



Anna Avramenko

# CFD-BASED OPTIMIZATION FOR WIND TURBINE LOCATIONS IN A WIND PARK



Anna Avramenko

## **CFD-BASED OPTIMIZATION FOR WIND TURBINE LOCATIONS IN A WIND PARK**

Dissertation for the degree of Doctor of Science (Technology) to be presented with due permission for public examination and criticism in the Auditorium 1316 at Lappeenranta-Lahti University of Technology, Lappeenranta, Finland on the 10<sup>th</sup> of May, 2019, at noon.

Acta Universitatis  
Lappeenrantaensis 851

Supervisors Professor Jari Hämäläinen  
LUT School of Engineering Science  
Lappeenranta-Lahti University of Technology  
Finland

Professor Sergey Lupuleac  
Institute of Applied Mathematics and Mechanics  
Department of Applied Mathematics  
Peter the Great St. Petersburg Polytechnic University  
Russian Federation

Reviewers Ph.D. Antonio Segalini  
Department of Mechanics  
KTH Royal Institute of Technology  
Sweden

Associate-Professor Gabor Janiga  
Institute of Fluid Dynamics and Thermodynamics (ISUT)  
Otto von Guericke University Magdeburg  
Germany

Opponent Professor Jens Nørkær Sørensen  
Technical University of Denmark (DTU)  
Denmark

ISBN 978-952-335-366-4  
ISBN 978-952-335-367-1 (PDF)  
ISSN 1456-4491  
Lappeenranta-Lahti University of Technology LUT  
LUT University Press 2019

# Abstract

**Anna Avramenko**

**CFD-based optimization for wind turbine locations in a wind park**

Lappeenranta 2019

82 pages

Acta Universitatis Lappeenrantaensis 851

Diss. Lappeenranta-Lahti University of Technology

ISBN 978-952-335-366-4, ISBN 978-952-335-367-1 (PDF), ISSN 1456-4491

Searching for the optimal positions of wind turbines in a wind farm for their maximal power generation has an important and difficult role in the wind energy industry. Several factors have to be considered while designing a wind farm. These include geographical constraints, the maximum desired installed capacity of the wind farm, wake effects, noise assessment, the total cost and visual impact (Mittal and Taylor (2012)). The fundamental aim is to reduce the total costs associated with the wind farm while maximizing power production. 'Micro-siting' is the process of optimizing the layout of the wind farm.

The determination of wind turbine positions can be treated as an optimization problem that can be solved by various methods. In this thesis, an optimization tool based on evolutionary algorithms is developed together with a fast computational fluid dynamics (CFD) model to improve solutions for the wind turbine placement problem.

This work presents the development of a wind farm optimization tool with CFD based on the Reynolds-averaged Navier-Stokes (RANS) approach. First, the use of CFD in wind farm modelling is validated with laboratory measurements. This is the starting point for this study. Since accurate three-dimensional (3D) modelling is very time-consuming, a fast depth-averaged modelling method was developed for the wind farm optimization process. The developed fast modelling method was validated with accurate 3D modelling. Even though some of the flow characteristics are lost, the depth-averaged model predicts the velocity and power sufficiently well.

In wind farm optimization, the monitored values and objective function have nonsmooth behaviour when CFD simulations are used. Therefore, optimization of the wind farm has been performed with evolutionary algorithms to avoid difficulties in the gradient calculations of the objective function. In general, evolutionary algorithms are shown to be very convenient for this optimization problem, although they require a lot of CFD simulations during the optimization process. Power maximization was chosen as an objective function for optimization in the presented case examples. The CFD-based optimization of the wind farm introduced in this thesis allows researchers to find a more effective wind turbine layout than before. Even if the optimization algorithm is only implemented for one real hill in this work, it can be extended to different geographical geometries as well.

Keywords: wind farm, optimization, CFD, evolutionary algorithms.



## **Acknowledgements**

This study was carried out in the LUT School of Engineering Science at Lappeenranta-Lahti University of Technology, Finland, during 2011-2019.

I want to express my gratitude to my supervisor, Professor Jari Hämäläinen, for providing guidance during all these years. Special thanks go to my supervisor, Professor Sergey Lupuleac, who shared his time and knowledge about CFD. I also want to thank my colleagues from Peter the Great St.Petersburg Polytechnic University, especially Evgeny Petukhov, for helping me with practical aspects of the optimization algorithm.

Besides my advisor, I would like to thank the rest of my thesis committee: Associate-Professor Gabor Janiga, Ph.D. Antonio Segalini, and Professor Jens Nørkær Sørensen, for their insightful comments and encouragement, but also for the hard question which incited me to widen my research from various perspectives.

Special thanks go to my parents, Anzhelika Kosmacheva and Vasiliy Kosmachev, for their continuous support during my whole life. Most of all, I would like to thank my husband, Yury Avramenko, for his love and encouragement, and my children, Victor and Tatiana, for those moments when I can stop thinking about work and studying.

Thanks.

Anna Avramenko  
May 2019  
Lappeenranta, Finland



# Contents

Abstract

Acknowledgments

Contents

List of Symbols	9
<b>1 Introduction</b>	<b>11</b>
1.1 Wind turbine and wind farm . . . . .	11
1.2 The CFD technique . . . . .	12
1.3 Wind turbine modelling . . . . .	14
1.4 Wind farm optimization . . . . .	15
1.5 The objectives and contents of the thesis . . . . .	18
<b>2 The depth-averaged governing equations and optimization algorithm</b>	<b>19</b>
2.1 Depth-averaged governing equations . . . . .	19
2.2 The AD Model . . . . .	25
2.2.1 The AD model for 3D simulation . . . . .	26
2.2.2 The AD model for depth-averaged numerical modelling . . . . .	30
2.3 Evolutionary algorithms . . . . .	31
<b>3 Validation and CFD results</b>	<b>37</b>
3.1 The RUSHIL wind tunnel study . . . . .	37
3.2 Wind tunnel tests in the PRISME laboratory . . . . .	44
3.3 The Lillgrund wind farm . . . . .	48
3.4 The flow simulation of the wind turbine on the hill . . . . .	52
3.4.1 Simple case . . . . .	53
3.5 Askervein Hill . . . . .	54
<b>4 Optimization results</b>	<b>63</b>
4.1 The simple hill . . . . .	64
4.2 Askervein Hill . . . . .	67
<b>5 Conclusions</b>	<b>75</b>
<b>References</b>	<b>77</b>



## List of Symbols

$2D$	Two-dimensional
$3D$	Three-dimensional
$ABL$	Atmospheric boundary layer
$AD$	Actuator disc
$AL$	Actuator line
$AS$	Actuator surface
$CFD$	Computational Fluid Dynamics
$CIA$	Cluster identification algorithm
$CPU$	Central processing unit
$DA$	Depth-averaged
$DE$	Differential evolution
$DES$	Detached-eddy simulation
$DNS$	Direct numerical simulation
$EU$	European Union
$HAWT$	Horizontal axis wind turbines
$LES$	Large eddy simulation
$MIP$	Mixed-integer programming
$MOGA$	Multi-objective genetic algorithm
$RANS$	Reynolds-averaged Navier-Stokes
$RRF$	Rotating reference frame
$UDF$	User-defined functions
$VAWT$	Vertical axis wind turbines
$\alpha$	Permeability
$\beta, \beta_{xx}, \beta_{xy}$	Boussinesq coefficients
$\gamma$	Vector of optimization variables
$\epsilon$	Dissipation rate
$\kappa$	Kármán's constant
$\mu$	Dynamic viscosity
$\mu_t$	Turbulent eddy viscosity
$\nu$	Kinematic viscosity
$\xi$	Arbitrary parameter
$\rho$	Density
$\sigma_k$	Turbulent Prandtl number for turbulent kinetic energy
$\sigma_\epsilon$	Turbulent Prandtl number for turbulent dissipation rate
$\bar{\tau}$	Stress tensor
$\omega(i)$	Frequency of the wind corresponding to the $i$ th-wind direction
$\Omega$	Set of constraints
$a$	Axial induction factor
$c_k, c_\epsilon$	Empirical constants
$c_f$	Friction coefficient
$d_1$	Bottom level of the channel
$d_2$	Top level of the channel

$g$	Gravity
$\Delta m$	Thickness of the medium
$k$	Turbulence kinetic energy
$n$	Constant
$\vec{n}$	Normal vector
$p$	Pressure
$r_1, r_2, r_3$	Constants
$\vec{u} = (u_1, u_2, u_3)$	Velocity vector
$u'_i u'_j$	Reynolds stress
$A$	Area
$B$	Number of decision variables
$C_{1\epsilon}, C_{2\epsilon}, C_{3\epsilon}$	Constants used in $k - \epsilon$ equations
$C_2$	Inertial resistance factor
$CS$	Control surfaces
$C_T$	Thrust coefficient
$CV$	Control volume
$D$	Depth of the channel
$DM$	Diameter
$E$	Absolute error
$\langle f \rangle$	Depth-averaged function
$F$	Scale factor
$F_A$	Force
$G_b$	Production of turbulence kinetic energy due to buoyancy
$G_k$	Production of turbulence kinetic energy
$H$	Hill height
$HH$	Hub height
$L$	Characteristic length
$NP$	Size of the population
$P$	Total power of the wind farm
$P_{kv}, P_{\epsilon v}$	Source terms
$P_g$	Population of generation $g$
$P_h$	Production of $k$
$P_i$	Power of the turbine
$Re$	Reynolds number
$S$	$z$ -directional velocity profile
$S_i, S_{mx}, S_{my}$	Momentum source terms
$S_{mass}$	Mass source
$U_*$	Friction velocity
$U_\infty$	Free-stream velocity
$V$	Volume
$Y_m$	Contribution of the fluctuating dilatation in compressible turbulence

# 1 Introduction

In this work, wind turbine placement is optimized in a wind farm using an objective function that represents the cost per unit of power produced by the wind farm for a particular wind distribution function. The wind distribution function, in general, represents a model of wind variations in terms of speed and direction, averaged over a month or many months. A evolutionary algorithm is employed for optimizing the placement of the wind turbines. The developed two-dimensional computational fluid dynamics (CFD) model is utilized to calculate the depth-averaged velocity of the wind farm.

## 1.1 Wind turbine and wind farm

A wind turbine is a device that converts kinetic energy into electrical energy. This is achieved by blades, which are attached to a hub that rotates in response to the aerodynamic force of the wind on the blades. This rotation drives a generator that produces electricity that is transferred to the electrical power grid. A wind farm is a group of collocated wind turbines and may be thought of as a wind-driven power station.

Wind power produces electric power from wind flow through wind turbines. Wind energy is a renewable and a promising alternative power source in comparison with power produced by burning fossil fuels.

In 2017 16.8 GW of additional wind power capacity was installed in Europe and 15.7 GW in the European Union (EU) according to Fraile and Mbistrova (2018). It was a record year for annual installations. Next to gas installations, wind energy is the second largest form of power generation in Europe, where the total net installed capacity is 169 GW. Europe increased its wind energy in 2017 by 20% compared to 2016. Eighty per cent of all wind power was produced onshore – and the rest was produced offshore. Germany installed 6581 MW of wind power capacity in 2017, increasing its capacity by 15%. The United Kingdom comes after Germany with 4270 MW of installed capacity. This is five times more installations than in 2016. In third place is France with 1694 MW (9% growth on the previous year). The next three countries are Finland (577 MW), Belgium (467 MW) and Ireland (426 MW).

In this thesis only horizontal axis wind turbines (HAWT) are considered. HAWTs have more advantages compared to vertical axis wind turbines (VAWT). For example, the power coefficients of HAWTs are higher than in VAWTs. Also, they are more stable in mechanical behaviour and therefore, their larger size is acceptable. Nowadays, VAWTs are not commercially competitive and they are not currently produced in significant quantities (Carcangiu (2008); Burton et al. (2001)). More information about VAWTs can be obtained from the work of Hau (2000).

Many wind farm installations have been built offshore, but recently a large number are

being built onshore (Hyvarinen and Segalini (2017)). However, it should be noted that the terrain can be complex, with hills and valleys. Therefore, deep knowledge of how wind turbine power depends on the surrounding terrain is necessary in order to identify the optimal placements of wind turbines at onshore sites.

The practical constraints associated with operating and developing a wind farm relate to the suitability of the land and access to it, wind speeds and the available equipment. This leads on to considering the interaction of each wind turbine with its neighbours and that their wakes contribute to the loss of power. Wake evaluation requires accurate modelling of all the factors that affect its development, such as the turbulence and flow speed, the form of the atmospheric boundary layer (ABL), features of the turbine and the complexity of the terrain (Vafiadis et al. (2013)).

Numerical modelling and optimization are used in industry to improve different characteristics, such as the design of a product, final efficiency and the cost of a product. Laboratory tests are inefficient for product optimization because of their cost, time requirements and trial-and-error methods. Therefore, nowadays numerical methods are very popular because they provide the required information using simulation. Even this, however, does not deal with cases when the effect of design or process parameters on the final product is not exactly known. Thus, mathematical optimization is used to find the best process combinations and design parameters.

## 1.2 The CFD technique

CFD is a part of fluid mechanics, which uses data structures and numerical analysis to solve and analyze problems that involve fluid flows. The Navier-Stokes equations, which are discussed in the next chapter, are the fundamental concept of almost all CFD problems. CFD is a technique for replacing the partial derivatives or the integrals in equations with discretized algebraic forms, which are then solved to obtain the flow field values at discrete points in time and/or in space (Anderson (1995)). The end product of CFD is a number of values as opposed to a closed-form analytical solution.

The CFD technique is very powerful and is used in a wide range of non-industrial and industrial application areas, such as in the aerodynamics of aircraft, turbomachinery, electrical engineering, chemical process engineering and biomedical engineering. Usually the CFD procedure consists of three main components: a pre-processor, a solver and a post-processor (Versteeg and Malalasekera (1995)). In the pre-processor a user formulates the problem to be solved. It involves determining the computational domain (the geometry of the model), grid generation (the division of the geometry into sub-domains), boundary condition assignment, setting the physical parameters of media, and so on.

The solver is used for problem solving, with help of different numerical methods, such as finite volume and finite element methods. For example, in ANSYS Fluent it is possible to

choose pressure-based or density-based numerical methods (Versteeg and Malalasekera (1995); Kosmacheva (2011)). Velocity is obtained from the momentum equations for both methods. In the pressure-based method, the pressure is obtained by solving a pressure or pressure correction equation. In the density-based approach, the density is obtained from the continuity equation while the pressure is determined from the equation of state. ANSYS Fluent solves the governing integral equations for the conservation equations and in both cases a control volume-based technique is used.

In the post-processor the user works with the results using visualization tools. These include: vector plots, 2D and 3D surface plots and grid display.

One main goal in the numerical modelling of turbulent flows is to create a model that can obtain general quantities, such as fluid velocity, for use in the modelled system. For turbulent flows, the complexity of the phenomena involved in turbulence and the range of length scales make most modelling methods very expensive. The resolution required to resolve all turbulence scales is computationally impossible. The basic approach in such cases is to obtain numerical modelling methods to approximate unresolved phenomena. Various CFD models have been proposed to resolve the original 3D Navier-Stokes equations and to model the flow field over wind turbines:

- Reynolds-Averaged Navier-Stokes (RANS) method. The RANS approach cannot predict turbulent eddy motions, because the Reynolds averaging method consists of replacing the randomly changing flow characteristics (velocity, pressure, density) by the sums of the averaged and fluctuating components. This method is explained in more detail in Section 2.1.
- Direct numerical simulation (DNS). DNS does not contain any additional equations. The time-dependent Navier-Stokes equations are solved with a very small time step on a dense spatial grid. Because of the large volume of information obtained during numerical simulation, the most important aspect is the average flow values obtained during problem solving. However, some researchers use this method for wind turbine investigation (for example, in Rongliang et al. (2013) and Qamar et al. (2014)).
- Large eddy simulation (LES). LES is a grid filtering approach, based on the hypothesis that the large scales of turbulence are the most influential and the smallest scales are universal and can be modelled. Therefore, LES directly resolves the turbulent structures larger than the mesh size, but at the same time models the turbulent structures smaller than the mesh size (Agafonova (2017)). LES is a very popular tool due to the accurate flow computation. LES is currently applied to the simulation of the ABL and flow behaviour investigation by using a wind turbine or wind farm (Wu and Porte-Agel (2011); Agafonova (2017); Martinez-Tossas et al. (2015)).
- Hybrid RANS-LES methodology. The difficulties associated with the use of the standard LES models, particularly in near-wall regions, has led to the development of hybrid models that attempt to combine the best aspects of RANS and LES

methodologies in a single solution strategy. An example of a hybrid technique is the detached-eddy simulation (DES) approach. This model attempts to treat near-wall regions in a RANS-like manner, and treat the rest of the flow in a LES-like manner. This method reduces the computational cost compared to LES or direct numerical simulation (DNS). The DES method was used by Sorensen et al. (2004) for CFD computations of wind turbine blade loads.

However, in a complex domain, CFD-RANS simulations are still widely used by researchers, because of their cheap computational cost relative to the more advanced LES or DES; this was the case in the work of Stergiannis et al. (2017), Astolfi et al. (2018), Zhang et al. (2017) and Souza et al. (2017). In this work, the ANSYS Fluent solver was used for CFD-RANS simulation of the flow behaviour in the wind farm located on a non-flat terrain. The considered wind farm consists of one to five Nordtank NTK 500/41 wind turbines with LM 19.1 m blades, as described by Mikkelsen (2003).

### 1.3 Wind turbine modelling

Numerical modelling is a very effective method for the analysis of wind turbine performance and aerodynamics. Thus a rise in the number of numerical studies has been seen during recent years for all HAWT aerodynamics features, performed on many different levels, ranging from the actuator disc (AD) model, integrated by CFD calculations, to the full 3D Navier-Stokes models that have been carried out by Kang and Hirsch (2001). Here, several methods are presented, with various degrees of accuracy, to simulate turbines on a wind farm. Four models will be considered: rotating reference frame (RRF), actuator line (AL), actuator surface (AS) and AD models (Adamski (2013)).

The RRF model is used for a full scale 3D RANS numerical study. In this method Navier-Stokes equations are resolved in a stationary (inertial) reference frame or moving (non-inertial) reference frame. The 3D CFD-RANS simulation of the Nordtank NTK 41/500 was investigated by using the RRF model in the work of Carcangiu (2008). Different turbine characteristics were compared with experiments and previous research. Although this modelling describes the flow behaviour in the near wake region and around wind turbine blades, the RRF model is computationally expensive because it requires complex meshing. Therefore, less expensive methods are presented to calculate the performance of the wind turbine and behaviour of the flow field.

To reduce computational efforts, various methods have been developed, ranging from the simplest AD model to more complex ones, such as the AL and AS models. While the AD method assumes the turbine to be a porous medium disk, the AL and AS models represent the blades of a turbine as a line or a surface. The main advantage of these models is the presentation of separate blades by airfoil data. However, this method requires strong knowledge of aerodynamics to describe blade airfoils. The AD method was chosen in this work for several reasons:

- to reduce the computational time of CFD simulation,
- the real aerodynamic profile parameters are not needed in 2D DA (depth-averaged) modelling,
- the AD model is adopted in most commercial CFD.

The CFD technique is a very promising tool for investigating wind farms, but one main problem is the complex 3D geometry of wind farms. The pre-processing (i.e. generating a good quality mesh and modelling the detailed geometry) is very time consuming. Furthermore, the actual simulation for a 3D wind farm takes a few days with a standard computer. Therefore, the development of faster approaches in order to produce a tool for real design work are needed, such as using depth-averaged equations for wind flow modelling. Depth-averaged equations have been used for flows in a closed channel, such as in the headbox of a paper machine (Hämäläinen and Tiihonen (1995); Hämäläinen et al. (2008)). Also, this method was initially used for the optimization of Chevron-type plate heat exchangers by Lyytikäinen et al. (2009) and it presents the general flow behaviour with very good accuracy.

Using the depth-averaged method, only one 2D mesh should be built and the channel's depth can be represented horizontally with the source terms as in Navier-Stokes equations. Thus, the complicated pre-processing work of the geometry generation can be ignored. Also in this case, simulation is much faster, because this method uses a 2D mesh instead of a 3D mesh. The main problem for a fast numerical modelling tool based on depth-averaged equations is simulating the wind turbine in the 2D case. Unfortunately, it is not possible to make a simulation of the wind turbine in a general way for 2D modelling. Therefore, a new 2D AD approach was developed specially for this case using the AD model. The porous media model in ANSYS Fluent was used to implement the AD model for 3D simulation, and source terms were added to the Navier-Stokes equations to obtain satisfactory results for 2D modelling.

Three different numerical examples were considered. Firstly, flow simulation over a 2D hill without the turbine was modelled in 3D and 2D. The results were compared with experiments to check how the derived 2D equations predict averaged velocity. Then, 2D and 3D numerical modelling of one wind turbine on a smooth terrain were considered to evaluate how the new 2D AD method predicts the averaged velocity field. Finally, simulation of the wind turbine on Askervein Hill was done to combine both of these methods.

## 1.4 Wind farm optimization

One wind farm optimization problem means to find the optimal positions of wind turbines in a wind farm in order to maximize the total power of a wind farm. The main challenges in using wind energy are its temporal instability and spatial non-uniformity (Song et al.

(2014)). The wake flow of wind turbines and the terrain topography are important reasons for the spatial non-uniformity. A wind turbine produces regions of wake flow where the velocity of the wind decreases. The influence of the wake flow on the downstream turbine reduces the power performance and the wind farm's efficiency. The problem is more complicated if the effects of a complex terrain are considered. Thus, a large amount of theoretical work has been done to develop methods for the accurate evaluation of wind resources.

The wake flow of wind turbines was investigated over many years and different models have been obtained by researchers for later use in optimization. These methods can be divided into two main groups: analytical and computational models. The wake velocity in the analytical method is described by a group of analytical terms, whereas in the computational method, as described above, Navier-Stokes equations are solved numerically to achieve the wake velocity field.

For flat terrain, the logarithmic velocity profile (Ohya (2001)), statistical methods (Akpınar (2013)) and Weibull distribution (Kidmo et al. (2015)) are effective. For complex terrains, numerical models such as the CFD technique (Uchida and Ohya (2003)) are more suitable for optimization and evaluation. As for its temporal instability, many investigations have studied the prediction of wind speed and power performance of wind farms. Statistical methods for the prediction of wind velocity include the measure-correlation-prediction (MCP) method (Dinler (2013)), the ARMA algorithm (Wu et al. (2014)) and the Gaussian process regression approach (Yu et al. (2013)). In a similar manner, for scenarios with complex terrain, combined models (Wu et al. (2014)) or numerical models are required for reliability. This work concentrates on the wind farm layout optimization problem, which is related to the spatial non-uniformity.

Many optimization methods can be used for the wind farm optimization problem, including the genetic algorithm, binary coding (Grady et al. (2005)) and real coding (Wan et al. (2009)), particle swarm optimization (Wan et al. (2010); Gu and Wang (2013)), greedy algorithm (Ozturk and Norman (2004)), the Monte Carlo method (Marmidis et al. (2008)), simulated annealing (Rivas et al. (2009)) and lazy greedy algorithm (Zhang et al. (2011); Song et al. (2014)). The linear method is used to model turbine wake flow in most of these studies, which was first proposed by Jensen (1983) and Katic et al. (1986).

To obtain the wind resources at a given location, the software called WASP, developed by the Riso National Laboratory, was designed. The program includes analysis of the efficiency and production of the wind farm and climate estimations (Herbert-Acero et al. (2014)). The wake model of Katic et al. is incorporated in the software. Nowadays, WASP is considered a standard program for the analysis of wind resources and is frequently used in several optimization software packages, like WindFarmer v5.2 and WindPro.

Segalini and Castellani (2017) compared three independent software packages to estimate the flow field and the power production of a wind farm located on complex terrain. Three

programs were considered: ORFEUS (linearised solver), WASP and WindSim (fully non-linear solver). WindSim and ORFEUS use the AD method to simulate the turbines. According to the results, the two linearised solvers allow for a numerically-efficient and faster solution when assessing polar efficiency.

The interaction of the wake flow of wind turbines should be studied in order to perform wind farm optimization on non-flat terrain. Song et al. (2014) proposed a greedy algorithm for wind farm optimization where the virtual particle model is used for wake flow analysis, which allows the present method to be applied for non-uniform flow on non-flat terrains. Also, cluster identification algorithm (CIA) and multi-objective genetic algorithm (MOGA) were described and implemented by Rosales (2012) to maximize wind power and wind power efficiency at a wind farm located on complex terrain.

The effectiveness of the CFD method for wind farm optimization was demonstrated in previous studies. Kuo et al. (2016) proposed a method that combines CFD with mixed-integer programming (MIP) for solving a wind farm optimization problem on non-flat terrains. MIP is applied for optimization, while CFD modelling is used for accuracy improvement of the wake deficit predictions. Feng and Shen (2014) initially simulated wind flow behaviour on non-flat terrains without wind turbines using CFD. Then an adapted Jensen wake model was introduced, using the inflow conditions as input and considering the terrain features. Using this combined method and a random search algorithm, the optimization problem was solved on a Gaussian-shaped hill.

CFD-based modelling applied to wind farm design gives invaluable information about the efficiency of the wind farm. Traditional ways to optimize simulation results in the design are trial-and-error experiments or systematically changed geometry parameters. However, they are quite impractical and time-consuming when the behaviour of the simulation results with design parameters is not well understood or when the number of the design parameters is high. Therefore, modelling is often coupled with a mathematical optimization algorithm. Using experiments or heavy simulation methods, such as 3D CFD modelling, would lead to a months-long central processing unit (CPU) time when coupled with mathematical optimization. Hence, faster modelling methods are needed for the real optimization processes.

Therefore, a depth-averaged method was applied to the governing equations in this work to reduce the dimension of the domain from 3D to 2D, due to which the simulation time decreased by several times. This thesis proposes an algorithm that couples 2D DA CFD modelling with evolutionary algorithms in order to optimize the positions of wind turbines on complex terrains.

## 1.5 The objectives and contents of the thesis

The aim of this thesis is to develop a new approach to the wind farm optimization problem. This is done by a combination of 2D DA CFD modelling and an evolutionary algorithm. Fast 2D DA numerical modelling was performed by using depth-averaged equations that were derived by integrating 3D Navier-Stokes equations over the height of the domain. The goal of this derived approach was to reduce the computational time of CFD simulations. With this method only one 2D mesh is needed and then the height of the domain can be presented in every horizontal position with the source terms in governing equations. Thus, the geometrical pre-processing work can be ignored. To include turbines in 2D simulations, the new 2D AD method was developed. This approach allows a satisfactory velocity field to be obtained compared to 3D modelling. Wind farm optimization was done using genetic algorithms.

Chapter 2 provides the derivation of the depth-averaged equations. This derivation is based on our journal papers and conference articles (Avramenko and Haario (2015); Avramenko et al. (2015); Avramenko et al. (2016); Avramenko et al. (2017)). The equations are solved with the commercial CFD software package ANSYS Fluent. The model is coupled with the optimization algorithm via a Matlab-based interface program. The optimization algorithm and the optimization procedure are also explained in Chapter 2. The whole simulation setup, including the modelled geometries, model parameters and simulation procedure, is presented in Chapter 3. Also in this chapter, the 3D simulations are compared with experiments and then 3D simulations are used for the validation of the 2D DA model. The optimization setup and results are presented in Chapter 4. Finally, the conclusions of this thesis and some suggestions for future work are introduced in Chapter 5.

## 2 The depth-averaged governing equations and optimization algorithm

This chapter is devoted to the theory. The modelling equations and optimization algorithm used in this thesis are presented here. The fast depth-averaged approach is used in the optimization and the accurate 3D CFD model is applied for the validation of this fast model. The 3D modelling is also compared with experiments.

### 2.1 Depth-averaged governing equations

In order to gain a good insight into the equations for flow in two horizontal dimensions, they are derived by integration of 3D flow over the depth of the channel. For incompressible and steady-state flow, the equations can be written as the system of RANS equations in the following form:

$$\nabla \cdot \vec{u} = 0, \quad (2.1)$$

$$\rho \nabla \cdot (\vec{u} \vec{u}) = -\nabla p + \nabla \cdot \bar{\tau}, \quad (2.2)$$

where  $\rho$  is density,  $p$  denotes pressure,  $\bar{\tau}$  is the stress tensor and  $\mu$  is the effective viscosity. All terms are time-averaged in RANS equations. The effective viscosity includes dynamic and turbulent-eddy viscosities.

The nonlinear Reynolds stress term of RANS requires additional modelling to close the RANS equation for solving, and has led to the creation of many different turbulence models. Stergiannis et al. (2017) considered full HAWT rotor CFD simulations using different RANS turbulence models and compared them with the AD model and experiments. As a result, the  $k - \epsilon$  turbulence model showed excellent agreement with experiments in the mid-wake and far wake. Even if the standard  $k - \epsilon$  model does not accurately predict small-scale effects, it is acceptable for high Reynolds number flows and it is used in 3D and depth-averaged modelling in this thesis. The turbulent kinetic energy  $k$  and its dissipation rate  $\epsilon$  are obtained from the following transport equations:

$$\rho \vec{u} \cdot \nabla k = \nabla \cdot \left[ \left( \mu + \frac{\mu_t}{\sigma_k} \right) \nabla k \right] + P_k - \rho \epsilon, \quad (2.3)$$

$$\rho \vec{u} \nabla \cdot \epsilon = \nabla \cdot \left[ \left( \mu + \frac{\mu_t}{\sigma_\epsilon} \right) \nabla \epsilon \right] + C_1 \frac{\epsilon}{k} P_k - C_2 \rho \frac{\epsilon^2}{k}, \quad (2.4)$$

where the rate of turbulent energy production is modelled by:

$$P_k = \frac{\mu_t}{2} |\nabla \vec{u} + \nabla \vec{u}^T|^2, \quad (2.5)$$

$$\mu_t = \rho C_\mu \frac{k^2}{\epsilon}. \quad (2.6)$$

The RANS equations (2.1)-(2.2), the two coupled transport equations (2.3)-(2.4) and the eddy viscosity formula (2.6) form a closed system of equations for  $\vec{u}$ ,  $p$ ,  $k$ ,  $\epsilon$ ,  $\mu_t$ .

Depth-averaged conservation laws need to be derived in the same form as standard 2D-equations in ANSYS Fluent, such that all extra terms are included in the source terms. In addition, it has to be noted that velocities are depth-averaged velocities so that the depth-averaged velocity vector is  $\langle \vec{u} \rangle = (\langle u_1 \rangle, \langle u_2 \rangle)$  where

$$\langle u_i(x, y) \rangle = \frac{1}{D} \int_{d_1}^{d_2} u_i dz \quad i=1,2, \quad (2.7)$$

where  $D(x, y) = d_2(x, y) - d_1(x, y)$  is the depth of the channel.

The geometry will be represented by the channel, where a bottom surface is a non-flat terrain and the top surface is the symmetry boundary condition which can be written as

$$\frac{\partial u_1}{\partial z} = 0, \quad \frac{\partial u_2}{\partial z} = 0, \quad u_3 = 0. \quad (2.8)$$

The wall boundary condition on the complex terrain can be presented in the following form:

$$u_1(x, y, d_1) = 0, \quad u_2(x, y, d_1) = 0, \quad u_3(x, y, d_1) = 0. \quad (2.9)$$

Assuming that gravity is insignificant, the pressure can be approximated as a constant in the  $z$ -direction.

In ANSYS Fluent, the steady-state conservation laws with source terms and using depth-averaged velocities are in the form of

$$\nabla \cdot (\rho \langle \vec{u} \rangle) = S_{mass}, \quad (2.10)$$

$$\nabla \cdot (\rho \langle \vec{u} \rangle \langle \vec{u} \rangle) = -\nabla p + \nabla \cdot \bar{\tau} + \rho \vec{g} + \vec{S}_m, \quad (2.11)$$

Standard equations 2.10-2.11 in ANSYS Fluent are 2D, but now in our implementation  $\vec{u}$  is depth-averaged. In depth-averaged simulations, the source terms  $S_{mass}$  and  $S_m$  refer to the depth of the channel. Thus, it is possible to perform flow simulations using a 2D mesh which saves computing resources.

For turbulent flow simulations in ANSYS Fluent, the turbulence kinetic energy  $k$  and its dissipation rate  $\epsilon$  are obtained from the following transport equations in a steady-state form:

$$\nabla \cdot (\rho \langle \vec{u} \rangle k) = \nabla \cdot \left[ \left( \mu + \frac{\mu_t}{\sigma_k} \right) \nabla k \right] + G_k + G_b - \rho \epsilon - Y_m + S_k, \quad (2.12)$$

$$\nabla \cdot (\rho \langle \vec{u} \rangle \epsilon) = \nabla \cdot \left[ \left( \mu + \frac{\mu_t}{\sigma_\epsilon} \right) \nabla \epsilon \right] + C_{1\epsilon} \frac{\epsilon}{k} (G_k + C_{3\epsilon} G_b) - C_{2\epsilon} \rho \frac{\epsilon^2}{k} + S_\epsilon. \quad (2.13)$$

In these equations the user-defined source terms are  $S_k$  and  $S_\epsilon$ .

Integrating Equation (2.1) along the depth, using the Leibniz rule:

$$\frac{\partial}{\partial l} \int_{a(y,l)}^{b(y,l)} f(x, y, l) dx = \int_{a(y,l)}^{b(y,l)} \frac{\partial f}{\partial l} dx - f(a, y, l) \frac{\partial a}{\partial l} + f(b, y, l) \frac{\partial b}{\partial l} \quad (2.14)$$

and considering the boundary conditions (2.8)-(2.9), the continuity equation becomes

$$\left( \frac{\partial}{\partial x} (D \langle u_1 \rangle) + \frac{\partial}{\partial y} (D \langle u_2 \rangle) \right) = 0. \quad (2.15)$$

According to this equation, mass source can be written as

$$\rho \left( \frac{\partial \langle u_1 \rangle}{\partial x} + \frac{\partial \langle u_2 \rangle}{\partial y} \right) = - \underbrace{\frac{\rho}{D} \left( \frac{\partial D}{\partial x} \langle u_1 \rangle + \frac{\partial D}{\partial y} \langle u_2 \rangle \right)}_{=: S_{mass}}. \quad (2.16)$$

The conservation law of momentum is derived with the control element that is shown in Figure 2.1. Horizontal differential lengths of the element are constants ( $dx$  and  $dy$ ). Then, all four edges are different in their heights and are represented by  $D$ ,  $D_2$ ,  $D_3$  and  $D_4$ . Using Taylor's formula these heights are:

$$D = D \quad (2.17)$$

$$D_2 = D + \frac{\partial D}{\partial x} dx \quad (2.18)$$

$$D_3 = D + \frac{\partial D}{\partial y} dy \quad (2.19)$$

$$D_4 = D + \frac{\partial D}{\partial x} dx + \frac{\partial D}{\partial y} dy \quad (2.20)$$

Convective acceleration in Equation 2.2 is derived with momentum flux. The momentum flux to the element through the surface  $S_{in}^x$  in the  $x$  direction is (momentum·velocity·area)

$$\rho \langle u_1 \rangle \langle u_1 \rangle \left( \frac{D + D_3}{2} \right) dy \quad (2.21)$$

and with Taylor's formula the momentum flux from the element through the surface  $S_{out}^x$  is

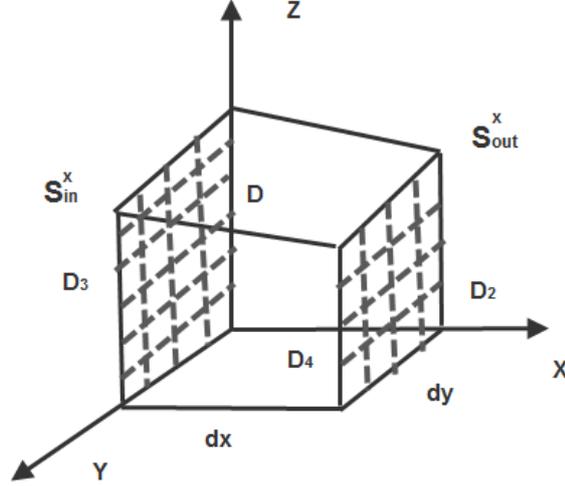


Figure 2.1: Control element with changing height.

$$(\rho \langle u_1 \rangle \langle u_1 \rangle + \frac{\partial(\rho \langle u_1 \rangle \langle u_1 \rangle)}{\partial x} dx) \left( \frac{D_2 + D_4}{2} \right) dy. \quad (2.22)$$

The change in the momentum flux in the  $x$  direction after neglecting the second order terms due to their smallness is

$$-\rho \langle u_1 \rangle^2 \frac{\partial D}{\partial x} dx dy - \frac{\partial(\rho \langle u_1 \rangle^2)}{\partial x} dx dy D \quad (2.23)$$

and the change in the  $x$  direction momentum flux due to the mass flow in the  $y$  direction is

$$-\rho \langle u_1 \rangle \langle u_2 \rangle \frac{\partial D}{\partial y} dy dx - \frac{\partial(\rho \langle u_1 \rangle \langle u_2 \rangle)}{\partial y} dy dx D. \quad (2.24)$$

Thus, the depth-averaged convective acceleration term in the  $x$  direction can be written as

$$-\nabla \cdot (\rho \langle \vec{u} \rangle \langle u_1 \rangle) + \langle u_1 \rangle S_{mass} \quad (2.25)$$

and the depth-averaged convective acceleration term in the  $y$  direction can be written as

$$-\nabla \cdot (\rho \langle \vec{u} \rangle \langle u_2 \rangle) + \langle u_2 \rangle S_{mass}. \quad (2.26)$$

Equations 2.25 and 2.26 lead to the following depth-averaged convective acceleration term

$$-\nabla \cdot (\rho \langle \vec{u} \rangle \langle \vec{u} \rangle) + \langle \vec{u} \rangle S_{mass}. \quad (2.27)$$

The stress tensor in Equation 2.11 is defined as

$$\bar{\tau} = [\mu(\nabla \cdot \langle \vec{u} \rangle + \nabla \cdot \langle \vec{u} \rangle^T) - \frac{2}{3} \nabla \cdot \langle \vec{u} \rangle]. \quad (2.28)$$

Other directional terms in Equation 2.28 are handled with the help of simulated depth-averaged velocities, except  $\tau_{xz}$  and  $\tau_{yz}$  which means the  $z$  direction. Depth-averaging of the  $z$ -directional derivatives of the stress tensor in the 3D momentum equation leads to the following expression:

$$\frac{1}{D} \int \frac{\partial \tau_{iz}}{\partial z} dz = \frac{1}{D} (\tau_{iz}(D) - \tau_{iz}(0)), i = 1, 2, \quad (2.29)$$

where  $\tau_{xz}$  and  $\tau_{yz}$  are the friction forces on the bottom and top surfaces.

Let us assume that the velocity profile in the vertical direction is known (Hämäläinen and Tiihonen (1995); Lyytikäinen (2009)):

$$u_1(x, y, z) = \langle u_1(x, y) \rangle S(z), \quad (2.30)$$

$$u_2(x, y, z) = \langle u_2(x, y) \rangle S(z), \quad (2.31)$$

$$S(z) = \frac{n+1}{n} \left( 1 - \left| 1 - \frac{1}{D} z^n \right| \right), \quad (2.32)$$

where  $n = 7$  for turbulent flow.

For example, Figure 2.2 presents profile  $S(z)$  if  $D = 1, n = 7$ .

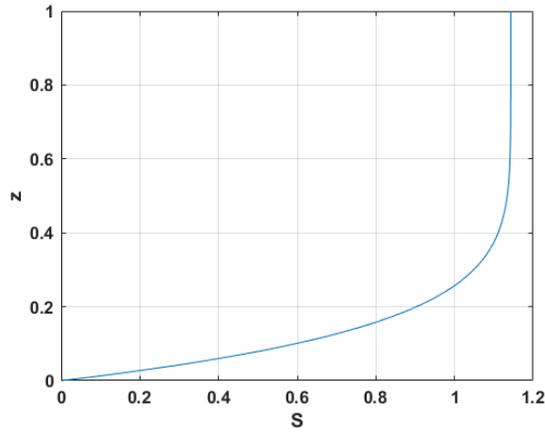


Figure 2.2: Profile  $S(z)$  when  $D = 1$  and  $n = 7$ .

Consequently,

$$\tau_{13} = \mu \frac{\partial u_1}{\partial z} + \mu \frac{\partial u_3}{\partial x} = \mu \frac{\partial u_1}{\partial z} = \mu \langle u_1 \rangle \frac{\partial S}{\partial z}, \quad (2.33)$$

$$\tau_{23} = \mu \frac{\partial u_2}{\partial z} + \mu \frac{\partial u_3}{\partial y} = \mu \frac{\partial u_2}{\partial z} = \mu \langle u_2 \rangle \frac{\partial S}{\partial z}, \quad (2.34)$$

The derivative of the velocity profile leads to

$$\frac{\partial S}{\partial z} = \frac{n+1}{D} \left(1 - \frac{z}{D}\right)^{n-1}, \quad (2.35)$$

when  $0 < z < D$ . Thus, the derivative of the velocity profile on the bottom (terrain level) is

$$\left. \frac{\partial S}{\partial z} \right|_{z=0} = \frac{n+1}{D} \quad (2.36)$$

and on the top  $z = D$  is

$$\left. \frac{\partial S}{\partial z} \right|_{z=D} = 0. \quad (2.37)$$

Hence,

$$\frac{1}{D} (\tau_{iz}(D) - \tau_{iz}(0)) = -\frac{\mu \langle u_i \rangle (n+1)}{D^2}, \quad i = 1, 2. \quad (2.38)$$

When gravity is insignificant, the pressure can be approximated as a constant in the  $z$  direction. It leads to a very simple depth-averaging of the pressure gradient so that we get

$$\frac{1}{D} \int \nabla p dz = \frac{d_2 - d_1}{D} \nabla p = \nabla p. \quad (2.39)$$

Now the steady-state 2D DA conservation law of momentum can be written as

$$\rho \nabla \cdot (\overline{\langle u \rangle \langle u \rangle}) = -\nabla p + \nabla \cdot \bar{\tau} + \underbrace{\langle u \rangle S_{mass}}_{=: S_m} - \frac{8\mu \langle u \rangle}{D^2} \quad (2.40)$$

where  $S_m$  is the momentum source term.

Since the transport equations in the  $k - \varepsilon$  model cannot be integrated over the depth, Rastogi and Rodi (1978) and Rodi (1984) analytically developed the  $k - \varepsilon$  model for applying it in the depth-averaged model using the following transport equations for open-channel flow:

$$\langle u_1 \rangle \frac{\partial k}{\partial x} + \langle u_2 \rangle \frac{\partial k}{\partial y} = \frac{\partial}{\partial x} \left( \frac{\nu_t}{\sigma_k} \frac{\partial k}{\partial x} \right) + \frac{\partial}{\partial y} \left( \frac{\nu_t}{\sigma_k} \frac{\partial k}{\partial y} \right) + P_h + P_{k\nu} - \epsilon, \quad (2.41)$$

$$\langle u_1 \rangle \frac{\partial \epsilon}{\partial x} + \langle u_2 \rangle \frac{\partial \epsilon}{\partial y} = \frac{\partial}{\partial x} \left( \frac{\nu_t}{\sigma_\epsilon} \frac{\partial \epsilon}{\partial x} \right) + \frac{\partial}{\partial y} \left( \frac{\nu_t}{\sigma_\epsilon} \frac{\partial \epsilon}{\partial y} \right) + c_{1\epsilon} \frac{\epsilon}{k} P_h + P_{\epsilon\nu} - c_{2\epsilon} \frac{\epsilon^2}{k}, \quad (2.42)$$

where

$$P_h = \nu_t \left[ 2 \left( \frac{\partial \langle u_1 \rangle}{\partial x} \right)^2 + 2 \left( \frac{\partial \langle u_2 \rangle}{\partial y} \right)^2 + \left( \frac{\partial \langle u_1 \rangle}{\partial y} + \frac{\partial \langle u_2 \rangle}{\partial x} \right)^2 \right] \quad (2.43)$$

is the production of  $k$  due to the interactions of turbulent stresses with the horizontal mean-velocity gradient. In the calculations,  $k, \varepsilon, \nu_t$  are not exactly depth-average values (they are calculated from the above equations), but Equations (2.41)-(2.42) can still be taken into consideration as depth-averaged forms of the 3D equations. At the same time, all terms occurring from the non-uniformity of vertical profiles are supposed to be involved in the source terms  $P_{kv}$  and  $P_{\varepsilon\nu}$ . Rastogi and Rodi (Rastogi and Rodi (1978)) associated the additional source terms with the friction velocity  $U_*$  by suggesting

$$P_{kv} = c_k U_*^3 / D \quad (2.44)$$

$$P_{\varepsilon\nu} = c_\varepsilon U_*^4 / D^2 \quad (2.45)$$

and they determined  $U_*$  from the usual quadratic friction law

$$U_* = \sqrt{c_f (\langle u_1 \rangle^2 + \langle u_2 \rangle^2)}, \quad (2.46)$$

where  $c_f$  is a friction coefficient. The empirical constants  $c_k, c_\varepsilon$  can be written as

$$c_k = \frac{1}{\sqrt{c_f}}, \quad (2.47)$$

$$c_\varepsilon = 3.6 \frac{c_{2\varepsilon}}{c_f^{3/4}} \sqrt{c_\mu}, \quad (2.48)$$

where  $c_\mu = 0.09, c_{1\varepsilon} = 1.44, c_{2\varepsilon} = 1.92, \sigma_k = 1, \sigma_\varepsilon = 1.3$ .

Now the source terms for 2D depth-averaged equations have been introduced. These sources are used in the 2D simulations of flow behaviour through the wind farm.

## 2.2 The AD Model

The AD model is the ANSYS Fluent implementation of AD theory, also referred to as linear momentum theory (Adamski (2013); Mozafari (2010)). In this model the aerodynamic effect of rotating blades is represented by a pressure discontinuity over an infinitely thin disk with an area equal to the swept area of the rotor. The thin disk is modelled as a porous media in the AD model that involves a pressure difference, but not a velocity difference. That is, the pressure in front of the disk is greater than behind it, but the velocity evolves continuously across the disk. As shown in Figure 2.3, the flow going through the disk is represented by a streamtube.

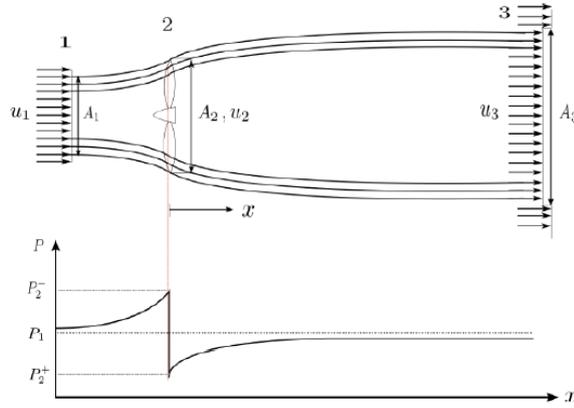


Figure 2.3: A schematic of the flow going over turbine blades modelled as an AD (Adamski (2013)).

The relation between velocity components at each section of the streamtube (i.e. the inlet, location of the turbine and outlet) and the power extracted by the turbine from the flow is derived from control volume analysis of the equations of the conservation of mass, momentum and energy. A combination of these equations with a model of flow in a porous medium is used in the implementation of the AD model for simulation of the flow around a horizontal axis turbine.

### 2.2.1 The AD model for 3D simulation

So as to derive the governing equations for the physics of the flow through the turbine and to also derive the relations between different variables in the flow field, one-dimensional, steady-state, incompressible flow through the turbine is assumed. A control volume is considered to consist of a streamtube around the turbine, starting upstream of the device and ending a large distance from the device, where the velocity is supposed to be uniform and unidirectional in the streamwise direction. Figure 2.3 shows a schematic of these assumptions. Three important locations along the streamlines are considered in the analysis. Section 1 is the free stream condition upstream of the turbine, Section 2 is located at the turbine plane and Section 3 is the outlet, far downstream of the turbine, where the pressure recovers to an ambient value. Based on these assumptions, and performing a control volume analysis, we can derive the governing equation and final relationship. The equation for mass conservation law in a control volume is given by

$$\frac{\partial}{\partial t} \int_{CV} \rho dV + \int_{CS} \rho \vec{u} \cdot \vec{n} dA = 0, \quad (2.49)$$

where  $CS$  stand for the control surfaces (boundaries),  $CV$  is the control volume,  $A$  is

the area,  $V$  is the volume and  $\vec{n}$  is the normal vector. Based on the steady-state flow assumption, the first term on the left-hand side of Equation (2.49) will be equal to zero. Since the boundaries of the control volume are formed by the streamtube and no mass can enter or leave these boundaries, the final form of Equation (2.49) will be

$$u_1 A_1 = u_2 A_2 = u_3 A_3. \quad (2.50)$$

Similarly, the general form of the momentum equation for a control volume will be

$$\frac{\partial}{\partial t} \int_{CV} \rho \vec{u} dV + \int_{CS} \rho \vec{u} (\vec{u} \cdot \vec{n}) dA = \int_{CS} -p \vec{n} dA + \int_{CS} \vec{\tau} \cdot \vec{n} dA + \int_{CV} \rho \vec{g} dV. \quad (2.51)$$

Analogous to the applied assumptions for the equation for the conservation of mass, steady flow and no flow across the streamtube boundaries, Equation (2.51) reduces to

$$\rho u_3^2 A_3 - \rho u_1^2 A_1 = F_A, \quad (2.52)$$

where  $F_A$  is the force that the turbine exerts on the flow.

Having a steady-state and incompressible flow and taking into account the fact that the flow is along a streamline and there are no frictional losses, Bernoulli's equation can be used upstream and downstream of the AD (Mozafari (2010)):

$$p_i + \frac{1}{2} \rho u_i^2 = \text{const}. \quad (2.53)$$

The above equation can be applied along a streamline between two points. Firstly, let us apply it from the inlet to a point just before the AD and also between a point just after the AD and the outlet:

$$p_1 + \frac{1}{2} \rho u_1^2 = p_2^+ + \frac{1}{2} \rho u_2^2. \quad (2.54)$$

$$p_2^- + \frac{1}{2} \rho u_2^2 = p_3 + \frac{1}{2} \rho u_3^2. \quad (2.55)$$

Subtracting Equation (2.54) from Equation (2.55) and multiplying both sides by  $A_2$  gives

$$\frac{1}{2} \rho A_2 u_1^2 \left[ 1 - \left( \frac{u_3}{u_1} \right)^2 \right] = A_2 (p_2^+ - p_2^-) = F_A. \quad (2.56)$$

Comparing Equation (2.56) to Equations (2.50)-(2.52), the next equation can be obtained:

$$u_2 = \frac{1}{2} (u_1 + u_3). \quad (2.57)$$

Equation (2.57) shows that fluid velocity at the location of the turbine will be less than the free stream velocity. This is due to the power extraction by the device from the flow

because of the pressure difference between the front and the back of the turbine. The ratio of this velocity reduction to the free stream velocity is the new variable called the axial induction factor, which is defined as

$$a = \frac{u_1 - u_2}{u_1}. \quad (2.58)$$

Using the above Equations (2.57) - (2.58), the relation between velocities at different cross sections of control volume as a function of axial induction factor will be

$$\frac{u_2}{u_1} = 1 - a, \quad (2.59)$$

$$\frac{u_3}{u_1} = 1 - 2a. \quad (2.60)$$

Therefore, the final expression for the power extracted from the turbine and the efficiency of the device will have the form of

$$P = F_A u_2 = \frac{1}{2} \rho u_1^3 A_2 [4a(1 - a)^2], \quad (2.61)$$

$$C_P = 4a(1 - a)^2, \quad (2.62)$$

where  $C_P$  is a power coefficient of the turbine. At this stage, the derived governing equations and relations between various velocity components along the streamtube and the efficiency of the turbine in the flow is combined with the implementation of a porous medium in ANSYS Fluent. It can be obtained by equating the pressure difference across the device based on the AD model and the modelled pressure difference across the porous media in ANSYS Fluent. This will provide the opportunity to model a horizontal axis turbine with the AD model.

Porous media are modelled in ANSYS Fluent as a momentum source term, which is added to the standard Navier-Stokes equations. The source term consists of a viscous loss term (the first term on the right-hand side of Equation (2.63)) and an inertial loss term (the second term on the right-hand side of Equation (2.63)). For simple homogeneous porous media, the source terms can be written as:

$$S_i = - \left( \frac{\mu}{\alpha} u_i + \frac{1}{2} C_2 \rho |u| u_i \right), \quad (2.63)$$

where  $\mu$  is fluid viscosity,  $\alpha$  represents the permeability,  $C_2$  is the inertial resistance factor and  $u$  represents the velocity.

The pressure drop within the porous region is then

$$\Delta p = -S_i \Delta m \quad (2.64)$$

where  $\Delta m$  is the thickness of the medium (the thickness of the turbine in this work). In this equation,  $\alpha$  and  $C_2$  are the only unknown coefficients that need to be evaluated in

order to estimate the pressure drop across the porous media.

According to Equation (2.56), the pressure drop calculated by the AD model is

$$\Delta p = \frac{1}{2}\rho(u_1^2 - u_3^2). \quad (2.65)$$

The above equation can be rewritten by applying Equation (2.58) as

$$\Delta p = \frac{1}{2}\rho u_2^2 \frac{4a}{1-a}. \quad (2.66)$$

Finally, by combining Equation (2.64) and Equation (2.66), it can be seen that

$$\Delta p = \frac{1}{2}\rho u_2^2 \frac{4a}{1-a} = \left( \frac{\mu}{\alpha} u_2 + \frac{1}{2} C_2 \rho u_2^2 \right) \Delta m. \quad (2.67)$$

The required coefficients can be found by comparing the coefficients in front of the  $u_2$  terms and the  $u_2^2$  terms:

$$\frac{1}{\alpha} = 0, \quad (2.68)$$

$$C_2 = \frac{4a}{\Delta m(1-a)}. \quad (2.69)$$

In order to show the facilities of the AD model on real rotors, the calculation of the Nordtank NTK 500/41 wind turbine with LM 19.1 m blades was carried out. This turbine rotates at 27.1 rpm and has a diameter of 41 m. The hub height of turbine is 60 m. The turbine is operational between wind speeds of 4 and 25 m/s and maximally generates 600 kW. The power distribution, measured by Paulsen (1995) at Riso, and power coefficient  $C_P$  for this turbine are presented in Figure 2.4 (Mikkelsen (2003)).

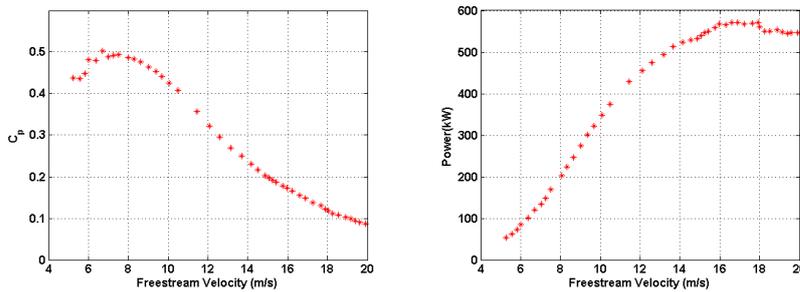


Figure 2.4:  $C_P$  (left) and power distribution (right) for the Nordtank NTK 500/41 wind turbine with LM 19.1m blades.

The most AD simulation methods use the thrust coefficient to get the axial induction factor. However, here the power coefficient was used. For example, suppose the free-

stream velocity is 10 m/s and the power coefficient is approximately equal to 0.4, as in Figure 2.4. The power coefficient is a measure of wind turbine efficiency that is often used by the wind power industry. A rough estimate of the device efficiency is needed to calculate the axial induction factor using Equation (2.62). It can be obtained from higher fidelity numerical models (i.e. the RRF model) or from experimental tests of the device. The values of  $a$  and  $u_1$  provide us with a range for  $u_2$  and  $\Delta P$  using Equations (2.59), (2.57) and (2.56) respectively. With these two ranges, the value for  $C_2$  is calculated using Equation (2.64). The thickness of the turbine can be taken as 1 m (Svenning (2010)), therefore  $C_2 = 0.6136$  according to the previous algorithm.

### 2.2.2 The AD model for depth-averaged numerical modelling

In this section, the new approach for wind turbine modelling in 2D simulation is presented. The total power of the wind farm is given by:

$$P = \sum_{i=1}^n P_i, \quad (2.70)$$

where  $n$  represents the number of the turbines and  $P_i$  is the power of each turbine:

$$P_i = \frac{\rho A C_P u^3}{2}, \quad (2.71)$$

where  $A$  is the disk area and  $u$  is the velocity. As seen from previous equation, all terms are known except for the velocity. This means that all other flow characteristics such as pressure, turbulence quantities, etc. are not needed for the wind farm optimization. Therefore, the goal of the 2D AD model is to obtain the same velocity from 2D modelling compared to the depth-averaged velocity in the 3D simulation. Two approaches were considered in the beginning. Firstly, it was decided to present the turbine as a rectangle where its length is the thickness of the turbine and the width is its diameter (Figure 2.5).

This method allows the use of the porous media model described in the previous section. The source term of the momentum equation is described by the Equation (2.63) for this method. However, the thickness of the turbine is small ( $\approx 0.5 - 3m$ ) compared with the size of the domain (a few kilometres). Thus, the mesh should be dense, which in turn causes an increase in computational time. Also it should be noted that the 2D mesh must be rebuilt when the position of the turbine changes. As a result, this method is not suitable here.

The second idea comes from the porous jump boundary conditions used in ANSYS Fluent. Porous jump conditions are applied to simulate a thin porous medium that has known velocity (pressure-drop) characteristics. It is essentially a 1D simplification of the full porous media model available for cell zones. Here, only one line (the diameter of the turbine) is represented as a turbine instead of a rectangle. The turbine is included in the governing equations as a source term, described by Equation (2.72) using user-defined functions (UDFs):

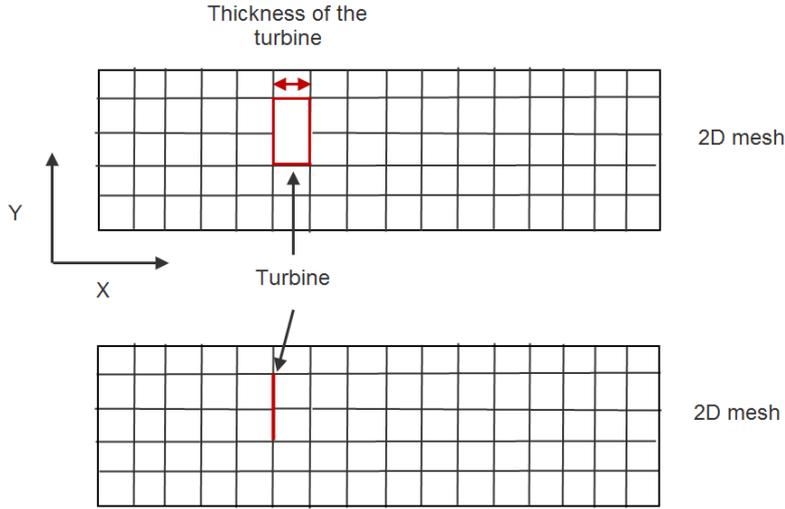


Figure 2.5: Two variants of turbine geometry in 2D simulation.

$$\Delta p = \left( \frac{\mu}{\alpha} u + \frac{1}{2} C_2 \rho u^2 \right) \Delta m, \quad (2.72)$$

where  $u$  is the velocity normal for the porous face,  $\Delta m$  is the thickness of the turbine.

As in the case with the depth-averaged equations, this approach allows keeping the initial mesh, which is very important for the optimization. The viscous loss term is also zero, but the inertial resistance factor for this model differs from the same coefficient for the 3D model, obtained in the previous section. It was numerically found for the Nordtank NTK 500/41 wind turbine with LM 19.1 m blades by comparing depth-averaged velocities from 3D and 2D simulations, and it was equal to  $C_2 = 0.11 \text{ 1/m}$ .

Finally, if the case consists only the complex terrain without turbines, Equations (2.16), (2.40), (2.41) and (2.42) are used. If the case with turbines on the flat terrain is considered, then only the source term from Equations (2.72) is added to the governing equations. When the turbines are located on the complex terrain, Equations (2.16), (2.40), (2.41), (2.42) and (2.72) should be solved.

## 2.3 Evolutionary algorithms

The basic idea of any optimization is that the best possible solutions or solution for a particular problem can be found. In this thesis, wind farm optimization is considered. Thus, optimization variables are continuous and the optimization algorithm is combined with a CFD model. Such optimization problems are called CFD-based or model-based optimization problems. Objective functions are calculated based on the model outputs

and this often makes the solution process very time-consuming.

Based on the number of objectives to be minimized or maximized, the optimization problems can be divided into single-objective or multi-objective optimization problems. The single-objective optimization problem can be defined as follows:

$$\min[f(\gamma)], \gamma \in \Omega, \quad (2.73)$$

where  $\gamma$  is the vector of optimization variables and  $\Omega$  denotes the set of constraints (linear or nonlinear constraints). In the wind farm optimization, the ultimate goal is to maximize the total power generation of the wind farm, which leads naturally to one-objective optimization.

There are many optimization methods and approaches for performing wind farm optimization, as described in Section 1. All these methods can be divided into gradient-based (derivative information of the objective function is used in the classical sense to find optimal solutions) and gradient-free (when information about the derivative of the objective function is unavailable, unreliable or impractical to obtain) approaches. In wind farm optimization, the objective function is not differentiable in a classical way because CFD simulations are used for its calculation. In gradient-based optimization the result depends on the starting point of the optimization process. This leads to the fact that these methods find the local optimum. Thus, in this work, evolutionary algorithms were used in a gradient-free approach. Of course, full 3D simulations cannot be used in optimization, but instead a fast modelling method based on depth-averaged equations was presented.

Evolutionary algorithms are optimization methods that imitate the evolution and selection of nature (Avramenko (2005); Eiben and Smith (2003)). The general scheme of this method is presented in Figure 2.6.

Each individual consists of optimization variables  $\gamma = (\gamma_1, \dots, \gamma_n)^T$ . At the beginning of the optimization, the initial population (a number of individuals) is determined randomly. Then, the objective function is calculated for the initial population. If the optimization criteria is not satisfied, the creation of new individuals begins. A new generation is created using selection, crossover and mutation (Spears (2000); Gujarathi and Babu (2016)). Selection means the choice of parents for the new offspring. In the crossover, new children (offspring) are generated using the genes of two selected parents. Crossover is divided into binary- or real-coded crossovers as presented in Figure 2.7.

In a binary-coded crossover each variable in a gene is of binary nature - it takes the value 0 or 1. Then, every offspring is created by mixing bits of two parent genes. In the real-coded crossover (used in this work), each variable is presented with a real number and the crossover is performed for one variable at time. Crossover can be done with three different methods: the one-point, two-point and uniform methods. A description of each method can be found in Magalhaes-Mendes (2013).

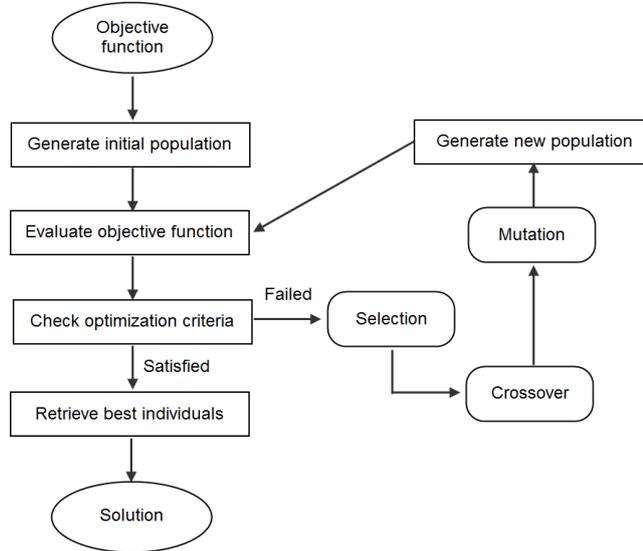


Figure 2.6: The structure of evolutionary algorithms.

Mutation comes after crossover and it changes the genes of the offspring in a random way. Thus, the genes in offspring differ from the initial population. Mutation is important for diversity creation in the population. However, a high mutation rate can make the solutions diverge from the optimum. After mutation, the objective function of the offspring is calculated. The new and better offspring replaces the parents and a new generation is produced. This cycle is repeated until the optimization criteria (for example, maximum number of generations) are obtained.

During the last few years, evolutionary algorithms have been widely used in various problem areas and have successfully found optimal solutions. There are a few main paradigms of evolutionary computation techniques: genetic algorithms, evolutionary strategies, evolutionary programming, etc. In this thesis, a detailed description of one such evolutionary algorithm, namely differential evolution (DE), is provided. This algorithm is a class of evolutionary programming developed by Storn and Price (1995) for optimization problems over continuous domains.

Let us consider the next optimization problem: find  $X = (x_1, \dots, x_B)$  to minimize  $f(X)$  subject to boundary constraints  $x_i^{(L)} \leq x_i \leq x_i^{(U)}$ ,  $i = 1, \dots, B$ , where  $B$  is the number of decision variables.

The DE algorithm works as follows (Price et al. (2011)):

1. Generate a random initial population (suitable solution for the problems).  $P_0 =$

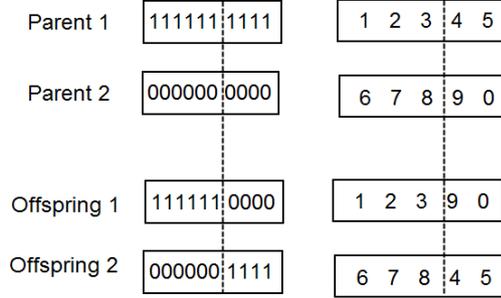


Figure 2.7: Binary-coded crossover (left) and real-coded crossover (right).

$x_{j,i,0} = rand_j[0, 1](x_j^{(U)} - x_j^{(L)}) + x_j^{(L)}$ ,  $i = 1, \dots, NP$ ,  $j = 1, \dots, B$ , where  $P_g$  is the population of generation  $g$ ,  $NP$  is the size of the population and  $rand[0, 1]$  is a uniformly generated random number between 0 and 1.

- Evaluate the objective function for the initial population.
- Mutation: DE generates a trial parameter vector for every individual by crossing a noisy vector with the target vector. A noisy vector is generated by randomly selecting three individuals  $r_1, r_2, r_3 \in 1, \dots, NP$ ,  $r_1 \neq r_2 \neq r_3$  and adding the weighted difference vector between the last two of them to the first one as follows:

$$v_{j,i,g} = x_{j,r_3,g} + F(x_{j,r_1,g} - x_{j,r_2,g}), \quad (2.74)$$

where  $i = 1, \dots, NP$ ,  $j = 1, \dots, B$  and  $F \in (0, 1+]$  is the scale factor. Individuals  $r_1, r_2, r_3$  can be the same as target vector  $i$ .

- Crossover: a trial vector is generated using crossover with the target vector so that:

$$u_{j,i,g} = \begin{cases} v_{j,i,g} & \text{if } rand_j[0, 1] \leq CR \vee j = k, \\ x_{j,i,g} & \text{otherwise} \end{cases} \quad (2.75)$$

where  $CR \in [0, 1]$  is the crossover probability,  $k \in 1, \dots, B$  is a random parameter index, chosen once for each  $i$ .

- Correcting boundary constraint violations: the created trial vector might violate boundary constraints, since DE is able to advance outside the original initialization range. Boundary constraint violations can be fixed in several ways, for example as follows:

$$u_{j,i,g} = \begin{cases} 2x_j^{(L)} - u_{j,i,g} & \text{if } u_{j,i,g} < x_j^{(L)} \\ 2x_j^{(U)} - u_{j,i,g} & \text{if } u_{j,i,g} > x_j^{(U)} \\ u_{j,i,g} & \text{otherwise} \end{cases} \quad (2.76)$$

6. Selection: the trial vector is selected for the next generation if it is equal to or better than the target vector:

$$X_{i,g+1} = \begin{cases} U_{i,g} & \text{if } f(U_{i,g}) \leq f(X_{i,j}) \\ X_{i,g} & \text{otherwise} \end{cases} \quad (2.77)$$

7. Return the best solution in the final population if the final condition is satisfied. Otherwise, use the new generation for a further run of the algorithm and return to Step 3.

The goal in optimization is to find the optimal parameter values so that the objective function is minimized. In this thesis the objective function is only the total power of the wind farm. Only one generated geometry and one mesh is needed in the depth-averaged model. Otherwise, geometry and mesh generation would be performed for every individual and that would slow down the whole optimization process significantly. After that, the initial population is generated and every individual is evaluated using 2D CFD simulation. The new population is produced using Steps 3-6 in the previous algorithm. CFD simulations are launched via an optimization program in Matlab. Firstly, the program defines the parameter values for one individual. Secondly, Matlab generates a Fluent journal file and UDFs for the source terms of the governing equations. Then, ANSYS Fluent is called on to calculate the depth-averaged velocity of each turbine in the wind farm. Finally, Matlab uses these values to calculate the objective function of each individual and restores the optimization algorithm.



### 3 Validation and CFD results

In this section, full 3D CFD modelling is validated by comparing the results to measured quantities. Two different cases were considered: flow simulation over a hill without the turbine, and a numerical modelling of the wind turbine on smooth terrain. Both of these cases were compared with laboratory experiments. Turbulence, mean velocity and concentration profiles were investigated in the RUSHIL wind tunnel experiment over symmetric two-dimensional hills with different aspect ratios (Khurshudyan et al. (1980)). A porous disc, produced of metallic mesh, that generates the same velocity deficit as the wind turbine and the three-blade rotating wind turbine was studied using wind tunnel tests in the PRISME laboratory (Aubrun et al. (2013)). The overall accuracy of the model is essential, but detailed point-wise pressures, velocities or turbulence quantities are not the main focus of this work. The velocity profiles and the depth-averaged velocity were used in the validation of the 2D DA and 3D CFD modelling and experiments. The satisfied results of these two tests gave us confidence about obtaining a good solution in the wind turbine simulation on a real hill. Errors, the geometry details and the dependency on mesh density in CFD are also discussed in this section.

#### 3.1 The RUSHIL wind tunnel study

The Khurshudyan experiment is well known and it has been used by several authors (Castro and Apsley (1997); Nakayama and Nagai (2010); Sip and Ludek (2016); Sladek et al. (2009)). A theoretical concept and wind tunnel experiments concerning pollutant diffusion and flow behaviour over two-dimensional hills for different aspect ratios were described by Khurshudyan et al. (1980). Three hills were considered that have steep, medium and small slopes in this experiment. Measurements were made of turbulent and mean velocity profiles upwind, over and downwind of each of the hills. These experiments were chosen to evaluate the derived 2D governing equations and compare them with 3D simulations and wind tunnel measurements.

The shapes of single 2D hills are given by the following parametric equations:

$$x = \frac{1}{2}\xi \left[ 1 + \frac{a^2}{\xi^2 + m^2(a^2 - \xi^2)} \right], \quad (3.1)$$

$$z = \frac{1}{2}m\sqrt{a^2 - \xi^2} \left[ 1 - \frac{a^2}{\xi^2 + m^2(a^2 - \xi^2)} \right], \quad |\xi| \leq a, \quad (3.2)$$

$$m = \frac{H}{a} + \sqrt{\left(\frac{H}{a}\right)^2 + 1}. \quad (3.3)$$

Hill height is represented by  $H = 0.117$  m,  $a$  is the half-width of the hill and  $\xi$  represents an arbitrary parameter. The axis along the flow direction is  $x$  and  $z$  is directed straight up. The equations do not include the  $y$  variable because the surface is two-directional.

The hills, described by Equations (3.1)-(3.2), have smooth symmetric forms in the vertical direction and smoothly join a flat surface at the points  $x = \pm a$ . These equations describe not a single surface but a two-parameter family of surfaces, the parameters being aspect ratio  $n = a/H$  and height  $H$ .

The hills studied in the wind tunnel had the shapes described by Equations (3.1)-(3.2), and aspect ratios of  $n = 3$ ,  $n = 5$  and  $n = 8$  (maximum slope angles of  $26^\circ$ ,  $16^\circ$  and  $10^\circ$ , respectively); they were named Hill3, Hill5 and Hill8 (see Figure 3.1).

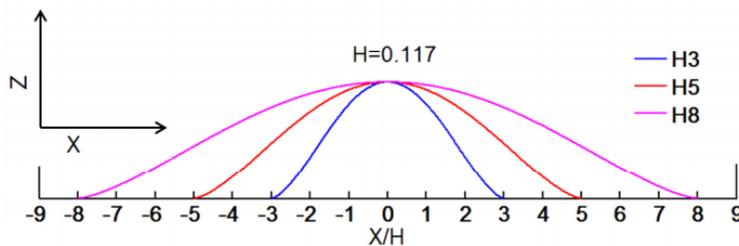


Figure 3.1: The shapes of Hill3, Hill5 and Hill8.

The flow over Hill3 separated on the lee slope and formed a recirculation zone called a cavity. Within the cavity, the reversed mean flow was found to have a mean speed of about 20% of the free-stream velocity. The flows over Hill5 and Hill8 did not separate. The speed-up of the flow over the tops and the slow-down on the lee sides were observed to be larger over the steeper hills in the experiments. Hill3 was considered in the current work in order to compare the results of 2D DA and 3D numerical modelling with the experiments.

Most of measurements were done with a free-stream velocity  $U_\infty = 4$  m/s, giving a Reynolds number  $Re = 3.1 \times 10^4$ . At the hill location, a depth of the boundary layer was about 1 m with a roughness length of  $z_0 = 0.157$  mm and a friction coefficient of  $u_*/U_\infty = 0.047$ . Numerical computations were done over Hill3 using the standard  $k - \epsilon$  turbulence model. The computational domain in 3D was extended by  $\pm 5$  m up- and downwind of the top of the hill by 1 m in a vertical direction and by 1 m in the y-direction.

A block-structured meshes were built in ANSYS IcemCFD. Mesh-dependence tests were investigated for a coarse mesh (4 million elements), a standard mesh (10 million elements) and a fine mesh (15 million elements). As a result, the unsuitability of the first two types of mesh was discovered since the reverse flow effect was not observed on these meshes. A fine hexahedral mesh was refined in the domain around the hill in the vertical direction according to a geometric progression so as to better resolve the flow around the separation point. The  $y^+$  value for the refined mesh equals 5.

The boundary conditions are shown in Figure 3.2.

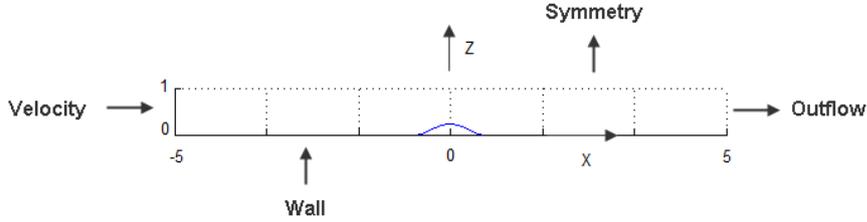


Figure 3.2: The boundary condition for the 3D simulation.

The initial velocity, in the inlet, has a logarithmic profile:

$$u = \frac{u_*}{\kappa} \ln \frac{z}{z_0}, \text{ if } z \leq H, \quad (3.4)$$

$$u = u_*, \text{ if } z > H, \quad (3.5)$$

where  $\kappa = 0.41$  is Kármán's constant. The turbulent kinetic energy  $k = 0.0012 \text{ m}^2/\text{s}^2$  and turbulent dissipation rate  $\epsilon = 9.6 \times 10^{-6} \text{ m}^2/\text{s}^3$  were obtained from the experiments. The periodic boundary conditions were applied on the borders in the y-direction, but they are not presented in Figure 3.2.

The computational domain for 2D DA simulation is 10 m long (the x-axis) by 1 m wide (the y-axis). The mesh consists of approximately 148 000 uniformly distributed quadratic elements. The initial velocity is calculated by integrating the velocity profile, presented by Equations (3.4)-(3.5), over the depth of the channel. It equals 3.8237 m/s. The height of channel  $D$  is the distance between the top and bottom levels of the channels:  $D = d_2(x_1, x_2) - d_1(x_1, x_2)$ . In this case  $d_2 = \text{const} = 1$ ,  $d_1 = 0$ , if  $x < -a$  and  $x > a$ , and  $d_1$  is described by parametric Equations (3.1)-(3.2) if  $-a \leq x \leq a$ , where  $a = 0.351$  for Hill3. To avoid difficulties in the gradient calculation, the smooth function  $f(x)$  was derived, which has the same shape as Hill3 (Figure 3.3):

$$f(x) = 74.72x^8 - 1.2x^7 - 13.39x^6 + 0.27x^5 + 6.57x^4 - 0.02x^3 - 1.70x^2 + 0.00026x + 0.117, \quad (3.6)$$

The geometry parameters are written into the source terms that are included in the governing equations via the UDFs of ANSYS Fluent. The computational domain and boundary conditions are presented in Figure 3.4.

Hereafter, boundary conditions for depth-averaged modelling are the same for all cases in this thesis: the initial velocity in the inlet, outflow in the outlet and the periodic boundary condition in the lateral surfaces. Only the length, the width of the computational domain and the value of the initial velocity differ from each other.

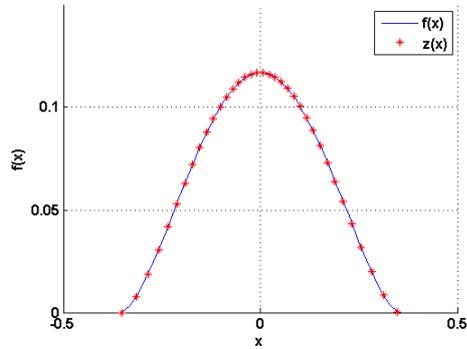


Figure 3.3: Hill3, described by parametric equations and function  $f(x)$ .

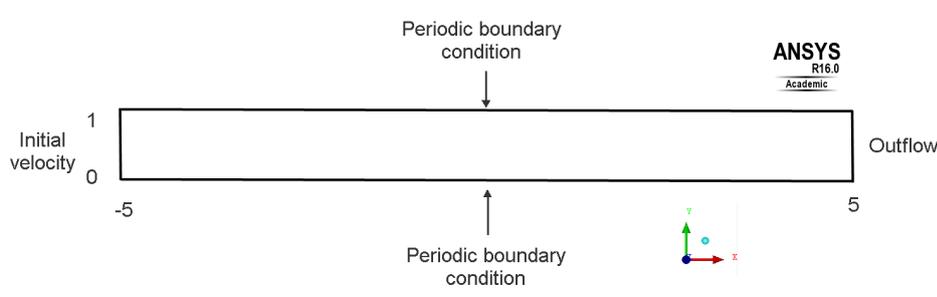


Figure 3.4: The boundary condition for the 2D DA simulation.

In the RUSHIL study, the flow separation on the lee slope was obtained for Hill3 and this was confirmed by 3D numerical simulation (Figure 3.5).

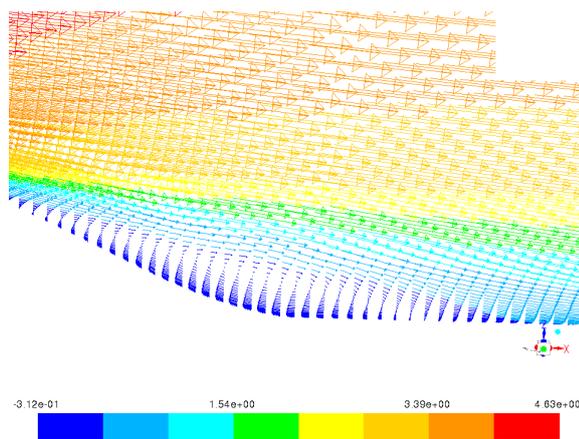


Figure 3.5: The velocity vectors of streamwise velocity for 3D flow modelling over the hill in a vertical plane perpendicular to the hill.

Figure 3.6 shows the mean velocity profiles (the x-component) computed in the 3D simulation at the upwind base, summit and downwind base.

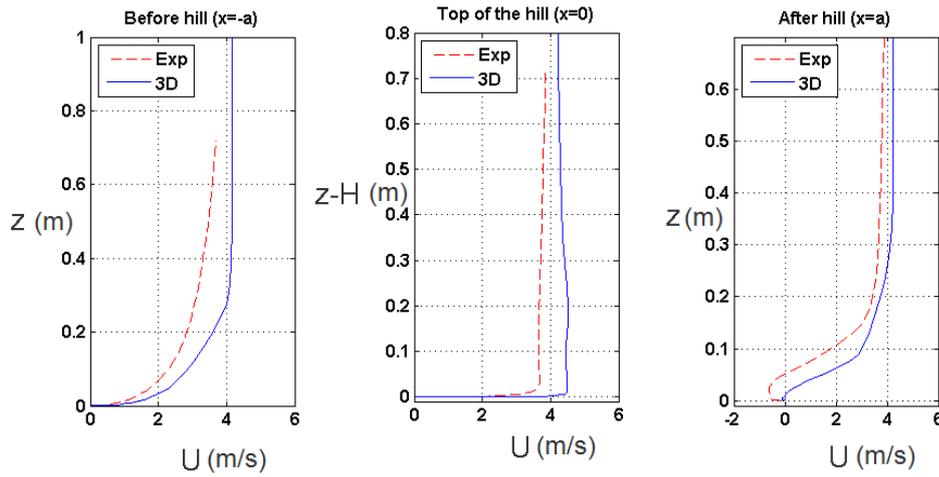


Figure 3.6: The RUSHIL experiment: a comparison of velocity profiles over Hill3.

In each case, the agreement with the experiment is good, particularly in respect of the maximum speed-up at the summit. This confirms the effectiveness of comparatively simple turbulence models (such as the  $k - \epsilon$  model) in predicting the mean flow in perturbed deep boundary layers.

The depth-averaged velocity values of the 3D simulations were calculated by Equation (2.7) for five different points of the hill: before, after and in the middle of the hill. Table 3.1 and Figure 3.7 present the results of the depth-averaged velocities obtained from the experiment and numerical modelling for these points. It can be seen that average velocity values are accurately predicted by using the depth-averaged approach.

Table 3.1: A comparison of depth-averaged velocities (m/s) from 2D DA and 3D simulations with the experiment.

x	2D	3D	Experiment
-0.351	3.82656	3.8359	3.2167
-0.175	4.11568	4.1249	3.7546
0	4.33	4.3403	3.923
0.175	4.11355	4.124	3.6573
0.351	3.82692	3.8365	3.3368

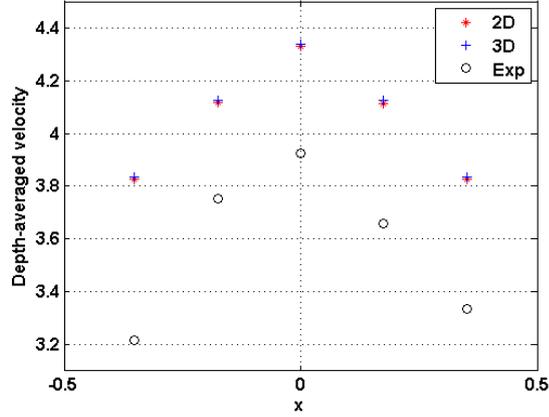


Figure 3.7: The comparison of depth-averaged velocities from the 2D DA and 3D simulations with the experiment.

The mean absolute percentage error can be calculated using the data from Table 3.1:

$$E = \frac{\sqrt{\sum_{i=1}^n | \langle v_i \rangle - \langle w_i \rangle |}}{\langle v_i \rangle} 100\%, \quad (3.7)$$

where  $\langle v_i \rangle$  and  $\langle w_i \rangle$  are depth-averaged velocities, calculated at  $i$ -point and obtained from 2D DA and 3D simulation or from the experiments;  $n = 5$  is the total number of points. The errors between

- the 3D simulation and the experiment is 13.50%,
- the 2D DA simulation and the experiment is 13.22%,
- the 2D DA and 3D simulation is 0.24%.

Figure 3.8 shows the turbulent kinetic energy profiles computed in the 3D simulation at the summit and downwind base.

Table 3.2 and Figure 3.9 present the results of the depth-averaged turbulent kinetic energy obtained from the experiment, 3D and 2D DA simulations.

The mean absolute percentage errors of turbulent kinetic energy between

- the 3D simulation and the experiment is 21.37%,
- the 2D DA simulation and the experiment is 21.81%,
- the 2D DA and 3D simulation is 4.05%.

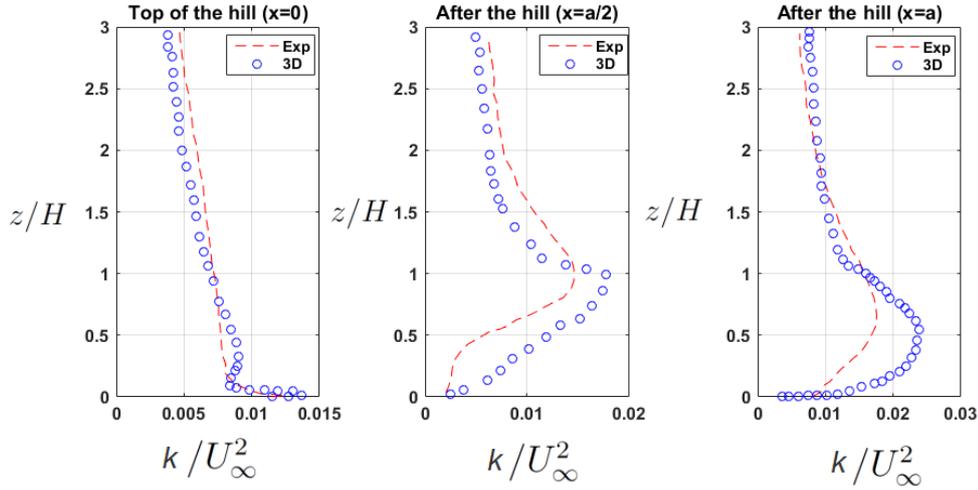


Figure 3.8: The RUSHIL experiment: turbulent kinetic energy profiles over Hill3.

Table 3.2: A comparison of depth-averaged turbulent kinetic energy ( $m^2/s^2$ ) from 2D DA and 3D simulations with the experiment.

x	2D	3D	Experiment
-0.351	0.0312	0.045	0.0663
-0.175	0.0408	0.0531	0.0785
0	0.0771	0.084	0.0967
0.175	0.0986	0.1078	0.1197
0.351	0.1246	0.1522	0.1276

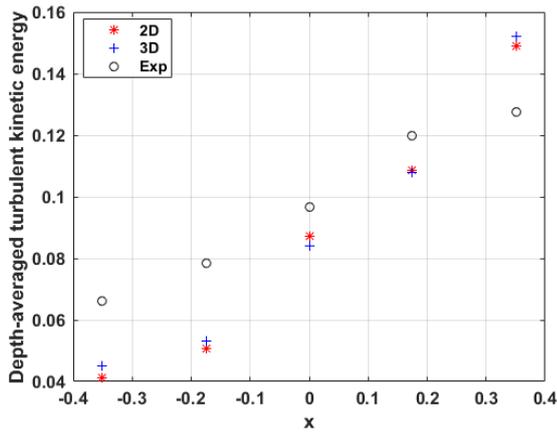


Figure 3.9: The comparison of depth-averaged turbulent kinetic energy from the 2D DA and 3D simulations with the experiment.

Overall, the solution shows a good level of agreement. The choice of the turbulence model is expected to have a very significant influence on the results, and more advanced turbulence models (such as the Reynolds Stress Model) might provide better agreement, even in the calculated turbulent kinetic energy.

It is easily understood that the results significantly depend on the height of the domain used in the 3D and 2D DA numerical modelling. The 3D computational domain usually extends by  $+4H \sim +8H$  m up, according to previous works on the modelling of the wind flow over complex terrain (Abdi and Bitsuamlak (2014); Blocken et al. (2015)). With these heights of the domain, the 3D models give good approximations of the total velocity deficits. The height dependence was also performed for the 2D DA simulation. The results for the distance of 2H, 5H, 10H and 15H above Hill3 in the vertical direction are presented in Table 3.3.

Table 3.3: Height dependence for the 2D DA simulation.

x	2H	5H	10H	15H
-0.351	3.8321	3.8279	3.8260	3.8252
-0.175	4.7951	4.2539	4.0468	3.9743
0	5.7296	4.5863	4.2052	4.0780
0.175	4.8000	4.2547	4.0472	3.9744
0.351	3.8463	3.8313	3.8272	3.8258

With the increasing depth of the channel, the depth-average velocity will converge in to four, as can be seen from Table 3.3. The simulation gives satisfied results; for example, the way in which the height of the domain above the hill ranges from 5H to 10H is acceptable. A comparison between the 2D DA and 3D simulation results of depth-averaged velocity verifies the usability of the 2D model in the simulation of the flow field over the complex terrain.

### 3.2 Wind tunnel tests in the PRISME laboratory

In this study, the laboratory measurements were made in order to compare the results of two models: a wind turbine with three blades, described in Pascheke and Hancock (2009), and a porous metallic disc, producing precisely the same velocity deficit as the wind turbine. In this thesis, the wind turbine is modelled as an AD. Therefore, the flow characteristics of a simplified model with a non-rotating disc are of interest here.

Measurements were done for a modelled decaying isotropic turbulent flow (turbulence intensity is equal to 4% at the location of the rotor disc) and an atmospheric boundary layer flow (at hub height, turbulence intensity is 13%) in order to evaluate the effect of the approach flow conditions. Only the second model was considered here.

All experiments were done in the closed circuit wind tunnel of the PRISME laboratory. The three blade wind turbine with a diameter (DM) of 416 mm and with the thrust coefficient  $c_T = 0.5$  was considered in the experiment. In accordance with the AD model, the thrust coefficient can also be presented as

$$C_T = 4a(1 - a). \quad (3.8)$$

This means that the axial induction factor is  $a = 0.1464$ . Unfortunately, no information about the thickness of the wind turbine and the porous disk was provided in the experiment. Therefore, it was decided to choose the thickness as  $\Delta m = 0.01$  m. According to this thickness, the inertial resistance factor for 3D AD model is  $C_2 = 68.6051$ . Likewise, as in the case with Nordtank NTK 500/41 wind turbine, the inertial resistance factor for the 2D AD model of  $C_2 = 17$  was found numerically.

A porous disc that was designed as a metal mesh was made in order to obtain the same speed deficit at  $x = 0.5DM$  downstream of the disk as the speed deficit after a wind turbine. The diameter of the disk was the same as the diameter of the wind turbine. A circle of  $0.2DM$  diameter at the centre of the porous disc with a density of 35% was created. The mesh has a density of 45% in the other areas of the disk. Since the mesh in the experiment is unstructured, the similar mesh was reconstructed for the 3D CFD simulation (Figure 3.10).

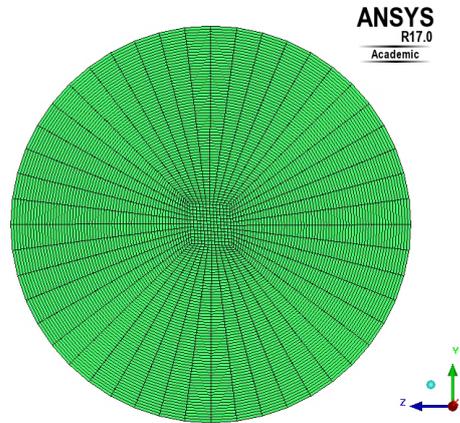


Figure 3.10: The porous disc that generates a similar space-averaged velocity deficit as the rotating wind turbine.

At the hub height  $HH = 300$  mm, the turbulence intensity is  $I = 13\%$  and the upstream mean velocity is  $u_0 = 2.5$  m/s. Therefore, the friction velocity is defined as:

$$u_* = \frac{\kappa u_0}{\log(HH/z_0)} = 0.1113 \text{ m/s}, \quad (3.9)$$

where  $\kappa = 0.41$  is Kármán's constant and  $z_0 = 0.03$  mm is the roughness length obtained

from the experiment. Then, the initial velocity profile can be presented as

$$u = \frac{u_*}{\kappa} \log\left(\frac{z}{z_0}\right). \quad (3.10)$$

The velocity profile was measured at an altitude of 900 mm above the ground because of experimental limitations. Figure 3.11 presents the initial vertical profile of mean streamwise velocity non-dimensionalized with the mean streamwise velocity at hub height obtained from the experiment and calculated by Equation (3.10).

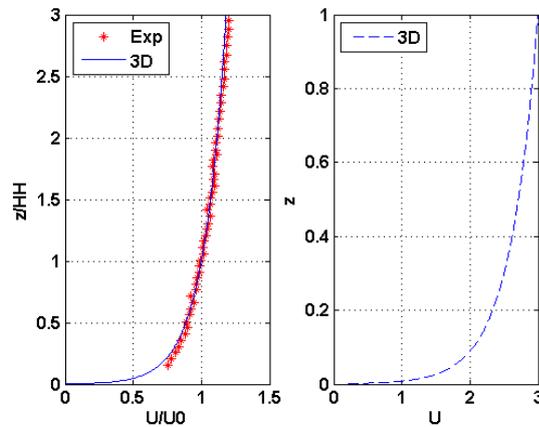


Figure 3.11: The vertical profile of mean streamwise velocity non-dimensionalized with the mean streamwise velocity at hub height (left) and the initial vertical profile of mean streamwise velocity (right).

The computational domain and the boundary condition in the 3D case are presented in Figure 3.12.

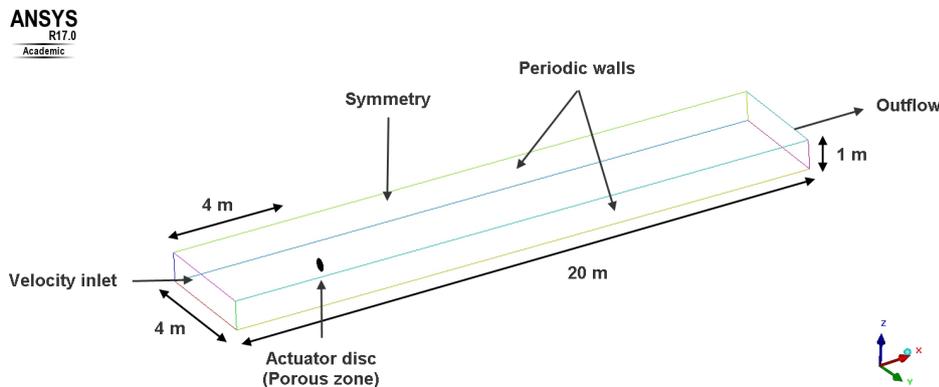


Figure 3.12: The computational domain and boundary condition for the 3D simulation.

ANSYS Icem was used to build a hexahedral mesh of approximately 19 million volumes. Around the AD, the grid was H-shaped and it was optimized to achieve sufficient resolution. Also, the mesh has been refined along the ground to model the boundary layer with the help of wall functions.

The turbine was modelled here as an AD using porous zone model. The permeability of the disk was set to a high value to cancel out viscous effects and only account for the effect of friction. The parameters applied in ANSYS Fluent for the AD model described in Section 2.4 are summarized below:

- Face permeability ( $\alpha$ ):  $1 \times 10^{10} \text{ m}^2$
- Inertial resistance factor ( $C_2$ ): 68.6051

It is relatively easy to see from the 3D case that the 2D computational domain is 20 m long (the x-axis) by 4 m wide (the y-axis). The 2D mesh consisted of approximately 63 000 quadratic elements. The mesh was graded in the AD region. In this 2D DA numerical modelling, Equation (2.72) is only added to the governing equations as a source term using UDFs.

Figures 3.13 and 3.14 show the mean streamwise velocity deficit profiles ( $(u_0 - u(y))/u_0$ ) versus the spanwise direction measured at hub height and turbulence intensity at  $x/DM = 0.5$  and  $x/DM = 3$  downstream of the porous disc.

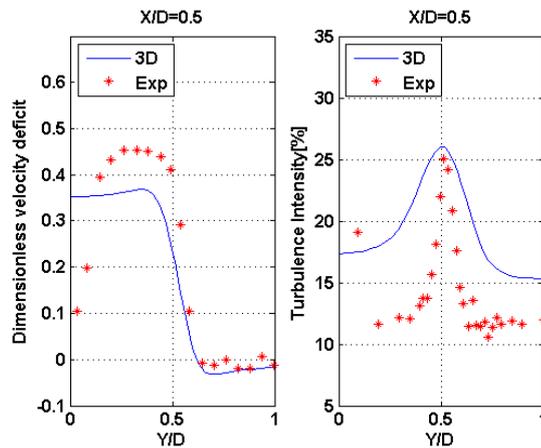


Figure 3.13: The mean streamwise velocity deficit (left) and turbulence intensity (right) downstream of the porous disc, measured at hub height at  $x/DM = 0.5$

Regarding the results at  $x/DM = 0.5$ , it is clear that they show some differences. These can be explained by the difference in the metal mesh from the experiment and the 3D mesh in the simulation. Also, the reason can be the turbine thickness, which was chosen

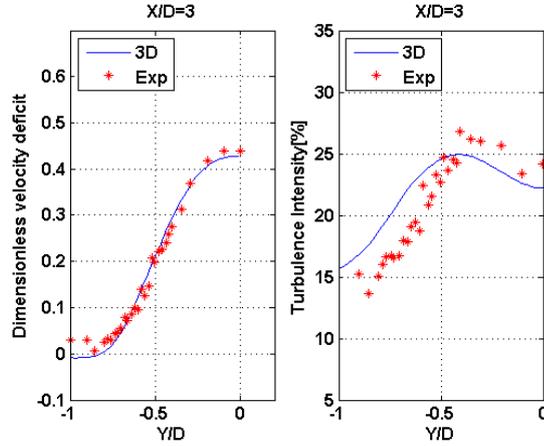


Figure 3.14: The mean streamwise velocity deficit (left) and turbulence intensity (right) downstream of the porous disc, measured at hub height at  $x/DM = 3$ .

based on intuition. The wake that naturally develop further downstream at  $x/DM = 3$  becomes almost similar and the turbulent intensity is approximately in accordance with the experiment results.

Figure 3.15 shows the profile of X velocity for the horizontal plane at the hub height for 3D simulation (top) and profile of depth-averaged velocity obtained from 2D DA simulation (bottom).

Table 3.4 shows a performance comparison between the values of the 2D and 3D depth-averaged velocity downstream of the porous disc, ranging from  $x = 1DM$  to  $x = 5DM$ .

Table 3.4: A performance comparison between the values of the 2D and 3D depth-averaged velocities (m/s).

x	3D	2D
1DM	2.2487	2.2682
2DM	2.2272	2.2436
3DM	2.2263	2.2348
4DM	2.2332	2.2313
5DM	2.2438	2.2398

As a result, depth-averaged numerical modelling predicts the value of the velocity very well and the absolute error is  $E = 0.5445\%$  using Equation (3.7).

### 3.3 The Lillgrund wind farm

The present chapter is devoted to 2DDA RANS computations on the Lillgrund wind farm. The main goal of this work is to document the quality of 2DDA RANS computations using the AD method for the wind farm. The Lillgrund wind farm is presented in Figure 3.16.

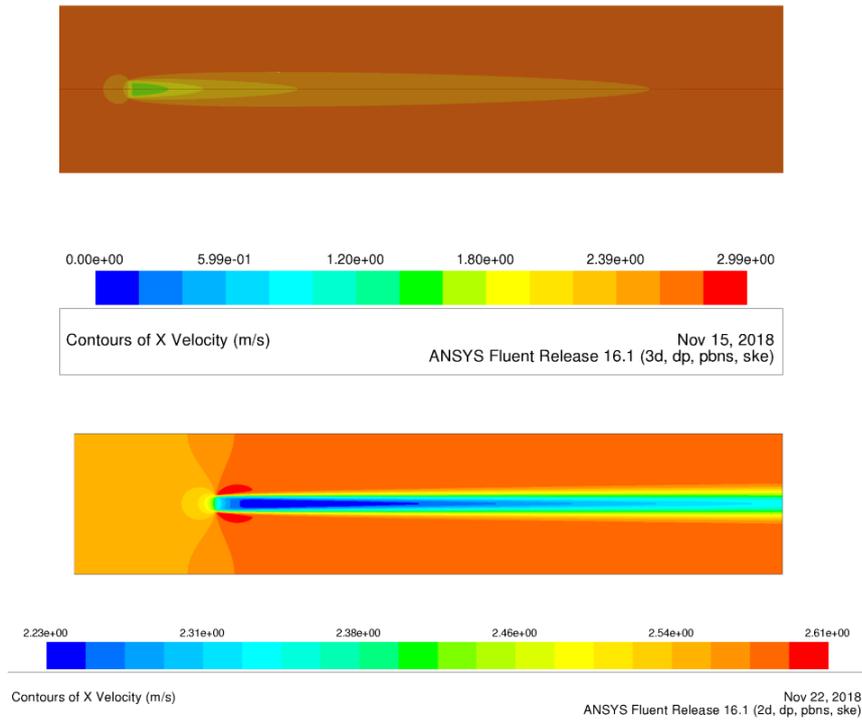


Figure 3.15: X velocity profile for the horizontal plane at the hub height for 3D simulation (top) and velocity profile obtained from 2D DA simulation (bottom).

The presented farm is very dense, because the internal distance between wind turbines for specific wind directions is as small as 6.6 turbine radius. As can be seen from Figure 3.16, two turbines are missing inside the Lillgrund wind farm in Row E due to the shallowness of the water in that particular spot. More information about this farm and its performance can be found in Dahlberg (2009)

The presented wind farm consists of 48 Siemens SWT93-2.3 MW turbines. There is no public geometry and airfoil data about this turbine except the radius, which is equal to 46.5 m. Therefore, a downscaled version of the conceptual National Renewable Energy Laboratory (NREL) 5 MW turbine is used (Jonkman et al. (2009)). The power coefficient of the NREL 5MW turbine is equal to 0.485 here and the thrust coefficient is 0.812. The radius of the turbine used in the simulations is 46.5 m. The inertial resistance factor for 2DDA simulation was found numerically and it is 0.121. More information about agreement between the turbines is presented in Nilsson et al. (2014).

For this analysis, only simulations of the 120 inflow angle are performed with the different turbulent intensity of 4%, 5.7% and 7.7% (see Figure 3.19), as it was in Nilsson et al.

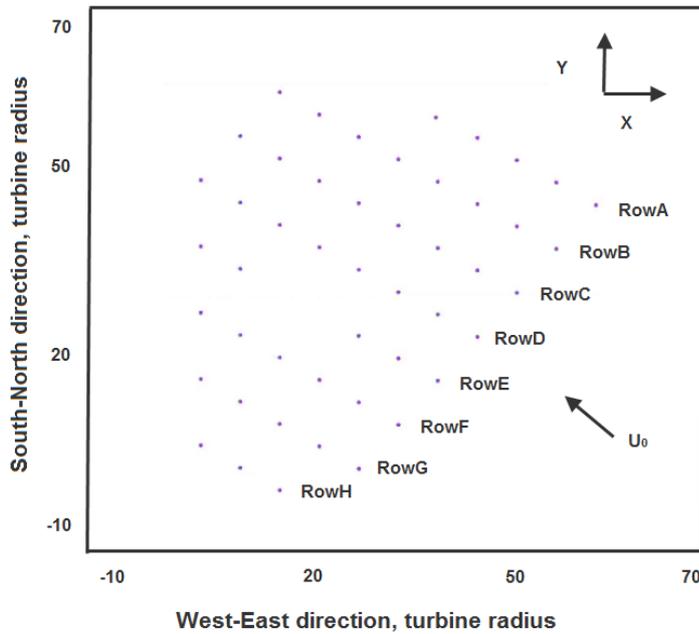


Figure 3.16: Layout of the Lillgrund wind farm (Nilsson et al. (2014)).

(2014). The initial velocity is equal to 8.6 m/s. The boundary conditions are periodic at the lateral boundaries. Outflow boundary conditions are used at the outlet and velocity inlet boundary conditions are set at the inlet.

Simulations are performed using three meshes, where one is coarse mesh (30 967 quad elements), the second one is the reference mesh (130 682 quad elements) and the last one is fine mesh (538 731 quad elements). The dependency of the power production, obtained from 2DDA simulation, in relation to grid resolution is analyzed. The power of each turbine in this research is calculated by Equation 2.71, where velocity is the depth-averaged velocity in the centre of the turbine, obtained from simulation. The relative power of the current turbine is the power of this turbine divided by the power of the first turbine in the row. The simulations are performed with the turbulence intensity equal to 5.7% (Figure 3.17).

As is possible to see from the previous figure, there are small differences between the different meshes. The relative power of the turbines decreases slightly due to increased mesh size and gets close to the values obtained from Nilsson et al. (2014). Because of the available computational resources, it was decided not to perform simulation with higher

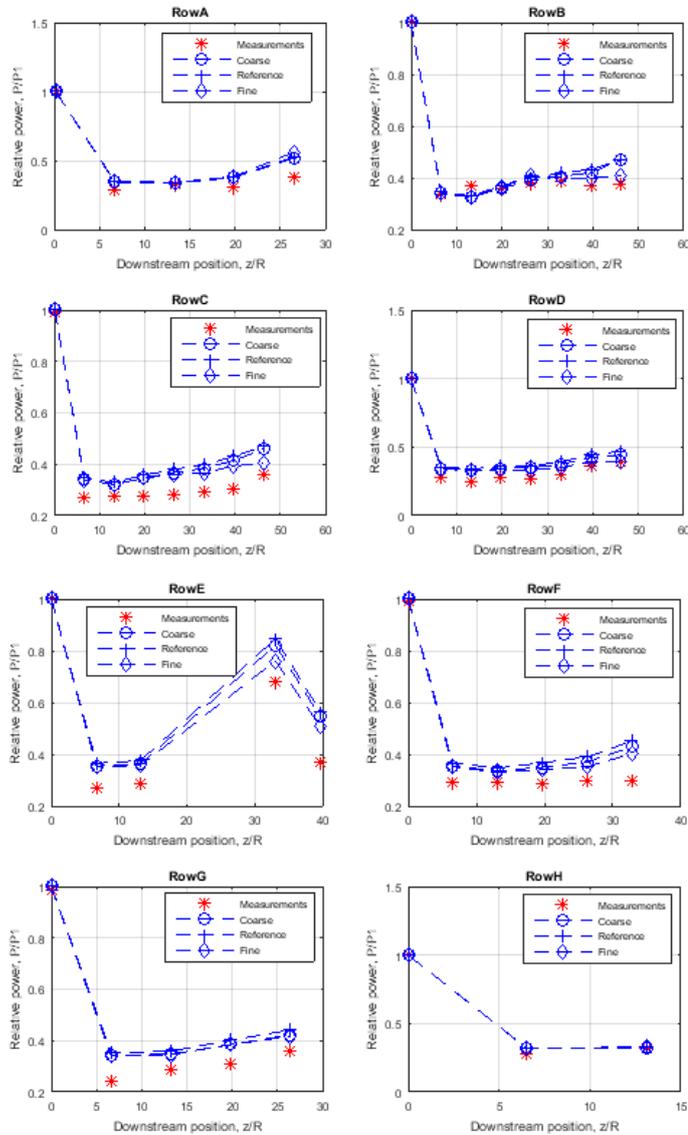


Figure 3.17: Effect on the relative power for the turbines in Row A-H by varying the mesh size.

resolution. The differences between the results when using the reference and the fine meshes are small. The reference mesh is used in the following analysis because of limitations on available computational time.

A contour plot of the mean streamwise velocity is presented in Figure 3.18. Figure 3.19 show the relative power for Row A-H in the wind farm for different turbulent intensity.

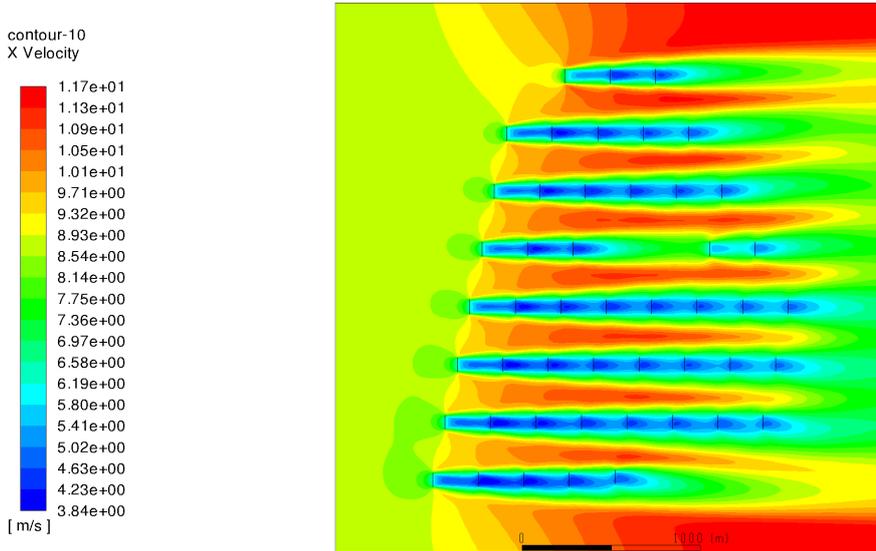


Figure 3.18: Mean streamwise velocity obtained from 2DDA simulation.

There is a small difference in the turbines in the second row, and there is also a certain error behind the recovery gap. However, in general, the agreement between data obtained from Nilsson et al. (2014) and the computed data is satisfied. The most important result is that this method predicts the wake behaviour very well. For the further optimization, it is more important to obtain the right behaviour of the power production in the wind farm than the concrete values of the power production.

As a result, the 2DDA method cannot give the detailed information about the wake behaviour, turbulence quantities, velocity profiles and so on. The model based on 2DDA must be calibrated with the experiment or 3D simulation to obtain the right values of necessary parameters. However, the obtained results and fast computational time give us the confidence to use this method in optimization algorithms.

### 3.4 The flow simulation of the wind turbine on the hill

The numerical simulation of the wind turbine on complex terrain is considered in this section. The results of the 3D simulation are compared with the 2D DA CFD predictions. Two different cases are presented:

1. A simple case: the terrain topography consists of one ideal shape with one turbine.
2. A generated terrain, that is much closer to Askervein Hill, with one turbine.

For both of these cases, the Nordtank NTK 500/41 wind turbine, presented and described in previous chapters, was used.

### 3.4.1 Simple case

In this case, simulations were used to compute the wind fields of the Nordtank NTK 500/41 wind turbine on the hill, as described by:

$$z = H \times \exp(-((x - 400)^2 + (y - 500)^2)/3000), \quad (3.11)$$

where  $H = 60$  m is the height of the hill. The 3D simulations are performed using a domain of  $550 \times 260 \times 200$  elements with dimensions of  $2000$  m  $\times$   $800$  m  $\times$   $400$  m (wind direction, cross-wind direction and vertical direction). The inlet velocity profile is given by

$$u = \frac{u_*}{\kappa} \ln \frac{z}{z_0}, \quad (3.12)$$

where  $\kappa$  is Kármán's constant,  $u_* = 0.654$  and  $z_0 = 0.03$ . The boundary conditions are summarized in Figure 3.20.

ANSYS Icem was used to build a hexahedral mesh of approximately 20 million volumes. Around the AD, the grid was H-shaped (Figure 3.21) and it was optimized to achieve sufficient resolution. Also, the mesh was refined along the hill to model the boundary layer with the help of wall functions.

Here, the computational domain divides into two regions: a region formed of the volume of the disc swept by the rotor blades, and a region formed of the rest of domain. Mesh generation experienced serious difficulties due to the different levels of geometric scales represented: the boundary layer thickness (0.5 m), the diameter of the turbine (41 m), the height of the hill (60 m) and the length of the domain (2000 m). Therefore, a modular procedure for the mesh generation was used to take care of the different levels of the local geometric scales.

The computational domain for the 2D DA simulation was 2 km long (the x-axis) by 0.8 km wide (the y-axis). The mesh was graded in the AD region and consisted of approximately 150 000 quadratic elements. The inlet velocity was 13.47723 m/s.

The vertical profiles of mean streamwise velocity through the centre of the turbine are presented in Figure 3.22.

Figure 3.23 shows the profile of X velocity for the horizontal plane at the hub height for 3D simulation (top) and profile of depth-averaged velocity obtained from 2D DA simulation (bottom).

Table 3.5 provides information about the values of the 2D and 3D depth-averaged velocity downstream of the porous disc at  $x = 2DM$  to  $x = 10DM$ .

Table 3.5: The 2D and 3D depth-averaged velocity (m/s) for the simple hill.

x	3D	2D
2DM	12.8419	12.7528
3DM	12.8635	12.8444
4DM	12.9547	13.0002
5DM	13.0136	13.0346
6DM	13.0510	13.0705
7DM	13.0771	13.1364
8DM	13.0963	13.1891
9DM	13.1113	13.2325

It is possible to see from Table 3.5 that the depth-averaged velocity is accurately predicted in 2D. More information about this simple hill can be found in the work of Avramenko et al. (2015). In this paper, two scenarios were considered. Firstly, numerical modelling of flow over a simple hill was examined. Secondly, simulations of the wind turbine on a simple hill were carried out. The results show that the depth-averaged equations predict the general flow behaviour with good accuracy.

### 3.5 Askervein Hill

Askervein, or Askernish Hill as it is sometimes referred to locally, is located along the west coast of South Uist island in Scotland. Askervein Hill has been widely investigated through experiments, field measurements and simulations by many researchers. This hill is relatively isolated. It has an approximately elliptic shape with minor and major axes of one and two kilometres respectively. The height of Askervein Hill is 126 m above sea level and 116 m above its base. Figure 3.24 shows the topographic contours of Askervein Hill. The lines A, AA and B are the main axes of the hill. Lines A and AA are perpendicular to line B. Line A is directed through the hill top located at  $x = -304.6875$  m,  $y = 281.25$  m. Line AA connects the centre point of the domain ( $x = 0$ ,  $y = 0$ ). Lines A and AA are aligned at an angle of  $223^\circ$  counterclockwise from the east. Line B passes through the hill top and the centre point of the domain and it is oriented at an angle of  $133^\circ$  anticlockwise from the east (Murali and Rajagopalan (2017)).

The test case consists of an analytically defined hill. The shape is defined by the following expression:

$$z = H e^{-\left(\frac{x^2}{2s_1^2} + \frac{y^2}{2s_2^2}\right)}, \quad (3.13)$$

where  $H = 116$  m,  $s_1 = 562$  and  $s_2 = 278$ .

As can be seen from Figure 3.25, the shapes of the generated hill and Askervein Hill are close to each other in terms of shape.

The horizontal and vertical extents of the domain are  $8000 \text{ m} \times 8000 \text{ m} \times 1200 \text{ m}$ . The centre point of the ground and the hilltop are located at  $x = 0, y = 0$ . The lines A and B are directed along the Y-axis and X-axis accordingly. The terrain is set as a no-slip wall boundary. The surface roughness height  $z_0 = 0.03 \text{ m}$  according to Walmsley and Taylor (1996). A symmetry boundary condition is set on the top surface of the domain. The north and south boundaries of the computational domain are set as the periodic boundaries. The outflow boundary condition is used in the outlet. The inlet velocity and turbulence profiles are same as those of Montavon (1998) and given by Equation (3.12), and

$$\epsilon = \frac{U_*^3}{hK\eta} e^{-\left(\frac{c}{1+c}+a\right)\frac{\eta^2}{2}}, \quad (3.14)$$

$$k = \sqrt{\frac{\kappa z}{C_\mu U_*}} \epsilon \left( \frac{U_*}{hK} \left[ \frac{c}{\eta(\eta+c)} - \left( \frac{c}{1+c} + a \right) \eta + a \right] \right)^{-\frac{1}{2}}, \quad (3.15)$$

where  $B = 7.91$ ,  $c = 0.183$ ,  $\eta = z/h$ ,  $h = 1090$  and  $a = 1/c$ .

A hexahedral mesh of approximately 25 million volumes was built for this case. The H-shaped grid was optimized to achieve sufficient resolution around the AD. Also, the mesh was refined along the hill to model the boundary layer with the help of wall functions.

The 2D simulations were performed using a domain of  $801 \times 801$  elements with dimensions of  $8000 \text{ m} \times 8000 \text{ m}$  (wind direction and cross-wind direction). The velocity in the inlet region was set as  $15.21468 \text{ m/s}$ .

Figure 3.25 presents the vertical profiles of mean streamwise velocity downstream of the turbine.

As the previous figure demonstrates, after the point has been reached where the maximum velocity deficit occurs, the wake begins to recover at a comparatively normal recovery rate. The rate of the wake recovery decreases considerably as we go further downstream.

Figure 3.27 shows the profile of X velocity for the horizontal plane at the hub height for 3D simulation (top) and profile of depth-averaged velocity obtained from 2D DA simulation (bottom).

The values of the 2D and 3D depth-averaged velocity downstream of the porous disc at  $x = 2DM$  to  $x = 20DM$  are compared in Table 3.6.

As with previous cases, the absolute error (Equation (3.7)) is small ( $E = 1.82\%$ ). As a result, the depth-averaged model is accurate for the velocity calculations, but one should be cautious about using the depth-averaged model for detailed predictions of turbulence components, pressure and other variables. However, it is possible to save computational

Table 3.6: The 2D and 3D depth-averaged velocity (m/s) for Askervein Hill.

x	3D	2D
3DM	15.4922	15.5914
6DM	15.4541	15.5282
8DM	15.4174	15.4717
10DM	15.3681	15.4073
12DM	15.3095	15.339
15DM	15.2196	15.2375
18DM	15.1370	15.1467
20DM	15.0891	15.0955

and human resources by replacing the full governing equations with the depth-averaged equations. A complex 3D computational domain is no longer needed. Instead, 2D equations with source terms are solved in a fixed and simple 2D domain. Also, this method is very fast from the point of view of computational time. The calculation time of one 2D numerical modelling takes only a few minutes and a 3D simulation takes two to three days. However, despite the fact that they lose some 3D flow characteristics, this method presents the flow behaviour with a very good accuracy. Further, this approach is combined with mathematical optimization to find the best positions of wind turbines in the wind farm.

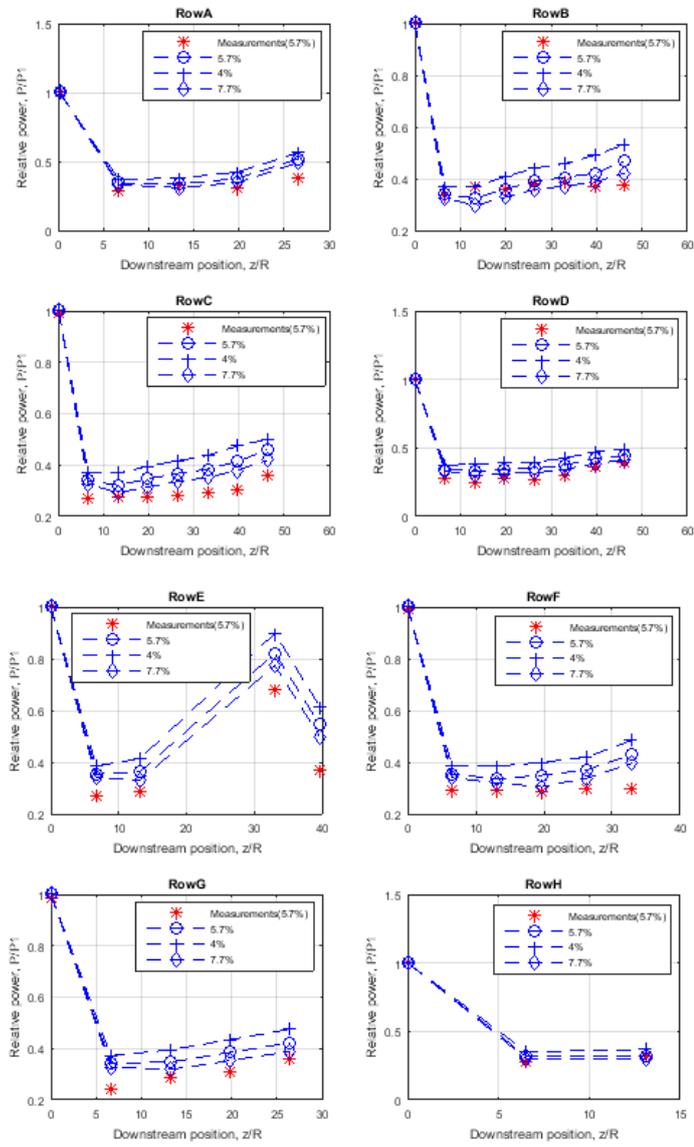


Figure 3.19: Relative power for Row A-H.

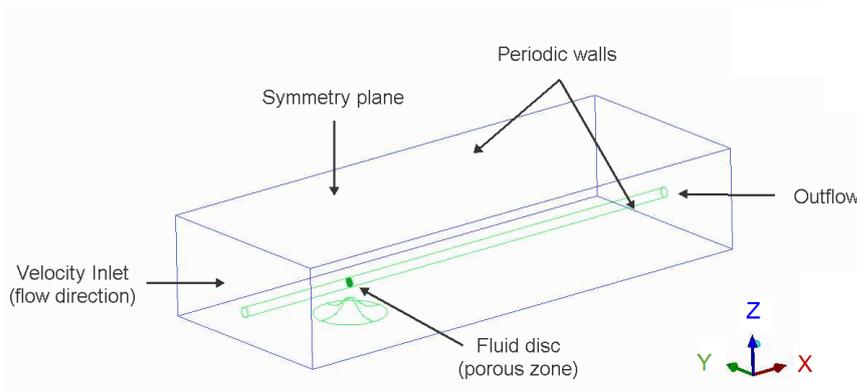


Figure 3.20: Boundary conditions.

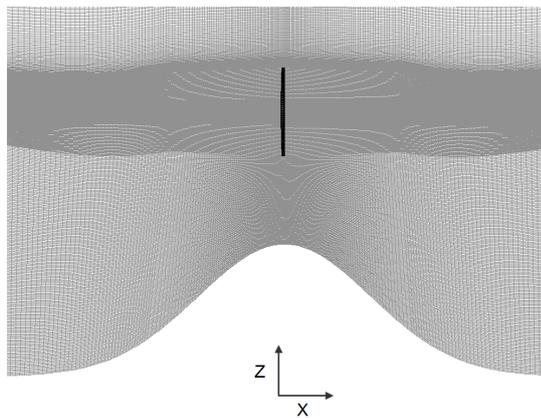
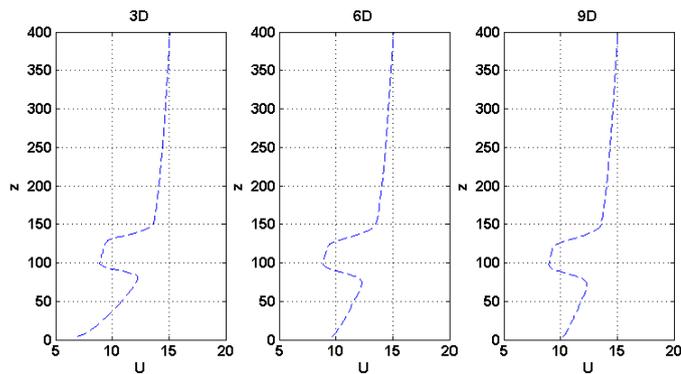


Figure 3.21: Mesh around the actuator disc.

Figure 3.22: Vertical profiles of mean streamwise velocity downstream of the porous disc at  $x = 3D$ ,  $x = 6D$  and  $x = 9D$ .

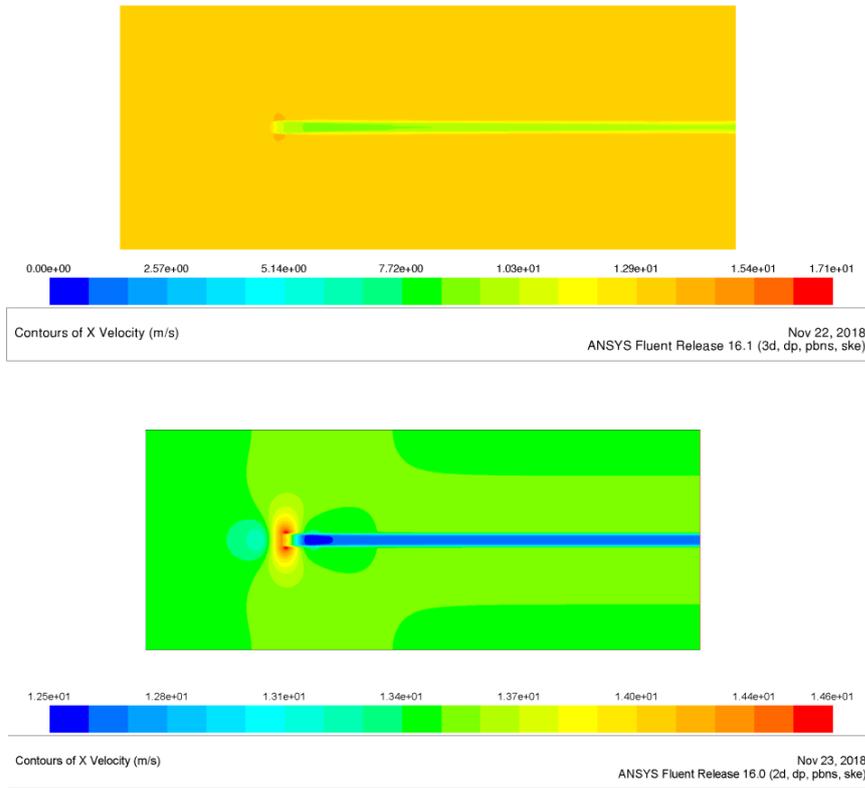


Figure 3.23: X velocity profile for the horizontal plane at the hub height for 3D simulation (top) and velocity profile obtained from 2D DA simulation (bottom).

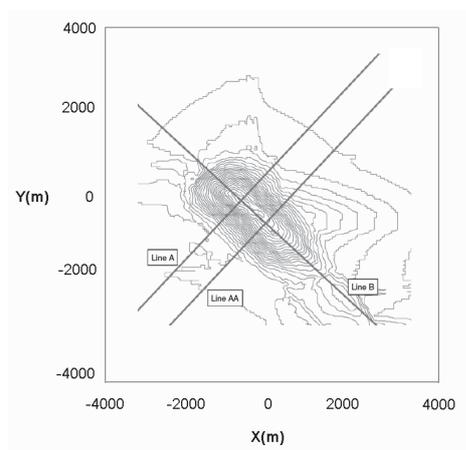


Figure 3.24: The topographic contours of Askervein Hill.

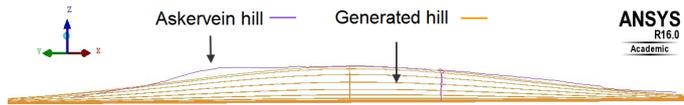


Figure 3.25: Topographic cross-sections through the hills.

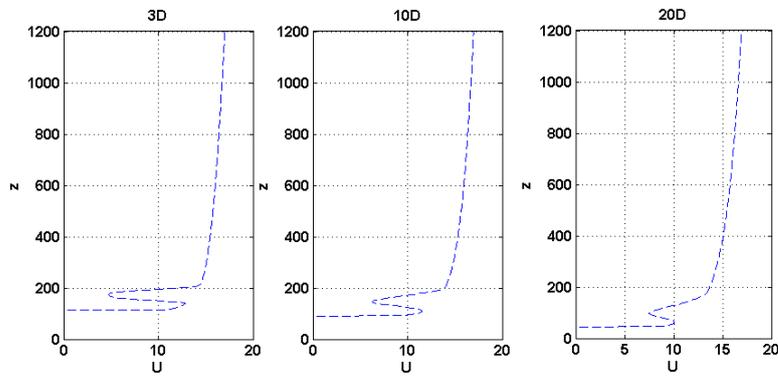


Figure 3.26: Vertical profiles of mean streamwise velocity downstream of the porous disc at  $x = 3DM$ ,  $x=10DM$  and  $x=20DM$ .

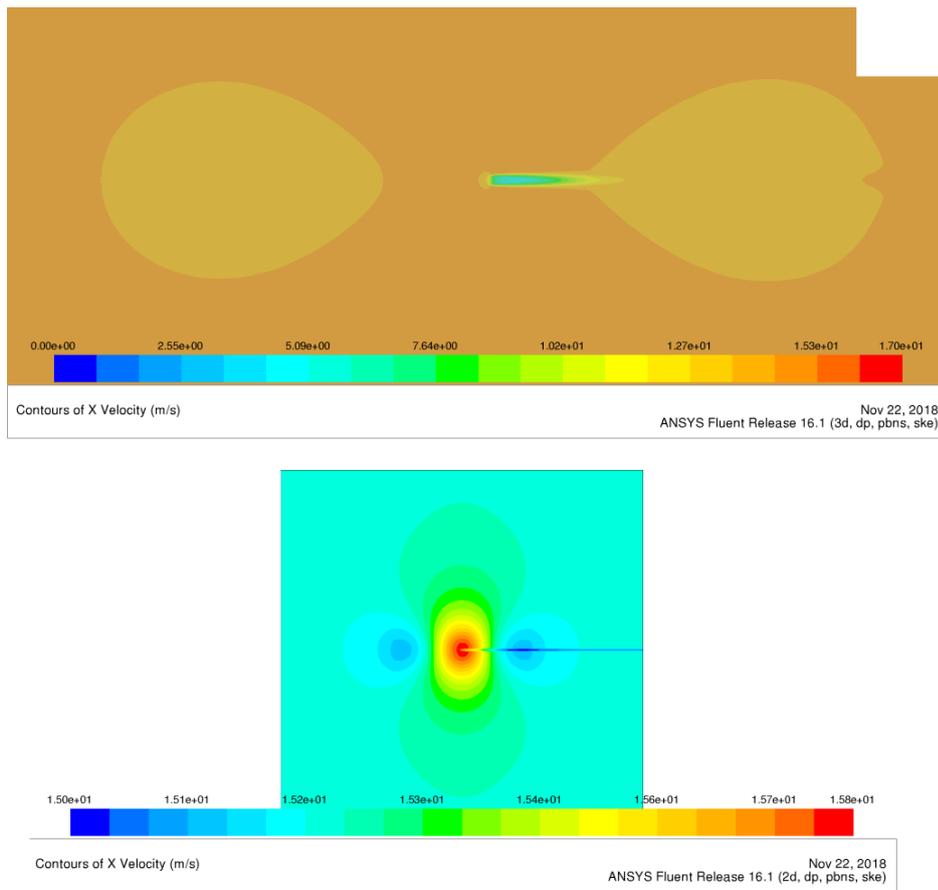


Figure 3.27: X velocity profile for the horizontal plane at the hub height for 3D simulation (top) and velocity profile obtained from 2D DA simulation (bottom).



## 4 Optimization results

In this chapter, the numerical results of the optimization examples are presented. The optimization algorithm and the modelling methods have been presented in the previous chapters. In the optimization algorithm, the calculation of objective functions was done with the fast depth-averaged model. It should be pointed out that one 3D simulation takes about 24 hours to converge using four processors, while solving a depth-averaged model takes less than 15 minutes using one processor (the time of convergence depends on the number of turbines). As a result, the whole optimization process with depth-averaged modelling takes approximately one week using one processor.

The overall goal of the optimization is to find the best position of turbines in the wind farm on a complex terrain. The number of wind turbines is fixed in this study and the Nordtank NTK 500/41 wind turbine is used here. The cost function to be maximized is the total power of the wind farm, calculated by using Equation (2.70). The objective of this chapter is to show that the depth-averaging equations can be applied for the optimization, but the real industrial optimization task is not in the focus of this work.

Two examples are presented in this work:

1. Wind farm optimization on the simple hill, as described by Equation (3.11).
2. The optimization of wind turbines on an analytically defined hill (the shape is described by Equation (3.13)) that is close in shape to Askervein Hill.

The number of turbines varies from one to two for the simple hill and from one to five for Askervein Hill. Single-objective optimization was accomplished using DE. The parameter values used for the optimization with DE are given in Table 4.1.

Table 4.1: The parameter values for DE.

Parameter	Value
Number of generations	50
Number of individuals	30
Crossover probability (CR)	0.9
Scale factor (F)	0.9

In all cases there were the following box constraints for all optimization parameters:

1. One constraint was the maximum power of one turbine. According to Figure 2.4, the maximum power of the presented turbine cannot exceed 600 KW.
2. The boundary constraint violations (the location of the wind turbines). For the simple hill the boundary box constraints look like  $300 \leq x \leq 700$ ,  $200 \leq y \leq 600$  m, where the centre of the hill is  $x = 500$ ,  $y = 400$ . The boundary box

constraints for the real hill are described by the following expression  $-2000 \leq x \leq 2000$ ,  $-2000 \leq y \leq 2000$  m. The point corresponding to the centre of the real hill is  $x = 0, y = 0$ .

3. The distance between turbines. It is possible to see from the real wind farm that the turbines are located at a certain distance from each other. They cannot be very close to each other. In this work, the distance between the centre of the turbines cannot be less than 70 m for the simple hill and 100 m for Askervein Hill. If the real case is different, this constraint can easily be changed.

#### 4.1 The simple hill

In the beginning, the optimization of one turbine is done to check the optimization algorithm. All the 2D CFD characteristics (the size of the computational domain, the wind direction, the mesh, etc.) are the same as in Section 3.3.1. In accordance with the practice, the best position of the turbine on the complex terrain with one hill is on the top of the hill. The initial population, the 15<sup>th</sup>, 30<sup>th</sup> and 50<sup>th</sup> generations, calculated during the optimization algorithm, are presented in Figure 4.1.

Hereinafter, the individual must be considered as one layout of turbines. The circle in Figure 4.1 is the top of the hill, the red star represents the best individual in the current population, and blue stars denote other individuals. Evolution can be seen when comparing different generations to each other. After 50 generations it was noted that every individual in the population had almost the same location. Hence, the optimization process had converged to a single optimum. The best location was validated by simulating it in 3D and comparing the solution to the initial location. The power increased by about 4.45%. The fast 2D DA model predicted the power increase accurately (4.41% in 2D DA), as can be seen in Figure 4.2.

In the second case, the total power of the wind farm was maximized for two turbines in the wind farm. It should be noted that the mesh for the 2D DA simulation is still the same as in Section 3.3.1. The turbines are added to the equations by using UDFs and the mesh does not need to be rebuilt, even if the number of hills or its shapes are changed. This is the main advantage of the depth-averaged method. The solutions of the 1st, 15th, 30th and 50th generations are shown in Figure 4.3.

The best individuals in the final population are located along the y-axis at approximately the same distance from the centre of the hill. The best geometry provided growth in the total power of the wind farm compared to the initial geometry of 6.43% in 3D and 6.58% in 2D DA, as can be seen from Figure 4.4.

Only one wind direction was considered in the previous examples. In reality, wind direction changes a lot over time. To include the direction of the wind in the optimization, the wind rose for the current landscape is considered. A wind rose is a circular plot used by

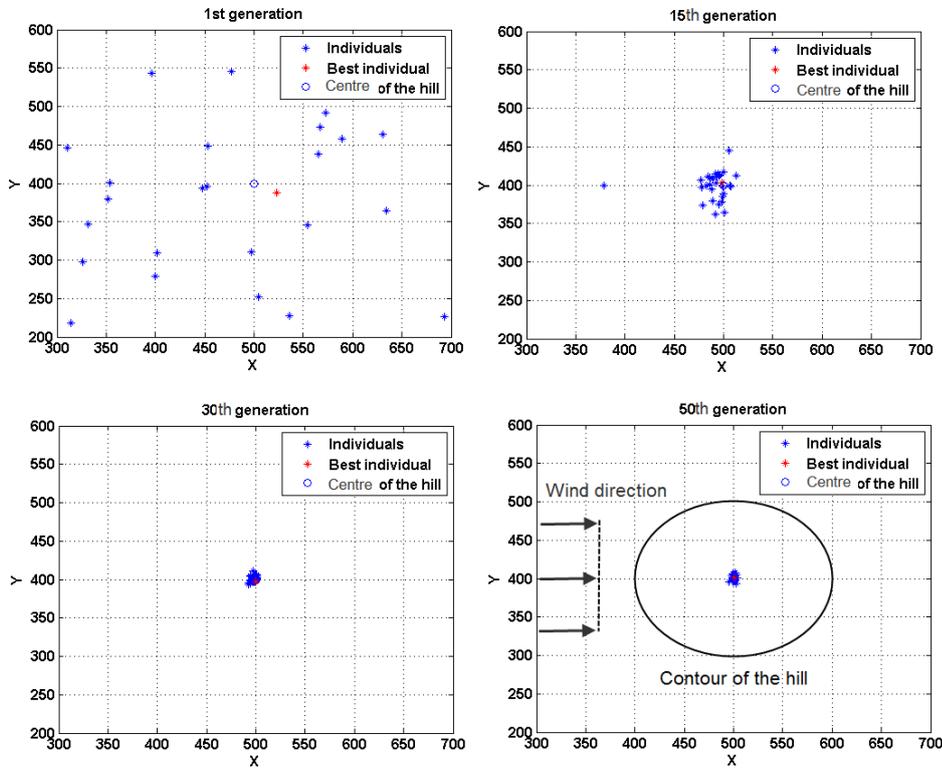


Figure 4.1: One turbine on the simple hill: the 1st, 15th, 30th and 50th generations.

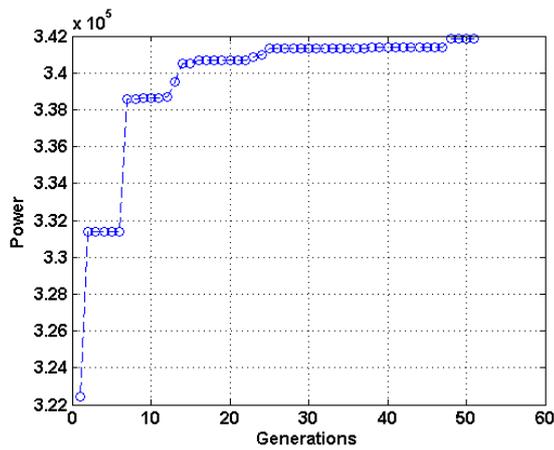


Figure 4.2: One turbine on the simple hill: the relationship between the power and generations.

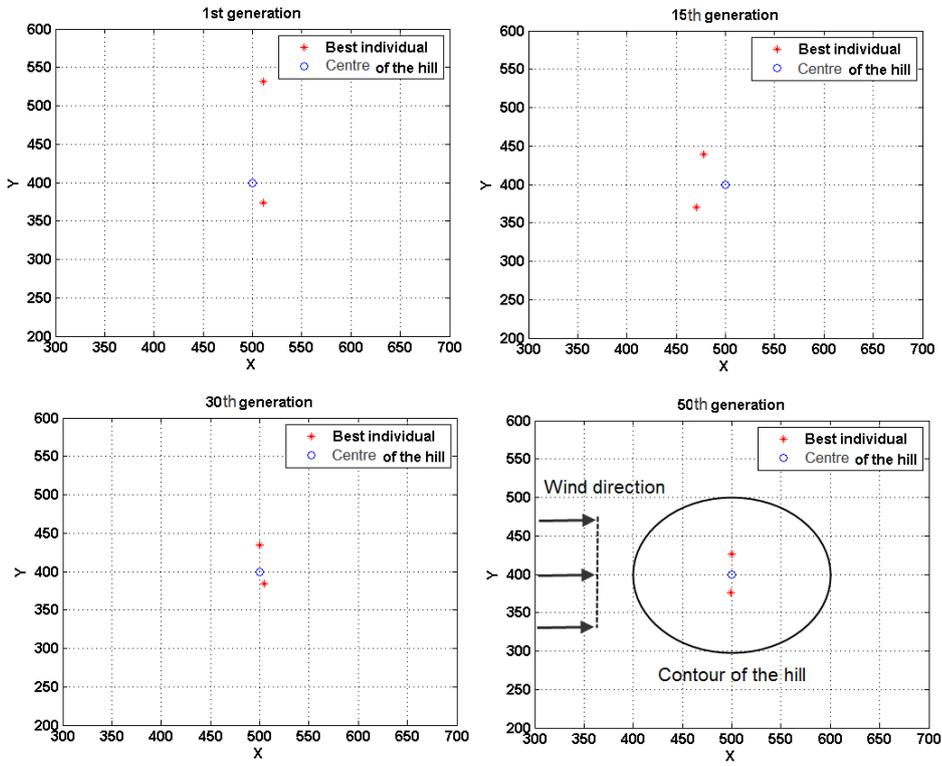


Figure 4.3: Two turbines on the simple hill: the 1st, 15th, 30th and 50th generations.

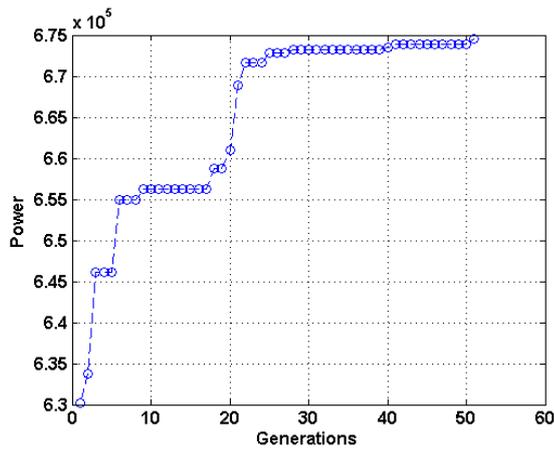


Figure 4.4: Two turbines on the simple hill: the relationship between the power and generations.

meteorologists to give a summary view of how the direction and speed of the wind are distributed at a concrete location (Figure 4.11). The wind rose presents the frequencies of the wind coming from different directions over a defined time period. The length of each line corresponds to the frequency that the wind blows per unit of time from a particular direction. A graphic of the wind rose may also include information about wind speed ranges. In this case, each line is divided into colour-coded bands that show the ranges of the wind speed. In the real optimization, the cost function should be defined as follows:

$$P = \sum_{i=1}^n \omega(i)P_i, \quad (4.1)$$

where  $n$  is the number of wind directions,  $\omega(i)$  denotes the frequency of the wind corresponding to the  $i$ th-wind direction and  $P_i$  is the total power of the wind farm, calculated for the  $i$ th wind direction and speed. In this work, only two wind directions are considered with one wind speed.

In this case, the wind blows from the west and south with various frequencies: 99% and 1%, 80% and 20%, 65% and 35%, 50% and 50% respectively. Figures 4.5 and 4.6 show the results for all these cases.

The results for the frequencies 99% and 1% are almost the same as in Figure 4.3, because the previous example can be considered as case with the frequencies 100% and 0%. The results practically do not differ from each other, which agrees with the theory. In the last case, the turbines are almost located at the same distance from the centre of the hill in the direction of  $45^\circ$  from the north. This result also agrees with theory, because the hill is spherical and in the case with the one wind speed, the turbines should be on a line oriented  $\pm 45^\circ$  from the north.

## 4.2 Askervein Hill

In this subsection, the analytically defined hill, close in shape to Askervein Hill, described by Equation (3.13), is considered a complex terrain. Firstly, the optimization of the one turbine on this hill is presented. Figure 4.7 presents the initial population and also the 15<sup>th</sup>, 30<sup>th</sup> and 50<sup>th</sup> generations obtained during the optimization algorithm.

As can be seen from Figure 4.7, the algorithm converges to the optimal solution. The best position of the one turbine, as in the case of the simple hill, is on the top of the hill. The calculated power is presented in Figure 4.8.

Compared to the initial position, the total power output is 2.92% higher in 2D DA modelling and 3.08% higher in 3D modelling.

In the following case, the optimization of the wind farm, containing two turbines, is pre-

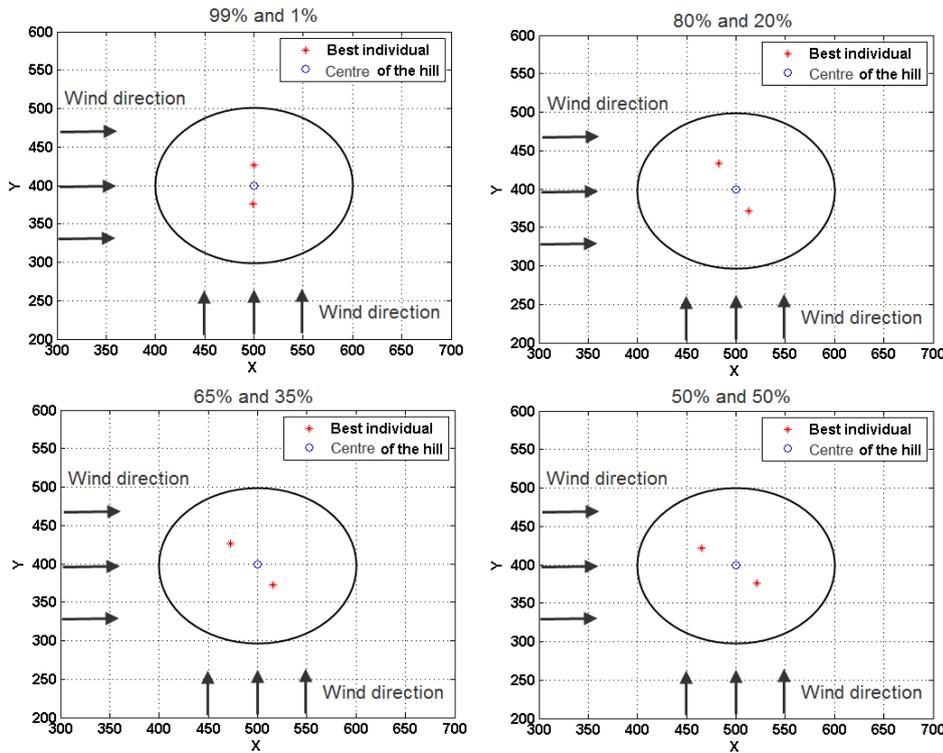


Figure 4.5: Two turbines on the simple hill: the final results when the wind blows from the west and south with various frequencies.

sented. Figure 4.9 shows the best positions of the turbines for different generations. The final results differ from the same case with the simple hill (see Figure 4.3). This can be explained by the fact that the evolutionary algorithm found this local maximum using the present initial population. This means that in the case of another initial population, the results could be different. Despite this, the total power increased by 4.59% in 2DDA (Figure 4.10) and 4.83% in 3D.

In the third optimization, the number of turbines was increased to five. One of the goals of this optimization was to obtain data about the full wind map over Askervein Hill. The wind rose (see Figure 4.11) found in Stangroom (2004) provides information for the twelve wind rose directions. All twelve directions can be used if required, but only two of them were modelled here, that is winds from the south and from the west with frequencies of 56% and 44%.

In reality, Askervein Hill is oriented at an angle of  $133^\circ$  anticlockwise from the east (line B in Figure 3.24) and the wind blows from the west and south. In this work, to simplify calculations, the hill is oriented along the x-axis, but the wind blows from  $227^\circ$  and  $317^\circ$  east to keep the correct wind direction relative to the hill. Three different wind regimes

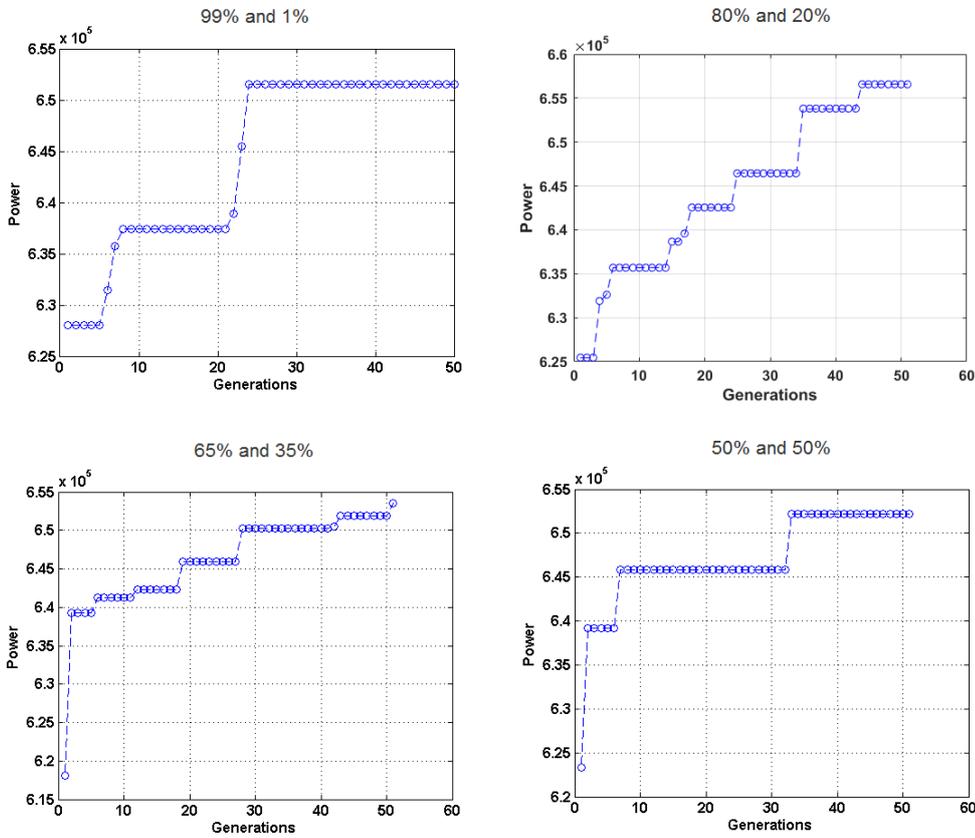


Figure 4.6: Two turbines on the simple hill: the relationship between the power and generations when the wind blows from the west and south with various frequencies.

were selected ( $227^\circ$ ,  $317^\circ$  from the east, and from both of these directions with frequencies of 56% and 44%) and optimal layouts were obtained for each regime (see Figures 4.12-4.13).

The total power is increased by 5.02% for Case 1, 4.39% for Case 2 and 4.68% for Case 3 when compared to the total power values calculated for the initial layouts. These examples demonstrate that the depth-averaged method produces layouts that can compete with other optimization algorithms in comparable conditions and outperforms optimization with full 3D simulations.

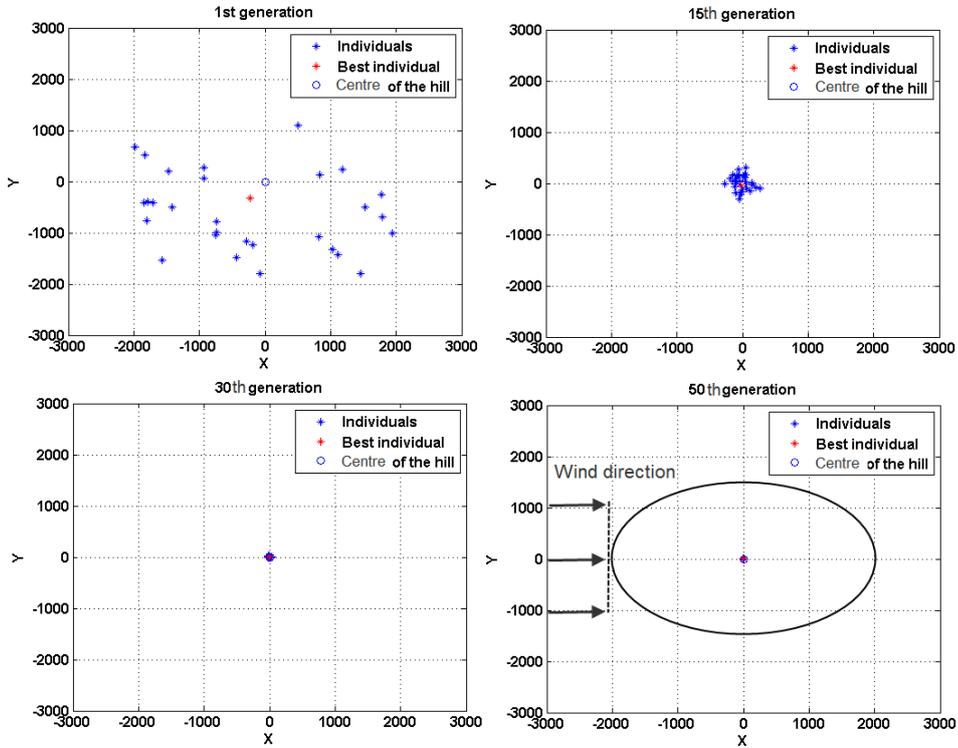


Figure 4.7: One turbine on Askervein Hill: the 1st, 15th, 30th and 50th generations.

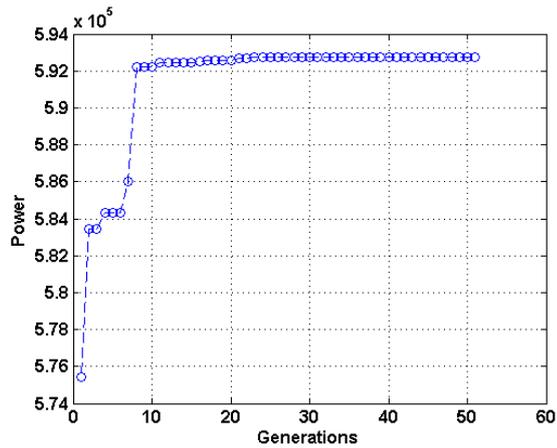


Figure 4.8: One turbine on Askervein Hill: the relationship between the power and generations.

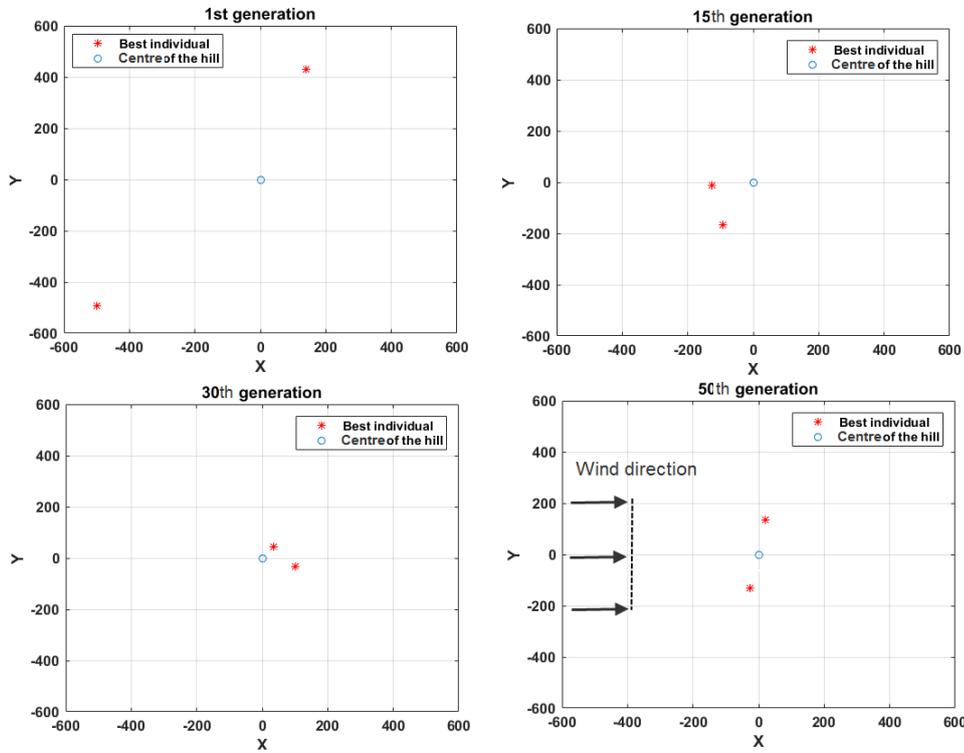


Figure 4.9: Two turbines on Askervein Hill: the 1st, 15th, 30th and 50th generations.

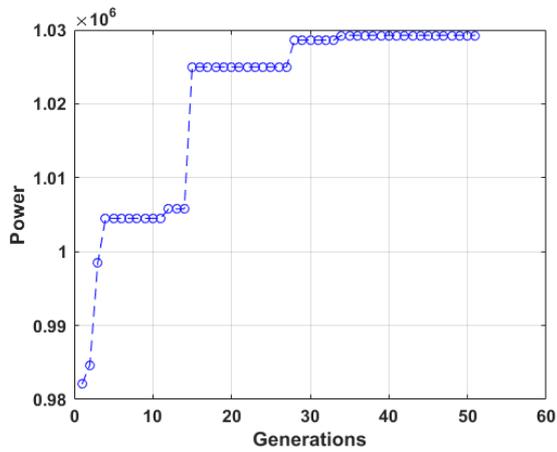


Figure 4.10: Two turbines on Askervein Hill: the relationship between the power and generations.

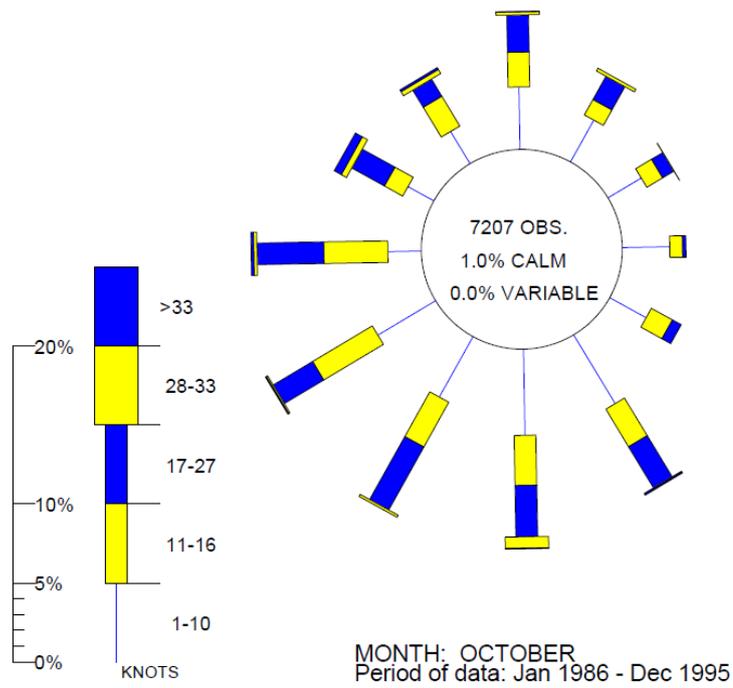


Figure 4.11: The wind rose of Askervein Hill, obtained from Stangroom (2004).

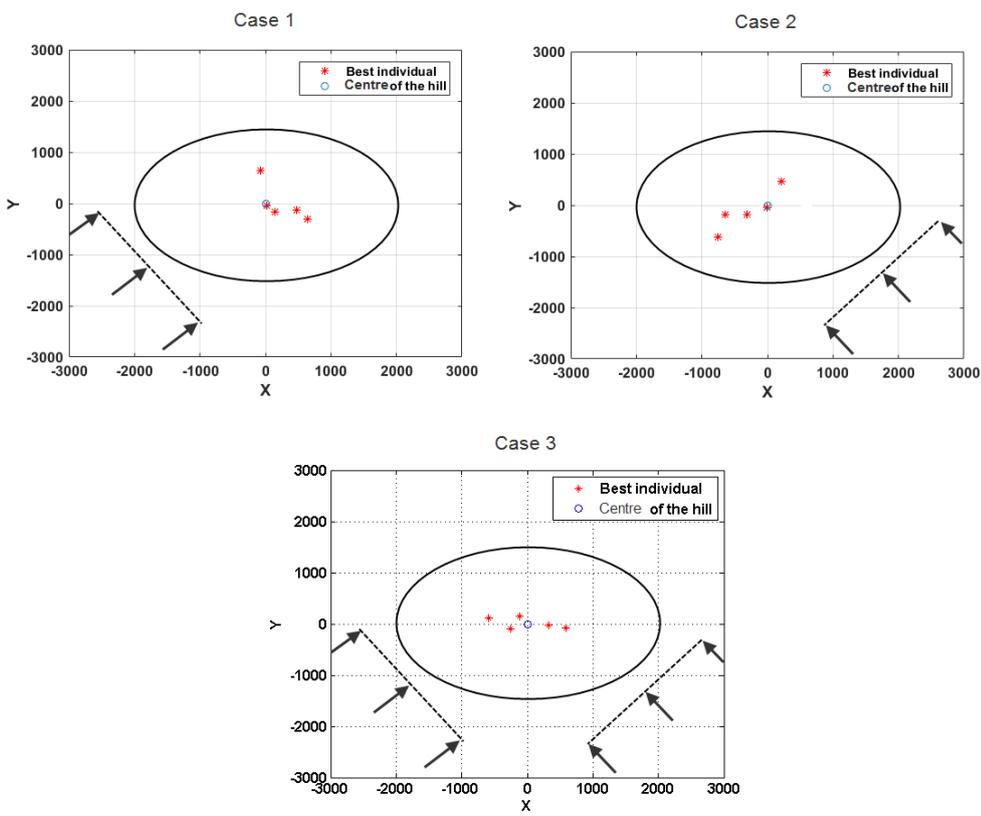


Figure 4.12: Five turbines on Askervein Hill: the results for different cases.

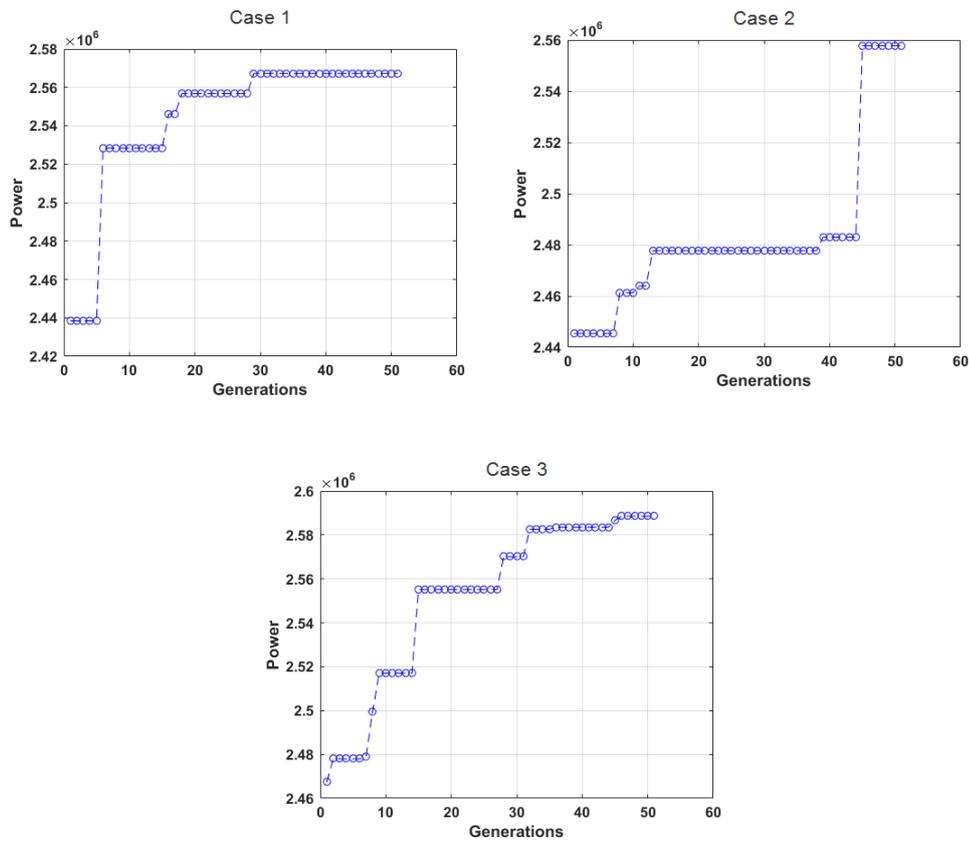


Figure 4.13: Five turbines on Askervein Hill: the relationship between the power and generations for different cases.

## 5 Conclusions

CFD-based modelling and the optimization of a wind farm on complex terrain were performed in this thesis. The obtained results were compared with laboratory experiments. Since 3D CFD simulation often takes a long time, a fast depth-averaged model was introduced for wind turbine modelling on the complex terrain. This method was combined with the optimization algorithm for the wind farm optimization. The 2D CFD simulation is based on the depth-averaged equations, but it predicts the values of depth-averaged velocity accurately. This approach allows the optimization of the position of wind turbines in the wind farm in a reasonable time. In the wind farm optimization, the main idea is to find the best position for wind turbines on the complex terrain. After optimization, the best layout was compared with the more accurate 3D model to obtain more detailed information about the improved total power of the wind farm.

The initial point of this work was to derive 2D governing equations using the depth-averaged approach. Firstly, equations were obtained for flow over the complex terrain without turbines. Then, the new method was proposed to model the turbines in 2D simulation. The depth of the channel and turbines are included in the governing equations via the source terms. The usability of the 3D and 2D models was proved by comparing the results with laboratory measurements. Two experiments were considered: the RUSHIL wind tunnel study, and wind tunnel tests in the PRISME laboratory. In the first case, the flow simulation over one two-dimensional hill was investigated. A numerical modelling of the wind turbine on smooth terrain was performed in the second experiments. In both of these tests, the error between depth-averaged velocity, calculated in 2D and 3D, was less than 5%. This made it possible to expect that the results would also be sufficiently good for the wind turbine on the complex terrain. Two geometries were considered: a simple hill and Askervein Hill. The final results showed that the depth-averaged model predicts the velocity with very good accuracy. The fast model was used in the optimization algorithm since it made the process much faster.

The single-objective differential evolution algorithm was used in the optimization, which aimed at improving only one condition (the total power of the wind farm). The optimization of the wind farm consisting of one to five turbines was performed on two hills: firstly, on the simple hill and then on Askervein Hill. The objective function for the optimization was calculated using a fast model that is fast and accurate. In general, the final total power of the wind farm was higher by 5% than the power calculated for the initial position of turbines. The optimization results were also examined using accurate 3D CFD simulation.

The main advantage of this method is its fast reaction to the variations of a complex terrain and the number of turbines. Therefore, the depth-averaged approach could be a useful tool for wind farm designers and could be used as an optimization tool during the design phase. Although the method presented in this work makes it possible to carry out wind farm optimization, there is still a need for further development. Firstly, an analytically defined terrain was considered in this work to avoid problems in the gradient

calculation of the depth of the channel. Secondly, all simulation was performed on a personal computer using the ANSYS Fluent serial solver. For the real wind farm, where the computational domain is more than  $5 \text{ km} \times 5 \text{ km}$ , parallel computing should be used to reduce the time taken for simulation. The development of modelling and optimization of the wind farm for real-life situations should be continued.

## References

- Abdi, D. and Bitsuamlak, G. (2014). Wind flow simulations on idealized and real complex terrain using various turbulence models. *Advances in Engineering Software*, 75, pp. 30–41.
- Adamski, S.J. (2013). *Numerical Modeling of the Effects of a Free Surface on the Operating Characteristics of Marine Hydrokinetic Turbines*. Master's thesis. University of Washington.
- Agafonova, O. (2017). *A Numerical Study of Forest Influences on the Atmospheric Boundary Layer and Wind Turbines*. Ph.D. thesis. Lappeenranta University of Technology.
- Akpınar, A. (2013). Evaluation of wind energy potentiality at coastal locations along the north eastern coasts of Turkey. *Energy*, 50, pp. 395–405.
- Anderson, J. (1995). *Computational Fluid Dynamics: The Basics with Applications*. New York: McGraw-Hill, Inc.
- Astolfi, D., Castellani, F., and Terzi, L. (2018). A Study of Wind Turbine Wakes in Complex Terrain Through RANS Simulation and SCADA Data. *Journal of Solar Energy Engineering*, 140, pp. 389–403.
- Aubrun, S., Loyer, S., Hancock, P., and Hayden, P. (2013). Wind turbine wake properties: Comparison between a non-rotating simplified wind turbine model and a rotating model. *Journal of Wind Engineering and Industrial Aerodynamics*, 120, pp. 1–8.
- Avramenko, A., Agafonova, O., Avramenko, Y., and Haario, H. (2017). Fast numerical modelling method for wind flow investigation based on depth-averaged equations. *WIT Transactions on Engineering Sciences*, 118, pp. 21–29.
- Avramenko, A., Agafonova, O., Sorvari, J., and Haario, H. (2015). Numerical modeling of flow around a wind turbine using a fast method based on depth-averaged equations. *Global Journal of Pure and Applied Mathematics*, 11, pp. 5057–5073.
- Avramenko, A., Agafonova, O., Sorvari, J., and Haario, H. (2016). Simulation of depth-integrated flow over a hill. *AIP Conference Proceedings*, 1738, pp. 480093–1–480093–4.
- Avramenko, A. and Haario, H. (2015). Numerical Modeling of the Depth-Averaged Flow Over a Hill. *International Journal of Mathematical, Computational, Statistical, Natural and Physical Engineering*, 9, pp. 11–14.
- Avramenko, Y. (2005). *Case-Based Design Method for Chemical Product and Process Development*. Ph.D. thesis. Lappeenranta University of Technology.
- Blocken, B., Hout, A., Dekker, J., and Weiler, O. (2015). CFD simulation of wind flow over natural complex terrain: Case study with validation by field measurements for Ria de Ferrol. *Journal of Wind Engineering and Industrial Aerodynamics*, 147, pp. 43–57.

- Burton, T., Sharpe, D., Jenkins, N., and Bossanyi, E. (2001). *Wind Energy Handbook*. John Wiley & Sons Ltd: Chichester.
- Carcangiu, C. (2008). *CFD-RANS Study of Horizontal Axis Wind Turbines*. Ph.D. thesis. University of Cagliari.
- Castro, I. and Apsley, D. (1997). Flow and dispersion over topography: A comparison between numerical and laboratory data for two-dimensional flows. *Atmospheric Environment*, 31, pp. 839–850.
- Dahlberg, J. (2009). *Assessment of the Lillgrund windfarm*. LG Pilot Report.
- Dinler, A. (2013). A new low-correlation MCP (measure-correlate-predict) method for wind energy forecasting. *Energy*, pp. 152–160.
- Eiben, A. and Smith, J. (2003). *Introduction To Evolutionary Computing*. Springer.
- Feng, J. and Shen, W.Z. (2014). Wind farm layout optimization in complex terrain: A preliminary study on a Gaussian hill. *Journal of Physics: Conference Series*, 524.
- Fraille, D. and Mbistrova, A. (2018). *Wind in power 2017. Annual combined onshore and offshore wind energy statistics*. Wind Europe.
- Grady, S., Hussaini, M., and Abdullah, M. (2005). Placement of wind turbines using genetic algorithms. *Renewable Energy*, 30, pp. 259–270.
- Gu, H. and Wang, J. (2013). Irregular-shape wind farm micro-siting optimization. *Energy*, 57, pp. 535–544.
- Gujarathi, A. and Babu, B. (2016). *Evolutionary Computation: Techniques and Applications*. CRC Press, Taylor & Francis Group.
- Hämäläinen, J. and Tiihonen, T. (1995). Modelling and simulation of fluid flows in a paper machine headbox. *ICIAM 95, Issue 4: Applied sciences, especially mechanics (minisymposia)*, Ed. by E. Kreuzer, O. Mahrenholtz, *Third International Congress on Industrial and Applied Mathematics, Hamburg*, pp. 62–66.
- Hämäläinen, J., Tiihonen, T., Madetoja, E., and Ruotsalainen, H. (2008). CFD-based optimization for a complete industrial process: Papermaking. *Optimization and Computational Fluid Dynamics*, D. Thevenin and G. Janiga (eds.), Springer.
- Hau, E. (2000). *Wind turbines*. Berlin: Springer. 380–412 p.
- Herbert-Acero, J., Probst, O., Rethore, P., and Larsen, G. (2014). A Review of Methodological Approaches for the Design and Optimization of Wind Farms. *Energies*, pp. 6930–7016.
- Hyvarinen, A. and Segalini, A. (2017). Effects From Complex Terrain on Wind-Turbine Performance. *Journal of Energy Resources Technology*, 139, pp. 051205–051215.

- Jensen, N.O. (1983). *A note on wind generator interaction*. Denmark: Riso National Laboratory.
- Jonkman, J., Butterfield, S., Musial, W., and Scott, G. (2009). *Definition of a 5-MW reference wind turbine for offshore system development*. National Renewable Energy Laboratory: Technical Report NREL/TP-500-38060.
- Kang, S. and Hirsch, C. (2001). Features of the 3D flow around wind turbine blades based on numerical solutions. *In Proceedings of EWEC 06*.
- Katic, I., Hojstrup, J., and Jensen, N. (1986). A simple model for cluster efficiency. *European wind energy association conference and exhibition*, pp. 407–410.
- Khurshudyan, L., Snyder, W., and Nekrasov, V. (1980). *Flow and dispersion of pollutants over two-dimensional hills*. United States Environmental Protection Agency.
- Kidmo, D., Danwe, R., Doka, S., and Djongyang, N. (2015). Statistical analysis of wind speed distribution based on six Weibull Methods for wind power evaluation in Garoua, Cameroon. *Revue des Energies Renouvelables*, 18, pp. 105–125.
- Kosmacheva, A. (2011). *Flow simulation in a gas compressor*. Master's thesis. Lappeenranta University of Technology.
- Kuo, J., Romero, D., Beck, J., and Amon, C. (2016). Wind Farm Layout Optimization on Complex Terrains - Integrating a CFD Wake Model with Mixed-Integer Programming. *Applied Energy*, 178, pp. 404–414.
- Lyytikäinen, M. (2009). *Modelling and Optimising of Chevron-type Plate Heat Exchangers*. Ph.D. thesis. University of Kuopio.
- Lyytikäinen, M., Hämäläinen, T., and Hämäläinen, J. (2009). A fast modelling tool for plate heat exchangers based on depth-averaged equations. *International Journal of Heat and Mass Transfer*, 52, pp. 1132–1137.
- Magalhaes-Mendes, J. (2013). A Comparative Study of Crossover Operators for Genetic Algorithms to Solve the Job Shop Scheduling Problem. *WSEAS Transactions on Computers*, 12.
- Marmidis, G., Lazarou, S., and Pyrgioti, E. (2008). Optimal placement of wind turbines in a wind park using Monte Carlo simulation. *Renewable Energy*, 33, pp. 1455–1460.
- Martinez-Tossas, L.A., Churchfield, M., and Meneveau, C. (2015). Large Eddy Simulation of wind turbine wakes: detailed comparisons of two codes focusing on effects of numerics and subgrid modeling. *Journal of Physics: Conference Series*, 625.
- Mikkelsen, R. (2003). *Actuator Disc Methods Applied to Wind Turbines*. Ph.D. thesis. Technical University of Denmark.

- Mittal, A. and Taylor, L. (2012). Optimization of Large Wind Farms Using a Genetic Algorithm. *ASME 2012 International Mechanical Engineering Congress and Exposition*, 7, pp. 3159–3172.
- Montavon, C. (1998). *Simulation of Atmospheric Flows over Complex Terrain for Wind Power Potential Assessment*. Ph.D. thesis. Ecole polytechnique federale de Lausanne.
- Mozafari, A. (2010). *Numerical Modeling of Tidal Turbines: Methodology Development and Potential Physical Environmental Effects*. Master's thesis. University of Washington.
- Murali, A. and Rajagopalan, R. (2017). Numerical simulation of multiple interacting wind turbines on a complex terrain. *Journal of Wind Engineering and Industrial Aerodynamics*, 162, pp. 57–72.
- Nakayama, H. and Nagai, H. (2010). Large-Eddy Simulation on turbulent flow and plume dispersion over a 2-dimensional hill. *Advances in Science and Research*, 4, pp. 71–76.
- Nilsson, K., et al. (2014). Large-eddy simulations of the Lillgrund wind farm. *Wind Energy*, 18, pp. 449–467.
- Ohya, Y. (2001). Wind-tunnel study of atmospheric stable boundary layers over a rough surface. *Boundary-Layer Meteorology*, 98, pp. 57–82.
- Ozturk, U. and Norman, B. (2004). Heuristic methods for wind energy conversion system positioning. *Electric Power Systems Research*, 70, pp. 179–185.
- Pascheke, F. and Hancock, P. (2009). Influence of ABL characteristics on wind turbine wakes: surface roughness and stratification. *Proceedings of the International Workshop Physmod*.
- Paulsen, U.S. (1995). *Konceptundersogelse Nordtank 500/41 Strukturelle laster (in Danish)*. Riso National Laboratory.
- Price, K., Storn, R., and Lampinen, J. (2011). *Differential Evolution : A Practical Approach to Global Optimization*. Springer-Verlag Berlin and Heidelberg.
- Qamar, A., Zhang, W., Gao, W., and Samtaney, R. (2014). Shear Layer and Flow Transition on a Low Speed Full Wind Turbine Blade. *19th Australasian Fluid Mechanics Conference*.
- Rastogi, A. and Rodi, W. (1978). Predictions of heat and mass transfer in open channels. *Journal of the Hydraulics Division, ASCE, No. HY3*, pp. 397–420.
- Rivas, R., Clausen, J., Hansen, K., and Jensen, L. (2009). Solving the turbine positioning problem for large offshore wind farms by simulated annealing. *Wind Engineering*, 33, pp. 287–298.

- Rodi, W. (1984). *Turbulence Models and Their Application in Hydraulics - A State of the Art Review*. Ashgate Pub Co.
- Rongliang, C., Zhengzheng, Y., and Yubo, Z. (2013). Simulating 3D Flows Passing Wind Turbine Rotors with a Domain Decomposition Method on a Moving Domain. *Procedia Engineering*, 61, p. 405.
- Rosales, C. (2012). *Optimal Placement of Wind Turbines on Non-flat Terrain Using Cluster Identification and Multi-objective Genetic Algorithm*. Master's thesis. The University of Texas at el Paso.
- Segalini, A. and Castellani, F. (2017). Wind-farm simulation over moderately complex terrain. *Journal of Physics: Conference Series*, 854.
- Sip, V. and Ludek, B. (2016). RANS solver for microscale pollution dispersion problems in areas with vegetation: Development and validation. *arXiv.org e-Print Archive*.
- Sladek, I., Kozel, K., and Janour, Z. (2009). On the 2D-Validation Study of the Atmospheric Boundary Layer Flow Model Including Pollution Dispersion. *Engineering Mechanics*, 16, pp. 323–333.
- Song, M., Chen, K., He, Z., and Zhang, X. (2014). Optimization of wind farm micro-siting for complex terrain using greedy algorithm. *Energy*, 67, pp. 454–459.
- Sorensen, N., Johansen, J., and Conway, S. (2004). CFD computations of wind turbine blade loads during standstill operation. *Technical Report R-1465, Riso National Laboratory*.
- Souza, L., Janiga, G., and Thevenin, D. (2017). Multi-objective optimisation of the model parameters for the realisable k-e turbulence model. *Progress in Computational Fluid Dynamics*, 17, pp. 90–101.
- Spears, V. (2000). *Evolutionary Algorithms: The Role of Mutation and Recombination*. Springer.
- Stangroom, P. (2004). *CFD modelling of wind flow over terrain*. Ph.D. thesis. University of Nottingham.
- Stergiannis, N., Beeck, J., and Runacres, M. (2017). Full HAWT rotor CFD simulations using different RANS turbulence models compared with actuator disk and experimental measurements. *Wind Energy Science*, 6.
- Storn, R. and Price, K. (1995). *Differential Evolution - A Simple and Efficient Adaptive Scheme for Global Optimization Over Continuous Spaces*. International Computer Science Institute.
- Svenning, E. (2010). *Implementation of an actuator disk in OpenFOAM*. Bachelor thesis. Chalmers University of Technology.

- Uchida, T. and Ohya, Y. (2003). Large-eddy simulation of turbulent airflow over complex terrain. *Journal of Wind Engineering and Industrial Aerodynamics*, 91, pp. 219–229.
- Vafiadis, K., Stergiannis, N., Turlidakis, A., and Rados, K. (2013). Computational Investigation of Horizontal Axis Wind Turbine Wake. *EWEA Conference*.
- Versteeg, H. and Malalasekera, W. (1995). *Introduction to Computational Fluid Dynamics. The finite volume method*. Longman Scientific & Technical.
- Walmsley, J. and Taylor, P. (1996). Boundary-layer flow over topography: Impacts of the Askervein study. *Boundary-Layer Meteorology*, 78, pp. 291–320.
- Wan, C., Wang, J., Yang, G., and Zhang, X. (2009). Optimal siting of wind turbines using real-coded genetic algorithms. *European wind energy association conference and exhibition*.
- Wan, C., Wang, J., Yang, G., and Zhang, X. (2010). Optimal micro-siting of wind farms by particle swarm optimization. *Advances in swarm intelligence. First International conference*, pp. 198–205.
- Wu, B., Song, M., Chen, K., and He, Z. (2014). Wind power prediction system for wind farm based on auto regressive statistical model and physical model. *Journal of Renewable and Sustainable Energy*, 6:013101.
- Wu, Y. and Porte-Agel, F. (2011). Large-eddy simulation of wind-turbine wakes: Evaluation of turbine parametrisations. *Boundary-Layer Meteorology*, 138, pp. 345–366.
- Yu, J., Chen, K., Mori, J., and Rashid, M. (2013). A Gaussian mixture copula model based localized Gaussian process regression approach for long-term wind speed prediction. *Energy*, 61, pp. 673–686.
- Zhang, Y., Hou, G., and Wang, J. (2011). A fast algorithm based on the submodular property for optimization of wind turbine positioning. *Renewable Energy*, 36, pp. 2951–2958.
- Zhang, Y., Sun, Z., Zuijlen, A., and Bussel, G. (2017). Numerical simulation of transitional flow on a wind turbine airfoil with RANS-based transition model. *Journal of Turbulence*, 18, pp. 879–898.

## ACTA UNIVERSITATIS LAPPEENRANTAENSIS

813. KARTTUNEN, ELINA. Management of technological resource dependencies in interorganizational networks. 2018. Diss.
814. CHILD, MICHAEL. Transition towards long-term sustainability of the Finnish energy system. 2018. Diss.
815. NUTAKOR, CHARLES. An experimental and theoretical investigation of power losses in planetary gearboxes. 2018. Diss.
816. KONSTI-LAAKSO, SUVI. Co-creation, brokering and innovation networks: A model for innovating with users. 2018. Diss.
817. HURSKAINEN, VESA-VILLE. Dynamic analysis of flexible multibody systems using finite elements based on the absolute nodal coordinate formulation. 2018. Diss.
818. VASILYEV, FEDOR. Model-based design and optimisation of hydrometallurgical liquid-liquid extraction processes. 2018. Diss.
819. DEMESA, ABAYNEH. Towards sustainable production of value-added chemicals and materials from lignocellulosic biomass: carboxylic acids and cellulose nanocrystals. 2018. Diss.
820. SIKANEN, EERIK. Dynamic analysis of rotating systems including contact and thermal-induced effects. 2018. Diss.
821. LIND, LOTTA. Identifying working capital models in value chains: Towards a generic framework. 2018. Diss.
822. IMMONEN, KIRSI. Ligno-cellulose fibre poly(lactic acid) interfaces in biocomposites. 2018. Diss.
823. YLÄ-KUJALA, ANTTI. Inter-organizational mediums: current state and underlying potential. 2018. Diss.
824. ZAFARI, SAHAR. Segmentation of partially overlapping convex objects in silhouette images. 2018. Diss.
825. MÄLKKI, HELENA. Identifying needs and ways to integrate sustainability into energy degree programmes. 2018. Diss.
826. JUNTUNEN, RAIMO. LCL filter designs for parallel-connected grid inverters. 2018. Diss.
827. RANA EI, SAMIRA. Quantitative approaches for detecting emerging technologies. 2018. Diss.
828. METSO, LASSE. Information-based industrial maintenance - an ecosystem perspective. 2018. Diss.
829. SAREN, ANDREY. Twin boundary dynamics in magnetic shape memory alloy Ni-Mn-Ga five-layered modulated martensite. 2018. Diss.
830. BELONO GOVA, NADEZDA. Active residential customer in a flexible energy system - a methodology to determine the customer behaviour in a multi-objective environment. 2018. Diss.

- 831.** KALLIOLA, SIMO. Modified chitosan nanoparticles at liquid-liquid interface for applications in oil-spill treatment. 2018. Diss.
- 832.** GEYDT, PAVEL. Atomic Force Microscopy of electrical, mechanical and piezo properties of nanowires. 2018. Diss.
- 833.** KARELL, VILLE. Essays on stock market anomalies. 2018. Diss.
- 834.** KURONEN, TONI. Moving object analysis and trajectory processing with applications in human-computer interaction and chemical processes. 2018. Diss.
- 835.** UNT, ANNA. Fiber laser and hybrid welding of T-joint in structural steels. 2018. Diss.
- 836.** KHAKUREL, JAYDEN. Enhancing the adoption of quantified self-tracking wearable devices. 2018. Diss.
- 837.** SOININEN, HANNE. Improving the environmental safety of ash from bioenergy production plants. 2018. Diss.
- 838.** GOLMAEI, SEYEDMOHAMMAD. Novel treatment methods for green liquor dregs and enhancing circular economy in kraft pulp mills. 2018. Diss.
- 839.** GERAMI TEHRANI, MOHAMMAD. Mechanical design guidelines of an electric vehicle powertrain. 2019. Diss.
- 840.** MUSIIENKO, DENYS. Ni-Mn-Ga magnetic shape memory alloy for precise high-speed actuation in micro-magneto-mechanical systems. 2019. Diss.
- 841.** BELIAEVA, TATIANA. Complementarity and contextualization of firm-level strategic orientations. 2019. Diss.
- 842.** EFIMOV-SOINI, NIKOLAI. Ideation stage in computer-aided design. 2019. Diss.
- 843.** BUZUKU, SHQIPE. Enhancement of decision-making in complex organizations: A systems engineering approach. 2019. Diss.
- 844.** SHCHERBACHEVA, ANNA. Agent-based modelling for epidemiological applications. 2019. Diss.
- 845.** YLIJOKI, OSSI. Big data - towards data-driven business. 2019. Diss.
- 846.** KOISTINEN, KATARIINA. Actors in sustainability transitions. 2019. Diss.
- 847.** GRADOV, DMITRY. Experimentally validated numerical modelling of reacting multiphase flows in stirred tank reactors. 2019. Diss.
- 848.** ALMPANOPOULOU, ARGYRO. Knowledge ecosystem formation: an institutional and organisational perspective. 2019. Diss.
- 849.** AMELI, ALIREZA. Supercritical CO2 numerical modelling and turbomachinery design. 2019. Diss.
- 850.** RENEV, IVAN. Automation of the conceptual design process in construction industry using ideas generation techniques. 2019. Diss.





ISBN 978-952-335-366-4  
ISBN 978-952-335-367-1 (PDF)  
ISSN-L 1456-4491  
ISSN 1456-4491  
Lappeenranta 2019