

An open source FORTRAN90 Code for aerodynamic design of wind turbine blades - V3 *

M. Korjahn, bewind GmbH, Rendsburg, Germany
A.P. Schaffarczyk, Kiel University of Applied Sciences, Kiel, Germany
D. Schlipf, Flensburg University of Applied Sciences, Germany
contact: Alois.Schaffarczyk@FH-Kiel.de

Version 3: May 21, 2025

Abstract

A specific aerodynamic design methodology has been implemented into an open source FORTRAN90 code. It is available via github: <https://github.com/Schaffarczyk/KSS/>

*CFD Lab Report No, 128, Kiel, Germany

Contents

1	Introduction	6
2	The Blade-Element Momentum (BEM) formulation	6
2.1	General	6
2.2	Multiple Solutions of the BEM-Equation	7
2.3	Tip-correction	7
2.4	Glauert Correction $cT(a)$ for $a > a_c$	11
2.5	Sign convention	11
3	Code Description	11
3.1	A short note on FORTRAN	11
3.2	Numerical Approach	12
3.3	Main program	12
3.4	Memory mangement	12
3.5	Subroutines	12
3.6	Functions	13
3.7	Compilation	14
3.7.1	LINUX	14
3.7.2	Windows	14
3.8	Modularization	14
3.9	Work Flow	14
4	Output Files and Tools for Visualization	14
4.1	Description of Output Files	14
4.2	Description of GNUPLOT scripts	15
5	Mode: Analysis	15
5.1	Input data	15
6	Mode: Design	22
6.1	Aerodynamic Optimization	22
6.1.1	Betz' and other criteria	22
6.1.2	Change of Twist	24
6.1.3	Change of Thickness - Arrangement of Profiles	24
7	Sample cases	25
7.1	Case 1: The NREL 5MW Baseline blade	25
7.2	Case 2: A 15MW WT reference blade	28
7.3	Case 3: A 20 MW blade	30
8	Comparison with <i>Xturb</i>	30
9	Summary of investigated blades	30

10 Summary, Discussion and Outlook	32
11 Disclaimer	32
12 List of Abbreviations	32

Draft

List of Figures

1	Example of distributions of multiple solutions from balancing aerodynamic (lift) forces and local (axial) momentum.	8
2	Example of how a binary search algorithm finds multiple solution along a whole blade (CIG10MW)	9
3	Variation of Angle of Attack (CIG10MW)	10
4	Sample thickness distributions	18
5	Refined thickness distribution from NREL 5 MW	19
6	Thickness distribution change to decrease mass for a 15 MW reference blade [24]	20
7	Betz and Sørensen	24
8	KSS Optimisation after 5 twist loops: $a(r)$	26
9	KSS Optimisation (Own 20MW, $r_{tip} = 133m$) after 5 twist loops: some angles	27
10	cP map from KSS, including control path (magenta) and an extended pitch variation $-15^\circ < \vartheta < 20^\circ$	28
11	cP map from NREL [4]	29
12	Comparison of circulation for an own 20 MW blade	30

List of Tables

1	Part of a sample job description file <i>Machine.in</i> based on NREL's 5MW reference wind turbine	16
2	Sample blade description file <i>BlaDes.in</i> for NREL/IEAwind 15MW reference wind turbine	16
3	Sample Thickness assigment file <i>ProThick.in</i> for NREL/IEAwind 15MW reference wind turbine	17
4	Sample Thickness distribution file <i>ThickDis.in</i> for NREL/IEAwind 15MW reference wind turbine	17
5	Sample output file <i>Bem.out</i> for NREL/IEAwind 15MW reference wind turbine	21
6	Sample aerodynamic data <i>FF21.aer</i> for NREL/IEAwind 15MW reference wind turbine	23
7	Parameters describing a design case	23
8	Rated rotor power of NREL 5MW baseline from various sources. HAWC data from Mr. Hinrichs, aerodyn engineering GmbH, bladed from Mr. Maniok, DNV If all value are equally weighted $P = 5356 \pm 89$ kW results	25
9	Comparison for NREL/IEAwind 15 MW reference wind turbine . . .	28
10	Summary of investigated blades	31
11	list	32

1 Introduction

Here we report on an independent FORTRAN 90/95 (F90 for short) implementation of a purely aerodynamic wind turbine blade design methodology. It is intended to be as detailed as possible although a fair amount of knowledge of wind turbine aerodynamics is necessary, see, for example, [25, 23, 22]. Some - but not all - needs form a more practical point of view (maximum chord and twist, etc.) have been taken into account. This is in contrast to [10] who tried to find out what maximum c_P may be reached without any restrictions by *practicability*.

Our basis, a lecture of Korjahn [12], presents a so-called two-point design. In a first attempt, however, we only describe an implementation of the first part: to reach at a blade with as large parts with optimum $a(r)$ ¹ as possible. *Design point 2* consists of a certain safety against gusts corresponding to stalled parts on the blade. Therefore the AOA has to have some safety-margin against this AOA-stall.

Main objectives of our approach are:

- understand and formalize the approach and
- compare to other codes.

Further or additional sources from

1. Korjahn [12],
2. Bak [1],
3. Schmitz [23] and
4. Johansen et al. [10].

are recommended to complement understanding.

Important: The code is by no means fool-proof:

Bugs may still be present.

Input data and esp. the aerodynamic one has to be checked as far as possible before its use.

2 The Blade-Element Momentum (BEM) formulation

2.1 General

(Conservation of) momentum on independent annular rings[25, 17, 20] serves as the fluid mechanical foundation. Key elements of a complete BEM formulation are

¹Usually the term Betz-optimum is related to $a(r) = 1/3 = \text{const.}$ Very recently [9] it has been shown that including wake-rotation and tip-loss a somewhat different distribution results: Close to the hub $a \rightarrow 0.25$ and close to the tip $a \rightarrow 0.4$.

already discussed in [5] and further refined in classical text from Wilson et al. [20] and de Vries [17], for example. Any applied aerodynamic wind turbine investigation uses it and it is still today regarded as the *work horse*. CFD can hardly claim to be more accurate although a large number of assumptions inherent to the BEM model have been overcome by this very different approach. The most recent analysis of the BEM-method is given in [7] with some focus on mathematical rigor and optimization.

2.2 Multiple Solutions of the BEM-Equation

It was already recognized by Wilson et al [20] p 108 ff. that under some specific circumstances the BEM-equations may not produce unique solutions. Maniaci [3] and Ning [21] further elaborate on this subject. To give an example, Fig. 1 (for a special section of a special commercial blade) shows the local thrust coefficient cT_{loc} as function of a , the axial induction, defined by $u_1 := (1 - a) \cdot U_{-\infty}$, $0 \leq a \leq 0.5$. Momentum balance gives

$$c_T = 4a \cdot (1 - a), \quad (1)$$

whereas the counter-force from lift is its proper projection into the direction of the wind. Note, that this is a approximate description only, as a' (circumferential induction) is neglected as well as tip-correction (see section 2.3) and a correction of c_T for larger values of a ($a > a_c$), see section 2.4.

Our investigation for a 20 MW blade of 133 m length confirm this. It was recognized that the traditional fixed-point iteration did then not converge but simply jumps between two different values.

As a consequence we improved our implementation and search algorithms as follows:

- Firstly, scan the whole a -region $0 \leq a < 1$ in steps of 0.01. If there are solutions found by a binary search algorithm,
- start using standard fixed-point iteration until a maximum of number is reached
- switch to Newton(-Raphson) to to more iterations, if no convergence is reached:
- use a binary search algorithm *rtbis* from [18]

Fig. 2 shows an example. Therefore the argument of [20], p. 31

As a guiding principle, it is suggested that the solution to be chosen in the case of multiple solutions is the one that maintains continuity of angle of attack along the blade span.

can be confirmed.

2.3 Tip-correction

Momentum theory is valid only for an infinite number of blades $B \rightarrow \infty$. As in most designs B is small ($B < 4$) this has to be taken into account. Indeed, as was shown

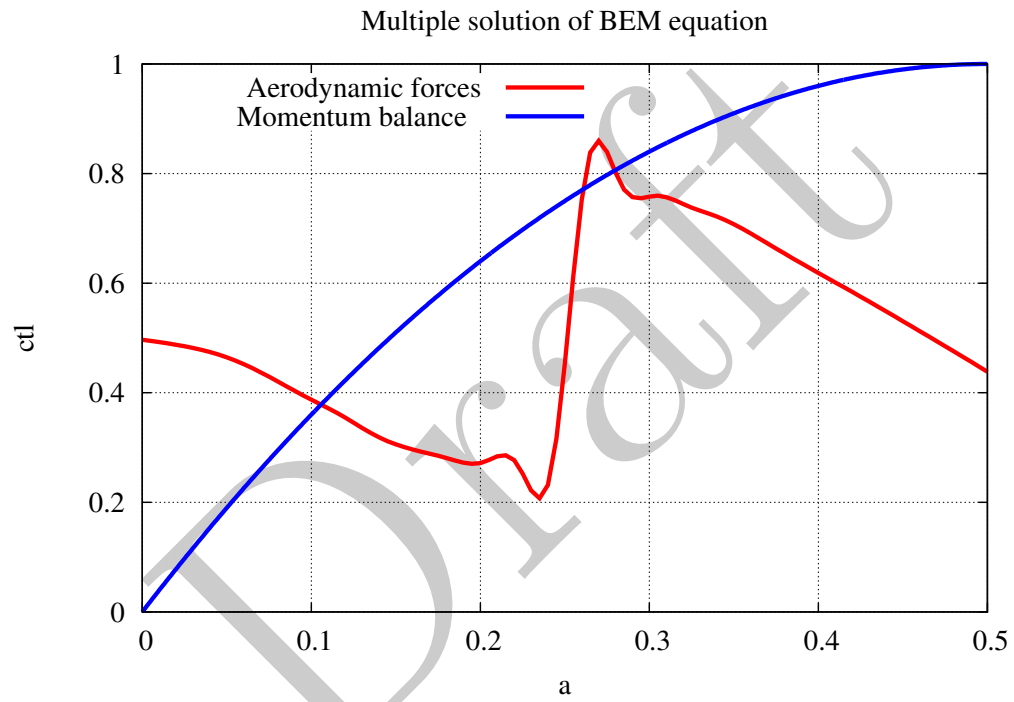


Figure 1: Example of distributions of multiple solutions from balancing aerodynamic (lift) forces and local (axial) momentum.

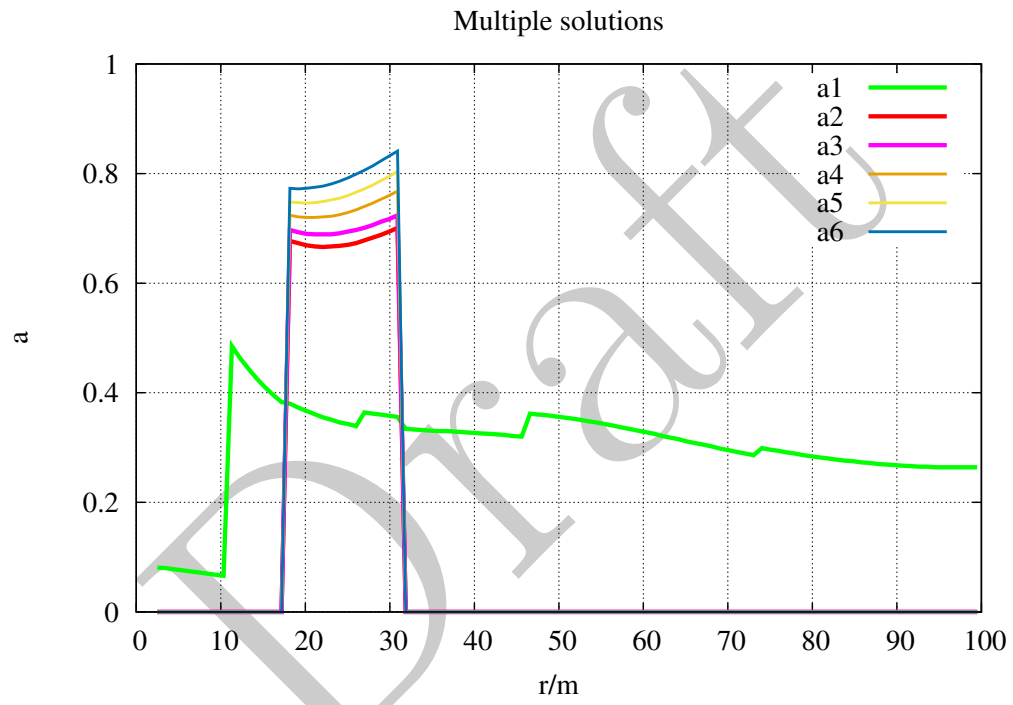


Figure 2: Example of how a binary search algorithm finds multiple solution along a whole blade (CIG10MW)

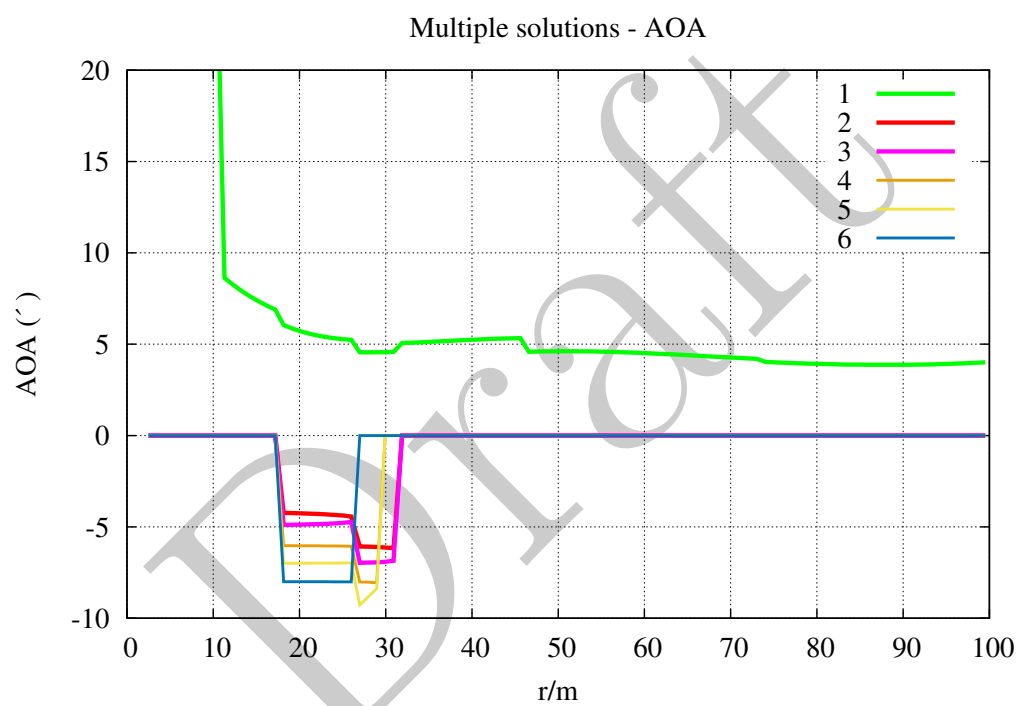


Figure 3: Variation of Angle of Attack (CIG10MW)

very early *tip losses* are as large as profile drag and must be accurately embedded into the code. Traditionally, the simplest (and oldest) version uses a factor F , which reduces circulation to zero when $r \rightarrow r_{tip}$:

$$F = \frac{2}{\pi} \arccos \left(\exp \left[-\frac{B/2(1 - r/R_{tip})}{r/R_{tip} \sin(\varphi)} \right] \right) . \quad (2)$$

Here, φ is the all-important flow angle defined by

$$\tan(\varphi) = \frac{1 - a}{\lambda(1 + a')} , \quad (3)$$

which refers to $\lambda = \omega \cdot r / U_{\infty}$ the local Tip-Speed-Ratio (TSR). $\omega = RPM \cdot \pi / 30$ is the angular velocity.

2.4 Glauert Correction $cT(a)$ for $a > a_c$

It is further well known:

- Momentum theory is valid for $0 \leq a \leq 0.5$ only, and
- for some a_c Eq. 1 becomes inaccurate.

To some extent this can be cured by *empirical engineering* extensions which are listed, for example in [7], Table 2.1. In principle at some $a_c = 0.33...0.4$ a more or less linear curve is attached to Eq. (1) so that $c_t(a = 1)$ approaches (somehow) 2 for $a \rightarrow 1$.

2.5 Sign convention

Sign issues esp. for the pitch are notorious. Following our basic interpretation

$$\alpha \text{ (AOA)} = \varphi \text{ (flow angle)} - \vartheta \text{ (twist)} , \quad (4)$$

we define **pitch** as being added to twist.

3 Code Description

3.1 A short note on FORTRAN

We will not defend ourselves why we use the *Grandfather of all (higher) programming languages* but only use it. It has been tried to write a code as simple as possible to make understanding and changes (**i.e. maintainability**) as simple as possible. A minimum set of algorithmic features is used. Especially, no tricks from Computer Science have been used. This code was(is) developed using *gfortran*, version 4.8.5 from 2015, see section 3.7. A more detailed description is given in [14].

3.2 Numerical Approach

Everybody knows from own experience the NEWTON-(Raphson) method for finding zeros of functions ($f(x_0) = 0$) or SIMPSON's rule for 1D-Integration. The ultimate manifestation of numerical algorithms may be found in the famous (series of) text book(s): *NUMERICAL RECIPES in Fortran 77* [18]. From here we have copied

1. SPLINE and
2. SPLINT.

Unfortunately, some overshooting at sharp corners might still be possible. *Bezier* polynomials were coded independently from its definition see [15].

3.3 Main program

KKS.f contains the main code. Its is merely a collection of routine-calls for set-up and definition of the course of investigation:

1. (dynamic) memory allocation,
2. Read *machine.in*,
3. Read in spline thickness distribution from *ThickDis.in*,
4. Analyze a given set of profiles, L2D-max.
5. If design-mode is chosen:
6. An initial BEM calculation is performed and
7. chord and twist might be (Bezier-) smoothed or
8. Twist may changed to improve c_P , see section 6.1 .

3.4 Memory mangement

mem.f defines all GLOBAL data. To use each module must start with a

```
use mem
command line.
```

3.5 Subroutines

1. BEM: Bem iteration as described in [22],
2. ANALYSIS: Analyse a blade described in *BlaDes.in* using routine *BEM*,
3. DESIGN: Determines chord and twist by given profiles and Betz [22], optimum criteria. See section 6.1 for more information about optimization.,

4. AIRFOILS: searches given α, c_L, c_D data for maximum of c_L/c_D , $c_{L,max}$ and slope at zero lift.
5. TipShapeEL: defines an elliptic (vertical tangent) tip shape geometry,
6. TipShapePara: defines an parabolic tip shape geometry determined by smooth slope at the beginning of the tip region and zero chord at tip.
7. BEZIER: smooting of a given data set according to the definition of Bezier polynoms,
8. smoothCHORD: smooths chord with subroutine BEZIER,
9. smoothTWIST: smooths twist with subroutine BEZIER,
10. improveTWIST: (1) tries to get locally as close as possible to 1/3 by adaption of twist, see Eq. 9. (2) J.N. Sørensen's approach, (3) a *constant circulation* approach

3.6 Functions

1. FNOVERK: (*Real Function N Over K*) calculates Binomial coefficients (BC) needed in BEZIER, (special numerical care has to be take because of strong growth of BCs),
2. FPR: Prandtl's tip loss F, see [22],
3. FBu: Burton's tip loss, see [19, 2],
4. chDES: optimum chord distribution from Betz (or Schmitz, or Glauert) [25],
5. chBe: Betz optimum chord, $a' \equiv 0$,
6. chSc: Schmitz optimum chord, a' is included,
7. TWIST: twist (in deg) from Betz, $a = 1/3$ and $a' = 0$, so only local TSR enters,
8. AHANSEN: gives Glauert's correction as stated in [6],
9. ANREL: gives Glauert's correction as stated in [13].

$$c_T(a) = \frac{8}{9} + \left(4F - \frac{40}{9}\right)a + \left(\frac{90}{9} - 4F - \right) \cdot a^2. \quad (5)$$

10. thickpoly: cubic interpolation of thickness
11. twistbend: simple table for inclusion of twist-bend
12. apot: (1) Betz, (2) J.N. Sørensen, (3) constant circulation (very approximate)

3.7 Compilation

3.7.1 LINUX

For compilation we prefer use (in a SUSE-LINUX environment): [14] with mandatory options (see **compDesCode.cmd**):

-fno-automatic helps to resolve notorious issues with local variables in SUBROUTINES and FUNCTIONS

-fbounds-check to avoid not allowed memory access

-O3 high optimization

A typical job with 100 section and 100×100 wind-speed and pitch variation takes a few minutes on our simple *Intel(R) Core(TM) i3-4160 CPU @ 3.60GHz* CPU.

3.7.2 Windows

minGW (= minimum gcc for Windows) was used to compile a windows executable file. It is also available

3.8 Modularization

Apart from mem and KKS the following modules

- mem.f, declares global variables,
- Sub.f, a set of used subroutines and functions,
- SubNum.f, numerical functions: SPLINE and SPLINT from [18],

help to put some structure into the code.

3.9 Work Flow

All input has to be provided as ASCII files, as well as all output will be provided in the same format. Post-processing then has to be performed by other tools. We prefer GNUPLOT [16, 8] see section 4.2.

4 Output Files and Tools for Visualization

4.1 Description of Output Files

- *Bem.out*: sectional output from BEM, see table 5,
- *des.out*: sectional output from design: rsec, twist, chird, thick, c_L^{des} and AOA_{des} ,
- *ProProp.out*: from routine AIRFOILS: design properties of given airfoil data.

4.2 Description of GNUPLOT scripts

- *a.gpl*: plots $a(r)$. data from *Bem.out*
- *chord.gpl*: plots $c(r)$. data from *Bem.out* an *BlaDes.in* and *des.out*
- *cL-aoa.gpl* and *cL-cD.gpl*: plots polars. data from **.aer*
- *cL-des.gpl* plots $c_L^{des}(r)$. data from *PrOpt/Bem.out*
- *twist.gpl*: plots $\vartheta(r)$. data from *Bem.out* an *BlaDes.in* and *des.out*
- and some more

5 Mode: Analysis

5.1 Input data

The code is able to operate in two modes:

1. Design mode, set bit *DesMode* to .T. in file *Machine.in*,
2. if not, then the code will operate to analyze a given blade.

Then, in addition, other files:

- *BlaDes.in* (r , twist, chord, attached prfile name),
- *ProThick.in* which maps airfoil names to their (relative) thickness,
Important: This file has to have a header line
- *ThickDis.in* which gives with few entries r/R_{tip} and corresponding t/c ,
- **.aer*: files with airfoil data $AOA(\alpha)$, c_L , c_D , c_M (not used).

are needed. Most of the input (more or less) should be self-explaining. Lines with 3 numbers a are: first, last and number of values (wind speed and pitch).

Below examples are given,

Fig. (5) and Fig. (4) give an impression of what kind of thickness distribution have been and will be used.

Recently, [24] investigated a 15 MW reference blade in more detail. For reducing the mass from 71 to 63 to a considerable change of thickness (see Fig 6) was necessary. This underlines again the importance of this file.

Main output file is *Bem.out*

```

DesMode      .F.
BladeNo      3.0
dens         1.225
Rhub         0.3
Rtip         63.0
windrage     11.4 11.4 1
rpm          12.1
Pitchr       0.0 0.0 1
Nsec         70
. . . . .

```

Table 1: Part of a sample job description file *Machine.in* based on NREL’s 5MW reference wind turbine

```

r      Twist    chord    Profile Name
0.0000 15.5946  5.2000  CYL1
2.4000 15.5911  5.2083  CYL2
4.8000 15.4288  5.2357  FF50
7.2000 14.9891  5.2889  FF50
9.6000 14.3263  5.3606  FF50
12.0000 13.4947  5.4433  FF50
15.0000 12.3001  5.5510  FF50
18.0000 11.0323  5.6501  FF50
20.2841 10.0834  5.7110  FF36
. . . . .
118.7179 -1.3914  1.7239  FF21
118.9744 -1.3628  1.6717  FF21
119.2308 -1.3336  1.5813  FF21
119.4872 -1.3038  1.4157  FF21
119.7436 -1.2734  1.1027  FF21
120.0000 -1.2424  0.5000  FF21

```

Table 2: Sample blade description file *BlaDes.in* for NREL/IEAwind 15MW reference wind turbine

NAME	rel thickness
CYL1	1.0
CYL2	0.95
FF50	0.5
FF36	0.36
FF33	0.33
FF30	0.3
FF27	0.27
FF24	0.24
FF21	0.21

Table 3: Sample Thickness assignment file *ProThick.in* for NREL/IEAwind 15MW reference wind turbine

0.0000	1.0
0.0200	0.98
0.1500	0.500
0.2452	0.360
0.3288	0.330
0.4392	0.301
0.5377	0.270
0.6382	0.241
0.7717	0.211
1.0000	0.211

Table 4: Sample Thickness distribution file *ThickDis.in* for NREL/IEAwind 15MW reference wind turbine

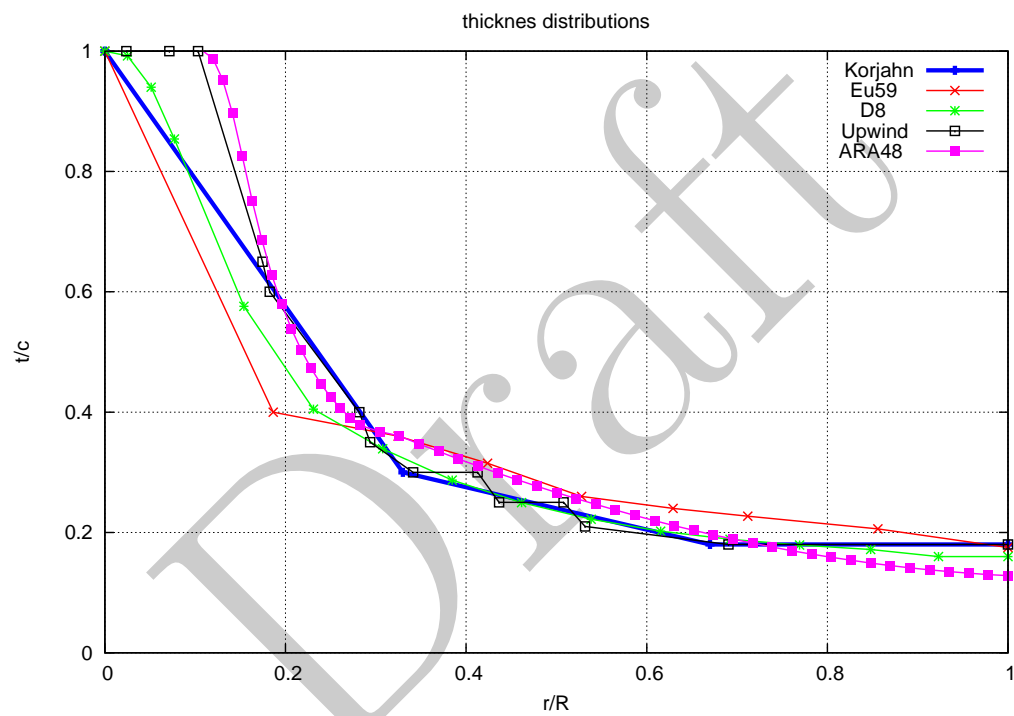


Figure 4: Sample thickness distributions

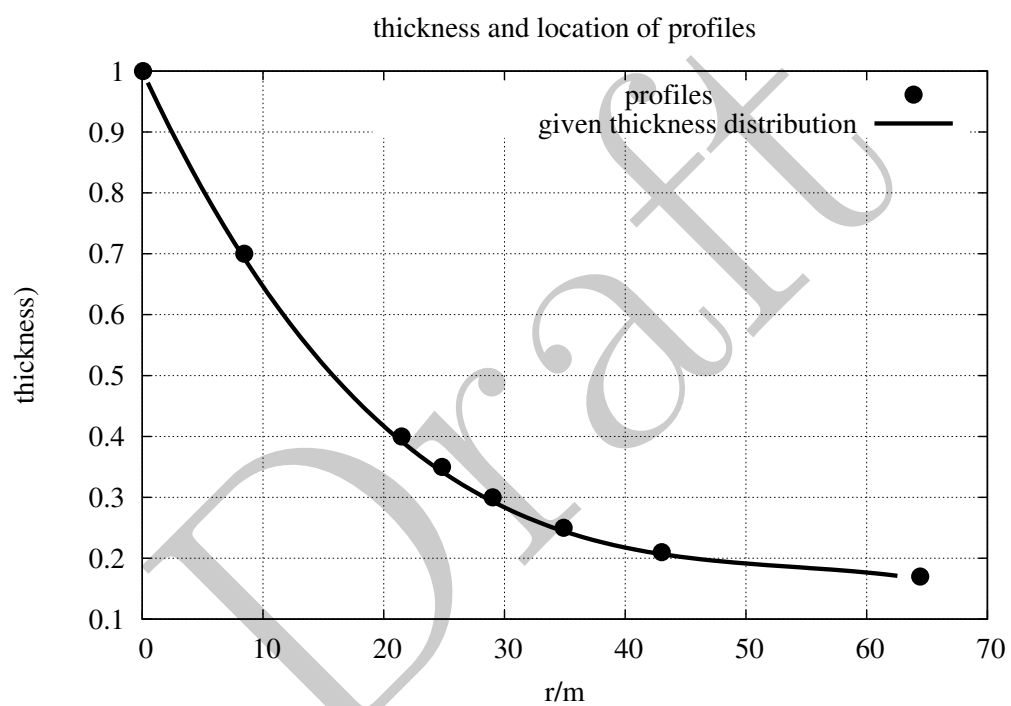


Figure 5: Refined thickness distribution from NREL 5 MW

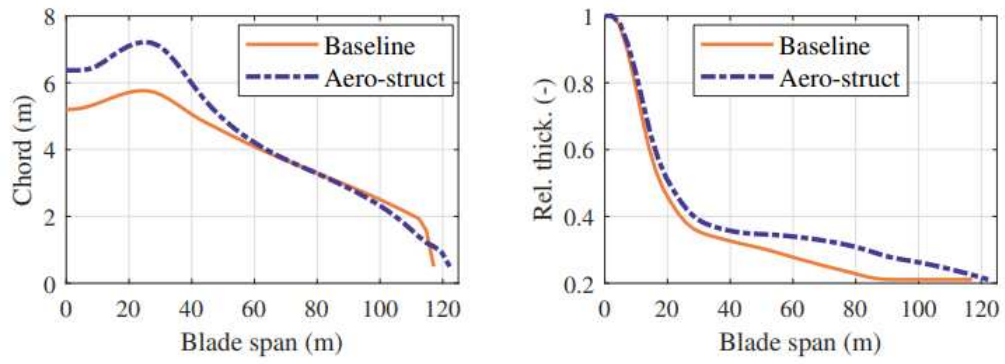


Figure 4: Chord and relative thickness distributions.

Figure 6: Thickness distribution change to decrease mass for a 15 MW reference blade [24]

r	a	aP	F	w	chord	twist	phi	aoa	cL	cD	L2D	cNo	cTa	dT	dQ	iter	err	th	Prof
0.598	0.334	-0.334	1.0	6.004	5.202	15.596	87.4	71.8	0.000	0.481	0.000	0.481	-0.022	0.20	-0.01	3	0.0E+00	0.99	CYL2
1.794	0.134	-0.134	1.0	7.861	5.205	15.598	82.3	66.7	0.000	0.444	0.000	0.440	-0.059	0.31	-0.04	3	0.0E+00	0.98	CYL2
2.990	0.076	-0.076	1.0	8.519	5.213	15.573	77.3	61.7	0.000	0.388	0.000	0.379	-0.085	0.31	-0.07	3	-0.2E-06	0.97	CYL2
4.186	0.045	-0.045	1.0	9.009	5.226	15.495	72.5	57.0	0.000	0.303	0.000	0.289	-0.091	0.27	-0.08	3	-0.2E-06	0.95	CYL2
5.382	0.040	-0.020	1.0	9.353	5.246	15.347	67.5	52.2	0.060	0.303	0.197	0.303	-0.061	0.31	-0.06	4	-0.1E-04	0.92	FF50
6.578	0.038	-0.002	1.0	9.730	5.273	15.128	62.8	47.7	0.144	0.297	0.484	0.330	-0.008	0.36	-0.01	4	-0.3E-04	0.89	FF50
7.774	0.040	0.011	1.0	10.149	5.305	14.848	58.4	43.5	0.245	0.283	0.864	0.370	0.060	0.44	0.07	5	-0.9E-07	0.85	FF50
8.970	0.043	0.021	1.0	10.608	5.340	14.519	54.2	39.7	0.356	0.263	1.354	0.421	0.135	0.56	0.18	6	0.0E+00	0.81	FF50
10.16	0.049	0.028	1.0	11.106	5.379	14.144	50.4	36.3	0.469	0.238	1.966	0.482	0.210	0.70	0.31	6	0.0E+00	0.76	FF50

Table 5: Sample output file *Bem.out* for NREL/IEA wind 15MW reference wind turbine

6 Mode: Design

If the design flag is switched on, the lower part of *Machine.in* is important and will be used. As chord from Betz and/or Schmitz usually will give non-zero chord at any sections, a tip-length must be given within its range chord goes down to zero. Two shapes can be chosen:

- Parabolic "P" and
- Elliptic "E".

The last one give a vertical tangent at the tip and the first one is a simple quadratic parabola defined by two points (begin and end (=rtip)) and slope of $c(r)$ at the begin of the tip region.

Note: *ImpChord* flag is not used at the moment. Improvement of chord (to proceed to more monotonic-like behavior, see [12] sheet 11) is achieved by smoothing with *Bezier* polynomials.

In addition to Korjahn's $a = 1/3$ approach $DesSchema = 1$, we use $DeSchema = (2,3)$ for other Schemas (with very low differences), as described in section 6.1.

6.1 Aerodynamic Optimization

6.1.1 Betz' and other criteria

We will not give a complete list or discuss in much detail the various interpretations of what an optimization might be and how this may be reached. Only the following three are implemented:

- Betz: $a(r) = 1/3 = const.$,
- Sørensen's (2022), modified *Glauert's* approach[9], see Fig. 7 and
- Joukovskie's: $\Gamma(r) = const.$

(1) Betz: no further explanation needed.

(2) Sørensen [9] argued that close to tip tip $a \rightarrow 2/5$, see Fig. 7.

(3) The last one is more subtle; to first order (in a) it is equivalent to Betz', but to 2nd order:

$$a(1 - a) = \frac{\lambda B \Gamma(1 + a')}{4\pi}. \quad (6)$$

To complete, a model for a' has to be added, for example

$$a' = \frac{1 - 3a}{4a - 1} \text{ (Glauert) or} \quad (7)$$

$$a' = \frac{2}{9} \frac{1}{\lambda_r^2} \text{ (Simple approximation) .} \quad (8)$$

$DesSchema = 1$ refers to Betz' aerodynamic optimized blade $a = 1/3$ for all spans.

.				
10.00000	-0.88209	0.01386	-0.04397	
-8.00000	-0.62981	0.01075	-0.05756	
-6.00000	-0.37670	0.00882	-0.06747	
-4.00000	-0.12177	0.00702	-0.07680	
-2.00000	0.12810	0.00663	-0.08283	
-1.00000	0.25192	0.00664	-0.08534	
0.00000	0.37535	0.00670	-0.08777	
1.00000	0.49828	0.00681	-0.09011	
2.00000	0.62052	0.00698	-0.09234	
3.00000	0.74200	0.00720	-0.09447	
4.00000	0.86238	0.00751	-0.09646	
5.00000	0.98114	0.00796	-0.09828	
6.00000	1.09662	0.00872	-0.09977	
7.00000	1.20904	0.00968	-0.10095	
8.00000	1.31680	0.01097	-0.10163	
9.00000	1.42209	0.01227	-0.10207	
.				

Table 6: Sample aerodynamic data *FF21.aer* for NREL/IEAwind 15MW reference wind turbine

.	
Tiplen	3.0
twmax	20.
chmax	5.5
ImpChord	.F.
ImpTwist	.F. 1
ImpThick	.F.
TwistB	.F.
DesSchema	2

Table 7: Parameters describing a design case

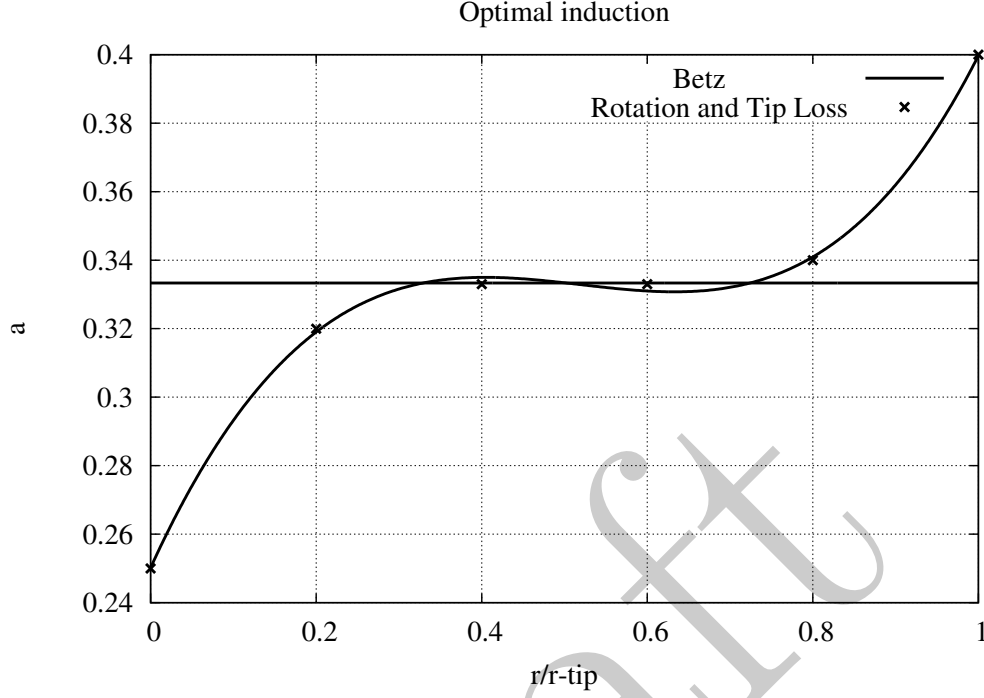


Figure 7: Betz and Sørensen

DesSchema = 2 refers to Sørensen's different $a(r)$ distribution, [9]. Fig. 7. See [10] for further discussion about aerodynamic optimization and the textbooks in the list of references, esp. [23].

DesSchema = 3 is implemented but seems not to give any significant changes. In any case we will only change twist for optimization.

6.1.2 Change of Twist

Our approach to relate twist and a runs via the flow angle and reads as: If $da := a_{opt}(r) - a_{loc} \neq 0$ then locally change twist by

$$d\varphi = -\frac{1}{\lambda_r(1 + a')}da . \quad (9)$$

6.1.3 Change of Thickness - Arrangement of Profiles

As stated in [12] (sheet 19, first figure) a further means to improve performance is to reduce thickness in the inner part. This is provided in our code by setting the *ImpThick*-bit and providing a changed thickness distribution *ThickDis.new*.

At the very end of *Machin.in* a flag *TwistB* (binary) can be set. It is a very simple way to take *pre-twist* into account. If set to .T. a table of additional twist deformation (along the blade) as function of wind speed has to be given.

7 Sample cases

7.1 Case 1: The NREL 5MW Baseline blade

As this turbine [11] was also used by Korjahn [12] as a sample case we here report first on output from KSS as well.

Table 8: Rated rotor power of NREL 5MW baseline from various sources. HAWC data from Mr. Hinrichs, aerodyn engineering GmbH, bladed from Mr. Maniok, DNV If all value are equally weighted $P = 5356 \pm 89$ kW results

Source	Value/kW	Deviation (%)
TP 38060	5267	0.0
KSS	5248	-0.4
FAST	5293	0.5
bladed	5322	1.0
HAWC2	5323	1.1
q-blade	5364	1.8
Xturb	5418	2.9
WZX	5478	4.0
wt-perf	5494	4.3

Except of (out-dated) *wt-perf* (and some other codes) accuracy is within a 1 % range. Difference of KKS to TP38060 is only 0.4 % This may serve as a hint that this BEM-implementation is not totally wrong.

To have a glimpse of the design capability Fig. 8 shows $a(r)$ after 5 loops of twist optimization. cP improves from 0.4682 to 0.4685. We agree with [12], sheet 16, that

For radius < 15 m optimum induction is not possible.

In addition, Fig. 9 shows some angles for further assessment.

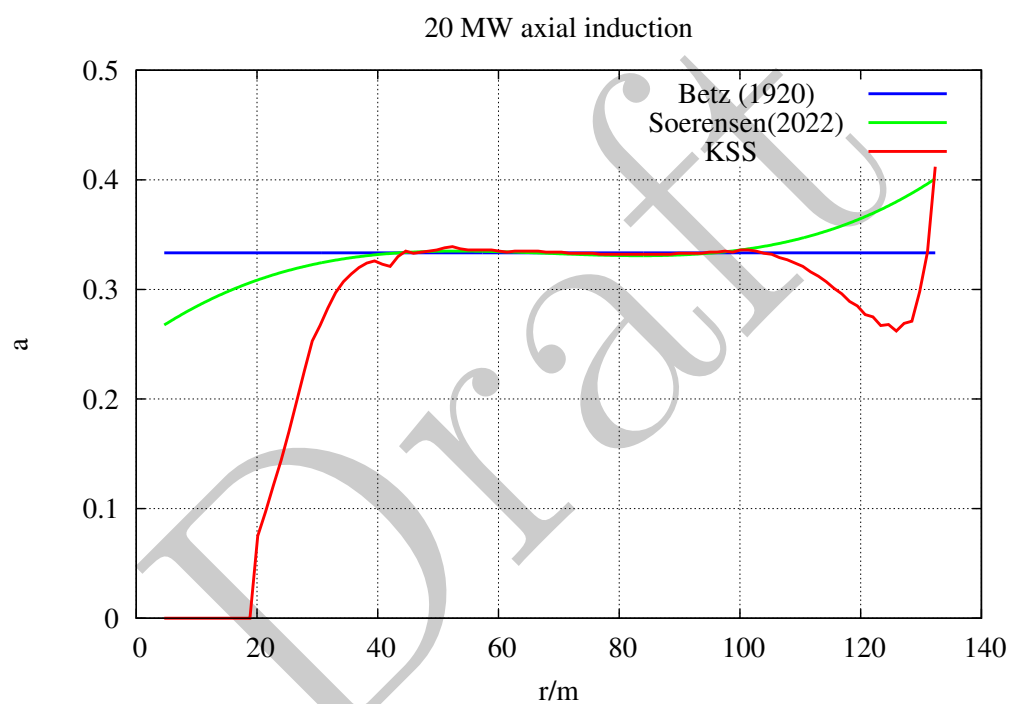


Figure 8: KSS Optimisation after 5 twist loops: $a(r)$

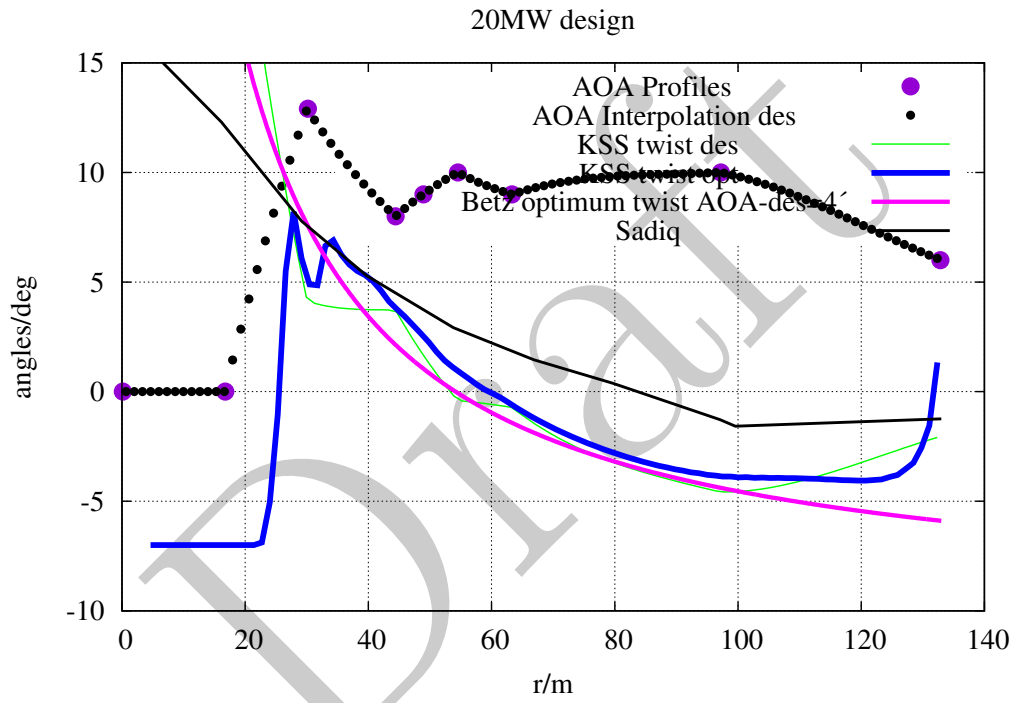


Figure 9: KSS Optimisation (Own 20MW, $r_{tip} = 133m$) after 5 twist loops: some angles

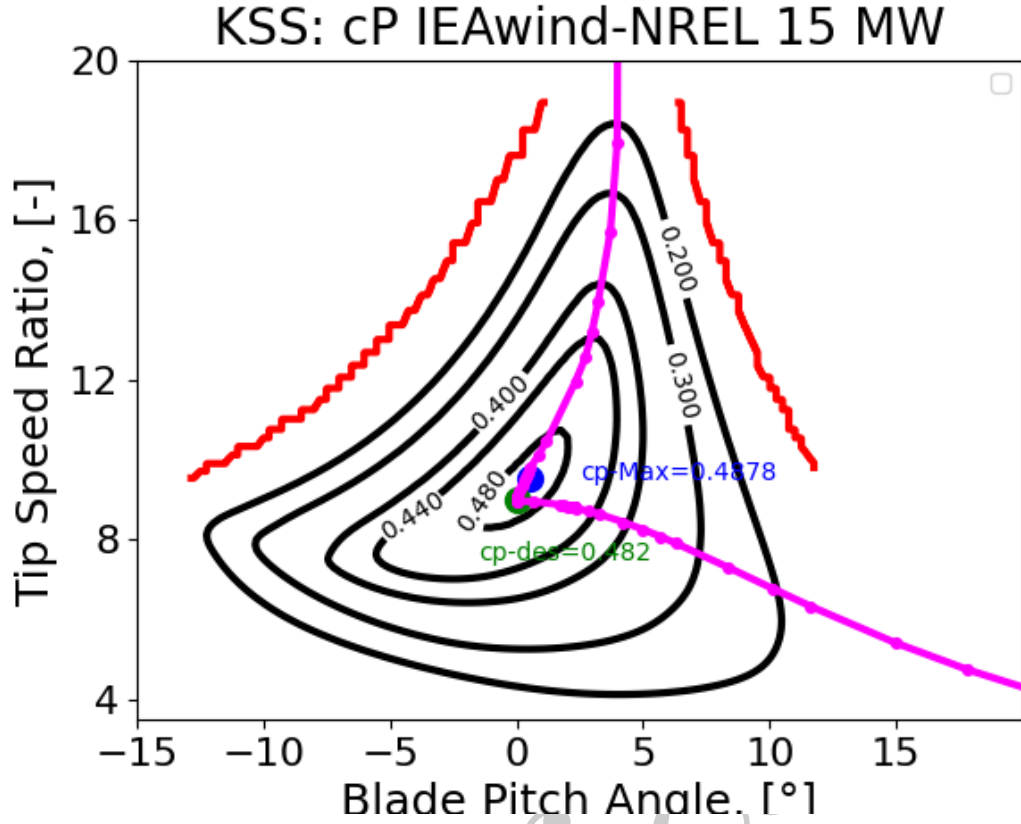


Figure 10: c_P map from KSS, including control path (magenta) and an extended pitch variation $-15^\circ < \vartheta < 20^\circ$.

7.2 Case 2: A 15MW WT reference blade

The blade described in [4] was also analyzed with KSS, see Figs. 10 and 11. Some difference are visible, but agreement at rated conditions, see table 9 (comparing c_P and c_T) are again within 1% range.

Table 9: Comparison for NREL/IEAwind 15 MW reference wind turbine

Source	c_P	c_T
TP 76698	0.489	0.804
KSS	0.484	0.768

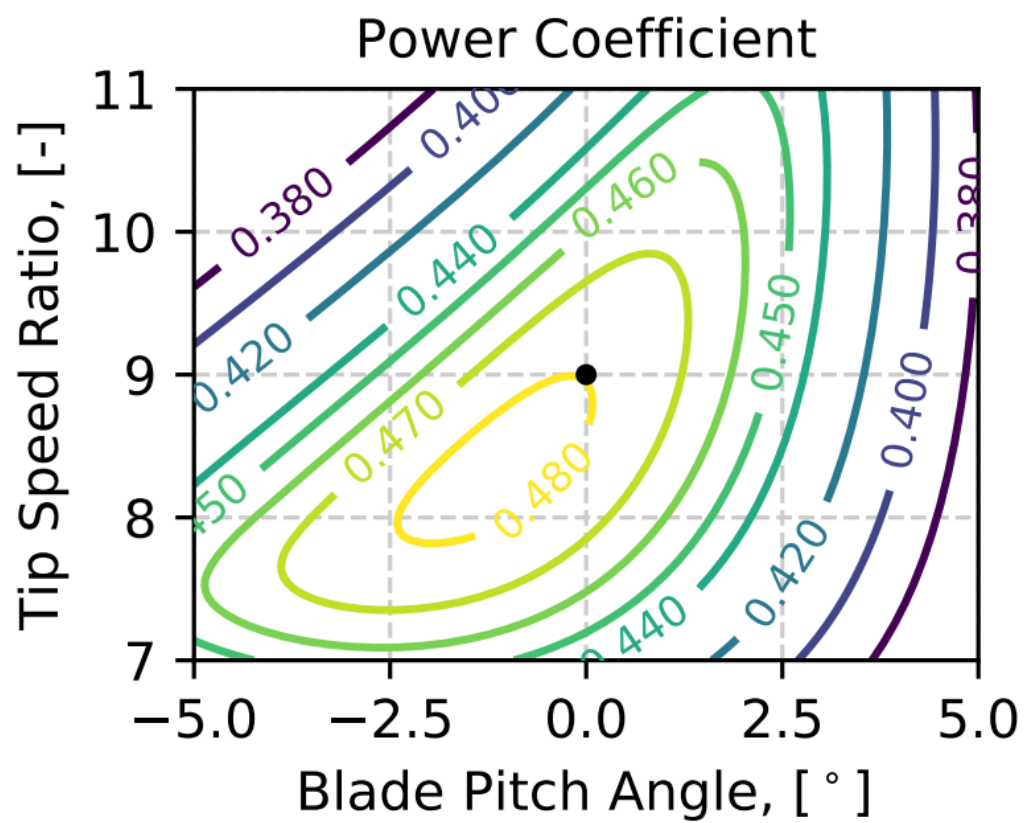


Figure 11: cP map from NREL [4]

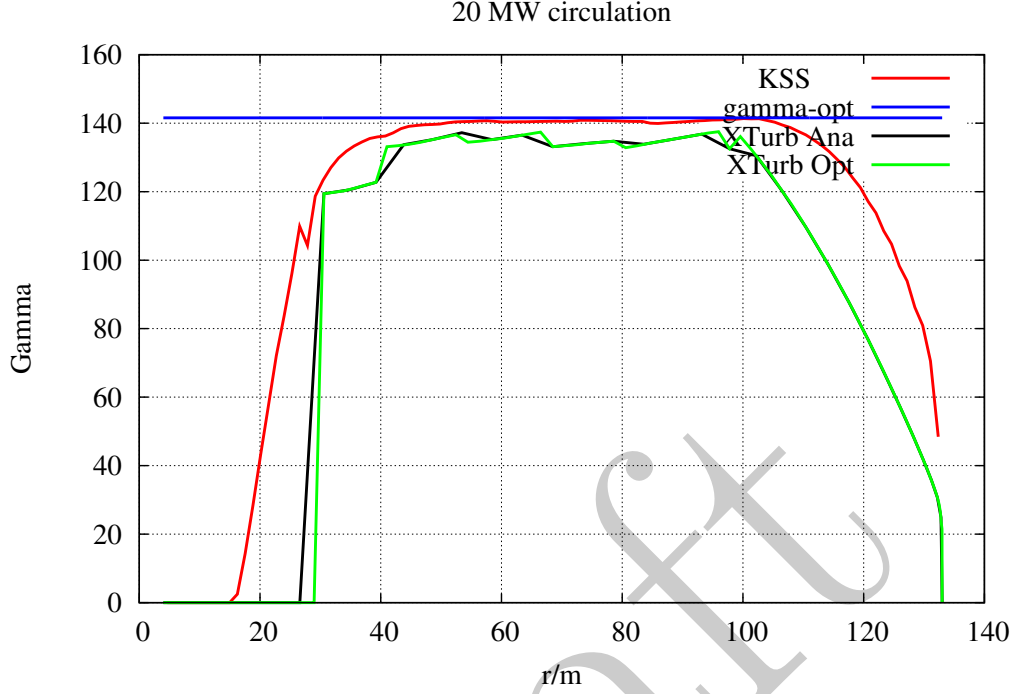


Figure 12: Comparison of circulation for an own 20 MW blade

7.3 Case 3: A 20 MW blade

[26] investigated the aerodynamic design of a 20MW blade in more detail. A maximum $cP = 0.4826$ was reached by Betz-mode of optimization after 6 loops and an initial design with $cP = 0.4821$. This is less than 1%. It seems that the restrictions are so strong that noticeable improvement is difficult.

8 Comparison with *Xturb*

Xturb by Sven Schmitz [23] offers some tools for optimization as well. We compared our 20 MW design [26], see Fig. 12. Unfortunately, XTurb showed an unexpected and non-explainable decrease close to tip ($r/R_{tip} > 0.77$) which makes a reasonable comparison impossible. Nevertheless, the initial design can be improved from $cP_{xT,A} = 0.4299$ to $cP_{xT,D} = 0.4347$ which is about 1.1% higher.

9 Summary of investigated blades

Table 10 shows a summary of investigated blades so far:

Table 10: Summary of investigated blades

Name	Rated Power	Rotor Diameter/m
SkyWind NG	0.6 - 1 kW	1.5
Baltic Thunder	$\sim 1 - 5$ kW	1
Braak	~ 10 kW	101
SWT	1 kW	1.5
SWT(WINDFLOH)	14 kW	10.3
NASA-AMES	15 kW	11
E30	150 kW	30
GROWIAN	1000 kW	100
NecMicon 80	2 MW	80
MM92	2 MW	92
NREL-Baseline	5 MW	126
DTU	10 MW	178
CIG10MW	10 MW	200
		220
NREL	15 MW	239
	17 MW	248
KUAS20	20 MW	284
22-RWT	22 MW	266

10 Summary, Discussion and Outlook

Based on the lecture notes from [12] a F90 blade analyze and design tool has been implemented using F90. It was applied and tested to a wide range of WT-blades, see table 10. Agreement with other sources/codes seems to be within a 1% range or smaller seems to be possible.

Findings:

- Choice of thickness distribution and polar data has probably the strongest influence on performance data for a wind turbine blade.
- Improvement (of c_P) by the proposed 1-point algorithm seems to be within a 1%-range only.

11 Disclaimer

As mentioned earlier, a set of **meaningful** input data has to be provided. There is almost no **consistency** check of input data and this implies that the user has to have some knowledge to avoid crashing of the code and/or meaningless output. Again and again we emphasize:

Theorem 1. *You can't calculate what you haven't understood.*

Theorem 2. *Only start the calculation, if you know the answer.*

12 List of Abbreviations

Table 11: list

	meaning
CIG	Chinese, India, Germany
DNV	Det Norske Veritas
DTU	(The) Danish Technical University
HAWC	Horizontal Axis Wind turbine simulation Code
KUAS	Kiel University of Applied Sciences
NREL	National renewable Energy Laboratory (USA)
SWT	small wind turbine

References

- [1] C. Bak. Aerodynamic design of wind turbine rotors. page chapter 3, 2013.
- [2] T. Burton, N. Jenkins, E. Bossanyi, D. Sharpe, and M. Graham. *Wind Energy Handbook, 3rd. Ed.* John Wiley and Sons, Hoboken, NJ, USA, 2020.
- [3] D.C Maniaci. An Investigation of WT-Perf Convergence Issues. In *49th AIAA Aerospace Sciences meeting*.
- [4] Gaertner, E. and Rinker, J. and Sethuraman L and Zahle, F. and Anderson, B. and Barter, G. and Abbas, N. and Meng, F. and Bortolotti, P. and Skrzypinski, W. and Scvott, G. and Feil, R. and Bredmose H. and Dykes, K. and Shileds, M and Alen, Chr. and Vislli, A. Definition of the IEA 15-Megawatt Offshore Reference Wind Turbine. NREL/TP-5000-75698, NREL, March 2020.
- [5] H. Glauert, editor. *Airplane Propellers*. Reprint: Stanford University, CA, USA, 1935.
- [6] M. Hansen. *Aerodynamics of Wind Turbines, 3rd. Ed.* Routledge, London, UK, 2015.
- [7] J. Ledoux and S. Rizzo and J. Salomon. Analysis of the Blade Element Momentum Theory. *SIAM Journal on Applied Mathematics*, (81(6)):2596–2621, 2021.
- [8] P. K. Jannert. *gnuplot IN ACTION, 2nd Ed.* Manning Publications, 2016.
- [9] J.N. Sørensen. Glauert revisited. In *Proceedings of EAWC conference The Science of Making Torque from the Wind*, June 2022.
- [10] J. Johansen, H. A. Madsen, M. Gaunaa, and C. Bak. Design of a wind turbine rotor for maximum aerodynamic efficiency. *WIND ENERGY*, 12:261 – 273, 2009.
- [11] J. Jonkman, S. Butterfield, W. Musial, and G. Scott. Definition of a 5-mw reference wind turbine for offshore system development. Technical Report NREL/TP-500-38060, NREL, Golden, Co, USA, 2009.
- [12] M. Korjan. Predesign rotor blade (full example). Technical Report unpublished, bewind GmbH, Rendsburg, Germany, 2021.
- [13] M. L. Buhl, Jr. A New Empirical Relationship between Thrust Coefficient and Induction Factor for the Turbulent Windmill State. NREL/TP-500-36834, NREL.
- [14] NN. <https://gcc.gnu.org/fortran/>.

- [15] NN. de.wikipedia.org/wiki/B{\unhbox\voidb@x\bgroup\let\unhbox\voidb@x\setbox\@tempboxa\hbox{e\global\mathchardef\accent@spacefactor\spacefactor}\accent18e\egroup\spacefactor\accent@spacefactor}zierkurve.
- [16] NN. www.gnuplot.info.
- [17] O. de Vries. Fluid Dynamic Aspects of Wind Energy Conversion. Technical report, AGARDograph Bo. 242, 1979.
- [18] W. Press, A. Teukolsky, W. Vetterling, and B. Flannery. *NUMERICAL RECIPES in Fortran, 2nd Ed. The Art of Scientific Computing*. Cambridge University Press, Cambridge, UK, 1992.
- [19] Ramdin, S.F. Prandtl tip loss factor assessed. Master’s thesis, Delft University of Technology, 2017.
- [20] R.E. Wilson and P.B.S Lissaman and S.N. Walker. Aerodynamic Performance of Wind Turbines. Technical report, Oregon State University, 1976.
- [21] S.A. Ning. A simple solution method for the blade element momentum equations with guaranteed convergence. *Wind Energy*, (17):1327–1345, 2014.
- [22] Schaffarczyk, A.P. *Introduction to Wind Turbine Aerodynamics, 3rd Ed.* SpringerNature, Berlin, 2024.
- [23] S. Schmitz. *Aerodynamics of Wind Turbines*. Wiley, 2020.
- [24] Scott, Samuel and Greaves, Peter and Macquart, Terence and Pirrera, Alberto. Comparison of blade optimization strategies for the IEA 15 MW reference wind turbine. In *Proceedings of EAWC conference The Science of Making Torque from the Wind*, June 2022.
- [25] J. N. Sørensen. *General Momentum Theory for Horizintal Axis Wind Turbines*. Springer International Publishing Switzerland, 2016.
- [26] Syed, S. Aerodynamic Design of a 20MW blade. Master’s thesis, Kiel University of Applied Sciences, 2022.