# AeroDyn Meshes and Related Calculations

## Rotation Notation/Convention

 (from global to local)

or equivalently:



where *X*/*Y*/*Z* are global coordinates, *x*/*y*/*z* are local coordinates,  is the DCM from global to local, and // are the unit vectors of the local coordinate system expressed in the global coordinate system.



where function returns the 3 Euler angles of the *x*-*y*-*z* (1-2-3) rotation sequence used to form  (that is, first a rotation  about the global *X* axis, followed by rotation  about the *Y’* axis, followed by rotation  about the *Z’’* axis ) defined as follows:

Note the following simplifications:







## AeroDyn Reference Meshes

*Tower*

 = reference position of the *j*th node in the tower

 = reference orientation of the *j*th node in the tower

*Hub*

 = reference position of the hub (sent to AeroDyn from Driver at initialization)

 = reference orientation of the hub (sent to AeroDyn from Driver at initialization)

The AeroDyn standalone driver will initialize these as follows:





Note:  is defined about the negative *yn* axis.

*Blade*

 = reference position of the node at the root of the *k*th blade (sent to AeroDyn from Driver at initialization)

 = reference orientation of the node at the root of the *k*th blade (sent to AeroDyn from Driver at initialization)

The AeroDyn standalone driver will initialize these as follows:





 = reference position of the *j*th node in the *k*th blade

 = reference orientation of the *j*th node in the *k*th blade

Note:  is defined about the positive *yb* axis and  is defined about the negative local *zb* axis; there is no rotation about the *xb* axis because sweep is assumed to be sheared.

## Initialization Inputs to BEMT from AeroDyn:

 = hub radius (along the blade) of the *k*th blade

 = local radius (along the blade) of the *j*th node in the *k*th blade

 = tip radius (along the blade) of the *k*th blade

where  represents the magnitude (2-norm) of vector 

## Inputs to AeroDyn from Driver:

*Tower*

 = translational displacement (relative) of the *j*th node in the tower

 = displaced rotation (absolute orientation) of the *j*th node in the tower

 = translational velocity (absolute) of the *j*th node in the tower

 = rotational velocity (absolute) of the *j*th node in the tower

 = undisturbed wind speed computed at 

The AeroDyn standalone driver will set these as follows:









*Hub*

 = translational displacement (relative) of the hub

 = displaced rotation (absolute orientation) of the hub

 = translational velocity (absolute) of the hub

 = rotational velocity (absolute) of the hub

The AeroDyn standalone driver will set these as follows:







*Blade*

 = translational displacement (relative) of the node at the root of the *k*th blade

 = displaced rotation (absolute orientation) of the node at the root of the *k*th blade

 = translational velocity (absolute) of the node at the root of the *k*th blade

 = rotational velocity (absolute) of the node at the root of the *k*th blade

 = translational displacement (relative) of the *j*th node in the *k*th blade

 = displaced rotation (absolute orientation) of the *j*th node in the *k*th blade

 = translational velocity (absolute) of the *j*th node in the *k*th blade

 = rotational velocity (absolute) of the *j*th node in the *k*th blade

 = undisturbed wind speed computed at 

The AeroDyn standalone driver will set these as follows:











## AeroDyn Tower Influence (Tower Potential Flow and Tower Shadow):

Note: the tower-influence models (tower potential flow and tower shadow) are only valid for small tower deflections; so, first throw an error to avoid a division-by-zero error below if for any Line2 element of the tower mesh, the nodes are collocated whereby . For each Line2-element node of the blade mesh, a nearest-neighbor Line2 element or node of the tower mesh is then found in the deflected configuration. There are 3 possibilities (1, 2a, and 2b in the order of precedent) presented below.

Next, a local tower coordinate system at the nearest tower point is defined such that  is directed along the tower axis; / is the plane normal to the tower axis; the local relative wind speed, , is in the / plane with positive  along positive ; and  is transverse to this plane. Next, find the local position vector to the *j*th node in the *k*th blade from the nearest tower point, ; the local // components of  normalized by the tower radius, //; the local relative wind speed normal to the tower axis, ; the local tower diameter, ; and the local tower drag coefficient, . Trigger an error (indicating a tower strike) if .

1. Find the nearest-neighbor Line2 element of the tower mesh for which the blade Line2-element node projects orthogonally onto the tower Line2-element domain (following an approach similar to the Line2\_to\_Line2 mapping search for motion and scalar quantities). That is, for each node of the blade mesh, an orthogonal projection is made onto all possible Line2 elements of the tower mesh and the Line2 element of the tower mesh that is the minimum distance away is found, calculated as distance in the deflected configuration, , where:



 with 

Then:



 (the rest of the equations can be skipped if , because then  and there is no tower influence)











1. Find the nearest-neighbor node () in the tower Line2-element domain (following an approach similar to the Point\_to\_Point mapping search for motion and scalar quantities). That is , for each node of the blade mesh, the node of the tower mesh () that is the minimum distance away is found, calculated as distance in the deflected configuration, , where:



1. If the tower node is found within the tower () (i.e., if the tower node is not at the top or bottom of the tower), then:



 (the rest of the equations can be skipped if , because then  and there is no tower influence)



 (ignore the potentially small residual of )







1. If the tower node is found on top or bottom of the tower ( or ), then:



 (the rest of the equations can be skipped if , because then  and there is no tower influence)



 (the COS() is used to taper off the tower influence above and below the tower;  and  is not used for )







Finally, the tower influence is calculated as follows:



where:

 = axial and transverse velocity deficit fractions from tower potential flow

 = axial velocity deficit fraction from tower shadow

## Inputs to BEMT from AeroDyn:

Note: a local disk coordinate system is defined such that  is normal to the disk, / are in the disk,  is in the / plane with positive  along negative , and  is normal to this plane (the azimuth angle is defined relative to this ) as follows:

 where:

 = relative wind speed of the *j*th node in the *k*th blade

 = disk-averaged relative wind speed

*Hub*

 = rotor speed

 = inflow skew angle

*Blade*

 = azimuth angle of the *k*th blade

Note: first a coordinate system is formed that is equivalent to , but without the blade-pitch angle:



Next, a coordinate system is formed that is equivalent to , but without the live sweep (due to in-plane deflection), blade-pitch and twist (aerodynamic + elastic) angles:



 = local pitch + twist (aerodynamic + elastic) angle of the *j*th node in the *k*th blade

 = local radius (normalized distance from the rotor centerline) of the *j*th node in the *k*th blade

where

 = displaced position of the *j*th node in the *k*th blade relative to the hub

 = normal component (normal to the plane, not chord) of the inflow velocity of the *j*th node in the *k*th blade

 = tangential component (tangential to the plane, not chord) of the inflow velocity of the *j*th node in the *k*th blade

## Outputs to AeroDyn from BEMT:

 = inflow velocity of the *j*th node in the *k*th blade

 = inflow angle of the *j*th node in the *k*th blade

 = normal force coefficient (normal to the plane, not chord) of the *j*th node in the *k*th blade

 = tangential force coefficient (tangential to the plane, not chord) of the *j*th node in the *k*th blade

 = pitching moment coefficient of the *j*th node in the *k*th blade

## Outputs from AeroDyn to Driver:

*Tower*

 = force per unit length of the *j*th node in the tower

 = moment per unit length of the *j*th node in the tower

where:

 = relative wind speed of the *j*th node in the tower

 = relative local *x*-component of wind speed of the *j*th node in the tower

 = relative local *y*-component of wind speed of the *j*th node in the tower

 = relative wind speed normal to the tower at node *j*

 = local *x*-component of force per unit length of the *j*th node in the tower

 = local *y*-component of force per unit length of the *j*th node in the tower

*Blade*

 = force per unit length of the *j*th node in the *k*th blade

 = moment per unit length of the *j*th node in the *k*th blade

where:

 = dynamic pressure of the *j*th node in the *k*th blade

 = normal force per unit length (normal to the plane, not chord) of the *j*th node in the *k*th blade

 = tangential force per unit length (tangential to the plane, not chord) of the *j*th node in the *k*th blade

 = pitching moment per unit length of the *j*th node in the *k*th blade

## Write Output

This is a list of all possible output parameters for the AeroDyn module. The names are grouped by meaning, but can be ordered in the OUTPUTS section of the AeroDyn input file as you see fit. BαNβ, refers to output node β of blade α, where α is a number in the range [1,3] and β is a number in the range [1,9], corresponding to entry β in the *BlOutNd* list. TwNβ refers to output node β of the tower and is in the range [1,9], corresponding to entry β in the *TwOutNd* list.

|  |  |  |
| --- | --- | --- |
| Channel Name(s) | Units | Description |
| *Tower* | | |
| TwNβVUndx, TwNβVUndy, TwNβVUndz | (m/s), (m/s), (m/s) | Undisturbed wind velocity at TwNβ in the local tower coordinate system |
| TwNβSTVx, TwNβSTVy, TwNβSTVz | (m/s), (m/s), (m/s) | Structural translational velocity at TwNβ in the local tower coordinate system |
| TwNβVrel | (m/s) | Relative wind speed at TwNβ |
| TwNβDynP | (Pa) | Dynamic pressure at TwNβ |
| TwNβRe | (-) | Reynolds number (in millions) at TwNβ |
| TwNβM | (-) | Mach number at TwNβ |
| TwNβFdx, TwNβFdy | (N/m), (N/m) | Drag force per unit length at TwNβ in the local tower coordinate system |
| *Blade* | | |
| BαAzimuth | (deg) | Azimuth angle of Bα |
| BαPitch | (deg) | Pitch angle of Bα |
| BαNβVUndx, BαNβVUndy, BαNβVUndz | (m/s), (m/s), (m/s) | Undisturbed wind velocity at BαNβ in the local blade coordinate system |
| BαNβVDisx, BαNβVDisy, BαNβVDisz | (m/s), (m/s), (m/s) | Disturbed wind velocity at BαNβ in the local blade coordinate system |
| BαNβSTVx, BαNβSTVy, BαNβSTVz | (m/s), (m/s), (m/s) | Structural translational velocity at BαNβ in the local blade coordinate system |
| BαNβVrel | (m/s) | Relative wind speed at BαNβ |
| BαNβDynP | (Pa) | Dynamic pressure at BαNβ |
| BαNβRe | (-) | Reynolds number (in millions) at BαNβ |
| BαNβM | (-) | Mach number at BαNβ |
| BαNβVIndx, BαNβVIndy | (m/s), (m/s) | Axial and tangential induced wind velocity at BαNβ |
| BαNβAxInd, BαNβTnInd | (-), (-) | Axial and tangential induction factors at BαNβ |
| BαNβAlpha, BαNβTheta, BαNβPhi, BαNβCurve | (deg), (deg), (deg), (deg) | Angle of attack, pitch+twist angle, inflow angle, and curvature angle at BαNβ |
| BαNβCl, BαNβCd, BαNβCm,  BαNβCx, BαNβCy, BαNβCn, BαNβCt | (-), (-), (-),  (-), (-), (-), (-) | Lift force, drag force, pitching moment, normal force (to plane), tangential force (to plane), normal force (to chord), and tangential force (to chord) coefficients at BαNβ |
| BαNβFl, BαNβFd, BαNβMm,  BαNβFx, BαNβFy, BαNβFn, BαNβFt | (N/m), (N/m), (N·m/m),  (N/m), (N/m), (N/m), (N/m) | Lift force, drag force, pitching moment, normal force (to plane), tangential force (to plane), normal force (to chord), and tangential force (to chord) per unit length at BαNβ |
| *Rotor* | | |
| RtSpeed | (rpm) | Rotor speed |
| RtTSR | (-) | Rotor tip-speed ratio |
| RtVAvgxh, RtVAvgyh, RtVAvgzh | (m/s), (m/s), (m/s) | Rotor-disk-averaged relative wind velocity in the hub coordinate system |
| RtSkew | (deg) | Rotor inflow-skew angle |
| RtAeroFxh, AeroFyh, AeroFzh,  RtAeroMxh, AeroMyh, AeroMzh | (N), (N), (N)  (N·m), (N·m), (N·m) | Total rotor aerodynamic load in the hub coordinate system |
| RtAeroPwr | (W) | Rotor aerodynamic power |
| RtArea | (m2) | Rotor swept area |
| RtAeroCp, RtAeroCq, RtAeroCt | (-), (-), (-) | Rotor aerodynamic power, torque, and thrust coefficients |

*Tower*















*Blade*



























*Rotor*









, where the integral is computed via a line-to-point mapping transfer of  to the hub.

, where the integral is computed via a line-to-point mapping transfer of  and  (with moment arm ) to the hub.



