

**TNO report**

## **ECN Aero-Module**

### **User's manual, v252**

Date	14 June 2019
Author(s)	K. Boorsma, M. Caboni
Number of pages	78 (incl. appendices)
Number of appendices	1

All rights reserved.

No part of this publication may be reproduced and/or published by print, photoprint, microfilm or any other means without the previous written consent of TNO.

In case this report was drafted on instructions, the rights and obligations of contracting parties are subject to either the General Terms and Conditions for commissions to TNO, or the relevant agreement concluded between the contracting parties. Submitting the report for inspection to parties who have a direct interest is permitted.

©2019 Copyright TNO

# Executive Summary

The wind turbine community is in need of more sophisticated tools for evaluating aerodynamic blade loads. The predictive capability of the current practice of BEM modelling falls short for e.g. yawed flow and dynamic inflow cases. In addition to that, the variety in the several engineering extensions used between different BEM implementations is huge. A better approach for modelling the rotor aerodynamics is presented by the free vortex wake code AWSM. This approach includes more physics, however the resulting computations are more time consuming. The major assumption used in both aerodynamic models lies in the use of two-dimensional aerodynamic profile coefficients as an input.

ECN part of TNO has assembled the current state of the art of the above mentioned aerodynamic models in the ECN Aero-Module. Care has been taken to allow for flexibility between modelling choices. The package is to be coupled to arbitrary simulation software that solves the structural dynamics of a wind turbine.

The first part of this document briefly describes the included models and their background. The second part consists of the user's manual. The third part gives a validation using experimental results.

**Acknowledgement** Financial support for creating the ECN Aero-Module was given in part by the national INNWIND programme. Additional support was given by the EU UPWIND and EU AVATAR programmes.

# Contents

1	Introduction .....	4
2	Model description .....	6
2.1	Environment .....	6
2.2	Airfoil data .....	7
2.3	Structural model .....	12
2.4	BEM.....	13
2.5	AWSM .....	16
3	Usage of software .....	19
3.1	Coordinate systems.....	19
3.2	Input .....	22
3.3	Output .....	40
4	Validation .....	44
4.1	NREL UAE Phase VI .....	45
4.2	Mexico.....	62
A	Usage of AWSM for external field analysis .....	75
A.1	The input file <code>inextpoints.txt</code> .....	75
A.2	The output file <code>extfield&lt;&gt;.dat</code> .....	77

# 1 Introduction

The wind turbine community is in need of more sophisticated tools for evaluating aerodynamic blade loading. The predictive capability of the current practice of BEM modelling falls short for e.g. yawed flow and dynamic inflow cases. In addition to that, the variety in the several engineering extensions used between different BEM implementations is huge. A more accurate approach for modelling the rotor aerodynamics is presented by a free vortex wake model. This approach includes more physics, however the resulting computations are more time consuming.

ECN part of TNO has assembled the current state of the art of the above mentioned aerodynamic models in the ECN Aero-Module. The package is to be coupled to arbitrary simulation software that solves the structural dynamics of a wind turbine. Blade position and velocity are given as an input to the ECN Aero-Module and forces and moments are then communicated back to the structural code. In addition to the coupled option, it is possible to run the software stand-alone for the purpose of aerodynamic calculations on a rigid turbine. An overview of the resulting program content is given in Figure 1.

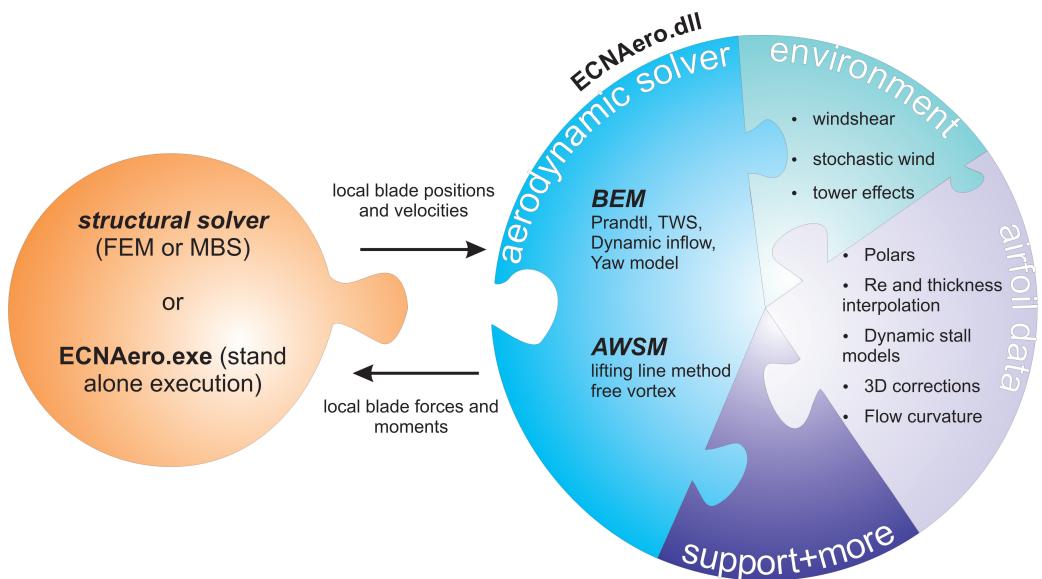


Figure 1: Overview of the program content

The two aerodynamic models included in the ECN Aero-Module are the Blade Element Momentum (BEM) method similar to the implementation in PHATAS [18] and a free vortex wake code in the form of AWSM [29]. Both of these models need feeding from the aerodynamic coefficients (lift, drag and moment polars) of the airfoils present in the wind turbine blades. In addition to that, both of these models need information on the wind field and are influenced by the proximity of the tower.

The ECN Aero-Module is written using object oriented FORTRAN language and is divided into several modules. This modular program set-up allows for easy variation between and transparency of aerodynamic models. It is the aim of ECN part of TNO to give the user maximum flexibility and offer a wide range of aerodynamic modelling options to choose from. Hence the trade-off between accuracy and computation time is the responsibility of the user. In addition to that, this approach enables the use of the same models for tower effects, wind excitation (stochastic or deterministic wind fields) and airfoil data (including dynamic stall and rotational effects) for both the BEM and AWSM approach.

The first part of this document briefly describes the included models and their background. The second part consists of the user's manual. The third part gives a validation using experimental results.

## 2 Model description

The model description is discussed separately for each module as illustrated in Figure 1. The external conditions are firstly described in the environment module. The data and procedures related to the input of two-dimensional aerodynamic profile coefficients are addressed thereafter in the airfoil data module. Also the handling of information coming from the structural model is discussed. Finally, the aerodynamic models are briefly described in the section on BEM and AWSM.

### 2.1 Environment

This module describes the external conditions to the wind turbine rotor. The environment module includes a part that describes the wind and a part that describes the tower.

#### 2.1.1 Wind

A relatively simple wind model is implemented that requires the input of hub height wind speed in three directions, changeable in time. Horizontal and vertical wind shear can be added to this wind field if desired. A description of the input file format is given in section 3.2.2.

In addition to this simple wind field it is possible to read in a SWIFT [30], TurbSim [13] or Mann [2] file, that includes a stochastic wind field, varying in space and time. These programs write a binary file including all three components of the velocity vector after specifying the appropriate IEC standard, turbulence intensity and average wind speed at hub height. Horizontal and vertical wind shear can be included as well. A full 3D interpolation of the wind field using Taylor's hypothesis of frozen turbulence [20] is implemented to obtain the actual wind speed at the required position.

The wind field is defined with respect to the fixed inertial coordinate system.

#### 2.1.2 Tower

The presence of the tower induces velocities in the rotor plane, dependent on the azimuth position of the blades. The dipole model from PHATAS [18] for upwind placed rotors featuring a cylindrical tower has been implemented. The tower induced velocities are taken into account for the local blade velocities only and do not influence the freestream wind velocity used in the BEM momentum equations. A linearly varying tower diameter is assumed between tower top and tower base.

Since the tower induced velocity depends on the wind velocity incident on the tower (which includes the rotor induced velocity), the tower effect is recalculated for every iteration that is needed to solve the BEM or AWSM equations. An approximation lies in the fact that the tower induced velocities for each element are calculated using the tower diameter and tower incident wind velocity at a tower height equal to the element height at 180 degree azimuth (blade pointing downward). Using the towerbase and towertop coordinates, the appropriate location of the tower is determined by linear interpolation.

In the current version of ECN Aero-Module, the tower effect is disabled for VAWTs.

## 2.2 Airfoil data

The aerodynamic calculation requires specification of the lift, drag and moment coefficients of the airfoils used in the blades. A special module exists within the ECN Aero-Module that deals with the procedures related to obtaining and processing the airfoil data. The required airfoil data may be obtained from wind tunnel tests, two-dimensional computations or the Aerodynamic Table Generator (ATG) of ECN part of TNO [8].

### 2.2.1 Spline functionality

A purposely built interpolation module is included to interpolate airfoil characteristics between segments instead of assuming a constant airfoil for each section. The interpolated airfoil characteristics are then determined using the specified thickness distribution along the blade. Interpolation in the lookup table is also performed using this spline functionality. Using the INTERPOL keyword, it is possible to choose between a linear and spline interpolation method.

In addition to this, it is possible to specify a Reynolds number for each polar. Hence multiple polars with different Reynolds numbers can be defined for each profile. The profile characteristics used for calculating the aerodynamic forces are then interpolated for the appropriate Reynolds number, thereby including a Reynolds number dependence.

### 2.2.2 Angle of attack evaluation point

For an airfoil with a uniform camber line, it can be shown using thin airfoil theory that for a fully laminar flow the strength of the circulation (lift) is such that the flow at the three-quarter chord location is in the direction of the camber line. This was previously outlined by Lindenburg [17]. Based on this, the angle of attack in curved flow (e.g. a coned rotor, torsional flexibility) should be evaluated using the flow direction at the three-quarter chord location. In the ECN Aero-Module it is possible to choose where the angle of attack is evaluated for usage in the airfoil lookup tables using the AOAEval keyword. This option is active for both BEM and AWSM. In AWSM only the velocities associated with blade motion are evaluated at the three-quarter chord location.

The chosen method does not only have consequences for the evaluation of structural velocities, but also for the wind and tower induced velocities. In addition to that, a moment coefficient correction is applied if the angle of attack evaluation point is set to the three-quarter chord location.

$$\Delta Cm = -\frac{\pi}{8} \frac{\alpha_{34} - \alpha_{14}}{0.5} \cos(\alpha_{14}) , \quad (2.1)$$

with

$\Delta Cm$	[ $-$ ]	moment coefficient correction
$\alpha_{34}$	[rad]	three-quarter chord based angle of attack
$\alpha_{14}$	[rad]	quarter chord based angle of attack

The effective velocity that is used to obtain forces from the coefficients is taken from the components at the quarter chord position, irrespective of the choice of AOAEval. The angle of attack that is used to decompose the obtained lift and drag forces in rotorplane direction is also evaluated at the quarter chord location irrespective of this choice.

### 2.2.3 3D airfoil coefficient correction method

The 3D correction model as developed by Snel [19] to account for the effects of rotation on the coefficients (noticeable in the inboard region), has been implemented. Since the airfoil coefficients

are composed of a weighted average from two data sets, the application of the 3D correction must be applied after this weighting.

The correction is applied to the lift coefficient and takes the following form.

$$Cl_{3D} = 3.1 \left( \frac{c}{r} \right)^2 \left( \frac{\Omega r}{U_{eff}} \right)^2 (Cl_{pot} - Cl_{2D}) , \quad (2.2)$$

with

$Cl_{3D}$	[ $-$ ]	corrected 3D lift coefficient
$Cl_{2D}$	[ $-$ ]	2D lift coefficient
$c$	[m]	local chord
$r$	[m]	local radius in rotorplane
$\Omega$	[rad/s]	rotational speed of rotor
$U_{eff}$	[m/s]	effective incoming velocity at element
$Cl_{pot}$	[ $-$ ]	potential flow lift coefficient
$\alpha$	[rad]	local angle of attack
$\alpha_0$	[rad]	zero lift angle of attack

The correction is applied for angles of attack below 50 degrees and is linearly faded to zero between 30 and 50 degrees.

## 2.2.4 Dynamic stall models

Dynamic stall models account for the fact that the local state at each airfoil section is not able to respond instantaneously to changing conditions like turbulence, blade deformation or tower effects. Especially for separated flow conditions, the dynamic value of the airfoil coefficients will differ from the static look-up value. Several models exist to correct for these effects and the models implemented in the ECN Aero-Module are summarized below.

1. Snel first order [26]
2. Snel second order [26]
3. Beddoes-Leishmann [15, 16]

### Implementation of Snel first and second order models

These models account for dynamic effects in separated flow and are especially developed for application in a wind turbine environment. Its numerical implementation does not require inputting airfoil specific parameters, which makes it popular for application in an aero-elastic code. The first order model consists of a linear differential equation for the dynamic lift increment, driven by the difference with the potential flow lift and angle of attack history. The second order model includes the first order model and adds higher frequency dynamics of self excited nature. More details can be found in [26]. The implementation of the Snel first and second order dynamic stall routines have been verified against the implementation in PHATAS.

### Implementation of Beddoes-Leishmann model

The emphasis of the Beddoes-Leishmann method lies on a more complete physical representation of the overall unsteady aerodynamics. Although originating from helicopter research, it is intended as a general model for airfoils undergoing dynamic inflow conditions. The model essentially consists of four subsystems:

1. Attached flow model, including an impulsive component and circulatory (shed vorticity) component.

2. A trailing edge separation module to include the effect of the dynamic separation point position.
3. A leading edge separation module to predict dynamic stall onset. Leading edge separation lag is included.
4. A vortex module to calculate the possible vortex induced aerodynamic forces.

These four sub-models are connected in an open loop system, where output from one model serves as input for the next. More details can be found in [15, 16]. Here it is noted that the model uses the potential flow lift curve slope to determine the steady part of the circulatory lift component, with the consequence that also in steady attached flow conditions the used lift and drag coefficients can deviate from the specified static polars. The specific implementation in the ECN Aero-Module is based on the work described in [12, 14, 22], and [3]. This implementation is briefly described below.

- Wind reference frame

As suggested in [3] the wind reference frame is used instead of the body fixed reference frame (normal and tangential force coefficients). For low angles of attack the difference obviously is small. The only exception lies in the vortex lift, which is calculated in terms of  $C_n$  and decomposed in the wind reference frame afterwards.

- Angle of attack

The angles of attack are modified to ensure the angle of attack is limited between  $-90^\circ$  and  $90^\circ$ :

$$\begin{aligned} \alpha_{mod} &= \alpha - 180 & \alpha > +90 \\ \alpha_{mod} &= \alpha + 180 & \alpha < -90 \end{aligned}, \quad (2.3)$$

with

$\alpha_{mod}$	[°]	Modified angle of attack for usage in Beddoes-Leishman model
$\alpha$	[°]	Original angle of attack

In addition to this a fading function ( $\cos^2 \alpha$ ) is applied to angle of attack variations that are fed into the model. This in order to avoid queer results at high angles of attack.

- Parameters

Several parameters are used throughout the model. Some depend on airfoil shape, most of them are time constants. The default values used in the ECN Aero-Module are given in Table 1, although they can be customized using the relevant keywords.

Table 1: Implemented default Beddoes-Leishmann parameter values as defined in [15, 16]

$A_1$	$A_2$	$b_1$	$b_2$	$K_\alpha$	$T_p$	$T_f$	$T_v$	$T_{vl}$	$A_{cd}$
0.3	0.7	0.14	0.53	0.75	1.5	5	6	5	0.13

- Shed vorticity

As mentioned, the influence of shed vorticity on the profile aerodynamics is included in the Beddoes-Leishman model. Since this effect is accounted for by AWSM as well, this contribution is excluded for AWSM simulations by not correcting the equivalent angle of attack with deficiency functions.

To incorporate the effects of time-varying flow velocity into the attached flow module of the Beddoes-Leishman model, the downwash instead of angle of attack at the three quarter

chord location is used in the deficiency functions to model the effect of shed vorticity in the wake.

- Potential flow lift curve slope

For non-cylindrical shapes ( $t/c < 0.8$ )), the potential lift curve slope is determined depending on the LINREG keyword, i.e. either using linear regression or a constant value of  $2\pi$ . For a relative thickness above ( $t/c > 0.8$ )), the potential lift curve slope is set to 0.

- Trailing edge separation module

As it was found that the Kirchoff flow separation module gave erroneous results for high angles of attack, some modifications were implemented [14]. Several options are given to the user as governed by the FSMETHOD keyword, of which the FFA implementation [3] is taken as default.

- Leading edge stall criterium

As suggested in [22], the occurrence of leading edge stall is determined by monitoring the force coefficient compared to its maximum value. Instead of the maximum normal force however the maximum lift force coefficient is taken. The criterium triggers vortex position tracking, which influences the magnitude of the vortex induced force and moment.

- Moment coefficient

The model contribution to the dynamic moment coefficient originate from the impulsive and vortex module:

$$\Delta Cm = Cm_i + Cm_v , \quad (2.4)$$

with

$\Delta Cm$  [-] Dynamic addition to static moment coefficient (positive nose up)

$Cm_i$  [-] Dynamic addition due to impulsive lift

$Cm_v$  [-] Dynamic addition due to vortex lift.

As suggested in [3] the impulsive contribution can be estimated by adding the impulsive lift:

$$Cm_i = 0.25 Cl_i , \quad (2.5)$$

with

$Cl_i$  [-] Impulsive lift coefficient.

The addition due to the vortex module depends on the vortex position:

$$Cm_v = -0.2(1 - \cos(\pi\tau_v/T_{vl})) Cl_v , \quad (2.6)$$

with

$\tau_v$  [-] Nondimensional vortex time

$Cl_v$  [-] Vortex lift contribution.

- Drag

The model contribution to the dynamic drag coefficient originates from several modules:

$$\Delta Cd = Cm_i + Cd_v , \quad (2.7)$$

with

$\Delta Cd$  [-] Dynamic addition to static drag coefficient

$Cd_i$  [-] Dynamic addition due to the shed wake induced drag

$Cd_s$  [-] The change in drag due to the separation point position being different from its static position

$Cd_v$  [-] Dynamic addition due to vortex induced normal force.

The shed wake induced drag is calculated by means of

$$Cd_i = \sin(\alpha - \alpha_E) Cl_f , \quad (2.8)$$

with

$\alpha$  [°] Angle of attack

$\alpha_E$  [°] Angle of attack corrected for shed wake effect

$Cl_f$  [-] Lift coefficient corrected for non-linear trailing edge separation.

The separation drag is calculated by means of

$$Cd_s = A_{cd}(Cl - Cl_f) , \quad (2.9)$$

with

$A_{cd}$  [-] Constant taken from Table 1

$Cl$  [-] Static lift coefficient.

The vortex drag is calculated by means of decomposing the vortex induced normal force in drag direction.

A comparison of the performance of the various dynamic stall models can be found in [12, 14]

## 2.3 Structural model

The model in the ECN Aero-Module that deals with the information from the structural part allows for the specification of one structural control point per radial station, located at the quarter chord position. There is no restriction with respect to the spanwise location of the chosen structural control points for communication with the ECN Aero-Module. The parameters given at these stations are then evaluated at the appropriate spanwise location for the chosen aerodynamic solver by using a linear or spline interpolation method, depending on the INTERPOL keyword. This method has the advantage that discretization of elements for structural and aerodynamic purpose can be different since the discretization requirements for both purposes are often not the same.

The deformed position and orientation and accompanied translational and rotational structural velocities of the several radial segments are taken into account for each time step. In the case of a BEM-computation, the distribution of BEM-annuli in the rotorplane is recalculated every time step to properly account for the deformed geometry in the aerodynamic formulation.

## 2.4 BEM

BEM relates the axial and tangential momentum equations for a streamtube (annulus) to the local blade forces. These equations are solved iteratively, yielding the rotor induced velocities in axial and tangential direction. Several engineering extensions exist to account for the shortcomings of this approach. Within the ECN Aero-Module it is possible to switch off or vary many of these options, or use the default settings. The programmed BEM extensions are described below.

### 2.4.1 Element or annulus approach

The BEM formulation features an iterative solution procedure. The convergence criterium for this procedure is based on the annulus averaged axial induction. The user has the freedom to choose between two basic options for evaluation of the BEM equations.

The first option is the element based approach, in which the momentum equations are solved for each element separately and the resulting axial induction for each element is used to calculate the annulus averaged axial induction. This option is also implemented in PHATAS. Since convergence is obtained using the annulus averaged induction averaged over the blades, local momentum equilibrium is not enforced strictly. Therefore, in case of non-uniform inflow conditions it is recommended to prescribe a suitable minimum number of iterations for convergence using the COUNTERMIN keyword.

The second option is the annulus based approach, in which the variables necessary for the evaluation of the momentum equations (e.g. axial force, wind speed, annulus area) are firstly averaged over the annulus. Then the momentum equations are solved for the annulus and the annulus averaged axial induction is obtained. Especially in the case of wind shear, the second option is expected to be less accurate.

### 2.4.2 Oblique inflow

During turbine operation the rotor plane is often angled with respect to the incoming wind field. Conditions different from axial flow can arise due to wind direction variation or a tilted rotor. The oblique inflow results in an azimuthal variation of the wind velocity that is experienced by an element during a rotor revolution. This advancing and retreating blade effect is implicitly included in the velocities coming from the wind and structure modules. To account for the variation of axial induction within each annulus, the model as defined by Schepers [23, 24] and implemented in PHATAS [18] is employed. A skew function  $f_{skew}$  is determined for each element as a function of effective yaw angle, azimuth angle and radial location. This skew function then relates the local induction at each element to the annulus averaged axial induction. The skew function from the yaw model [24] was originally developed from the correlation between annulus averaged and local induction velocities for an annulus by means of wind tunnel measurements. As an alternative to this model, the Glauert model [9] is implemented as well.

The yaw and tilt angle are determined from the orientation of the rotorplane with respect to the inertial reference system orientation (usually at the tower base), taking into account the rotor averaged wind direction. Yaw and tilt angle are then combined into an effective yaw angle.

### 2.4.3 Prandtl correction

To account for the finite number of blades (i.e. the rotor plane is not a solid disk), the Prandtl correction [21] (optionally for both root and tip) is calculated for each element. For the evaluation of this function, root- and/or tipvortex location is necessary.

The Prandtl tip factor at radius  $r$  is calculated using the formula below.

$$F = 2/\pi \arccos(e^{-\pi(R-r)/d}) , \quad (2.10)$$

with

$$d = (2\pi R/B)(U_{w_{tip}} - 0.5U_{i_{tip}})/\sqrt{(\Omega R + V_{i_{tip}})^2 + (U_{w_{tip}} - 0.5U_{i_{tip}})^2} , \quad (2.11)$$

where

$F$	[ $\cdot$ ]	Prandtl factor
$d$	[m]	distance between trailing vortex sheets
$R$	[m]	tip radius measured in rotorplane
$B$	[ $\cdot$ ]	number of blades
$U_{i_{tip}}$	[m/s]	annulus averaged axial induction of tip element
$V_{i_{tip}}$	[m/s]	annulus averaged tangential induction of tip element
$U_{w_{tip}}$	[m/s]	axial component of wind on tip element.

The axial induced velocity used for the calculation of the velocity of the tip vortices (inside the square root of equation 2.11) is taken as the average of the axial induction inside and outside the wake. For the root factor the same formula is used with the relevant variables evaluated at the location of the root vortex. For a radial position inboard from the root vortex, the Prandtl factor is zero. Root and tip factors are multiplied to result in a 'total' Prandtl factor. Although the position of the tip vortex is clear, the root vortex location remains a debatable issue. This position can either be specified by the user (AEROROOT) or calculated from the specified blade geometry. Furthermore, since the values of the induction at the root and tip are not yet known while solving the momentum equations, the calculation of the Prandtl factor is incorporated in the iterative convergence procedure. The Prandtl factor then relates the annulus averaged axial and tangential induction to the local induction at each element.

Combining the Prandtl factor with the skew factor yields the following relationship between local and annulus averaged inductions

$$U_i = U_{i_{ann}}/F - \bar{U}_{i_{ann}} \cdot f_{skew} , \quad (2.12)$$

and

$$V_i = V_{i_{ann}}/F , \quad (2.13)$$

with

$f_{skew}$	[ $\cdot$ ]	skew factor
$U_i$	[m/s]	local axial induction
$V_i$	[m/s]	local tangential induction
$V_{i_{ann}}$	[m/s]	annulus averaged tangential induction of the annulus considered
$U_{i_{ann}}$	[m/s]	annulus averaged axial induction (as used in momentum equations)
$\bar{U}_{i_{ann}}$	[m/s]	annulus averaged axial induction (averaged over blades).

#### 2.4.4 Dynamic inflow model

The ECN dynamic inflow model [27] has been implemented. This model adds another term to the axial momentum equation to account for the aerodynamic rotor 'inertia' in the case of pitch action, rotational speed variation or wind speed variation. See also equation 2.15. The term is proportional to the time derivative of the annulus averaged axial rotor induction and has a dependency on the radial position.

The term is proportional to a factor depending on the radial location and the first time derivative of the annulus averaged axial induction. This time derivative is evaluated by using the values

of the annulus averaged axial induction of the two previous time steps, where time step size is unrestricted:

$$\frac{\partial \bar{U}_{i_{ann}}}{\partial t} = p \cdot \bar{U}_{i_{ann}(t_0)} - c \cdot \bar{U}_{i_{ann}(t_{-1})} - d \cdot \bar{U}_{i_{ann}(t_{-2})}, \quad (2.14)$$

with  $c = -\frac{\Delta t^2}{\Delta t_1^2 - \Delta t_1 \Delta t_2}$ ,  $d = \frac{\Delta t_1}{\Delta t_1 \Delta t_2 - \Delta t_2^2}$ ,  $p = c + d$ ,  $\Delta t_1 = t_0 - t_{-1}$  and  $\Delta t_2 = t_{-1} - t_{-2}$ . For computational efficiency it is chosen to evaluate the time constant needed for the dynamic inflow model on the basis of undeformed rotor geometry.

## 2.4.5 Turbulent wake state model

For heavily loaded rotors, BEM theory predicts flow reversal in the wake, whilst in reality the wake transforms into a turbulent state by sucking in air from outside the streamtube. To account for this effect the momentum equation is replaced by a turbulent wake state (TWS) equation if the annulus averaged axial induction coefficient exceeds a user specified value. The default value for this parameter is 0.38. The quadratic relationship between axial force and induction in the momentum equation is then replaced by a linear relationship tangent to the original quadratic line at the specified induction value.

The resulting axial momentum equation then yields

$$F_{ax} = 2\rho A U_{i_{ann}} (U_\infty - U_{i_{ann}}) + 2\rho R A f(r) \frac{\partial \bar{U}_{i_{ann}}}{\partial t}, \quad (2.15)$$

while for the turbulent wake state (TWS) this is

$$F_{ax} = 2\rho A (a_T^2 U_\infty^2 + U_{i_{ann}} U_\infty (1 - 2a_T)) + 2\rho R A f(r) \frac{\partial \bar{U}_{i_{ann}}}{\partial t}, \quad (2.16)$$

and the tangential momentum equation yields

$$M = 2\rho r A V_{i_{ann}} (U_\infty - U_{i_{ann}}), \quad (2.17)$$

with

$F_{ax}$	[N]	total axial force of the annulus considered
$M$	[Nm]	total torque of the annulus considered
$\rho$	[kg/m <sup>3</sup> ]	air density
$R$	[m]	tip radius of the rotor (measured in rotorplane)
$r$	[m]	radial location of the annulus (measured in rotorplane)
$A$	[m <sup>2</sup> ]	area of the annulus considered
$a_T$	[·]	specified axial induction coefficient for transition to TWS
$U_\infty$	[m/s]	annulus averaged axial wind of the annulus considered
$f(r)$	[·]	time constant factor depending on radial position of the annulus.

## 2.5 AWSM

The Aerodynamic Windturbine Simulation Module (AWSM) has been developed at ECN by van Garrel [29]. The main scope was to keep the advantages of BEM codes in terms of calculation time and ease of use, but to obtain a superior quality, especially concerning wake and time dependent wake-related phenomena. The AWSM code [10, 11, 29] is based on generalized lifting line theory in combination with a free vortex wake method. The main assumption in this theory is that the extension of the geometry in span-wise direction is predominant compared to the ones in chord and thickness direction; because of this, the real geometry is represented with a line passing through the quarter chord point of each cross section and all the flow field in chord-wise direction is concentrated in that point (figure 2). In the next two paragraphs, the bases of the code are presented; the complete description can be found in [29].

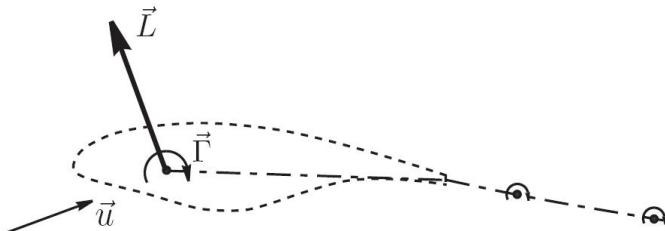


Figure 2: Flowfield model

### 2.5.1 Lifting line model description

Considering the elementary force ( $dF$ ) generated by the geometry, this can be calculated by using the three-dimensional form of the Kutta-Jukowsky equation (equation 2.18).

$$dF = \rho \Gamma V \times dl , \quad (2.18)$$

with

$dF$	[N]	differential aerodynamic force vector
$\rho$	[kg/m <sup>3</sup> ]	air density
$\Gamma$	[m <sup>2</sup> /s]	vortex line strength
$V$	[m/s]	local fluid velocity
$dl$	[m]	directed differential vortex length vector

For each point along the lifting line, the local velocity  $V_i$  can be calculated as superimposition of local free stream velocity and local induced velocity (2.19), where  $v_{ij}$  is the dimensionless induced velocity calculated from Biot-Savart formula (equation 2.20, according to the nomenclature used in figure 3).

$$V_i = V_\infty + \sum \frac{\Gamma_j v_{ij}}{c_j} \quad (2.19)$$

$$V = \frac{\Gamma}{4\pi} \frac{\bar{r}_1 \times \bar{r}_2}{|\bar{r}_1 \times \bar{r}_2|^2} \bar{r}_0 \cdot \left( \frac{\bar{r}_1}{|\bar{r}_1|} - \frac{\bar{r}_2}{|\bar{r}_2|} \right) \quad (2.20)$$

It should be noted that  $dF$  can also be calculated if the section values of lift coefficient  $C_l$  and angle of attack  $\alpha_i$  are known. 2.21 can be written.

$$|\bar{dF}_i| = |\rho \Gamma_i (\bar{V}_\infty + \sum_{j=1}^N \frac{\Gamma_j \bar{v}_{ij}}{c_j}) \times \bar{dl}_i| = \frac{1}{2} \rho V_\infty^2 C_l(\alpha_i) dS_i \quad (2.21)$$

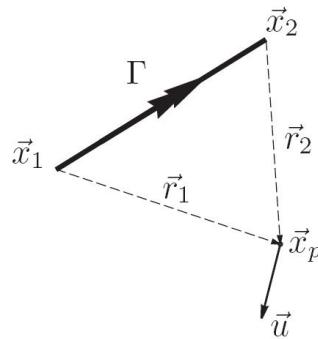


Figure 3: Position vectors describing the geometry for a horseshoe vortex

$$\alpha_i = \tan^{-1} \frac{\bar{V}_i \cdot \bar{u}_{ni}}{\bar{V}_i \cdot \bar{u}_{ai}} \quad (2.22)$$

with

$dF$	[N]	differential aerodynamic force vector
$\alpha_i$	[deg]	local angle of attack for wing section i
$c$	[m]	chord
$c_j$	[m]	chord for wing section j
$dF$	[N]	differential aerodynamic force vector
$dl$	[m]	directed differential vortex length vector
$dS_i$	[m <sup>2</sup> ]	differential planform area at control point i
$\Gamma$	[m <sup>2</sup> s <sup>-1</sup> ]	vortex line strength
$r_0$	[m]	vector from beginning to end of vortex segment
$r_1$	[m]	vector from beginning of vortex segment to arbitrary point in space
$r_2$	[m]	vector from end of vortex segment to arbitrary point in space
$\rho$	[kg m <sup>-3</sup> ]	fluid density
$u_{ai}$	[·]	chordwise unit vector at control point i
$u_{ni}$	[·]	normal unit vector at control point i
$V$	[m s <sup>-1</sup> ]	local fluid velocity
$V_\infty$	[m s <sup>-1</sup> ]	velocity of the uniform flow or free stream
$v_{ij}$	[·]	dimensionless velocity induced at control point j by vortex i , having a unit strength

In AWSM, the effects of viscosity are taken into account through the user-supplied nonlinear relationship between local flow direction and local lift, drag and pitching moment coefficients; this means that equation 2.21 can be numerically solved minimizing the difference between the elementary force obtained from sectional properties and the one calculated from the Kutta-Jukowsky formula.

## 2.5.2 Vortex wake

As in the continuous flowfield representation, the vorticity is shed from the trailing edge of the configuration surface and convected downstream in the AWM flow model as time advances. The blade geometry consists of one or more strips that carry a vortex ring whose bound vortices are located at the quarter chord position and at the trailing edge. The vortex strengths  $\Gamma$  of these vortex rings are to be determined. Each timestep  $\Delta t$  new vortex rings with these strengths are shed from the trailing edge and joined with the older vortex rings. These vortex rings together will form a vortex lattice. A sketch of the wake geometry for three strips after four timesteps is shown in figure

4. The position of the first shed free spanwise vortex behind the trailing edge (TE) lies at some fraction between the current TE position and the wind-convected TE position from the previous timestep. In accordance with vortex-lattice practice this fraction is chosen to be 25 percent of the chord. Upstream of this position the vortex rings have a strength equal to the corresponding vortex ring at the configuration. The position of the downstream part of the wake is determined each timestep by convection of the wake vortex-lattice nodes. This convection is applied in two separate steps; convection by the onset wind velocity and convection by the "induced velocity" of all bound- and trailing. Notice that the wake shed vortices are formed by the adjoining sides of two vortex rings from successive timesteps. This means that they cancel each other in case the vortex ring strengths are identical. For steady flows therefore only the trailing vortices are active.

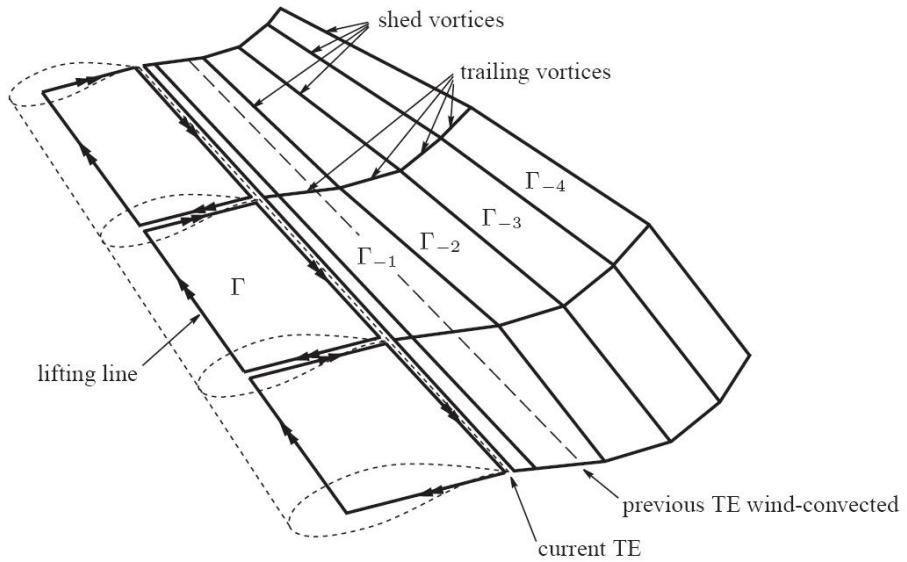


Figure 4: Wake geometry

## 3 Usage of software

The software is suitable for usage on both Linux and Windows and supports execution on both 32 and 64 bit machines. The source code of the ECN Aero-Module has been written in FORTRAN and compiled using the ifort compiler.

### 3.1 Coordinate systems

The several HAWT coordinate systems used in the ECN Aero-Module are based on Germanischer Lloyd conventions [1]. They are highlighted in Figures 5 and 6.

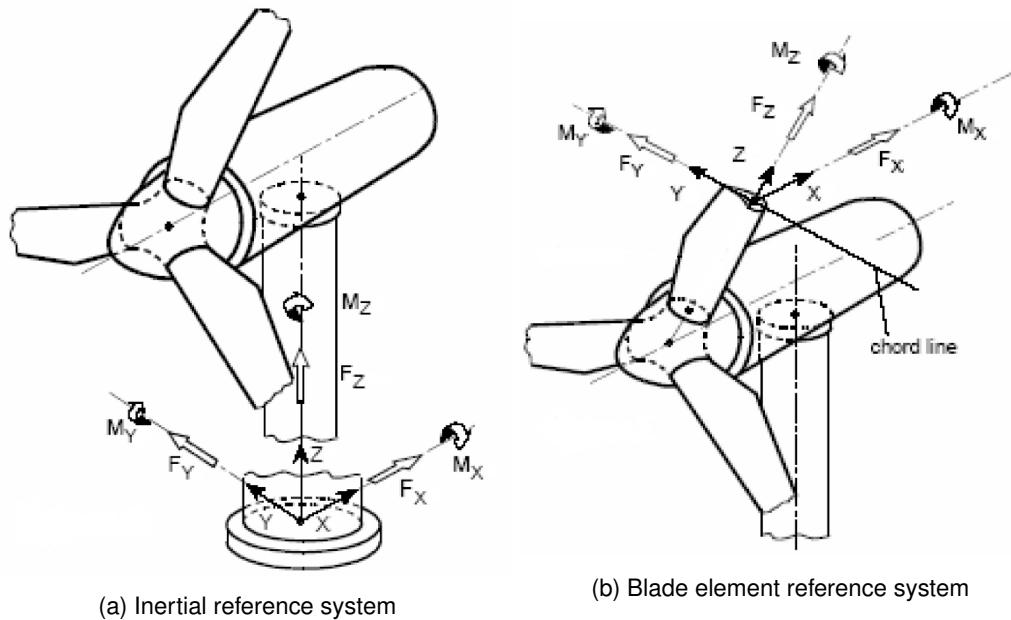


Figure 5: Coordinate systems according to Germanischer Lloyd convention [1]

For stand alone execution, a rigid turbine is defined based on these coordinate systems. The exact geometry definition is highlighted in Figure 7 for a HAWT and Figure 8 for a VAWT. Both of these coordinate systems are made up of different XYZ coordinate frames, each characterized by the rotations  $\alpha$ ,  $\beta$  and  $\gamma$  respectively about the X, Y and Z axis. Not all of these rotations are however actively used by ECN Aero-Module in the definition of the wind turbine position. Active rotations in the ECN Aero-Module coordinate systems for HAWTs and VAWTs are highlighted in red respectively in Figures 7 and 8. Tables 2 and 3 show the translation and rotation order for the HAWT and VAWT coordinate systems, respectively.

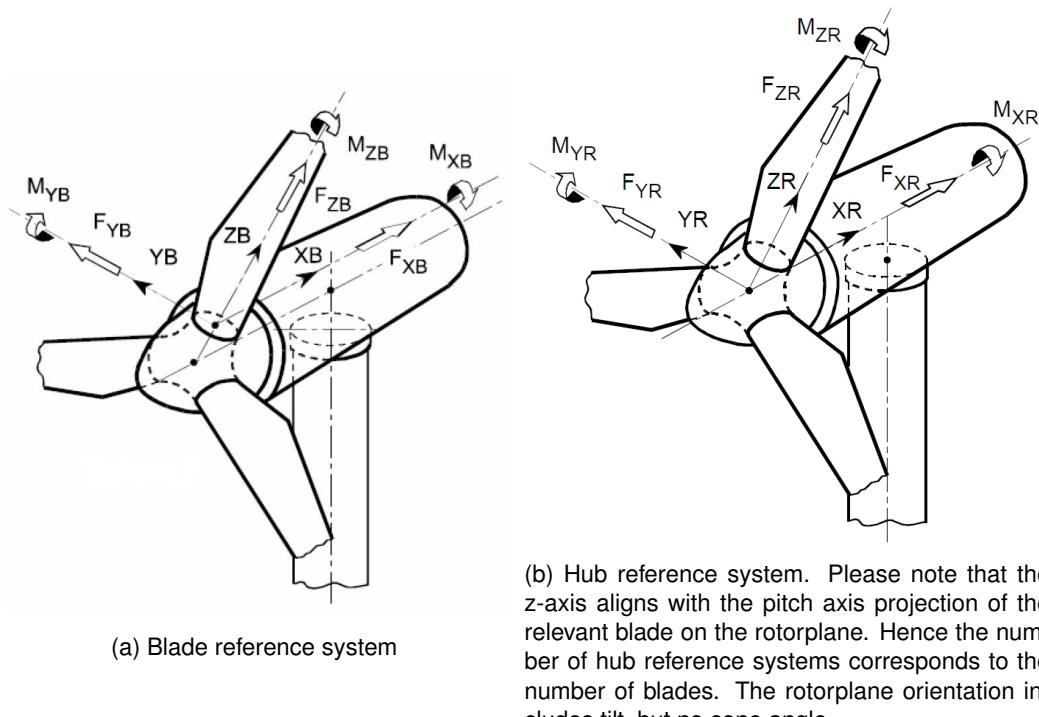


Figure 6: Coordinate systems according to Germanischer Lloyd convention [1] ctd

order	symbol	type	keyword
1	$H_T$	translation	(HUBHEIGHT, XNAC2HUB, ZNAC2HUB)
2	yaw ( $\gamma_T$ )	rotation	YAWANGLE
3	tilt ( $\beta_T$ )	rotation	TILTANGLE
4	$\Delta_{HH}$	translation	ZNAC2HUB
5	$O$	translation	-XNAC2HUB
6	azimuth ( $\alpha_H$ )	rotation	
7	$R_H$	translation	BLADEROOT
8	cone ( $\beta_B$ )	rotation	CONEANGLE
9	pitch ( $\gamma_B$ )	rotation	-PITCHANGLE
10	$R_B (X_B, Y_B, Z_B)$	translation	AEROPROPS
11	alpha ( $\alpha_E$ )	rotation	AEROPROPS
12	beta ( $\beta_E$ )	rotation	AEROPROPS
13	twist ( $\gamma_E$ )	rotation	AEROPROPS (-)
14	quarter chord	translation	AEROPROPS

Table 2: Translation and rotation order for the HAWT coordinate system.

The definition of radial distances used throughout the ECN Aero-Module is illustrated in Figure 9. These distances denote the radial distance to the structural control points (section 2.3). A 'deformed' and 'curvilinear' version are defined, illustrated in Figure 9. The 'deformed' radial distance is defined as the z-coordinate of the structural control point position expressed in the hub coordinate system of the relevant blade (Figure 6b). Hence this distance is measured along the projection of the pitch axis on the rotorplane, originating in the rotor center. If the blade deforms this projected distance will change as well, hence the name 'deformed' radial distance.

To be able to correlate this 'deformed' radial distance to the sectional characteristics that may vary along the span of the blade, an 'curvilinear' radial distance is defined. This variable represents the curvilinear distance along the blade from the pitch bearing to the relevant structural control point. The 'curvilinear' radial distance can be calculated by summing the incremental distance (in x-, y-

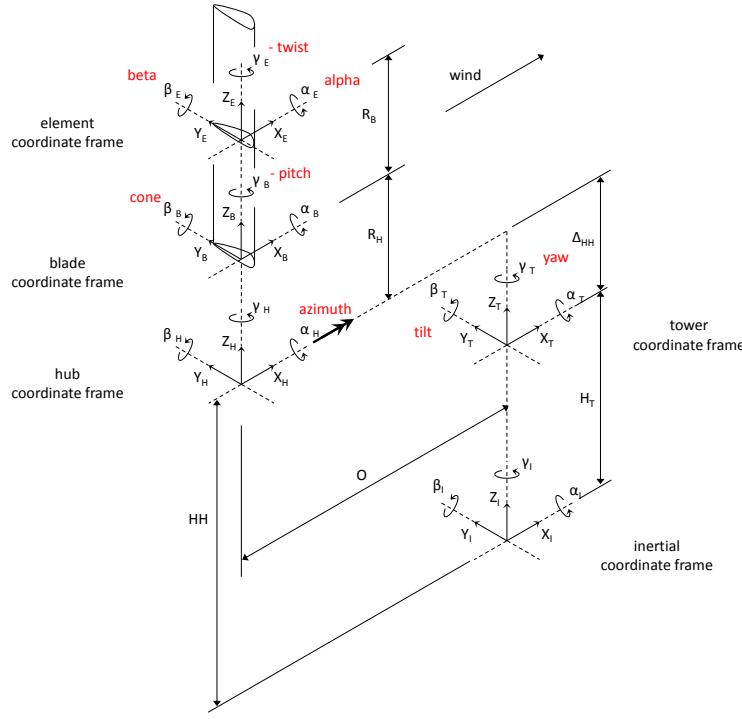


Figure 7: Coordinate system for HAWT currently implemented in ECN Aero-Module. Different coordinate frames are shown.  $O$  is the rotor overhang,  $H_T$  is the tower height, and  $\Delta_{HH}$  is the distance between the tower top and the rotor hub center along the  $Z_T$  axis.  $R_H$  is the hub radius, while  $R_B$  is the distance between the blade element and the blade root as specified by  $X_B$ ,  $Y_B$  and  $Z_B$ . Names of active angles are highlighted in red.  $HH$  represents the hub height.

order	symbol	type	keyword
1	$H_T$	translation	(HUBHEIGHT, ZNAC2HUB)
2	roll ( $\alpha_T$ )	rotation	YAWANGLE
3	tilt ( $\beta_T$ )	rotation	TILTANGLE
4	$\Delta_{HH}$	translation	ZNAC2HUB
5	azimuth ( $\gamma_T$ )	rotation	
6	$R$	translation	-XNAC2HUB
7	$R_H$	translation	BLADEROOT
8	cone ( $\beta_B$ )	rotation	CONEANGLE
9	pitch ( $\gamma_B$ )	rotation	-PITCHANGLE
10	$R_B (X_B, Y_B, Z_B)$	translation	AEROPROPS
11	alpha ( $\alpha_E$ )	rotation	AEROPROPS
12	beta ( $\beta_E$ )	rotation	AEROPROPS
13	twist ( $\gamma_E$ )	rotation	AEROPROPS (-)
14	quarter chord	translation	AEROPROPS

Table 3: Translation and rotation order for the VAWT coordinate system.

and z-direction) between the structural control points positioned inboard from the relevant structural control point. The 'curvilinear' radial distance is assumed to be constant for each specific control point throughout a simulation. Hence although the control points are located in the quarter chord positions (which not necessary coincide with the neutral line for flatwise and edgewise bending), the axial deformation due to blade bending is assumed to be small.

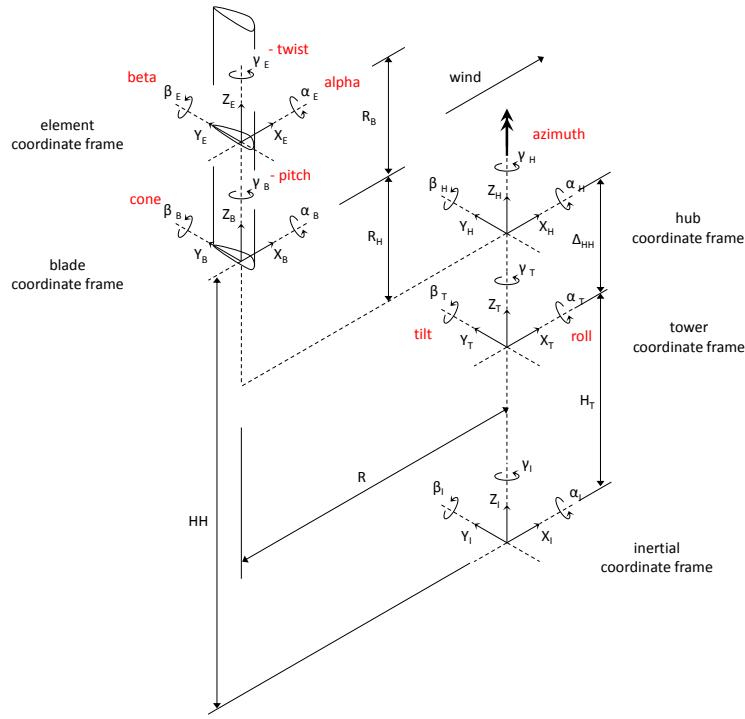


Figure 8: Coordinate system for VAWT currently implemented in ECN Aero-Module. Different coordinate frames are shown.  $R$  is the rotor radius,  $H_T$  is the tower height, and  $\Delta_{HH}$  is the distance between the tower top and the rotor hub center along the  $Z_T$  axis.  $R_H$  is the distance between the rotor hub center and the blade root along the  $Z_H$  axis, while  $R_B$  is the distance between the blade element and the blade root as specified by  $X_B$ ,  $Y_B$  and  $Z_B$ . Names of active angles are highlighted in red.  $HH$  represents the hub height.

## 3.2 Input

In order to run the ECN Aero-Module, several input files must be placed in the specified working directory. For stand-alone execution this working directory is the location of the executable. An initialisation file is used that should contain the name of the main input file. For stand-alone execution this file should be named `ecnaero.ini` whilst for a coupled simulation the filename can be chosen depending on the implementation and interfacing with the chosen structural solver. An example of an initialisation file is given below.

---

Initialisation example file

---

```

! Should contain the name of the main input file.
! Comment lines should be preceded by !
input.txt

```

The main input file contains several keywords (section 3.2.1) that define the relevant aerodynamic parameters. The filenames containing the wind description (section 3.2.2) and the airfoil coefficients (section 3.2.3) are also determined in this file. All input files should be located in the working directory. Please note that the length of the absolute path of the working directory should not exceed 1024 characters.

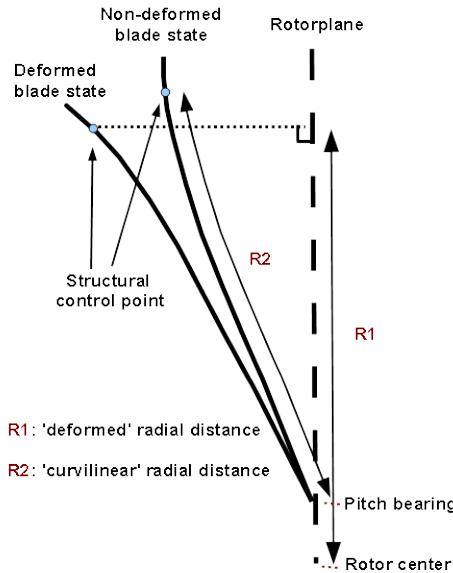


Figure 9: Definition of 'deformed' and 'curvilinear' radial distance (HAWT)

### 3.2.1 Keywords (main input file)

The available keywords in the main input file are summarised below. For each keyword, the variable type is indicated together with the SI units between [] and the default value between <> if applicable. The INTEGER variable type is 4-bit and the REAL variable type is in double precision. The keywords can be placed in a random order. The keyword values need to be placed aft of the keyword either on the same line or on the next line. Care should be taken not to include a keyword name in a CHARACTER\*256 variable keyword. Comment lines can be included by starting with an exclamation mark. An example is included further below.

The keywords can be divided in a general part and specific parts for blade definition, stand alone execution, airfoil data, environment data, BEM and AWSM.

#### General

AEROMODEL [-] <1> INTEGER

Aerodynamic model to be used.

- 1 BEM
- 2 AWSM

DEBUGFILE [-] <0> INTEGER

Toggle to enable writing of specialist files containing a time history of detailed (aerodynamic) variables.

- 0 OFF
- 1 ON

TURBINETYPE [-] <1> INTEGER

Type of turbine to be modeled.

- 1 Horizontal axis turbine HAWT
- 2 Vertical axis turbine VAWT

INCLUDE [-] <-> CHARACTER\*256

Filename of an additional file which can be used to define extra keywords. The data given in this file will be read in the same way as the main input file. The file must be positioned in the working directory. It is recommended to use this additional file for detailed input settings (advanced users only).

INTERPOL [-] <2> INTEGER

Toggle to specify interpolation method for obtaining aerodynamic coefficients and interpolation between structural and aerodynamic control points.

- 1 Linear
- 2 Spline [28]

LOGFILENAME [-] <-> CHARACTER\*256

Filename of the logfile, which contains a summary of the specified turbine and model choices.

SUBITERFLAG [-] <1> INTEGER

Keyword to specify how to deal with multiple calls (subiterations) to the ECN Aero-Module during the same timestep in case of implicit time stepping.

- 0 Only perform aerodynamic calculation for increasing timestep. For implicit time stepping, aerodynamic calculations are only performed for the first subiteration. In between aerodynamic calculations, the output is kept constant.
- 1 Perform aerodynamic calculation for each (sub)iteration.
- 2 Perform aerodynamic calculation for subiteration numbers greater than or equal to 9. For the subiterations before that, the output values are linearly extrapolated based on the values of the 2 previous time steps. In between interpolations and aerodynamic calculations, the output is kept constant.
- 3 Perform aerodynamic calculation for the first subiteration and subiteration numbers greater than or equal to 9. In between aerodynamic calculations, the output is kept constant.

## Blade definition

AEROPROPS table [-] <-> REAL

Table defining the blade geometry using the format below. The number of lines in the table should exceed the value of one. In addition to that it should be noted that for the BEM model, this table only specifies the blade geometry and does not determine the aerodynamic element discretization. For AWSM however the table also defines the corner points and hence the aerodynamic element discretization. Hence for AWSM the last point specified in the table should be located close to (or at) the tip, otherwise the total blade length is not modelled. The current version of the ECN Aero-Module allows for the definition of one blade geometry per rotor.

zB [m]	chord [m]	t/c [-]	twist [deg]	quarter chord [%c]	xB [m]	yB [m]

- $z_B$

$z_B$  is defined as the z-coordinate of the blade coordinate system (Figure 6a), originating in the pitch bearing. It can also be denoted as the radial distance along the blade pitch axis from the blade root.

- chord

The local chord corresponding to the  $z_B$  location.

- t/c

The local thickness chord ratio corresponding to the  $z_B$  location.

- twist

The local twist angle corresponding to the  $z_B$  location.

- quarter chord

The chordwise location of the quarter point with respect to the point defined by  $z_B$ ,  $x_B$  and  $y_B$ . The direction is positive pointing from the leading edge towards the trailing edge of the cross section.

- $x_B$

The x-coordinate corresponding to the  $z_B$  location, expressed in the blade coordinate system. This variable is used in the case of flatwise prebend or definition of a winglet.

- $y_B$

The y-coordinate corresponding to the  $z_B$  location, expressed in the blade coordinate system. This variable is used in the case of edgewise prebend or a swept blade.

The location and orientation of the quarter chord points is defined in the following order. The variables  $z_B$ ,  $x_B$  and  $y_B$  define a point in space and also a change in orientation with respect to the origin of the blade coordinate system. After the specification of  $z_B$ ,  $x_B$  and  $y_B$ , the twist angle is applied. Then the specified quarter chord distance is applied to translate the element along the chord and align the point with the chosen quarter chord point location. A schematic figure illustrating the blade definition using these variables is given in Figure 10. It is the responsibility of the user to supply the aerodynamic coefficients corresponding to the orientation of the local chord direction as defined above.

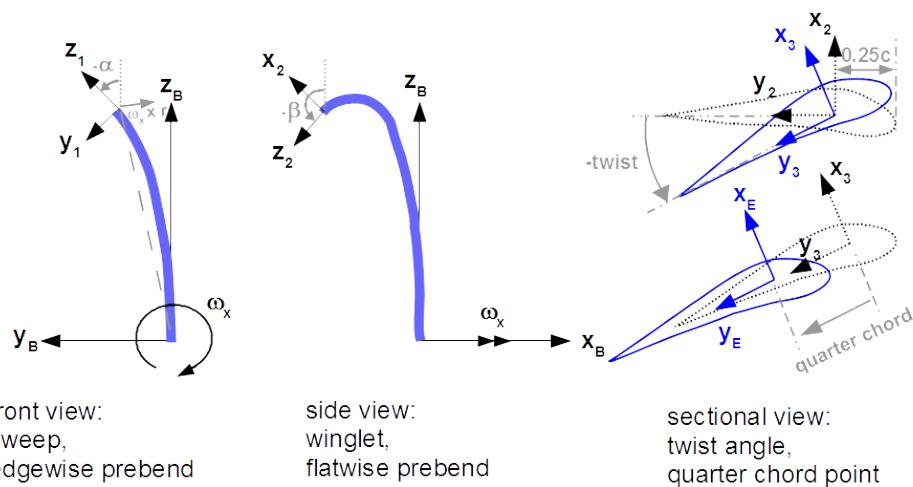


Figure 10: Definition of blade geometry using the AEROPROPS table input

**AEROROOT [m] <-> REAL**

Position of the root vortex. Only necessary if the Prandtl root correction is used for BEM or a prescribed wake is chosen for AWSM. The location is defined along the curvilinear radial distance (Figure 9). This keyword is optional and if it is not used in the input file, the position is calculated from the specified rotor blade geometry.

**BLADELENGTH [m] <-> REAL**

The blade length measured along the pitch axis. This is identical to the z-coordinate of the blade tip in the blade coordinate system (Figure 6a), originating in the blade root or pitch bearing.

**BLADEROOT [m] <0.0> REAL**

In case of a HAWT this is the distance between the rotor center and the blade root location, measured perpendicular to the rotor axis. Often denoted as hub radius. In case of a VAWT this parameter is used to denote the rotor radius (Figure 8).

**Stand alone****CONEANGLE [deg] <0.0> REAL**

Cone angle of the turbine, defined according to GL conventions.

**HUBHEIGHT [m] <40.0> REAL**

Hub height of the wind turbine. For HAWT and VAWT the hub height is defined as the center of the circle described by the lowest point of each blade (blades' root) during each rotor revolution.

**NROFBLADES [-] <3> INTEGER**

Number of blades on the turbine. A minimum value of 1 and a maximum value of 4 is required. If the software is coupled to a structural solver, this information has to supplied in the initialise call.

**PITCHANGLE [deg] <0.0> REAL**

Pitch angle of the blades, defined according to GL conventions.

**RPM [rpm] <15.0> REAL**

Rotational speed of the rotor, defined according to GL conventions. Hence positive direction is clockwise rotation from a viewpoint in front of the turbine.

**STARTAZIMUTH [deg] <0.0> REAL**

Starting azimuth of blade 1 ( $0^\circ$  is blade pointing upwards).

**TBEGIN [s] <0.0> REAL**

Start time of the simulation.

**TEND [s] <3.0> REAL**

End time of the simulation.

**TILTANGLE [deg] <0.0> REAL**

Tilt angle of the turbine, defined according to GL conventions.

**TIMESTEP [s] <0.01> REAL**

Time step size.

XNAC2HUB [m] <0.0> REAL

X-location of hub in nacelle coordinate system, defined according to GL conventions. Please note that this variable is negative for an upwind placed rotor.

YAWANGLE [deg] <0.0> REAL

Yaw angle of the turbine, defined according to GL conventions. Please note that this variable does not include the wind direction and merely consists of geometric yaw.

ZNAC2HUB [m] <0.0> REAL

Z-location of hub in nacelle coordinate system, defined according to GL conventions.

## Airfoil data

AOAEVAL [-] <1> INTEGER

Toggle to specify chordwise location used for angle of attack determination.

- 0 Quarter chord location
- 1 Three-quarter chord location

COEFFFILENAME [-] <-> CHARACTER\*256

Filename that contains the filenames of the data files with the profile coefficients that are included in the blades, as explained in section 3.2.3. The file must be positioned in the working directory.

CORR3DTYPE [-] <1> INTEGER

Toggle to specify correction type for rotational effects.

- 0 No correction
- 1 Snel method (section 2.2.3)

DYNSTALLTYPE [-] <1> INTEGER

Toggle to determine dynamic stall model.

- 0 No dynamic stall model
- 1 Snel first order dynamic stall model
- 2 Snel second order dynamic stall model

LINREG [-] <1> INTEGER

Toggle to specify how the potential flow lift curve slope is determined which is used for calculation of 3D rotational effects and dynamic stall modeling.

- 0 Assume a value of  $2\pi$
- 1 Determine the slope using linear regression on the user specified lift curve between MINAOALIN and MAXAOALIN. It is noted that at least two points must be present within the specified range.

MINAOALIN [deg] <-3.0> REAL

Start of angle of attack range for determination of the potential flow lift curve slope in case LINREG=1.

MAXAOALIN [deg] <5.0> REAL

End of angle of attack range for determination of the potential flow lift curve slope in case LINREG=1.

## *Beddoes-Leishman model related keywords*

FSMETHOD [-] <2> INTEGER

Toggle to determine flow separation modeling method (Beddoes-Leishman only), see also [14].

- 0 Kirchoff
- 1 Øye
- 2 FFA
- 3 Larsen

For the significance of the below keywords, referral is made to the relevant paper describing the model [14].

BL_A1	[·]	<0.30>	REAL
BL_A2	[·]	<0.70>	REAL
BL_B1	[·]	<0.14>	REAL
BL_B2	[·]	<0.53>	REAL
BL_KA	[·]	<0.75>	REAL
BL_TP	[·]	<1.50>	REAL
BL_TF	[·]	<5.00>	REAL
BL_TV	[·]	<6.00>	REAL
BL_TVL	[·]	<5.00>	REAL
BL_ACD	[·]	<0.13>	REAL

#### *Onera model related keywords*

ON_LAML	[·]	<0.17>	REAL
ON_SIGL	[·]	<6.28>	REAL
ON_R0	[·]	<0.20>	REAL
ON_R2	[·]	<0.20>	REAL
ON_A0	[·]	<0.30>	REAL
ON_A2	[·]	<0.20>	REAL
ON_E2	[·]	<0.53>	REAL
ON_SL	[·]	<3.14>	REAL

## Environment

AIRDENSITY [kg/m<sup>3</sup>] <1.225> REAL

Density of the air.

DYNVISC [kg/(ms)] <1.79 × 10<sup>-5</sup>> REAL

Dynamic viscosity of the air.

HORSHEAR [%/m] <0.0> REAL

Only used for simple wind. The applied horizontal wind shear varies linearly with the lateral (y-) coordinate and is expressed in % of the wind speed (at y=0 for the given height) per meter. The horizontal shear is related to the fixed inertial coordinate system at the tower base. A positive value yields an increased wind for a negative y-coordinate.

RAMPTIME [s] <0.0> REAL

This keyword defines the length of the optional wind ramp during the first timesteps. The wind ramp is applied from the first timestep until the specified RAMPTIME according to equation 3.1.

RAMPFACTOR [-] <1.0> REAL

This keyword defines the scaling of the specified wind for the optional wind ramp during the first timesteps. The specified wind is scaled with RAMPFACTOR and interpolated linearly according to

$$U_{w,m} = U_w(\text{RAMPFACTOR} + \frac{t - t_0}{\text{RAMPTIME} - t_0}(1 - \text{RAMPFACTOR})) \quad , \quad (3.1)$$

with

$U_{w,m}$	[s]	modified wind velocity applied at current time step
$U_w$	[s]	specified wind velocity at current time step
$t$	[s]	time at current time step
$t_0$	[s]	time at first time step.

Hence for a factor of 0.4 the wind starts with 40% of the specified wind speed and linearly builds up to the specified wind speed as a function of time. A clarifying illustration is added in Figure 11.

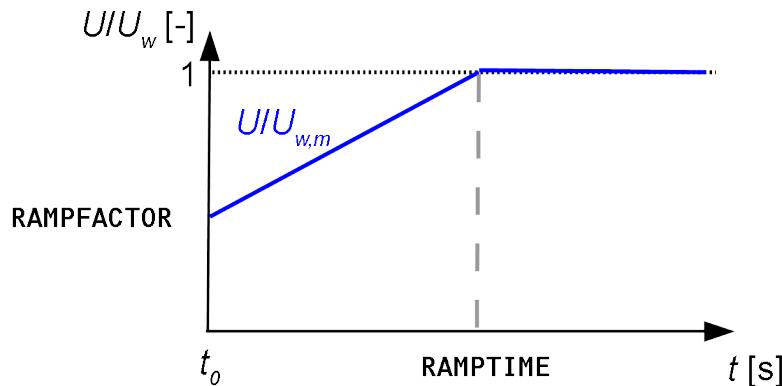


Figure 11: Definition of RAMPFACTOR and RAMPTIME

SOS [m/s] <340.3> REAL

Speed of sound in the air, as used in the Beddoes-Leishmann dynamic stall model to account for compressibility.

SHEAREXP [-] <0.0> REAL

Only used for simple wind. Vertical shear exponent in accordance with wind profile power law. A value of zero yields no vertical wind shear.

TOWERBASERADIUS [m] <0.0> REAL

Radius of the tower at the tower base to calculate the tower effect.

TOWERTOPRADIUS [m] <0.0> REAL

Radius of the tower at the tower top to calculate the tower effect.

WINDINTERPL [-] <2> INTEGER

Toggle to specify the wind interpolation type for Turbsim and Mann wind formats.

1 Linear

2 Cubic

WINDFILENAME [-] <-> CHARACTER\*256

Name of the file that contains the wind field description. The file must be positioned in the working directory.

WINDTYPE [-] <1> INTEGER

Toggle to specify the wind type.

1 Simple wind

2 SWIFT wind

3 TurbSim wind (Bladed style)

4 Mann wind (see also section 3.2.2 for instructions)

Z0 [-] <0.0> REAL

Only used for simple wind. Roughness length of the terrain. A value of zero yields no vertical wind

shear.

## BEM

ATRANSITION [-] <0.38> REAL

Value of the axial induction above which the turbulent wake state (TWS) model is used.

COUNTERMIN [-] <50> INTEGER

Minimum number of iterations to be used for convergence of momentum equations.

DYNINFLOW [-] <1> INTEGER

Toggle to specify whether the dynamic inflow model should be used.

0 No dynamic inflow model

1 Dynamic inflow model (section 2.4.4)

INDUCTIONDRAG [-] <0> INTEGER

Toggle to specify whether the drag force should be included in the induction calculation.

0 Drag not included

1 Drag included

NROFBEMELEMENTS [-] <14> INTEGER

Number of elements (annuli) that are used for the BEM calculation.

PRANDTLROOT [-] <1> INTEGER

Toggle to specify whether the Prandtl correction should be applied to the root.

0 No Prandtl root correction

1 Prandtl root correction (section 2.4.3)

PRANDTLLTIP [-] <1> INTEGER

Toggle to specify whether the Prandtl correction should be applied to the tip.

0 No Prandtl tip correction

1 Prandtl tip correction (section 2.4.3)

YAWMODEL [-] <1> INTEGER

Yaw model to be used to model skewed wake effect.

0 no yaw model

1 ECN yaw model [23, 24]

2 Glauert yaw model [9]

## AWSM

AERORESULTSFILE [-] <aeroresultsout.dat> CHARACTER\*256

Name of the file that contains AWSM results in terms of aerodynamic characteristics.

TIMESTARTAERO [-] <-10> INTEGER

Start timestep to save the aerodynamic data.

TIMEENDAERO [-] <-10> INTEGER

End timestep to save the aerodynamic data.

AEROOUTPUTINCREMENT [-] <-10> INTEGER  
 Timestep increment to save the aerodynamic data.

GEOMRESULTSFILE [-] <geomresultsout.dat> CHARACTER\*256  
 Name of the file that contains the data related to the geometry and the wake as calculated by AWSM.

TIMESTARTGEOM [-] <-10> INTEGER  
 Start timestep to save the geometry data.

TIMEENDGEOM [-] <-10> INTEGER  
 End timestep to save the geometry data.

GEOMOUTPUTINCREMENT[-] <-10> INTEGER  
 Timestep increment to save the geometry data.

TRACELEVEL | [-] <2> INTEGER  
 Level of detail in trace information of AWM simulation. A lower value corresponds to more details being outputted.

- 5 [MSG] Informational message; always written to the enabled output streams
- 4 [FTL] Something bad has happened, the system is of little use to anyone
- 3 [ERR] A specific task may have failed, but the system will keep running
- 2 [WRN] Something went wrong but the system will try to recover from it
- 1 [INF] Informational messages about what's happening
- 0 [DBG] Hints used during the initial coding and debugging process

STDOUTSWITCH [-] <0> INTEGER  
 Toggle to specify if the standard output unit (the screen) is used to output trace information.  
 0 No  
 1 yes

LOGTRACESWITCH [-] <0> INTEGER  
 Toggle to specify if the log file logoutawsm.dat is used to output trace information.  
 0 No  
 1 yes

CENTEREDCONTRPOINTS [-] <0> INTEGER  
 Toggle to specify if the control points are placed in the centre of each section.  
 1 Yes  
 0 No. The points are distributed according to a cosine distribution

STREAMWISEWAKEPOINTS [-] <-> INTEGER  
 Number of wake points in streamwise direction.

FREESTRMIWSEWAKEPOINTS [-] <-> INTEGER  
 Number of wake points in streamwise direction free to roll up through wake self-influence.

PRSCRBWAKE [-] <0> INTEGER  
 Toggle to specify whether the wake convection is prescribed for the fixed wake points. An illustration of the possible options in combination with the chosen wake points is given in Figure 12.  
 0 No wake prescription. For FREESTRMIWSEWAKEPOINTS=0 the wake convects with the undisturbed wind velocity. Otherwise the wake convects with the undisturbed wind velocity minus the spanwise ('blade') averaged induction at the free to fixed wake transition.  
 1 Wake convection prescribed as a function of local blade induction and axial distance from rotorplane  
 2 Constant wake convection as fraction of undisturbed wind velocity prescribed by CONVECFATOR

CONVECFATOR [-] <0.0> REAL

Factor to modify or specify wake prescription convection speed.

PRSCRBWAKE=1 Induced wake vortex convection speed  $u_i$  is modified according to

$$u_i = u_i \cdot (1 + \text{CONVECFATOR})$$

PRSCRBWAKE=2 Induced wake vortex convection speed  $u_i$  is set to fraction of undisturbed wind

$$u_i = U_\infty \cdot \text{CONVECFATOR}$$

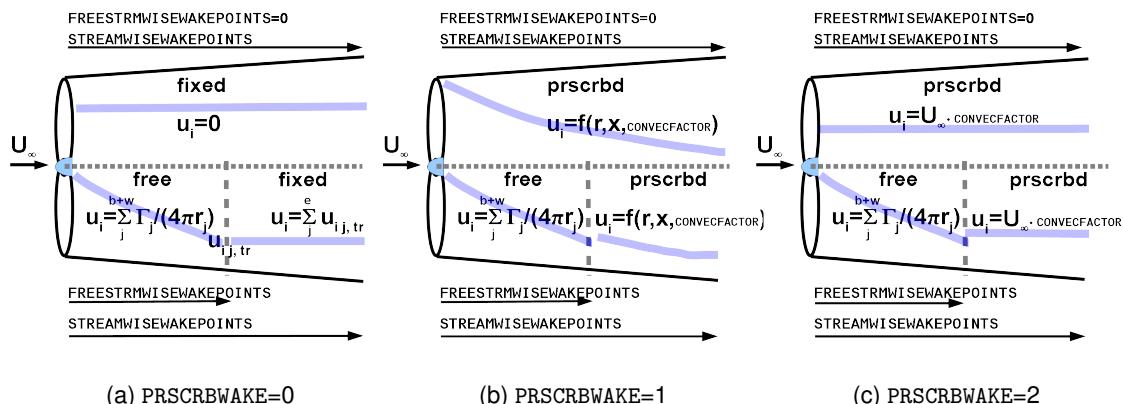


Figure 12: Illustration of possible wake configurations. The indicated formulas describe how the induced wake vortex convection speed  $u_i$  is modeled. The top rotor half mimics a simulation without free wake points, the bottom rotor half features a partial free wake.

VORTEXCUTOFF [-] <2> INTEGER

Toggle to set the implementation of vortex cut-off radius to limit induced velocities close to the core.

1 linear

2 smooth

WAKECUTOFFRADIUS [-] <0.1> REAL

Value of the cut-off radius used for calculating induced velocities in the wake. Specified as a fraction of the element vortex line length.

CIRCCONVCRIT [-] <0.0001> REAL

Circulation convergence criterion.

LIFTCUTOFFRADIUS [-] <0.001> REAL

Value of the cut-off radius used for calculating induced velocities at the lifting line control points. Specified as a fraction of the element vortex line length.

NEWRAPNEIGHBOURS [-] <2> INTEGER

Number of neighbors to be taken into account in Newton-Raphson matrix in order to guarantee numerical stability.

STARTSMOOTHINGS [-] <3> INTEGER

NUMITAEROCONV [-] <100> INTEGER

Maximum number of iterations in order to reach aerodynamic convergence.

NUMITEQMAX [-] <10> INTEGER

Maximum number of iterations in order to reach structural equilibrium. CURRENTLY NOT IN USE.

EXTFIELDFLAG [-] <0> INTEGER

Toggle to specify whether the induced velocities are also to be calculated in the specified external field points. The complete description is given in appendix A.

- 0 No
- 1 yes

**GROUNDFLAG [-] <2> INTEGER**

Toggle to specify if the ground effect is taken into account.

- 0 No
- 1 Yes. The ground effect is simulated by mirroring the real geometry. An accurate but computational expensive option.
- 2 No. The turbine is not mirrored, but for the calculation of the local wind components, positions of the wake below the ground level are set to zero. The wake positions themselves are not restricted for the calculation of rotor induction.
- 3 Yes. Like option 2 but now the wake positions below the ground level are really set to zero and hence the wake itself is constrained to be above the ground level. The expansion of the wake is not completely free.

**GROUNDLEVEL [-] <0.0> REAL**

Level of the ground in case of calculation in ground effect (GROUNDFLAG>0). Defined in terms of the z-coordinate of the inertial reference system (Figure 5a).

**PARALLEL [-] <0> INTEGER**

Toggle to specify whether the calculation of AWSM free vortex wake induced velocities is to be distributed over the available cores using OpenMP ® (parallelization).

- 0 No
- 1 yes

**WAKEREDUCTIONSTART [-] <0> INTEGER**

Streamwise wakepoint number after which wake reduction should be applied. For a WAKEREDUCTIONSTART parameter less than 1 or greater than STREAMWISEWAKEPOINTS, the wake reduction technique will not be applied.

**WAKEREDUCTIONSKIP [-] <0> INTEGER**

Number of streamwise wakepoints to skip after the specified WAKEREDUCTIONSTART in case of wakereduction. For a WAKEREDUCTIONSKIP parameter greater than FREESTRMWISEWAKEPOINTS, the wake reduction technique will not be applied.

An example of a main input file is included below.

```
----- Main input example file -----
```

---

```

!----- General -----
!
AEROMODEL           1      ! 1:BEM 2: AWSM
LOGFILENAME         logfile.dat
!INCLUDE            specialist_input.txt
!

!----- Blade definition -----
!

AEROPROPS
!zB [m] chord [m] t/c [-] twist [deg] C14 [%c] xB [m] yB [m]
0.020   0.195    1.000  0.000    -4.9    0.0    0.0
0.025   0.090    1.000  0.000    -2.3    0.0    0.0
0.090   0.090    1.000  0.000    -2.3    0.0    0.0
0.165   0.165    0.625  8.200    -1.1    0.0    0.0
0.240   0.240    0.250  16.40     0.0    0.0    0.0
0.465   0.207    0.250  12.10     0.0    0.0    0.0
0.690   0.178    0.250  8.300     0.0    0.0    0.0
0.815   0.166    0.250  7.100     0.0    0.0    0.0
0.915   0.158    0.230  6.100     0.0    0.0    0.0
1.015   0.150    0.210  5.500     0.0    0.0    0.0
1.140   0.142    0.210  4.800     0.0    0.0    0.0
1.265   0.134    0.210  4.000     0.0    0.0    0.0
1.365   0.129    0.196  3.700     0.0    0.0    0.0
1.465   0.123    0.180  3.200     0.0    0.0    0.0
1.590   0.116    0.180  2.600     0.0    0.0    0.0
1.815   0.102    0.180  1.500     0.0    0.0    0.0
1.955   0.092    0.180  0.700     0.0    0.0    0.0
1.983   0.082    0.180  0.469     0.0    0.0    0.0
2.012   0.056    0.180  0.231     0.0    0.0    0.0
2.040   0.011    0.180  0.000     0.0    0.0    0.0
BLADELENGTH          2.04
BLADEROOT            0.21
!

!----- Stand alone -----
!

CONEANGLE           0.0
HUBHEIGHT            5.2
NROFBLADES          3
PITCHANGLE          -2.3
RPM                  424.5
STARTAZIMUTH        0.0
TBEGIN               0.0
TEND                 3.0
TILTANGLE             0.0
TIMESTEP              0.001
XNAC2HUB            -2.13
YAWANGLE              0.0
ZNAC2HUB              0.37
!

!----- Airfoil data -----
!
```

```

!-----
COEFFFILENAME           airfoils.dat
CORR3DTYPE              1 !0: No correction 1: Snel
DYNSTALLTYPE             1 !0: No DS 1:Snel1 2: Snel2 3:B-Leishmann 4: Onera
!-----
! Environment -----
!-----
AIRDENSITY               1.225
HORSHEAR                  0.0
SHEAREXP                  0.0
TOWERBASERADIUS            0.5
TOWERTOPRADIUS              0.5
WINDFILENAME                wind.dat
WINDTYPE                   1      ! 1: Simple 2: SWIFT 3: TurbSim
!-----
!BEM -----
!-----
NROFBEMELEMENTS           14
PRANDTLROOT                 1
!
!AWSM -----
!-----
STREAMWISEWAKEPOINTS        541
FREESTRMWISEWAKEPOINTS       540
WAKEREDUCTIONSTART            0
WAKEREDUCTIONSSKIP            0
PARALLEL                     0 !0: off 1: on

```

### 3.2.2 Wind

The WINDTYPE and WINDFILENAME keywords specify the wind field. In the case of a simple wind field as described in 2.1.1, an ASCII input file is used that needs to be placed in the working directory. This file should consist of four columns with the time and the three wind speed components, expressed in the fixed inertial coordinate system. An example of the content of a simple wind file is included below.

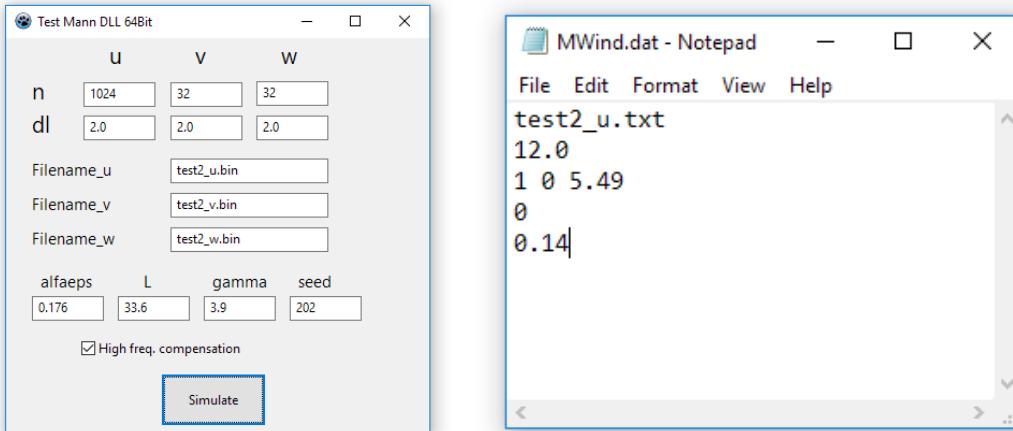
time [s]	u [m/s]	v [m/s]	w [m/s]
0.0	10.0	0.0	0.0
5.0	10.0	0.0	0.0
10.0	10.0	0.0	0.0
20.0	10.0	0.0	0.0

Simple wind example file

In the case of a SWIFT, Turbsim or Mann wind field, the binary file as made by the relevant program is read in during the initialization step. For large wind turbines and long simulation times, it is the responsibility of the user to make sure that enough computer internal memory is available for storing the wind field. If the calculation time exceeds the duration of the specified wind field, the wind field is repeated for both the simple and stochastic wind options. The position of both the simple and stochastic wind fields is defined in the fixed inertial coordinate system.

Irrespective of the wind type chosen, an optional wind ramp can be specified at the start of the simulation (to facilitate initialization of AWSM calculations) using the RAMPTIME and RAMPFACTOR keywords.

**Mann wind fields** The input of Mann type wind fields deserves some special attention. Running the Mann 64bit turbulence generator [2], a simple user interface appears as shown in Figure 13a. For setting the appropriate input values, the reader is referred to dedicated publications on this



(a) User interface of the Mann generator      (b) Example of ECN Aero-Module Mann wind input file

Figure 13: Screenshot of Mann wind field related input

matter. File names for the turbulence fields for the three velocity components are entered in the center part of the form. After pressing the Simulate button the program generates the three files of the turbulence field (u-, v-, and w-direction). A fourth file, containing a summary of the input of the generated turbulence field, is also generated. The name of this file is the same as Filename\_u with a different extension (.txt). Now we need to prepare the input files of the ECN Aero-Module in order to use the generated Mann wind field.

- Create a Mann wind input file with the following input lines (example in Figure 13b):
  1. File name of the summary text file that has been generated by the Mann wind field generator (`mann_64bit.exe`).
  2. Wind speed at the center of the Mann box that is added to the Mann fluctuations.
  3. Location of the center of the front surface of the Mann box (x, y, z) in meters w.r.t. the inertial reference frame.
  4. Time shift of the Mann wind field in seconds.
  5. Target turbulence intensity (optional, choose zero value for no target). When target  $T_I > 0$ , the Mann wind field will be scaled (equally in all directions) to match the target value in x-direction. In that case also an offset will be applied to match the mean wind speed at the center of the box with the supplied wind speed from LINE 2 (x-direction), to correct for the mean of the Mann fluctuations not being zero.
- In the main input file, set the name of the above created Mann wind input file at `WINDFILENAME`.
- In the main input file, choose `WINDTYPE 4`.
- In the main input file, choose a value for either `Z0` or `SHEAREXP` for a logarithmic or power law average wind profile. When both parameters are zero, a uniform average wind profile will be used.
- Make sure that the ECN Aero-Module Mann wind input file, the Mann summary file (.txt) and the three binary files (.bin) are located in the input directory.

### 3.2.3 Airfoil data

The COEFFFILENAME keyword specifies the filename that contains the blade profile filenames. The file merely consists of the filenames that contain the aerodynamic coefficients, placed in random order. An example of the content of this file is included below.

---

```
____ Blade profile filenames example file _____
! table with airfoil files
cylinder.dat
du91w2250.dat
risoea121.dat
naca64418.dat
```

A profile coefficient file contains lift, drag and moment coefficients as a function of angle of attack. The angle of attack data needs to comprise the range between -180 and 180 degrees. The coefficients can be specified in a four column table. However the possibility exists to specify the lift, drag and moment coefficients for different angle of attack series. In addition to that it is possible to specify multiple polars within one file for different Reynolds numbers. An example is included below.

---

```
____ Profile coefficient example file _____
! ECN Aero-module input file for airfoil data
! source: ATG version 3.1.0, december 2007

Airfoil_Name Cylinder
t/c          1.0           ! thickness ratio w.r.t. chord

format 1      !  1: alfa-cl-cd-cm      ; 2: alfa-cl; alfa-cd; alfa-cm

Reynolds_Nr 1.0E+05
-180. 0. 1. 0.
    0. 0. 1. 0.
180. 0. 1. 0.

Reynolds_Nr 5.0E+05
-180. 0. 0.25 0.
    0. 0. 0.25 0.
180. 0. 0.25 0.

Reynolds_Nr 2.0E+06
-180. 0. 0.4 0.
    0. 0. 0.4 0.
180. 0. 0.4 0.

Reynolds_Nr 1.0E+07
-180. 0. 0.6 0.
    0. 0. 0.6 0.
180. 0. 0.6 0.
```

### 3.2.4 Setting up an AWSM analysis

As mentioned in the previous paragraphs, AWSM is a time dependent code. In order to obtain reliable results, it is necessary to properly set up the parameters in AWSM. The scope of this paragraph is to provide some guidelines to set some of AWSM parameters.

- **TEND**

AWSM needs some time to initialize and reach a steady condition, because the wake needs to develop from zero wake length. Unless the objective of the analysis is to investigate the response of the rotor at the very beginning, some minimal duration for the simulation should be ensured. A good preliminary estimation of TEND can be done considering the diameter of the rotor, the wind speed and the expected axial induction (equation 3.2). This rule of thumb basically states that the rotor wake needs to be developed approximately 3 diameters downstream in order to obtain valid results. An estimate for the axial induction can easily be obtained from a BEM simulation. In case of coupled execution (e.g. with SIMPACK), TEND is not used, but the corresponding parameter in the structural code.

$$TEND > 3 \frac{D}{U_{\infty}(1 - a)}, \quad (3.2)$$

with

D	[m]	rotor diameter
$U_{\infty}$	[m/s]	undisturbed wind velocity
a	[-]	rotor average axial induction factor

- **Time step**

In case of structural-coupled analysis, this value is prescribed by the structural code. For stand-alone calculations this parameter is defined by the **TIMESTEP** keyword. In order to have a good numerical stability and accuracy of the results, a value should be assigned that produces angular steps not larger than 10 degrees. However in many applications (e.g. aero-elastic calculations) the time step requirement will already yield smaller values than this requirement. It is noted that application of small time steps leads to high computation times. In addition to that it can be observed that AWSM will not run for very small time steps.

- **STREAMWISEWAKEPOINTS**

**STREAMWISEWAKEPOINTS** parameter is directly connected to **TIMESTEP** and **TEND** since it states how many wake points there are in the specified time interval. In general, a value corresponding to a convected wake distance of 3 rotor diameters (similar to equation 3.2) is recommended.

- **FREESTRMWISEWAKEPOINTS**

AWSM can work with a fixed or free wake. A free wake is more accurate but also time consuming. Depending on the application, the number of free wake points should ideally correspond to a convected wake distance of 2 rotor diameters. This is considered a good compromise between accuracy and computational time. The number of points necessary to achieve this can be estimated analogue to equation 3.2. Please note that from the free to fixed wake transition onwards, AWSM will apply a convection speed using the blade average induction of the last free wake point.

- **Prescribed wake**

In addition to a fixed or free wake operation the **PRSCRBWAKE** keyword allows to prescribe the axial wake convection speed. In cases where a small time step size is required, it can be too computational expensive to use a free wake distance of at least two diameters. The **PRSCRBWAKE=1** option allows to estimate the convection speed of the fixed wake points based on the calculated blade induction, thereby enforcing dependency on the operational condition. Although this convection can further be tuned using the **CONVECFATOR** keyword, it is not recommended to change this value for **PRSCRBWAKE=1**. To still capture some of the near wake dynamics a good compromise can be found by using a number of free streamwise wakepoints corresponding to at least 360° azimuth divided over the number of blades (e.g. a minimum of 12 points for a 3 bladed rotor with a timestep of 10°). As long as the axial induction factor roughly is below 0.3 this method was shown to yield good results compared to free wake simulations, although dependency on time step size has not been verified.

- Wake reduction

In case of small timesteps (aero-elastic calculations), the number of wake points needed to cover the required wake length is numerous. To reduce the required number of streamwise wake points, a skipping routine is implemented that skips WAKEREDUCTIONSKIP streamwise wakepoints after WAKEREDUCTIONSTART wake points. As such the STREAMWISEWAKEPOINTS and FREESTRMWISEWAKEPOINTS have to be adjusted accordingly. The associated vorticity of the newly formed vortex lattice is taken equal to the last vortex element before skipping. The settings are to be chosen as a trade-off between accuracy and computational effort. As a rule of thumb for HAWTs, the azimuthal discretization of the shed vorticity lines can be as rough as  $10^\circ$  after a wake convected distance amounting to roughly 0.5 rotor diameters.

For example, a HAWT simulation featuring a 200 m diameter rotor at 8 m/s wind and 9 rpm, with an axial induction factor of 0.2 at a timestep resulting in a  $2^\circ$  azimuth step would need around 2500 wake points to have the wake as long as 3 diameters as can be verified using 3.2. If we would like to reduce the azimuthal discretization to  $10^\circ$  after the rotor wake has convected approximately 0.5 diameter downstream, we can set WAKEREDUCTIONSTART to 400 and WAKEREDUCTIONSKIP to 4, in which case we would need to change STREAMWISEWAKEPOINTS to  $400 + (2500 - 400)/(4+1) = 820$ . A similar calculation can be performed for the FREESTRMWISEWAKEPOINTS parameter. It is recommended to have a look at [5] and to experiment with this technique before applying it. Less experience exists applying this technique to VAWTs, where it is anticipated that due to the large variation of circulation with azimuth angle, merging shed vorticity lines in close proximity to the rotor can have a large effect on the results in comparison to HAWTs.

- Wind set-up

Sometimes AWSM experiences numerical stability problems. Often, those problems can be related to the starting vortex. In order to improve AWSM stability, an increasing value for the wind speed can be prescribed, starting off with a relatively low wind speed. In this way, the loads along the blade are gradually increasing up to the desired value. A ramp up time RAMPTIME which covers approximately  $90^\circ$  rotor revolution and a RAMPFACTOR of 0.3 are rough guidelines for normal operating conditions. However especially the necessary value of RAMPFACTOR can strongly depend on the operating conditions under consideration.

- Convergence

Apart from the wind ramp, several other measures can be taken to promote convergence. In case of divergence, the Aeroresults file (see section 3.3) can be viewed to inspect the development of the vorticity distribution in time. This usually reveals the cause for the divergence. Sometimes commenting out the cylindrical root sections in the AEROPROPS table (using an exclamation mark) solves the problem. In addition to that care should be taken with the radial distribution of corner points from the AEROPROPS table. Specification of these points too close to each other can also hamper convergence. Apart from these issues it must be noted that especially for load cases with high axial induction (towards the turbulent wake state), convergence can be hard to obtain.

- Parallelization

Calculating the induced velocities at the free wake points is computationally expensive, increasing with vortex lattice dimensions. This calculation can be parallelized with OpenMP® using the PARALLEL keyword. The parallelization is performed over the free streamwise wake point calculation loop. Here it is assumed that the streamwise and spanwise number of wake points is the same between the blades, which is always the case for the ECN Aero-Module application. The OpenMP® library needs to be available for this application to work. The relevant file (libomp5md.dll for Windows applications) needs to be accessible for the ECN Aero-Module, e.g. by sharing it in the path. It is possible to control the number of processors used by the application by defining the relevant OMP\_NUM\_THREADS environment variable in linux or windows.

## 3.3 Output

The ECN Aero-Module automatically creates the specified output directory to which several logfiles and optional debugfiles are written.

### 3.3.1 Logfiles

- **AerofoilData.out**  
This file contains a summary of the airfoil data that has been read in from the files specified through COEFFFILENAME.
- **AeroOutput.txt**  
This file contains the resulting wind statistics and aerodynamic axial force, torque and flapwise blade root moment per blade as a function of time and azimuth angle. For a HAWT, axial force is in the x-direction of the hub coordinate system. For a VAWT the axial force is defined in the x-direction of the inertial reference frame. The flapwise blade root moment is only calculated in case of a HAWT and set to zero for a VAWT because of the possible non-cantilevered suspension for this turbine type.
- **ErrorLog.dat**  
Warnings and error messages are written to this file.
- **LogFile**  
The name of this file needs to be specified in the main input file by means of the LOGFILENAME keyword. The file contains a summary of the used aerodynamic model by means of keywords values. Several geometric for the different blades and the hub height wind speed at the first timestep are given at the bottom of the file.
- **AeroPower.dat**  
This file is only written in the case of stand-alone execution and contains the aerodynamic power and axial force as a function of time and azimuth angle. For a HAWT, axial force is in the x-direction of the hub coordinate system. For a VAWT the axial force is defined in the x-direction of the inertial reference frame.

### 3.3.2 AWSM logfiles

If AWSM is used as solver, several specific output files are created.

- **logoutawsm.dat**  
This file contains trace information related to the AWSM simulation. This file is only written when the LOGTRACESWITCH keyword is enabled. The amount of outputted trace information can be adjusted using the TRACELEVEL keyword.
- **Aeroresults file**  
The name of this file needs to be specified in the main input file by the AERORESULTSFFILE keyword. This file contains, for each time step and for each body, the aerodynamic data along the body coming from AWSM calculation. The structure of the file is detailed below.
  - **X, Y, Z**  
These are the coordinates (in inertial reference system) of the control points determined by AWSM, in which the calculations have been done. It should be noticed that they are not the same points prescribed in the input file.
  - **Gamma**  
Gamma indicates the local circulation at specific location.

- **Alpha, C1, CD**  
They indicate the local angle of attack (in degrees), lift and drag coefficients at specific location.
- **Ui, Vi, Wi**  
These variables indicate the local induced velocities at the specified location. The velocities are expressed in the x, y and z-direction respectively of the inertial reference system.
- **Striparea**  
The blade is divided in strips. Striparea indicates the area of each portion.
- **df-xyz-inert, df-xyz-part, df-xyz-chord**  
For each station of the blade, the forces acting at that sections are indicated, decomposed according the reference system. df-xyz-inert and df-xyz-part are identical and referred to the inertial reference system; so the first column is the axial component and the other two, the in plan components. df-xyz-chord is instead referred to the local chord, so the first column is tangential to the chord, the second one is the radial component and the third one is the component normal to the chord.
- **TIME**  
This parameter indicates the actual simulation time.
- **T P S J**  
These parameters provide information about the actual timestep and the location of each station (T=timestep, P=part, S=segment, J=station)

- **Geomresultfiles**

The name of this file needs to be specified in the main input file by the GEOMRESULTSFILe keyword. The information about the geometry of the bodies and the wake are stored in these files. Three files are generated with extension DAT1, DAT2 and DAT3 containing respectively data about the blade geometry, the blade geometry and the wake, only the wake. All these files are in ASCII format and they are formatted to be opened with Tecplot software. The coordinates are expressed in the inertial reference system.

- **DAT1**  
The GEOMRESULTSFILe.DAT1 file contains the coordinates (X,Y,Z) of the corner points of the blade geometry. For each timestep and each part, the corner points are printed. They are formatted in a matrix, where the columns are X,Y,Z coordinates, times the number of control points in chordwise direction (typically two: leading edge and trailing edge). The rows represent the stations in radial direction.
- **DAT2**  
The GEOMRESULTSFILe.DAT2 file contains the coordinates (X,Y,Z) of the corner points of the blade geometry and the wake. In addition to these data, also the vorticity is stored. The geometrical data are formatted in separated blocks with a matrix structure, where the columns represent the stations in radial direction and the rows, the stations in chordwise direction. Typically, a first block of data is written representing the X coordinates, followed by Y, Z and vorticity blocks. For the wake a similar structure is used, where the columns represent the stations in radial direction and the rows, the ones in axial direction. It should be noticed that the geometrical data refer to the corner points, while the wake geometry refers to the control points.
- **DAT3**  
The GEOMRESULTSFILe.DAT3 file contains the coordinates (X,Y,Z) of the wake. In addition to these data, also the vorticity is stored. The data are formatted in a matrix shape, where the columns represent the coordinates (X,Y,Z) and the vorticity, while the rows take into account the time development of the wake.

- Extfieldfiles

This file contains the detailed information about external field points. In particular, the location (in inertial reference system) of each point is indicated, followed by the pressure coefficient ( $C_p$ ) and the induced velocity components (in inertial reference system) for the specific point.

### 3.3.3 Specialist files

The writing of specialist files containing a more detailed description of several calculated aerodynamic variables is optional and can be specified using the `DEBUGFILE` keyword in the main input file. A separate file is written for each variable and blade. The first two columns of each file consist of the time and azimuth angle, the remaining columns contain the variable value at the different element locations as indicated by the header. An overview of the significance of the several variables is given in Table 4.

Since these files are kept open during time stepping, care must be taken for the structural solver not to use the same output FORTRAN units. Therefore the FORTRAN unit numbers corresponding to the specialist files are specified above 1000.

Table 4: Summary of available specialist file variables

name	unit	BEM	AWSM	variable
rCurvi	[m]	x		curvilinear radial distance (Figure 9)
rDef	[m]	x		deformed radial distance (Figure 9)
counter	[‐]	x		number of iterations for BEM iterative loop
AOA	[°]	x	x	angle of attack used for table lookup
AOA_C14	[rad]	x		angle of attack evaluated at quarter chord location
n	[N/m]	x	x	distributed force in normal direction, normal to chord, positive from airfoil pressure to suction side
t	[N/m]	x	x	distributed force in tangential direction, parallel to chord positive from trailing to leading edge
Gamma	[m <sup>2</sup> /s]		x	vortex line strength $\Gamma$ (equation 2.18)
C <sub>l</sub>	[‐]	x	x	lift coefficient
C <sub>d</sub>	[‐]	x	x	drag coefficient
C <sub>m</sub>	[‐]	x	x	moment coefficient
C <sub>n</sub>	[‐]	x		normal force coefficient
C <sub>t</sub>	[‐]	x		tangential force coefficient
u <sub>Ind</sub>	[m/s]	x	x	local rotor induced axial velocity, ref hub (HAWT) or inertial (VAWT)
v <sub>Ind</sub>	[m/s]	x	x	local rotor induced tangential velocity, ref hub (HAWT) or inertial (VAWT)
w <sub>Ind</sub>	[m/s]	x	x	local rotor induced radial velocity, ref hub (HAWT) or inertial (VAWT)
u <sub>IndAnnAv</sub>	[m/s]	x		annulus averaged rotor induced axial velocity (hub coordinate system)
dUindDt	[m/s]	x		time derivative of annulus averaged rotor induced axial velocity
u <sub>Eff</sub>	[m/s]	x		effective resultant incoming velocity at an element
chord	[m]	x	x	local chord
Re	[‐]	x		Reynolds number
F_prandtl	[‐]	x		Prandtl factor
f_skew	[‐]	x		skew factor
posC14x	[m]	x	x	x-position of quarter chord point w.r.t. inertial system
posC14y	[m]	x	x	y-position of quarter chord point w.r.t. inertial system
posC14z	[m]	x	x	z-position of quarter chord point w.r.t. inertial system
DirChordx	[rad]	x		angular orientation of quarter chord point w.r.t. inertial x-axis
DirChordy	[rad]	x		angular orientation of quarter chord point w.r.t. inertial y-axis
DirChordz	[rad]	x		angular orientation of quarter chord point w.r.t. inertial z-axis
velStrucC14x	[m/s]	x	x	structure velocity in element (BEM) or inertial (AWSM) x-direction
velStrucC14y	[m/s]	x	x	structure velocity in element (BEM) or inertial (AWSM) y-direction
velStrucC14z	[m/s]	x	x	structure velocity in element (BEM) or inertial (AWSM) z-direction
RotChordx	[rad/s]	x		angular velocity of quarter chord point around element x-axis
RotChordy	[rad/s]	x		angular velocity of quarter chord point around element y-axis
RotChordz	[rad/s]	x		angular velocity of quarter chord point around element z-axis
vTowCorC14x	[m/s]	x		tower induced velocity at quarter chord point in element x-direction
vTowCorC14y	[m/s]	x		tower induced velocity at quarter chord point in element y-direction
vTowCorC14z	[m/s]	x		tower induced velocity at quarter chord point in element z-direction
windC14x	[m/s]	x		wind velocity at quarter chord point in element x-direction
windC14y	[m/s]	x		wind velocity at quarter chord point in element y-direction
windC14z	[m/s]	x		wind velocity at quarter chord point in element z-direction
velIndx	[m/s]	x		rotor induced velocity at element in element x-direction
velIndy	[m/s]	x		rotor induced velocity at element in element y-direction
velIndz	[m/s]	x		rotor induced velocity at element in element z-direction

NB Filenames are preceded by blade number and followed by \_BEM or \_AWSM, e.g. B1uEff\_BEM.txt

## 4 Validation

A validation of the ECN Aero-Module has been performed by a comparison to available wind tunnel data. The known wind input in the tunnel enables a more accurate comparison compared to field data. The New Mexico [6, 7] and NREL UAE Phase VI [25] experiments are suitable for this purpose since they feature unsteady pressures sensors along the blade span and hence allow us to compare the azimuthal variation of the blade loading.

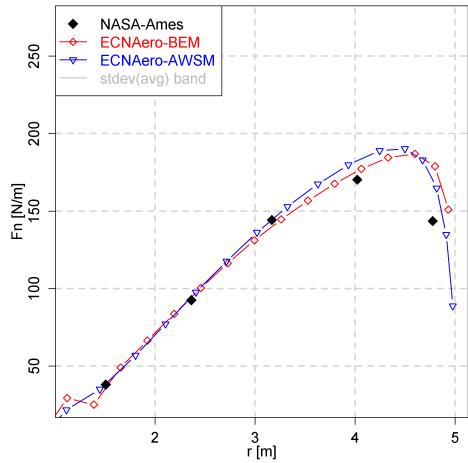
All calculations feature a dynamic stall model (DYNSTALLTYPE=1) and the default settings are used for the BEM and AWSM simulations. The yaw model of Schepers (YAWMODEL=1) has been used for the BEM simulations. Both simulations in axial and yawed flow are subject to comparison. In addition to that, the results for a dynamic operation (pitching step) are compared. More results including a coupled simulation (using SIMPACK as structural solver) compared to field measurements can be found in [4].

## 4.1 NREL UAE Phase VI

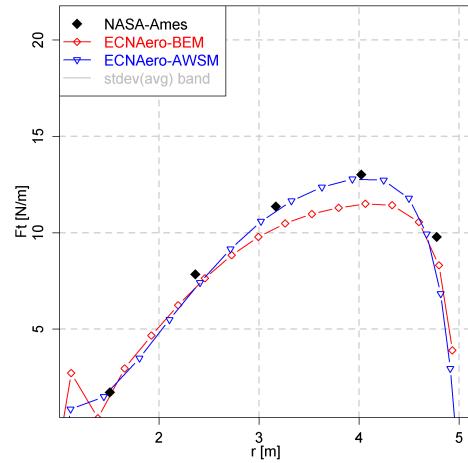
The NREL UAE Phase VI experiment in the NASA Ames wind tunnel features a 2 bladed 10 m diameter turbine. More details can be found in [25]. The chosen data points for comparison feature a rotational speed of 72 rpm. Firstly the radial distribution of normal and tangential force are compared for axial flow. Then the azimuthal variation of these sectional forces (at 30%, 47%, 63%, 80% and 95%) are compared for yawed flow ( $\text{Yaw}=30^\circ$ ) conditions. Finally the time history of the forces is given for two test cases which include a sudden pitch step.



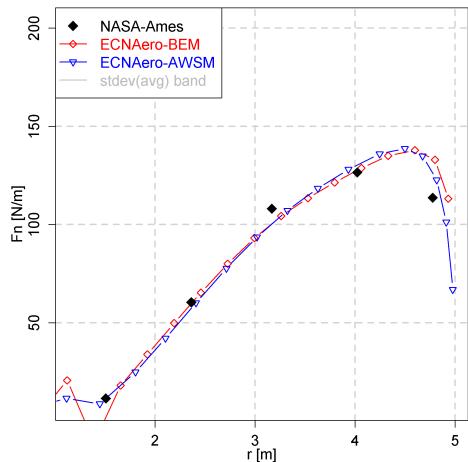
#### 4.1.1 Axial flow



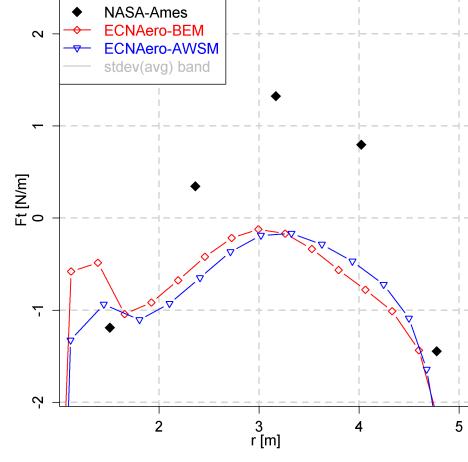
(a) Normal force,  $U_{\infty}=5$  m/s, pitch=0°, 72 rpm



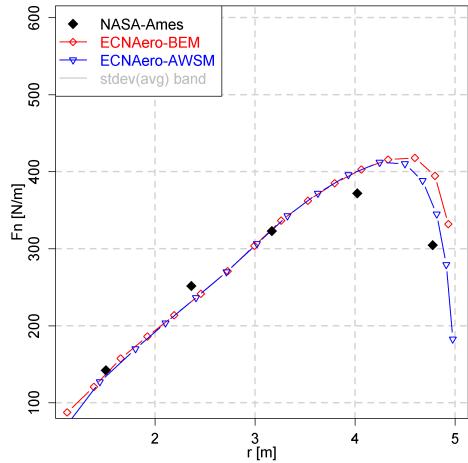
(b) Tangential force,  $U_{\infty}=5$  m/s, pitch=0°, 72 rpm



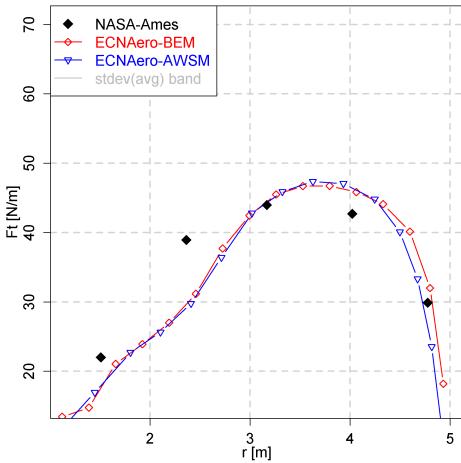
(c) Normal force,  $U_{\infty}=5$  m/s, pitch=3°, 90 rpm



(d) Tangential force,  $U_{\infty}=5$  m/s, pitch=3°, 90 rpm

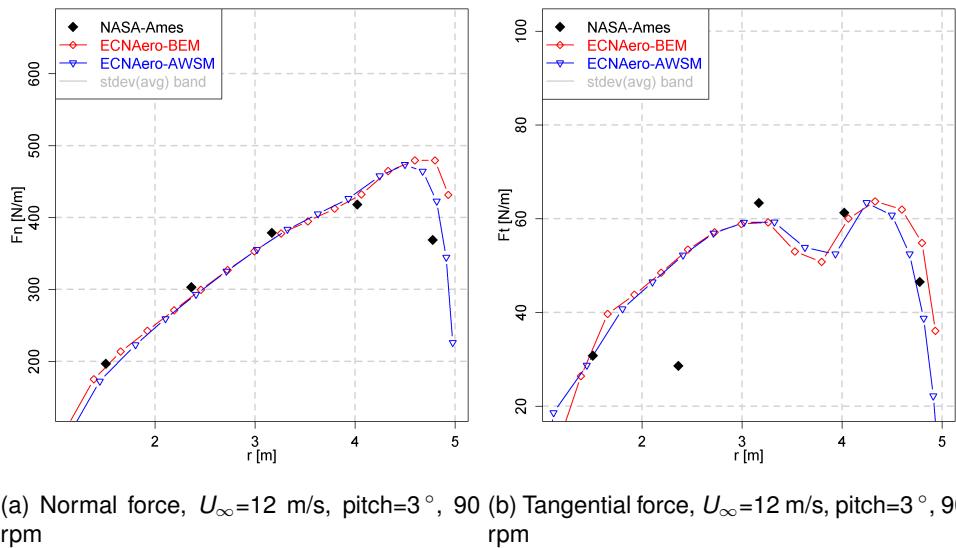


(e) Normal force,  $U_{\infty}=10$  m/s, pitch=3°, 90 rpm



(f) Tangential force,  $U_{\infty}=10$  m/s, pitch=3°, 90 rpm

Figure 14: Radial distribution of NREL UAE Phase VI sectional forces for axial flow conditions  
©2019 - Copyright TNO - All Rights Reserved



(a) Normal force,  $U_\infty=12$  m/s, pitch=3°, 90 rpm (b) Tangential force,  $U_\infty=12$  m/s, pitch=3°, 90 rpm

Figure 15: Radial distribution of NREL UAE Phase VI sectional forces for axial flow conditions, continued



#### 4.1.2 Yawed flow

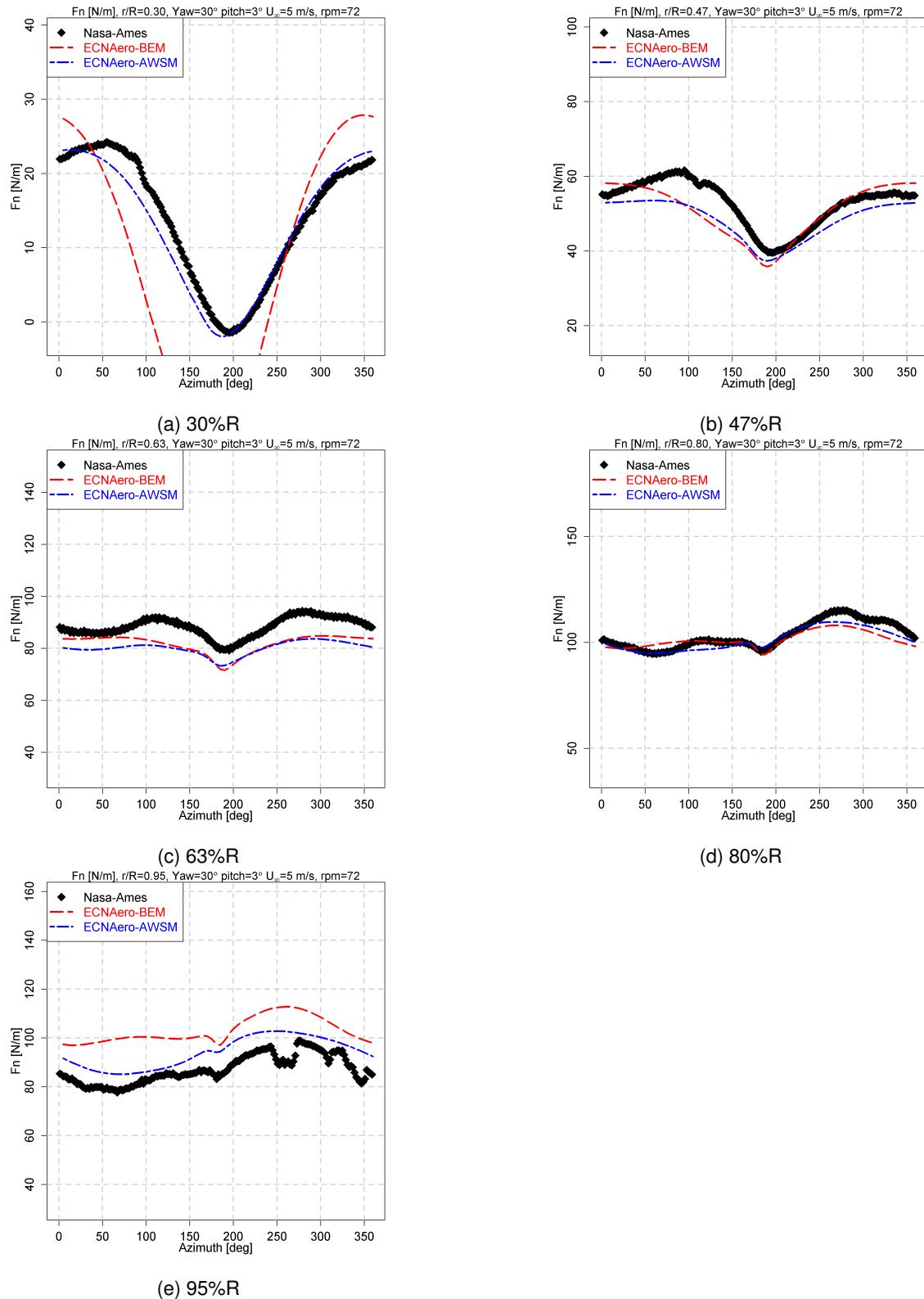


Figure 16: NREL UAE Phase VI normal force variation with rotor azimuth for yawed flow conditions ( $U_{\infty}=5$  m/s, 72 rpm, pitch=3°, Yaw=30°)

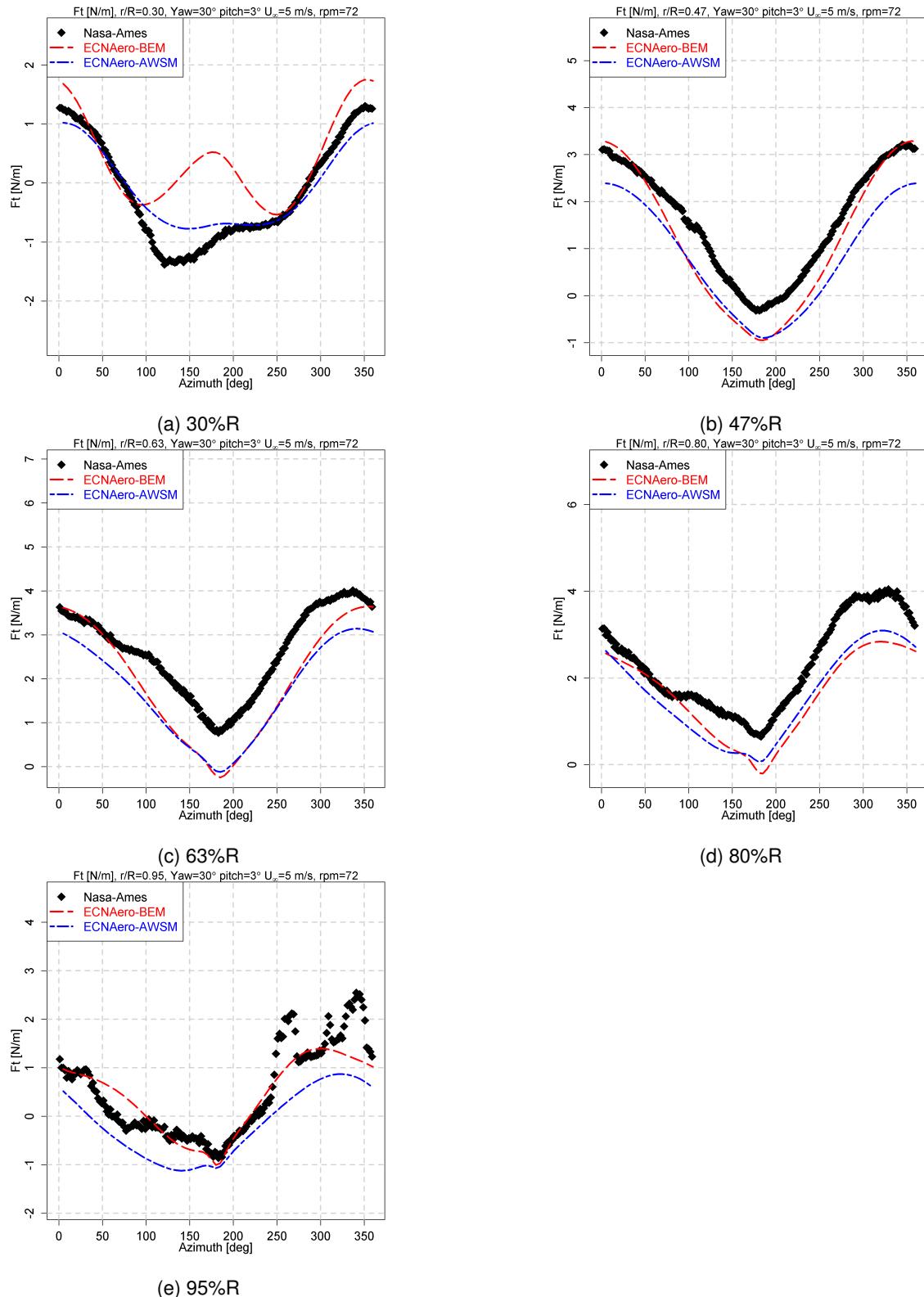


Figure 17: NREL UAE Phase VI tangential force variation with rotor azimuth for yawed flow conditions ( $U_\infty=5 \text{ m/s}$ , 72 rpm, pitch=3°, Yaw=30°)

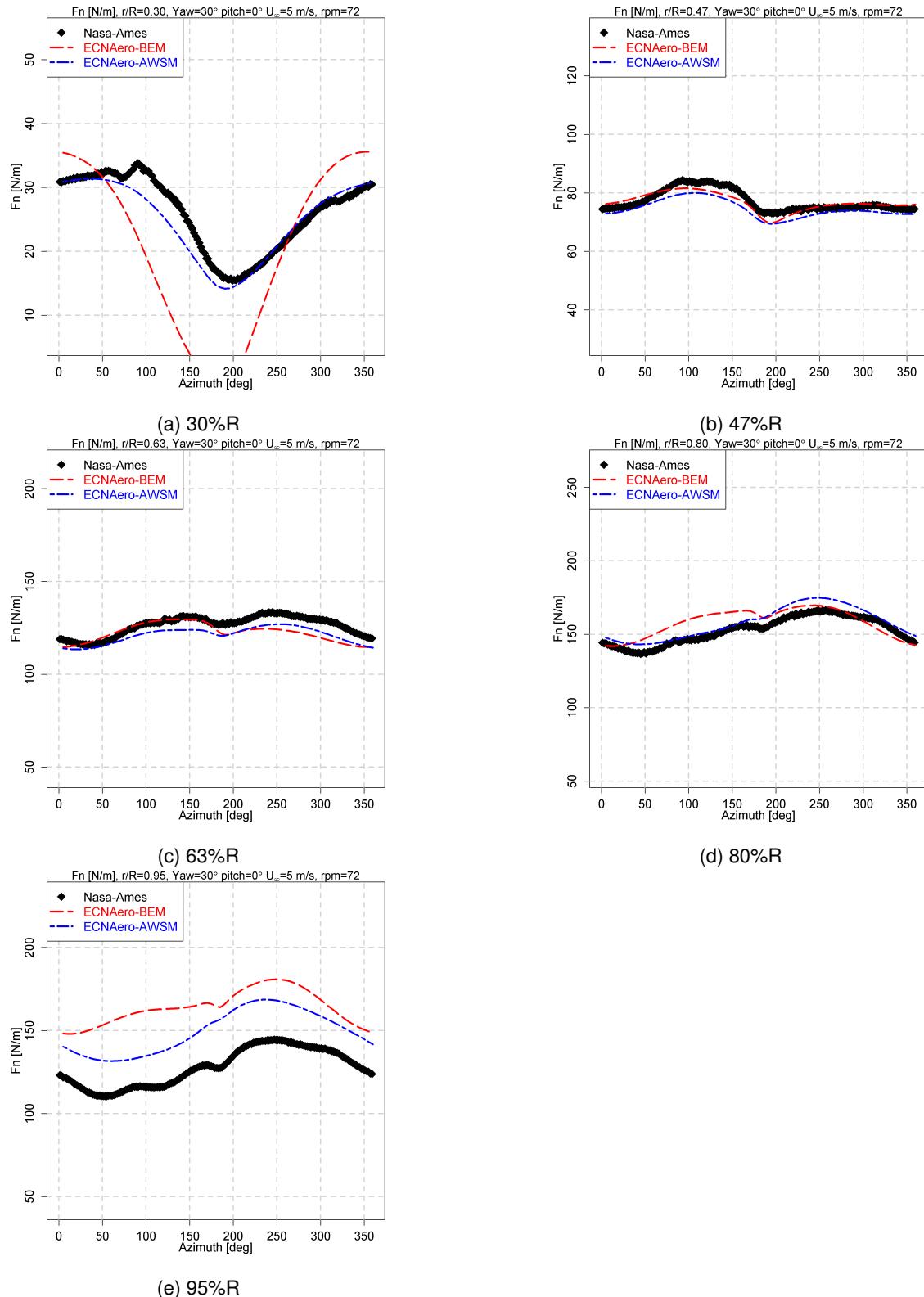


Figure 18: NREL UAE Phase VI normal force variation with rotor azimuth for yawed flow conditions ( $U_{\infty}=5$  m/s, 72 rpm, pitch=0°, Yaw=30°)

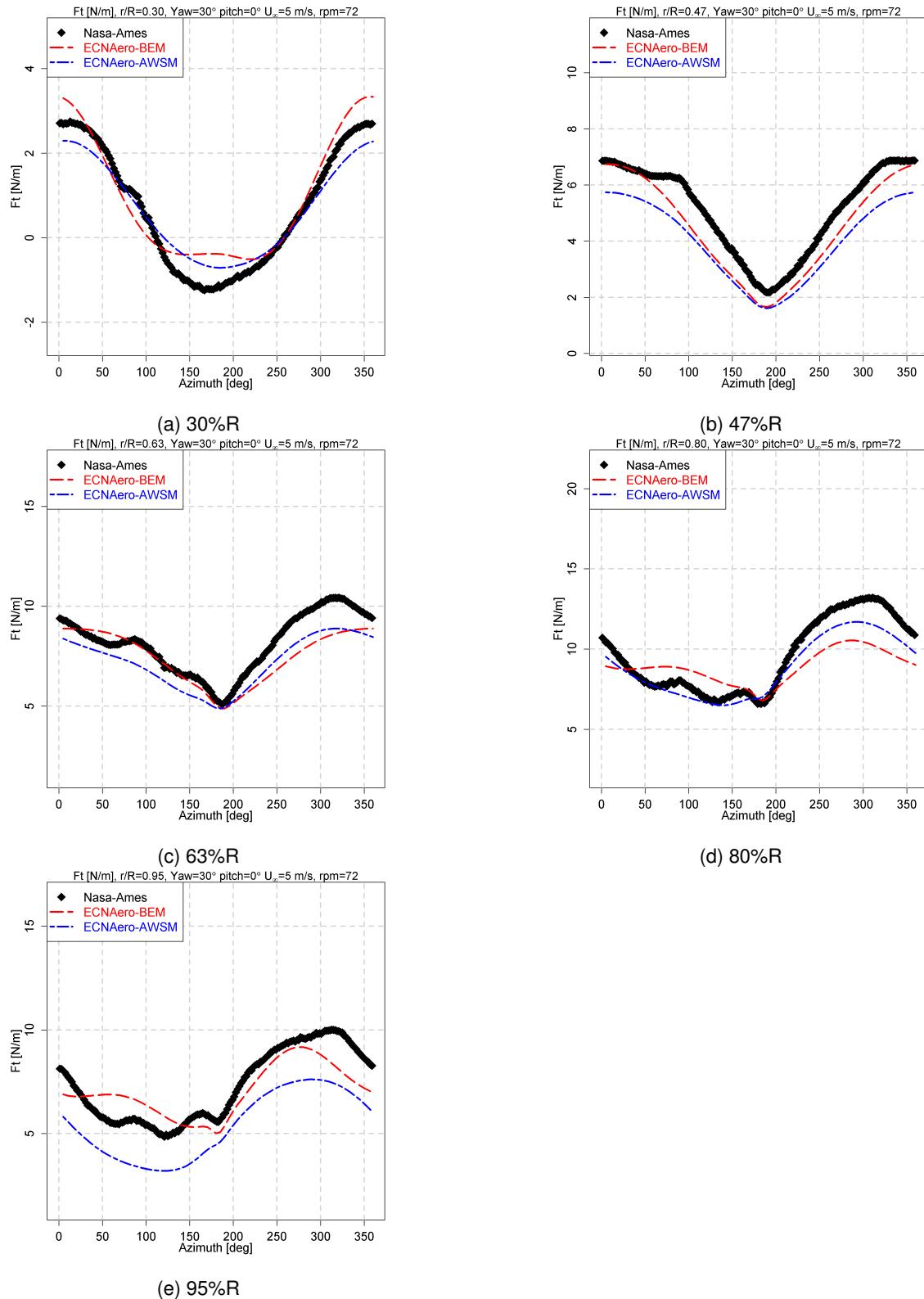


Figure 19: NREL UAE Phase VI tangential force variation with rotor azimuth for yawed flow conditions ( $U_{\infty}=5$  m/s, 72 rpm, pitch=0°, Yaw=30°)

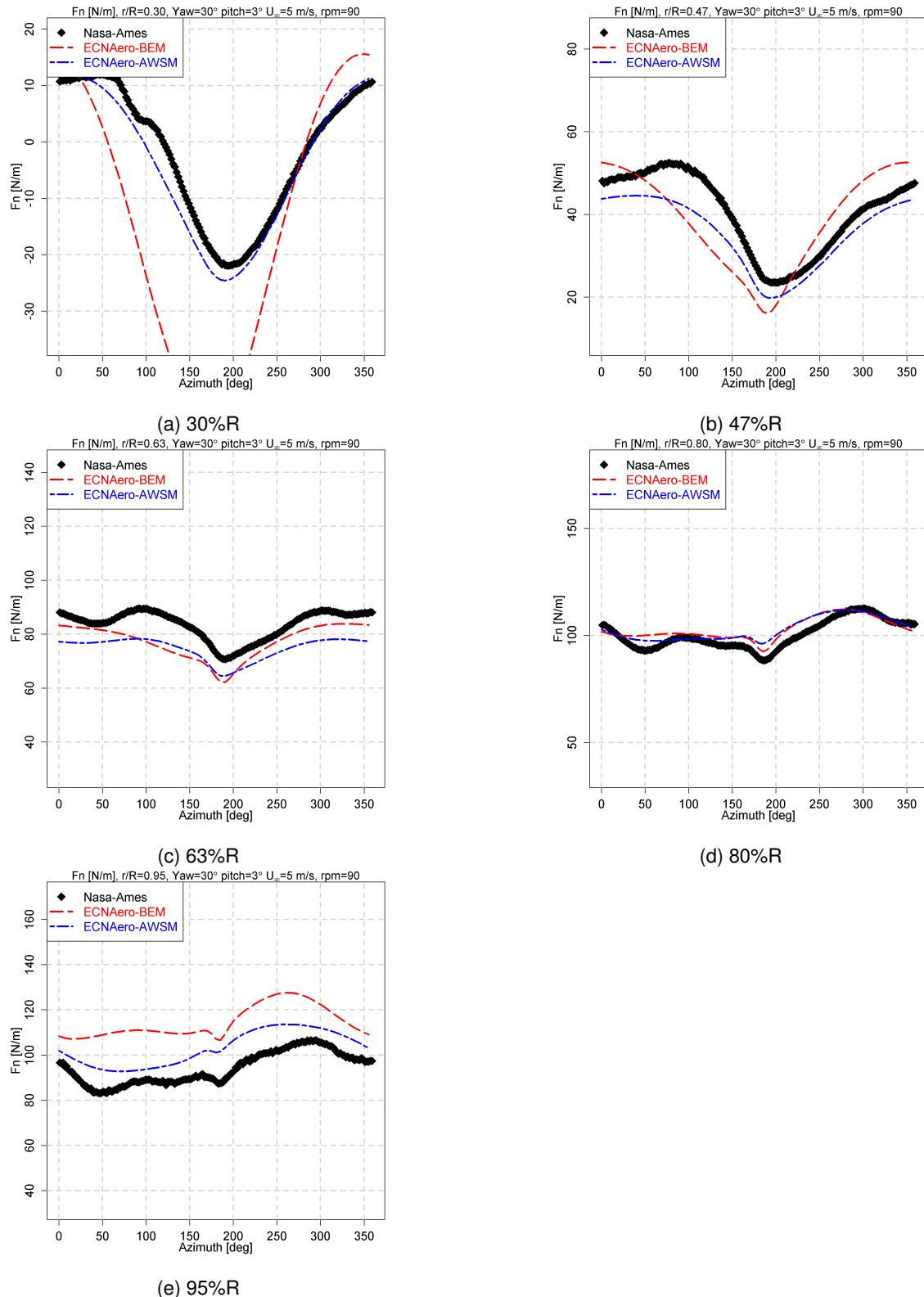


Figure 20: NREL UAE Phase VI normal force variation with rotor azimuth for yawed flow conditions ( $U_{\infty}=5$  m/s, 90 rpm, pitch=3°, Yaw=30°)

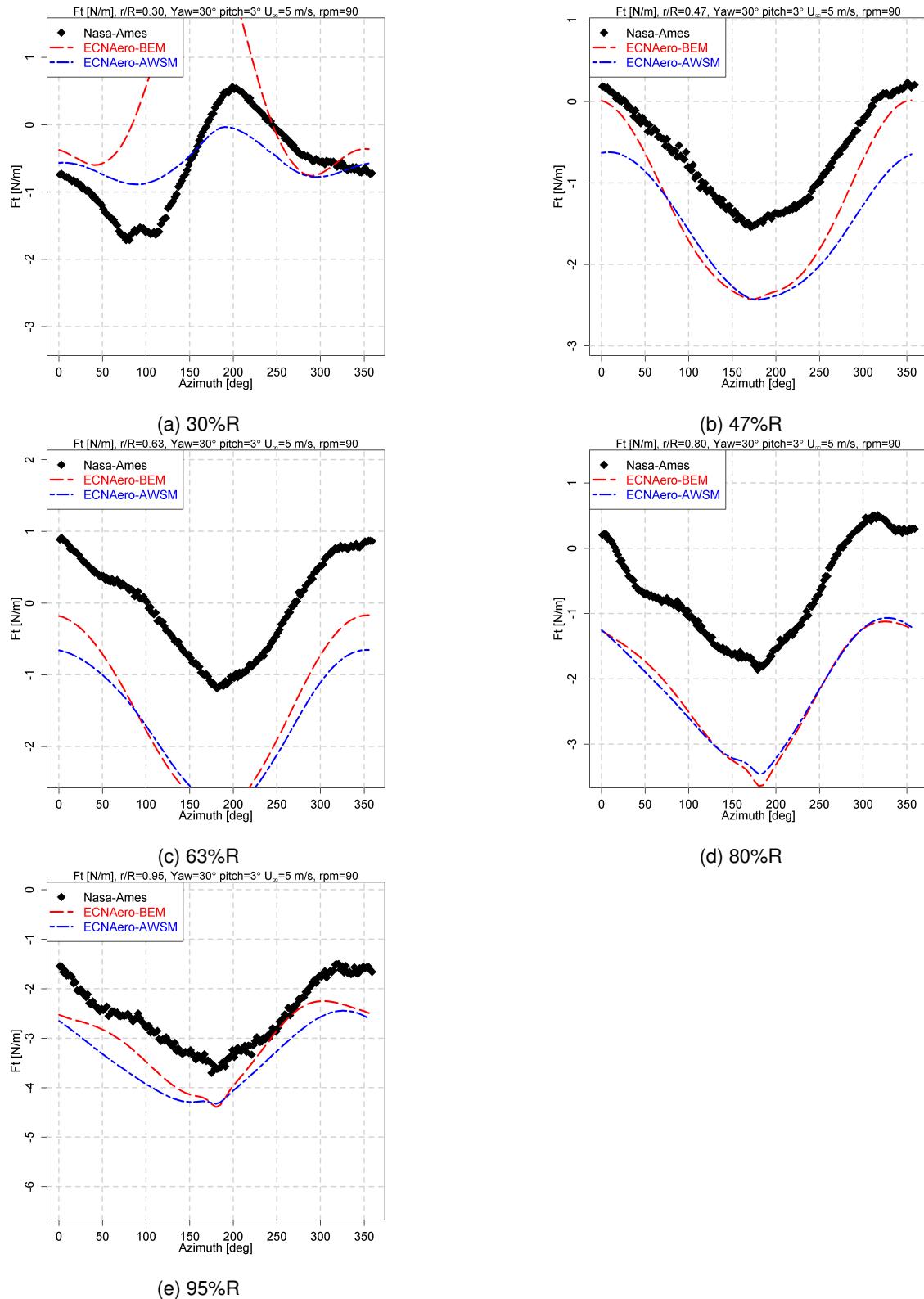


Figure 21: NREL UAE Phase VI tangential force variation with rotor azimuth for yawed flow conditions ( $U_{\infty}=5$  m/s, 90 rpm, pitch=3°, Yaw=30°)



### 4.1.3 Dynamic inflow

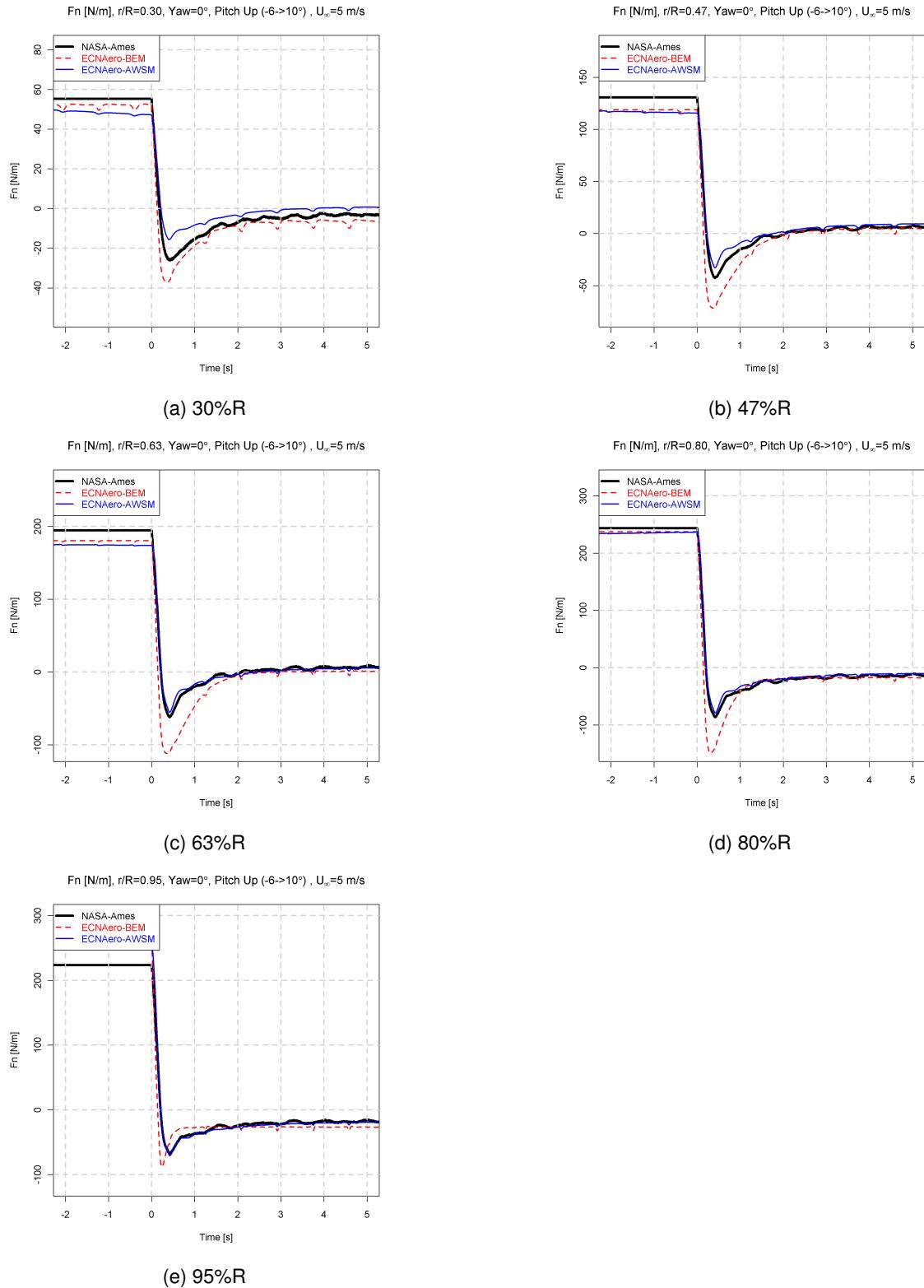


Figure 22: NREL UAE Phase VI normal force variation with time for an upward pitching step ( $U_\infty=5$  m/s, 72 rpm, pitch from  $-6^\circ$  to  $10^\circ$ )

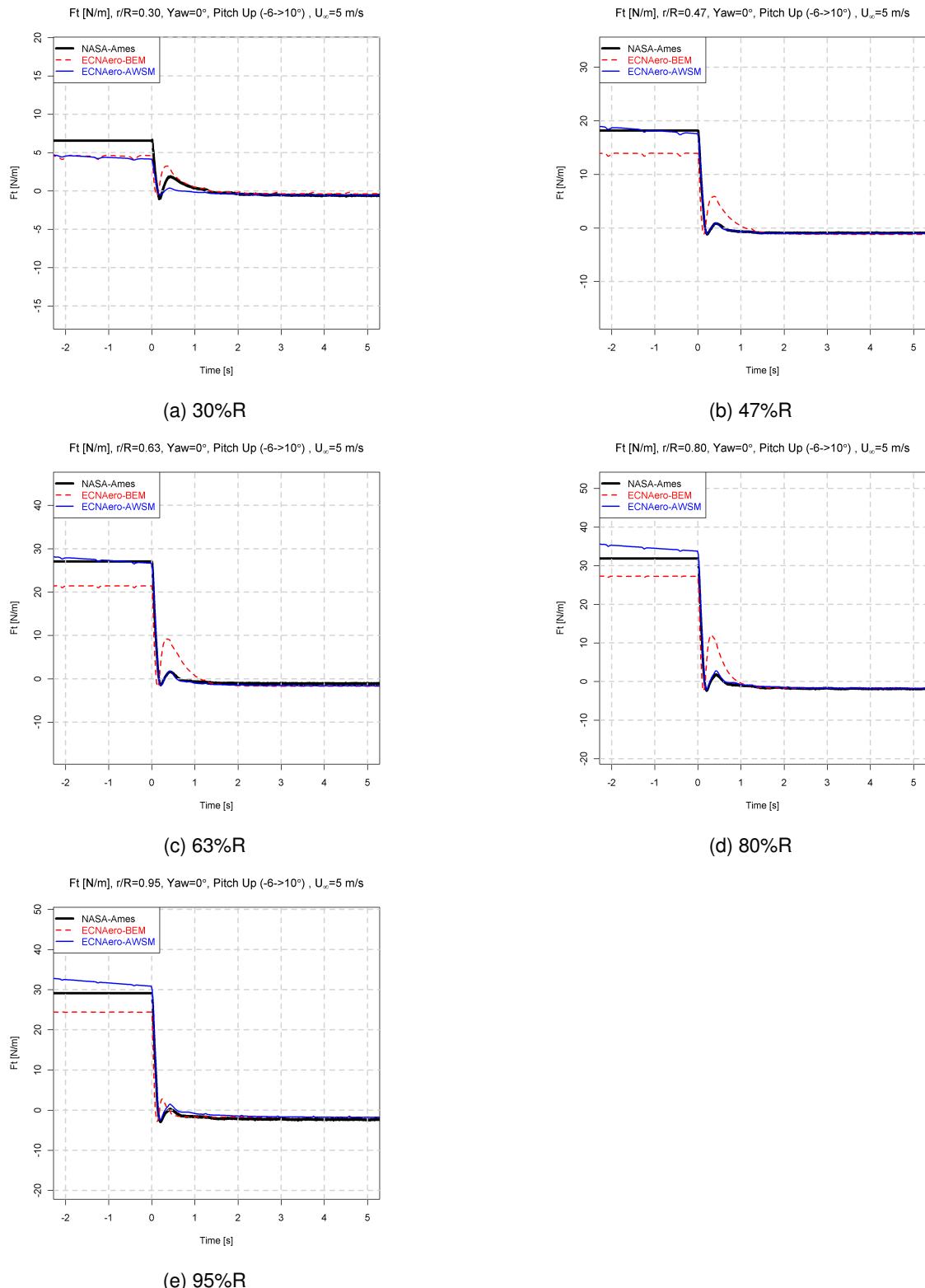


Figure 23: NREL UAE Phase VI tangential force variation with time for an upward pitching step ( $U_{\infty}=5$  m/s, 72 rpm, pitch from  $-6^{\circ}$  to  $10^{\circ}$ )

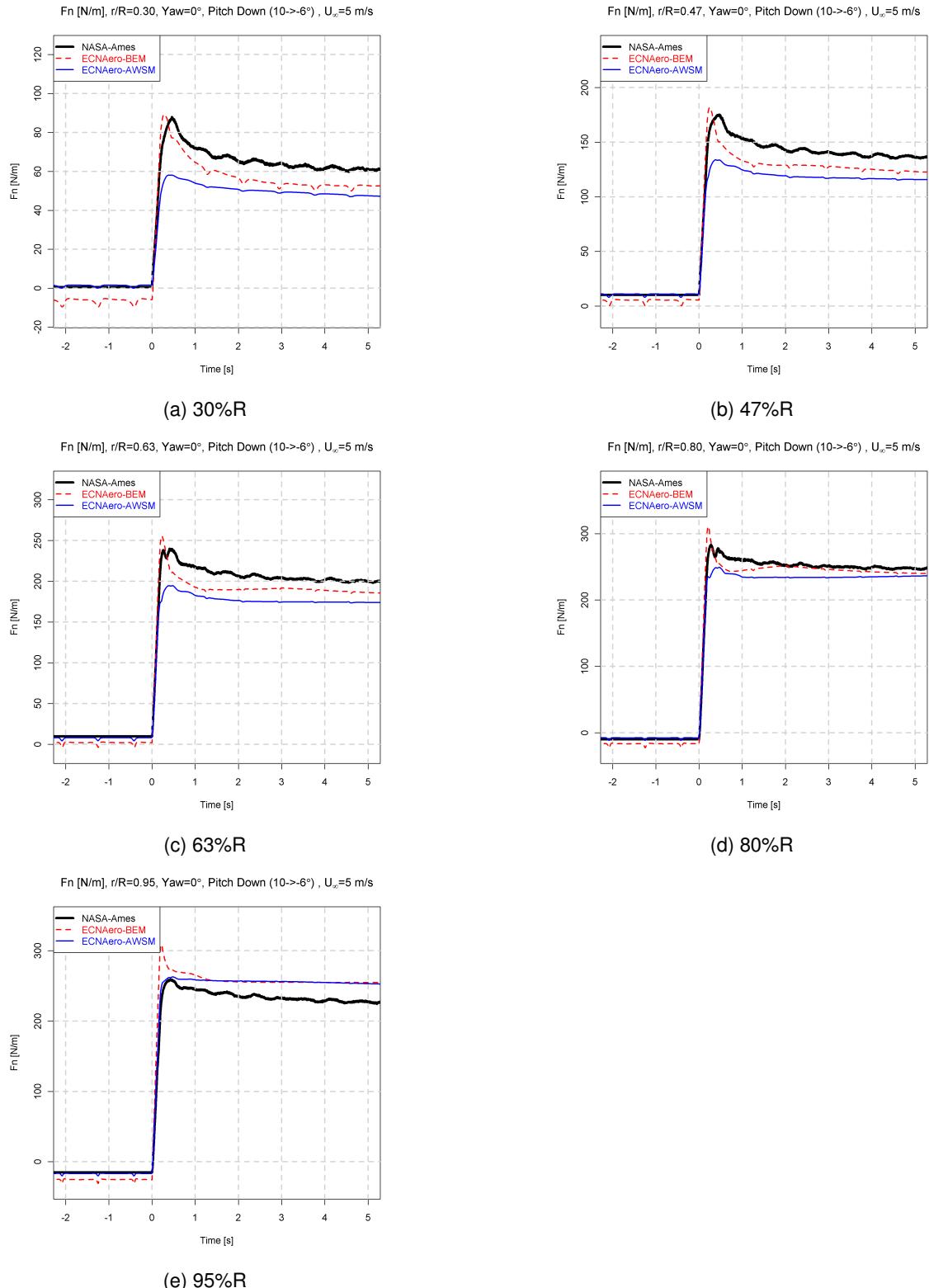


Figure 24: NREL UAE Phase VI normal force variation with time for an downward pitching step ( $U_{\infty}=5$  m/s, 72 rpm, pitch from  $10^{\circ}$  to  $-6^{\circ}$ )

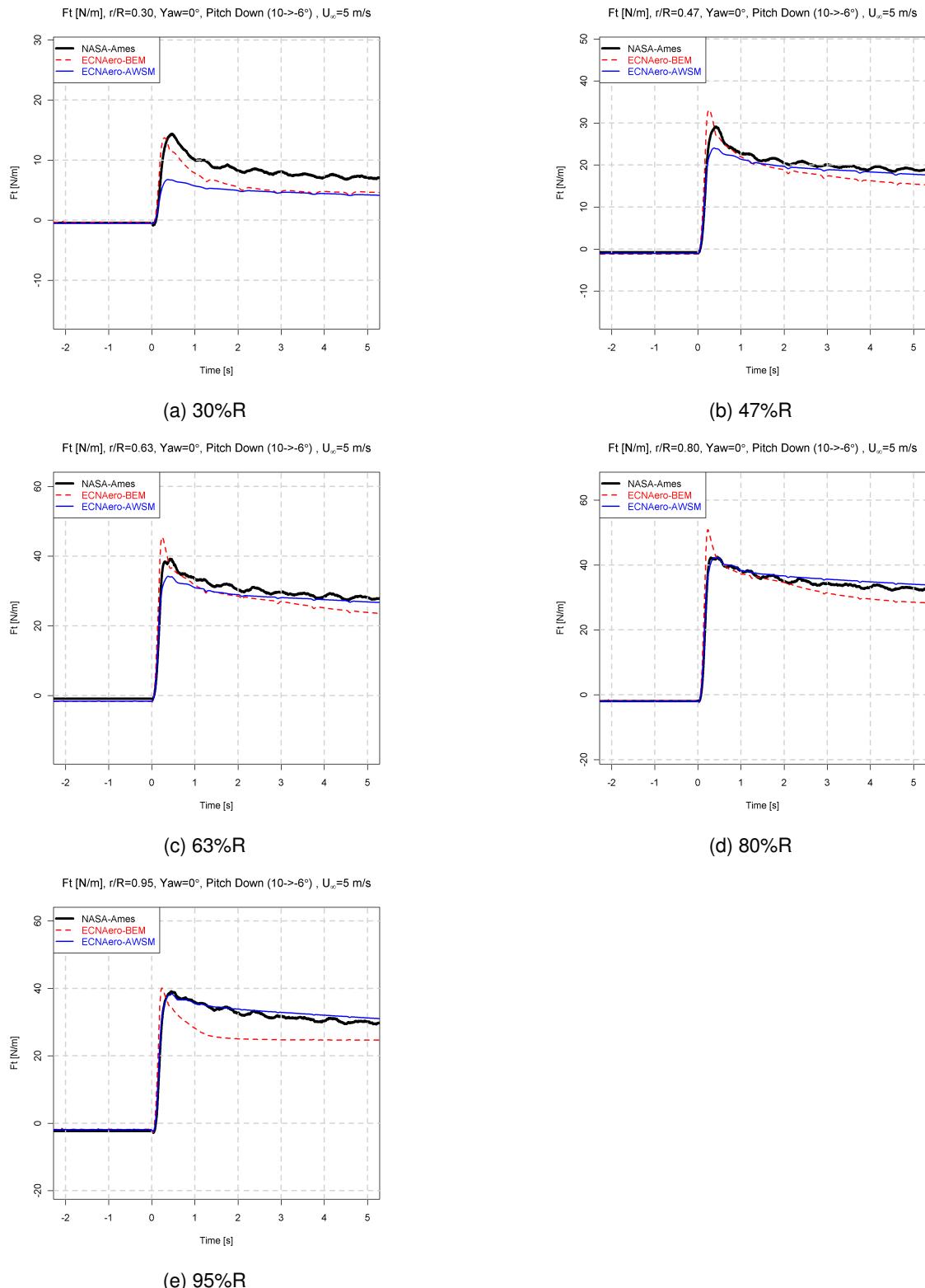


Figure 25: NREL UAE Phase VI tangential force variation with time for a downward pitching step ( $U_{\infty}=5$  m/s, 72 rpm, pitch from  $10^{\circ}$  to  $-6^{\circ}$ )

## 4.2 Mexico

The New Mexico experiments feature a 3 bladed 4.5 m diameter turbine positioned in the large low speed facility of the German-Dutch Wind Tunnels (DNW-LLF). More details can be found in [6, 7]. The chosen data points for comparison feature a rotational speed of 425 rpm and a pitch angle of  $-2.3^\circ$ . Firstly the radial distribution of normal and tangential force are compared for axial flow. Then the azimuthal variation of these sectional forces (at 25%, 35%, 60%, 82% and 92%) are compared for yawed flow conditions.

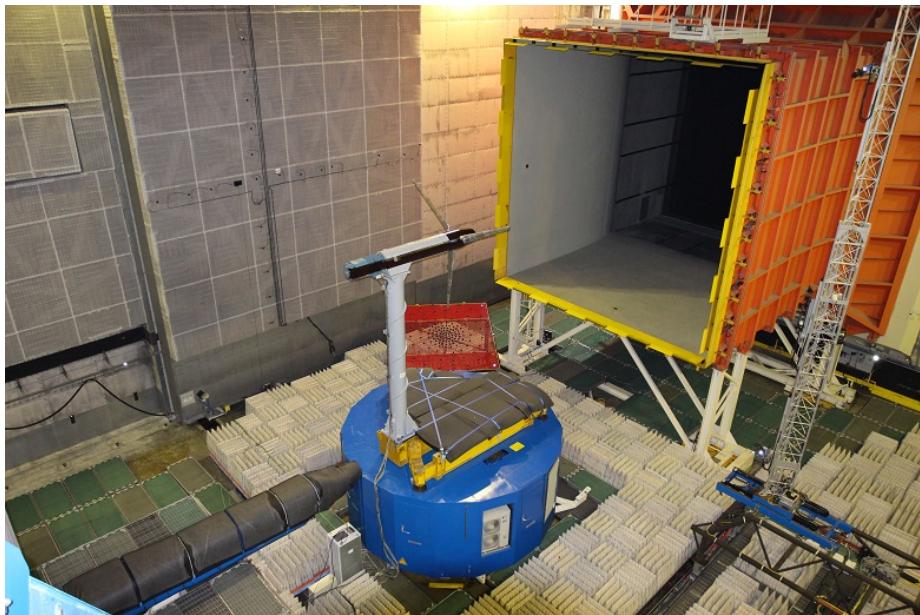


Figure 26: The New Mexico experiment



#### 4.2.1 Axial flow

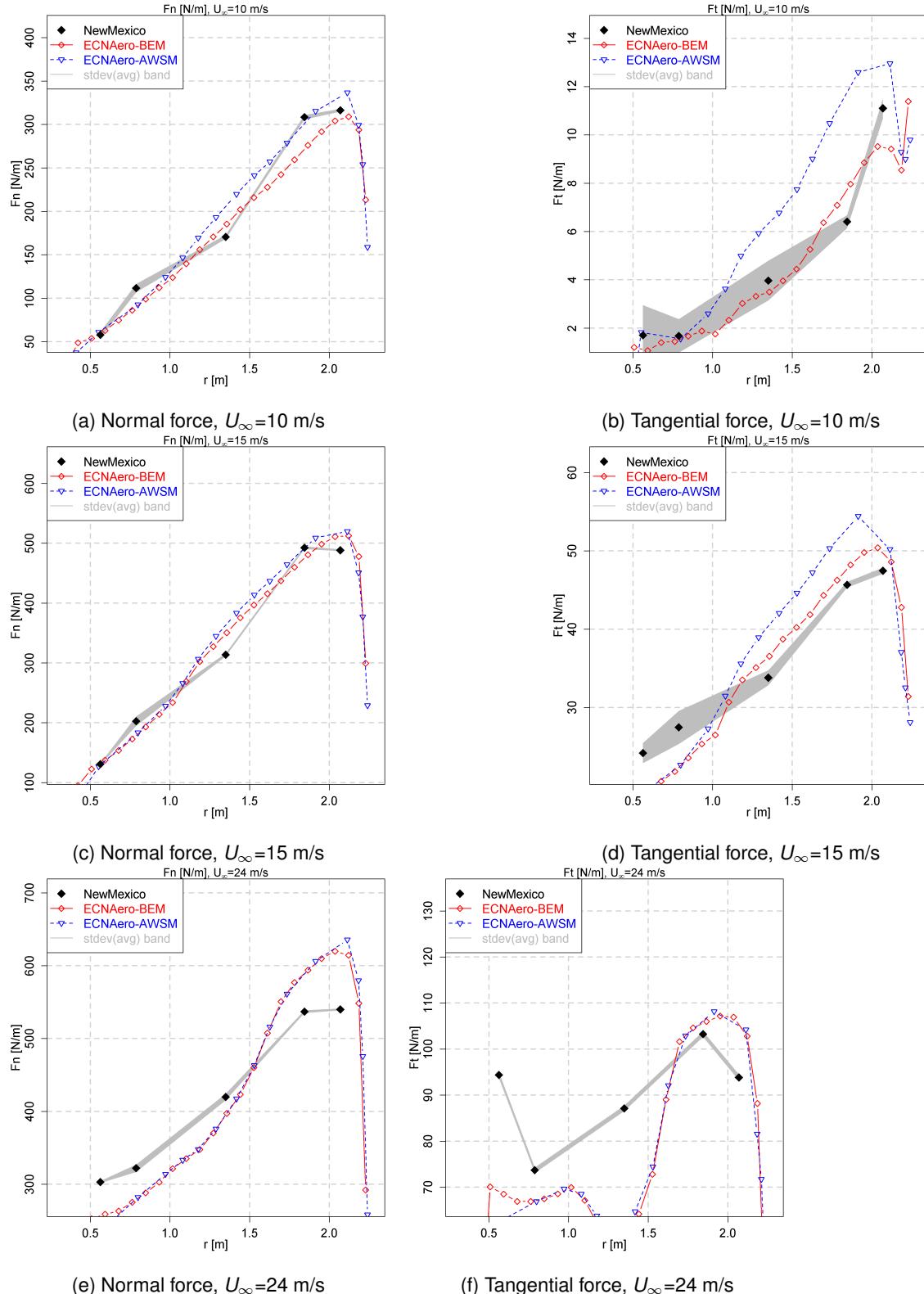
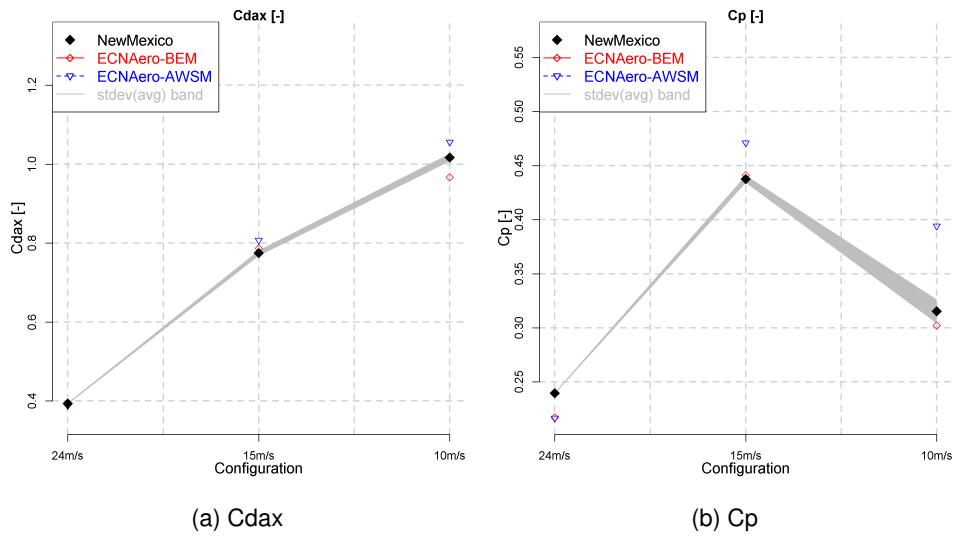


Figure 27: Radial distribution of New Mexico sectional forces for axial flow conditions, pitch=-2.3°, 425 rpm

Figure 28: New Mexico thrust and power coefficients, pitch= $-2.3^\circ$ , 425 rpm

#### 4.2.2 Yawed flow

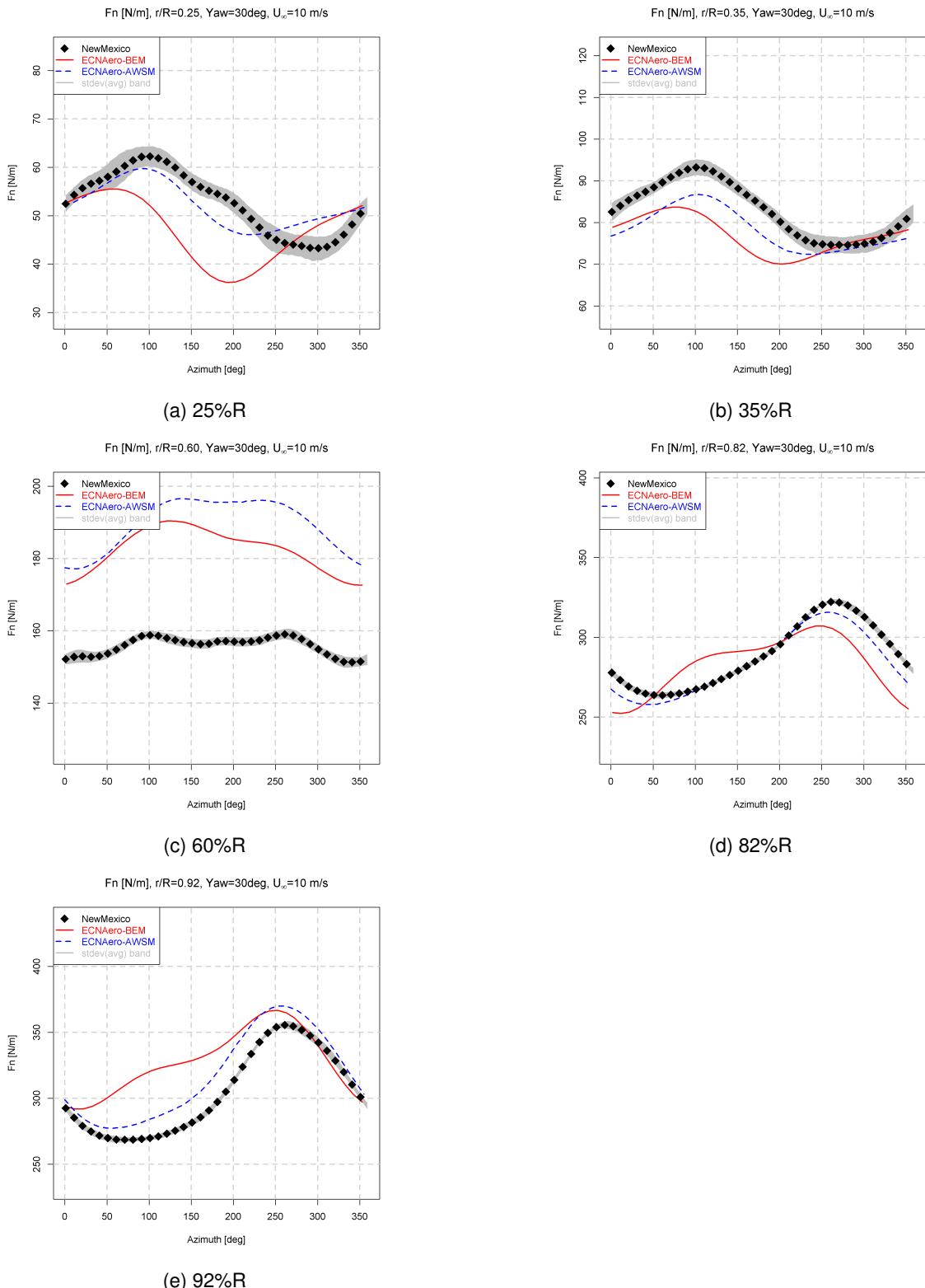


Figure 29: New Mexico normal force variation with rotor azimuth for yawed flow conditions ( $U_\infty=10 \text{ m/s}$ , 425 rpm, pitch= $-2.3^\circ$ , Yaw= $30^\circ$ )

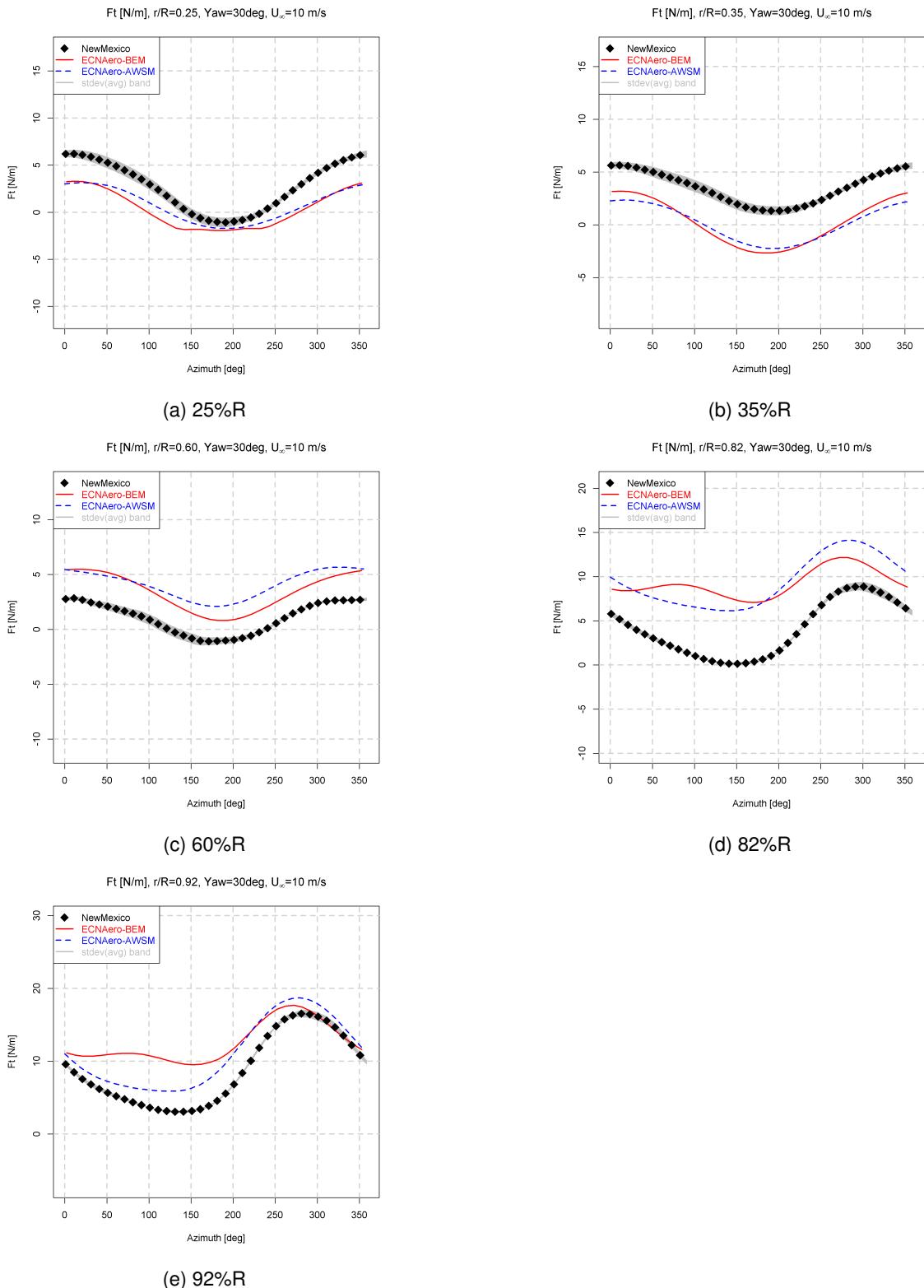


Figure 30: New Mexico tangential force variation with rotor azimuth for yawed flow conditions ( $U_\infty=10 \text{ m/s}$ , 425 rpm, pitch=-2.3°, Yaw=30°)

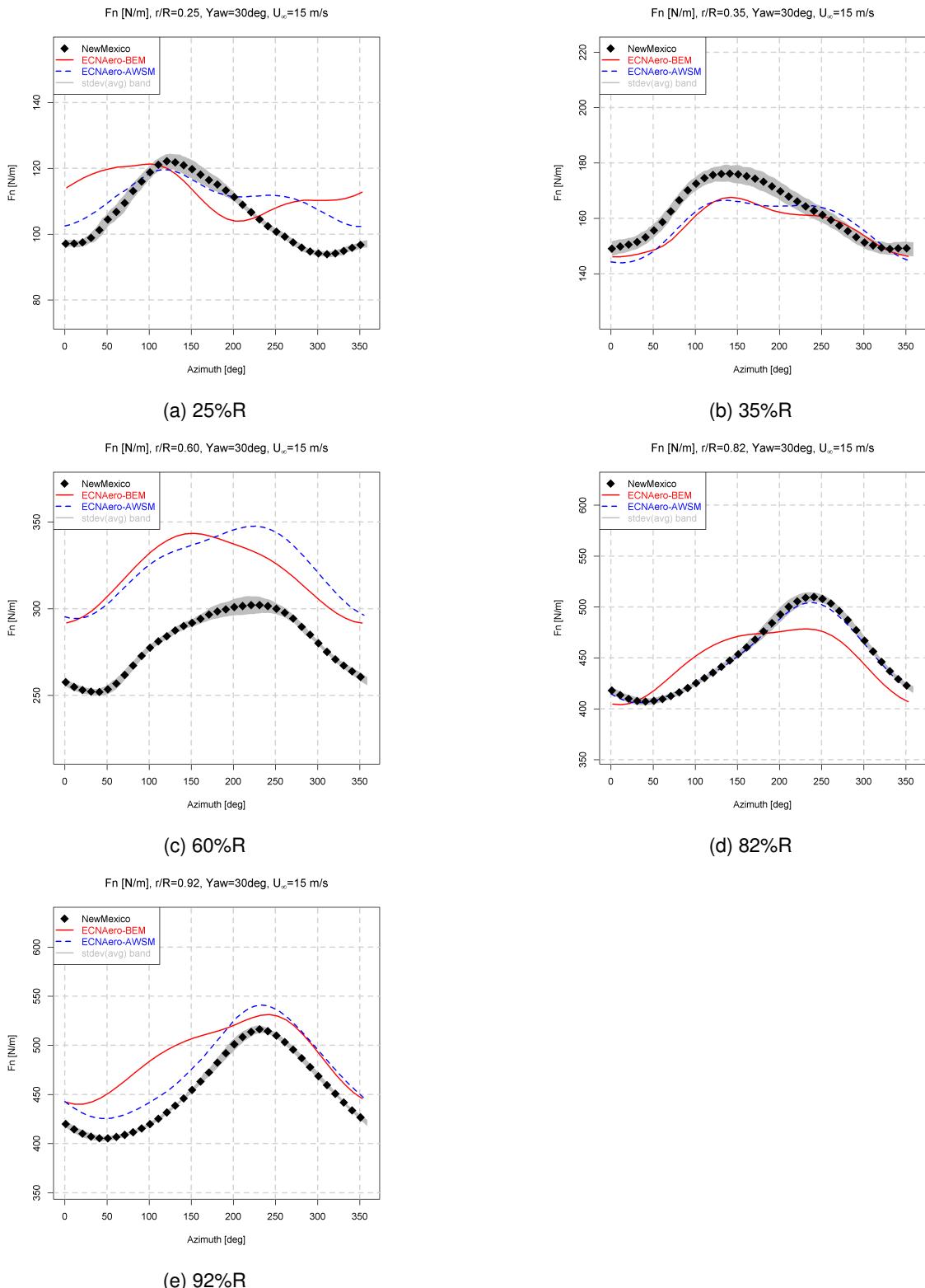


Figure 31: New Mexico normal force variation with rotor azimuth for yawed flow conditions ( $U_\infty=15$  m/s, 425 rpm, pitch= $-2.3^\circ$ , Yaw= $30^\circ$ )

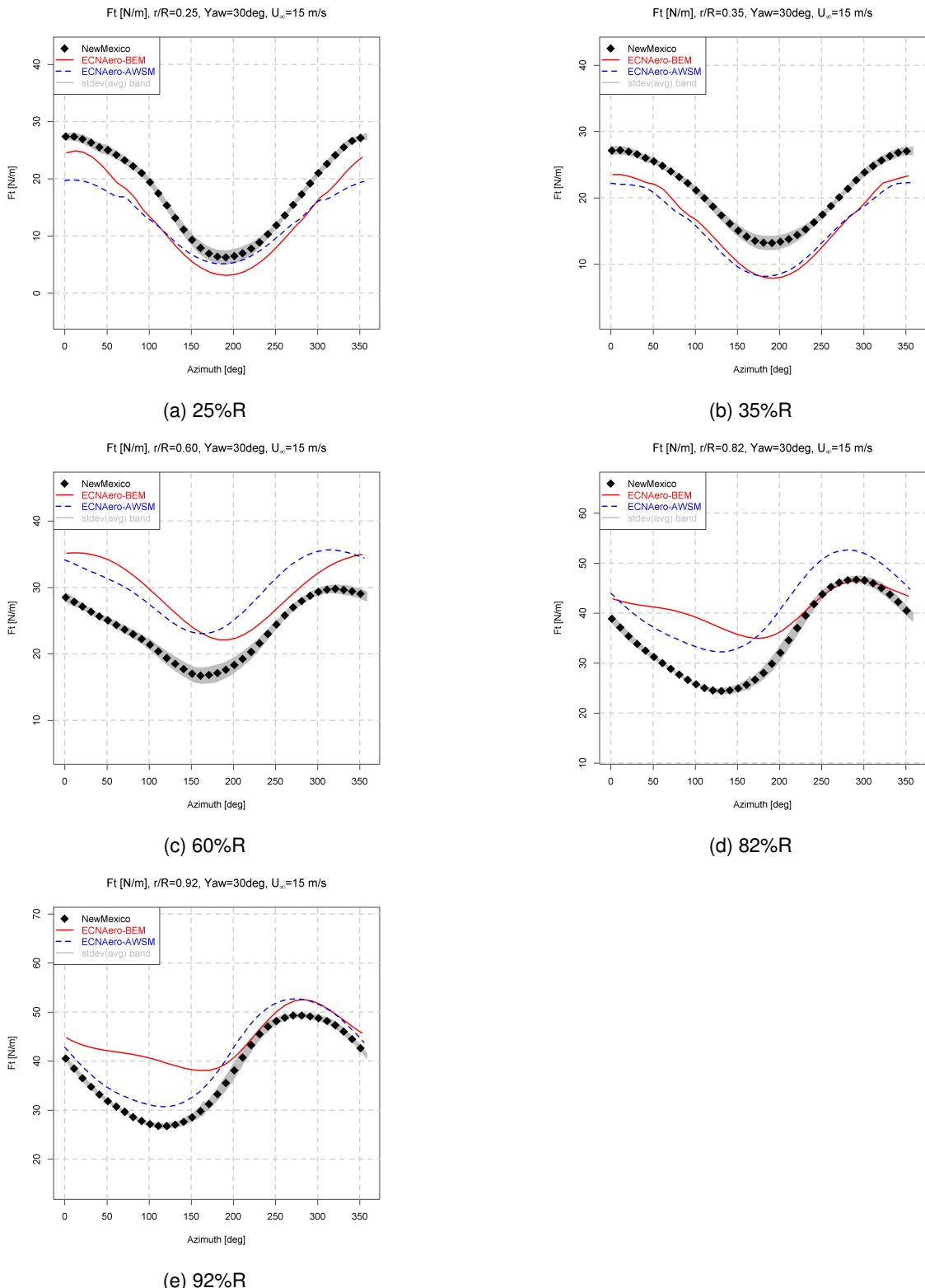


Figure 32: New Mexico tangential force variation with rotor azimuth for yawed flow conditions ( $U_\infty=15$  m/s, 425 rpm, pitch=-2.3°, Yaw=30°)

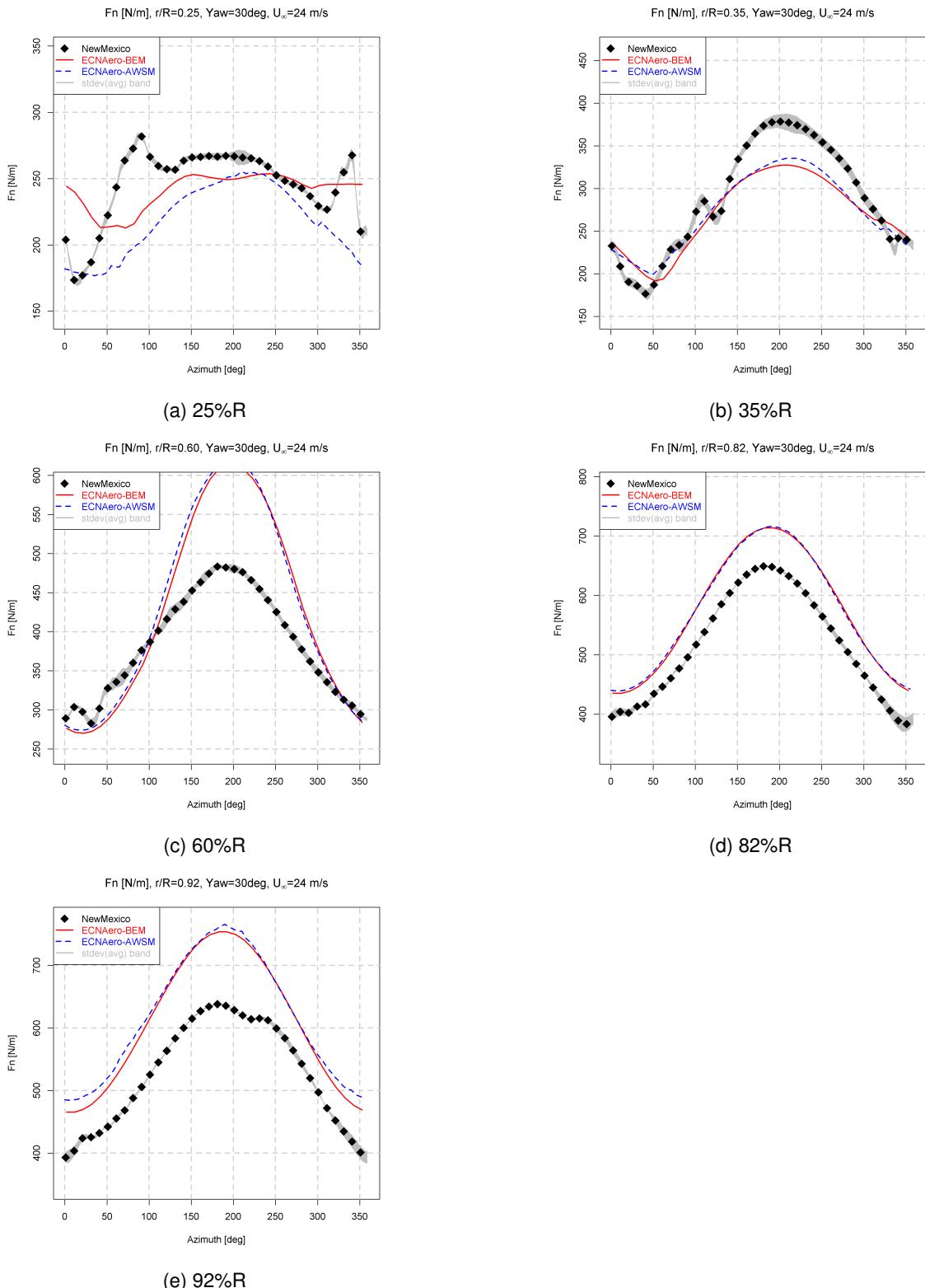


Figure 33: New Mexico normal force variation with rotor azimuth for yawed flow conditions ( $U_\infty=24$  m/s, 425 rpm, pitch= $-2.3^\circ$ , Yaw= $30^\circ$ )

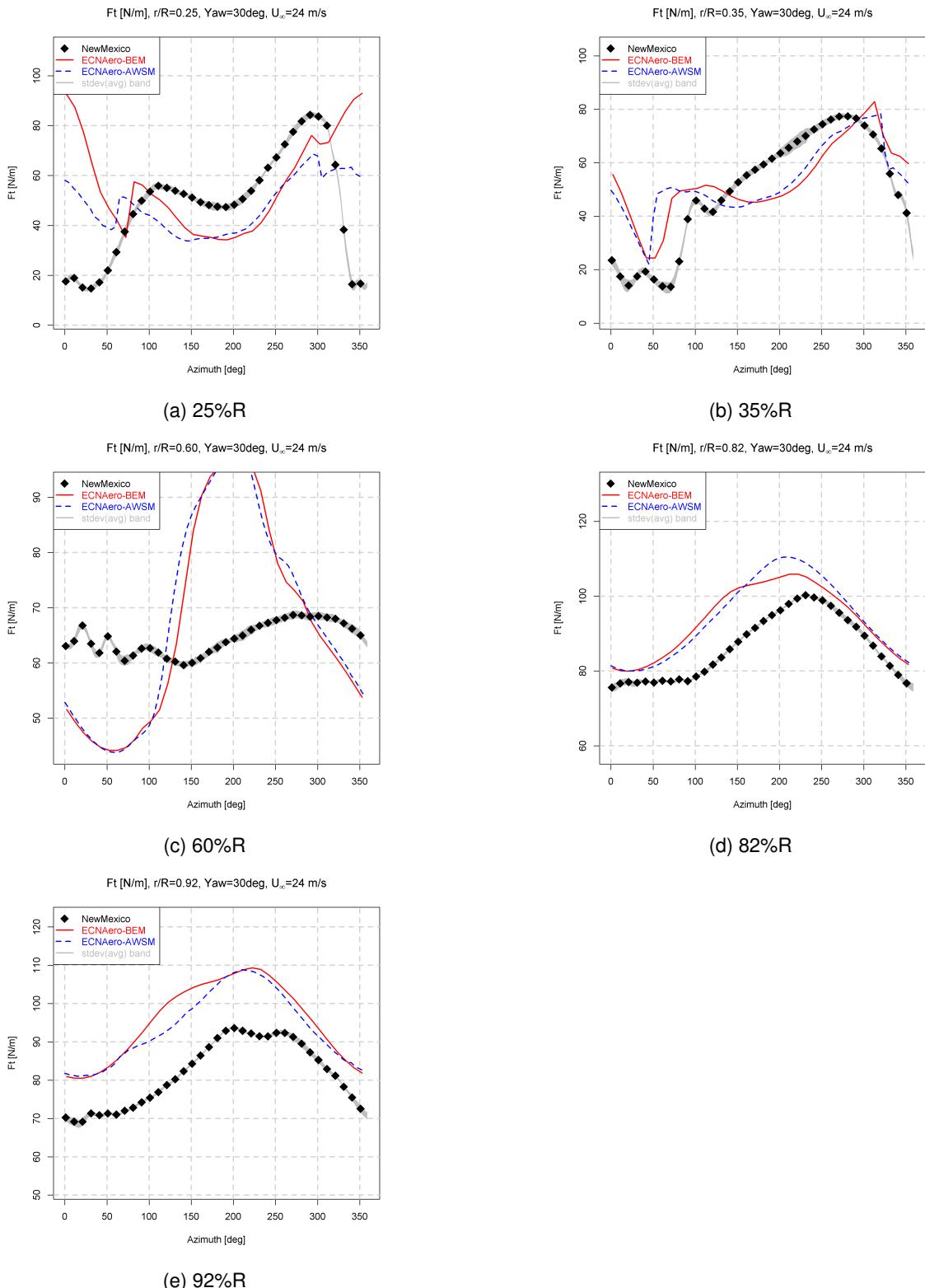


Figure 34: New Mexico tangential force variation with rotor azimuth for yawed flow conditions ( $U_\infty=24$  m/s, 425 rpm, pitch=-2.3°, Yaw=30°)

## Bibliography

- [1] *Germanischer Lloyd, Guideline for the Certification of Wind Turbines*, 2003 edition, 2003. with supplement 2004.
- [2] <http://www.hawc2.dk/download/pre-processing-tools>. 2018.
- [3] A. Björck. DYNSTALL: Subroutine package with a dynamic stall model. Technical Report FFAP-V-110, FFA, Sweden, 2000.
- [4] K. Boorsma, F. Grasso, and J.G. Holierhoek. Enhanced approach for simulation of rotor aerodynamic loads. Technical Report ECN-M-12-003, ECN, presented at EWEA Offshore 2011, Amsterdam, 29 November 2011 - 1 December 2011, 2011.
- [5] K. Boorsma, L. Greco, and G. Bedon. Rotor wake engineering models for aeroelastic applications. *Journal of Physics: Conference Series*, 1037(6):062013, 2018.
- [6] K. Boorsma and J.G. Schepers. New MEXICO Experiment, Preliminary overview with initial validation. Technical Report ECN-E-14-048, ECN, September 2014.
- [7] K. Boorsma and J.G. Schepers. In *Rotor experiments in controlled conditions continued: New Mexico*, Proceedings of the Science of Making Torque from Wind, Technical University of Munich, 2016.
- [8] E.T.G. Bot. Aerodynamic table generator. Technical Report ECN-C-01-077, ECN, 2001.
- [9] H. Glauert. A General Theory of the Autogyro. Technical Report ARC R&M No.1111, ARC R&M, 1926.
- [10] F. Grasso. AWSM external field calculations. Technical Report ECN-X-09-043, ECN, 2009.
- [11] F. Grasso. AWSM ground and wind shear effects in aerodynamic calculations. Technical Report ECN-E-10-016, ECN, 2010.
- [12] J.G. Holierhoek, J.B. De Vaal, A. H. Van Zuijlen, and H. Bijl. Comparing different dynamic stall models. *Wind Energy*, online 2012, 2012.
- [13] B.J. Jonkman and M.L. Buhl Jr. TurbSim Users Guide. Technical Report NREL/TP-500-39797, ECN, 2006.
- [14] M.A. Khan. Dynamic stall modeling for wind turbines. Master's thesis, TU-Delft, Aerospace Engineering, 2018.
- [15] J.G. Leishman and T.K. Beddoes. A generalized method for unsteady airfoil behaviour and dynamic stall using the indicial method. In *42nd Annual Forum*, Washington D.C., June 1986. American Helicopter Society.
- [16] J.G. Leishman and T.K. Beddoes. A semi-empirical model for dynamic stall. *Journal of the American Helicopter Society*, 34:3–17, 1989.
- [17] C. Lindenburg. Investigation into rotor blade aerodynamics. Technical Report ECN-C-03-025, ECN, 2003.
- [18] C. Lindenburg and J.G. Schepers. Phatas-IV aeroelastic modelling, release "dec-1999" and "nov-2000". Technical Report ECN-CX-00-027, ECN, 2000.
- [19] B.O.G. Montgomerie, A.J. Brand, J. Bosschers, and R.P.J.O.M Van Rooij. Three-dimensional effects in stall. Technical Report ECN-C-96-079, ECN, 1996.

- [20] H.A. Panofsky. *Atmospheric turbulence*. 1959.
- [21] L. Prandtl and A. Betz. *Vier Abhandlungen zur hydrodynamik und aerodynamik*. Göttingen, Germany, 1927.
- [22] R. Santos Pereira, J.G. Schepers and M. Pavel. "Validation of the Beddoes Leishman Dynamic Stall model for Horizontal Axis Wind Turbines using Mexico data". In *49th AIAA Aerospace Sciences meeting*, January 2011.
- [23] J.G. Schepers. An engineering model for yawed conditions, developed on basis of wind tunnel measurements. Technical Report AIAA-1999-0039, AIAA, 1999.
- [24] J.G. Schepers and Vermeer L.J. Een engineering model voor scheefstand op basis van windtunnelmetingen. Technical Report ECN-CX-98-070, ECN, 1998.
- [25] D. Simms, M. Hand, L.J. Fingersh, and D. Jager. Unsteady aerodynamics experiment phases II-IV test configurations and available data campaigns. Technical Report NREL/TP-500-25950, NREL, 1999.
- [26] H. Snel. Heuristic modelling of dynamic stall characteristics. In *Conference proceedings European Wind Energy Conference*, pages 429–433, Dublin, Ireland, October 1997.
- [27] H. Snel and J.G. Schepers. Joule1: Joint investigation of dynamic inflow effects and implementation of an engineering model. Technical Report ECN-C-94-107, ECN, 1994.
- [28] H. Spath. *Spline-Algorithmen zur Konstruktion glatter Kurven und Flächen*. Oldenbourg R. Verlag GmbH, 1973.
- [29] A. Van Garrel. Development of a wind turbine aerodynamics simulation module. Technical Report ECN-C-03-079, ECN, 2003.
- [30] D. Winkelaar. SWIFT: Program for three-dimensional wind simulation, part 1: Model description and program verification. Technical Report ECN-R-92-013, ECN, 1992.

# A Usage of AWSM for external field analysis

AWSM is able to perform calculations of induced velocity, but only in correspondence of points on the body. Often, it is beneficial to visualize the flow field at other locations. For this purpose it is required to perform calculations at specific points of the external domain. For wind turbines, it can be important to understand the effect of a complex field of motion, generated for example by several turbines at a prescribed location. Also, it can be necessary to compare numerical predictions with experimental data coming from a meteorological mast placed at a certain position.

## A.1 The input file `inextpoints.txt`

This file contains the information concerning the quantity and the location of the external field points where to perform calculations. In the present version of the module, three options are available to assign points:

- 1 Rectangular grid
- 2 Circular grid
- 3 User defined points

The number in the option description list indicates also the value to assign to the variable at the top of the input file. In the rest of the chapter, each option is described separately. It is also possible to use several distributions of points at the same time, just by indicating the amount of sets at the top of the `inextpoints.txt` file.

In addition to that it is possible to evaluate only trailing vortices for the induced velocity calculations instead of using the whole vortex lattice also including shed vorticity:

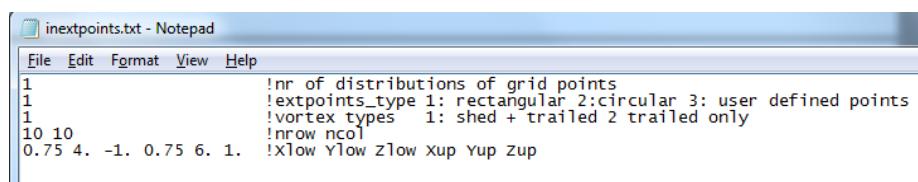
- 1 Shed and trailed vorticity contributions
- 2 Only trailed vorticity contributions

After this first information, data for specific sets of points must be provided, according to the description in the next subparagraphs.

### A.1.1 Rectangular grid

In order to assign a rectangular grid, the location of the lower-right and upper-left corners have to be prescribed and also the number of points along the edges. Table 5 summarizes the input data.

Here, an example of the input file structure is also provided:



```

inextpoints.txt - Notepad
File Edit Format View Help
1 !nr of distributions of grid points
1 !extpoints_type 1: rectangular 2:circular 3: user defined points
1 !vortex_types 1: shed + trailed 2 trailed only
10 10 !nrow ncol
0.75 4. -1. 0.75 6. 1. !xlow ylow zlow xup yup zup

```

Figure 35: Input file example for rectangular grid

Table 5: Input parameters for rectangular grid

Parameter	Type	Description
Xlow	Real	X location of the lower right corner [m]
Ylow	Real	Y location of the lower right corner [m]
Zlow	Real	Z location of the lower right corner [m]
Xup	Real	X location of the upper left corner [m]
Yup	Real	Y location of the upper left corner [m]
Zup	Real	Z location of the upper left corner [m]
nrow	Integer	Number of points along the vertical edge [-]
ncol	integer	Number of points along the horizontal edge [-]

### A.1.2 Circular grid

In order to assign a circular grid, the location of the centre of the circle and the radius have to be prescribed and also the number of points along the radial and angular directions. Table 6 summarizes the input data.

Table 6: Input parameters for circular grid

Parameter	Type	Description
XR	Real	X location of the centre [m]
YR	Real	Y location of the centre [m]
ZR	Real	Z location of the centre [m]
R	Real	Circle radius [m]
nr	Integer	Number of points along the radial direction [-]
nteta	Integer	Number of points along the azimuth direction [-]

Here, an example of the input file structure is also provided:

```

inextpoints.txt - Notepad
File Edit Format View Help
1 !nr of distributions of grid points
2 !extpoints_type 1: rectangular 2:circular 3: user defined points
1 !vortex_types 1: shed + trailed 2 trailed only
10 10 !nAzim nRadial
0 0 130 100 !xc yc zc R !!! xCentre yCentre zCentre RadiusCircle

```

Figure 36: Input file example for circular grid

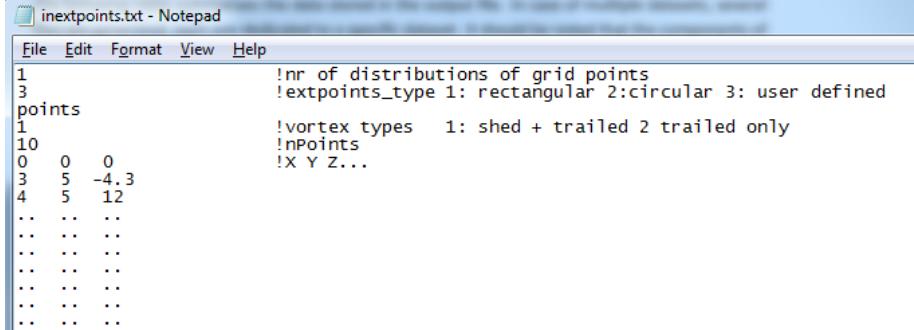
### A.1.3 User defined locations

In order to assign points in a user defined way, just the total number of points and their location have to be indicated. Table 7 summarizes the input data.

Here, an example of the input file structure is also provided:

Table 7: Input parameters for user defined locations

Parameter	Type	Description
X	Real	X location of the points [m]
Y	Real	Y location of the points [m]
Z	Real	Z location of the points [m]
n	Integer	Number of points [-]



```

inextpoints.txt - Notepad
File Edit Format View Help
1           !nr of distributions of grid points
3           !extpoints_type 1: rectangular 2:circular 3: user defined
points
1           !vortex types   1: shed + trailed 2 trailed only
10          !points
0   0   0
3   5   -4.3
4   5   12
...  ...
...  ...
...  ...
...  ...
...  ...
...  ...
...  ...
...  ...
...  ...

```

Figure 37: Input file example for user defined grid

## A.2 The output file extfield<>.dat

After the external field calculation has been performed, the obtained data are stored in the file `extfield<ID dataset>.dat`. This file is an ASCII file, but it is structured to be managed by using TECPLOT software. The following table summarizes the data stored in the output file. In case of multiple datasets, several files are generated, each one dedicated to a specific dataset. It should be noted that the components of velocity are representing the components of the actual **rotor induced** velocity at prescribed locations.

Table 8: Variables stored in the output file

Parameter	Type	Description
X	Real	X location of input points [m]
Y	Real	Y location of input points [m]
Z	Real	Z location of input points [m]
Cp	Real	Pressure coefficient [-]
U	Real	Velocity component [m/s]
V	Real	Velocity component [m/s]
W	Real	Velocity component [m/s]

# Index

AEROMODEL, 23  
AEROOUTPUTINCREMENT, 31  
AEROPROPS, 24  
AERORESULTSFILE, 30  
AEROROOT, 26  
AIRDENSTY, 28  
AOAEVAL, 27  
ATRANSITION, 30  
BLADELENGTH, 26  
BLADEROOT, 26  
BL\_A1, 28  
BL\_A2, 28  
BL\_ACD, 28  
BL\_B1, 28  
BL\_B2, 28  
BL\_KA, 28  
BL\_TF, 28  
BL\_TP, 28  
BL\_TVL, 28  
BL\_TV, 28  
CENTEREDCONTRPOINTS, 31  
CIRCCONVCRIT, 32  
COEFFFILENAME, 27  
CONEANGLE, 26  
CONVECFATOR, 31  
CORR3DTYPE, 27  
COUNTERMIN, 30  
DEBUGFILE, 23  
DYNINFLOW, 30  
DYNSTALLTYPE, 27  
DYNVISC, 28  
EXTFIELDFLAG, 32  
FREESTRMWISEWAKEPOINTS, 31  
FSMETHOD, 27  
GEOMOUTPUTINCREMENT, 31  
GEOMRESULTSFILE, 31  
GROUNDFLAG, 33  
GROUNLEVEL, 33  
HORSHEAR, 28  
HUBHEIGHT, 26  
INCLUDE, 24  
INDUCTIONDRAG, 30  
INTERPOL, 24  
LIFTCUTOFFRADIUS, 32  
LINREG, 27  
LOGFILENAME, 24  
LOGTRACESWITCH, 31  
MAXAOALIN, 27  
MINAOALIN, 27  
NEWRAPNNEIGHBOURS, 32  
NROFBEMELEMENTS, 30  
NROFBLADES, 26  
NUMITAEROCONV, 32  
NUMITEQMAX, 32  
ON\_A0, 28  
ON\_A2, 28  
ON\_E2, 28  
ON\_LAML, 28  
ON\_RO, 28  
ON\_R2, 28  
ON\_SIGL, 28  
ON\_SL, 28  
PARALLEL, 33  
PITCHANGLE, 26  
PRANDTLROOT, 30  
PRANDTLTIP, 30  
PRSCRBWAKE, 31  
RAMPFACTOR, 28  
RAMPTIME, 28  
RPM, 26  
SHEAREXP, 29  
SOS, 29  
STARTAZIMUTH, 26  
STARTSMOOTHINGS, 32  
STDOUTSWITCH, 31  
STREAMWISEWAKEPOINTS, 31  
SUBITERFLAG, 24  
TBEGIN, 26  
TEND, 26  
TILTANGLE, 26  
TIMEENDAERO, 30  
TIMEENDGEOM, 31  
TIMESTARTAERO, 30  
TIMESTARTGEOM, 31  
TIMESTEP, 26  
TOWERBASERADIUS, 29  
TOWERTOPRADIUS, 29  
TRACELEVEL, 31  
TURBINETYPE, 23  
VORTEXCUTOFF, 32  
WAKECUTOFFRADIUS, 32  
WAKEREDUCTIONSSKIP, 33  
WAKEREDUCTIONSTART, 33  
WINDFILENAME, 29  
WINDINTERPL, 29  
WINDTYPE, 29  
XNAC2HUB, 27  
YAWANGLE, 27  
YAWMODEL, 30  
Z0, 29  
ZNAC2HUB, 27