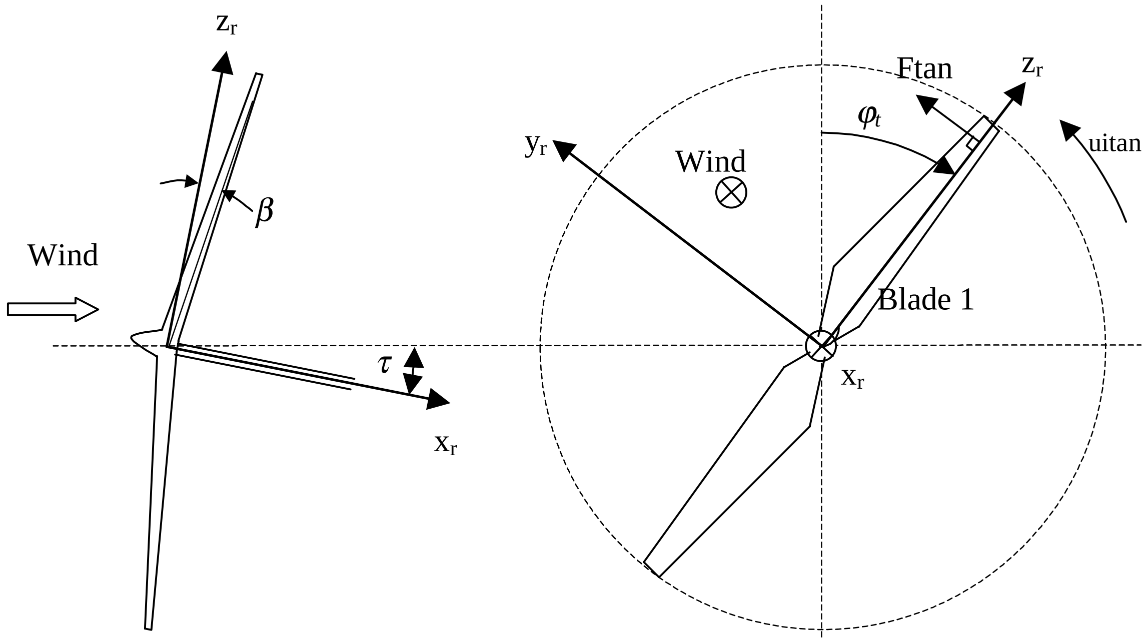
AERFORCE: Subroutine Package for unsteady Blade-Element/Momentum Calculations.

Anders Björck



View in the y_r-direction   View in the x_r-direction

FLYGTEKNISKA FÖRSÖKSANSTALTEN

THE AERONAUTICAL RESEARCH INSTITUTE OF SWEDEN

FLYGTEKNISKA
FÖRSÖKSANSTALTEN
THE AERONAUTICAL RESEARCH INSTITUTE OF SWEDEN
Box 11021, SE-161 11 Bromma, Sweden
Phone +46 8 634 10 00, Fax +46 8 25 34 81

Summary

A subroutine package, called AERFORCE, for the calculation of aerodynamic forces of wind turbine rotors has been written.

The subroutines are written in FORTRAN.

AERFORCE requires the input of airfoil aerodynamic data via tables as function of angle of attack, the turbine blade and rotor geometry and wind and blade velocities as input.

The method is intended for use in an aeroelstic code. Wind and blade velocities are given at a sequence of time steps and blade forces are returned.

The aerodynamic method is basically a Blade-Element/Momentum method. The method is fast and coded to be used in time simulations. In order to obtain a steady state solution a time simulation to steady state conditions has to be carried out.

The BEM-method in AERFORCE includes extensions for:

- Dynamic inflow: Unsteady modeling of the inflow for cases with unsteady blade loading or unsteady wind.
- Extensions to BEM-theory for inclined flow to the rotor disc (yaw model).
- Unsteady blade aerodynamics: The inclusion of 2D attached flow unsteady aerodynamics and a semi-empirical model for 2D dynamic stall.

Content

Symbols and notations

a	Induction factor in the direction normal to the rotor plane. $a = -u_i / V$. (Note the minus sign due to the definition of u_i being positive in the positive x_r-axis direction which perhaps not is the most usual convention.)
c	Local blade chord
C_l	Lift coefficient
C_d	Drag coefficient
C_m	Moment coefficient
C_T	Thrust coefficient. On a rotor level or on a an thrust for an annulus level.
F	Tip loss factor
$F_{\tan}$	Tangential force. Positive in the positive y_r-axis direction for blade No. 1.
IB	Index for blade number IB
IR	Index for radial element IR
NB	Number of Blades
q_z	Yaw angle. Rotation round the global *z*-axis
R	Rotor radius
r	Local blade radius
S	Rotor area. Also used on annulus level as the annulus area.
T	Thrust. Positive in the positive x_r-axis direction.
t	Time
dt	Incremental time step
U_w	Wake velocity. Representative velocity of the sum of free stream and induction in the rotor plane
u_i	Induction.
u	Induction used in section 3.2.4. Different in definition of positive direction relative to u_i .
V	Velocity. Most often referring to the free stream velocity

V_w	Free stream velocity. Mean free stream velocity over the rotor or some other weighted average of the free stream velocity in the rotor plane
v_X	x_r-component of the free wind speed made dimensionless with the absolute value of the free wind speed
v_{ip}	in-plane component of the free wind speed made dimensionless with the absolute value of the free wind speed
W	Velocity relative to local blade element.
α	Angle of attack
α_t	Teeter angle
β	Blade coning angle
χ	Wake skew angle. The angle between U_w and the x_r-axis.
φ_t	Azimuth angle of blade No 1 in a non-rotating *r*-system. Zero with the z_r-axis pointing upwards. (See Figure 1)
φ_{bl}	Rotation angle to blade position in the rotating *r*-system.
φ_χ	Azimuthal direction of the projection of U_w in the rotating *r*-system
λ	Tip speed ratio
Ω	Rotor rotational speed
θ	Local blade pitch angle.
ρ	Air density
τ	Time constant. Also tilt angle
2D	Two-dimensional
3D	Three- dimensional

Subscripts

d	In the rotor plane (disc plane)
b	Blade
ip	In-plane, meaning in the yz_r-plane
x	*x-direction*
y	*y-direction*

mom	Subscript to tell that it is a distance or area projected on the yz_r-plane
qs	"quasi steady" values
w	Quantity referring to the wake or for wake angle calculations
∞	Free stream conditions or conditions at infinity

Subscripts for coordinate systems

e	Index for the blade element coordinate system
g	Index for the global coordinate system
r	Index for the rotor coordinate system

Subscripts for induction

0	Azimuthal average of the induction. Average here meaning the average of the azimuthal induction variation due to yaw. The finite number of blades also causes an azimuthal variation of the induction but this variation is not considered here.
y	Azimuthal variation part of the induction
nor	Induction normal to the rotor plane. Positive in the positive x_r-axis direction
tan	Induction in the yz_r-plane. Positive in the positive y_r-axis direction for blade No. 1.
t	Same as index *tan.*
w	Used as index for normal rotor average normal induction.

1 Introduction

This report describes the subroutine package AERFORCE.
The subroutine package contains FORTRAN subroutines for the calculation of aerodynamic loads for Horizontal Axis Wind Turbines, HAWTs.

1.1 Aerodynamic methods

Short overview
Various methods can be used in order to calculate the aerodynamic forces acting on the blades of a wind turbine. A Review of different methods can be found in [1]. The most advanced being numerical methods to solve Navier Stokes equations for the global flow as well as the flow near the blades. The Navier Stokes equations neglecting the effects of viscosity are called the Euler equations. Various methods can be found in the literature where the Euler equation approximation is used to solve the global flow field. These methods use different approaches such as vortex particle methods or vortex filament methods.

The simplest, but also the fastest method is to use a momentum method. The blade forces, considered as an actuator disc, are then balanced with the global momentum change far down-stream and far upstream of the rotor.

The BEM method
The momentum method can be applied to an annular section stream tube around the rotor main axis, combined with blade forces obtained from aerodynamic tables as function of local inflow quantities. This method is called the Blade-Element/Momentum method (BEM) and is by far the most widely used method to calculate the aerodynamic forces on wind turbine blades. The BEM method as described by Wilson and Lissaman [2] forms the basis for most modern BEM design tools.

For the use in aeroelastic codes in design calculations, the aerodynamic method has to be very time efficient. The BEM-method (in different versions) is therefore the most commonly used method in aeroelastic codes. The evolution of computer capacity and numerical methods have recently led to free wake vortex methods or other solutions of the Euler equations being used even in aeroelastic codes for time simulation of wind turbine forces [3], [4], [5]. Still a BEM-method will be substantially faster and will likely remain as the method for the bulk of aeroelastic calculations for many years.

The BEM method as described by e.g. Wilson and Lissaman [2] is a steady method even though the method often is used in time simulations in a quasi-steady manner.

Extensions to BEM theory have, however, been presented during the past ten to fifteen years. These extensions cover the inclusions of models for unsteady variations of blade loads (“dynamic inflow”), and models for unsteady blade aerodynamics.

Further extensions to the BEM method of [2] is the modeling of inclined flow to the rotor disc (yaw and tilt).

1.1.1 The method used in the AERFORCE subroutine package

The method in AERFORCE is a BEM type method.

The method is fast and coded to be used in time simulations. In order to obtain a steady state solution a time simulation to steady state conditions has to be carried out.

The BEM-method in AERFORCE includes extensions for:

- Dynamic inflow: Unsteady modeling of the inflow for cases with unsteady blade loading or unsteady wind.
- Extensions to BEM-theory for inclined flow to the rotor disc (yaw model).

- Unsteady blade aerodynamics: The inclusion of 2D attached flow unsteady aerodynamics and a semi-empirical model for 2D dynamic stall.

The method can be characterized as a fast aerodynamic code for time simulation that includes effects of yawed flow, unsteady blade loading and unsteady airfoil aerodynamics (dynamic stall effects).

1.2 Structure of the report

The coordinate systems are described in chapter 2.

The aerodynamic model is described in chapter 3. The implemented BEM-method is described in this chapter.

Chapter 3 constitutes a description of the implemented aerodynamic methods. For a detailed description of the dynamic stall method, however, reference is made to [12]. The description assumes that the reader has basic knowledge of aerodynamics applied to wind turbines. Such knowledge can e.g. be obtained from the text-book by Freris [9] or from [2].

Details in the use of the subroutine package is to a large extent given in the source code but further guide lines needed for the implementation of the subroutine package in a calling program is given in chapter 4.

2 Coordinate systems

Forces and velocities in AERFORCE are described in three coordinate systems.

1. The global system: The system in which the free wind is given. The system is designated the *g*-system
2. The rotor system: A system that is attached to the rotor. The system is rotating with the rotor and has its *y* and *z*-axis in the disc-plane. In this system the normal induction is in the *x*-axis direction and the tangential induced velocity is an in-plane velocity. The system is designated the *r*-system.
3. The blade-element system: A coordinate system that is attached to each blade element. The system is designated the *e*-system.

2.1 Transformation matrices

A vector $\vec{r}$ given in the coordinate system, *a*, can be written in another coordinate system, *b*, by a transformation from the *a*-system to the *b*-system.

This is written as:

$$\vec{r}_b = S_{ab}\vec{r}_a$$

where S_{ab} is the transformation matrix of a vector from the *a*-system to the *b*-system. Going from one system to another, is obtained by a translation and a number of rotations. The latter is described by a rotation matrix.

In AERFORCE two rotation (transformation) matrices are needed:

- Transformation from the global system to the *r*-system
- Transformation from the r-system to the *e*-system

For the calculations in AERFORCE, only the rotation matrices are needed.

2.2 A typical turbine used to exemplify coordinate systems.

In order to exemplify, the coordinate systems are described using a turbine where the following rotations are used:

- Yaw angle: q_z.
- Tilt angle: τ
- Main shaft position angle: φ_t.
- Teeter angle: α_t
- Coning angle: β
- Local pitch angle: θ

  θ is the sum of collective blade pitch and blade twist.

2.3 Global system

Attached to e.g. the tower root. This system is referred to as the *g*-system

2.4 Rotating rotor system, r-system

The rotor system is a system that has its y and z-axis in the disc-plane. The system is rotating with the rotor. The *x*-axis is perpendicular to the rotor disc. For zero turbine yaw, tilt and teeter, the *x*-axis of the *r*-system coincides with the global *x*-axis.

Figure 1 below shows the system (for a turbine with zero teeter).

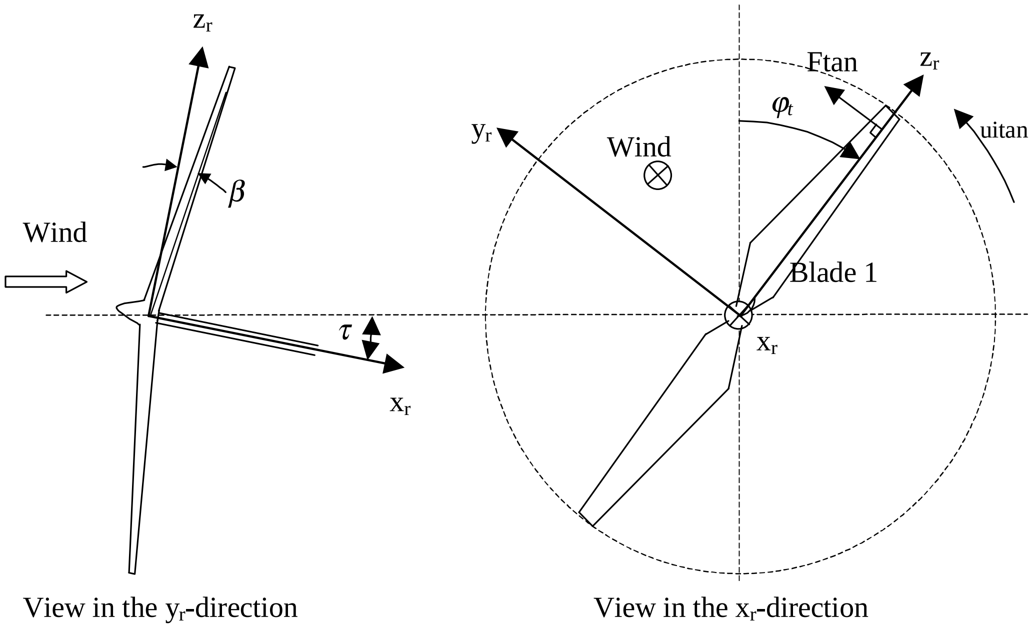


Figure 1. The r coordinate system

To go from the global system to the *r*-system an intermediate non-rotating system is introduced. This system is called the *n*-system.

To go from the *g*-system to the *n*-system:

i) A yaw angle rotation, q_z, around the z_g-axis and

ii) then a tilt angle τ around the new intermediate *y*-axis.

The x_n-axis is parallel to the rotor-axis.

From the *n*-system to the *r*-system:

i) Rotation φ_t around the x_n-axis.

ii) Then, a teeter-rotation, α_t, around the teeter-axis.

If the δ_3-angle is zero, the teeter rotation is around the intermediate y-axis, which then also will be the y_r-axis.

The r-system has its center at the blade intersection at the hub.

The rotor thrust and normal induction used in momentum theory is perpendicular to the rotor-system. Tangential induction lies in the *y*,*z*-plane of the *r*-system.

The transformation matrix S_{gr} is a rotor quantity, meaning that only one matrix is required.

2.5 Blade-element system, e-system

The blade-element system has its *y*-axis aligned with the local blade chord axis. The *z*-axis is aligned with the blade 25% chord axis. Figure 2 shows the *r*-and *e*-systems.

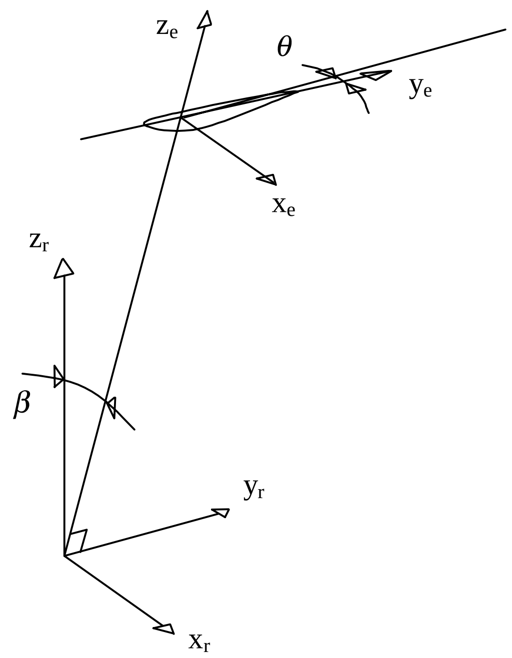


Figure 2. The e coordinate system

For transformation from the global or the rotating system one matrix is needed for each blade element (blade IB and radius IR).

For the case of zero coning, zero θ and no elastic deflection, the *r*-system and *e*-system of blade #1 coincide.

2.5.1 A note on the implementation of transformations in version AERFORCE 2b

In the current version of AERFORCE (AERFORCE_2b) the transformation matrix S_{re} from the r to the e-system is not used. Instead the sum of the pitch angle and the local twist is used together with the blade coning angle as input to the subroutine. This is a simplification made to reduce computation time in the calling program VIDYN. S_{re} is set in AERFORCE as:

```
sre(1,1) =  cpt*cb
sre(1,2) =  cpt*sb*sfbl - spt*cfbl
sre(1,3) = -cpt*sb*cfbl - spt*sfbl

sre(2,1) =  spt*cb
sre(2,2) =  spt*sb*sfbl+ cpt*cfbl
sre(2,3) = -spt*sb*cfbl+ cpt*sfbl
```

where

```
cpt=cos(pitch + twist)
spt=sin(pitch + twist)
cb=cos(beta)
sb=sin(beta)
cfbl=cos(fibl)
sfbl=sin(fibl)
```

fibl is used to go to blade # IB and is calculated in AERFORCE as fibl(ib)=real(ib-1)/real(nb)*2*pi

The third row in the S_{re}-matrix is not used and therefore not set.

The directions of pitch and twist are shown in Figure 2 for the pitch angle, θ .

3 Calculation of aerodynamic forces

In the simple momentum theory for an actuator disc, the flow conditions and aerodynamic loads are assumed constant in the azimuthal and radial direction. Furthermore this theory is only valid for lightly loaded discs with an infinite number of blades. Still the momentum theory in conjunction with blade-element theory is widely used for a large range of cases. “Fixes” are added for heavy loading and finite number of blades. Furthermore the theory is often used on annulus level or even for each blade element separately, assuming independence of the conditions at different stream tubes.

For use in the area of the helicopter field, versions of the momentum theory is used for large forward speeds, corresponding to large yaw angles for wind turbines. Extensions to yaw angles can thus be found in the helicopter literature.

Blade-Element Momentum theory and induction

Blade-Element Momentum theory, BEM, is based on the calculation of the reaction forces from the blades in terms of a momentum change. This is formulated such that the blade forces cause “induced velocities” in the disc plane, in this report often referred to as the “induction”. The induction is a function of the blade loads. It is this last part, the coupling of blade loads to the induction, that is the tricky thing. This has led to the development of a number of different approaches.

The method described in this report, and implemented in the subroutines of AERFORCE, is one such approach.

Local blade loads are calculated by two-dimensional blade-element theory using look-up tables for lift and drag coefficients as function of local blade incidence. The blade element lift force is e.g. calculated as:

$$dL = 0.5\rho \cdot c \cdot dr \cdot W^2 \cdot C_l \tag{3.1}$$

The local wind, W is here is composed of the vector sum of the free wind, blade motion and the induction, so that the blade loads can be calculated once the induction is known.

The induced velocities are composed of the normal induced velocity and the tangential induced velocity.

$$\bar{u}_i = \bar{u}_{i,nor} + \bar{u}_{i,\tan} \tag{3.2}$$

The normal induced velocity is the reaction to the thrust-force of the rotor and the tangential induced velocity is the reaction to the extracted or produced torque.

In this report, the normal induced velocity is perpendicular to the rotor-plane and the tangential induced velocity is an in-plane velocity with no radial component.

$\bar{u}_{i,nor}$ is positive in the $-x_r$-direction. $\bar{u}_{i,\tan}$ is positive in the $-y_r$-direction for blade 1 (see Figure 1)

Extension to yawed flow

One method used for yawed flow can be traced back to the development of Glauert in 1926 (see e.g. [6] or [8] page 124). This method assumes a constant normal induction plus an azimuthal harmonic variation. The azimuthal variation is dependent on the angle the wake makes with the rotor plane, the skew angle χ,(see figure 3) and the radial position. The normal induced velocity is given by:

$$u_{i,nor} = u_{i0} \cdot (1 + \frac{r}{R} \cdot \tan\left(\frac{\chi}{2}\right)\sin(\varphi)) \tag{3.3}$$

where φ is the azimuth angle of point where the induction is calculated.

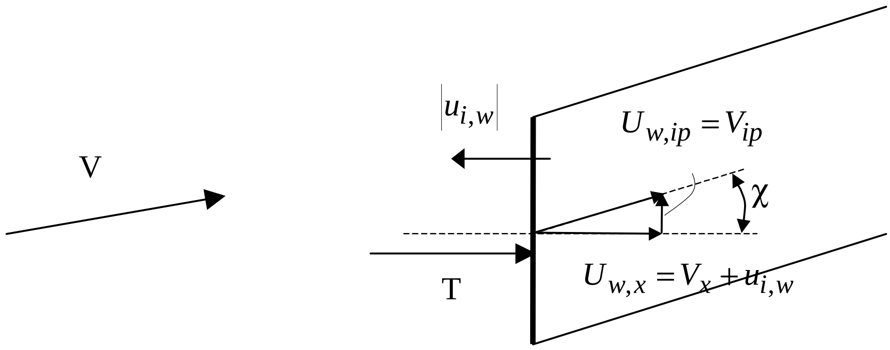


Figure 3. Velocities in the rotor-coordinate system and the direction of thrust[1]

Theory with radially and azimuthally independent induction

The BEM theory also has been widely used as described by Wilson and Lissaman e.g. in [2]. Normally, implementations of the theory, e.g. in VIDYN 7 [7], treat the loads as completely local and therefore with induction totally freely varying radially and azimuthally.

Extension to dynamic inflow

Extensions to the BEM theory to account for dynamic inflow and yawed flow also must be married with fixes for operation in the vortex brake stage and for fixes due to the number of blades not being infinite.

The method presented in the present report follows in large the work presented in the report of the EC project “Joint Investigation of Dynamic Inflow Effects and Implementation of an Engineering Method” [6]. More specifically it follows closely the method suggested by Stig Øye which in turn for yawed flow uses the formulation of Glauert.

[1] u_{i0} *is the same as* u_{iw} *in the figure.* u_{i0} *and* u_{iw} *are in AERFORCE defined positive in the positive* x_r*-direction. For positive T,* u_{i0} *will, however, be negative and the vector arrow for* u_{iw} *is therefore shown pointing to the left in the figure.*

Local induction or not

Problems in the application of the theory arise when the wind or the loading is varying over the rotor.

For the normal induced velocity in the case of yawed flow, equation (3.3) is used. In equation (3.3) it is normally assumed that u_{i0} (and the wake direction χ) is constant for the whole rotor. It is, however, not difficult to imagine that the induction can vary radially or that there should be an azimuthal variation even though the yaw angle is zero. E.g. in the case of zero yaw angle but with a strong wind gradient, or for the case of cyclic pitching. This will result in different local induced velocities, and a constant u_{i0} alone can not represent this case well. The application of the BEM theory as applied in version 7 of VIDYN, [7], will predict a variation of the induced velocity in these cases. On the other hand, for the yawed case in uniform wind, the loads and the induction will be the same when the blades are horizontally pointing upwind or downwind using the theory as applied in [7]. This will result in a zero yawing moment (around the hub), whereas in reality the induction has an azimuthal variation giving a restoring yaw-moment. Different ways in applying the BEM theory thus have pros and cons for different situations.

The methods described in the present report are a mix of treating the induction on a rotor-level, on an annulus level and independently for each blade. Details in implementing the method from [6], to a large extent, concerns how to treat the induction in terms of being radially or not radially varying and azimuthally or not azimuthally varying.

The actual implementation in AERFORCE is discussed in subsequent sections.

3.1 Calculation of the skew-angle.

The skew angle is treated as constant for all radial positions. A representative free stream velocity, $\vec{V}_w$, and a representative axial induced velocity, $\bar{u}_{i,w}$, is used to compute the wake direction, χ. The purpose is to average the spatial variation of the wind to get one representative wind for the wake angle calculation.

The representative free stream velocity, $\vec{V}_w$, is given through input at each time step. $\vec{V}_w$ could be taken as the mean velocity over the rotor or as some other weighted average.

The two cases of a yawed or a tilted rotor and any mix of these are treated in equal ways. This is done by representing the wake direction by a skew angle, χ, in combination with a projection onto the rotor plane of an azimuthal direction of the wake, φ_χ, see Fig. 4.

To compute χ, the wind $\vec{V}_w$ and $\bar{u}_{i,w}$ are added as vectors in the global system to get the wake velocity, $\vec{U}_w$:

$$\vec{U}_w = \vec{V}_w + \bar{u}_{i,w} \tag{3.1.1}$$

$\bar{u}_{i,w}$ is the disc averaged normal induction transformed to the global system[2].

Next $\vec{U}_w$ is transformed to the rotor system and the skew angle is calculated from:

$$\tan(\chi) = \frac{U_{w,ip}}{\left|U_{w,x}\right|} \tag{3.1.2}$$

[2] In version 2b of AERFORCE $\bar{u}_{i,w}$ is taken as the average of u_{i0} (multiplied with the tip loss factor) from r/R=60, 70 and 80% with weight factors 20%, 40% and 40%.

where

$$U_{w,ip}^2 = U_{w,y}^2 + U_{w,z}^2 \tag{3.1.3}$$

χ is always positive and assumes values between 0° and 90°.

$$\tan(\varphi_\chi) = \frac{-U_{w,y}}{U_{w,z}} \tag{3.1.4}$$

where φ_χ is given in the interval $[-\pi, \pi]$.

Figure 4 shows the azimuthal direction of the wake, φ_χ.

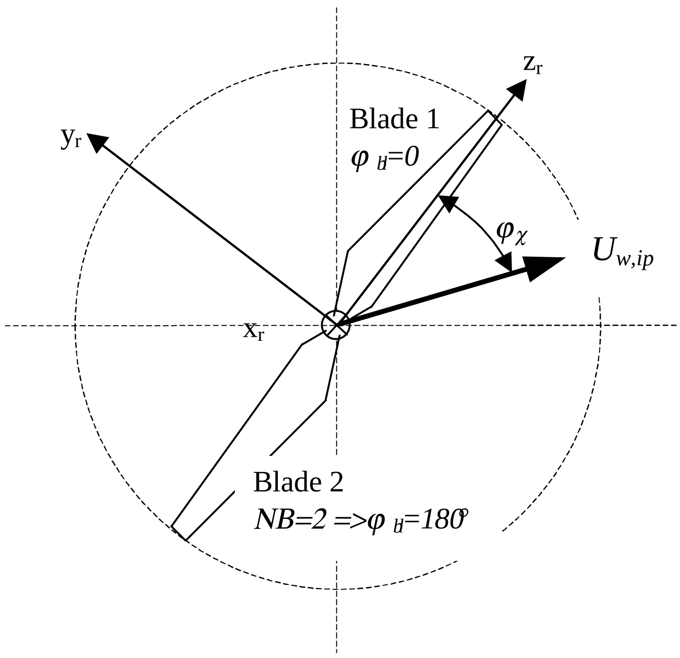


Figure 4. Wake direction in the rotor plane

The wake directions, χ, φ_χ and the quantities needed to compute the direction, $\vec{U}_w$, $\vec{V}_w$ and $\vec{u}_{i,w}$ are all quantities which are constant on a rotor-level meaning no radial or azimuthal variation.

In order to account for possible dynamics of the wake direction, a time lag filter is applied on $\vec{U}_w$ as suggested in the method of Øye [6, page 253].

In AERFORCE, χ, φ_χ are calculated by equations (3.1.1)-(3.1.4) in the following way:

1. Take $\vec{V}_w$ at the current time step defined in the global coordinate system.
2. Take $\vec{u}_{i,w}$ from the previous times step defined in the global coordinate system.
3. Add as vectors to get $\vec{U}_w$ as in equation (3.1.1)
4. Apply a simple 1st order filter with a time-constant T_w to $\vec{U}_w$. A value of $T_w = R/V_w$ is used.
5. Transform the lagged value of $\vec{U}_w$ to the rotor-system
6. Use equations (3.1.2)-(3.1.4) to get χ.

The above equations will work for flow to the turbine from all directions and it will work for dynamic response due to either a wind direction or due to turbine yawing motion.

3.2 Calculations of the induction

The induction is seen as an average value plus an azimuthal variation due to yaw.

For the normal induction this is written as:

$$u_{inor} = u_{i0} + u_{i,y} \tag{3.2.1}$$

and for the tangential induction as:

$$u_{i\tan} = u_{it0} + u_{it,y} \tag{3.2.2}$$

The last part of (3.2.1) and (3.2.2) are local values of radius and azimuth. The first part could be treated as a disc average but is in

AERFORCE an annulus quantity and calculated for each radial element[3].

The azimuthal average induction, u_{i0} and u_{it0}, are computed by quasi-steady BEM theory balancing blade forces with flow momentum forces as described in section 3.2.3. Dynamic values of u_{i0} and u_{it0} are then calculated through the two lag-equations of the Øye -model to mimic the fact that the wake and induced velocities do not reach the quasi-steady values for quick changes in loading conditions.

The sequence of calculations is:

1) Calculate the skew angle, χ, and the wake direction, φ_χ, in the wake as described in paragraph 3.1 based on the wind at the current time step and induction from previous time step

2) Use the azimuthal average induction , u_{i0} and u_{it0}, from the previous time step, χ and φ_χ from 1) and calculate the induction as the sum of the average and the azimuthal variation (described below)

3) Calculate blade loads combining the wind at the current time step and the induction from 2)

4) Use these blade loads to calculate new quasi steady values of the induction by quasi steady blade element momentum theory.

5) Calculate dynamic values of the induction, using the Øye method.

6) Save values of the induction for next time step

[3] The average induction u_{i0} refers to the azimuth average induction neglecting the tip loss factor. The average induction used in the momentum balance will be u_{i0} multiplied with the Prandtl tip loss factor.

3.2.1 Azimuthal average induction

The azimuthal average induction is seen as an annulus average. This average is calculated using blade element momentum theory as described in sections 3.2.3-3.2.5. The blade load at a particular radius used for these calculations is the sum of the blade loads from the NB loads at the same radii. The so derived average induction is designated $u_{i0,a}$ and $u_{it0,a}$ where the index a means that it is an annulus average. The induction used for the calculation of blade loads is this average plus the azimuthal variation.

There are, however, alternative ways of looking at the average induction u_{i0} and u_{it0}. Another way would be to use local values of u_{i0} and u_{it0}. The induction at blade *IB* at radius *IR* would then be composed of the local values of $u_{i0}(IB, IR)$ and $u_{it0}(IB, IR)$ plus the azimuthal variation.

The version, where $u_{i0,a}$ and $u_{it0,a}$ are derived from an average annulus load, is used in the current version 2b of AERFORCE.

Index a in $u_{i0,a}$ and $u_{it0,a}$ is in the subsequent sections skipped and u_{i0} and u_{it0} refer to the annulus average value obtained from the mean of the *NB* blade loads.

3.2.2 Variation of the induced velocity due to yaw

In order to compute the loads at the current time step, values for u_{i0} and u_{it0} must be found.

This is done by using u_{i0} and u_{it0} values from the previous time step.

The azimuthal variation is calculated as:

$$u_{i,y} = u_{i0} \cdot f(\tfrac{r}{R}) \tan\left(\frac{\chi}{2}\right)\cos\left(\varphi_{bl} - \varphi_{\chi}\right) \tag{3.2.2.1}$$

$$u_{it,y} = u_{it0} \cdot f(\tfrac{r}{R}) \tan\left(\frac{\chi}{2}\right) \cos\left(\varphi_{bl} - \varphi_{\chi}\right) \tag{3.2.2.2}$$

$f(\tfrac{r}{R})$ is taken from [6] as $r/R \cdot F$ where F is the Prandtl tip loss factor. F is included since the average induction corrected for NB blades is $u_{i0} \cdot F$.

φ_{bl} is the rotation angle of the blade around the x-axis in the rotor coordinate system. If the coordinate system e.g. is attached to blade 1 and the blade are assumed stiff, the angles will be $\varphi_{bl} = \frac{(IB-1)}{NB} 2\pi$, so that in this case for a three bladed rotor, φ_{bl} =0 for blade 1, φ_{bl} =120 for blade 2 and φ_{bl} =240 for blade 3.

3.2.3 Blade loads and Quasi-steady values of the induction

In order to calculate the induction, BEM-theory is used. The quasi-steady BEM-theory in the normal direction, neglecting the tip loss factor, can be written:

$$\rho S \left| \vec{V}_\infty + \vec{u}_{i0} \right| \cdot 2\vec{u}_{i0} = -T \tag{3.2.3.1}$$

Here $\vec{u}_{i0}$ and T are normal to the rotor-plane. T is the thrust force calculated from the blade loads, defined positive in the x_r -direction. T is in itself a function of the induction so that an iterative scheme is normally needed to solve (3.2.3.1).

For yawed flow the rotor area projected on the x_r -axis, S, is used as suggested by Glauert (see e.g. descriptions in [6] or [8]). The momentum equation is thus solved for a representative mass flow and not the mass flow taken as the vector product of the local flow vector and the normal to the rotor plane. Equation (3.2.3.1) is valid for the two limiting cases of 0 and 90° yaw angle where 90° yaw angle represents the case of a lightly loaded disc. Glauert further assumed that the same expression also holds for yaw angles between

0 and 90° yaw and equation (3.2.3.1) is used for all yaw angles in AERFORCE

3.2.3.1 Pure axial flow

For pure axial flow in simple actuator disc theory, equation 2.2.3.1 can be written as function of the induction factor and becomes the well known equation

$$4a|1-a| = C_T \tag{3.2.3.1.1}$$

where the thrust coefficient

$$C_T = T/(0.5 \cdot \rho \cdot S \cdot V^2) \tag{3.2.3.1.2}$$

a is the induction factor. With u_i positive in the x-direction and $V_\infty = V = V_x$, $a = -u_i/V$.

For a rotor with a finite number of blades, both axial flow and yawed flow C_T is calculated from the thrust divided by the tip loss factor to account for the effect of a finite number of blades (see section 3.2.3.3).

Equation (3.2.3.1.1) can lead to a singularity at a=1. Further, with an induction factor, a, above ≈ 0.3 the solution does not agree well with measurements. An empirical curve is therefore often applied. Several such suggestions can be found. Anderson (see [9] page 85) suggests a linear extension above a value $a = a_{t1} = 0.32$.

$$C_T = c_0 + c_1 \cdot a \tag{3.2.3.1.3}$$

with

$$c_1 = 4 - 8 \cdot a_{t1} \tag{3.2.3.1.4}$$

$$c_0 = 4 \cdot a_{t1} - 4 \cdot a_{t1}^2 - c_1 \cdot a_{t1} \tag{3.2.3.1.5}$$

This linear extension does not necessarily need to extend to $a=\infty$. The solution at $a=\infty$ e.g. represents the hover case for a helicopter.

Another way of plotting equation (3.2.3.1.1) is to plot it as dimensionless induction, $\bar{v} = \frac{-u_i}{\sqrt{T/2\rho S}}$ versus dimensionless wind speed $\bar{V} = \frac{-V}{\sqrt{T/2\rho S}}$ as often done in helicopter literature. (Note that the sign convention for the thrust often is different in helicopter literature and that the sign convention for u_i often is different in wind energy literature from that of this report). In such a diagram the singularity at a=1 of equation (3.2.3.1.1) becomes the asymptote $\bar{v} = -\bar{V}$.

In AERFORCE the empirical linear empirical curve of Anderson is used except for $a > a_{t2}$, where the momentum equation of (3.2.3.1.1) is used. The value a_{t1} is an input value to AERFORCE. a_{t2} is the value of *a* where the empirical curve according to equation (3.2.3.1.3) crosses the original momentum equation (3.2.3.1.1) at *a*>1 (in Figure 5 at *a*≈1.4).

Figure 5 shows equation (3.2.3.1.1), the semi-empirical extension of Anderson, the semi-empirical extension used in AERFORCE (a_{t1}=0.32) and a sempi-emprical correction suggested by Young rendered in [10]. Figure 6 shows the same curves but plotted in a diagram as dimensionless induction versus dimensionless wind speed.

The semi-empirical fix used in AERFORCE is thus identical to the type of linear curve suggested by Anderson, but used differently for large induction factors. Note that the cases $a=+\infty$ and $a=-\infty$ actually represents the same case of a hovering rotor (or any vase with *V*=0) The retention of the original momentum equation for values of $a_{t2} < a < \infty$ makes the equation work for the whole register of *V* and *T*.

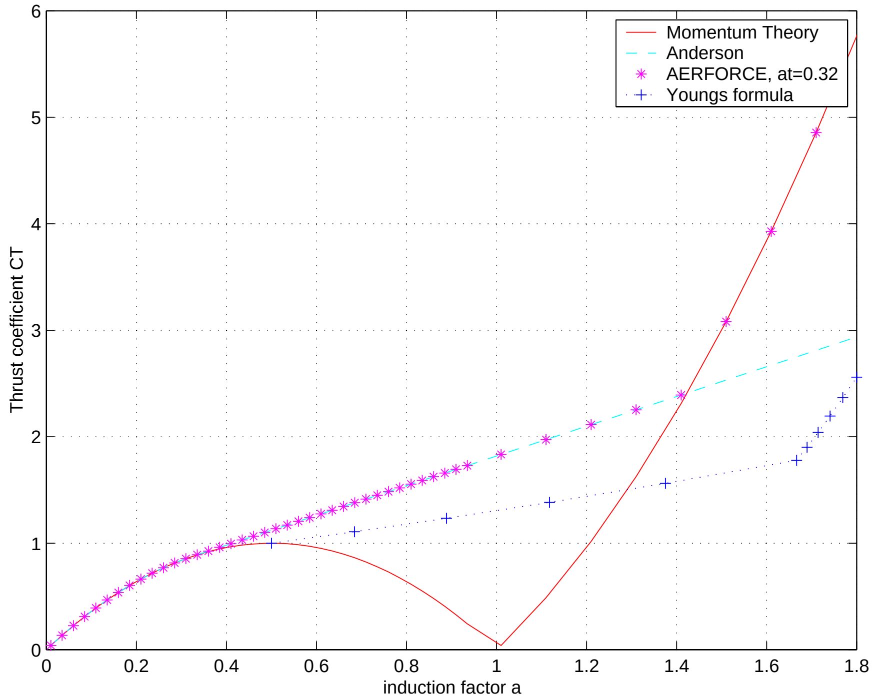


Figure 5. Thrust coefficient as function of the induction factor

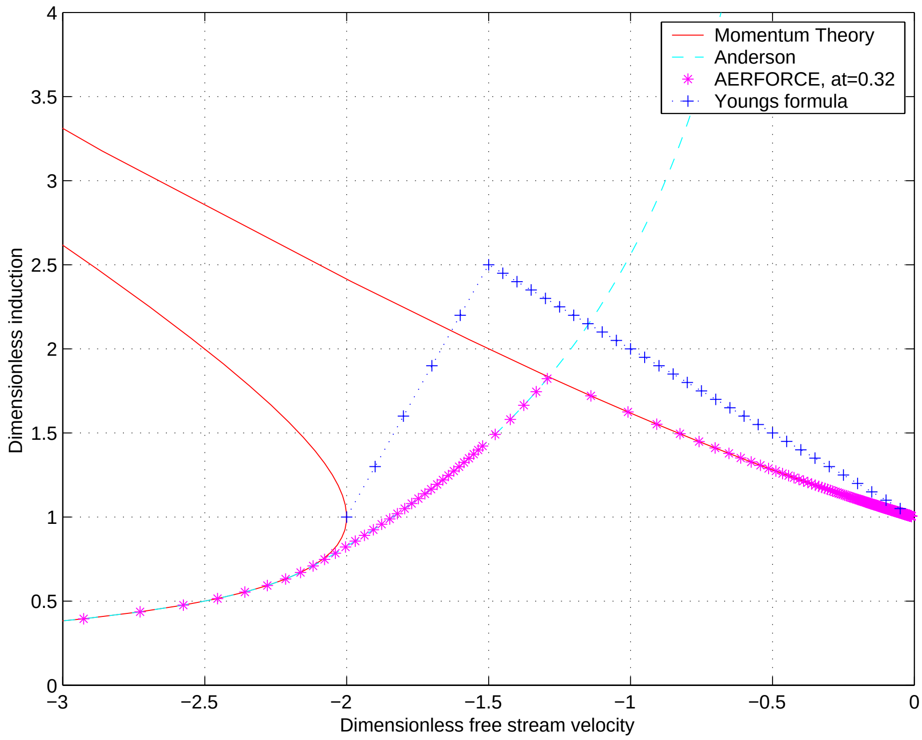


Figure 6. Dimensionless induction as function of dimensionless wind speed

3.2.3.2 Yawed flow

For yawed flow equation (3.2.3.1) is used to calculate the average normal induction. The area, S, is taken as the whole annulus area (for radius r, $S = 2\pi r dr$ but with corrections for coning as explained in section 3.4, $S = 2\pi \cdot r_{mom} \cdot dr_{mom}$).

For wind with a positive x_r-component, equation (3.2.3.1) is written:

$$4a\sqrt{(v_x - a)^2 + v_{ip}^2} = C_T \tag{3.2.3.2.1}$$

v_x and v_{ip} are the free wind components in the x_r-direction and the in-plane component normalized with the absolute value of the free wind, V_∞. a is the normal induction normalized in the same way: $a = -u_i / V_\infty$.

For pure axial flow, equation (3.2.3.2.1) becomes identical to (3.2.3.1.1) with the solution $a = 0.5\left(1 - \sqrt{1 - C_T}\right)$ for a<1.

With a non-zero in-plane velocity, the singularity at a=1 actually vanishes. Still the momentum curve equation (3.2.3.2.1) should be corrected with the semi-empirical fix in the turbulent wake and vortex ring state ($a_{t1} < a < a_{t2}$).

The induction factor a is found from equation (3.2.3.2.1) when $C_T < C_{T1}$ and when $C_T > C_{T2}$, where

$$C_{T1} = 4a_{t1}\sqrt{(v_x - a_{t1})^2 + v_{ip}^2} \tag{3.2.3.2.2}$$

and

$$C_{T2} = 4 \cdot a_{t2}\sqrt{(a_{t2} - v_x)^2 + v_{ip}^2} \tag{3.2.3.2.3}$$

In the intermediate region, a generalization of the semi-empirical curve with a linear $C_T(a)$ variation is used.

In the intermediate region, a is given from:

$$a = 0.5 \cdot \left(p - \sqrt{p^2 - 4q} \right) \tag{3.2.3.2.4}$$

with

$$k_{a1} = \frac{\sqrt{(v_x - a_{t1})^2 + v_{ip}^2}}{(1 - a_{t1})} \tag{3.2.3.2.5}$$

$$k_{a2} = \frac{\sqrt{(v_x - a_{t2})^2 + v_{ip}^2}}{(a_{t2} - 1)} \tag{3.2.3.2.6}$$

$$k_2 = \frac{k_{a2} - k_{a1}}{a_{t2} - a_{t1}} \tag{3.2.3.2.7}$$

$$p = \frac{c_0}{c_1} + \frac{k_{a1}}{k2} - a_{t1} \tag{3.2.3.2.8}$$

$$q = \frac{c_0 \cdot k_{a1}}{c_1 \cdot k_2} - \frac{c_0 \cdot a_{t1}}{c_1} - \frac{C_T}{c_1 \cdot k_2} \tag{3.2.3.2.9}$$

For pure axial flow, $k_{a1} = 1$ and $k_{a2} = 1$. This leads to division by zero in equation (3.2.3.2.7).

In AERFORCE, this is solved by adding a small in-plane velocity. If $v_{ip} < eps$ then v_{ip} is set to eps and $v_x = \sqrt{1 - v_{ip}^2}$. This solves the problem without making the resulting induction being significantly different from the value obtained with v_{ip} =0.

Figure 7 shows the solution of equation (3.2.3.2.1) and the same curve with the semi-empirical fix in the intermediate region. The yaw angle is 20 degrees.

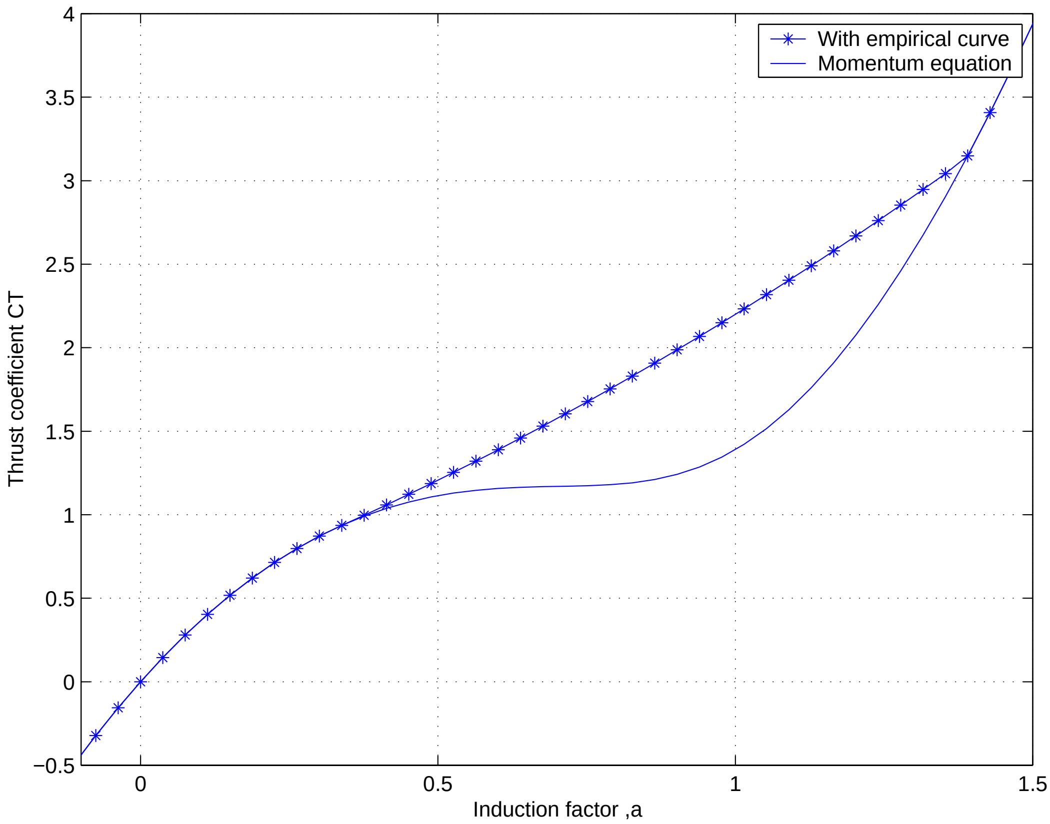


Figure 7, The thrust coefficient as function of the mean induction for yawed flow.

The difference between the results from the momentum equation and the semi-empirical fix decreases as the yaw angle increases and become practically the same at large yaw angles.

The induction factor, a, is obtained by normalizing with V_∞. A normalization with V_x would produce slightly different results. The difference is, however, believed to be small compared to other differences to an exact solution and the effect of a different choice of normalization velocity has not been investigated.

3.2.3.3 Tip loss factor

The tip loss factor, *F*, is calculated using Prandtl's formula.

The spacing between "vortex sheets" is calculated using the velocity

$$V_d = V_\infty (1 - a)\,. \tag{3.2.3.3.1}$$

which is used to calculate a tip speed ratio.

$$\lambda = \frac{R \cdot \Omega}{V_d} \tag{3.2.3.3.2}$$

F will be based on local conditions since $V_d = V_d(r)$

$$f = -0.5 \cdot NB \cdot \Omega \cdot (R - r) / R \cdot \lambda \tag{3.2.3.3.3}$$

$$F = 2/\pi \cdot \cos^{-1}(e^f) \tag{3.2.3.3.4}$$

If $V_d < 0$, then *F* is set to unity.

In order to be able to treat the case of low or zero rotation a minimum tip speed ratio is used in equation (3.2.3.3.2). In version 2b of AERFORCE, a minimum λ of 2 is used.

The use of *F* in the momentum equations differ in different versions of the momentum theory. The tip loss factor is used to factor the effective velocity change caused by the blade loads. The $\Delta V = 2 \cdot a \cdot V$ component in the momentum equation is therefore multiplied by *F*. The mass flow part through the rotor could also be multiplied by *F*. The latter is suggested by Wilson in [2] who writes the axial momentum equation as:

$$4 \cdot Fa(1 - Fa) = C_T \tag{3.2.3.3.5}$$

In AERFORCE, however, the other approach is used. *F* is then only added to the ΔV -part, which for the axial momentum equation can be written as:

$$4 \cdot a(1-a) = \frac{C_T}{F} \tag{3.2.3.3.6}$$

3.2.3.4 Tangential flow induction

The tangential flow induction is solved as a balance between the blade torque and the tangential momentum change. Only half of the induction in the far wake should be used in the induction used to calculate the angle of attack, hence the factor 2 in the denominator of equation (3.2.3.4.1) (see e.g. [9] page 77 ff). With the tip loss factor included the tangential induction used in the calculation of the angle of attack is:

$$u_{it0} = -\frac{F_{\tan}}{2 \cdot \rho \cdot S \cdot V \cdot F} \tag{3.2.3.4.1}$$

where $F_{\tan}$ is the sum force from all blades and *S* the annulus area.

3.2.3.5 A filter on the free stream velocity

The free stream velocity that is input to AERFORCE every time step might vary in time. For the calculation of the blade forces, the instant velocity at each blade element position is used. In the momentum equations, however, it is the velocity of a mass flow that should be representative. Since the momentum equations are solved on an annulus level, *V* used in the momentum equations is taken as the average free stream velocity for the *NB* blade elements. Further to average over the *NB* blades, a filter is applied to the velocity before it used in the momentum equations. A first order lag filter with a time constant τ is used. The time constant is chosen as the time for the rotor to rotate xx degrees for a rotor with a tip speed of 70 m/s.

$$\tau = xx \frac{\pi}{180} \frac{R}{70} \tag{3.2.3.5.1}$$

xx is set to 10 degrees in version 2b of AERFORCE.

The filtering is implemented as:

$$V_{filtered}(t_i) = \frac{V(t_i) + \frac{\tau}{dt} V_{filtered}(t_{i-1})}{1 + \frac{\tau}{dt}} \tag{3.2.3.5.2}$$

3.2.3.6 Blade loads

The thrust and blade torque is calculated using blade element theory. An angle of attack and a relative velocity are calculated using the instant free stream velocity, induction from the previous time step and instant blade element velocities. The main blade velocity of course normally comes from the blade rotation.

With the angle of attack, values of non-dimensional blade forces in the local wind reference system is obtained through look up tables of $C_l(\alpha)$, $C_d(\alpha)$ and $C_m(\alpha)$. These static values can be used as found (quasi steady profile aerodynamics), or a dynamic stall model can be used to calculate unsteady values of $C_l(\alpha)$, $C_d(\alpha)$ and $C_m(\alpha)$.

The blade forces are then obtained by equation (3.1) and in equivalent ways for the drag and local pitching moment.

3.2.4 Dynamic values of the induction

Equation (3.2.3.1) is the momentum equation for steady flow. Extensions to a time varying inflow can also be found in the literature.

One solution is put forward by ECN in [6], (here written somewhat differently to see the time constant more clearly).

$$\tau \cdot V \frac{du}{dt} + 4u(V - u) = C_T \cdot V^2 \tag{3.2.4.1}$$

where *u* here is the induction taken as $u = a \cdot V$

With τ equal to zero, equation (3.2.4.1) is identical to the normal momentum equation, e.g. (3.2.3.1) or (3.2.3.1.1) except the sign of *u*.

With a time constant, τ, much larger than the time stepping the Blade-Element/Momentum equation now can be solved without iterating for the static solution.

One calculation scheme with a single time constant is proposed by Øye in [11] as follows:

Compute the blade loads with angle of attack and relative velocity based on the induction from the previous time step and the free stream velocity from the current time step. Use this load to get a sort of quasi-steady value for the induction and apply a filter to the induction.

$$u' = \frac{T(V_n, u_{n-1})}{2 \cdot F \cdot \rho \cdot S \cdot (V_{n-1} - u_{n-1})} \tag{3.2.4.2}$$

and

$$u_n = u_{n-1} + (u' - u_{n-1}) \frac{dt}{\tau} \tag{3.2.4.3}$$

The same principle is used in AERFORCE but another model of Øye with two time constants [6] is used.

Blade forces are calculated based on the current wind speed and the induction from the previous time step. These blade forces projected on the x_r -axis are used to get the thrust, which is normalized to get the thrust coefficient.

$$C_T = \frac{Thrust}{F \cdot 0.5 \cdot \rho \cdot S \cdot V^2} \tag{3.2.4.4}$$

where V is the filtered average velocity at the current time step.

A quasi steady value of the induction is obtained from equation (3.2.3.2.1) or using the semi-empirical fix if $C_{T1} < C_T < C_{T2}$.

This induced velocitiy is viewed as the “quasi-steady” induced velocity in the normal direction without further iteration as suggested by Øye [6, page 251].

In the tangential direction, the in-plane blade forces are used to get a “quasi-steady” value for the tangential induction from equation (3.2.3.4.1).

As the next step the dynamic values of normal and tangential are calculated using the two differential equations in the Øye model from [6]

$$y + \tau_1 \frac{dy}{dt} = x + k \cdot \tau_1 \frac{dx}{dt} \tag{3.2.4.5}$$

$$z + \tau_2 \frac{dz}{dt} = y \tag{3.2.4.6}$$

These equations are solved with the “input variable” x set to the “quasi-steady” values of the induction normal and tangential respectively.

The output, z will be the dynamic values of the induction.

The constant k and time constants τ_1 and τ_2 are taken from [6] as:

$$k=0.6 \tag{3.2.4.7}$$

$$\tau_1 = \frac{1.1}{1-1.3a}\frac{R}{V} \tag{3.2.4.8}$$

with a maximized to 0.5.

$$\tau_2 = (0.39 - 0.26 \frac{r}{R}^2) \cdot \tau_1 \qquad (3.2.4.9)$$

The implementation in AERFORCE to solve (3.2.4.5) and (3.2.4.6) is, for the “normal direction”:

$$y_{nor} = \frac{u_{i0,qs} + k \cdot \tau_1 (u_{i0,qs} - u_{i0,qs\_1}) + \tau_1 \cdot y_{nor\_1}}{1 + \tau_1} \qquad (3.2.4.7)$$

$$u_{i0} = \frac{y_{nor} + \tau_2 \cdot u_{i0\_1}}{1 + \tau_2} \qquad (3.2.4.8)$$

Here _1 variables refer to values from the previous time step. Subscript *qs* refer to the “quasi-steady” values. Corresponding equations are also used for the tangential induction.

3.2.5 Alternatives for the calculations of the induction

As mentioned in section 3.2.1 there are alternative ways of calculating the “zero”-component of the induction. However, in the current version of AERFORCE, only the alternative when it is derived from the annulus mean from the *NB* blade loads is available.

A further question can be raised whether the azimuthal variation best is added to the tangential velocity or not. Two choices are possible in AERFORCE:

With LUITAN=1 the induction is computed with $u_{it,y}$ added as in equation (3.2.1.1), but if LUITAN=2, then $u_{it,y}$ is neglected and the tangential induction will be u_{it0}.

3.3 Blade loads with dynamic stall

Aerodynamic section coefficients are calculated using a dynamic stall model or taken as static values directly taken from the aerodynamic tables of $C_l(\alpha)$, $C_d(\alpha)$ and $C_m(\alpha)$. The dynamic stall model is an implementation of the Beddoes-Leishman model. The implementation in the AERFORCE subroutine package is described in [12].

An input parameter is used to determine weather the dynamic stall should be used or not. If LCNCL=0, then static values of $C_l(\alpha)$, $C_d(\alpha)$ and $C_m(\alpha)$ are used. If LCNCL=2, then the dynamic stall model is used.

3.4 Blade coning

Any variation of the induction in the x_r-direction is neglected. Coning is only considered as a reduction in effective area in the momentum equations.

The coning angle is input as a common coning angle for all blades.

The blade forces are first calculated in the *e*-system. The area $c \cdot dr$ in equation (3.1) is input to the subroutine for each element (variable name: `area`). This area should be the blade area in the *e*-system and not any projected area on any other system. To obtain the thrust and the tangential force the blade forces are then transformed to the *r*-system:

$$\overline{F}_r = S_{re}^T \cdot \overline{F}_e \tag{3.4.1}$$

The matrix S_{re} contains the blade coning angle.

When momentum forces are calculated the area projected on a plane with the x_r-axis as a normal is used. This area is calculated as:

$$S = 2\pi \cdot r_{mom} \cdot dr_{mom} \tag{3.4.2}$$

r_{mom} as well as dr_{mom} and R_{mom} is input to the subroutine (`r_mom, dr_mom` and `rtip_mom`).

r_{mom} as input should be $r_{mom} = r \cdot \cos(\beta)$ and analogous for dr_{mom} and R_{mom}.

Both r and r_{mom} and R and R_{mom} are input to AERFORCE 2b. Since the coning angle, β, in version 2 of AERFORCE also is input the dual input of both r and r_{mom} and R and R_{mom} in principle is unnecessary, but has been kept for "historical reasons".

3.5 Calculation scheme

The following are given for the calculation at time step *i*:

Values from the previous time step:

- $u_{i,w}$ given in the global system
- $u_{i0}(IR)$ and $u_{it0}(IR)$ given in the r-system
- Tip correction factor $F(IR)$

Values at the current time step:

- A value for V_w given in the global system
- Values of the free-stream velocity at each blade-element point. The free-stream velocity, $V(IB, IR)$, is given in the global coordinate system. **Corrections for tower blockage effects should be included in the value for the free-stream velocity**.
- Values of the absolute velocity of the blade-element. This velocity, $V_b(IB, IR)$, is given in the global coordinate system.
- A rotation matrix from the global system to the *r*-system
- Pitch and twist for each blade element and blade coning angle for all blades to construct the rotation matrixes from *r*-system to the *e*-system.

Calculation scheme:

A. Calculate χ and φ_χ in the rotor system as described in section 3.2

B. Use $u_{i0}(IR)$ and $u_{it0}(IR)$ from the previous time step and calculate $u_{i,y}(IB,IR)$ and $u_{it,y}(IB,IR)$ in the *r*-system as described in section 3.2.2. Sum to obtain the normal and tangential induction by equations (3.2.1) and (3.2.2)

C. Transform the induction to the *e*-system.

D. Calculate effective velocity and angle of attack and then calculate blade forces, $F_b(IB,IR)$.

E. Transform forces and the free stream velocity to the *r*-system and use BEM theory to calculate new values the quasi steady induction, $u_{i0,qs}(IR)$ and $u_{it0,qs}(IR)$.

F. Apply dynamic inflow equations save the values as $u_{i0}(IR)$ and $u_{it0}(IR)$ for the next time step.

G. Calculate a value of $u_{i,w}$ from a weighted mean of u_{i0} from selected radial stations.

H. Transform $u_{i,w}$ to the global system and save the value for the next time step

3.6 Limitations of AERFORCE

The main limitations in AERFORCE lie in the limitations of the aerodynamic method itself.

The idea is that AERFORCE should be a robust code. It should work for all imaginable cases. That it should work means here that it should give a result within the limitations of the theory.

The code therefore works in “turbine mode” as well as in “propeller mode”. It works for the flow to the rotor disc at any yaw angle and for the rotor rotating in any direction as well as with the turbine parked.

For the code to work at all inflow angles, however, requires that airfoil data are given for the appropriate range of angles of attack, preferably for the range [-180, 180] degrees.

The result in a calculation is very dependent on the tables of airfoil data given as input. These data must be corrected for 3D-effects due to rotation as described in e.g. [14], [15].

3.7 Constraints on time step length

The method is intended for time simulations in a time-stepping manner.

As noted in section 3.2.4 no iteration of induction is made since the time step is assumed much shorter than the time constants for the induction. Of the two time constants in equations (3.2.4.8) and (3.2.4.9), τ_2 is the shortest with a minimum value of $0.14\frac{R}{V}$. As long as the time step is shorter than this the induction will also likely converge nicely to its static value if, a steady solution is sought.

If the “time true” dynamic behavior of the induction should be caught, as in an aeroelastic simulation, a shorter time step is, however, required. This is most often automatically fulfilled since the requirements to solve for the structural dynamics most often require a rather short time step. Regarding the calculation of the induction the requirements is that the time step should be at least as short as corresponding to an azimuth division of 10°.

For the dynamic stall part, a shorter time step is generally required. The dynamic stall method is programmed such that the use of too long a time step does not cause any problems other than that the solutions tends to the quasi steady solution. With the aim to model the unsteady airfoil effects, the time step must however be sufficiently short. The relevant time scale is related to the wind speed relative to the airfoil and the local airfoil chord. The unsteady airfoil effects scale with the dimensionless time:

$$\Delta s = \Delta t \frac{W}{c/2} \tag{3.7.1}$$

where W is the velocity relative to local blade element.

A minimum value of $\Delta s = 0.5$ is required for most cases.

W in equation (3.7.1) can be approximated with $r\Omega$. The requirements on the time step then becomes hardest at the tip and, with $\Delta s = 0.5$, the requirements on the time step then van be formulated as:

$$\Delta t < 0.25 \frac{c_{tip}}{V_{tip}} \tag{3.7.2}$$

where c_{tip} is the chord of the tip element and $V_{tip} = R\Omega$.

4 Input to the AERFORCE subroutine, common-areas and error handling.

The AERFORCE subroutine is called with a number of formal parameters but communication via common-blocks is also used.

Most variables in these common-blocks are used for internal communication between different subroutines in the package, but some are needed to be set by the calling routine as explained in subsequent sections.

The variables that are input as formal parameters are explained in the Fortran source file and for some cases commented below.

4.1 Conventions for variable names.

For the solution of differential equations, variable values at the previous time step is needed.

Variable names ending with **_1** refers to variable values at the previous time step.

In Subroutine AERFORCE there are some variables named xx_track. The extension **_track** is used to tell that these are variables used to take values out from the subroutine (via common-block /trackvar999/) for e.g. plotting. The rows in AERFORCE were _track variables are set can be removed without disturbing the function of the subroutine or affecting results.

4.2 Common areas in Dyncl_c.inc

This file includes common-areas with variables that are needed by the dynamic stall subroutines e.g. storage of airfoil aerodynamic coefficients.

4.2.1 Storage of static airfoil data.

The aerodynamic static airfoil data should be stored in data tables Cl(alfa), Cd(alfa) and Cm(alfa).

These tables are found in the common area `/profidata996/`

Each airfoil is associated with one table.

Data for at least one airfoil is needed.

4.2.1.1 Minimum requirements when dynamic stall calculations are not made (lcncl=0)

Static airfoil data are stored in variables:
`alfastin`, **`clstin`**, **`cdstin`** and **`cmstin`**.

These variables are matrices `alfastin(ia,ip)`, `clstin(ia,ip)` etc. The second index refers to table no. i.e. which airfoil it is. The first index refers to the row number in the airfoil data table.

An example of one airfoil data table is shown below[4]

Alfa	Cl	Cd	Cm
-10.0	-0.8600	0.0150	-0.0375
-4.0	-0.1283	0.0067	-0.0785
-2.0	0.1252	0.0068	-0.0834
2.0	0.6273	0.0073	-0.0925
4.0	0.8748	0.0079	-0.0964
6.0	1.1190	0.0086	-0.0997
8.0	1.3525	0.0102	-0.1014
10.0	1.5349	0.0154	-0.0973
12.0	1.5899	0.0271	-0.0843
14.0	1.5957	0.0511	-0.0837
16.0	1.5963	0.0797	-0.0868

[4] Note that the angle of attack as input to `alfastin(.,.)` should be in radians and not degrees

This table contains 11 rows. If it would refer to airfoil number 1, then the 11 C_l-values should be stored in `clstin(i,1)` with i from 1-11.[5]

The angle of attack stored in `alfastin(ia,ip)` should be in radians. If original airfoil tables contain data with the angle of attack in degrees, a conversion to radians has to be made before they are stored for use by AERFORCE. Additional to `alfastin`, `clstin`, `cdstin` and `cmstin` the variable **`nin(`**`ip`**`)`** (in the same common area as `alfastin` et.c.) is also needed. `nin` tells how many rows that are used in each table. In the example above, `nin(1)=11` should be set.

`npr` is used to tell how many airfoil tables that are used.

`ipr``(ir)` is the airfoil pointer for radial station `ir`.

One alternative that could be used is that airfoil data has been pre-processed before calls to AERFORCE. E.g. interpolation in thickness from input airfoil data tables. The storage could then be one airfoil data table per radial element in the common areas. `ipr(ir)` should then be the same as the radial station number and `npr=nr.`

4.2.1.2 Range of angle of attack in input airfoil tables.

The range of angles of attack in each airfoil data table must cover the angles of attack that will be encountered during the calculations. If an angle of attack is calculated outside the input range, then the program will halt (in subroutine interp999) when C_l, C_d and C_m values are sought.

---

[5] The table must of course include the full range of angles of attack that will be encountered during the calculations. If an angle of attack is calculated to e.g. 20 degrees and the table with alfa[-10,16] is used, then the program will stop.

To cover the full 360 degrees range of angles of attack, α should be given in the range [-180,180] degrees.

4.2.1.3 Additional requirements when dynamic stall calculations are made (lcncl=2)

If dynamic stall calculations are made, values must be set also to variables `cninvin, cnsepin, fnstin, alfa0, alfa02 cn_alfa` and `cn_alfa2` in the common area `/profidata996/`

(`cn1pos and cn1neg` is used only for one version of calculation of vortex lift (`lvormeth=1`)).

The way dynamic stall calculations are made is dependent on which choice of sub-methods that are selected. The use of the dynamic stall model is described in [12].

If `lfmeth=4` is used for the calculation of separation lag, then the matrices `cninvin, cnsepin, fnstin` and vectors `alfa02` and `cn_alfa2` are automatically filled by a call to subroutine `kirchmake_c` the first time AERFORCE is called (`first_call`=1 or `first_call`=2).

If `lfmeth=4` is used, no additional requirements regarding static airfoil data than to fill `alfastin, clstin, cdstin, cmstin` and `nin` is needed. Hence the same requirements as needed if dynamic stall not is used.

However, in order for the subroutine `kirchmake_c` to be able to calculate sensible values of `alfa0, alfa02 cn_alfa` and `cn_alfa2,` the angle of attack division around α=0 should be sufficiently dense and the $C_l(\alpha)$ curve should be fairly linear around C_l=0. Otherwise the calculated value of `alfa0` and `cn_alfa` might be badly estimated. The same holds for α around ±180 degrees.

4.2.2 Dynamic stall model parameters

4.2.2.1 Choices between submodels

Variables to store choices for which sub-model to use for the dynamic stall calculations are stored in the common-area

/`typeparam`/ These variables need to be set outside of the AERFORCE subroutine package.

The different sub-models are described in [12].

If `lcncl`=0, then the dynamic stall model is not used. The setting of the rest of the variables in /`typeparam`/ and the variables in /`potcoeff`/, /`tfparam`/, /`vorparam`/ can then be ignored (they are not used then).

4.2.2.2 Coefficients and time constants

The area /`potcoeff`/ contains the coefficients for the inviscid circulatory lift response. Recommended values for these coefficients can be found in [13] as: `coeffa1=0.3, coeffa2=0.7, coeffb1=0.13, and coeffb2=0.53`

/`tfparam`/ and `/vorparam`/ contain input variables for the separation lag part and the vortex lift part of the dynamic stall model. How to choose these parameters is described in [12].

4.2.3 Variables used for the calculation of dynamic stall

The areas /`curval`/ and `/oldval`/ contain variables used for the calculation of the dynamic stall (state variables). Values at the current time step is stored in the variables in /`curval`/ and the values at the previous time step is stored in `/oldval`/. None of the variables in /`curval`/ or `/oldval`/ need to be set outside of the AERFORCE subroutine package except for one occasion as explained in section 4.6.

4.3 Common areas in aer_com999.inc

Variables that are needed for the previous time step for the calculation of dynamic inflow and the azimuthal induction variation are stored in the common area /`bem_oldval`/. Corresponding values for the current time step are stored in /`bem_curval`/. None of the variables in /`bem_urval`/ or `/bem_oldval`/ need to be set outside of the AERFORCE sub-routine package except for one occasion as explained in section 4.6.

4.4 Common areas in aer_track999.inc

As mentioned in section 4.1, there are variables named xx_track. The extension _track is used to tell that these are variables used to take values out from the subroutine (via common-block /trackvar999/) for e.g. plotting. The rows in AERFORCE were _track variables are set can be removed without disturbing the function of the subroutine or affect results

4.5 Matrix and array sizes set in Matzise.inc.

This file includes the sizes of arrays and matrices.

```
c----------------------------------------------------------------------
c Include file with size of matrices
c----------------------------------------------------------------------
      integer lda,ldp,ldb,lde
      PARAMETER(lda=149, ldp=10, ldb=3, lde=30)
```

`lda` is the dimension of the number of rows in the matrices for variables of the airfoil tables, e.g. `CLSTIN(lda,lde)`. Note, however that this dimension of the matrices should not be mixed up with the number of rows (with data) that is stored for each airfoil, `npr.`

lde is the dimension of the number blade elements for each blade. I.e. the maximum number of blade elements per blade that can be used.

ldb is the dimension of the number blades. I.e. the maximum number of blades that can be used.

ldp is the dimension of the number of tables for airfoil data. i.e. the maximum number of airfoils as input. This parameter is, however, not used in the current version of the subroutine package. It was used to dimension matrices `CLSTIN(lda,ldp)` etc and `nin(ldp)` etc. `ldp` is now exchanged with `lde`. If storage area is critical and the method of assigning one set of airfoil table data fore each radius not is used, then the old dimensioning can be used. Then exchange `lde` with `ldp` in the declaration of all variables in the common area /`profidata996`/.

4.6 Special call for the first time step

The first occasion during a time simulation when AERFORCE is called is a bit special. At this time step no values for state-variables from the previous time step are available. Also some initial settings should be made.

There is therefore a variable **`first_call`** as a formal parameter to the routine.

At the first time step, `first_call` should be set to 1. All `_1` variables will then be set to predefined values by call to Subroutines `initoldvar_c` and `init_aerforce_oldvar` from subroutine AERFORCE and by call to `clcd_firsttime` from subroutine `clcdcalc_c`.

However, the initial values for all _1 variables could also be user set. This need could e.g. occur if a time simulation is stopped and needs to be restarted. In this case `first_call` should be set to 2 the first time AERFORCE is called. All _1 variables in common areas /bem_oldval/ and /oldval/ should also have been read in from e.g. a dump of the corresponding values in /bem_curval/ and /curval/ when the previous run was stopped.

For all subsequent time steps, except the first one, first_call must be set to zero.

4.7 Error and warning handling

Most possible error cases are handled by stopping the execution with a fortran STOP command. Error print-out is then made to a file linked to unit "loguint", which is input to the AERFORCE subroutine.

The file linked to unit "loguint" is also used for warning print-outs.

4.8 Input requirements to AERFORCE

Besides from the values to the formal parameters of the subroutine AEERFORCE values in the following variables in common blocks must be set:

I. Airfoil force coefficient tables in /profidata996/ as explained in section 4.2.1.

II. Choices for which sub-model to use for the dynamic stall calculations are stored in the common-area /typeparam/ as explained in section 4.2.2.1.

III. Coefficients and time constants stored in the common-areas /potcoeff/, /tfparam/ and /vorparam/ as explained in section 4.2.2.2.

If the airfoil forces are modeled as static forces (dynamic stall model not used), then the only variable of II) and III) above that needs to be set is lcncl=0.

4.9 File names

The following files are required:

File	**Subroutines and functions**
aerforce_2b.f	Subroutine **aerforce** Subroutine **init_aerforce_oldvar**
momentum_subs.f	Subroutine **acalc** Subroutine **ftipcal** Function **asolve** Subroutine **uiwake_index**
clcdcalc_c.f	Subroutine **clcdcalc_c** Subroutine **clcd_firsttime**
kirchmake_c	Subroutine **kirchmake_c**
interp996.f	subroutine **interp996**
dyncl_c.f	Subroutine **dyncl_c**
vo_sub_c	Subroutine **vocal1_c** Subroutine **vocal2_c**
v_esti_c.f	subroutine **v_esti_c**
initoldvar_c.f	Subroutine **initoldvar_c**
cu_2_old_aer_c.f	Subroutine **cu_2_old_aer_c** subroutine **cur_aerforce_2_old**

Table 1. Fortran source filed needed for the aerforce subroutine package

And the include files

matsize.inc
aer_com999.inc¨
dyncl_c.inc
aer_track999.inc

A driver program **aersim.f** is an example of the use of the AERFORCE subroutine package.

Figure 8 below shows an example of the digital visual fortran workspace for the program **aersim**.

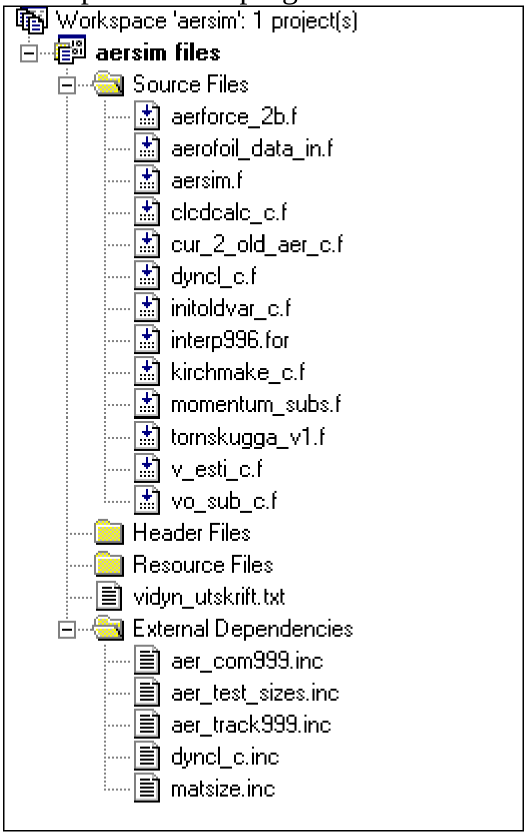


*Figure 8. Needed source and include files for program **aersim***

References

[1] Snel, H, “Review of the Present State of Rotor Aerodynamics”, Wind Energy, Pilot Issue Spring 1998.

[2] Wilson, R. E. and Lissaman, P. B. S. “Applied Aerodynamics of Wind Power Machines”. Oregon State University, 1974.

[3] “State of the Art of Aeroelastic Codes for Wind Turbine Calculations”, IEA 28th Meeting of Experts, Lynby, Denmark April 11-12, 1996, Edited by Maribo Pederesen, Dept of Fluid Mechanics, Technical University of Denmark.

[4] “Forskning i Aeroelasticitet – EFP-98” Risø-R-1129(DA), August 1999 (In Danish), Edited by Helge Aagard Madsen.

[5] Voutsinas S.G., “Development of a New Generation of Design Tools for Horizontal Axis Wind Turbines”, NTUA, Final Report JOU2-CT92-0113, Athens,1995

[6] Snel, H. and Schepers, J.G. “Joint Investigation of Dynamic Inflow Effects and Implementation of an Engineering Method”, Netherlands Energy Research Foundation, ECN. ECN-C—94-107, April 1995.

[7] Johansson, Hjalmar “Calculation of the stationary blade load aerodynamics in VIDYN”. (In Swedish) TG-rapport nr 98-15, May 1998.

[8] Bramwell, A. R. S. “Helicopter Dynamics”, chapter 4, Edward Arnolds (publishers) LTD, London 1976

[9] Freris, L. L. “Wind Energy Conversion Systems”, Prentice Hall, 1990

[10] Padfield, G. D., ”Helicopter flight dynamics: the theory and applications of flying qualities and simulation modelling”, AIAA education series, Washington , DC : AIAA, 1996, ISBN 1-56347-205-8

[11] Øye, S. “Unsteady Wake Effects Caused by Pitch-Angle Changes”, IEA 1st symposium on Aerodynamics of Wind Turbines, London 15 october 1986, Edited by Maribo Pedersen, Danish Technological University.

[12] Björck, A. “ DYNSTALL: Subroutine Package with a Dynamic stall model”. FFAP-V-110, The Aeronautical Research Institute of Sweden, June 2000.

[13] J.G. Leishman and T.S. Beddoes, "A Generalised Model for Airfoil Unsteady Aerodynamic Behaviour and Dynamic Stall Using the Indicial Method", Presented at the 42nd Annual Forum of the American Helicopter Society, Washington D.C. June 1986

[14] A. Björck, “Dynamic Stall and Three-dimensional Effects, Final report for the EC DGXII Joule II Project, JOU2-CT93-0345”, FFA TN 1995-31, January 1996, The Aeronautical Research Institute of Sweden

[15] Bak, Christian, et al. “Airfoil Charactersitics for Wind Turbines” Risø-R-1065(EN), Risø National Laboratory, Roskilde, Denmark, March 1999

Issuing organization	Document no.	
Flygtekniska Försöksanstalten Box 11921 S - 161 11 Bromma Sweden	FFA TN 2000-07	
	Date May 2000	Security Unclassified
	Reg. No.	No. of pages 62
Sponsoring agency Swedish National Energy Administration	Project no. VU0313	Order/Contract STEM P11556-1
Title AERFORCE: Subroutine Package for unsteady Blade-Element/Momentum Calculations		
Author Anders Björck		
Checked by Björn Montgomerie Wind Energy Section	Approved by Sven-Erik Thor Head Wind Energy Section	

Abstract

A subroutine package, called AERFORCE, for the calculation of aerodynamic forces of wind turbine rotors has been written. The subroutines are written in FORTRAN.

AERFORCE requires the input of airfoil aerodynamic data via tables as function of angle of attack, the turbine blade and rotor geometry and wind and blade velocities as input.

The method is intended for use in an aeroelstic code. Wind and blade velocities are given at a sequence of time steps and blade forces are returned.

The aerodynamic method is basically a Blade-Element/Momentum method. The method is fast and coded to be used in time simulations. In order to obtain a steady state solution a time simulation to steady state conditions has to be carried out.

The BEM-method in AERFORCE includes extensions for:

1.) Dynamic inflow: Unsteady modeling of the inflow for cases with unsteady blade loading or unsteady wind.
2.) Extensions to BEM-theory for inclined flow to the rotor disc (yaw model).
3.) Unsteady blade aerodynamics: The inclusion of 2D attached flow unsteady aerodynamics and a semi-empirical model for 2D dynamic stall.

Keywords/Nyckelord

Aerodynamics, Wind Energy, Unsteady Aerodynamics, Dynamic Stall, Wind turbine

Distribution	STEM	VKK-råd	FFA
Copy No	1-7	8-17	18-50