



NTNU – Trondheim
Norwegian University of
Science and Technology

Wake Modelling using an actuator disk model in openFOAM

Anne Mette I Nodeland

Master of Energy and Environmental Engineering

Submission date: June 2013

Supervisor: Lars Sætran, EPT

Norwegian University of Science and Technology
Department of Energy and Process Engineering

EPT-M-2013-85

MASTER THESIS

for

Stud.tech. Anne Mette Nodeland

Spring 2013

Wake Modelling using an actuator disk model in openFOAM***Modellering av turbin-vake med en aktuator-disk modell i openFOAM*****Background and objective**

In wind parks design, basic analytical models that rely on the typical self-similarity nature of the wakes are the standard in the industry. However, recently, more accurate models where either actuator disk or actuator line codes to represent the rotor coupled with RANS or LES (Large Eddy Simulations) methods for the wake are starting to be used. In these types of simulation the rotor is simplified as either a disk or series of lifting lines. Drag and lift curves for the air foils in the blade are used to calculate the thrust and thus the velocity deficit in the wake. The wake is then resolved with a volume mesh using either LES or RANS methods.

The simultaneous rapid development and improvement of open source CFD codes like OpenFOAM, where the source code is available then modifiable seems to indicate that the trend for the near future in wake simulations leads towards implementations of either actuator disks or actuator lines in openFOAM.

Simple actuator disk and actuator line models are currently freely available within the openFOAM community and they are a good starting point for wake simulations. In the present thesis an actuator disk model with a RANS approach implemented in openFOAM will be used.

The following tasks are to be considered

The thesis will be divided in two parts. In the first part the actuator disk model will be validated towards the experiments carried out at NTNU during the so-called Blind Tests (1 and 2).

In the second part, a complete wind farm (Vindeby) will be modelled and the results will be compared both with experimental results and with previous computations.

-- ” --

Within 14 days of receiving the written text on the master thesis, the candidate shall submit a research plan for his project to the department.

When the thesis is evaluated, emphasis is put on processing of the results, and that they are presented in tabular and/or graphic form in a clear manner, and that they are analyzed carefully.

The thesis should be formulated as a research report with summary both in English and Norwegian, conclusion, literature references, table of contents etc. During the preparation of the text, the candidate should make an effort to produce a well-structured and easily readable report. In order to ease the evaluation of the thesis, it is important that the cross-references are correct. In the making of the report, strong emphasis should be placed on both a thorough discussion of the results and an orderly presentation.

The candidate is requested to initiate and keep close contact with his/her academic supervisor(s) throughout the working period. The candidate must follow the rules and regulations of NTNU as well as passive directions given by the Department of Energy and Process Engineering.

Risk assessment of the candidate's work shall be carried out according to the department's procedures. The risk assessment must be documented and included as part of the final report. Events related to the candidate's work adversely affecting the health, safety or security, must be documented and included as part of the final report. If the documentation on risk assessment represents a large number of pages, the full version is to be submitted electronically to the supervisor and an excerpt is included in the report.

Pursuant to "Regulations concerning the supplementary provisions to the technology study program/Master of Science" at NTNU §20, the Department reserves the permission to utilize all the results and data for teaching and research purposes as well as in future publications.

The final report is to be submitted digitally in DAIM. An executive summary of the thesis including title, student's name, supervisor's name, year, department name, and NTNU's logo and name, shall be submitted to the department as a separate pdf file. Based on an agreement with the supervisor, the final report and other material and documents may be given to the supervisor in digital format.

- Work to be done in lab (Water power lab, Fluids engineering lab, Thermal engineering lab)
 Field work

Department of Energy and Process Engineering, 14. January 2013



Olav Bolland
Department Head



Lars Sætran
Academic Supervisor

Co-Supervisors: Dr Luca Oggiano, Siri Kalvig, Tor Anders Nygård

PREFACE

This thesis concludes my five year Master of Technology education in Energy and Environmental Engineering at NTNU in Trondheim. The thesis was performed during spring 2013 and after 20 weeks and 300,000 processor hours, you finally hold the finished product in your hands.

The thesis was created in collaboration with the wind department at IFE, Institute for Energy technology. I want to thank my main supervisor at IFE, Roy Stenbro, who did his best to answer all the questions I had, and to provide me with council on my ideas and problems. I am also grateful to Luca Oggiano from IFE who helped me organize the cooperation. In addition, I would like to thank Tor Anders Nygård who came with helpful knowledge within CFD.

I want to extend my gratitude to Siri Kvalvig from the University of Stavanger and Eirik Manger at Acona Flow Technology who provided me with the model turbine set-up in OpenFOAM and took their time to answer my questions and helped me understand the actuator line model.

Tommy Fredriksen also worked with the actuator line code during his master thesis, and I would like to thank him for the many hours of discussions we had about the code and our results.

I would also like to thank my supervisor at NTNU Lars Sætran, for his assistance with my thesis and for providing me with many processing hours at the NTNU supercomputer.

Last but not least, I would like to thank my friend Anders who used his computer knowledge to help me improve the efficiency of my data processing, and my family and friends supporting me through five years of studying.

Anne Mette Nodeland

Anne Mette Nodeland, Trondheim 10.06.13

ABSTRACT

Two "Blind tests" have been performed at NTNU. Researchers were asked to send in results from a simulation of a model turbine in a wind tunnel to compare the different results with the measured values. There was a large spread in the simulation results, showing that additional testing and development needs to be made in order to increase the accuracy of the modelling methods.

This thesis uses the "Blind test 1" as the set-up for the numerical simulations, and the actuator line code created by NREL to model the wind turbine. The actuator line method divides each blade into actuator line elements and distributes the forces from the line elements onto the grid.

A parameter study has been performed using the actuator line code, and guidelines have been created describing how to use the code to achieve the best possible results. This thesis has pointed out a few current problems with the actuator line implementation, including a difficulty with achieving a grid independent solution. The actuator line code mostly overestimated both thrust and power compared to the experimental values, and the best results were found from the grid producing the minimum thrust and power values.

The numerical results have been compared to the "Blind test 1" and "Blind test 2" experimental values, including thrust, power and velocity deficit and turbulent kinetic energy in the wake behind the turbines. The hub and tower has been included in the numerical simulation, proving to have a large effect on the turbulent kinetic energy.

The conclusion is that by following the introduced guidelines, the method is able to predict the experimental results from both of the "Blind tests" in a good manner.

SAMMENDRAG

To blindtester har blitt utført ved NTNU. Forskere ble bedt om å sende inn resultater fra en simulering av en modellvindmølle i en vindtunnel, for å sammenligne de forskjellige resultatene med målte verdier. De innsendte resultatene hadde stor spredning, noe som indikerer at mer testing og utvikling må bli utført for å øke nøyaktigheten på simuleringsmetodene.

Denne studien tar utgangspunkt i "Blind test 1" for å lage det numeriske oppsettet, og "actuator line"-koden laget av NREL har blitt brukt til å simulere vindturbinen. "Actuator line"-koden deler hvert blad inn i en rekke "actuator line"-elementer, for så å distribuere kreftene fra elementene til cellene i det numeriske nettverket rundt.

Det har blitt gjennomført en parameterstudie ved å bruke "actuator line" koden, og retningslinjer for hvordan koden kan brukes for å oppnå best mulig resultat har blitt laget. Denne studien har oppdaget et par problemer med den nåværende implementasjonen av "actuator line"-koden, inkludert at det er vanskelig å oppnå en løsning som er uavhengig av det numeriske nettverket. "Actuator line"-koden overestimerer for det meste både skyvekraft og effekt sammenlignet med de eksperimentelle verdiene, og de beste resultatene ble funnet ved å velge det numeriske nettverket som gav minimum skyvekraft og effekt.

De numeriske resultatene har blitt sammenlignet med de eksperimentelle verdiene fra "Blind test 1" og "Blind test 2", inkludert skyvekraft, effekt og hastighet og turbulent kinetisk energi i vaken bak turbinene. Nav og tårn har blitt inkludert i den numeriske simuleringen, noe som viste seg å ha stor effekt på turbulent kinetisk energi.

Konklusjonen er at ved å følge de introduserte retningslinjene, klarer koden å predikere de eksperimentelle resultatene på en god måte.

LIST OF SYMBOLS

ROMAN LETTERS

A	area	$[m^2]$
a	coefficient in the finite volume discretized equations	[-]
B	constant in the law of the wall	[-]
b	number of blade elements	[-]
c	chord length	$[m]$
C_D	drag force coefficient	[-]
$C_\mu, C_{1\epsilon}, C_{2\epsilon}$	constants in the $k - \epsilon$ model	[-]
C_P	power coefficient	[-]
C_T	thrust coefficient	[-]
D	diffusion conduction	$[kg/s]$
d	distance	$[m]$
e	unit vector	[-]
F	convection mass flux	$[kg/s]$
\mathbf{F}, \mathbf{f}	force vector	$[kgm/s^2]$
g	gravity vector	$[m/s^2]$
i, j	tensor indices	[-]
k	turbulent kinetic energy	$[m^2/s^2]$
L	characteristic length	$[m]$
m	mass	$[kg]$
n	outer normal vector	[-]
p, P	pressure	$[N/s^2]$
P	power	$[W]$
r	radius	$[m]$
S_Φ	source term	[-]
S_r	non-uniform part of the source term	[-]
S_u	uniform part of the source term	[-]
\mathbf{u}, u, U, v, w	velocity	$[m/s]$
\hat{u}, \hat{v}	pseudo velocities in the <i>SIMPLER</i> algorithm	$[m/s]$
T	thrust	$[N]$

$T1$	upstream turbine	[-]
$T2$	downstream turbine	[-]
t	time	[s]
u_τ	friction velocity	[m/s]
V	volume	[m ³]
x, y, z	Cartesian coordinates	[m]

GREEK LETTERS

α	angle of attack	[-]
α_p	under-relaxation factor for the pressure	[-]
$\Delta x, \Delta y, \Delta z$	grid spacings	[m]
δ_{ij}	Kronecker delta	[-]
$\partial\Omega$	boundary	[m ²]
ϵ	rate of viscous dissipation of k	[m ² /s ³]
ϵ	Gaussian width parameter	[m]
η_ϵ	regulation kernel	[-]
Γ	diffusion coefficient	[kg/sm]
κ	von Karmann's constant	[-]
λ	local pitch angle	[-]
μ	viscosity	[kg/sm]
μ_t	turbulent viscosity	[kg/sm]
ν	kinematic viscosity	[m ² /s]
ρ	density	[kg/m ³]
σ	Prandtl number	[-]
τ_w	wall shear stress	[kg/ms ²]
Φ	conserved variable	[-]
φ	angle between relative velocity and rotor plane	[-]
Ω	control volume	[m ³]
ω	angular rotor velocity	[rad/s]

SUPERSCRIPT

+	dimensionless
'	fluctuating value
*	preliminary variables
**	once corrected velocity components
1	at the actuator disk
2	at a distance after the actuator disk
<i>c</i>	correction factor
superscript <i>cc</i>	double correction factor in the <i>PISO</i> algorithm
<i>in</i>	at the entry of the channel

SUBSCRIPT

<i>axial</i>	axial component
<i>blade</i>	on the blade
<i>E, W, N, S</i>	eastern, western, northern, southern nodes of control volume
<i>e, w, n, s</i>	eastern, western, northern, southern faces of control volume
<i>max</i>	maximum value
<i>min</i>	minimum value
<i>nb</i>	neighbours
<i>R</i>	center node of control volume
<i>rel</i>	relative
<i>rot</i>	rotational
<i>u, R</i>	center node of control volume with <i>u</i> in the mid point

ABBREVIATIONS

<i>BT1</i>	"Blind test 1"
<i>BT2</i>	"Blind test 2"
<i>CFD</i>	computational fluid dynamics
<i>DNS</i>	direct numerical simulation
<i>FVM</i>	finite volume method
<i>LES</i>	large eddy simulation

<i>NREL</i>	national renewable energy laboratory
<i>NTNU</i>	Norwegian University of Science and Technology
<i>PISO</i>	pressure implicit with splitting of operators
<i>RANS</i> or <i>URANS</i>	unsteady Reynolds averaged Navier-Stokes
<i>SIMPLE</i>	semi-implicit method for pressure-linked equations
<i>SOWFA</i>	simulator for offshore wind farm applications
<i>Re</i>	Reynolds number
<i>TSR</i>	tip speed ratio

CONTENTS

1	Introduction	1
1.1	Prior works	3
2	Theory	5
2.1	Governing equations	5
2.2	Turbulent flow	5
2.2.1	Reynolds-averaged Navier-Stokes equations	7
2.2.2	Governing equations of the $k - \epsilon$ model	9
2.2.3	Wall boundary layer	11
2.3	Wind energy	14
2.3.1	Momentum theory	16
2.3.2	Actuator line	19
3	Numerical methods in OpenFOAM	23
3.1	Finite volume method	23
3.2	Grid	27
3.3	Boundary conditions	28
3.4	Solution algorithms	30
3.4.1	SIMPLE	30
3.4.2	PISO	32
3.4.3	Linear solver control	33
4	Numerical set-up	35
4.1	Wind tunnel and model turbine	35
4.2	Boundary conditions	38
4.3	Airfoil data	38
4.4	Mesh and OpenFOAM applications	40
5	Results and Discussion	45
5.1	A visualization of ϵ	46
5.2	Parameter study	52
5.2.1	Part 1 - Identifying parameters and their relationship	56

5.2.2	Part 2 - ϵ chosen from grid resolution	66
5.2.3	Part 3 - ϵ chosen from chord length	71
5.2.4	Part 4 - Proposal for best use of the actuator line model	81
5.3	Blind test 1	83
5.4	Blind test 2	100
6	Further work	111
7	Conclusion	113
A	A closer look at the grid refinement	119
A.1	Thrust over time for $\epsilon = 2.5\Delta x$	119
A.2	Thrust over time for $\epsilon = 0.0084m$	122
B	Noise in plots	126
C	Thrust and power graphs for $\epsilon = 0.0084m$ and $\Delta x = 0.023m$	133

1 INTRODUCTION

In 2011 NOWITECH/NORCOWE organized a "Blind test 1" (BT1) work shop on wind turbine technology where researchers and people from the industry were invited to perform simulations of a wind turbine in order to investigate the current accuracy of existing models. A well tested wind turbine in a wind tunnel at NTNU was used as the model set-up, and results were submitted from nine different groups. The results included thrust, power and wake profiles behind the turbine [8]. The results from this experiment were very spread and a logarithmic scale was needed to get all the results in one plot for the turbulent kinetic energy. This showed that additional work needs to be done to improve the accuracy of wind turbine calculations. A similar blind test called "Blind test 2" (BT2) was arranged in 2012, where the difficulty was slightly increased by having two turbines in the tunnel, 3 diameters apart.

Several methods for wind turbine performance and wakes modelling exists. The models range from very complex direct numerical simulation (DNS) calculations, to a more simple approach like an actuator disk or line model, or the Blade Element Momentum (BEM) method. At the current state of computer resources, full DNS calculations for a large wind turbine or wind park is very costly, and the simpler models are preferred. In order to utilize these models, it is important to study their performance and accuracy.

Originally, this thesis was suppose to focus on the actuator disk method, and validate it against the "Blind test 1" and "Blind test 2" experiments, followed by a simulation of a complete wind park. In cooperation with both my supervisor at NTNU and at IFE, the focus was soon shifted towards the actuator line model.

The actuator line model requires a lot of input variables, and many of these input parameters depend on each other. Currently no clear guidelines on how to determine these parameters exist. The first main goal of this thesis is to identify the most important parameters, how they are related to each other and how they affect the flow. The result will be specific guidelines on how to best utilize the actuator line model. The second main goal is

to evaluate the NREL SOWFA actuator line model for the purpose of the wind tunnel and model turbine at NTNU using the developed guidelines. This will be done by comparing the simulated thrust, power and wake development to measurements from the "Blind test 1" and "Blind test 2".

This thesis starts with an introduction to the relevant theory. Turbulent flow in a channel can be modelled in different ways. In this study the Reynolds averaged Navier-Stokes (RANS) equations are used for the governing equations, and the turbulence is modelled using the $k - \epsilon$ model. These concepts will be presented in theory followed by an introduction to wind turbine theory.

In the next section, an introduction to numerical methods, including the finite volume method and different numerical solvers, will be given. After this, the numerical set-up for the simulations will be presented, such as grid and wind tunnel description. Thereafter a parameter study will be performed, and guidelines for the use of the actuator line code will be developed. The guidelines will be used to simulate the "Blind test 1" and "Blind test 2" set-up, and compared to the experimental values. The most important conclusions drawn in this study will be presented last, followed by suggestions for further work.

The open source computational fluid dynamics (CFD) tool OpenFOAM version 2.1.1 is used for the simulations in this study. The library offers a complete framework for a Finite Volume discretization including the handling of the mesh, preprocessing, a variety of available solvers, and post-processing. OpenFOAM is fully parallelized based on MPI and scales up well to very large numbers of processors. This tool is becoming more and more popular, due to its good performance and because many alternative CFD tools are very costly. However, while the code provides documentation in itself, the commenting in the code is limited, and an external documentation for most parts of the code does not exist. In addition, the user friendliness of the program is low, and beginners are confronted with a steep learning curve.

1.1 PRIOR WORKS

This project focuses on an actuator line model. The model was first introduced in 2002 by Sørensen and Shen [6]. The body forces from the blade acting on the wind are assigned to three rotating lines and distributed from these lines onto the surrounding mesh. The wind flow is calculated using the Navier-Stokes equations.

In 2003 Mikkelsen [12] used the actuator line model and the actuator disk model to discuss the accuracy of assumptions in the BEM method. He performed numerical experiments for coned rotors, yawed inflow and tunnel blockage. He also rewrote the actuator line model from the original vorticity-velocity formulation by Sørensen and Shen into pressure-velocity variables, and used a solver created by Sørensen and Michelsen called the *EllipSys3D* solver. Mikkelsen et al. used the actuator line model on isolated turbines to study a turbine in an atmospheric boundary layer [13] and on a row of turbines to study the wake interaction for multiple turbines [14].

Ivanell [5] applied the actuator line model and the *EllipSys3D* solver to investigate the wake behind the rotor, with special consideration to circulations close to the turbine. Ivanell et al. [4] investigated the stability of tip and root vortices using the actuator line model.

Troldborg et al. [26] used the actuator line model to study the effect of inflow turbulence on wake characteristics and Troldborg [25] utilized the actuator line model in his thesis to study the turbine wakes for different flow conditions, using the *EllipSys3D* solver and the LES turbulence model.

Churchfield et al. [3] performed LES simulations of the Lillgrund wind park in Sweden using the actuator line model. Churchfield et al. [2] has made an OpenFOAM implementation of the actuator line model that will be utilized in this study.

Martinez [11] used the OpenFOAM code created by Churchfield to compare the actuator disk and line models and develop recommendations for the use of the actuator line model. He suggested relationships between some of the parameters in the actuator line model, and he concluded that the important

parameter ϵ should be linked to the grid spacing in the streamwise direction. ϵ governs how far and with which value the forces from the blade elements are spread out onto the surrounding grid. Martinez also performed his calculations using the LES turbulence model.

Shives and Crawford [22] performed a parameter study on the actuator line model on an infinite span wing, a constant circulation wing and an elliptically loaded wing, arguing that the parameter ϵ should be chosen from a physical turbine length, i.e. the chord length. The CFD tool ANSYS CFX was used, with RANS equations and the $k - \omega$ SST turbulence model.

Schmitz et al. [19] has investigated the accuracy of the actuator line, with special interest in ϵ . They looked at basing ϵ on either the grid resolution, the chord length or using an elliptic model.

As can be seen above, most of the simulations performed using the actuator line model, combined it with a LES turbulence model. It is interesting to see how the model functions with an unsteady RANS calculation, which is computationally less heavy and thus more suitable for an industrial purpose. The actuator line model will in this study be used with RANS equations and the $k - \epsilon$ turbulence model. This was chosen because it is widely used in the industry, and the $k - \epsilon$ model is well tested and documented. Simulations will be performed with ϵ chosen from the grid spacing and from the chord length, to see which one produces the best results.

2 THEORY

This section will start by introducing the governing equations for fluid flow, followed by basic theory for wind energy and the actuator line model.

2.1 GOVERNING EQUATIONS

The main governing equations for all fluid flows are the Navier-Stokes equations. The incompressible Navier-Stokes equations for a 3 dimensional Newtonian fluid consist of the continuity equation,

$$\nabla \cdot \mathbf{u} = 0 \quad (1)$$

and the momentum equations,

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = -\frac{1}{\rho} \nabla p + \nu \nabla^2 \mathbf{u} + \mathbf{f} \quad (2)$$

where \mathbf{u} is the velocity vector, p is the pressure, ρ is the density and ν is the kinematic viscosity, the latter is considered to be a constant [29].

2.2 TURBULENT FLOW

An important variables characterizing the flow is the Reynolds number, a dimensionless number correlating the inertial forces and the viscous forces as

$$Re = \frac{UL}{\nu}, \quad (3)$$

where U denotes the mean velocity in the flow and L is a characteristic length scale [29].

Turbulence occurs at high Reynolds numbers, i.e. where the inertial forces are dominant. The flow is characterized by random fluctuations, e.g. eddies and vortices [29]. Turbulent flow is complex and therefore difficult to model.

In a turbulent flow, eddies of several sizes exist. The largest eddies have the same characteristic scales as the mean flow, while the smallest eddies are described by the Kolmogorov time scale; the fastest fluctuations in the flow, Kolmogorov length scale; the length scale for energy dissipation and the Kolmogorov velocity scale; the velocity of the smallest eddies. In 1922,

Lewis F. Richardson discovered that energy is transported from the larger eddies to the smaller, and the smallest eddies are dissipated into heat due to the viscosity. He summarized this turbulent energy cascade with a poem;

*Big whorls have little whorls
That feed on their velocity,
And little whorls have lesser whorls
And so on to viscosity.*

– Lewis F. Richardson (1922)

Turbulent flow can be simulated using different methods, and these methods can be grouped into three main categories [27]:

Direct numerical simulation (DNS)

The Navier-Stokes equations are solved directly including all turbulent velocity fluctuations on a grid fine enough to resolve the Kolmogorov length scale, and with a time step smaller than the Kolmogorov time scale.

Large eddy simulation (LES)

Solves a rewritten form of the Navier-Stokes equations which included the larger eddies, but rejects the small eddies. The energy transported to the small eddies is taken into account by means of a so-called sub-grid scale model.

Reynolds-averaged Navier-Stokes equations (RANS)

The main focus is on the mean value of the flow, and how the turbulent fluctuations affect the mean value. Turbulence models are used to model terms that appear in the equations due to interaction between the fluctuating variables. This is the most used simulation method used in the industry, because it is widely tested and the least computational heavy of the three methods. RANS equations will be used for the simulations in this project, and a more thorough explanation of this method will be presented in section 2.2.1. Some authors like to distinguish between RANS and unsteady RANS (URANS), but no such separation will be used in this thesis.

2.2.1 REYNOLDS-AVERAGED NAVIER-STOKES EQUATIONS

Turbulent flow properties are often modelled as a time-averaged value Φ plus a fluctuation value $\phi'(t)$, so that $\phi(t) = \Phi + \phi'(t)$. This is called Reynolds decomposition and the time-average of the fluctuation value, $\overline{\phi'(t)}$, is by definition zero. The following decomposed variables for velocity and pressure, $\mathbf{u} = \mathbf{U} + \mathbf{u}'$ and $p = P + p'$, are inserted into the Navier-Stokes equations (eqn. (1)-(2)). The main focus is on the mean value of the flow, therefore a time average of the equations is taken [27].

Most of the fluctuating components disappear with the time averaging, but the convective terms become

$$\overline{\frac{\partial u^2}{\partial x}} = \frac{\partial(UU)}{\partial x} + \frac{\partial \overline{u'u'}}{\partial x}, \quad (4)$$

$$\overline{\frac{\partial uv}{\partial x}} = \frac{\partial(UV)}{\partial x} + \frac{\partial \overline{u'v'}}{\partial x}. \quad (5)$$

shown here for the x -direction. The last terms in equations (4) and (5) have the mean of two fluctuating quantities multiplied with each other, which is not zero. They are convective acceleration terms and measure the intensity of the turbulence [29]. The general names for these terms are Reynolds stresses, not due to the physical interpretation of the variables, but due to their mathematical function in the Navier-Stokes equations [27].

The Reynolds-averaged Navier-Stokes equations without external body forces become [27]

$$\nabla \cdot \mathbf{U} = 0, \quad (6)$$

$$\begin{aligned} \frac{\partial U}{\partial t} + U \frac{\partial U}{\partial x} + V \frac{\partial U}{\partial y} + W \frac{\partial U}{\partial z} = & -\frac{1}{\rho} \frac{\partial P}{\partial x} + \nu \left(\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} + \frac{\partial^2 U}{\partial z^2} \right) \\ & + \frac{1}{\rho} \left[\frac{\partial(-\rho \overline{u'^2})}{\partial x} + \frac{\partial(-\rho \overline{u'v'})}{\partial y} + \frac{\partial(-\rho \overline{u'w'})}{\partial z} \right], \end{aligned} \quad (7)$$

$$\begin{aligned} \frac{\partial V}{\partial t} + U \frac{\partial V}{\partial x} + V \frac{\partial V}{\partial y} + W \frac{\partial V}{\partial z} = -\frac{1}{\rho} \frac{\partial P}{\partial y} + \nu \left(\frac{\partial^2 V}{\partial x^2} + \frac{\partial^2 V}{\partial y^2} + \frac{\partial^2 V}{\partial z^2} \right) \\ + \frac{1}{\rho} \left[\frac{\partial(-\overline{\rho u'v'})}{\partial x} + \frac{\partial(-\overline{\rho v'^2})}{\partial y} + \frac{\partial(-\overline{\rho v'w'})}{\partial z} \right], \end{aligned} \quad (8)$$

$$\begin{aligned} \frac{\partial W}{\partial t} + U \frac{\partial W}{\partial x} + V \frac{\partial W}{\partial y} + W \frac{\partial W}{\partial z} = -\frac{1}{\rho} \frac{\partial P}{\partial z} + \nu \left(\frac{\partial^2 W}{\partial x^2} + \frac{\partial^2 W}{\partial y^2} + \frac{\partial^2 W}{\partial z^2} \right) \\ + \frac{1}{\rho} \left[\frac{\partial(-\overline{\rho u'w'})}{\partial x} + \frac{\partial(-\overline{\rho v'w'})}{\partial y} + \frac{\partial(-\overline{\rho w'^2})}{\partial z} \right], \end{aligned} \quad (9)$$

The difference between these equations and the original Navier-Stokes equations is that besides the viscous shear, Reynolds stresses appear in the equations to account for the turbulent shear (the three last terms of equations (7), (8) and (9)). The governing equations consist of only 4 equations, but the Reynolds stresses introduce 6 additional unknowns. This is called the *closure problem* of the RANS equations.

The Reynolds stresses and the eddy viscosity need to be modelled, and several turbulence models have been developed. They vary in accuracy and in computational power. The main variable influencing the complexity of the model is how many additional transport equations that needs to be solved. A few examples of turbulence models include

The mixing length model

No additional transport equations are solved, but an algebraic expression for the eddy viscosity and the Reynolds stresses is developed. The kinematic turbulent viscosity can be expressed as a product of a turbulent length scale and a turbulent velocity scale. This is used to link the Reynolds stresses to a length scale and the mean velocity gradient. This model is easy to implement, well established, gives good results for thin shear layers, but is unable to describe separation and recirculation and only calculates mean properties. It is the simplest of the classical turbulence models [27].

The Reynolds stress model

Seven transport equations are solved, one for each of the Reynolds stresses, plus one for the dissipation rate of turbulent kinetic energy. The Reynolds stresses, together with the the dissipation rate of kinetic energy, are used to express the turbulent kinetic energy, ν_t . This is the most general classical turbulence model and it gives very accurate calculations of the mean properties of the flow in addition to all of the Reynolds stresses. The disadvantages are that the model requires large computational power for the seven partial differential equations, it is not as well validated as some of the other models and it does not perform better than the $k - \epsilon$ model for a few flow problems such as asymmetric jet [27].

The $k - \epsilon$ model

In this model two additional transport equations are solved, one for the turbulent kinetic energy and one for the dissipation of turbulent kinetic energy. These variables are used to calculate the kinematic turbulent viscosity and the Boussineq relationship is used to calculate the Reynolds stresses. This is the most widely used turbulence model, and it has therefore been thoroughly tested. It requires more computational power than the mixing length model, and it does not give good results for all types of flows such as unconfined, rotating, asymmetric jet flows and non-circular ducts [27]. The $k - \epsilon$ model is not a general or very accurate turbulence model and it has obvious limitations and disadvantages, but this is the most popular turbulence model in the industry mainly because it is so widely tested. Due to this, the $k - \epsilon$ model will be used for the turbulence calculations in this project, and it will be discussed in further detail in section 2.2.2.

2.2.2 GOVERNING EQUATIONS OF THE $k - \epsilon$ MODEL

The $k - \epsilon$ model solves one transport equation for the turbulent kinetic energy, and one for the rate of dissipation of the turbulent kinetic energy. These variables are used to give an approximation to the eddy viscosity and the Reynolds stresses [27].

2 THEORY

The turbulent kinetic energy is

$$k = \frac{1}{2} \left(\overline{u'^2} + \overline{v'^2} + \overline{w'^2} \right), \quad (10)$$

where u' , v' and w' are the fluctuating velocities in the x -, y - and z -direction respectively. The total kinetic energy is an addition of the mean kinetic energy and the turbulent kinetic energy

$$K + k = \frac{1}{2} (U^2 + V^2 + W^2) + \frac{1}{2} \left(\overline{u'^2} + \overline{v'^2} + \overline{w'^2} \right). \quad (11)$$

The rate of viscous dissipation of turbulent kinetic energy per unit mass is

$$\epsilon = 2\nu \overline{\frac{1}{2} \left(\frac{\partial u'_i}{\partial x_j} + \frac{\partial u'_j}{\partial x_i} \right) \cdot \frac{1}{2} \left(\frac{\partial u'_i}{\partial x_j} + \frac{\partial u'_j}{\partial x_i} \right)}, \quad (12)$$

which indicates how fast energy is dissipated from larger eddies to smaller eddies [27].

The eddy viscosity is defined as

$$\mu_t = \rho C_\mu \frac{k^2}{\epsilon}, \quad (13)$$

where C_μ is a dimensionless constant, and the Reynolds stresses are computed using the Boussinesq approximation (1877)

$$-\overline{\rho u'_i u'_j} = \mu_t \left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right) - \frac{2}{3} \rho k \delta_{ij}, \quad (14)$$

which is created by treating the Reynolds stresses in a similar way as the viscous stress is treated in a Newtonian fluid, by relating the Reynolds stresses to the mean flow [27].

Model transport equations for both k and ϵ have been made, containing the most important processes influencing these variables. For high Reynolds

numbers where $\mu_t \gg \mu$, the model equation for k is

$$\begin{aligned}
 \underbrace{\frac{\partial}{\partial t}(\rho k)}_{\text{Rate of change of } k} + \underbrace{\frac{\partial}{\partial x_i}(\rho k U_i)}_{\text{Convective}} &= \underbrace{\frac{\partial}{\partial x_j} \left[\left(\frac{\mu_t}{\sigma_k} \right) \frac{\partial k}{\partial x_j} \right]}_{\text{Diffusive}} \\
 &\quad \underbrace{\hspace{10em}}_{\text{transport of } k} \\
 + \underbrace{\frac{\mu_t}{2} \left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right) \cdot \left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right)}_{\text{Production}} &- \underbrace{\rho \epsilon}_{\text{Dissipation}} \\
 &\quad \underbrace{\hspace{10em}}_{\text{rate of } k}
 \end{aligned} \tag{15}$$

and likewise for ϵ

$$\begin{aligned}
 \underbrace{\frac{\partial}{\partial t}(\rho \epsilon)}_{\text{Rate of change of } \epsilon} + \underbrace{\frac{\partial}{\partial x_i}(\rho \epsilon U_i)}_{\text{Convective}} &= \underbrace{\frac{\partial}{\partial x_j} \left[\left(\frac{\mu_t}{\sigma_\epsilon} \right) \frac{\partial \epsilon}{\partial x_j} \right]}_{\text{Diffusive}} \\
 &\quad \underbrace{\hspace{10em}}_{\text{transport of } \epsilon} \\
 + C_{1\epsilon} \frac{\epsilon}{k} \frac{\mu_t}{2} \left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right) \cdot \left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right) &- \underbrace{C_{2\epsilon} \rho \frac{\epsilon^2}{k}}_{\text{Dissipation}} \\
 &\quad \underbrace{\hspace{10em}}_{\text{rate of } \epsilon}
 \end{aligned} \tag{16}$$

The constants in the equations are: $C_\mu = 0.09$, $\sigma_k = 1.00$, $\sigma_\epsilon = 1.30$, $C_{1\epsilon} = 1.44$, $C_{2\epsilon} = 1.92$. σ_k and σ_ϵ are the Prandtl numbers of k and ϵ respectively, connecting the diffusivities of the variables to the eddy viscosity ν_t [27]. Looking at the last term in the model equation for k , it is clear that the destruction of k is proportional to ϵ . This is logical because k denotes the creation of turbulent energy, while ϵ accounts for the destruction.

2.2.3 WALL BOUNDARY LAYER

The k - ϵ -equations (15) and (16) are only valid for fully developed turbulent flows. In the near wall area, the viscous effects become more significant than the turbulent stresses, and the flow is no longer fully developed. The k - ϵ model can not be used near the wall, and separate wall functions need

2 THEORY

to be made.

The boundary layer is divided into an inner region and an outer region. In the inner region viscous shear dominates and the shear stress is approximately equal to the wall shear stress. In the outer region turbulent shear stresses dominate.

In the inner region the flow is independent on the free stream properties, and u depends only on y , ρ , μ and τ_w , $u = u(y, \rho, \mu, \tau_w)$, where y is the distance from the wall and τ_w is the wall shear stress [28]. With 5 variables and 3 dimensions, the Buckingham pi-theorem [29] can be used to conclude that 2 dimensionless numbers are needed. A new velocity is defined, called the friction velocity, $u_\tau = \sqrt{\tau_w/\rho}$ which leads to

$$\pi_1 = u^+ = \frac{u}{u_\tau} \text{ and} \quad (17)$$

$$\pi_2 = y^+ = \frac{\rho y u_\tau}{\mu} = \frac{y u_\tau}{\nu}, \quad (18)$$

where u is the mean velocity in the near wall region, y is the distance from the wall and τ_w is the wall shear stress. These two pi-groups lead to the conclusion that

$$u^+ = f(y^+). \quad (19)$$

The inner boundary layer is divided into 3 different regions [28], see figure 1:

1. Viscous sublayer: $y^+ < 5$

The flow is dominated by viscous effects. In this region there is a linear relationship between the velocity and the distance from the wall, $u^+ = y^+$. This is shown in figure 1 with a red line.

2. Buffer layer: $5 < y^+ < 30$

The viscous and the turbulent (Reynolds) stresses are of similar magnitude. In this region, no defined relation between u^+ and y^+ exists, it is the transition from a linear relationship to the log-law relationship.

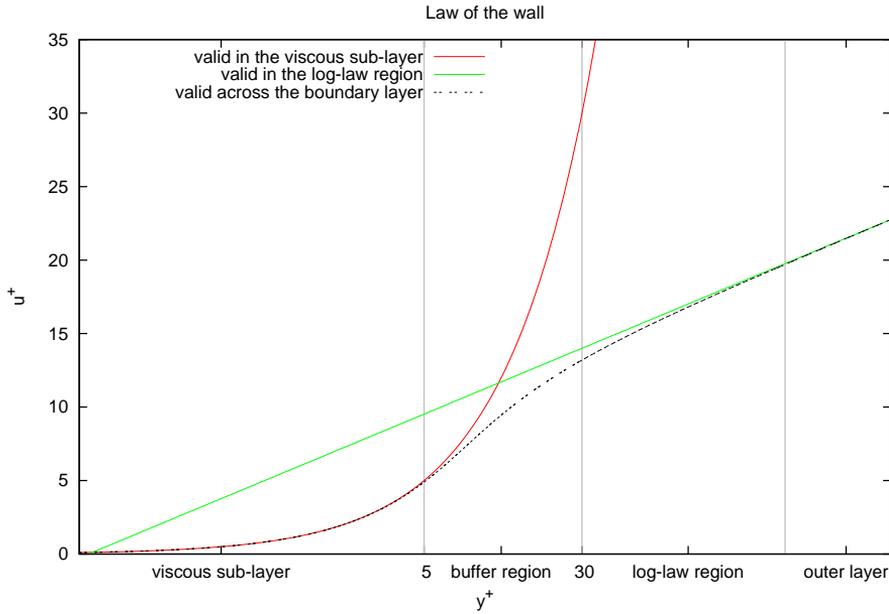


Figure 1: Velocity distribution in a boundary layer, red line is valid below $y^+ = 5$ and the green line shown the characteristic above $y^+ = 30$. The black dotted line is valid throughout the domain, see equation (20).

3. log-law region: $30 < y^+ < 300$

In this region the viscous effects are less important, the flow is dominated by turbulent stresses. The relationship is $u^+ = \frac{1}{\kappa} \ln(y^+) + B$, where $\kappa \approx 0.4$ is von Kármán's constant and $B \approx 5.5$. This is illustrated by the green line in figure 1.

In 1961 Spalding deduced a formulation that is valid in the entire wall-related region [28],

$$y^+ = u^+ + e^{-\kappa B} \left[e^{\kappa u^+} - 1 - \kappa u^+ - \frac{(\kappa u^+)^2}{2} - \frac{(\kappa u^+)^3}{6} \right], \quad (20)$$

and this function is showed with a black line in the figure 1.

For high Reynolds numbers the boundary layer region is very small. The flow near the wall can be taken care of by using two different methods; (1) by adding viscosity and damping terms to the model near the wall and use a locally refined mesh and (2) by using wall functions derived from the log-law relations in the node closest to the wall [18].

The details close to the wall is not the main focus of this project, and therefore refining the mesh and doing detailed calculations with alternative 1 is not necessary. In addition, a different turbulence model would need to be used close to the wall because the $k - \epsilon$ model is not valid there. The general way of treating near wall regions with the $k - \epsilon$ model is to introduce wall functions as in alternative 2, and this method will be used in this project. For alternative 2 the results near the wall are modelled and not simulated. The log-law is used for the entire boundary layer, and due to this the details in the viscous sub-layer are lost. In the OpenFOAM CFD tool, log-law wall functions, $u^+ = \frac{1}{\kappa} \ln(y^+) + B$, already exist. When using the wall functions, there is a demand that only one node should be placed within the boundary layer. This node will be placed in the middle of the boundary layer, i.e. no grid points will be placed in the viscous sub-layer. When analysing the flow it is important to remember that the first nodal point is modelled and not necessarily accurate.

2.3 WIND ENERGY

Wind turbines use airfoils to extract the kinetic energy in the wind blowing past the turbine. As illustrated in figure 2, a horizontal axis wind turbine (HAWT) consist of a tower, a nacelle and a rotor, consisting of rotor blades and a hub. The rotor on the most common wind turbines today, consist of three turbine blades.

Figure 3 shows a sketch of a general turbine airfoil, illustrating among other things the chord and the angle of attack. The airfoil uses the relative motion between the blade and the wind to create mechanical power. The wind flow past the airfoil creates a lower pressure on the upside of the blade, and a

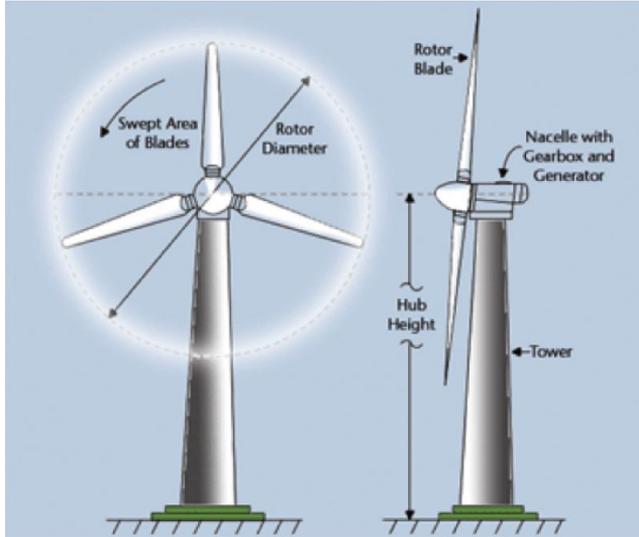


Figure 2: A general sketch of a wind turbine [24].

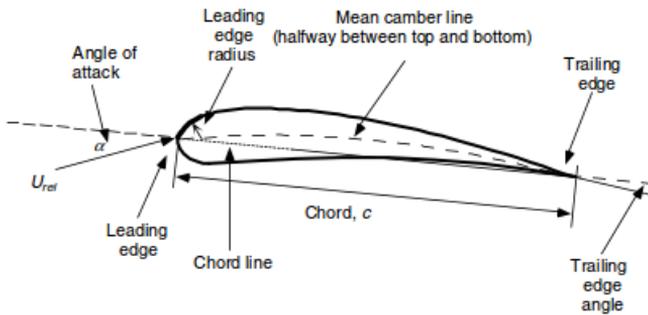


Figure 3: Airfoil nomenclature [10].

high pressure field below the blade. This pressure difference results in a lifting force acting on the blade. In addition, a drag force is created due to the pressure difference and due to viscous forces at the surface of the blade [10].

2.3.1 MOMENTUM THEORY

Momentum theory (also called the actuator disk theory, which is not the same as the generalized actuator disk method) is often used to estimate the power output, the thrust and the efficiency of an ideal wind turbine. The following derivations are based on a book by Manwell et al. [10]. The analysis starts with a stream tube, as shown in figure 4. It is assumed that the wind flows in and out through the ends of the stream tube, and no wind exits through the other tube boundaries. The wind turbine is modelled as an actuator disk. The actuator disk imposes forces on the flow, creating a discontinuity in the pressure.

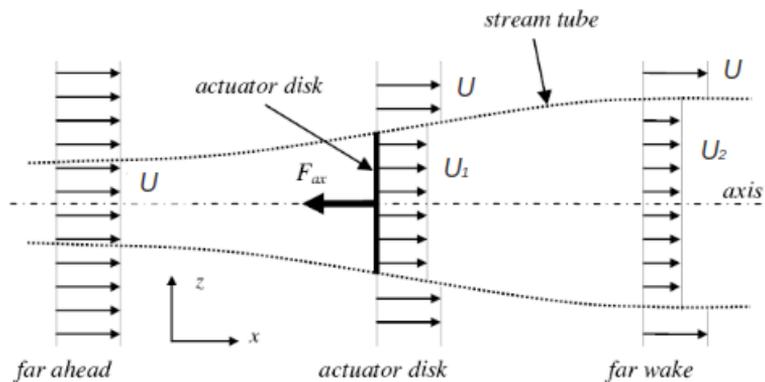


Figure 4: Actuator disk with stream tube, edited from [15].

The assumptions of this model are

- Homogeneous, steady state, incompressible and inviscid flow
- An infinite number of blades
- Non-rotating wake
- Pressure at the beginning and end of the stream tube are equal to the surrounding static pressure
- Uniform flow velocity at the disk, i.e. uniform thrust

The mass flow of air through an area A can be determined as

$$\frac{dm}{dt} = \dot{m} = \rho AU, \quad (21)$$

where U is the free stream wind velocity and ρ is the density of the wind, $\rho = 1.225 \text{ kg/m}^3$ at standard conditions. The mass flow through the stream tube will be constant.

Two important parameters in wind turbine performance is the thrust, the force acting from the wind on the wind turbine, and the power output. The thrust can be calculated using conservation of linear momentum,

$$T = U(\rho AU) - U_2(\rho AU)_2 = \dot{m}(U - U_2) \propto U^2, \quad (22)$$

where U_2 is the velocity at the outlet of the stream tube. A corresponding dimensionless thrust coefficient is found as

$$C_T = \frac{\text{Thrust force}}{\text{Dynamic force}} = \frac{T}{\frac{1}{2}\rho U^2 A}. \quad (23)$$

By taking into consideration the pressure discontinuity across the disk and using Bernoulli's equation twice, once upstream to the disk and once from the disk downstream, the thrust force can be written as

$$T = \frac{1}{2}\rho A(U^2 - U_2^2) \quad (24)$$

The power output, P , from the turbine equals the thrust multiplied with the velocity at the disk U_1 ,

$$P = \frac{1}{2}\rho AU_1(U^2 - U_2^2) \propto U^3, \quad (25)$$

and a dimensionless power coefficient is defined as

$$C_P = \frac{\text{Rotor power}}{\text{Wind power}} = \frac{P}{\frac{1}{2}\rho U^3 A}. \quad (26)$$

The power output of the wind turbine is proportional to the wind velocity cubed, which means that a small increase in wind velocity brings forth significant changes in power.

Several applications utilize different extensions of the momentum theory to create simplified computational techniques to calculate the flow through a wind turbine, some of these include;

Blade Element Momentum Theory (BEM)

The momentum theory regards flow only in one direction, and BEM extends this model to include rotation by including conservation of angular momentum.

Generalized actuator disk

An extension of BEM, and the difference between the two models is that BEM assumes radial independency which is not needed for the generalized actuator disk model. The forces acting on the rotor are calculated using airfoil data and corrected for 3D effects using a blade element approach. The major limitation of actuator disk models is the assumption of an infinite number of blades, meaning that the model is only valid for rotationally symmetric flows.

Actuator line

In contrast to the actuator disk which equally distributes a drag force on a disk, the actuator line model distributes the body forces from the rotor on lines representing the rotor blades. It was created to avoid the assumption of an infinite number of blades, and to improve the modelling of the tip and root vortices. The actuator line model will be used in this project, and a more thorough explanation of it will be given in section 2.3.2.

Actuator surface

Sørensen and Shen has made an extension of the actuator line model, were the forces are distributes not only along the blade, but also in the chordwise direction [21].

2.3.2 ACTUATOR LINE

The velocity field is modelled using the Navier-Stokes equations (1) and (2) where \mathbf{f} denotes the body forces acting on the flow field from the rotor blades. These body forces are calculated using two-dimensional tabulated airfoil data combined with a blade element method. Each blade is divided into a set of blade elements, and each spanwise element has a constant airfoil, chord, twist and oncoming wind [25].

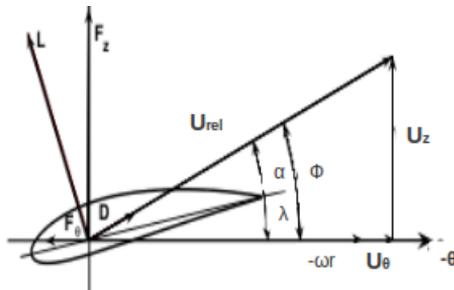


Figure 5: Cross section of an airfoil element showing velocity and force vectors [25].

The local velocity as shown in figure 5 between the flow and the rotating blades is given by

$$U_{rel} = \sqrt{U_z^2 + (\omega r - U_\theta)^2}, \quad (27)$$

where ω is the rotational speed of the turbine and r is the radius of the blade. The angle between the relative velocity U_{rel} and the rotor plane is

$$\varphi = \tan^{-1} \left(\frac{U_z}{\omega r - U_\theta} \right). \quad (28)$$

The local angle of attack is $\alpha = \varphi - \lambda$, where λ is the local pitch angle.

The forces from the turbine blades on the flow is calculated per spanwise length using the chord length of the blade and the local velocity, as

$$\mathbf{f}_{2D} = \frac{d\mathbf{F}}{dr} = \frac{1}{2} \rho U_{rel}^2 c (C_L \mathbf{e}_L, C_D \mathbf{e}_D), \quad (29)$$

where \mathbf{e}_L and \mathbf{e}_D are the unit vector in the direction of the lift and drag forces respectively and c is the chord length. The lift and drag coefficients used to calculate the force is given as tabulated airfoil data.

The force from a blade on the flow is calculated at each line element of the blade. To ensure that this force does not act in a point wise manner, a 3D Gaussian correction is made to smooth the force out over the blade by taking the convolution of the force with a regularization kernel as

$$\mathbf{f}_\epsilon = \mathbf{f} \otimes \eta_\epsilon, \quad (30)$$

where \mathbf{f} is the actuator point force, while \mathbf{f}_ϵ is a body force projected onto the grid. The regulation kernel is

$$\eta_\epsilon(r) = \frac{1}{\epsilon^3 \pi^{3/2}} \exp \left[- \left(\frac{r}{\epsilon} \right)^2 \right] \quad (31)$$

where r is the distance between a grid cell center and the actuator line element point and ϵ decides the Gaussian width, i.e. how far the force is distributed from its initial point [25]. The function decays to 1% of its maximum value for $r = 2.15\epsilon$ [3].

There are several parameters that need to be determined in the actuator line model, and a parameter study will be performed in section 5.2 to check the sensitivity of the solution to these parameters.

Advantages of the actuator line model [25]

- When using the airfoil data, fewer grid points are needed to model the turbine compared to a fully resolved computation, because the viscous boundary layer around each rotor blade is not calculated. This saves a lot of computational power for both actuator disk and line models.
- Several wake phenomenons such as tip, root and bound vortices, which is not resolved in an actuator disk model can be studied.
- A simple grid structure can be used, and grid generation is easy.

Disdvantages of the actuator line model [25]

- Many of the input parameters depend on each other, and currently no clear guidelines on how to determine these parameters exist.
- Input of airfoil data is needed such as lift and drag coefficients obtained numerically or experimentally, and additional corrections need to be made for e.g. 3D-effects. These airfoil data represent a common source of simulation inaccuracy.

3 NUMERICAL METHODS IN OPENFOAM

OpenFOAM is an open source CFD program written mainly in C++. The program comes with a range of solvers, each designed to solve a specific fluid mechanics problem, and in addition there are utilities to perform data manipulation. All source codes are available and the user can edit solvers or other applications to fit to their specific project. A short tutorial and programmer guide can be found on the OpenFOAM web page, but no extensive documentation of the program exists. Questions that arise can either be answered by using internet forums or by inspecting the source code.

OpenFOAM is based on the finite volume method, FVM, and has many numerical schemes implemented, both for time and space integration. Some of the solvers include *icoFoam*, a solver for transient laminar incompressible flow, *simpleFoam* (steady-state) and *pisoFoam* (transient) solver for incompressible, turbulent flow.

3.1 FINITE VOLUME METHOD

The finite volume method, FVM, starts with the integral form of the equations. The following equations are based on the derivations of Versteeg and Malalasekera [27]. The derivation here is shown with 1D and 2D examples, which can easily be expanded to a 3D case.

The Navier-Stokes equations are convection-diffusion equations with a source term, but the pressure and velocities are coupled and has to be solved simultaneously. In the following section a 1D steady convection-diffusion problem with a source for a conserved variable Φ is presented as an example.

A constant control volume Ω with boundary $\partial\Omega$ is considered. The conservation law of Φ in integral form is

$$\underbrace{\int_{\partial\Omega} \rho\Phi (u \cdot n) dA}_{convection} = \underbrace{\int_{\partial\Omega} (\Gamma\nabla\Phi) \cdot n dA}_{diffusion} + \underbrace{\int_{\Omega} S_{\Phi} dV}_{source}, \quad (32)$$

where Γ is a diffusion coefficient e.g. viscosity μ , and n is the outer normal vector of the control volume. Consider the control volume Ω to be a 1D

cylinder with an inlet at a western border, w , and outlet at an eastern border, e . Integrating across the cylinder yields

$$(\rho u A \Phi)_e - (\rho u A \Phi)_w = \left(\Gamma A \frac{d\Phi}{dx} \right)_{(e)} - \left(\Gamma A \frac{d\Phi}{dx} \right)_{(w)} + \bar{S} V_R, \quad (33)$$

where \bar{S} is the mean value of the source term and V_R is the volume of the central cell R .



Figure 6: The control volumes of the central cell R with its two neighbouring cells, W and E . w and e indicate the cell faces.

Adding several control volumes together creates a computational domain. A section of a domain is shown in figure 6, to show the notation used in the finite volume method. Lower case letters denote values at the faces between the nodal points, while upper case letters define the value at the center of the cell. R is the middle cell, while W is the cell at the left side of R , and E is the cell at the right side of R . The face w is between cell points R and W , while e is between R and E .

Convection mass flux F and diffusion conduction D is introduces as

$$F_e = (\rho u A)_e, \quad (34)$$

$$D_e = \frac{\Gamma A}{\Delta x_{RE}}, \quad (35)$$

where Δx_{RE} is the distance between node center R and E , the value at the western faces are treated similarly. Using this, equation (33) becomes

$$F_e \Phi_e - F_w \Phi_w = D_e(\Phi_E - \Phi_R) - D_w(\Phi_R - \Phi_W) + S_u + S_r \Phi_R, \quad (36)$$

where central differences have been used to discretize the diffusive fluxes, and the source term has been split into a uniform part and a part depending on the variable Φ . The continuity equation yields for the convection mass fluxes

$$F_e - F_w = 0. \quad (37)$$

The values of Φ at the faces for the convective fluxes can be approximated using different schemes. Two schemes will be used in the simulations for this study, one named Gauss linear in OpenFOAM which is the central FVM scheme, and the other one is Gauss upwind, which is the upwind FVM scheme. Both will be presented in the following.

Central FVM

With this method convective fluxes are approximated by linear interpolation between the cell midpoints

$$\begin{aligned} \Phi_e &= \frac{1}{2} (\Phi_R + \Phi_E), \\ \Phi_w &= \frac{1}{2} (\Phi_W + \Phi_R). \end{aligned} \quad (38)$$

The conservation equation now becomes

$$\begin{aligned} &\overbrace{\left[D_w - \frac{F_w}{2} + D_e + \frac{F_e}{2} - S_r \right]}^{a_R} \Phi_R \\ &= \underbrace{\left(D_w + \frac{F_w}{2} \right)}_{a_W} \Phi_W + \underbrace{\left(D_e - \frac{F_e}{2} \right)}_{a_E} \Phi_E + S_u, \end{aligned} \quad (39)$$

where $a_R = a_W + a_E - S_r$.

Upwind FVM

Using this approach, convection-diffusion equations are also convection-diffusion equations with a source term, but the pressure and velocities

are coupled and has to be solved simultaneously. The pressure appears in the momentum equation, but no transport or other equation exists for the pressure. Some examples of algorithms used to take care of this are shown in section 3.4, after a discussion about an appropriate grid has been made. tive fluxes are determined by choosing the value in upwind cell as

$$\begin{aligned}\Phi_e &= \begin{cases} \Phi_R, & \text{if } U_e > 0, \\ \Phi_E, & \text{if } U_e < 0, \end{cases} \\ \Phi_w &= \begin{cases} \Phi_W, & \text{if } U_w > 0, \\ \Phi_R, & \text{if } U_w < 0, \end{cases}\end{aligned}\quad (40)$$

where U_e and U_w are the velocities at the eastern and western faces¹. By defining $F^+ = \max(F, 0)$ and $F^- = \min(F, 0)$, equation (36) becomes

$$\begin{aligned}& \overbrace{[D_w - F_w^- + D_e + F_w^+ - S_r]}^{a_R} \Phi_R \\ &= \underbrace{(D_w + F_w^+)}_{a_W} \Phi_W + \underbrace{(D_e - F_e^-)}_{a_E} \Phi_E + S_u,\end{aligned}\quad (41)$$

where $a_R = a_W + a_E - S_r$.

The discretization can be expanded to 2D by adding cells to the south and north of the center cell. The convective mass fluxes and diffusion conductions are created for the southern and northern faces in a similar manner as for the western and eastern faces. The 2D version of equation (36) becomes

$$\begin{aligned}F_e \Phi_e - F_w \Phi_w + F_n \Phi_n - F_s \Phi_s &= D_e(\Phi_E - \Phi_R) - D_w(\Phi_R - \Phi_W) \\ &+ D_n(\Phi_N - \Phi_R) - D_s(\Phi_R - \Phi_S) + S_u + S_r \Phi_R.\end{aligned}\quad (42)$$

As mentioned above, the Navier-Stokes equations are convection-diffusion equations, but the coupling of pressure and velocity makes the discretization

¹Capital letters, e.g. U , indicate that the value is on the face, while lower case u is in the cell centre.

more difficult. The pressure appears in the momentum equation, but no transport or other equation exists for the pressure. Usually the conservation of mass is used to develop an equation for the pressure, as will be shown in section 3.4 where two numerical solution algorithms are presented.

3.2 GRID

There are two main types of grids, collocated and staggered. For a collocated grid, all variables are defined at the center of the cells, which can lead to difficulties when coupling the pressure and velocities. Take e.g. pressure field where every second cell has the same value of p , i.e. cell 1 is surrounded by two pressure values; one above, below and at the sides, and another one at the corners, the latter with the same pressure as cell 1 (this is also called a checker board pressure field). If the pressure gradient is approximated by linear interpolation at the west and east of a cell, the resulting term will not contain the pressure within cell R and the pressure gradient will be zero. To ensure that the pressure gradients are better approximated a new grid type is introduced, i.e. the staggered grid.

A staggered grid arrangement is shown in figure 7. The pressure is still defined at the center of the cells, but the velocities are defined at the cell faces. The u -velocities are defined at the western and eastern faces, while the v -velocities are defined at the southern and northern faces. It is practical to give the velocities at the cell faces, because this is where they are needed to calculate the flux into the cell. The three governing RANS equations (6), (7) and (8) will have three different grids, one with the pressure and other scalar quantities at the node centers, one shifted $\Delta x/2$ to the left with the u -velocity at the node centers and one shifted $\Delta y/2$ downwards with the v -velocity at the node centers [27].

The advantages of the staggered grid diminishes with complex geometries and when a non-Cartesian grid is used.

OpenFOAM operates with a collocated grid arrangement, where the velocities are stored in the center of the cells. To create a staggered grid, OpenFOAM uses a new variable that saves the interpolated velocity values

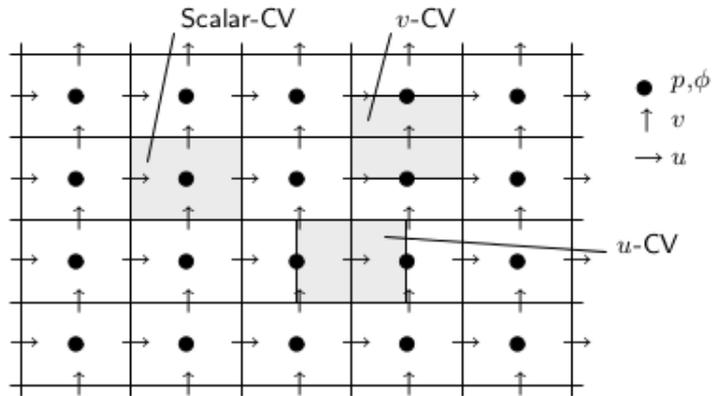


Figure 7: Illustration of a staggered grid arrangement [20].

at the faces.

3.3 BOUNDARY CONDITIONS

Boundary conditions need to be determined for the domain edges. Two main types of boundary conditions exist, Dirichlet, a given value at the face, and Neumann, a given flux at the face. The treatment of these boundary conditions will be shown for the steady convection-diffusion example in section 3.1 with the central discretization for the convective fluxes.

The two most common boundary conditions are the Dirichlet and the Neumann conditions. Both of these will be used in this thesis, as presented in section 4.

Dirichlet boundary condition:

A Dirichlet boundary condition, with a given value at the western border, $\Phi_w = \Phi_a$, is shown in figure 8. For the cell closest to the western border,



Figure 8: Dirichlet boundary condition with Φ given at the western border.

equation (39) becomes

$$\underbrace{\left[D_e + \frac{F_e}{2} + D_a - S_r \right]}_{a_R} \Phi_R = \underbrace{\left(D_e - \frac{F_e}{2} \right)}_{a_E} \Phi_E + \underbrace{S_u + F_a \Phi_a + D_a \Phi_a}_{\tilde{S}_u}, \quad (43)$$

where F_a is the convective mass flux at the western border, and D_a is the diffusion conduction at the western border. If the diffusion coefficient Γ is kept constant and on an equidistant grid, the diffusion coefficient will be constant throughout the domain, except for at the border. At the western border the distance between the two nodal points is $\Delta_{Ra} = \Delta_{RW}/2$, and the diffusion coefficient will be twice as large as for the rest of the domain.

Neumann boundary condition:



Figure 9: Neumann boundary condition with no change in Φ across the eastern border.

A Neumann boundary condition at the eastern border is shown in figure 9. It is assumed that the conduction at the eastern border is negligible, i.e.

$D_e = 0$. Φ is convected out of the domain, and the convection is modelled as $\Phi_e = F_e \Phi_R$. This can also be found by setting $\Phi_E = \Phi_R$. For the cell at closest to the eastern border, equation (39) becomes

$$\underbrace{\left[D_w + \frac{F_w}{2} - S_r \right]}_{a_R} \Phi_R = \underbrace{\left(D_w + \frac{F_w}{2} \right)}_{a_W} \Phi_W + S_u. \quad (44)$$

where the continuity equation, $F_e - F_w = 0$, is used. This is equivalent to setting $a_E = 0$ in the original equation (39).

3.4 SOLUTION ALGORITHMS

Different solution algorithm for solving the Navier-Stokes equations with a coupled velocity and pressure field exist. In this thesis the *PISO* solver will be used in the simulations. Both the *SIMPLE* and *PISO* solvers will be presented here, because the *PISO* solver is an expansion of the *SIMPLE* solver.

3.4.1 SIMPLE

SIMPLE stands for Semi-Implicit Method for Pressure-Linked Equations. A few simplifications are made in the derivation of the algorithm, starting by linearizing the non-linearities by calculating the convective mass fluxes at the the faces with the velocities from the previous time step. The idea is to calculate preliminary velocities that do not fulfil the continuity equation from the momentum equations using a guessed pressure field, or the pressure field from the previous iteration. Correction factors for the pressure and the velocities are found by using the continuity equation [27].

Initially, a pressure field p^* is guessed, which is then used to solve the x - and y -momentum equations to find u^* and v^* . For the x -direction with source term b_R this becomes

$$a_R u_R^* = \sum_{nb} (a_{nb} u_{nb}^*) - (p_e^* A_e - p_w^* A_w) + b_R, \quad (45)$$

where nb stands for neighbours. The velocities and the source term are at the middle of the computational cell, while the pressure terms are at

the cell faces. The a coefficients are defined as for the convection-diffusion equation from section 3.1. The areas A_e and A_w are assumed to be equal, $A_e = A_w = A_R$.

A correction factor for the pressure is defined so that $p = p^* + p^c$, and similar correction factors for the velocities. By subtracting equation (45) for u^* from equation (45) for u , the velocity correction equation becomes

$$a_R u_R^c = \sum_{nb} (a_{nb} u_{nb}^c) - (p_e^c - p_w^c) A_R \quad (46)$$

The main simplification of the *SIMPLE* algorithm is that the first term on the right hand side of equation (46) is neglected, along with the similar term for the y -momentum equation. The correction velocities u^c and v^c are now only functions of the p^c -field. p^c is found by using the continuity equation with $u = u^* + u^c$ and $v = v^* + v^c$ and with u^* and v^* calculated using p^* . The pressure correction equation is

$$a_{p,R} p_R^c = a_{p,W} p_W^c + a_{p,E} p_E^c + a_{p,S} p_S^c + a_{p,N} p_N^c + b_{p,R}, \quad (47)$$

where $a_{p,W} = \rho A_{u,R}^2 / a_{u,R}$, $a_{p,S} = \rho A_{v,R}^2 / a_{v,R}$ and similar for the other coefficients, and $b_{p,R} = -[(\rho A u^*)_e - (\rho A u^*)_w + (\rho A v^*)_n - (\rho A v^*)_s]$. The notations $a_{p,W}$, $a_{u,W}$ and $a_{v,W}$ are due to the staggered grid, where the three variables p , u and v have separate grid definitions. $a_{p,W}$ is the coefficient belonging to the western border of the grid cell with pressure at the mid point, and similar for u and v .

A coefficient α_p is used to enhance convergence. This is called under-relaxation because it limits the change in the variables per iteration, and the new pressure field is calculated to be $p = p^* + \alpha_p \cdot p^c$. α_p has a big influence on the convergence rate, but unfortunately there is no theory available to determine it. α_p is therefore chosen on experience, and a usual choice is $\alpha_p = 0.3$. Under-relaxation for the velocities is used in the iterative process when solving equation (45) for more advanced methods than *SIMPLE*. The velocity is under-relaxed as (for the x -direction)

$$u^{*(k+1)} = u^{*(k)} + \alpha_u \left(u^{*(k+1)} - u^{*(k)} \right), \quad (48)$$

and is often chosen as $\alpha_u = 0.7$. (k) and ($k + 1$) indicate the iteration steps [27].

The *SIMPLE* algorithm in summary: (1) guess pressure field, (2) calculate velocities, (3) solve pressure correction equation, (4) correct pressure and velocities and (5) start over again at point (2) until convergence [27].

3.4.2 PISO

PISO stands for Pressure Implicit with Splitting of Operators. This algorithm starts off with the 3 first steps of the *SIMPLE* algorithm, with the same coefficients and the same pressure-correction equation as in *SIMPLE*. Then it performs a second pressure correction to correct the pressure and velocities one more time. The second pressure-correction equation is derived in the same way as the *SIMPLE* pressure-correction equation.

After the first pressure correction, identical to the *SIMPLE* algorithm, the resulting corrected pressure and velocities become $p^{**} = p^* + p^c$, $u^{**} = u^* + u^c$ and $v^{**} = v^* + v^c$. In the second pressure correction equation, a new pressure correction factor is defined as p^{cc} so that $p = p^{**} + p^{cc}$. Similar correction factors for the velocities are made, u^{cc} and v^{cc} .

The discretized u -momentum equation (45) now becomes

$$a_{u,R}u_R^{**} = \sum_{nb} a_{u,nb}u_{nb}^* - (p_e^{**} - p_w^{**})A_R + b_R. \quad (49)$$

For the twice corrected velocities, a similar equation is obtained as

$$a_{u,R}u_R = \sum_{nb} a_{u,nb}u_{nb}^{**} - (p_e - p_w)A_R + b_R. \quad (50)$$

Note that the velocities in the first term on the right hand side of equation (49) and (50) are calculated from the non-corrected velocities, i.e. $u^* = u^{**} - u^c$ and $u^{**} = u - u^{cc}$.

Subtracting equation (49) from equation (50) yields

$$u_R = u_R^{**} + \frac{\sum_{nb} a_{u,nb}(u_{nb}^{**} - u_{nb}^*) + b_R}{a_{u,R}} - \frac{A_R}{a_{u,R}}(p_e - p_w). \quad (51)$$

Equation (51), and the corresponding equation in the y -direction, are used in combination with the continuity equation to derive the second pressure-correction equation.

The *PISO* algorithm in summary: (1) guess pressure field, (2) calculate velocities, (3) solve pressure-correction equation, (4) correct pressure and velocities, (5) solve pressure-correction equation, (6) correct pressure and velocities and (7) start over again at point (2) until convergence [27].

The *simpleFoam* solver in OpenFOAM is a steady state solver, while the *pisoFoam* solver is transient. The *pisoFoam* solver will be used in this thesis, where a transient solution of the wind turbine is of interest.

3.4.3 LINEAR SOLVER CONTROL

Linear solvers are methods used within the solution algorithms to compute a set of linear discretized equations. The linear solvers available in OpenFOAM are preconditioned conjugate gradient (for symmetric matrices), preconditioned biconjugate gradient (for non-symmetric matrices), solver using a smoother, generalised geometric-algebraic multi-grid and diagonal solver for explicit systems. More information about these can be found in [16]. The two linear solvers used in this thesis will be presented below.

PRECONDITIONED BICONJUGATE GRADIENT, PBICG

This linear solver is used for the velocity U and for the turbulent properties, k and ϵ , and it is an iterative solver for linear equations. A preconditioner transforms the set linear equations into form that fits better for numerical calculations. Several options for preconditioning of matrices in the conjugate gradient solvers exist, including diagonal incomplete-Cholesky (symmetric), faster diagonal incomplete-Cholesky (DIC with caching), diagonal incomplete-LU (asymmetric), diagonal, geometric-algebraic multi-grid and no preconditioning. The diagonal incomplete-LU preconditioner is used in this thesis, and it is closely related to LU factorization. For more information see [1].

GENERALISED GEOMETRIC-ALGEBRAIC MULTI-GRID, GAMG

This linear solver is used for the pressure p for the simulations in this thesis.

The main idea of this method is to first perform calculations on a coarse grid, and then map this solution onto a finer grid as an initial guess for the calculation. This grid refinement method may be performed in one or more steps, until a solution on the desired grid has been found. This solver is efficient, the extra time it takes with grid refinement and solution mapping is outweighed by the time saved to first do the calculations on a coarse grid. More information about this solver, and OpenFOAM in general can be found in [16].

4 NUMERICAL SET-UP

The main focus of this project is to perform wind turbine calculations using the actuator line model, where a model turbine at NTNU is the basis for the numerical set-up. The simulations in this thesis was compared to real wind tunnel data from the model turbine.

4.1 WIND TUNNEL AND MODEL TURBINE

In figure 10 the wind tunnel at NTNU and the model wind turbine is shown.



Figure 10: A picture of the wind tunnel with the model turbine at NTNU from "Blind test 1" [9].

Figure 11 shows a sketch of the model wind turbine used in "Blind test 1" and "Blind test 2".

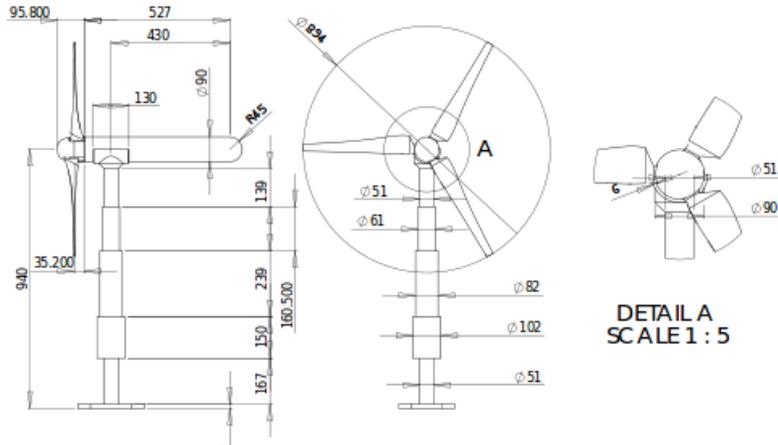


Figure 11: A schematic of the model turbine at NTNU used in "Blind test 1" and "Blind test 2" [17].

The tunnel is $11.15m$ long, has a width of $2.71m$ and it is $1.801m$ high at the entry and $1.851m$ high at the exit. The wind turbine is located $3.66m$ downstream of the inlet, the rotor center is $0.817m$ above the ground with a diameter of $0.903m$ (including the hub diameter). A general sketch of the wind tunnel and model turbine is shown in figure 12.

The simulations in this project have been performed at the design conditions which are a velocity of $U = 10m/s$ and a tip speed ratio of $TSR=6$. More information about the set-up of "Blind test 1" can be found in [9].

A simulation of the "blind test 2" will also be made and compared to the experimental values. For this case, the dimensions of the wind tunnel are equal to the "Blind test 1", but there are two turbines in the tunnel. The upstream turbine is placed $1.79m$ from the inlet, while the second turbine is placed 3 diameters behind the first one. The downstream turbine is the one from "Blind test 1" shown in figure 11. The upstream turbine has a few

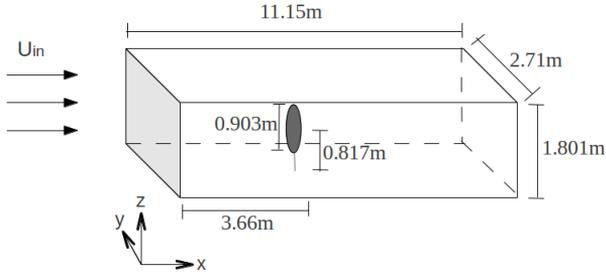


Figure 12: Schematic of the domain of the wind tunnel in "Blind test 1".

differences from the downstream turbine, but the differences are small and both turbines will be simulated using the same model. The first turbine is kept at $TSR = 6$, while the downstream turbine has $TSR = 4$. The inlet velocity is kept at $U = 10m/s$. Additional details about the set-up of "Blind test 2" can be found in [17].

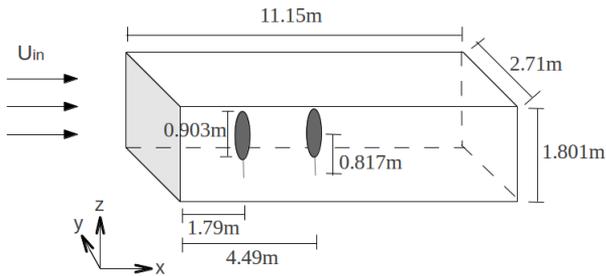


Figure 13: Schematic of the domain of the wind tunnel in "Blind test 2".

4.2 BOUNDARY CONDITIONS

The boundary conditions for the wind tunnel are given as

Inlet

- Given value for the velocity, ϵ ² and k , i.e. Dirchlet boundary condition. The inlet values are uniform across the inlet area.
- Zero gradient for the pressure, i.e. Neumann boundary condition.
- Calculated turbulent viscosity, ν_t .

Outlet

- Given value for the pressure.
- Zero gradient for ϵ and k .
- Calculated ν_t and velocity.

Walls

- Given value for the velocity.
- Zero gradient for the pressure.
- k , ϵ and ν_t calculated from the OpenFOAM wall functions based on the theory in section 2.2.3.

4.3 AIRFOIL DATA

The actuator line model needs airfoil data from the blade as inputs, consisting of C_D , C_L and the local angle of attack. The airfoil data used in this project has been acquired from participants in "Blind test 1". The airfoil data used in this thesis has been created by John Amund Lund at Meventus by using Xfoil. The airfoil data proved to work well in the "Blind test 1" and "Blind test 2" simulations, and is available at Reynolds numbers 10000,

²Turbulent dissipation of kinetic energy, not to be confused with the Gaussian width parameter of the actuator line model

30000, 50000, 75000, 100000, 150000 and 300000.

The Reynolds number is given as

$$Re = \frac{U_{blade}c}{\nu} \quad (52)$$

where c is chord length, ν kinematic viscosity and U_{blade} is the local velocity on the blade approximately

$$U_{blade} = \sqrt{U_{axial}^2 + U_{rot}^2}. \quad (53)$$

The axial velocity U_{axial} is the velocity on the blade in the x -direction, estimated to be the inlet velocity, $U_{in} = 10m/s$. The rotational speed of the blade, U_{rot} , is dependent on the location on the blade, i.e. the distance from the centre of the turbine. The rotational speed of the turbine is $\omega = 1281.8rot/min = \frac{1281.8 \cdot 2\pi}{60}rad/s$, and the rotational velocity on the blade at radius r is $U_{rot} = \omega r$.

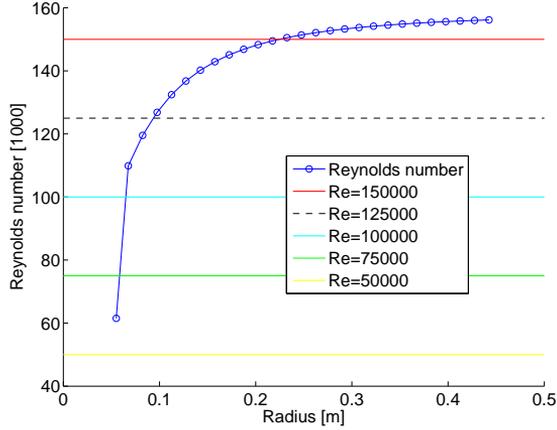


Figure 14: Reynolds number across the blade (divided by 1000).

The resulting plot of the Reynolds numbers across the blade is shown in figure 14. The horizontal lines indicate Reynolds numbers where airfoil data

is known. One extra line between $Re = 100000$ and $Re = 150000$ has been added to the figure (the black dashed line), to illustrate which points are closest to $Re = 100000$ and which are closer to $Re = 150000$. As seen in the figure, most of the points will be assigned a Reynolds number of 150000, but the three points closest to the center of the turbine will have lower Reynolds numbers and therefore different airfoil properties than the rest of the blade.

4.4 MESH AND OPENFOAM APPLICATIONS

The starting point for this thesis was a set-up in OpenFOAM acquired from Kalvig and Manger [7]. This included the actuator line model created by SOWFA (as explained below) and a pisoFOAM solver where the actuator line library was implemented. In addition, details from the "Blind test 1" were implemented, such as the tunnel dimensions, the mesh, the boundary conditions and the model turbine properties.

MESH

The mesh was generated using the OpenFOAM tool *blockMesh*. This is, as the name indicates, a mesh generation based on creating several blocks. The chosen grid is Cartesian, the x -direction is aligned with the flow, the y -direction is across the tunnel while the z -direction is upwards. The height of the tunnel has been modelled as constant and equal to the inlet height, i.e. $1.801m$. The mesh generation in this thesis was generated by first creating one large box, and then dividing this box a given number of times in each direction to save computational power.

The grid is refined in three steps around the turbine, where at each refinement the grid spacing is halved in each direction. The refinement was performed as

1. 1 diameter in front of the turbine, 2 diameters behind the turbine, $3/4$ diameter in the horizontal direction and to the tunnel boundaries in vertical direction (almost $3/4$ diameter)
2. $3/4$ diameter in front of the turbine, $7/4$ diameters behind the turbine and $1/2$ diameter in the rotor plane

3. $1/2$ diameter in front of the turbine, $3/2$ diameters behind the turbine and $1/4$ diameter in the rotor plane

Figure 15 gives an illustration of the refinement process in the xy -plane.

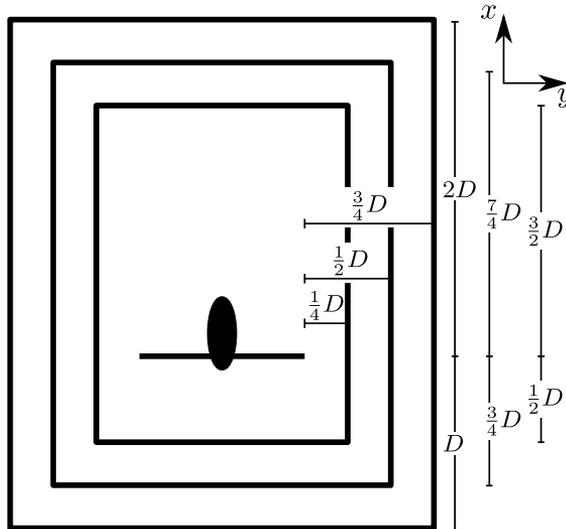


Figure 15: A sketch of the refinement area in the xy -plane.

The first refinement is 3 diameters from the inlet, 6 diameters from the outlet and 0.8 diameter at sides in the y -direction. In the z -direction the grid is refined to the roof and floor for the first refinement.

A closer look too see if this refinement procedure affects the results will be taken. Conclusions from this comparison can be found in appendix A.

PARALLELIZATION

In most of the simulations the mesh was divided into 80 boxes, which were assigned one processor each. The domain is divided into 10 equally sized boxes in the x -direction, four in the y -direction and two in the z -direction. The parallelization is based on MPI, and one of the advantages with OpenFOAM is its good scalability when using many processors. Simulations

where also performed with 16 processors, to check if the parallelization affected the results.

ACTUATOR LINE MODEL

Figure 16 illustrates the rotor as it is modelled by the actuator line model in the rotor plane. For each time step, the rotor blades (green) are moved and the forces acting from the blades on the flow are calculated. The local velocity in the wind field around the blade is used to calculate the lift and drag forces on the blade using equation (29). This creates a dynamic model, where effects like tip and root vortices can be investigated.

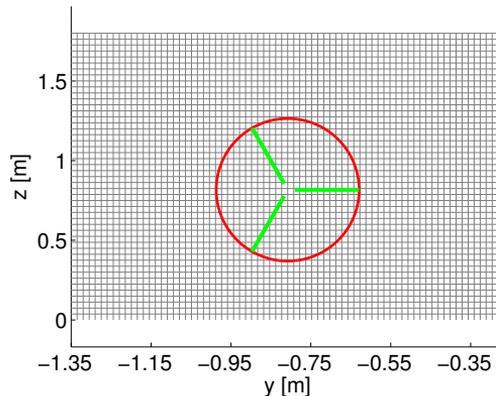


Figure 16: A sketch of the actuator line elements in the grid, green lines. The red circle indicates the sweeping area of the rotor. The grid is shown without grid refinement around the rotor.

The implementation of the actuator line model in OpenFOAM has been performed by Churchfield and Lee at NREL (National Renewable Energy Laboratory) as a part of SOWFA (Simulator for Offshore Wind Farm Applications) [2].

The SOWFA code does not take into consideration the hub and tower, therefore an implementation of the hub and tower will be performed when comparing results to the "Blind test 1" and "Blind test 2". The hub and

tower will be modelled using the blade elements of the actuator line code.

OPENFOAM SOLVER

The `pisoFOAM` solver in OpenFOAM (explained in section 3.4.2) was modified by Kalvig et al. [7] to include the actuator line implementation. This solver was used in the simulations in this study, combined with the RANS equations from section 2.2.1 and the $k - \epsilon$ turbulence model from section 2.2.2.

5 RESULTS AND DISCUSSION

The actuator line code consists of many parameters, and there is a need for best practice guidelines for using the model. This section will start with a visualization of the Gaussian width parameter ϵ , which is an essential parameter in the actuator line model.

Following this, a thorough parameter study will be performed, including a grid independency study. This leads to a proposal for how to use the actuator line model, with specification on how to determine the most important parameters. A summary of the parameters in this section are given in table 6;

Parameter	Description
Number of blade elements (b)	The number of line segments each blade is divided into. It needs to be large enough to create a smooth distribution of the blade forces.
Time step	It needs to be small enough for the flow to not pass through more than one cell per time step.
The Gaussian width parameter (ϵ)	ϵ spreads the forces from the blade element points onto the mesh, determining the force value and how far from the blade element it is distributed. It needs to be large enough to avoid numerical oscillations, but small enough to not overestimate the thrust and power output.
Grid	It needs to be small enough to capture the main features of the flow field.

Table 6: An overview of the parameters in the parameter study

The developed guidelines from the parameter study are validated against the wind tunnel measurements from "Blind test 1" and "Blind test 2". The simulated results are compared to the measurements for thrust, power and the velocity deficit and turbulent kinetic energy in the wake behind the turbines. A comparison has also been made to the "Blind test 1" and "Blind

test 2" candidates to see whether the actuator line model performs better or worse than other simulations.

In addition, an investigation of noise found in the results, the grid refinement procedure and an unexpected result for thrust and power at a given grid resolution have been performed, and the results are included in the appendices.

5.1 A VISUALIZATION OF ϵ

The variable ϵ is used to smear the force from the blade elements onto the grid cells around the blade. This section will take a closer look at this smearing, and how it affects the shape of the forces around the blade.

The force in each grid cell is dependent on d , which is the distance between one blade element and the mid point of a surrounding grid cell. d_{max} is the largest distance the forces are spread out from one blade element. The smearing is stopped by the solver when the distance between the blade element and the center of the grid cell is larger than

$$d_{max} = \epsilon \cdot \sqrt{\log(1000)} \approx 2.63\epsilon, \quad (54)$$

which is when the value of the force has reached 0.1% of its original value. This relation was found in the source code of the actuator line implementation.

The forces around one blade have been calculated. The regulation kernel from the actuator line model shown in equation (31) was used to spread the forces from the blade element point and out to the grid cells surrounding the blade point. In figure 17 the contours of the resulting force around one blade in the rotor plane are drawn. The blade is symmetrical, so a 2D representation is sufficient. The example case has 20 blade elements, and different values for ϵ have been used. The grid cells are twice the length of one blade element in both y - and z -directions. To get the distribution of the forces across the blade, a simulation was run. The simulated forces from the blade elements were used as the basis for creating figure 17.

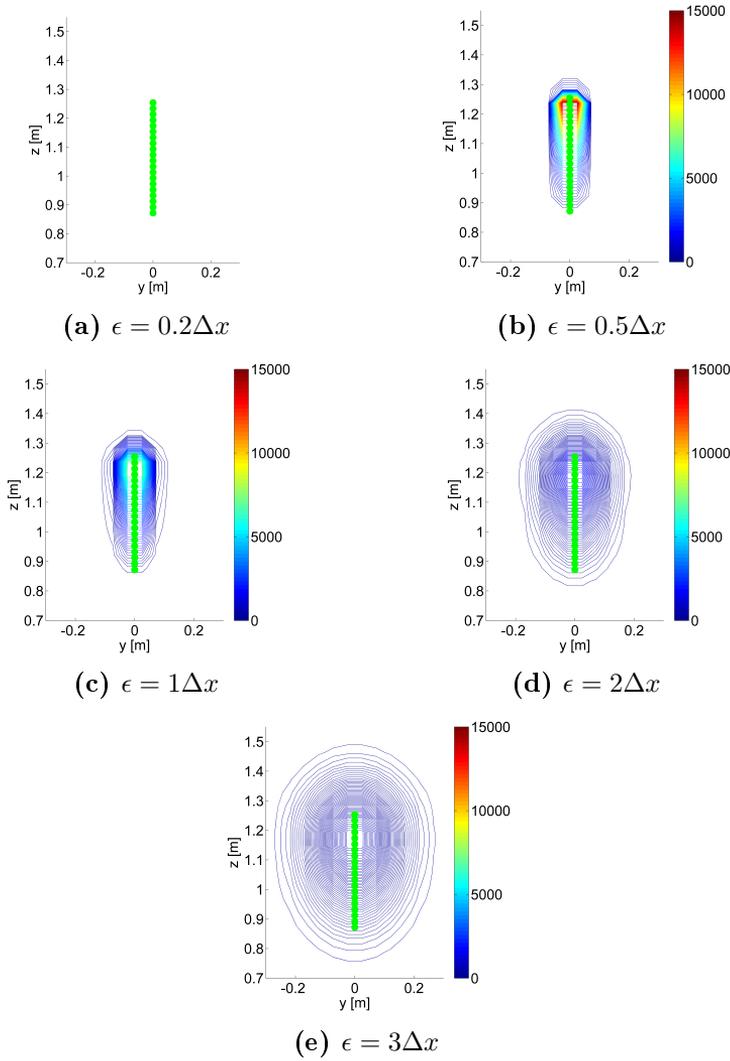


Figure 17: Force distribution around the blade with 20 blade elements, blade elements shown in green. The force from the blade elements is found through a simulation, and then spread out onto the grid using equation (31).

For lower ϵ the force is high and spread out over a small area, therefore the incoming wind will experience a swift and large velocity change close to the turbine. For higher ϵ the force is small, but spread out on a large area. The incoming velocity will be decelerated slowly, and the velocity change will start at a greater distance from the turbine than for small ϵ .

From figure 17a it is clear that $\epsilon = 0.2\Delta x$ is too small, the force from the blade is not distributed to any of the surrounding cells. Figure 17e for $\epsilon = 3\Delta x$ indicates a blade that is far larger than the actual size of the modelled blade, and in addition the forces extend above and below the blade dimensions creating an artificially long blade.

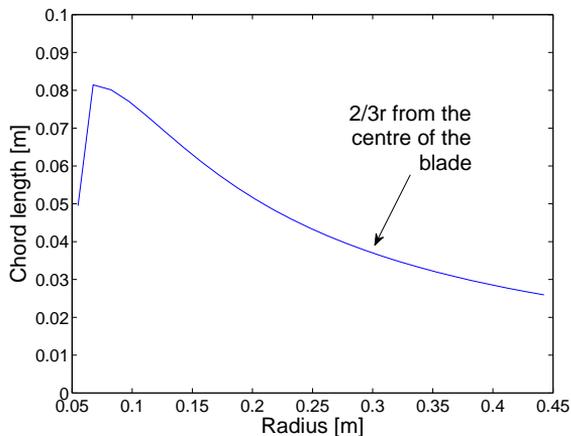


Figure 18: Chord length over blade radius.

The width of the different blades created in figure 17 spans from $0.167m$ for $\epsilon = 0.5\Delta x$ to $0.521m$ for $\epsilon = 3\Delta x$ at the thickest point. The chord length can be seen in figure 18, with an average of $c = 0.048m$. Although figure 17 illustrates the area of the forces acting on the blade and not the actual blade size, the differences between the blade size and the distributed forces should not be smaller. This shows that the grid in this example is too coarse to get a good model of the blade.

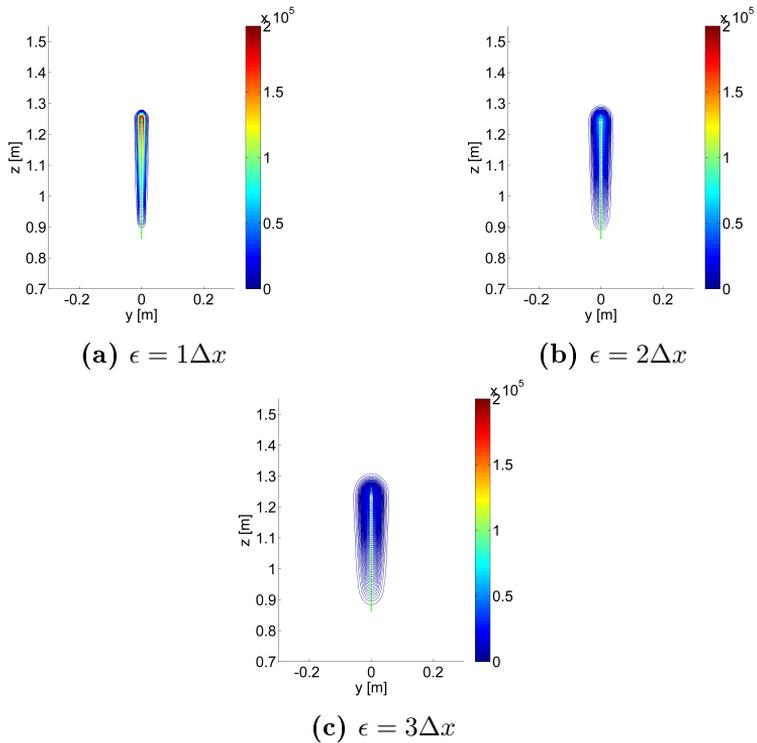


Figure 19: Force distribution around the blade with 100 blade elements, blade elements shown in green. The force from the blade elements is found through a simulation, and then spread out onto the grid using equation (31).

Figure 19 illustrates the same as figure 17, but with a finer grid and 100 blade elements. The areas of the distributed forces are greatly reduced, for $\epsilon = 1\Delta x$ the area is $0.046m$, $0.113m$ for $\epsilon = 3\Delta x$. The area of the force is still larger than than the chord length, but the difference is a lot smaller, especially for $\epsilon = 1\Delta x$.

In figure 17b and 19 it seems like the blade points closest to the rotor centre

do not spread out any forces. This is because the forces from these points are negligible compared to the other forces and do not show in these figures.

From the results in this section, it is apparent that ϵ should be linked to the dimension of the blade, i.e. the chord length c . In the following, a presentation on how this can be achieved is given.

ϵ stretches the blade over a distance of d_{max} in each direction from the blade elements. As an example, imagine the spreading is two dimensional. The forces are spread out d_{max} in front of the blade element and d_{max} behind the blade element. The width of the spread out forces is $2d_{max}$. To get a good representation of the blade $2d_{max}$ should be linked to the chord length, so that $c = 2d_{max}$.

Churchfield et al. [3] recommended to use $d = 2.15\epsilon$ to determine ϵ , which represents the distance where the force has reached 1% of its maximum value for. This is a less strict demand than using d_{max} and will be used in this thesis. A good representation of the size of the blade is then $\epsilon \approx c/(2 \cdot 2.15) = c/4.3$, as also suggested by [3]. Shives and Crawford suggested $\epsilon = c/4$ [22], but $\epsilon = c/4.3$ will be used in this thesis.

More power is generated close to the tip than at the root of the blade, therefore a chord length $2/3$ radius from the root is chosen as a representative chord length, i.e. $c = 0.0362m$ resulting in $\epsilon = 0.0084m$.

To achieve the demand of $\epsilon > 2\Delta x$, the grid spacing in the x -direction is limited to $\Delta x = 0.0042m$. If the grid is produced without refinement around the rotor, a total of 754 million grid cells are required, and with three grid refinement 33 million cells are needed. With an appropriate time step, the calculations with a grid spacing of $\Delta x = 0.0042m$, both with and without refinement, is very heavy and time consuming. Simulations with four refinements around the rotor was performed in section 5.2.3 to achieve $\epsilon > 2\Delta x$.

Martinez [11] showed that the solution is more dependent on the grid for smaller ϵ than for large. He performed simulations for four different ϵ , and for the finest ϵ an $\epsilon/\Delta x = 2$ was not enough for convergence of the solu-

tion. Between $\epsilon/\Delta x = 2$ and $\epsilon/\Delta x = 4$ the power output changes from $P = 1.91MW$ to $P = 1.98MW$, i.e. a change of 3.6%. Due to this, demanding such a small ϵ is a difficult starting point if grid independency is to be achieved.

5.2 PARAMETER STUDY

The actuator line model requires a lot of input variables. Some of the variables are given by the case specification such as inlet velocity and turbine geometry, including airfoil data. To perform a parameter study, it is important to identify the changeable variables that exist in the model, and to find possible correlations between the variables. A summary of the most important parameters is given in table 6.

NUMBER OF BLADE ELEMENTS

Each blade is divided into a set of blade elements. According to Martinez et al. [11] the number of blade elements should be high enough to make sure that the forces from the blades on the flow are smooth. They concluded that for each grid cell across the blade, there should be at least 1.5 blade elements.

In other words $\Delta r/\Delta y < 0.67$, where

$$\Delta r = \frac{\text{blade radius}}{\text{number of blade elements}}, \quad (55)$$

is the size of one blade element and Δy is the smallest grid spacing in the rotor plane. A conclusion from this is that Δr is dependent on the grid resolution in the rotor plane.

The simulations by Martinez et al. were performed with LES, while RANS was used for this thesis. A parameter study using RANS might not necessarily give the same conclusions as with LES.

TIME STEP

For the solution to be convergent, it is important that the flow does not pass through more than one cell for each time step. Normally the requirement for a flow with velocity U would be

$$\frac{U\Delta t}{\Delta x} \leq 1, \quad (56)$$

but in this case the velocity of the rotor blades has to be taken into consideration. The fastest velocity in the grid appears at the tip of each blade.

The tip speed ratio in this project is $TSR=6$, defined as the ratio of the blade tip speed to the free stream velocity U_{in} [10],

$$TSR = \frac{\omega R}{U_{in}}. \quad (57)$$

There are 21.36 rotations per second, and the tip moves at $60m/s$. In the following example, the grid spacing in the y - and z -direction is $\Delta y = 0.0094m$. In order to ensure that the tip does not move through more than one Δy per time step, when the tip is moving at $60m/s$, the time step has to be

$$\Delta t < \frac{0.0094m}{60m/s} = 0.00016s. \quad (58)$$

This equation does not take into account that the blade moves in a circular motion, while the grid is Cartesian.

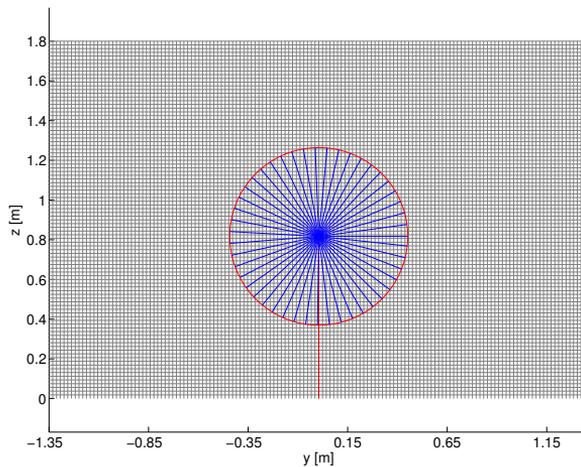


Figure 20: The grid and all the placements that one turbine blade occupies in the numerical code through one turbine rotation, $\Delta t = 0.001s$.

Figure 20 illustrates a grid of $\Delta y = \Delta z = 0.0094m$ in the rotor plane, together with one blade per time step (blue lines) during one second. In this figure the time step is $\Delta t = 0.001s$, which is too crude. The figure illustrates that the tip of the rotor blade passes more than one cell per time step.

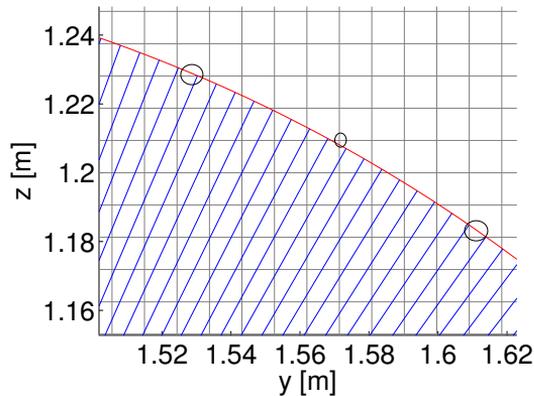


Figure 21: A closer look at the grid and the placements that one turbine blade occupies in the numerical code through one turbine rotation, $\Delta t = 0.0001s$. The black circles indicate grid cells that the rotor tip passes through, but that are skipped by the numerical blade. This is where the red line goes through a cell, but a blue line is never in this cell.

Figure 21 is an enlarged picture of figure 20, but with a time step of $\Delta t = 0.0001s$. According to equation (58) this time step is sufficiently small. The figure shows that because the computations in this project is performed on a Cartesian grid, the blade tip occasionally passes through a grid point without being calculated. This is considered to be negligible, but calculations with several time steps will be performed to check. The time step is, as in all CFD codes, dependent on the grid spacing.

GAUSSIAN WIDTH PARAMETER, ϵ

The Gaussian width parameter, ϵ , is an essential variable in the actuator line model, as was also seen in section 5.1 where ϵ was visualized. Currently

there exists no acknowledged generalized method to calculate it. The blade forces are originally point forces, and ϵ is used to spread the forces in three directions. If ϵ is chosen too small, the blades appear in the simulations as discontinuous and oscillations occur in the numerical solution. If, on the other hand, a too large ϵ is used, the blade forces are distributed too far from the blade creating a blade that seems too large, and the tip and root vortices are not simulated correctly [25] [11].

According to Troldborg and Martinez [25] [11], ϵ should be at least twice the local grid cell length in the x -direction to avoid numerical oscillations, i.e. ϵ is connected to the grid spacing. Shives and Crawford [22] recommended a stricter demand, ϵ larger than four times the grid spacing. The differences in these results might come from simulations using different CFD tools. Shives and Crawford used the ANSYS CFX tool, Troldborg the *EllipSys3D* solver and Martinez the same OpenFOAM actuator line code that will be used in this thesis.

In addition, ϵ should be connected to the chord length of the blade, in order to model the blade shape in a good way. Shives and Crawford used this approach to calculate flow around geometries simpler than a full turbine [22]. This approach was also recommended, but not performed by Churchfield et al. [3] because it requires a very fine grid.

The parameter study will include one part where ϵ is adjusted relative to the grid resolution, and one part where ϵ is kept constant at a value representing the size of the blade. This thesis present simulations for a full turbine where ϵ is chosen from the chord length, and this has not been performed previously.

MESH

According to Troldborg [25] the solution of the actuator line code will always have a certain dependency on the grid. He performed calculations on both a Cartesian and a polar grid, and compared the solutions for the different grid types. Martinez et al. [11] performed calculations for several grid densities and concluded that the power output was strongly connected to the grid resolutions.

The usual procedure for checking the mesh in a CFD set-up, is to repeatedly refine the mesh until the solution is converged, i.e. the solution is grid independent. For the actuator line model, such a dependency check is difficult to perform, because all of the above variables are connected to the mesh.

From this, two things can be concluded; (1) the grid is very important to achieve best possible results and (2) a solution independent of the grid is hard to find. Nevertheless during the parameter study, several simulations will be performed to see if a grid independent solution can be achieved.

PARAMETER STUDY PROCEDURE

The parameter study will be performed in multiple steps;

1. Investigate the dependence between different variables, and suggest equations that link them together
2. Perform a grid independency check where the variables are adjusted according to the developed relationships
3. Perform a grid independency check where the variables are adjusted according to the developed relationships, but where ϵ is kept constant at a value representing the blade size
4. The main goal of this thesis; develop a final suggestion for the best use of the actuator line model

5.2.1 PART 1 - IDENTIFYING PARAMETERS AND THEIR RELATIONSHIP

There are several inputs to the actuator line code, and some of these depend on each other, as seen above. This section will focus on identifying the relationships between the parameters, and see how the different parameters affect the result.

To start the parameter study a reference case was created, based on the knowledge gained from the literature, and from the analysis performed above.

The starting point for the grid resolution was chosen as the grid in the case

acquired from Kalvig and Manger [7]. The grid cell length in the x -direction is about 1.65 time the grid cell lengths in the y - and z -direction. The time step is chosen as calculated above, and divided by 1.5 to put in a security margin. The number of blade elements are chosen as twice the amount of grid cells across one blade. ϵ was in the starting case chosen as $0.0122m$.

The reference case is summarized in the table below;

Δt [s]	Cells	ϵ [m]	b	Δx [m]	Δy [m]	Δz [m]
0.0001	2.4 mill	0.0122	100	0.0155	0.0094	0.0094

Table 7: Variables for the starting point of the parameter study.

Simulations have also been performed using cubical grid cells, to see how this affects the results. The reference case for this is;

Δt [s]	Cells	ϵ [m]	b	Δx [m]	Δy [m]	Δz [m]
0.0001	4 mill	0.0122	100	0.0094	0.0094	0.0094

Table 8: Variables for a reference case with a cubical grid.

NUMBER OF BLADE ELEMENTS

The reference case in table 7 was used as a starting point to investigate the influence on the solution from the blade elements. Simulations were run at blade elements from 10 to 300. The number of computational grid points across the blade is kept constant at 47.7, and $\Delta r/\Delta y$ varies from 4.77 for 10 blade elements to 0.16 for 300 blade elements. Figure 22 shows the thrust and power for different number of blade elements.

From the thrust curve in figure 22a it can be seen that 10 blade elements are definitely too few, and also for 25 blade elements the thrust and power are lower than the converged value. The variation in thrust for blade numbers above 50 is minimal, the total difference in thrust in figure 22a is $0.04N$, or 0.001%. Figure 22b shows the power curve, and the variation in power is $0.5W$, i.e. 0.0025%, which is also a negligible difference. For 50 blade

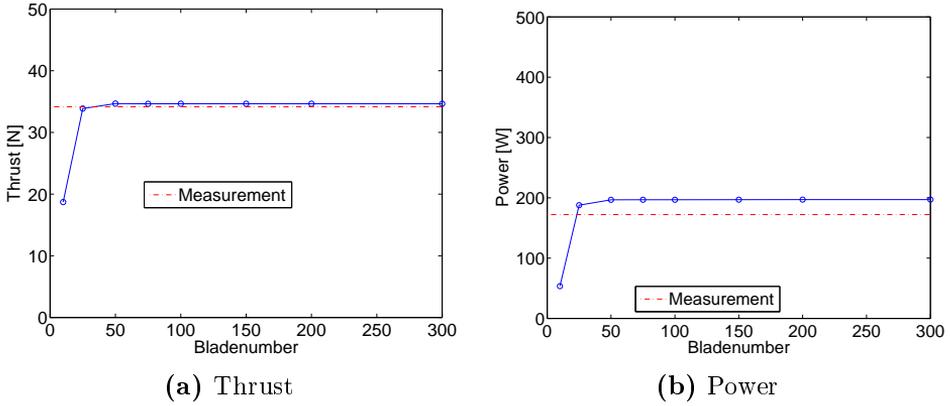


Figure 22: Thrust and power for varying number of blade points, with all other variables as in table 7.

elements, there is about one blade element per grid cell.

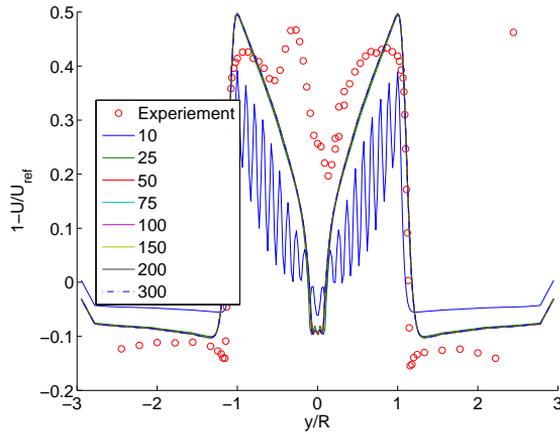


Figure 23: Velocity wake profile one diameter behind the rotor for different numbers of blade elements, red dots indicate the experiments from "Blind test 1". The hub and tower have not been included in these simulations.

Figure 23 shows the velocity wake shapes one diameter behind the turbine. It is clear that too few blade elements create non-physical oscillations in the velocity field. 10 blade elements result in a velocity wake shape that largely differs from the other simulations. The velocity wake shapes are identical for number of blade points higher than 50. The experimental results from "Blind test 1" are added as red dots, and will be discussed further in section 5.3.

Simulations were also performed with the cubical reference grid in table 8, with number of blade elements of 50, 100 and 150. The difference in the results is minimal, and the conclusion is that 50 blade element points is sufficient also for this grid resolution.

As a conclusion the number of blade elements has to be larger than the number of grid points across the blade in the rotor plane,

$$\Delta r < \Delta y, \tag{59}$$

which is a less crude than Martinez' demand of $\Delta r/\Delta y < 0.67$.

TIME STEP

The reference case in table 7 has been used as a starting point for the time step check. Additional simulations have been performed with time steps between $10^{-3}s$ and $10^{-5}s$. The resulting thrust and power curves have been shown in figure 24.

From the thrust curve in figure 24a and the power curve in figure 24b it is clear the the time step has a big influence on the solution. The difference in thrust between the simulations run with $\Delta t = 0.0001s$ and $\Delta t = 0.0002s$ is almost 10%, while between simulations run with $\Delta t = 0.0001s$ and $\Delta t = 0.00005s$ the difference is below 3%.

Previously it was calculated that the time step for this grid resolution had to be smaller than $0.00016s$. For a time step larger than this, e.g. $\Delta t = 0.0002s$, the thrust and power are above the converged values, while a time step smaller than $\Delta t = 0.00016s$, e.g. $\Delta t = 0.0001s$, is sufficient.

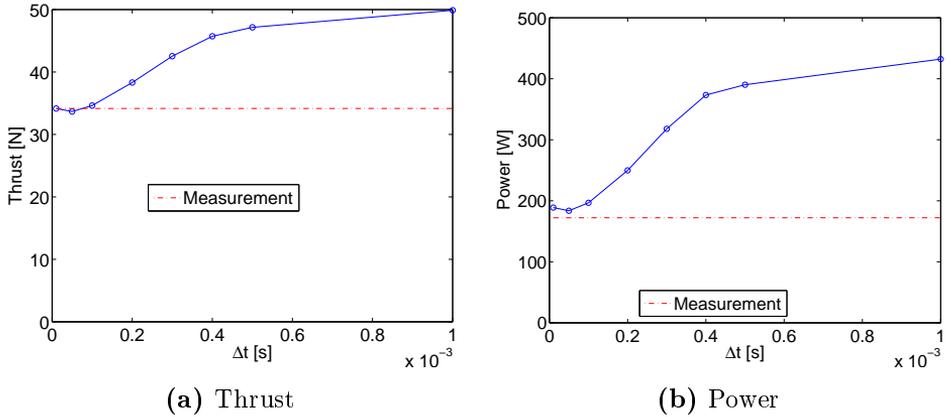


Figure 24: Thrust and power for varying Δt , with all other variables as in table 7.

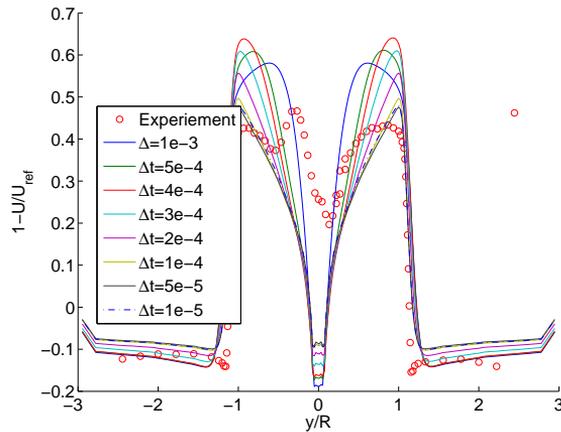


Figure 25: Velocity wake profile 1 diameter behind the rotor for different time steps, red dots indicate the experiments from "Blind test 1". The hub and tower have not been included in these simulations.

Figure 25 shows the velocity wake shapes one diameter behind the rotor. It is essential to the solution that the time step is chosen small enough,

and for time steps below $0.0001s$ the velocity wake shapes have a negligible difference.

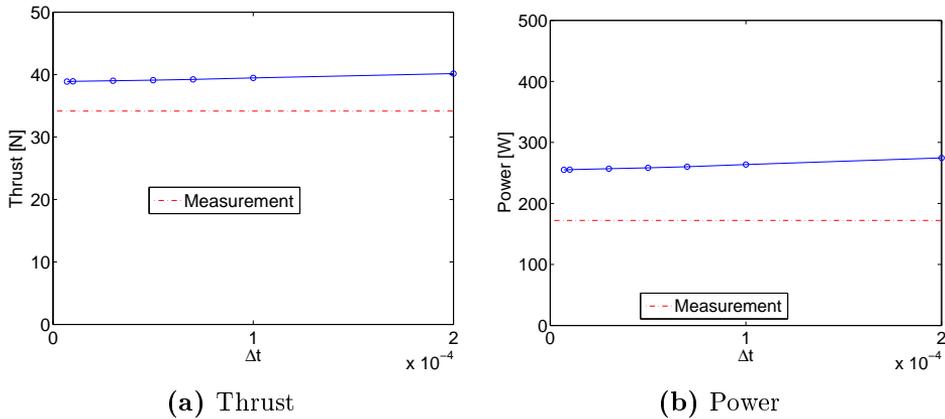


Figure 26: Thrust and power for varying Δt , with all other variables as in table 8.

Simulations were also performed with the cubical reference grid in table 8, with time steps between $2 \cdot 10^{-4}s$ and $7 \cdot 10^{-6}s$. The results are shown in figure 26. The thrust and power has less variation, but the value of the thrust and power is larger than in figure 24. The difference between $1 \cdot 10^{-4}s$ and $7 \cdot 10^{-6}s$ is only 1.5 % in thrust and 3% for power. For this grid resolution, as above, $\Delta t = 0.0001s$ is sufficient.

The conclusion is that the maximum time step calculated as $0.0001s$ above is both sufficient and necessary. The solution is more dependent on the time step for coarser grids.

ϵ

The parameters in table 7 was used as the initial condition for testing the influence on the solution due to ϵ . Cases were simulated with ϵ between $0.0061m$ and $0.0488m$.

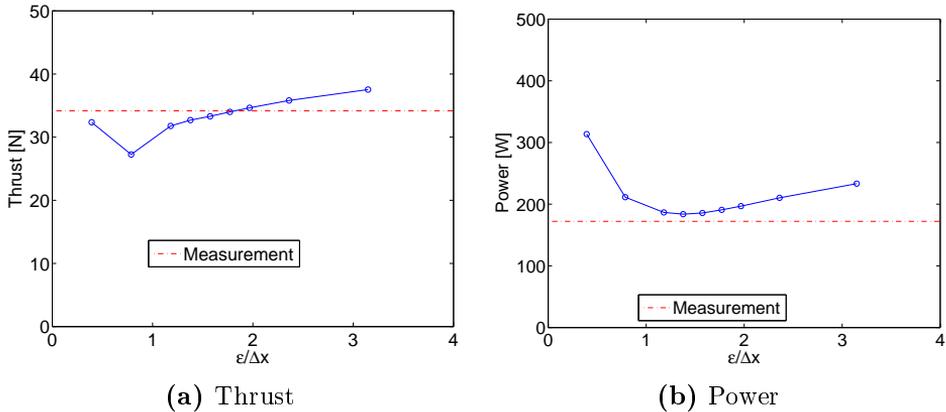


Figure 27: Thrust and power for varying ϵ , with all other variables as in table 7, Δx is the finest grid resolution in the x -direction of the domain.

Figure 27 shows the resulting thrust and power. Both thrust and power reach a minimum point, thrust has a minimum a bit lower than $\epsilon/\Delta x = 1$ and power has a minimum slightly above this limit. Thrust and power seem to converge, but for a large $\epsilon/\Delta x$. The difference in thrust between the $\epsilon/\Delta x = 3.2$ and $\epsilon/\Delta x = 2.4$ is almost 5 %, and the difference in power is 10 %.

Simulations were also performed with the cubical reference grid in table 8, with ϵ ranging from $0.0047m$ to $0.037m$. Figure 28a shows that the thrust increases with increasing ϵ , and the result starts to converge after $\epsilon = 2\Delta x$. In figure 28b the power is shown for various ϵ . The power reaches a minimum at $\epsilon = \Delta x$, while thrust is only decreasing, and the power increases more than the thrust after $\epsilon = 2\Delta x$. The total difference in power between $\epsilon = 2\Delta x$ and $\epsilon = 4\Delta x$ is 2%

The difference in figures 27 and 28 shows that ϵ is dependent on the grid in the x -direction. A finer grid leads a faster convergence for the thrust and power in terms of $\epsilon/\Delta x$, but also in terms of ϵ .

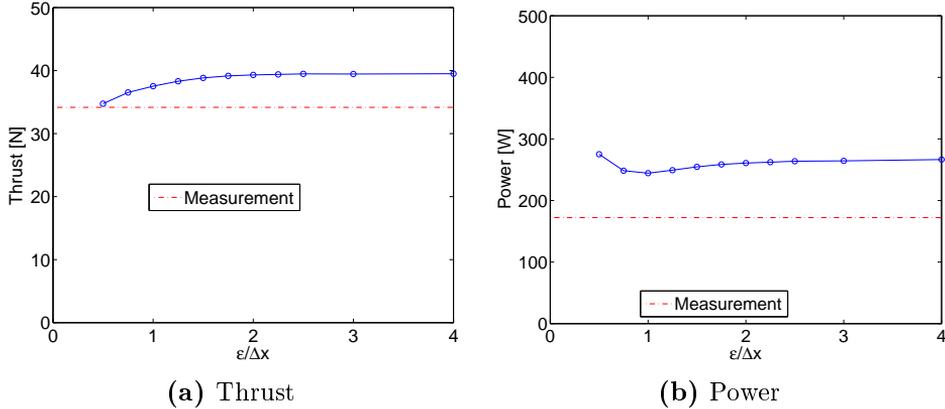


Figure 28: Thrust and power for varying ϵ , with all other variables as in table 8, Δx is the finest grid resolution in the x -direction of the domain.

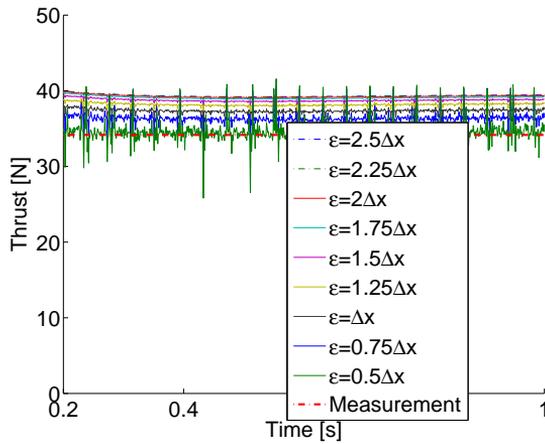


Figure 29: Thrust vs. time for various ϵ with a fixed $\Delta x = 0.0094m$ around the rotor.

Figure 29 shows thrust curves plotted over time for different values of ϵ . For low ϵ , there are large oscillations in the plot, while for larger values of ϵ these peaks are smaller and less significant. A detailed analysis of this problem can be found in appendix B.

It was found that the amplitude of the noise increases with decreasing ϵ . In addition, the noise occurs at certain blade angles, i.e. when the blade passes the y - or z -axis. A possible cause for this noise is the numerical implementation of the actuator line library, the OpenFOAM solver used or the grid configuration.

The conclusion from this study of ϵ is that the demand given by Troldborg for $\epsilon > 2\Delta r$ is sufficient for the case in table 8, although a larger ϵ is needed to achieve convergence for the case in table 7. $\epsilon = 2.5\Delta x$ will be used in the next section.

It has been shown that the solution is dependent on the grid resolution in the x -direction. It might be that the code works better with cubical grid cells. In the following a closer look at the mesh will be given.

MESH

The reference case in table 7 was used as the starting case to investigate the influence on the solution by the grid. The grid is first varied only in the x -direction, and then only in the rotor plane to see how this affects the results.

Simulations were performed with Δx (finest grid in the streamwise direction after the grid refinement is performed as shown in figure 15) ranging from $0.028m$ down to $0.0028m$, including $\Delta x = 0.0094m$ creating the cubical grid case from table 8. For $\Delta x = 0.028m$, 1.3 million cells are needed for the simulation, while $\Delta x = 0.0028m$ requires 13.4 million grid cells. $\epsilon/\Delta x$ ranges from 0.4 to 4.

Figure 30a and 30b show the thrust and power curves for different values of Δx when $\Delta y = \Delta z = 0.0094m$. Both thrust and power increase with decreasing grid spacing in the x -direction. The solution does not seem to converge to a given value for fine grids. The difference between the coarsest

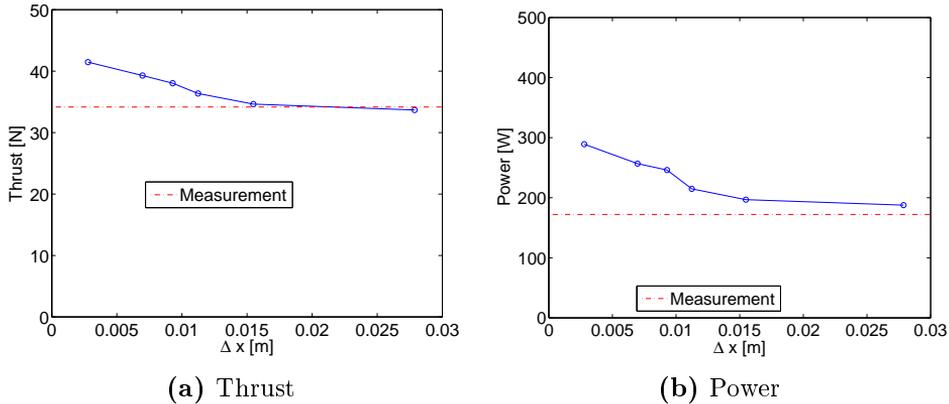


Figure 30: Thrust and power for varying Δx , with all other variables as in table 7.

and finest grid is about 19 % for the thrust and 35 % for the power.

Holding x constant, simulations have been performed with $\Delta y = \Delta z$ from $0.019m$ to $0.0038m$. The number of grid cells range from 0.6 million for the largest grid resolution in the rotor plane, up to 15 million cells for the finest grid.

Figure 31a and 31b shows the thrust and power curves for different $\Delta y = \Delta z$ (finest grid in y - and z -direction) with $\Delta x = 0.0155m$. Thrust and power increase with decreasing grid resolution in the rotor plane, as was also seen for decreasing grid resolution in the x -direction. The increase seems linear, and the solution does not convergence. The difference in the solution between the finest and the coarsest grid is about 10 % for the thrust and 15 % for the power.

In the last section it was seen that the time step and ϵ were dependent on the grid resolution in the x -direction. One mentioned theory was that the code did not work properly for grid cells that are non-cubical. Since the solution varies less with $\Delta y = \Delta z$ than with Δx , although both are producing non-cubical grids, this theory is unlikely to be true.

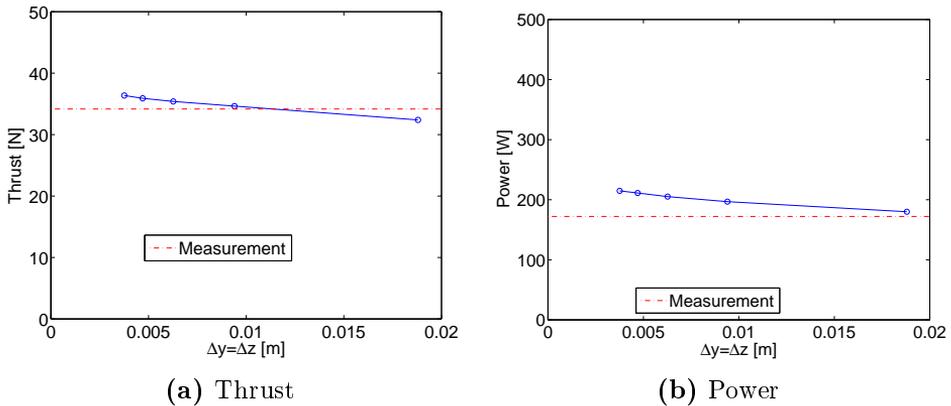


Figure 31: Thrust and power for varying grid in the rotor plane, with all other variables as in table 7.

The conclusion is that the grid has a large influence on the solution, and especially the grid in the x -direction. This conclusion is as expected, as the other parameters are correlated to the grid resolution. In addition, there seems to be no convergence towards a given value when changing Δx or the grid in the rotor plane. Convergence is needed to create a grid independent solution, and special attention will be given to the grid in the next sections.

5.2.2 PART 2 - ϵ CHOSEN FROM GRID RESOLUTION

In this part of the parameter study, a new reference case is chosen implementing the results from the last section as;

Δt [s]	Cells	ϵ [m]	b	Δx [m]	Δy [m]	Δz [m]
0.0001	4 mill	0.0233	100	0.0094	0.0094	0.0094

The difference from the reference case created as a starting point for the parameter study (see table 7) is that the grid is now cubical with a finer grid in the x -direction and ϵ is higher than before.

Relationships between the parameters are then used to perform a grid independency check. Simulations have shown that ϵ is correlated to the grid

resolution, and therefore ϵ will here be changed with the grid, i.e. large grid cell sizes yields a large ϵ and opposite.

To see if the grid refinement around the rotor has a large effect on the solution, simulations will be performed both with and without the grid refinement, and a closer look at this relationship is given in appendix A.

The conclusions from the last section is utilized and the relationship between the variables are kept constant and equal to the reference case. The following set of relationships will be used in the grid independency check;

- $\epsilon = 2.5\Delta x$
- $\Delta r = 2\Delta x$
- $\Delta t = \frac{0.65 * \Delta y}{60m/s}$
- $\Delta x = \Delta y = \Delta z$

INCLUDING GRID REFINEMENT AROUND THE ROTOR

Simulations have been performed for various grid resolutions with three grid refinements around the rotor. The table below shows the finest and coarsest grid resolution, with corresponding ϵ , blade elements, time step and number of grid cells in the domain.

Δt [s]	Cells	ϵ [m]	b	Δx [m]	Δy [m]	Δz [m]
0.0019	2800	0.44	5	0.19	0.19	0.19
0.00005	32 mill	0.0116	200	0.0047	0.0047	0.0047

Figures 32a and 32b show thrust and power for different grid resolutions, between $\Delta x = 0.0046m$ and $\Delta x = 0.19m$, where Δx is the finest grid resolution after refinement. A minimum point can be found for $\Delta x = 0.0186m$ in both graphs. The maximum variance in the plots is 16.14% for thrust and 35.88% for power. The difference between the minimum point and the finest grid is 5.5% in thrust and 7.7% in power. The variation in the fine grids is not very large, but the graphs do not appear to converge towards a

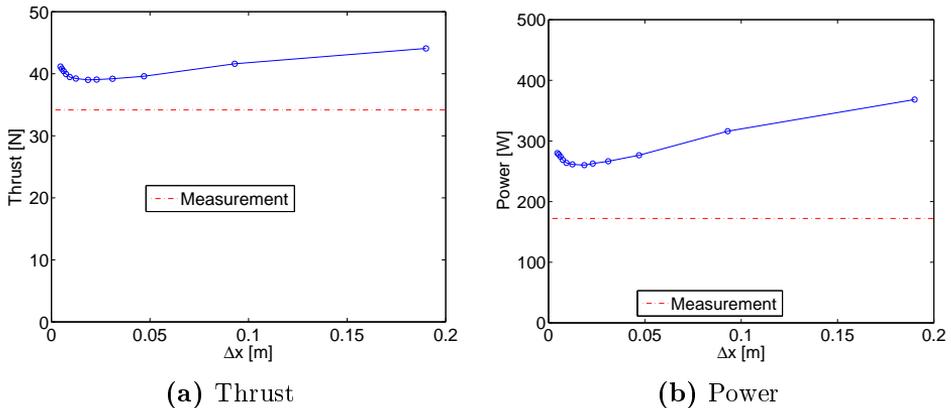


Figure 32: Thrust and power for different grid resolutions, with grid refinement around the rotor. The parameters are changed with the grid.

constant value.

As seen in section 5.2.1 (where the relationship between the model parameters was investigated) the thrust and power increase with decreasing grid resolution, while a decrease in ϵ and the time step has an opposite effect on the results.

One possible explanation for the trend in figure 32a and 32b is that different parameters have a stronger influence on the results for the coarse grid than for the fine grid. All the parameters are here changed in unison, and which parameter is most influential for a given simulation is hard to determine without running several new simulations.

Another possible explanation is that by changing ϵ with the grid resolution, the area of the distributed forces is changed. As seen in section 5.1, where ϵ is visualized, a higher ϵ leads to a larger area of forces spread out onto the surrounding grid cells. This implies that the turbine geometry changes between the simulations, and a solution independent of ϵ and the grid resolution is difficult to achieve. A similar grid independency check will be

performed in the next section for a constant ϵ .

WITHOUT GRID REFINEMENT AROUND THE ROTOR

To see if the grid refinement method described in section 4 has any influence on the solution, new cases were run with constant grid resolutions through the entire domain. The table below shows the finest and coarsest grid resolutions run without grid refinement around the rotor. Finer grid resolutions were not simulated because they were computationally too heavy. For comparison, the finest grid simulated has 29 million grid cells, while the same grid density around the rotor with three grid refinements only needs 2.1 million grid cells.

Δt [s]	Cells	ϵ [m]	b	Δx [m]	Δy [m]	Δz [m]
0.0019	11 000	0.44	5	0.19	0.19	0.19
0.00013	29 mill	0.031	75	0.012	0.012	0.012

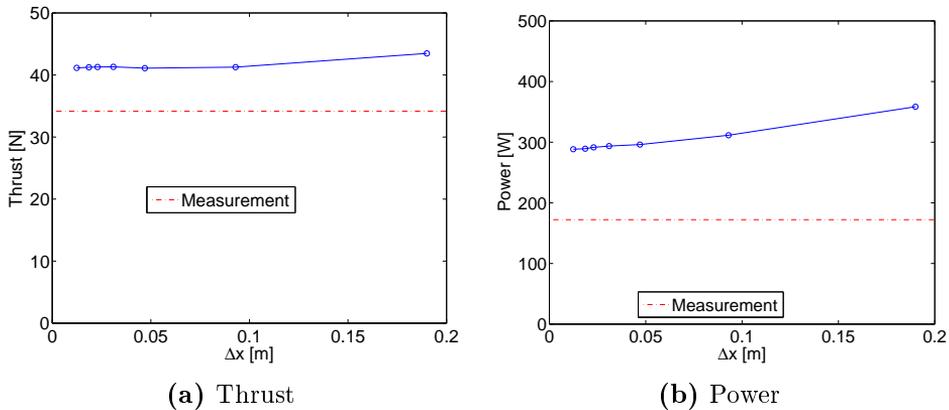


Figure 33: Thrust and power for different grid resolutions, without grid refinement around the rotor. The parameters are changed with the grid.

Figure 33 shows the resulting thrust and power for different grid resolutions. The total variation in both thrust and power is small. The thrust seems to

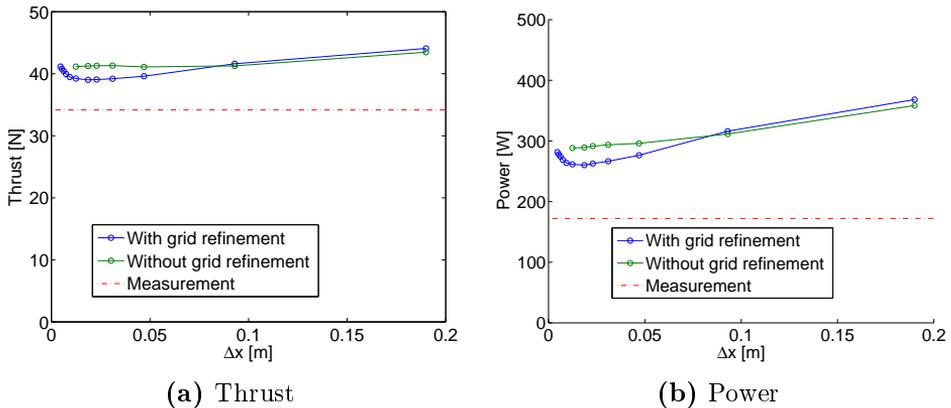


Figure 34: Thrust and power for different grid resolutions, with and without grid refinement around the rotor. The parameters are changed with the grid.

converge, while the power declines slightly for fine grids.

Figures 32 and 33 illustrate again the sensitivity of the code on the grid resolution. A more detailed picture where the thrust and power with and without refinement around the rotor is shown in the same figure can be seen in figure 34. The values for the cases with grid refinement have a convex shape with a minimum point. For the cases without refinement the trend is linear, flat for the thrust and slightly declining in the power graph. The non-refined simulations seem to converge, while the refined cases so not, and this is an indication that the refinement procedure does not function properly.

The differences in the results with and without grid refinement is maximum 5% for the thrust and 10% for power. This difference is investigated in detail in appendix A.

The conclusion is that a grid independent solution was not found for the simulations with grid refinement around the rotor, while the simulations without grid refinement seem to converge for fine grids. The average values

for thrust and power are high above the experimental values both with and without grid refinement. This gives strength to the thesis that ϵ should be linked to the chord length instead of the grid spacing in the streamwise direction (see section 5.1), in order to model the turbine geometry better.

5.2.3 PART 3 - ϵ CHOSEN FROM CHORD LENGTH

This part of the parameter analysis uses a constant ϵ chosen to represent the blade size, in contrast to the section above where ϵ was related to the grid resolution and varied with respect to the other parameters. In this way, the turbine forces are distributed at a constant distance from the turbine, in order to keep the blade size constant.

A closer look at the effect ϵ has on the shape and size of the blade was taken in section 5.1, and the conclusion was that $\epsilon = 0.0084m$ represents the blade size well. In addition, the requirement of $\epsilon < 2\Delta x$ this implies that an ϵ of $0.0084m$ needs to be accompanied with a Δx of maximum $0.0042m$. This is a very small grid spacing requiring many computational hours to simulate, and therefore simulations will also be run for $\epsilon < 2\Delta x$.

This section starts by changing the grid resolution in the entire domain, both with and without grid refinement, then the grid resolution is changed only in the x -direction and only in the rotor plane. Following this, a comparison between the results from this section and the last section where $\epsilon = 2.5\Delta x$ will be made.

As in section 5.2.2, the following parameters are used as a reference case (except for ϵ):

Δt [s]	Cells	ϵ [m]	b	Δx [m]	Δy [m]	Δz [m]
0.0001	4 mill	0.0084	100	0.0094	0.0094	0.0094

and the remaining parameters are varied according to these equations:

- $\Delta r = 2\Delta x$
- $\Delta t = \frac{0.65 * \Delta y}{60m/s}$

The different grid spacings that have been used are shown in table 9, together with the relationship between Δx and ϵ .

Table 9: Grid spacings and relationship between ϵ and the grid spacing for constant $\epsilon = 0.0084m$. Δx refers to the finest grid in the domain, and the grid cells are cubical. The two finest grid resolutions have been simulated using an extra grid refinement. Simulations have also been performed without any grid refinement down to a grid spacing of $\Delta x = 0.012m$.

$\Delta x[m]$	$\epsilon/\Delta x$
Three grid refinements	
0.047	0.18
0.031	0.27
0.023	0.36
0.019	0.45
0.012	0.68
0.0094	0.89
0.0074	1.13
0.0062	1.34
0.0053	1.58
0.0046	1.81
Four grid refinements	
0.0094	0.89
0.0047	1.79
0.0037	2.27
0.0031	2.71

In order to run the two finest grid resolutions, an additional grid refinement has been added at a distance of 0.18 diameters from the rotor. Simulations were also performed where the innermost refinement was 0.12 diameters from the rotor, but this produced clearly visible oscillations in the solution.

INCLUDING GRID REFINEMENT

The finest and coarsest simulations run with three grid refinements;

Δt [s]	Cells	ϵ [m]	b	Δx [m]	Δy [m]	Δz [m]
0.0005	33 000	0.0084	20	0.047	0.047	0.047
0.00005	32 mill	0.0084	200	0.0047	0.0047	0.0047

The finest and coarsest simulations run with four grid refinements;

Δt [s]	Cells	ϵ [m]	b	Δx [m]	Δy [m]	Δz [m]
0.0001	1 mill	0.0084	100	0.0094	0.0094	0.0094
0.000034	27 mill	0.0084	300	0.0031	0.0031	0.0031

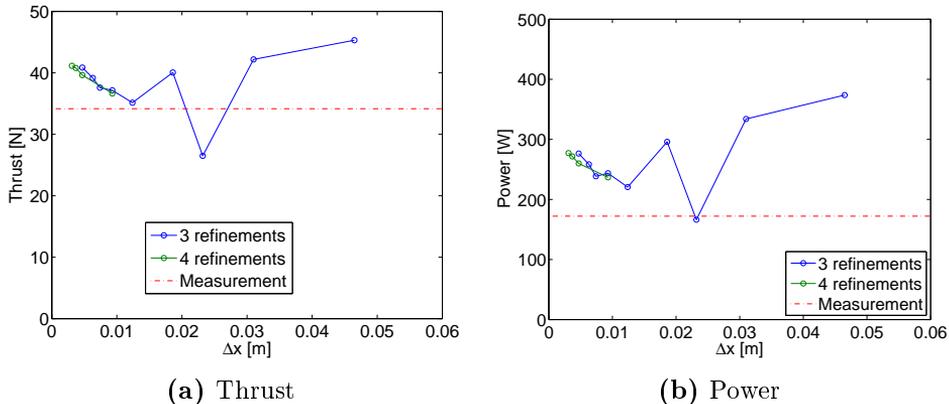


Figure 35: Thrust and power for varying grids with grid refinement around the rotor. Results have been plotted for three and four grid refinements, Δx indicates the finest grid in the domain (at the rotor). $\epsilon = 0.0084m$, the other parameters are changed with the grid.

Calculations have been performed with grid refinement for grid spacings between $\Delta x = 0.031m$ and $\Delta x = 0.0074m$, and figure 35 shows the thrust and power results. Both thrust and power are declining from the coarse to the fine grid, but there is no clear convergence.

The simulations with four grid refinements seem to underestimate the thrust and power slightly compared to the corresponding simulations for three grid refinements ($\Delta x = 0.0046m$ and $\Delta x = 0.0094m$). The results from the simulations performed with four refinements for $\Delta x = 0.0037m$ and $\Delta x = 0.0031m$ can be extrapolated by using the difference in thrust and power between three and four grid refinements for $\Delta x = 0.0046m$ and $\Delta x = 0.0094m$. If this is done, the graph for three grid refinements in figure 35 will continue to rise for fine grids. As for the simulations performed with three grid refinements, the trend for thrust and power for fine grids with

four grid refinements is increasing, and the graphs do not seem to converge.

One point in the thrust and power graphs is very inconsistent with the general trend of the graphs. When $\Delta x = 0.023m$ the thrust is $26.5N$ and the power is $192.2W$, far below the other values. This discrepancy has been described in detail in appendix C.

WITHOUT GRID REFINEMENT

The finest and coarsest grid resolutions for the simulations run without grid refinement is shown below;

Δt [s]	Cells	ϵ [m]	b	Δx [m]	Δy [m]	Δz [m]
0.0005	33 000	0.0084	20	0.047	0.047	0.047
0.00013	29 mill	0.0084	75	0.012	0.012	0.012

Calculations have been performed without grid refinement for grid spacings between $\Delta x = 0.047m$ and $\Delta x = 0.012m$. Calculations for finer grid spacings was not available due to the large memory usage of OpenFOAM.

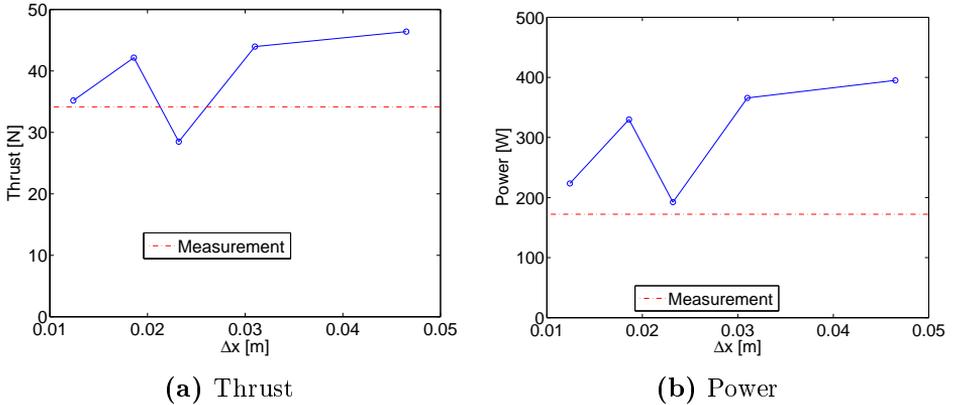


Figure 36: Thrust and power for varying grids without grid refinement around the rotor. $\epsilon = 0.0084m$, the other parameters are changed with the grid.

The thrust and power results are shown in figure 36. As for the coarser grids in figure 35, the thrust and power decrease with increasing grid resolutions. The inconsistency for $\Delta x = 0.023m$ also appears from the simulations without grid refinement, and this is (as mentioned before) regarded in detail in appendix C.

In appendix C, it is shown that the dip is dependent on the grid resolution and ϵ . Several simulations were performed for larger and smaller grid resolutions, showing that the dip starts at a grid spacing of $\Delta x = 0.02m$ and ends at $\Delta x = 0.027m$. Oscillations occur in the simulation for these grid resolutions, and the reason for this discrepancy is thought to be a numerical error.

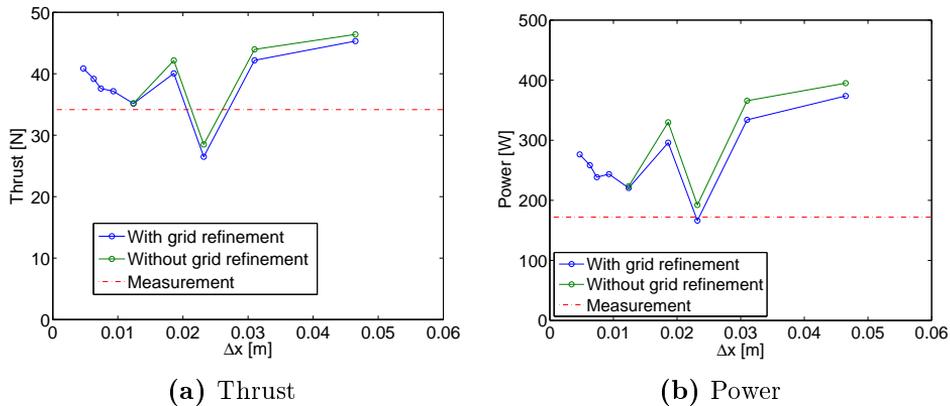


Figure 37: Thrust and power for varying grids with and without grid refinement around the rotor. $\epsilon = 0.0084m$, the other parameters are changed with the grid.

As a comparison, figures 35 and 36 are plotted together in figure 37. For this case, the refinement procedure seems to be a good tool to use in order to decrease the computational time, because the solution seems to be dependent on the grid density at the rotor and not in the far field. For the coarse grids, the cases with refinement calculates a slightly lower thrust

than without refinement, but for the finest grid with $\Delta x = 0.012m$, the differences in thrust and power are a negligible 0.1%. This difference is a lot smaller than what was seen in the last section, although the only parameter changed is ϵ . Possible explanations for this is given in appendix A.

The main problem here is that the thrust and power in figure 37a does not seem to converge. As mentioned by Troldborg [25], the actuator line model will always have a certain dependency on the grid resolution, but without a converged solution it is difficult to use the results. To investigate this, simulations were performed where only the grid in the x -direction or in the rotor plane was changed. The starting grid to this analysis was chosen to be $\Delta x = \Delta y = \Delta z = 0.0074m$, because the grid is not too computationally heavy, but fine enough to ensure that numerical oscillations are not significant in the results.

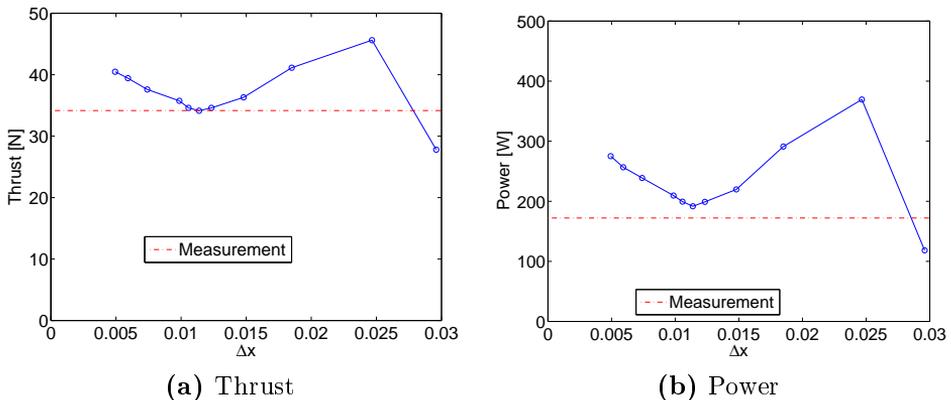


Figure 38: Thrust and power for a constant grid resolution in the rotor plane and for different Δx , with $\Delta y = \Delta z = 0.0074m$. $\epsilon = 0.0084m$, the other parameters are changed with the grid.

In figure 38 only the grid in the streamwise (x) direction was changed, and the grid in the rotor plane is kept at $\Delta y = \Delta z = 0.0074m$. It is clear that, as concluded previously, the grid in the streamwise direction has a large

effect on the solution. The trend in figure 38 is closely linked to the trend in figure 35. The thrust and power curves have a convex shape, and one point for a coarse grid that is far off from the others (here for $\Delta x = 0.03m$).

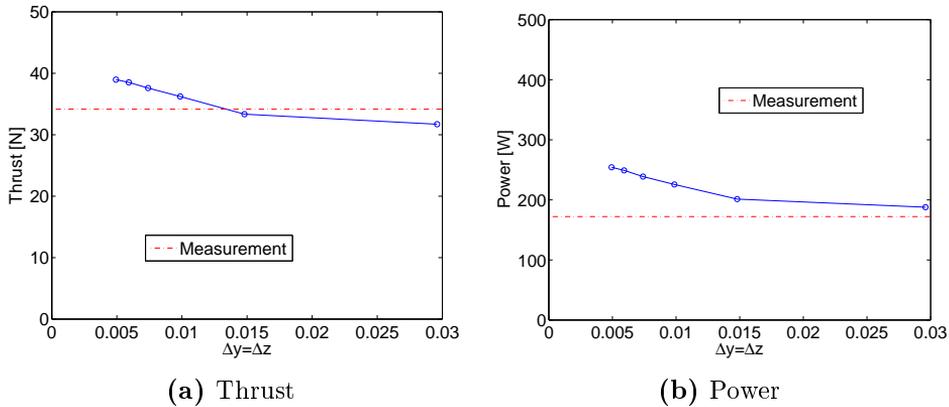


Figure 39: Thrust and power for a constant grid spacing in the x -direction and for different Δy and Δz , with $\Delta x = 0.0074m$. $\epsilon = 0.0084m$, the other parameters are changed with the grid.

Figures 39 shows the thrust and power graphs when only the grid in the rotor plane has been changed, and the grid in the x -direction is $\Delta x = 0.0074m$. This has a clear effect on the thrust and power, but the variation is smaller than in figure 38 where the grid in the x -direction is changed. In addition, the trend in figure 39 is not linked to the trend in figure 35, except that the thrust and power increase for fine grids. This shows that the grid in the x -direction is the main cause of the trend in figure 35, including the dip in thrust and power for one of the grid resolutions.

For fine grids in the x -direction, thrust and power are increasing, and for fine grids in the y - and z -direction, thrust and power are increasing. None of the graphs in figures 38 and 39 seem to converge, even for very small grid cells. This is consistent with the results from refining the grid in the entire domain, the thrust and power increase for fine grid resolutions. This

increase appears even when all the requirements found at the start of the parameter study are fulfilled.

Appendix A shows that for some of the simulations thrust stabilizes after $t = 0.5s$, while for others thrust increase after $t = 0.5s$ and stabilizes for $t = 1s$. The simulations were thrust stabilizes after $t = 0.5s$ are linked to a decrease in average thrust, while the opposite is true for the simulations with a rise in thrust after $t = 0.5s$. This accumulation of thrust over time is linked to the differences in the results from the refined and non-refined grids. It has also been linked to the lack of a converged solution, because the rise in thrust after $t = 0.5s$ appear for fine grid resolutions.

Possible explanations to the lack of convergence include

- According to Martinez et al. [11], the solution is more dependent on the grid for smaller ϵ . Perhaps the requirement found earlier that $\epsilon < 2\Delta x$ is too crude for a small ϵ and that the grid should be refined further. The grid is already very fine, and simulations for an even finer grid would be too computationally heavy.
- The implementation of the actuator line code used in this thesis was originally created for a LES turbulence model. The simulations in this thesis have been performed with RANS equations and a $k - \epsilon$ model, which may not yield the same results. It would be interesting to perform these same simulations using LES instead of RANS, and other turbulence models such as the $k - \omega$ model.
- There could be a bug in the numerical code. An indication that such a bug exists is shown in appendix A, where the results with and without grid refinement are compared.
- The actuator line code is too dependent on the grid resolution and ϵ , and can be further developed to take care of the models dependency on the grid resolution.

COMPARISON OF THE RESULTS FROM $\epsilon = 2.5\Delta x$ AND $\epsilon = 0.0084m$

In the following, a comparison between the results from this section and the last section will be given. In this analysis, the results from the last section will be named part 2 results, while the results from this section are called part 3 results.

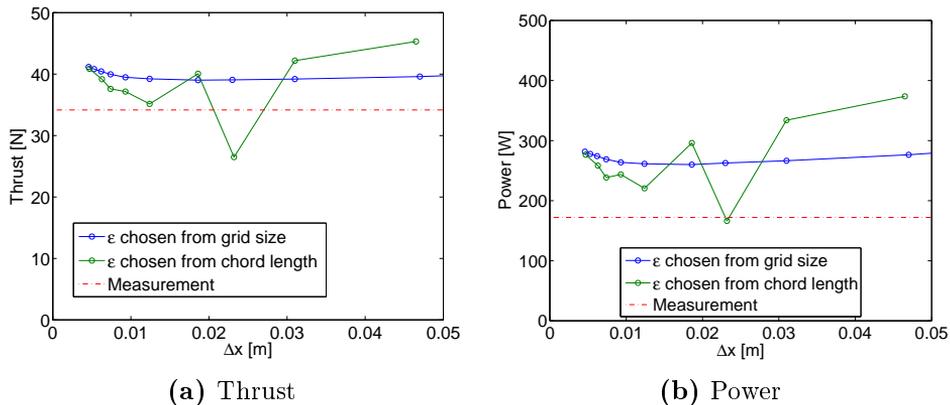


Figure 40: Thrust and power result from the last section, where $\epsilon = 2.5\Delta x$, and this section, where $\epsilon = 0.0084m$, with grid refinement around the rotor.

Figure 40 shows the thrust and power result from this section and the last section. Part 2 results clearly are less oscillating than part 3 results. In addition, the value of the thrust and power from part 2 are mostly higher than from part 3, and therefore further from the measured values.

For the finest grids in figure 40, the results from part 2 and 3 simulations coincide. This is because $\epsilon = 2.5\Delta x$ for part 3, and $\epsilon \sim 2\Delta x$ for the finest grids in part 2. The finest grid simulated in part 2 is $\Delta x = 0.0046m$. At this grid resolution the difference between part 2 and part 3 results are only 0.7% for thrust and 1.9% for power. This result is as expected, since the simulations are almost identical.

Although thrust and power do not converge for any of the two ways of

choosing epsilon, the conclusion for this thesis is that ϵ should be linked to the chord length of the blade. This conclusion is governed by two factors, (1) choosing ϵ from the chord length produced results closer to the measured values and (2) choosing ϵ from grid spacing does not ensure that the distributed forces represent the blade geometry of the turbine that is being simulated. The latter was shown more in detail in section 5.1, where a visualization of ϵ has been made.

The best result was achieved in this section when changing only the grid in the x -direction, shown in figure 38. For $\Delta x = 0.011m$ and $\Delta y = \Delta z = 0.0075m$ both thrust and power reach a minimum point close to the measured values. For this simulation, the grid in the x -direction is about 1.5 times the grid in the rotor plane. Also in section 5.2.1, the thrust and power were closer to the measured values for the non-cubical grid. This leads to the conclusion that a cubical grid with $\Delta x = \Delta y = \Delta z$ is not necessarily best suited for this model, although it is important to ensure that the grid spacing in the streamwise direction is not chosen too large compared to the rotor plane grid resolution.

The results in this thesis show that $\epsilon > 2\Delta x$ is a too strict relation to achieve good results compared to the measured values for this set-up. When ϵ is chosen from the chord length, simulations for several grid resolutions should be performed to find the minimum results of thrust and power. The grid in the streamwise direction is most important, and good results may be found by changing this grid resolution independently of the grid in the rotor plane. It is also important to make sure that the simulation is not in a dip with a lot of oscillations in the result, as the grid spacing $\Delta x = 0.023m$ is an example of in this section.

$\epsilon/\Delta x$ is only 0.76 for the best simulation, which is definitely below the previously recommended $\epsilon > 2\Delta x$. The conclusion is therefore that although $\epsilon < 2\Delta x$ may lead to numerical oscillations in the simulations, demanding $\epsilon > 2\Delta x$ produces too large outputs of thrust and power. These results will in section 5.3 be compared to the experimental values from "Blind test 1".

5.2.4 PART 4 - PROPOSAL FOR BEST USE OF THE ACTUATOR LINE MODEL

A summary will be given here, showing the most important conclusions of the parameter study above, and defining specific guidelines on how to determine the parameters. The proposal given assumes a Cartesian grid, and is based on several simulations performed using RANS and a $k - \epsilon$ turbulence model for a model turbine in a wind tunnel.

NUMBER OF BLADE ELEMENTS

The number of blade elements should be larger than the amount of grid cells across one blade in the rotor plane. To be certain, the blade element size may be chosen as half the grid resolution, i.e.

$$\Delta r \leq \frac{1}{2} \Delta y. \quad (60)$$

TIME STEP

The time step is chosen to ensure that the flow does not pass more than one grid cell face during one time step. The limiting factor is the tip speed of the blade, and the time step should be smaller than the time the tip uses to pass through one grid cell in the rotor plane, i.e.

$$\Delta t < \frac{\Delta y}{\text{tip speed}}. \quad (61)$$

An additional margin for the time step is needed since the grid is Cartesian. For this thesis about 0.65 of the calculated maximum time step was used, which proved to be sufficient.

ϵ

Previous recommendations from the literature often connect ϵ to the grid resolution and recommend $\epsilon > 2\Delta x$. In this thesis it is argued that ϵ should be chosen from the chord length of the blade in order to distribute the forces in a manner that represents the blade size well. This may be achieved by choosing an average or representative chord length c , and

$$\epsilon = \frac{c}{4.3}. \quad (62)$$

In this project the representative chord length was chosen $2/3$ from the base of the blade, where most of the force is generated.

GRID RESOLUTION

The actuator line model is highly dependent on the grid resolution, and a grid independent solution was not achieved in this thesis. It was shown that $\epsilon > 2\Delta x$ (as recommended often in the literature) yields solutions without much noise, but the thrust and power values are too large compared to the measurements.

To get a solution that well represents the measurements, the following procedure should be followed:

1. Run simulations for several grid resolutions. The time step and number of blade elements are varied with the grid according to the recommendations above.
2. Check the simulations for grid resolutions that produce unexpected results with a lot of noise.
3. Choose a suitable grid in the rotor plane and perform simulations where only the streamwise grid is changed, this can advantageously be done for multiple grid resolutions in the rotor plane.
4. Choose the simulation that produces the least thrust and power.

5.3 BLIND TEST 1

In this section, a comparison of the best simulation in the parameter study will be made to the experimental values of the "Blind test 1", and to the simulations performed by the other "Blind test 1" candidates. This section will test the developed guidelines, and see how well the actuator line code performs for the "Blind test 1" experiment. First the thrust and power output will be discussed, then the velocity and turbulent kinetic energy one diameter behind the rotor will be presented and discussed.

Following this, an implementation of the hub and tower has been made by using a code created by Tommy Fredriksen, and results from these simulations is presented. For the simulations with hub and tower, the velocity deficit is presented for one, three and five diameters behind the rotor, followed by the kinetic turbulent energy at the same distances. A comparison to the turbine measurements and to the "Blind test 1" candidates will be performed.

The following parameters are used in the set-up;

Δt [s]	Cells	ϵ [m]	\mathbf{b}	Δx [m]	Δy [m]	Δz [m]
0.00008	10.8 mill	0.0084	125	0.011	0.0075	0.0075

The resulting thrust and power is shown in table 10, and compared to the experimental values of "Blind test 1".

	Thrust [N]	C_T	Power [W]	C_P
Simulated	34.13	0.89	191.71	0.5
Experimental	34.16	0.89	172.16	0.45
Difference	0.088%		10.2%	

Table 10: Thrust and power from the "Blind test 1" simulation and experiment.

The thrust is simulated perfectly, but the power is overestimated.

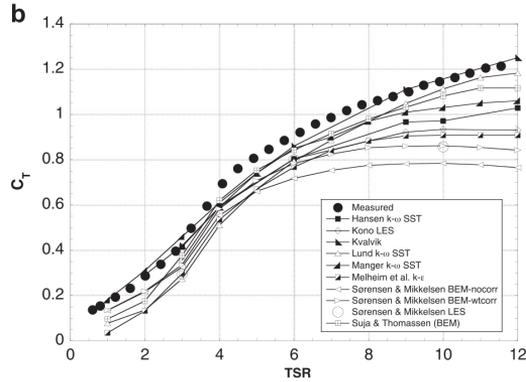


Figure 41: Experimental and simulated values of C_T for various TSR from "Blind test 1". The black circles indicate the experimental values [8]. The results from this study can be found in table 10.

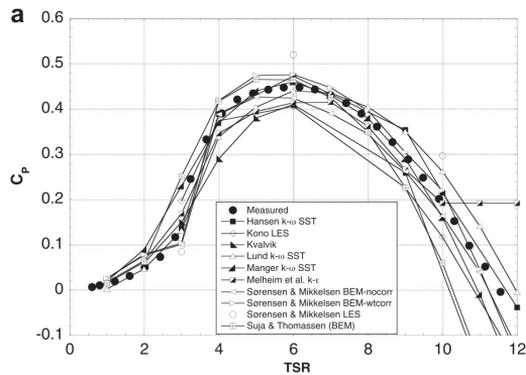


Figure 42: Experimental and simulated values of C_P for various TSR from "Blind test 1". The black circles indicate the experimental values [8]. The results from this study can be found in table 10.

Figure 41 and 42 shows the thrust and power coefficients from the "Blind test 1" experiment. The black dots indicate the experimental values, while

the other graphs represent the simulations. Additional information about the candidates and their simulations can be found in [8]. All the candidates underestimated the thrust, while power was both under- and overestimated. This shows clearly that the actuator line code overestimates both thrust and power compared to other simulation tools. None of the candidates at the "Blind test 1" estimated the thrust as well as in this thesis.

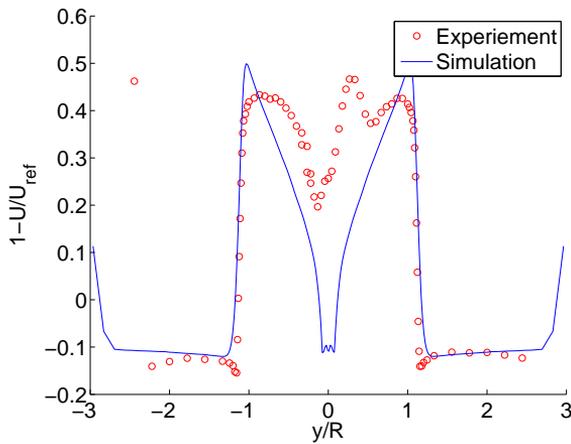


Figure 43: The velocity deficit one diameter behind the rotor, along the y -axis.

Figure 43 shows the velocity wake one diameter behind the turbine. The blue line indicates the simulations performed in this thesis, while the red dots are the experimental values. Figure 43 has several features that is expected of a velocity wake from a turbine. A speed up in the wind flow occurs at the sides of the rotor, and the wind is slowed down again at the wall. This is because the wind turbine represents an obstacle for the wind flow, and the wind is pushed to the sides of the rotor, gaining a velocity larger than the inlet velocity. At the sides of the turbine, the simulation is in good agreement with the measurements.

Figure 43 also shows an agreement between the simulation and the exper-

iment behind the tip of the rotor, where the wind is slowed down. The velocity deficit close to the tip is larger for the simulated value than the experimental, and the experimental profile has a more round shape close to the tip. This may be due to the grid resolution in the simulation being too coarse to resolve all the details in the tip vortices.

The simulation has clearly overestimated the velocity in the middle of the wake. This is due to the simulations being performed without a hub, and the wind is not slowed down between the roots of the blades. In addition, the experimental value has an asymmetric profile, which the simulation does not replicate. The asymmetric profile is created by the tower of the turbine, which is also not included in the simulations.

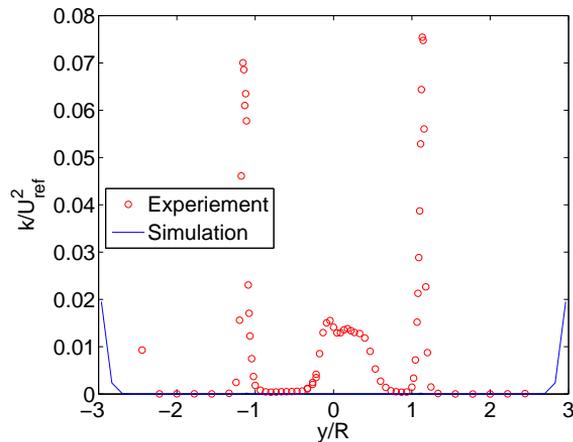


Figure 44: The turbulent kinetic energy one diameter behind the rotor, along the y -axis.

Figure 44 shows the kinetic turbulent energy one diameter behind the rotor. The blue line indicates the simulated value, while the red dots are the measured values from the experiment. The simulated values are so low that they are not visible with this y -axis (except close to the wall). The figure has been plotted again with a logarithmic y -axis shown in figure 45 to show

the details of the simulated wake.

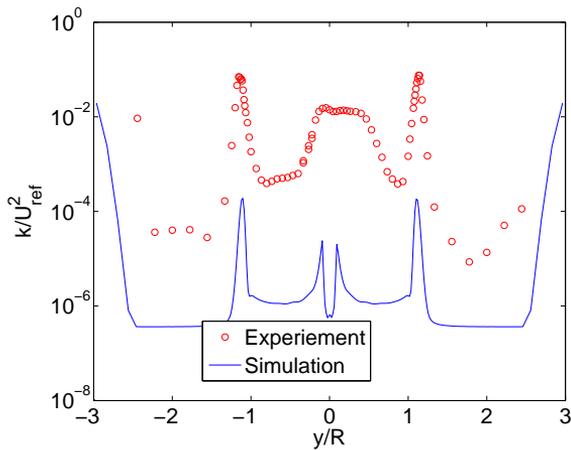


Figure 45: The turbulent kinetic energy one diameter behind the rotor, along the y -axis, with a logarithmic y -axis.

In figure 45, the trend in the simulated turbulent kinetic energy is similar to the measured trend. At the tip of the blade a large turbulence peak exists. These peaks come from the tip vortices, which are clearly underestimated. High turbulence is also present behind the center of the rotor, and this turbulence is created from the root vortices. The simulated profile has the root vortices, but it also has a fall in turbulence right at the center. This is because the hub is not included in the simulations and the wind flows right through the center of the turbine. For the measured values, the hub creates additional turbulence and the value of the turbulence is almost uniform at the center. The measured values show that the turbulence is asymmetrical, while the simulated turbulence profile is symmetrical. This can again be traced back to the fact that the actuator line code does not include the tower of the turbine.

As to the value of k , it is not modelled good enough. To look at the extreme case, the experiment has a maximum nondimensionalized value of 0.07541,

and at this position the simulated value is $1.8918 * 10^{-4}$, which is 400 times lower. Even in the area between the root and tip vortices, the measured value is about $4 * 10^{-4}$, while the simulation gives a k of $1.5 * 10^{-6}$, which is 250 times lower than the measured.

There are two main differences between the simulated and experimental wake and turbulence profiles;

1. The simulated profiles are symmetric around the z -axis, while the experimental profile has non-symmetric features. This non-symmetry is created by the tower, which has not been implemented.
2. The simulated velocities in the middle of the wake are higher than the experimental values, and the turbulent kinetic energy is largely underestimated. The hub has not been implemented, which means that the flow without hindrance passes through the center of the rotor. The hub in the model turbine blocks the wind from passing through the middle of the rotor, and creates a significant velocity deficit and turbulence behind the center of the turbine.

An implementation of the hub and tower in the actuator line code has been performed by Tommy Fredriksen during his work on his master thesis spring 2013. They are both created using actuator line elements with a given length and epsilon, in order to produce a representative size for the hub and tower. The implementation he created have been added to the simulation in this study to see how it affects the results.

A more accurate replication of the hub and tower could be created by using the mesh, but the same arguments for creating the rotor blades from the actuator line elements also hold here. Creating the hub and tower from the mesh brings forth two disadvantages (1) the mesh around the hub and tower have to be adapted and fitted to the shapes of the hub and tower and (2) the grid cells would have to be fine enough to resolve the boundary layer forming around the hub and tower.

Adding the hub and tower only had a very small effect on the average thrust and power output. The thrust and power are $T = 34.12N$ and $P = 191.99W$,

which is a negligible difference of 0.03% and 0.15% from previously. The hub and tower did however have a larger influence on the wake shapes behind the turbine, especially the turbulent kinetic energy.

In the following, the wake shapes behind the turbine will be presented from the simulation including the hub and tower. First the velocity wake will be presented, one, three and five diameters behind the turbine. After this the turbulent kinetic energy will be presented at the same distances from the turbine. The wake results from the other simulations in the "Blind test 1" have been added for comparison.

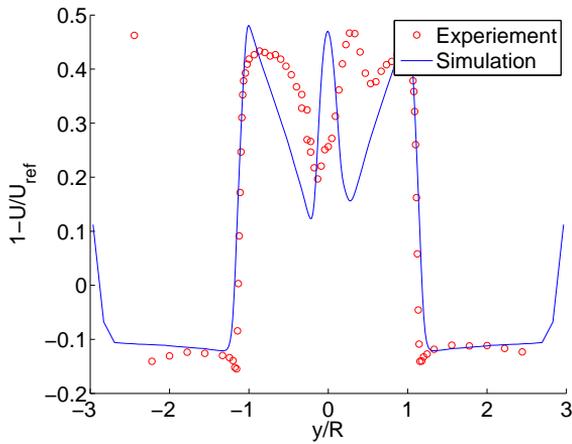


Figure 46: The velocity deficit one diameter behind the rotor, along the y -axis. The hub and tower are included in the simulation.

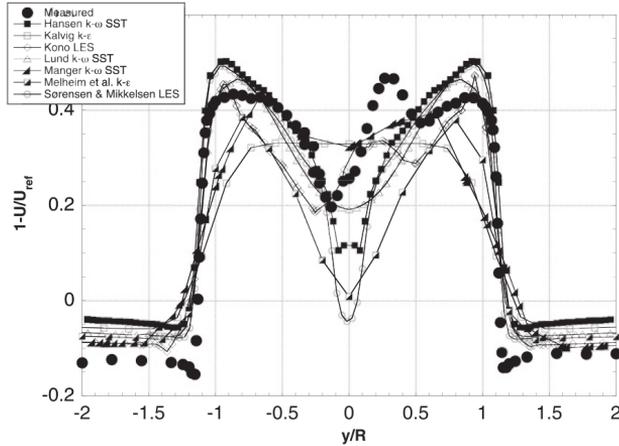


Figure 47: The velocity deficit one diameter behind the rotor, along the y -axis, "Blind test 1" results [8].

Figure 46 shows the resulting velocity profile one diameter behind the rotor. It is clear that the hub and tower slow the wind down behind the middle of the rotor, and there are also asymmetrical effects in the wake shape due to the tower. In addition, the peak in velocity deficit behind the tip of the blades is slightly lower and the shape is more round, although the peak velocity deficit is still overestimated.

Compared to the other "Blind test 1" candidates shown in figure 47, the actuator line code performs well. The wake width, placement of the peak velocity deficit and velocity on the sides of the turbine is modelled better than many of the other candidates.

Since the shapes of the hub, tower and blades are not modelled accurately, a perfect correlation between the simulated and the measured wake shapes is not expected. Taking this into consideration, the simulated wake shape in figure 46 is a good representation of the actual wake shape behind the model turbine.

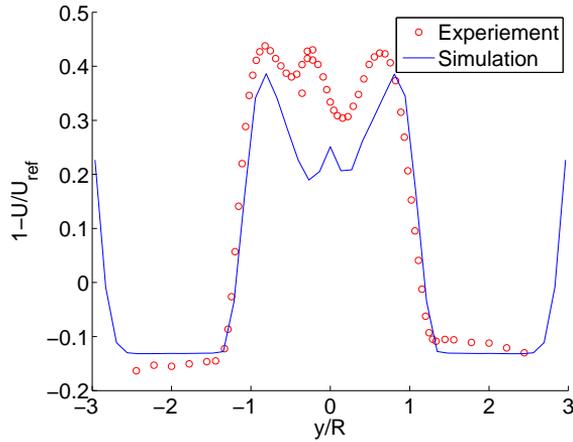


Figure 48: The velocity deficit three diameters behind the rotor, along the y -axis. The hub and tower are included in the simulation.

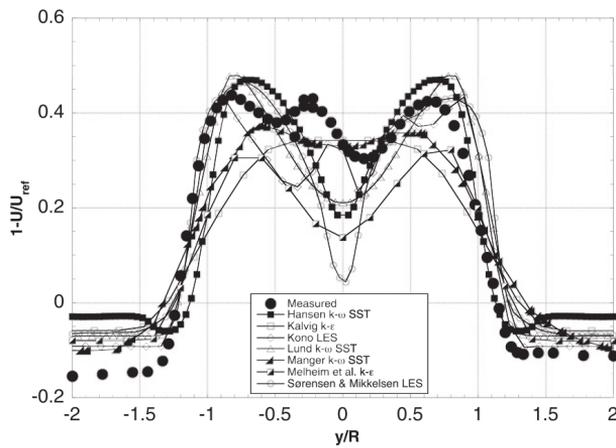


Figure 49: The velocity deficit three diameters behind the rotor, along the y -axis, "Blind test 1" results [8].

Figure 48 shows the velocity profile three diameters behind the rotor. The

width of the wake is still very well simulated, but the velocity deficit is slightly lower than the measurement. One diameter behind the rotor the peak velocity was higher than the measurements, but the velocity deficit has decreased more in the simulation than in the experiment. The simulated velocity diffusion is larger than the measured velocity diffusion.

The grid is only refined down to two rotor diameters behind the turbine. Because of this, the mesh at three diameters behind the turbine is coarse, and the details in the velocity wake are not resolved. This probably leads to a larger diffusion of the wake. To get a more accurate representation of the wake shapes behind the rotor, the mesh should be refined further behind the rotor.

The velocity in the middle of the rotor is not modelled as well as some of the other "Blind test 1" candidates shown in figure 49, but the velocity wake width is modelled better than many.

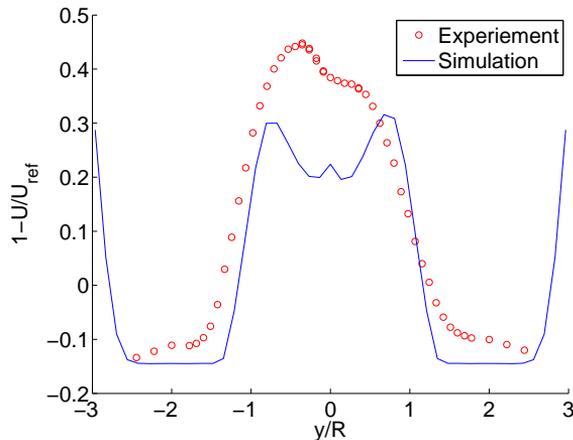


Figure 50: The velocity deficit five diameters behind the rotor, along the y -axis. The hub and tower are included in the simulation.

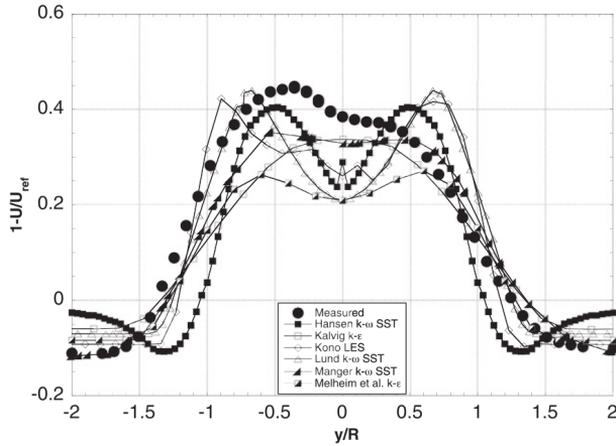


Figure 51: The velocity deficit five diameters behind the rotor, along the y -axis, "Blind test 1" results [8].

Figure 50 shows the velocity profile five diameters behind the rotor. Again the velocity deficit is underestimated, and the measured profile has a convex shape behind the center of the rotor, while the simulated profile has a concave shape.

The experimental wake is asymmetrical, while the simulated profile has become symmetrical. The width of the wake is still well represented by the simulation.

For this velocity wake profile, the other "Blind test 1" candidates mostly performed better. The main reason for this may be, as mentioned previously, that the wind flow is outside of the refined region, and the mesh is too coarse to simulate the details in the wake.

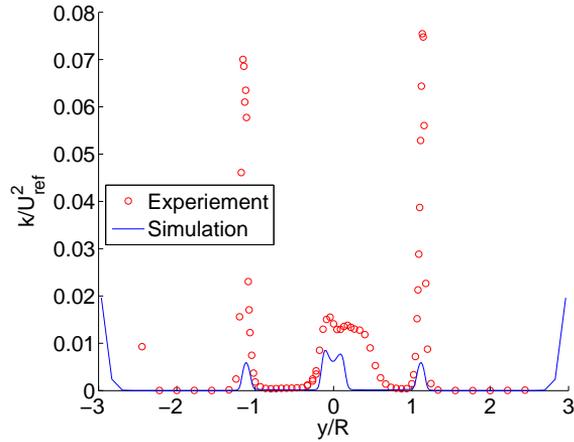


Figure 52: The turbulent kinetic energy one diameter behind the rotor, along the y -axis. The hub and tower are included in the simulation.

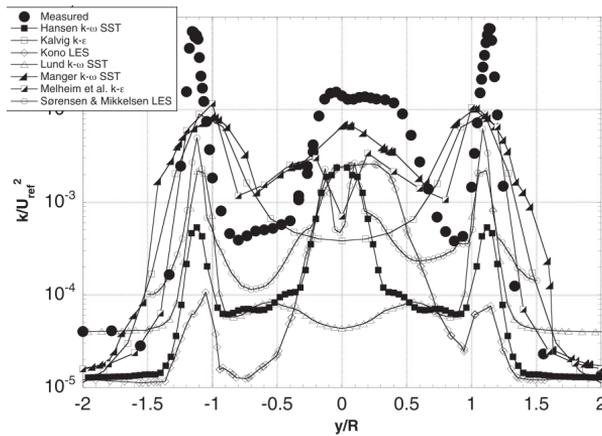


Figure 53: The turbulent kinetic energy one diameter behind the rotor, along the y -axis, "Blind test 1" results [8]. Notice that the y -scale is logarithmic.

The turbulent kinetic energy for the simulations with hub and tower is shown in figure 52. The hub and tower have created a lot more turbulence than in the original simulation, both behind the rotor center and behind the tip of the blades. The simulated value of k behind the center of the rotor is half of the measured value, and there is no significant drop in turbulence right at the center of the rotor as can be seen in the simulated turbulence in figure 44. Although the width of the simulated root vortex is too small compared to the experiment, this is a much better result.

The simulated value behind the tip of the blade corresponding to the largest measured value is 0.0059, i.e. only 12 times lower than the measured value of 0.0754. Although the difference is still high, and there is an order of magnitude between the two results, the improvement from the simulations without hub and tower is large. In addition, in the area between the tip and root vortices, the relationship between the simulated and measured value is around two.

Adding the hub and tower had a positive effect on the velocity deficit behind the turbine, but the turbulence was affected enormously. Going from at the largest underestimating the turbulence 400 times to only 12 is a huge improvement. In addition, none of the "Blind test 1" participants were close to simulating correct values for these peaks, the closest one was right above 0.01, i.e. 1/7 of the measured value (see figure 53). More improvements need to be made, especially for simulating the tip vortices.

Many experiments have shown that turbulence is difficult to simulate, and in particular low turbulent flows, which the "Blind test 1" is an example of. It is possible that the tip vortices are underestimated due to small scale effects of the model turbine, which the actuator line code does not take into consideration. It would be interesting to see the turbulence compared to measurements for a large scale turbine simulated with the actuator line code, including the hub and tower.

The underestimation of the velocity deficit may be caused by an underestimation of the turbulence behind the rotor. Turbulent effects will slow the wind down and create a larger velocity deficit.

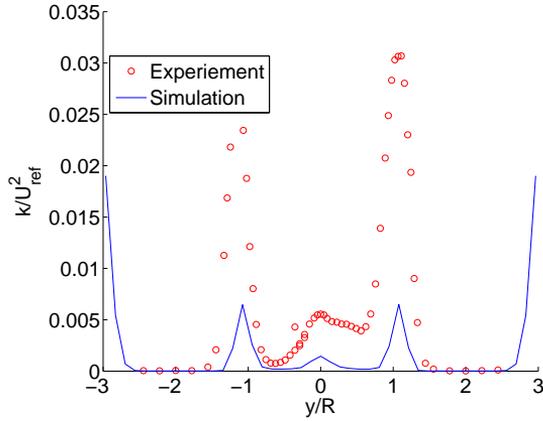


Figure 54: The turbulent kinetic energy three diameters behind the rotor, along the y -axis. The hub and tower are included in the simulation.

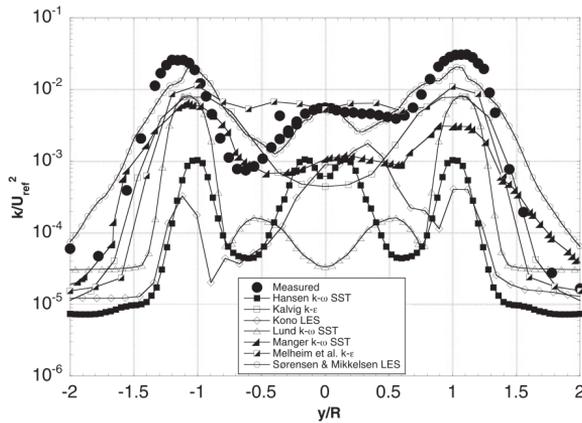


Figure 55: The turbulent kinetic energy three diameters behind the rotor, along the y -axis, "Blind test 1" results [8]. Notice that the y -scale is logarithmic.

Figure 54 shows the turbulent kinetic energy profile three diameters behind the rotor. The measured peaks have fallen in magnitude and become wider due to the spanwise turbulent diffusion. The decay of the peaks happens faster for the measurement than for the simulation, and the difference between the simulated and measured turbulent profile is therefore lower.

This is opposite of what can be seen for the velocity deficit three diameters behind the turbine. The velocity profile has a larger difference from the measured value at three diameters behind the turbine than one diameter behind. The velocity dissipated faster than the measured values, while the simulated turbulent kinetic energy has dissipated slower than the measurements.

The simulations from the "Blind test 1" are shown in figure 57. Most of the candidates have simulated a smaller dissipation of the turbulent kinetic energy than the measurements from one to three diameters behind the turbine. In addition, most of the "Blind test 1" candidates still underestimate the turbulence three diameters behind the turbine.

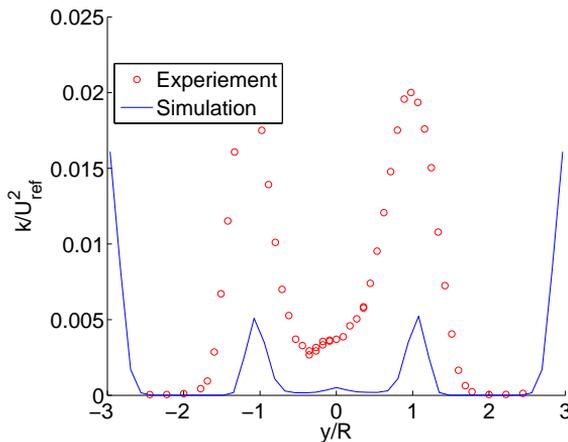


Figure 56: The turbulent kinetic energy five diameters behind the rotor, along the y -axis. The hub and tower are included in the simulation.

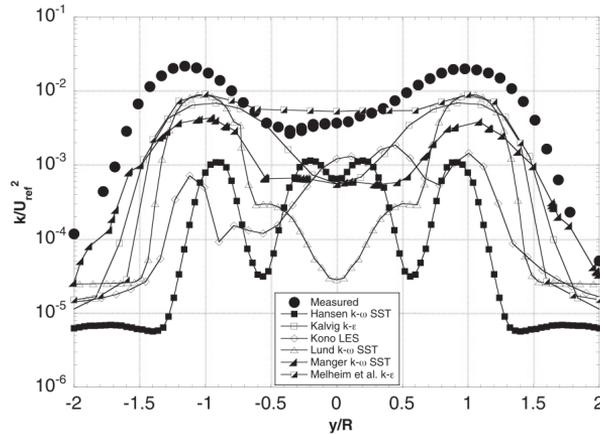


Figure 57: The turbulent kinetic energy five diameters behind the rotor, along the y -axis, "Blind test 1" results [8]. Notice that the y -scale is logarithmic.

In figure 56, the turbulent kinetic energy five diameters behind the rotor is shown. Again it can be seen that the turbulent peaks have decreased and widened spanwise. The simulation is now closer to the measurement, also because the turbulence from the root vortices is lower than from the tip vortices, which is not the case one diameter behind the rotor.

All of the other "Blind test 1" candidates underestimate the peak turbulent kinetic energy, but many of them model the turbulence better than the simulations in this study.

The conclusions are that by using the developed guidelines good results were achieved. The actuator line code models the thrust well, but the power is overestimated, and the wake shapes simulations are good.

The hub and tower had a large influence on the resulting wake profiles, especially the turbulent kinetic energy. After the hub and tower was implemented the wake profiles were well simulated one diameter behind the turbine, especially compared to the other "Blind test 1" candidates. If the

turbulent wake is of interest, the hub and tower need to be simulated.

At three and five diameters behind the turbine, the wake profiles did not correspond as well with the measurements as they did at one diameter behind the turbine. This is because the grid is not refined further than two diameters behind the rotor.

It was also shown that the hub and tower modelled in this thesis had only a very small influence on the thrust and power, 0.15% maximum. The hub and tower are therefore not important if only thrust and power is of interest. Tests should be made to see if this result is also applicable for larger turbines.

Given the modelling circumstances, these results are very good. Fairly easy models are used to simulate a very complex phenomenon, i.e. the $k - \epsilon$ for turbulence and the actuator line model for the turbine. The wake behind a rotor is a complicated three dimensional and fully turbulent periodic flow that even the most sophisticated CFD codes experience difficulties when trying to model. The "Blind test 1" has a low turbulence, which is more difficult to model than highly turbulent flows.

5.4 BLIND TEST 2

The "Blind test 2" was explained in section 4, and the schematic of the numerical set-up was shown in figure 13. The "Blind test 2" experiment is very close to the "Blind test 1", but an additional turbine has been added to the wind tunnel. In the following, the upstream turbine will be named T1, while the downstream turbine is labelled T2. The "Blind test 2" results have been collected from a presentation by Lars Sætran [23].

The numerical set-up for "Blind test 2" was created using the time step, ϵ , number of blade elements and grid from "Blind test 1". Simulations were performed with varying grid spacing in the x -direction, as was suggested in the proposed usage of the actuator line model developed in this thesis.

During the time span of this study, extensive testing with different grid resolutions for the "Blind test 2" was not performed, and only two simulations were run with varying grid in the x -direction. More simulations should be performed to find out if the chosen grid produces the minimum thrust and power.

The resulting grid producing the lowest thrust and power output is $\Delta y = \Delta z = 0.0074m$ in the rotor plane, and $\Delta x = 0.017m$ in the streamwise direction. The relationship between the grid in the rotor plane and in the x -direction is $\Delta x/\Delta y = 2.35$. The relationship between ϵ and the grid in the x -direction is now $\epsilon/\Delta x = 0.48$.

A comparison to the experiment will first be done for thrust and power, and then for the velocity and turbulent kinetic energy wake one, two and a half and four diameters behind the downstream rotor.

The simulations from the "Blind test 2" candidates have been added for comparison, but there are some differences from the "Blind test 1". The velocity wake is given by U/U_{ref} along the y -axis, while the simulations for this study are shown (as for "Blind test 1") as $1 - U/U_{ref}$. The turbulent kinetic energy wake is given as u' along the z -axis, while the simulations for this study are shown (as for "Blind test 1") as k/U_{ref}^2 along the y -axis.

The following parameters are used in the set-up;

Δt [s]	Cells	ϵ [m]	\mathbf{b}	Δx [m]	Δy [m]	Δz [m]
0.00008	10.8 mill	0.0084	125	0.017	0.0075	0.0075

The grid refinement introduced in section 4 has been extended in the x -direction to include the first turbine, and also extended behind the second turbine. From the placement of the original turbine (see figure 15 for the original grid refinement), the grid refinements in the x -direction are now

1. 3.5 diameters in front, 6 diameters behind
2. 3 diameters in front, 5 diameters behind
3. 2.5 diameters in front, 4.5 diameters behind

This refinement process ensures that 2.5 diameters behind the second turbine is within the finest grid refinement, while four diameters is outside of the grid refinement region. It will be interesting to see how this affects the wake modelling.

Table 11 shows the thrust and power output for the two turbines from the experiment and the simulation.

		Thrust [N]	C_T	Power [W]	C_P
T1	Simulated	38.03	0.99	178.44	0.46
	Experimental	31.9	0.83	176.77	0.46
	Difference	16%		0.94%	
T2	Simulated	14.54	0.38	45.54	0.12
	Experimental	14.22	0.37	46.11	0.12
	Difference	2.1%		1.24%	

Table 11: Thrust and power from the "Blind test 2" simulation and experiment.

The results are very good, and especially the power is not as overestimated as was seen in the "Blind test 1" results in the last section. The thrust for the upstream turbine is too high, at 16% above the experimental value. This difference is large and should be investigated further.

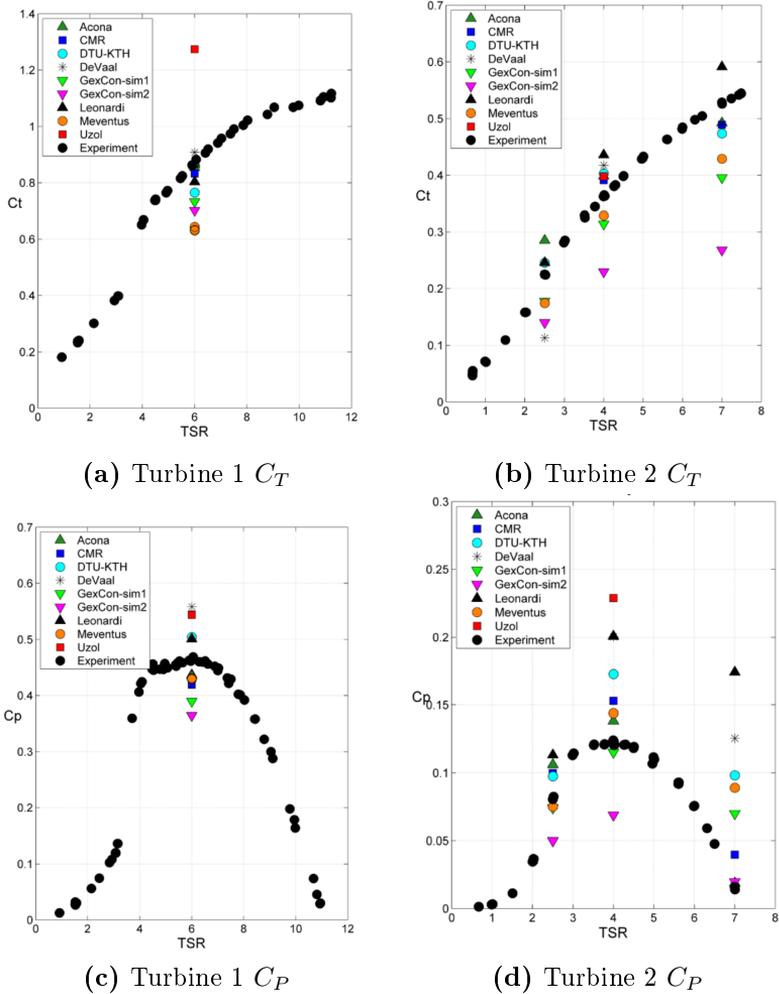


Figure 58: Experimental and simulated values of C_T and C_P for various TSR from "Blind test 2" [23]. The black circles indicate the experimental values. The results from this thesis can be found in table 11.

Figure 58 shows the thrust and power coefficients from the "Blind test 2" ex-

periment. The black dots indicate the experimental values, while the other graphs represent the simulations. Almost all the candidates underestimated the thrust for the upstream turbine, which is the only output largely overestimated in the simulation in this study. The thrust for the downstream turbine and power for both turbines was both under- and overestimated.

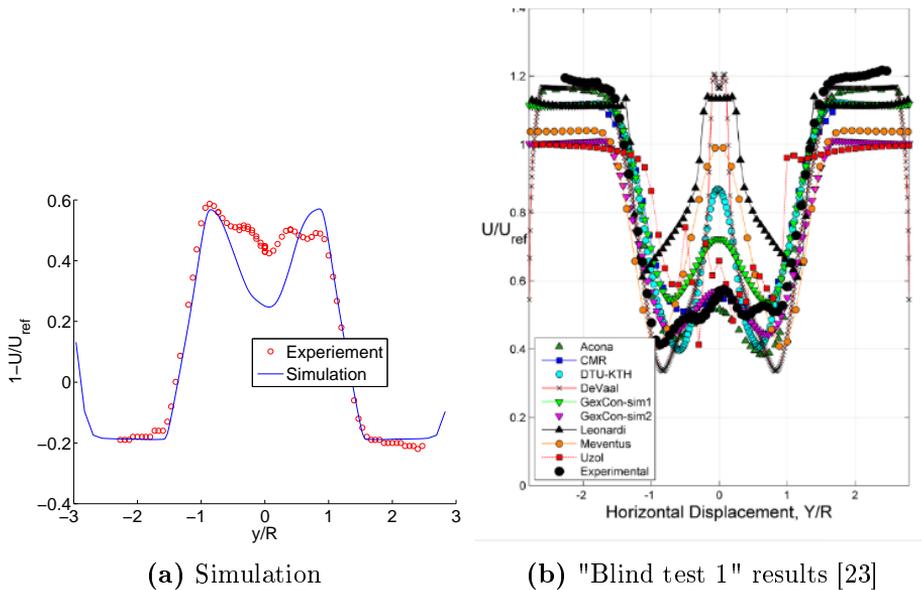


Figure 59: The velocity deficit one diameter behind the rotor, along the y -axis. The hub and tower are included in the simulation.

Figure 59a shows the velocity deficit one diameter behind the second turbine. One of the velocity deficit peaks is simulated perfectly compared to the experiment, while the other one is overestimated. The experiment has more asymmetry than the simulation, and also more velocity deficit behind the center of the rotor. The width of the wake is simulated very well.

Figure 59b shows that many of the "Blind test 2" candidates overestimated the velocity in the middle of the rotor. Compared to the other simulations

of the "Blind test 2", the velocity wake one diameter behind the second turbine is modelled well in this study.

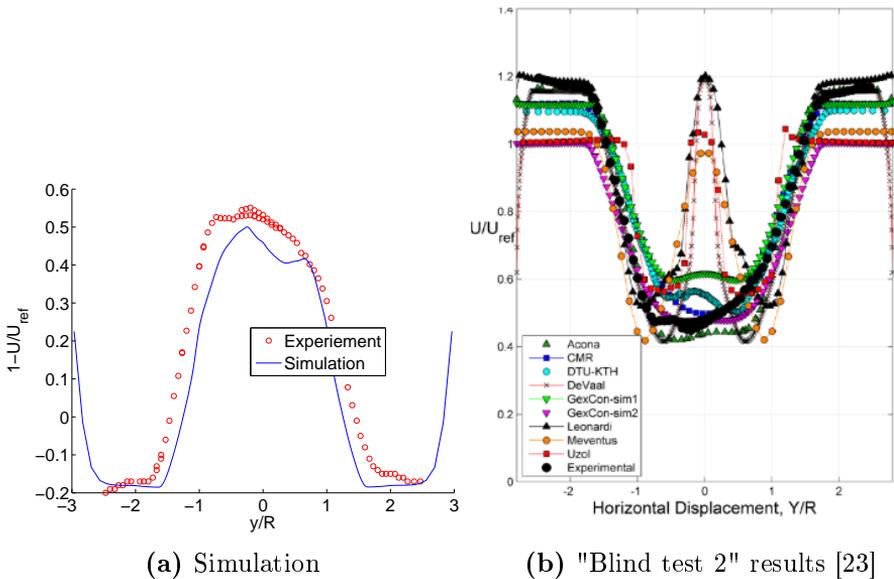


Figure 60: The velocity deficit two and a half diameters behind the rotor, along the y -axis. The hub and tower are included in the simulation.

Figure 60a shows the velocity profile two and a half diameters behind the second rotor. The simulated velocity deficit is slightly lower than the measurement, and the wake has an asymmetry that corresponds well with the measured values. The width of the wake is simulated too small compared to the experiment.

Compared to the "Blind test 2" candidates in figure 60b, the velocity is modelled better than many. It is easy to see from figure 60b which simulations included the hub, and which did not. The hub has a great influence on how well the velocity wake is simulated.

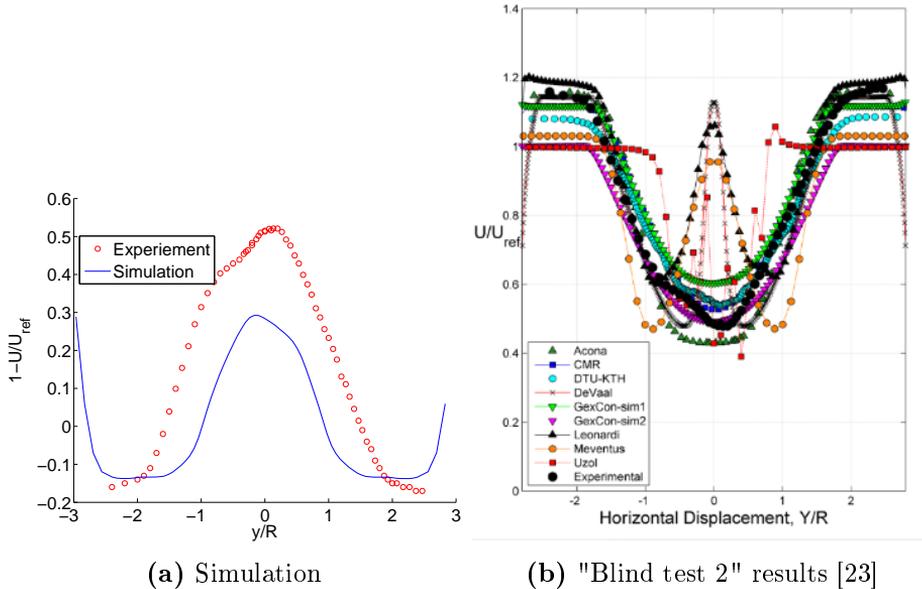


Figure 61: The velocity deficit four diameters behind the rotor, along the y -axis. The hub and tower are included in the simulation.

Figure 61a shows the velocity profile four diameters behind the second rotor. The velocity deficit is underestimated, and the wake width is too small, but the profile of the velocity wake corresponds well with the measurements. At four diameters behind the second rotor, the grid is no longer refined. This could affect the result as the wake is not resolved well enough, and the velocity is dissipated faster for the simulations than for the measurement.

The simulations in this study does not perform better than the "Blind test 2" candidates, and the reason for this is probably that the grid is not well enough refined four diameters behind the turbine.

The experimental values for the turbulent wake is given by u' , i.e. the fluctuating turbulent velocity in the x -direction. In section 2.2.2 the relationship between the turbulent kinetic energy and the fluctuating velocities

was given as

$$k = \frac{1}{2} (u'^2 + v'^2 + w'^2).$$

If an isotropic turbulence is assumed with $u' = v' = w'$, the turbulent kinetic energy can be calculated as $k = \frac{3}{2}u'^2$.

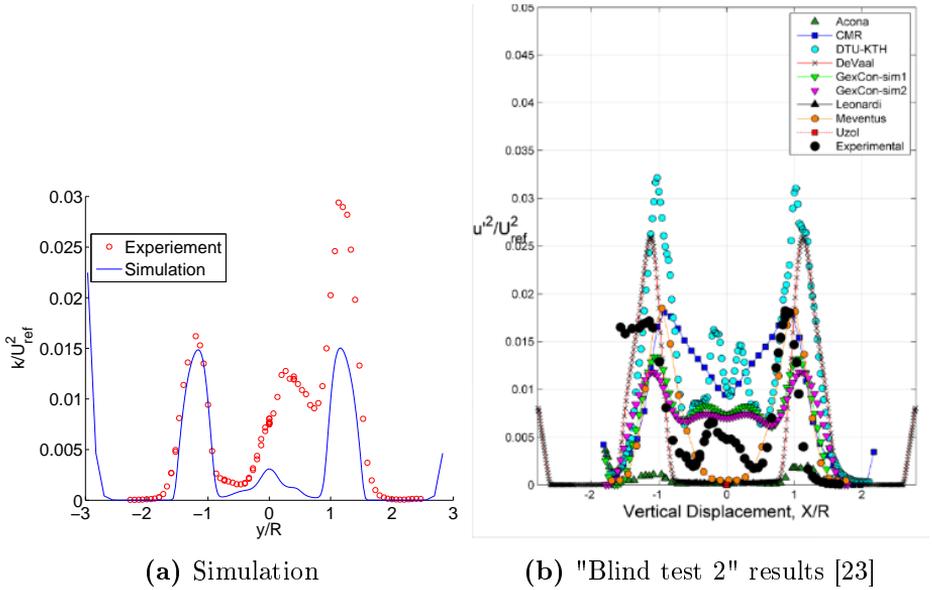


Figure 62: The turbulent kinetic energy one diameter behind the second rotor, along the y -axis. Notice that figure a shows the horizontal k (for the measurement $u' = 2/3k$ has been used), while figure b shows the vertical u' (along the z -axis for this thesis). The hub and tower are included in the simulation.

Figure 62a shows the measured and simulated turbulent kinetic energy one diameter behind the second turbine. The left half of the turbulent wake corresponds very well with the measurements, as was also seen for the velocity wake. The asymmetry is underestimated and the turbulence behind the other tip is simulated as half of the measurement.

Figure 62b shows the turbulent velocity fluctuation in the x -direction along the z -axis. A significant spread in the results exist, but the simulations are better for the "Blind test 2" than for the "Blind test 1", and a logarithmic scale is not needed to show the results. Since the results shown in figure 62b are along the vertical axis, and not horizontal as for the simulations in this study, the results are not directly comparable, but are added to give an indication of how well the "Blind test 2" candidates performed.

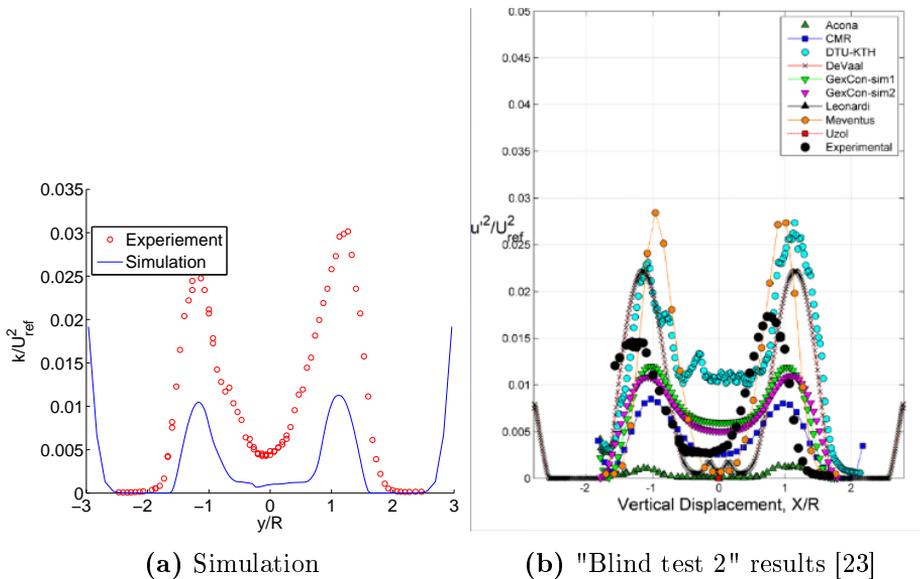


Figure 63: The turbulent kinetic energy two and a half diameters behind the second rotor, along the y -axis. Notice that figure a shows the horizontal k (for the measurement $u' = 2/3k$ has been used), while figure b shows the vertical u' (along the z -axis for this thesis). The hub and tower are included in the simulation.

Figure 63a shows the turbulent kinetic energy two and a half diameters behind the second rotor. The simulated turbulent wake has dissipated more than the measured wake. The measured wake has become less asymmetric,

and the kinetic energy behind the lowest tip in the last figure has risen, gaining energy from other parts of the wake.

Figure 63b shows the "Blind test 2" candidates' simulations in the vertical direction. Some of the candidates overestimate the turbulence, while others underestimate it like the simulations in this study did.

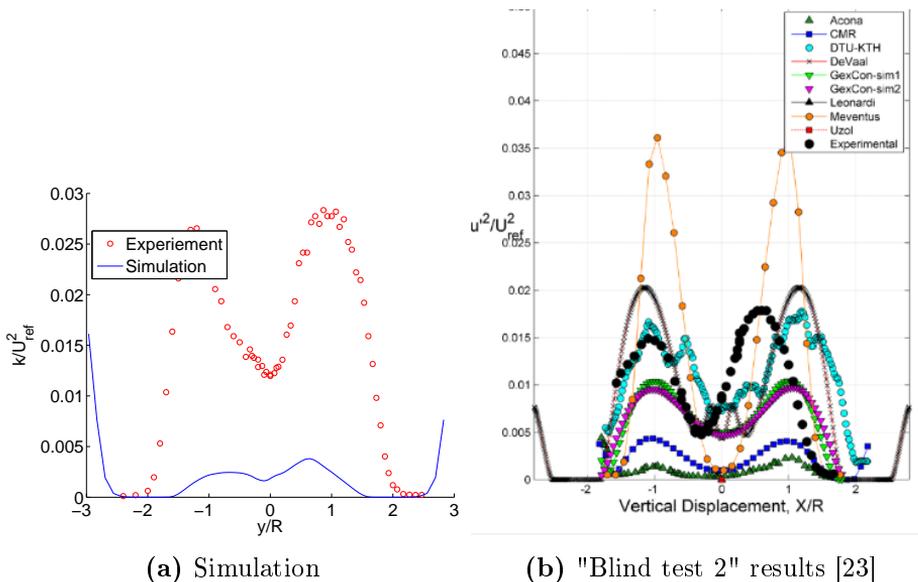


Figure 64: The turbulent kinetic energy four diameters behind the rotor, along the y -axis. Notice that figure a shows the horizontal k (for the measurement $u' = 2/3k$ has been used), while figure b shows the vertical u' (along the z -axis for this thesis). The hub and tower are included in the simulation.

Figure 64a shows the turbulent kinetic wake four diameters behind the second rotor. The simulated wake has significantly dissipated, while the measured wake has barely lost any energy. One reason for this big difference may be that the simulated wake is outside of the refinement region, and the grid is not fine enough to resolve the details in the wake. This leads

to a quick dissipation of the wake.

Many of the "Blind test 2" candidates represent the wake well at this distance from the turbine, but some still overestimate while others largely underestimate as was done in this thesis.

The conclusions from the comparison with the "Blind test 2" was that the thrust and power is very well simulated, except for the thrust in the upstream turbine.

The velocity wake shapes one and two and a half diameters behind the downstream rotor is very well modelled. The turbulent kinetic energy one diameter behind the second turbine is well modelled, while two and a half diameters behind the second turbine is slightly underestimated.

At four turbine diameters behind the second turbine both velocity and turbulent kinetic energy are modelled too low. This is probably because the wake is outside of the refined region. This shows how a well refined grid in the wake region affects the results.

The simulations in this study, and in the other "Blind test" candidates, simulate the turbulent kinetic energy wake shapes better for "Blind test 2" than for "Blind test 1". This is probably because the flow is more turbulent, which is easier to simulate.

The results from the "Blind test 2" simulations are good. This shows that the developed guidelines in the parameter study works well for this set-up also, and not just the "Blind test 1" set-up that they were developed for.

6 FURTHER WORK

The actuator line model needs additional testing and improvements before it can be suitable for an industrial purpose. The simulations in this thesis did not lead to a grid independent solution, which is a problem for a numerical model. Additional examinations need to be performed to find out why grid independency could not be achieved. Some possible explanations were mentioned in this thesis, providing someone with a good start to do further testing.

The calculations in this thesis were only performed at the design conditions of the wind turbine. Simulations need to be run to see if the relations developed in this thesis hold for conditions off the design conditions.

Since this was a model turbine in a wind tunnel, it might be subject to small scale effects. Simulations need to be performed for a full scale wind turbine and compared to measurements to see if the developed guidelines for the use of the actuator line can be extended to a full scale turbine.

All the simulations in this study used RANS equations and the $k - \epsilon$ model. The developed guidelines should be tested using RANS equations combined with other turbulence models, and by using LES.

Comparing the simulated turbulent kinetic energy wakes to the measurements in this thesis, it was found that the turbulence is underestimated, and the turbulent kinetic energy at the tip of the blades was particularly hard to simulate. It would be interesting to know if this is caused by the $k - \epsilon$ turbulence model, the actuator line model, small scale effects in the model turbine or the low turbulence level.

For the "Blind test 1" simulations the grid was only refined two diameters behind the turbine. The fine grid area should be extended past five diameters behind the turbine to see if this improves the wake shapes. This should also be done for the "Blind test 2" simulation, where the wake four diameters behind the downstream turbine is outside of the refined area.

In the "Blind test 2" set-up, only two different grid resolutions were simulated. More grid resolutions should be simulated, seeing if the chosen grid

resolution produce the lowest thrust and power output.

The "Blind test 1" simulation estimated the thrust very well, but the power was overestimated. The "Blind test 2" simulation estimated the thrust and power for the downstream turbine well, and the power for the upstream turbine, but the thrust for the upstream turbine was overestimated. This should be investigated further.

Most of the simulations in this thesis are performed with 80 processors. A simulation has also been performed with 16 processors, and this produced a negligible difference from the results with 80 processors. Additional testing should be done to identify if this parallelization affects the results, e.g. by running a simulation on only one processor.

A refinement procedure has been used in this thesis. There was a difference in the results between the refined and non-refined grids. An analysis performed to investigate this difference showed that it may come from an error in the code leading to an accumulation of thrust and power over time. The results from this analysis might also explain the lack of a grid converged solution, and should be investigate further.

The implementation of the actuator line code in this thesis was shown to create several spikes of higher and lower thrust and power over time. These spikes are probably due to a numerical error, and should not be present. Appendix B gives a possible explanation for the spikes, showing that they are created at certain turbine angles. One way to get rid of these spikes may be to create a grid that rotates with the rotor, while the grid around the rotor is kept stationary.

For one of the simulations in this study, large oscillations occurred and the average thrust and power results were much lower than expected. This is likely due to a numerical error that should be resolved.

7 CONCLUSION

The actuator line model requires many input parameters. Some of these are dependent on the turbine geometry, while some have to be determined in other ways. A parameter study was performed for the four main adjustable parameters, the number of blade elements, the time step, the Gaussian width parameter ϵ and the grid. In previous publications on the actuator line model, ϵ has mostly been connected to the grid spacing. This thesis argues that ϵ should be chosen from the chord length, in order to use the distributed forces to model the blade geometry correctly.

The following relations between the three first variables produced good results for the right grid resolution:

- $\Delta r \leq \Delta y$
- $\Delta t < \frac{\Delta y}{\text{tip speed}}$
- $\epsilon = \frac{c}{4.3}$

This thesis showed that the solution is highly dependent on the grid resolution, and especially the grid in the streamwise direction. No grid independent solution was found, but a method was developed to achieve results that well represents the measurements. The following procedure was created to determine the grid:

1. Perform several simulations with different grid resolutions keeping $\Delta x = \Delta y = \Delta z$.
2. Check for grids producing abnormal results and/or oscillations.
3. Choose one or more grids and perform simulations changing only Δx .
4. Choose the result producing the lowest thrust and power.

The actuator line model was compared to measurements from the "Blind test 1" and "Blind test 2" experiments. For the "Blind test 1" simulations, there was a good correlation between the measured and simulated thrust, but the power was overestimated. The velocity wakes corresponded well to the experiments, especially given that this is a simplified model and

7 CONCLUSION

not a fully resolved turbine calculation. The turbulent kinetic energy was mostly underestimated, but the trend of the simulated profile was close to the measured profile. A hub and tower was implemented using actuator line elements, and this improved the kinetic turbulent results greatly, showing their importance in simulating the turbine wake.

The "Blind test 2" simulations estimated the power very well for both turbines, and the thrust for the second turbine, but the thrust for the upstream turbine was overestimated. The velocity wakes corresponded well with the measurement, but the wake four diameters behind the rotor had dissipation too much according to the simulation, yielding a too large the velocity. This may be because the grid was not refined down to four diameters behind the second turbine.

REFERENCES

- [1] Owe Axelsson. *Iterative Solution Methods*. Cambridge University Press, March 1996.
- [2] Matt Churchfield and Sang Lee. NWTC design codes (SOWFA).
- [3] Matthew J. Churchfield, Sang Lee, Patrick J. Moriarty, Luis A. Martinez, Leonardi Stefano, Ganesh Vijayakumar, and James G. Brasseur. A large-eddy simulation of wind-plant aerodynamics. Conference paper (National Renewable Energy Laboratory (U.S.)) ; Nashville, Tennessee, January 2012. National Renewable Energy Laboratory, Office of Energy Efficiency and Renewable Energy,.
- [4] S. Ivanell, J. N. Sørensen, R. Mikkelsen, and D. Henningson. Numerical analysis of the tip and root vortex position in the wake of a wind turbine. *Journal of Physics: Conference Series*, 75(1):012035, July 2007.
- [5] Stefan S. A. Ivanell. *Numerical Computations of Wind Turbine Wakes*. PhD thesis, Royal Institute of Technology, 2005.
- [6] Jens Nørkær Sørensen and Wen Zhong Shen. Numerical modeling of wind turbine wakes. *Journal of Fluids Engineering*, 124(2):393–399, May 2002.
- [7] Siri Kalvig and Eirik Manger. Comparing "blind test" CFD calculations of different wind turbine modelling approaches. In *EAWWE 4th scientific conference "The Science of Making Torque from Wind"*, in review. Oldenburg, Germany, 2012.
- [8] Per-Åge Krogstad and Pål Egil Eriksen. "Blind test" calculations of the performance and wake development for a model wind turbine. *Renewable Energy*, 50:325–333, February 2013.
- [9] Per-Åge Krogstad, Pål Egil Eriksen, and J. Melheim. Blind test workshop; calculations for a model wind turbine. *Dept. Energy and Process Eng., NTNU, Trondheim, Norway*, March 2011.

REFERENCES

- [10] James F. Manwell, Jon G. McGowan, and Anthony L. Rogers. *Wind Energy Explained: Theory, Design and Application*. Wiley, 2 edition, February 2010.
- [11] Luis A. Martinez, Stefano Leonardi, Matthew J. Churchfield, and Patrick J. Moriarty. A comparison of actuator disk and actuator line wind turbine models and best practices for their use. *50th AIAA Aerospace Science Meeting*, January 2012.
- [12] Robert Mikkelsen. *Actuator Disc Methods Applied to Wind Turbines*. PhD thesis, Technical University of Denmark, June 2003.
- [13] Robert Mikkelsen, Jens N. Sørensen, and Niels Troldborg. Prescribed wind shear modelling combined with the actuator line technique. Milan, 2007.
- [14] Robert Mikkelsen, Jens N. Sørensen, Stig Øye, and Niels Troldborg. Analysis of power enhancement for a row of wind turbines using the actuator line technique. *Journal of Physics: Conference Series*, 75(1):012044, July 2007.
- [15] Offshore Moorings. Offshore moorings - floating windturbines. http://www.offshoremoorings.org/moorings/2008/group%20f/website/turbine_loading_stability.html, May 2013.
- [16] OpenFOAM Foundation. OpenFOAM user guide. <http://www.openfoam.org/docs/user/>, May 2013.
- [17] Fabio Pierella, Pål Egil Eriksen, Lars Sætran, and Per-Åge Krogstad. Invitation to the 2012 "Blind test 2" workshop calculations for two wind turbines in line. *Dept. Energy and Process Eng., NTNU, Trondheim, Norway*, March 2012.
- [18] Hermann Schlichting. *Boundary-layer theory*. McGraw-Hill, New York, 1979.

-
- [19] Sven Schmitz, Pankaj Jha, Joshua Dowler, Frank Kody, Dwight Brillembourg, and David Reich. Accuracy of current actuator-line modeling methods in predicting wind turbine blade loads. Penn State, January 2013.
- [20] Michael Schäfer. *Computational Engineering - Introduction to Numerical Methods*. Springer, 2006 edition, April 2006.
- [21] Wen Zhong Shen, Jens Nørkær Sørensen, Jianhui Zhang, Wen Zhong Shen, Jens Nørkær Sørensen, and Jianhui Zhang. Actuator surface model for wind turbine flow computations. *Proceedings of European Wind Energy Conference 2007*, 2007.
- [22] Michael Shives and Curran Crawford. Mesh and load distribution requirements for actuator line CFD simulations. *Wind Energy*, 2012.
- [23] Lars Sætran. Wind and wake modelling - blind test 2, DeepWind 2013 - 10th deep sea offshore wind R&D seminar. http://www.sintef.no/project/DeepWind%202013/Deepwind%20presentations%202013/F/S%C3%A6tran,L._NTNU_new.pdf, January 2013.
- [24] The Encyclopadie of Alternative Energy and Sustainable Living. wind turbine hub height. http://www.daviddarling.info/encyclopedia/H/AE_hub_height.html, 2013.
- [25] Niels Troldborg. *Actuator Line Modeling of Wind Turbine Wakes*. PhD thesis, Technical University of Denmark, November 2008.
- [26] Niels Troldborg, Jens N. Sørensen, and Robert Mikkelsen. Actuator line simulation of wake of wind turbine operating in turbulent inflow. *Journal of Physics: Conference Series*, 75(1):012063, July 2007.
- [27] H. Versteeg and W. Malalasekera. *An Introduction to Computational Fluid Dynamics: The Finite Volume Method*. Prentice Hall, 2 edition, February 2007.

REFERENCES

- [28] Frank White. *Viscous Fluid Flow*. McGraw-Hill Science/Engineering/Math, New York, 3 edition, January 2005.
- [29] Frank White. *Fluid Mechanics*. McGraw-Hill Science/Engineering/Math, New York, 6 edition, October 2008.

A A CLOSER LOOK AT THE GRID REFINEMENT

In sections 5.2.2 and 5.2.3 of the parameter study, simulations were performed both with and without grid refinement around the rotor. In section 5.2.2, ϵ was kept at $\epsilon = 2.5\Delta x$ and changed with the grid, while in section 5.2.3 ϵ was kept constant and connected to the blade chord length. This section will look into the differences in results between the refined and non-refined grid simulations.

For the simulations performed with $\epsilon = 2.5\Delta x$, a difference appeared in the thrust and power results between the cases run with and without grid refinement. These differences were almost 5% for thrust and 10% for power for the finest grid resolution ($\Delta x = 0.012m$ around the rotor). For the simulations run with $\epsilon = 0.0084m$, the difference between the refined and non-refined cases were negligible with a difference in thrust and power of 0.1% for the finest grid resolution ($\Delta x = 0.012m$, $\epsilon = 0.7\Delta x$).

The investigation of the refinement procedure will start by discussing the results from simulations ϵ is connected to the grid at $2.5\Delta x$, followed by a similar discussion for the results where ϵ is kept at $0.0084m$. One theory is that the development of the variables over time has an influence on the result, and a special focus will given to the thrust over time development. Thrust and power have similar trends over time, and therefore only the thrust trend will not be presented.

A.1 THRUST OVER TIME FOR $\epsilon = 2.5\Delta x$

INCLUDING GRID REFINEMENT AROUND THE ROTOR

ϵ is for this kept with a constant relationship to the grid resolution, i.e. $2.5\Delta x$, and the grid is refined three time around the rotor.. The specifications for these simulations can be found in section 5.2.2, including time step, blade elements, ϵ and grid resolution.

Figure 65 shows the variance of the thrust over time for different grid resolutions in some of the cases with three grid refinements. As the figure shows, for coarse grids the function declines and stabilizes around $t = 0.5s$.

A A CLOSER LOOK AT THE GRID REFINEMENT

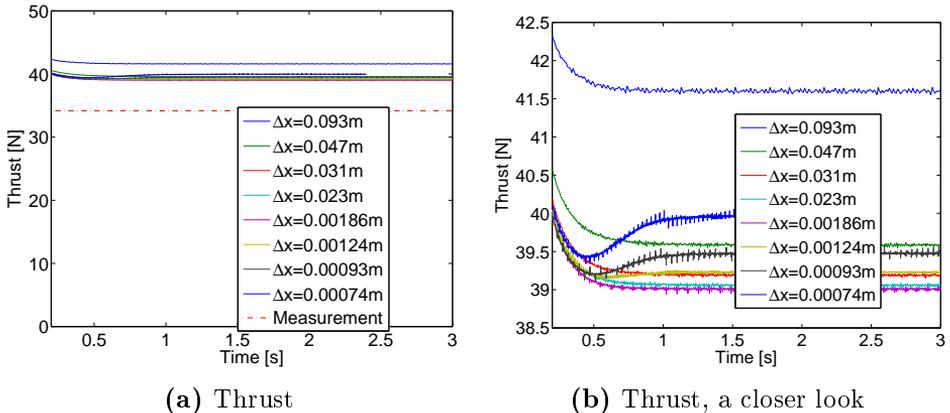


Figure 65: Thrust over time with grid refinement for $\epsilon = 2.5\Delta x$.

For grid resolutions finer than $\Delta x = 0.00186m$, the thrust curve increases after $t = 0.5s$, before it stabilizes around $t = 1s$.

$\Delta x = 0.00186m$ produces the lowest average thrust, as can also be seen in figure 32a (average thrust for different grid resolutions). The grid resolution $\Delta x = 0.00186m$ corresponds with the finest grid resolution not producing an increase in thrust after $t = 0.5s$ in figure 65.

To summarize; no change in thrust over time after $t = 0.5s$ leads to a declining average thrust for finer grids, while an inclination in thrust after $t = 0.5s$ leads to a rise in average thrust for finer grids.

WITHOUT GRID REFINEMENT AROUND THE ROTOR

The analysis performed above will now be done for the non-refined simulations where ϵ is kept at $2.5\Delta x$. Details for the simulations discussed below can be found in section 5.2.2. The only thing separating these simulations with the results from above is the far field grid. The resulting thrust and power should only be a function of the cell density around the rotor, and the refined and non-refined grid should yield the same results.

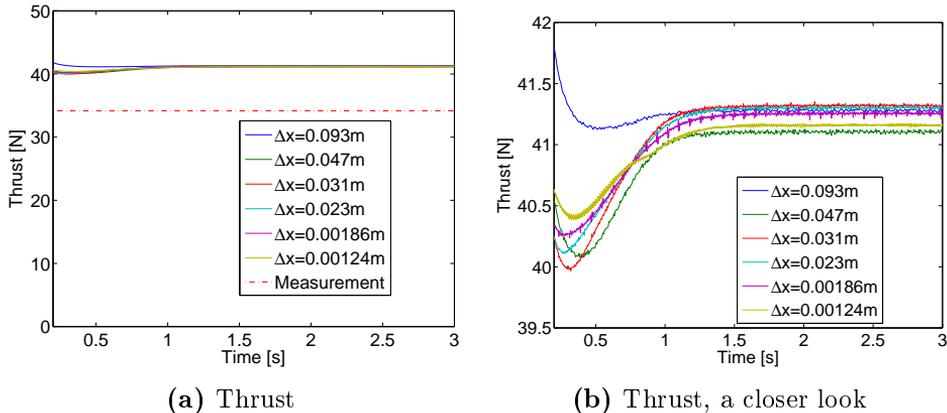


Figure 66: Thrust over time without grid refinement for $\epsilon = 2.5\Delta x$.

Figure 66 shows thrust over time for some of the non-refined cases. The graphs in figure 66 all have an inclination in thrust after $t = 0.5s$, and the solutions stabilize around $t = 1s$.

For this case, the average thrust (see figure 33a) declines slightly and then converges for finer grids at a value over 17 % higher than the measured thrust.

Compared to the refined cases at the same grid resolutions as here, the thrust declines. The cases without grid refinement produce higher thrust values than the cases with grid refinement, although the cell density is the same around the rotor.

To summarize; all of the graphs in figure 66 increases after $t = 0.5s$. The average thrust in the non-refined cases decreases only slightly before it converges. For the refined cases, the average thrust decreases for increasing grid resolutions down to $\Delta x = 0.00186m$.

With an inlet velocity of $U_{in} = 10m/s$, and an $11m$ long wind tunnel, the flow uses $\sim 1s$ from inlet to outlet of the tunnel. It is therefore not unlikely that some of the simulations use $1s$ to reach steady-state, depending on e.g.

the grid resolution. But should it also depend on ϵ ?

A.2 THRUST OVER TIME FOR $\epsilon = 0.0084m$

INCLUDING GRID REFINEMENT AROUND THE ROTOR

In section 5.2.3 ϵ is kept constant at $0.0084m$. The details for these simulations, including time step, blade elements and grid resolutions can be found in section 5.2.3.

There is very little difference in the resulting thrust and power between the refined and non-refined cases, and for the finest grid resolution the difference is only 0.1% for both thrust and power.

Figure 67 shows the thrust over time for the refined cases with $\Delta x < 0.01m$, while figure 68 shows the results for $\Delta x > 0.01m$.

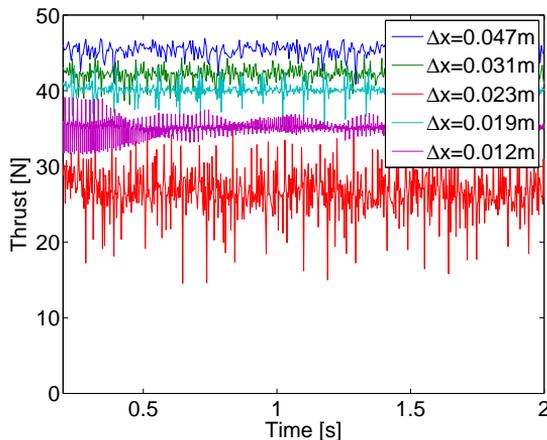


Figure 67: Thrust over time for varying grids with grid refinement around the rotor, with $\epsilon = 0.0084m$ and $\Delta x > 0.01m$.

The average thrust declines down to $\Delta x = 0.012m$. One point deviates from this trend, and will be discussed in the next section. For grids finer than $\Delta x = 0.012m$, the refined thrust and power graphs rise with decreasing grid resolution.

Since $\epsilon/\Delta x$ is small, only up to 0.7 for the finest grid in figure 67, a lot of

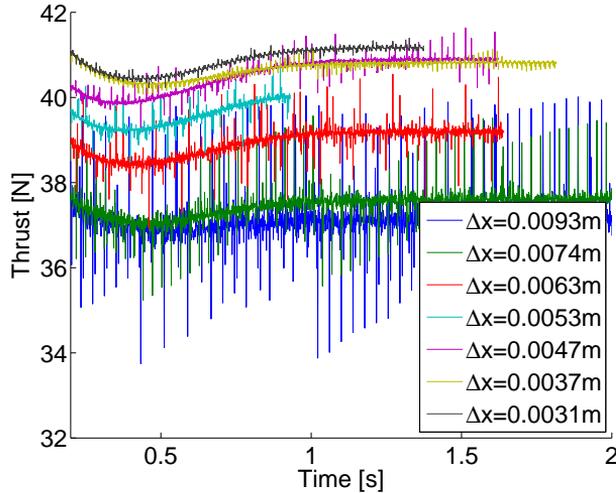


Figure 68: Thrust over time for varying grids with grid refinement around the rotor, with $\epsilon = 0.0084m$ and $\Delta x < 0.01m$. The two finest grid resolutions have an additional grid refinement.

noise is present in the plots (as discussed in appendix B). Figure 68 shows clearly that the noise decreases with increasing $\epsilon/\Delta x$.

The graph for $\Delta x = 0.023m$ varies a lot more than the others, and a closer look at this grid resolution will be given in appendix C and disregarded in the following analysis.

In figure 67 all of the graphs are stable after $t = 0.5s$, and the thrust decreases with increasing grid refinement. In figure 68 all of the graphs increase after $t = 0.5s$ and stabilizes after $t = 1s$. The average thrust decreases down to $\Delta x = 0.012m$ (the finest grid in figure 67), and increases for finer grids.

The two finest grids ($\Delta x = 0.0037m$ and $\Delta x = 0.0031m$) have been simulated with four grid refinements around the rotor. The extra grid refinement produce less variation in thrust after $t = 0.5s$, and also a smaller average thrust. This is equivalent with the results at the start of this section, where

A A CLOSER LOOK AT THE GRID REFINEMENT

the cases with three refinements had less variation of thrust over time than the cases with no refinement.

To summarize; as also found above, when the thrust over time is stable after $t = 0.5s$, the average thrust declines for decreasing grid resolution, while when thrust over time stabilizes after $t = 1s$, the average thrust increases for finer grids.

WITHOUT GRID REFINEMENT AROUND THE ROTOR

Simulations without grid refinement have been performed for grid resolutions down to $\Delta x = 0.012m$. ϵ is kept at $0.0084m$, and an overview of the other parameters can be found in section 5.2.3.

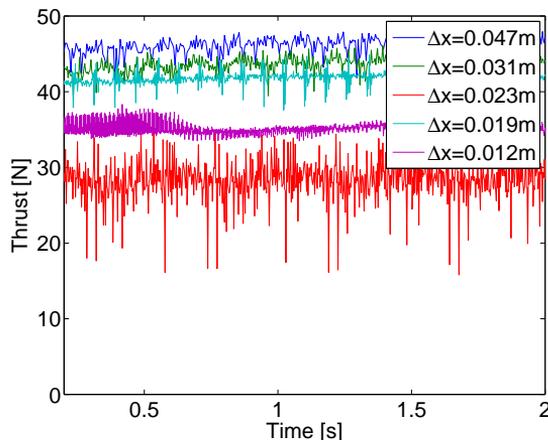


Figure 69: Thrust over time for varying grids without grid refinement around the rotor, with $\epsilon = 0.0084m$.

Figure 69 shows the simulations for $\epsilon = 0.0084m$ without grid refinement. A discrepancy can again be seen for $\Delta x = 0.023m$, and this grid resolution will be disregarded here and investigated in appendix A.

The graphs for the non-refined cases all stabilize after $t = 0.5s$. In addition, the average thrust values are very close to the values for the refined cases,

even for the odd point at $\Delta x = 0.023m$.

To summarize; the difference in thrust and power for the refined and non-refined cases is small. For the grid resolutions simulated both with and without grid refinement there is no increase in thrust over time after $t = 0.5s$.

One conclusion that can be drawn from this analysis is that a rise in thrust after $t = 0.5s$ might be the cause of the differences in the resulting thrust between the refined and non-refined cases where $\epsilon = 2.5\Delta x$. In addition, this analysis shows that the converged thrust result for the non-refined cases is not necessarily a correct simulation. This is because the non-refined cases where $\epsilon = 0.0084m$ does not converge, and the resulting thrust and power are larger than the measured values.

Another interesting conclusion from the refined cases is that for a graph that stabilizes after $t = 0.5s$ the thrust declines for decreasing grid resolutions, while the opposite is true for graphs that increase after $t = 0.5s$. The same was seen in the non-refined cases where $\epsilon = 0.0084m$. For $\epsilon = 2.5\Delta x$ the thrust increased after $t = 0.5s$, but the average thrust did not increase. This might be due to effects in the far field grid. This is a possible explanation of why a grid converged solution was not achieved in the parameter study.

Possible explanations for this rise in thrust after $t = 0.5s$:

- A natural time period for the solution to stabilize
- A bug in the numerical solver
- A bug in the actuator line implementation
- Thrust being accumulated over time
- Dampening factors in the code not functioning properly

B NOISE IN PLOTS

The analysis in this section is based on the reference case with a cubical grid;

Δt [s]	Cells	ϵ [m]	b	Δx [m]	Δy [m]	Δz [m]
0.0001	4 mill	0.0122	100	0.0094	0.0094	0.0094

Figure 29 shows thrust plotted over time for different ϵ , and it has been replotted here in figure 70. There are many peaks in the thrust curve, and the following will be an investigation of why they appear and what variables they are dependent on.

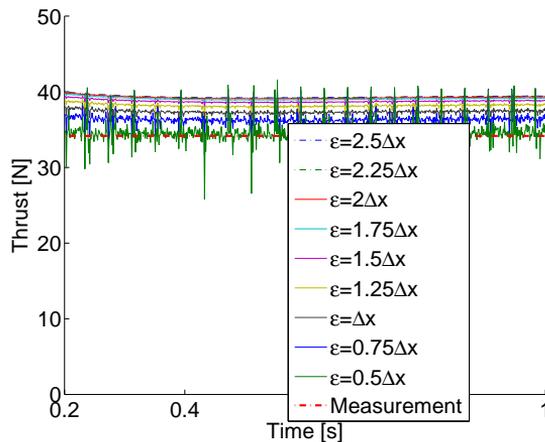


Figure 70: Thrust vs. time for various ϵ and a fixed $\Delta x = 0.0094m$ around the rotor.

From figure 70 it can be seen that the size of the peaks grow with declining values of ϵ .

A representation of this is given in figure 71 where the size of the peaks is compared to the total value of the thrust. For the smallest value of ϵ in this

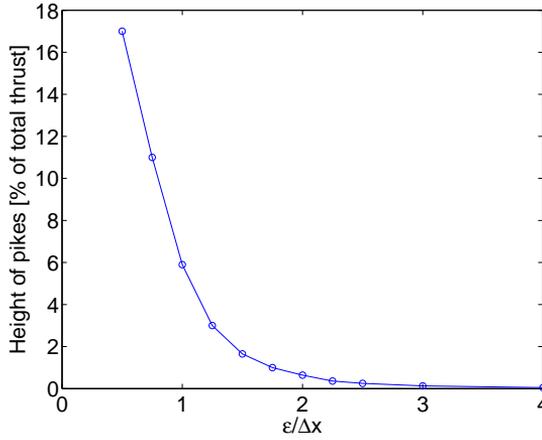


Figure 71: Height of peaks in thrust over time.

example, the peaks are 17% of the total thrust value, while for larger ϵ the height of the peaks become negligible. Figure 71 was created from simulations run with a grid spacing of $\Delta x = 0.0094m$, for other grid resolutions the values on the y -axis might change.

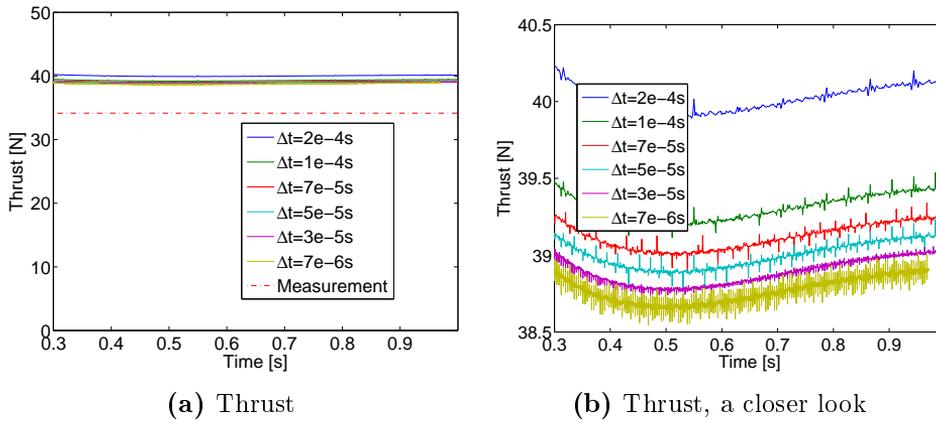


Figure 72: Thrust over time for different time steps.

B NOISE IN PLOTS

Figure 72 shows the thrust plotted over time for different values of the time step. It is clear the the frequency of the peaks increase with decreasing time step. A further investigation brings forth that the frequency is oposite linearly dependent on the time step, i.e. when the time step is halved, the frequency is doubled.

Figure 73 and 74 shows the pressure distribution across the rotor plane for two different time steps for the case where

Δt [s]	Cells	ϵ [m]	b	Δx [m]	Δy [m]	Δz [m]
0.0002	0.5 mill	0.047	50	0.019	0.019	0.019

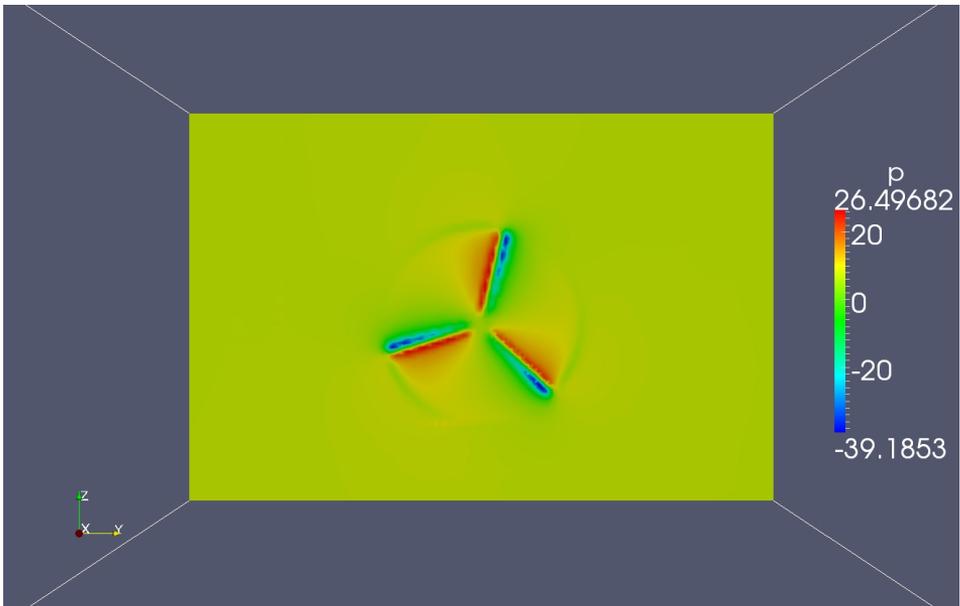


Figure 73: Pressure field in the rotor plane for time step 1.

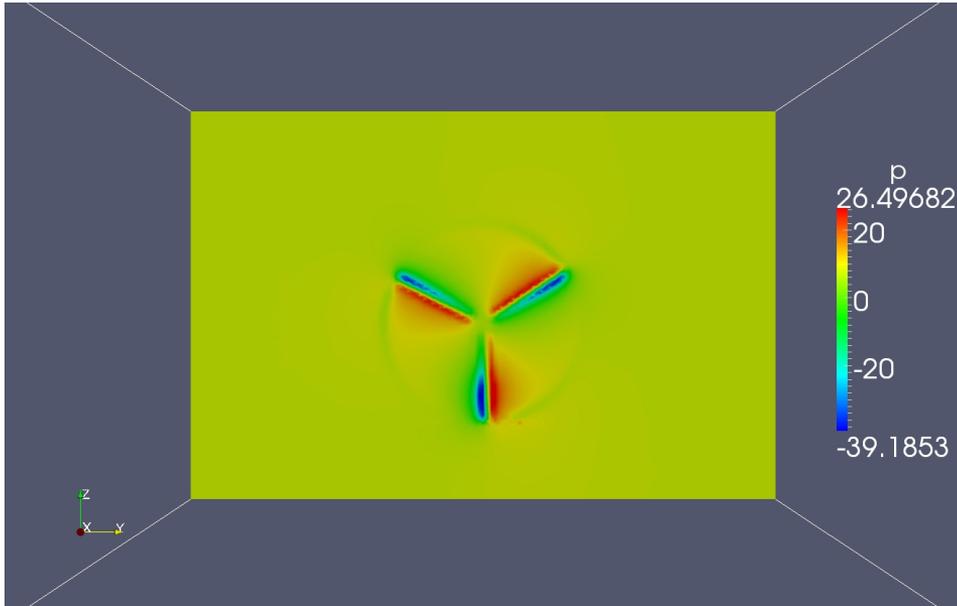


Figure 74: Pressure field in the rotor plane for time step 2.

In figure 73 it can be seen that the variation in pressure distribution across the blades is equal for all blade. In figure 74 one of the blades has a larger pressure variation across the blade than the other two blades. This phenomenon occurs each time one of the three blades has an angle of 0, 90, 180 or 270 degrees, i.e. when one of the blade is situated on either the y - or z -axis.

In addition, the effect of a too small ϵ for the grid resolution, where the blades forces are not distributed smoothly can be seen in figure 73. A few dots of larger or smaller pressure appear along the blade. The effect in 74 is even more distinguished for smaller $\epsilon/\Delta x$.

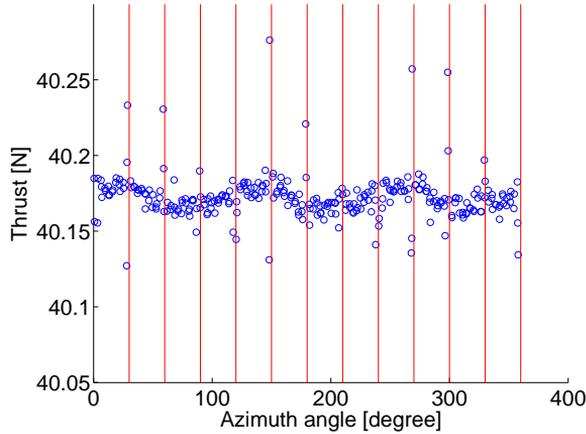


Figure 75: Thrust over azimuth angle for $\Delta t=2e4s$.

One blade passes the y - or z -axis every 30 degree rotation of the rotor. Figures 75 and 76 show thrust plotted over the azimuth angle (the angle of the rotor) for two different time steps for the same amount of time. Figure 76 comes from the reference case presented at the start of this section, while figure 75 is the same case, but with a time step of $\Delta t = 2 \cdot 10^4 s$. Lines for the angles 30, 60, 90, 120, etc. degrees have been drawn in the figure. From both figures it is clear that the peaks (the minimum and maximum values of the dots) occur at the red lines.

In figure 76 there are more peaks than figure 75. This is because for a smaller time step, more sample points are extracted from the model through one rotation, hence more often will a thrust value be extracted when one turbine blade is at the y - or z -axis.

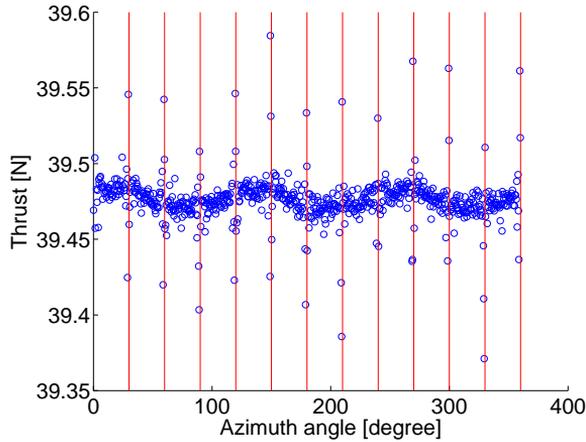


Figure 76: Thrust over azimuth angle for $\Delta t=1e4s$.

The conclusion is that the noise occurs due to discrepancies happening at certain turbine angles.

A possible explanation for this is that the blade replaces a lot of its grid cells as it passes through the top, bottom and side points, i.e. 0, 90, 180 and 270 degrees. Lets say that the blade is passing through the top point, i.e. 0 degrees. At one time step it is before 0 degrees, and at the next time step it has passed 0 degrees. If there is a grid cell face going vertically through the turbine, the blade will have all of its blade element points in one row of grid cell at one time step, and at the next time step the blade element points will have been moved to a new set of grid cells.

Figure 77 illustrates this phenomenon, where the blade elements (green dots) cross the 0 degree line of cell faces.

This problem may be fixed by making modifications in the solver or by changing the grid in the rotor plane e.g. to a polar coordinate system or have a grid that rotates with the turbine.

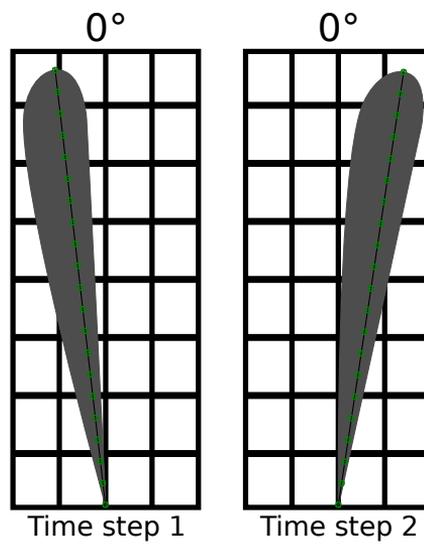


Figure 77: A rotor blade before it passes 0 degrees, and after. The green lines indicate the blade elements points across the blade.

C THRUST AND POWER GRAPHS FOR $\epsilon = 0.0084m$ AND $\Delta x = 0.023m$

The analysis in this section is based on the following case;

Δt [s]	Cells	ϵ [m]	\mathbf{b}	Δx [m]	Δy [m]	Δz [m]
0.00025	0.3 mill	0.0084	40	0.023	0.023	0.023

In section 5.2.3, ϵ was kept constant, while simulations for different grid resolutions were performed. When keeping ϵ constant at $0.0084m$, an inconsistent result appeared for $\Delta x = 0.023m$ (the case shown in the table above). Both thrust and power had a significant drop, and the oscillations across time were large.

To find the reason for this discrepancy, simulations with a variation in the time step, the number of blade elements, ϵ and grid were made. The time step and number of blade elements proved to have very little influence on the results. The dip is therefore dependent on the grid and on ϵ .

Several attempts will be made to find out why this dip appears. First, the thrust variation across time will be investigated. Thereafter, simulations for larger and smaller grid resolutions will be made. Last, the grid $\Delta x = 0.023m$ will be simulated for several values of ϵ , and compared to a coarser grid, $\Delta x = 0.031m$, and a finer grid, $\Delta x = 0.019$, also simulated for multiple values of ϵ .

THRUST OVER TIME

Figure 67 from the last section shows thrust over time for the cases with grid refinement around the rotor. The result for $\Delta x = 0.023m$ can clearly be seen in red, with a lot of noise in the plot. Appendix B takes a closer look at the noise in the thrust over time plots, and shows that the noise should decrease with growing ϵ . This means that it should decrease with $\epsilon/\Delta x$. $\Delta x = 0.023m$ clearly deviates from this trend, while the coarser and fine grid resolutions follow the trend.

C THRUST AND POWER GRAPHS FOR $\epsilon = 0.0084M$ AND $\Delta X = 0.023M$

Figure 69 from the last section show the thrust over time for the cases without grid refinement around the rotor. The figure shows the same tendency as figure 67. $\Delta x = 0.023m$ has a large variance and the mean value of the thrust is far below the others. This illustrates more clearly that the simulation for $\Delta x = 0.023m$ does not fit in to the trend of the simulations for coarser and finer grid resolutions.

SIMULATIONS FOR MORE GRID RESOLUTIONS

To investigate the dip further, more simulations were performed at grids close to the grid producing this point, and the result can be seen in figure 78.

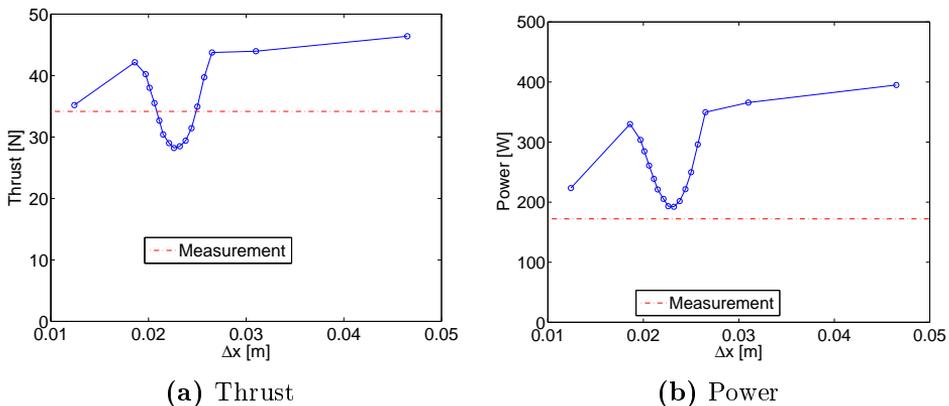


Figure 78: Thrust and power for varying grids without grid refinement around the rotor for constant $\epsilon = 0.0084m$.

Between the highest and lowest point in the dip, there is a difference in the thrust of 30% and in the power of 37%. This is a large variation, and can not be neglected.

The power and thrust graphs have a continuous dip for some grid resolutions, and it is not only one point that deviates from the general trend. This leads to a conclusion that the dip is caused by a numerical error, and not caused by a faulty simulation.

C THRUST AND POWER GRAPHS FOR $\epsilon = 0.0084M$ AND
 $\Delta X = 0.023M$

In addition, simulations performed in section 5.2.3 where the grid resolution is changed only in the x -direction and only in the rotor plane showed that the dip appeared only when the x -direction grid was changed. The grid spacing where the dip occurred was $\Delta x = 0.03m$, i.e. a little larger than $0.023m$. This shows that the dip is dependent on the grid in the streamwise direction.

SIMULATIONS FOR MORE VALUES OF ϵ

The results above are for a constant ϵ of $0.0084m$. Simulations have also been performed where ϵ was kept at $\epsilon = 2.5\Delta x$. The difference between these simulations is only ϵ , but the simulations with $\epsilon = 2.5\Delta x$ did not produce deviations for $\Delta x = 0.023m$. Because of this, ϵ must have a large influence on the problem.

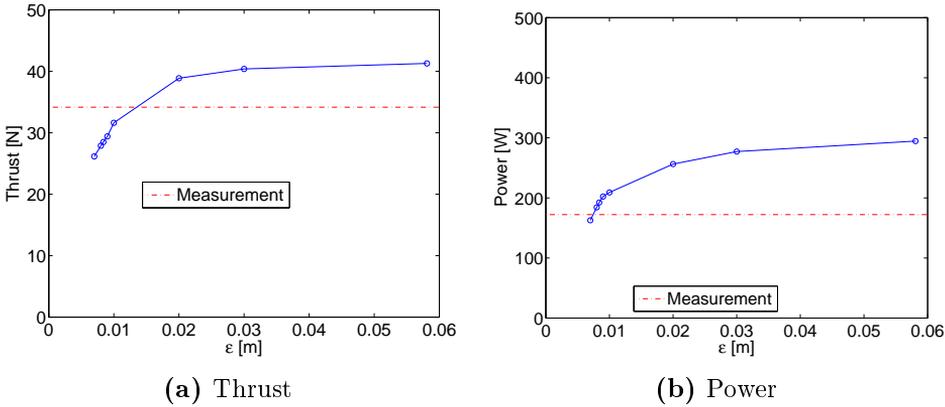


Figure 79: Thrust and power for $\Delta x = 0.023m$, for different ϵ . $\epsilon/\Delta x$ goes from 0.3 to 2.5.

Figure 79 shows simulations performed with $\Delta x = 0.023m$ for different values of ϵ . The relationship between ϵ and the grid resolutions range from $\epsilon/\Delta x = 0.3$ up to $\epsilon/\Delta x = 2.5$. The thrust increases with ϵ , and it stabilizes around $\epsilon/\Delta x = 1$.

In figures 27b and 28b from the parameter study, it was seen that the

C THRUST AND POWER GRAPHS FOR $\epsilon = 0.0084M$ AND $\Delta X = 0.023M$

power curve for varying ϵ has a bottom point around $\epsilon/\Delta x = 1$, and no such bottom point seems to be present in figure 79. For both thrust and power the results plummet for small ϵ . It would be interesting to know the reason to this trend.

For comparison, simulations for several ϵ have also been performed for one larger and one smaller grid. Figure 80 shows simulations with $\Delta x = 0.031m$ for different values of ϵ . The relationship between ϵ and the grid resolution ranges from $\epsilon/\Delta x = 0.38$ to $\epsilon/\Delta x = 2.5$. Figure 81 shows simulations with $\Delta x = 0.019m$ for different values of ϵ . The relationship between ϵ and the grid resolution ranges from $\epsilon/\Delta x = 0.23$ to $\epsilon/\Delta x = 2.5$.

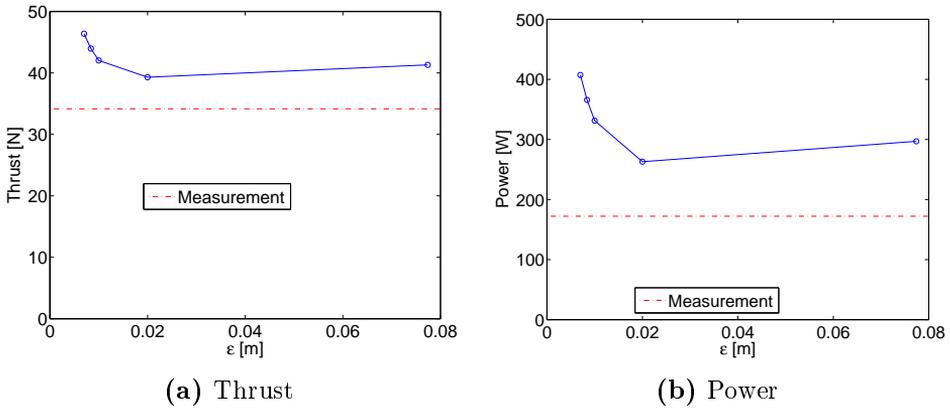


Figure 80: Thrust and power for a grid spacing of $\Delta x = 0.031m$ for different ϵ . $\epsilon/\Delta x$ goes from 0.38 to 2.5.

For $\Delta x = 0.023m$, both thrust and power has a significant declination for small ϵ , while the opposite is true for larger and smaller grid resolutions. In addition, the difference in thrust and power compared to coarser and finer grids occur at around $\epsilon/\Delta x = 1$. This is where the thrust and power for $\Delta x = 0.023m$ starts to decline. For $\epsilon/\Delta x > 1$, the difference in thrust and power is small between $\Delta x = 0.023m$ and the other grid resolutions.

It is worth to note that the variation in both thrust and power is lower for

C THRUST AND POWER GRAPHS FOR $\epsilon = 0.0084M$ AND
 $\Delta X = 0.023M$

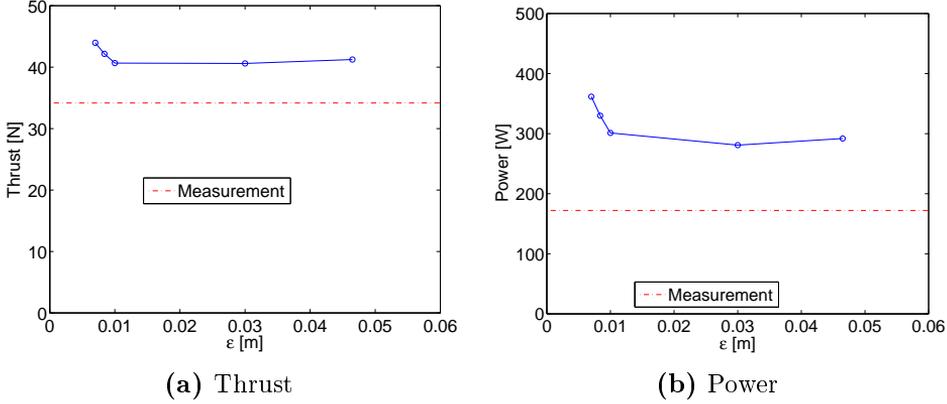


Figure 81: Thrust and power for a grid spacing of $\Delta x = 0.019m$ for different ϵ . $\epsilon/\Delta x$ goes from 0.23 to 2.5.

$\Delta x = 0.19m$ than for $\Delta x = 0.31m$. This implies (as mentioned before) that the solution is less dependent on ϵ for fine grids than for coarse grids.

The OpenFOAM processing tool was also used to search for an explanation to this deviation, by looking at among other things the velocity, pressure and body forces acting on the turbine.

Figure 82 shows the pressure variation across the rotor plane. This shows clearly that the blade forces are not distributed smoothly across the blade. This is because ϵ is too small for the grid, and a similar trend can also be seen for finer grid resolutions (see figure 74) and coarser grid resolutions. It is clear that the uneven distribution of forces across the blade can create oscillations. This might be the reason for the discrepancy for $\Delta x = 0.023m$ and $\epsilon = 0.0084m$.

C THRUST AND POWER GRAPHS FOR $\epsilon = 0.0084M$ AND $\Delta X = 0.023M$

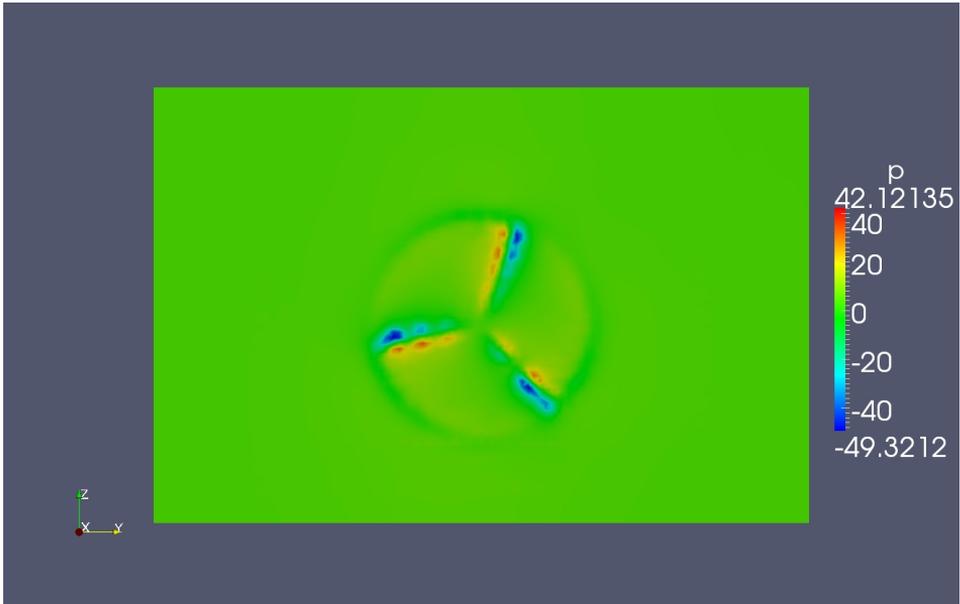


Figure 82: Pressure field in the rotor plane, showing that ϵ is too small for the grid to create a smooth distribution of the blade forces.

Other possible explanation for this discrepancy include:

- A resonance appear in the simulation for this exact set-up.
- The actuator line implementation contains a bug.
- A numerical phenomenon in the solution algorithm appears for this exact set-up, e.g. an error occurs in the under-relaxation procedure.
- This set-up leads to unexpected difficulties for the turbulence model, a shock in the turbulence model leads to a shock in the turbine and vice versa, creating an escalating problem.