lift coefficient, drag coefficient, pitching moment coefficient about the 25% chord point). |

There should be one Reynolds number data block for each Reynolds number at which foil data exists.

### 4.2.2 File Format Example

A file format example is given below.

---

Title: AFTitle
Thickness to Chord Ratio: 0.2
Zero Lift AOA (deg): 0.0
Reverse Camber Direction: 0

Reynolds Number: 1e6
BV Dyn. Stall Model - Positive Stall AOA (deg): 10
BV Dyn. Stall Model - Negative Stall AOA (deg): -10
LB Dyn. Stall Model - Lift Coeff. Slope at Zero Lift AOA (per radian): 6.28
LB Dyn. Stall Model - Positive Critical Lift Coeff.: 1.3
LB Dyn. Stall Model - Negative Critical Lift Coeff.: -1.3
AOA (deg) CL CD Cm25
-180.0 0.0 1.0 0.0
... ... ... ...
180.0 0.0 1.0 0.0

Reynolds Number: 5e6
...

---

## 4.3 CACTUS Namelist Input

This section describes the FORTRAN namelist input file for CACTUS. There are two namelist groups in the input file, *&ConfigInputs* and *&CaseInputs*.

### 4.3.1 Configuration Inputs

| | |
|---|---|
| **Regression Testing** | |
| *RegTFlag* | Set to 1 to perform a regression test (two iterations, generates _RegData.out output file), 0 for normal operation (default). |
| **Wall calculation** | |
| *GPFlag* | Set to 1 to use a ground plane, 0 otherwise. |
| *FSFlag* | Set to 1 to use a free surface, 0 otherwise. |
| *GPGridSF* | Factor on default ground plane grid spacing. |
| *FSGridSF* | Factor on default free surface near-field grid spacing. |
| **Calculation inputs** | |
| *nr* | Number of revolutions to perform (default 10) |
| *nti* | Number of time steps per revolution (default 16) |
| *convrg* | Convergence level for the revolution average power coefficient. (default no convergence check). Iteration will finish before *nr* revs if this level is hit. |
| *iut* | Number of iterations between wake convection velocity updates. If set to zero, the interval will be calculated automatically. If negative, wake convection velocities will be left at the values calculated at the time the wake element is created (no wake convection velocity updates). |
| *iWall* | Number of iterations between wall model updates (if wall calculation is active). |
| *ivtxcor* | 0 to use finite vortex core with width equal to maximum blade chord (default), 1 to not use core model. |
| *Incompr* | 1 to ignore any compressibility effects in models, 0 to include compressibility effects (default). |
| *vcrfb* | Factor on default bound vortex core radius. |
| *vcrft* | Factor on default trailing wake vortex core radius. |
| *vcrfs* | Factor on default spanwise wake vortex core radius. |

| | |
|---|---|
| *ifc* | 1 to use final convergence step, 0 to not (default). The discretization level is refined once/if initial convergence is reached before *nr* revolutions have been performed. |
| *nric* | Revolution number after which to switch to final convergence, if initial convergence level has not yet been achieved (optional). |
| *ntif* | Final number of time steps per revolution. This value will replace *nti* during final convergence. |
| *convrgf* | Final convergence level (default  0.0001). This level will replace convrg during final convergence. |
| *iutf* | Final number of iterations between wake updates. This value will replace *iut* during final convergence. Default behavior is the same as *iut*. |
| *ixterm* | 1 to ignore wake points beyond x/R = xstop, 0 to use all wake points (default). |
| *xstop* | If ixterm = 1, defines x/R beyond which wake points are ignored (default 5) |
| **Dynamic aero effects** | |
| *DSFlag* | 0 for no dynamic stall, 1 for Modified Boeing-Vertol model (default), 2 for Leishman-Beddoes model. |
| *PRFlag* | 0 for no element pitch rate aerodynamic effects, 1 to include these effects (default). |
| **Optional outputs** | |
| *Output_ELFlag* | 1 to output full detail element loads .csv file, 0 to omit this output (default) |
| *WallOutFlag* | 1 to output wall panel data (averaged over the last revolution), 0 to omit this output (default). |
| *DiagOutFlag* | 1 to output diagnostic info to standard output device each iteration, 0 to omit this output (default). |

### 4.3.2 Case Inputs

| | |
|---|---|
| *Jbtitle* | (string in single quotes) Job title. |
| **Operation point inputs** | |
| *RPM* | Rotor rotation rate (revs per minute). |
| *Ut* | Tip speed ratio with freestream flow speed (Utip/Uinf |
| *rho* | Density (slugs/ft^3) |
| *vis* | Dynamic viscosity (slugs/(ft*s)) |
| *tempr* | Temperature (degF) |

| | |
|---|---|
| *hBLRef* | Height of the effective freestream (99% freestream maybe) to be used in ground shear layer model (ft). |
| *slex* | Exponent for ground shear layer model (Ex. 1/2 for parabolic laminar BL model, 1/7 turbulent approx., 0 for constant freestream). |
| *hAG* | Height above ground at turbine geometry origin point (ft). Note that the ground plane is assumed to be oriented with its normal vector in the +y direction. |
| *dFS* | Depth below un-deflected free surface of the turbine geometry origin point (ft). Only used if free surface calculation is active. Note that the un-deflected free surface is assumed to be oriented with its normal vector in the +y direction. |
| **Geometry file** | |
| *GeomFilePath* | (string in single quotes) Path to turbine geometry input file. |
| **Airfoil section data** | |
| *nSect* | Number of airfoil section data tables to use (default 1) |
| *AFDPath* | (string in single quotes) Array (comma separated) of section data file path strings (size=nsect). |
| **Other parameters** | |
| *CDPar* | Additional parasitic interference drag coefficient based on "chord area" (chord squared) to be applied to the blade/strut interference drag calculation (default 0) |
| *CTExcrM* | Additional machine level excrescence torque coefficient, $C_{T,ExcrM} = T_{ExcrM} \big/ \frac{1}{2}\rho U_{\mathrm{Tip}}{}^2 R^3$. (default 0) |

### 4.3.3 File Format Example

A file format example is given below.

---

```
&ConfigInputs

    nr     = 10
    nti    = 16
    convrg = .0001
    iut    = 0

    DiagOutFlag = 1

/End
```

```
&CaseInputs

        jbtitle = 'Test VAWT'

        rho     = .002378
        vis     = .3739E-6
        tempr   = 60.0
        hBLRef          = 56.57
        slex    = 0.0
        hAG     = 15.0

        RPM     = 52.0
        Ut      = 5.0

        ! Turbine geometry
        GeomFilePath = './trunk/RegTest/TestVAWT.geom'

        ! Airfoil section data
        nSect   = 1
        AFDPath = './trunk/Airfoil_Section_Data/NACA_0015.dat'

/End
```

---

# 5 Output Description

This section describes the output files written by CACTUS.

## 5.1 Revolution Average Performance Data

Revolution averaged performance data for each revolution are written to a comma delimited file appended with *_RevData.csv*.

| | |
|---|---|
| *Rev* | Revolution number |
| *Power Coeff. (-)* | Revolution average machine power coefficient |
| *Tip Power Coeff. (-)* | Revolution average machine power coefficient normalized with $U_{Tip}$ instead of $U_\infty$ |
| *Torque Coeff. (-)* | Revolution average torque coefficient |
| *Fx Coeff. (-)* | Revolution average x component of force coefficient |
| *Fy Coeff. (-)* | Revolution average y component of force coefficient |
| *Fz Coeff. (-)* | Revolution average z component of force coefficient |
| *Power (kW)* | Revolution average machine power |

| | |
|---|---|
| *Torque (ft-lbs)* | Revolution average machine torque |
| *Delta CPU Time (s)* | CPU time for each revolution |
| *Total CPU Time (s)* | Total CPU time after each revolution |

## 5.2 Temporal Performance Data

Performance data for each time step are written to a comma delimited file appended with *_TimeData.csv*.

| | |
|---|---|
| *Normalized Time (-)* | Normalized simulation time $t_N = t\,\frac{U_\infty}{R}$ |
| *Theta (rad)* | Turbine rotational phase angle |
| *Rev* | Revolution number |
| *Torque Coeff (-)* | Torque coefficient |
| *Power Coeff (-)* | Power coefficient |
| *Fx Coeff. (-)* | X component of force coefficient |
| *Fy Coeff. (-)* | Y component of force coefficient |
| *Fz Coeff. (-)* | Z component of force coefficient |
| | |
| The following are replicated for each blade | |
| *Blade Fx Coeff (-)* | Contribution to x component of force coefficient from blade |
| *Blade Fy Coeff (-)* | Contribution to y component of force coefficient from blade |
| *Blade Fz Coeff (-)* | Contribution to z component of force coefficient from blade |
| *Blade Torque Coeff (-)* | Contribution to torque coefficient from blade |
| | |
| The following are replicated for each strut | |
| *Strut Fx Coeff (-)* | Contribution to x component of force coefficient from strut |
| *Strut Fy Coeff (-)* | Contribution to y component of force coefficient from strut |
| *Strut Fz Coeff (-)* | Contribution to z component of force coefficient from strut |
| *Strut Torque Coeff (-)* | Contribution to torque coefficient from strut |

## 5.3 Temporal Element Loads Data

When *Output_ELFlag* is set to 1 in the namelist input file, element loads data for each time step are written to a comma delimited file appended with *_ElementData.csv*.

| | |
|---|---|
| *Normalized Time (-)* | Normalized simulation time $t_N = t\,\frac{U_\infty}{R}$ |
| *Theta (rad)* | Turbine rotational phase angle |

| | |
|---|---|
| *Blade* | Blade number |
| *Element* | Element number |
| *Rev* | Revolution number |
| *AOA (deg)* | Element local flow angle of attack |
| *Re (-)* | Element Reynolds number based on element chord |
| *Mach (-)* | Element Mach number |
| *Ur (-)* | Local flow speed ratio with freestream $U_r = {U_{Loc}}/{U_\infty}$ |
| *CN (-)* | Element normal force coefficient, $C_N = N \Big/ \frac{1}{2}\rho {U_{Loc}}^2 A_E$ |
| *CT (-)* | Element tangential force coefficient, $C_T = T \Big/ \frac{1}{2}\rho {U_{Loc}}^2 A_E$ |
| *Fx (-)* | Contribution to x component of force coefficient from element |
| *Fy (-)* | Contribution to x component of force coefficient from element |
| *Fz (-)* | Contribution to x component of force coefficient from element |
| *te (-)* | Contribution to torque coefficient from element |

## 5.4 Wall Model Data

When a wall calculation is being performed and *WallOutFlag* is set to 1 in the namelist input file, summary output data for the wall calculation is written to a comma delimited file appended with either *_GPData.csv* for a ground plane calculation, or *_FSData.csv* for a free surface calculation.

### 5.4.1 Ground Plane Data

| | |
|---|---|
| *X/R (-)* | X location of the panel center normalized by R |
| *Y/R (-)* | Y location of the panel center normalized by R |
| *Z/R (-)* | Z location of the panel center normalized by R |
| *SourceDens/Uinf (-)* | Source density on the panel normalized by $U_\infty$ |

### 5.4.2 Free Surface Data

| | |
|---|---|
| *X/R (-)* | X location of the panel center normalized by R |
| *Y/R (-)* | Y location of the panel center normalized by R |
| *Z/R (-)* | Z location of the panel center normalized by R |
| *U/Uinf (-)* | Wall tangential velocity (nominal freestream direction) normalized by $U_\infty$ |

| | |
|---|---|
| *dH/R (-)* | Free surface height (above un-deflected height) normalized by R |

# 6 References


1. Murray, J., Barone, M., "The Development of CACTUS, a Wind and Marine Turbine Performance Simulation Code," AIAA Paper 2011-147, 2011.

2. Strickland, J. H., Smith, T., Sun, K., "A Vortex Model of the Darrieus Turbine: An Analytical and Experimental Study," Sandia National Laboratories, SAND81-7017, Albuquerque, NM, 1981.

3. Katz, J., Plotkin, A., Low Speed Aerodynamics, 2nd ed., Cambridge University Press, Cambridge, England, UK, 2001.

# Appendix A: Example Geometry Script

Below is an example geometry creation MATLAB script for a vertical axis wind turbine.

---

```
clear
close all

% Creates test VAWT geometry file

% Params
R=31.5;             % Center radius (ft)
HR=2.64;            % Height to radius ratio
CRr=0.07408;         % Root chord to radius
eta=.42;              % Blade mount point ratio (mount point behind leading
edge as a fraction of chord)
NBlade=2;
NBElem=5;
NStrut=0;       % number of struts
NSElem=5;
CRs=CRr;        % strut chord to radius
TCs=.15;        % strut thickness to chord

% Output filename
FN='TestVAWT.geom';

% Plot data?
PlotTurbine=1;

% Convert
dToR=pi/180;

% Create basic parabolic blade VAWT
Type='VAWT';
BShape=1;
T=CreateTurbine(NBlade,NBElem,NStrut,NSElem,R,[],[],[],Type,1,CRr,HR,eta,BSha
pe,CRs,TCs);

% Write geom file
WriteTurbineGeom(FN,T);

% Plot if desired
if PlotTurbine

    % Plot animated turbine rotation
    XLim=[-4,4];
    YLim=[-2,4];
    ZLim=[-4,4];

    % Plot element normals
    PlotVec=1;
    SFVec=.5;
```

```
    hf=figure(1);
    set(hf,'Position',[303   124   956   610])
    set(gca,'Position',[5.2743e-002  5.1245e-002  8.9979e-001  8.8141e-001])
    set(gca,'CameraPosition',[-52.1999   30.4749   62.2119])
    set(gca,'CameraUpVector',[1.8643e-001  9.7433e-001 -1.2615e-001])
    set(gca,'CameraViewAngle',6.3060e+000)
    grid on
    set(gcf,'Color','white');
    hl=light('Position',[-1,0,0]);
    set(gca,'Color','white');
    set(gca,'DataAspectRatio',[1,1,1])
    set(gca,'XLim',XLim,'YLim',YLim,'ZLim',ZLim)

    HIn=[];
    PhasePlot=linspace(0,2*pi,150);
    for i=1:length(PhasePlot)
       H=PlotTurbineGeom(T,hf,PhasePlot(i),HIn,PlotVec,SFVec);
       HIn=H;
       pause(.01);
    end

end
```