



NTNU – Trondheim
Norwegian University of
Science and Technology

Investigation and implementation of turbulent wind in a specialized software tool for offshore wind turbines

Hans Børge Pedersen

Master of Science in Engineering and ICT

Submission date: Januar 2014

Supervisor: Ole Gunnar Dahlhaug, EPT

Co-supervisor: Per Ivar Bruheim, Simis AS

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

*Big whirls have little whirls that feed on their velocity, and little whirls have lesser whirls
and so on to viscosity.*
- Lewis Fry Richardson

Summary

This thesis presents the theoretical models and standards behind the generation of turbulence in aeroelastic software. The required knowledge behind offshore wind have been investigated and the necessary changes from onshore wind generation are described. An integration between a turbulent wind generator Turbsim and an aeroelastic software, Ashes are implemented, tested and documented. The integration was focused on usability and the application of standards. A comparison with other tools for turbulent wind in the OC4 report was compared with the result produced with Turbsim. Alternatives to the implemented turbulent wind generator was investigated and described.

Sammendrag

Denne masteroppgaven presenterer de teoretiske modellene og standardene bak generering av turbulens i aeroelastisk programvare. Forskjellene mellom generering av vind på land of offshore er beskrevet. En integrasjon mellom programvare for generering av turbulent vind, Turbsim og den aeroelastiske programvaren Ashes er implementert, testet og dokumentert. Integreringen hadde stor fokus på brukervennlighet og at standarder ble fulgt. En sammenlikning med andre verktøy for turbulent vind i OC4 rapporten ble sammenliknet med resultatene fra Turbsim. Alternativer til Turbsim ble utforsket og beskrevet.

Preface

The purpose of this master thesis has been to get familiar with the methods of producing a set of numerical values representing the wind speed at different points across a two dimensional grid. This method is used in modern aeroelastic software to perform tests on a modelled wind turbine before construction. An integration of these methods in to an existing aeroelastic software, Ashes was performed.

I would like to thank my supervisors Ole Gunnar Dahlhaug and Per Ivar Bruheim. I would also like to thank Paul E. Thomassen for involving me in the field of renewable energy and providing me with useful literature on the subject. In Norway, it is considered by many that wind energy is not economically profitable with current energy prices. While this might be true with the current price of power, it is of my opinion that this can change, and more important, other countries can not rely as heavy on hydroelectric power as Norway. As carbon dioxide levels in the atmosphere is rising, wind power is one of the key solutions along with other renewables and can indirectly save lives if emission targets can be met, and the global warming is kept under control. I would also like to thank all the engineers at the National Wind Technology Center who provides open source software solutions with good documentation.

Contents

Summary	i
Sammendrag	ii
Preface	iii
Table of Contents	vii
List of Tables	ix
List of Figures	xii
Abbreviations	xiii
Terms and definitions	xv
1 Introduction	1
1.1 What is turbulence?	2
2 Use and theory of turbulent wind	3
2.1 Overview	3
2.2 How turbulence is originated	3
2.3 Standards	3
2.4 IEC Turbine classes	4
2.5 Statistical properties of turbulence	4
2.5.1 Wind Profile	5
2.5.2 Normal Turbulence Model	6
2.5.3 Spectral models of turbulence	6
2.6 Extreme gusts, winds, direction changes and shears	10
2.6.1 Extreme turbulence model	10
2.7 Coherence	11
2.7.1 Coherence in the Kaimal turbulence model	11
2.8 Turbsim	11
2.8.1 How Turbsim works	11
2.8.2 Wind speed distribution	12

3	Implementation and design	15
3.1	Tools	15
3.1.1	Qt	15
3.2	Design of the interface	16
3.2.1	Runtime options	17
3.2.2	Turbine/Model specifications	18
3.2.3	Meteorological boundary conditions	19
3.2.4	Extreme turbulence	20
3.3	Visualization	21
4	Testing	23
4.1	Verification and comparison	23
4.1.1	Specifications for OC4 tests	23
4.1.2	Comparison of data for load case 3.4a	24
4.1.3	Black box tests	25
5	Performance	31
5.1	Turbsim parameter study	31
5.1.1	Input	31
5.1.2	Equipment	31
5.1.3	Increase in runtime with number of gridpoints	34
5.1.4	Runtime variation with different timelength input.	36
5.1.5	64bit CPU performance	37
5.1.6	CPU usage	38
5.1.7	Runtime variation with different distances between gridpoints	39
5.1.8	Output files	40
5.1.9	Other factors in runtime	42
5.1.10	Memory requirements	42
5.1.11	Validation	44
6	Conclusion	47
6.1	Turbulent wind in aeroelastic software	47
6.2	Implementation	48
6.3	Testing	49
6.4	Offshore turbines	49
6.5	Parameters and performance	49
6.6	Further development	50
	Bibliography	53
	Appendix	55
.1	Appendix A: Runtimes for 4-1600 gridpoints	55
.2	Appendix B: A standard IEC Kaimal input file	57

.3

Appendix C: Quickstart guide for turbulent wind in Ashes 59

List of Tables

2.1	Turbine classes defined by IEC	4
4.1	Wind condition parameters for load case 3.4a.	23
5.1	Default values for the cases.	32
5.2	Variation in runtime with no input change.	32
5.3	Variation in runtime with different timelengths.	36
5.4	Variation in runtime between 32bit and 64bit Turbsim.	37
5.5	Variation in runtime with different Scale IEC parameters.	42
1	Measurements of runtime with increase in gridpoints. Performed on i7-3770K @ 3.50GHz.	56

List of Figures

2.1	Comparison of wind profile for offshore and onshore turbines using IEC specifications. 88,2 m hub height and 18 m/s average wind speed at hub height.	6
3.1	Qt Designer. Tool for creating GUI.	16
3.2	Turbulent wind user interface in Bladed. This might be overwhelming for new users.	17
3.3	Turbulent wind user interface in Ashes.	18
3.4	Runtime options. Blue fields are hidden from the user and set to default values. Yellow fields are available in advanced mode. Fields with a red line through are not available through the user interface. Fields with no added color are enabled for the user.	19
3.5	Options about the turbine/model. Blue fields are hidden from the user and set to default values. Yellow fields are available in advanced mode. Fields with a red line through are not available through the user interface. Fields with no added color are enabled for the user.	19
3.6	Options about the meteorological boundary conditions. Blue fields are hidden from the user and set to default values. Yellow fields are available in advanced mode. Fields with a red line through are not available through the user interface. Fields with no added color are enabled for the user.	20
3.7	New visualization of the wind forces in Ashes.	21
3.8	Earlier visualization of turbulent wind field in Ashes.	22
4.1	Comparison of probability density functions of longitudinal wind speed (OC4)	24
4.2	Comparison of probability density functions of longitudinal wind speed (OC4)	24
4.3	Comparison of probability density functions of lateral wind speed (OC4)	25
4.4	Comparison of probability density functions of vertical wind speed (OC4)	25
5.1	Comparison of laptop and stationary. This figure illustrates the variance of the runtime.	33
5.2	Comparison of laptop and stationary. This figure illustrates how the runtime develops with higher finer grids.	33

5.3	Runtime as function of gridpoints from 25 to 400 gridpoints.	34
5.4	Runtime as function of gridpoints from 400 to 5184 gridpoints with curves.	34
5.5	Runtime as function of timelength.	36
5.6	4 Turbsim processes.	38
5.7	Runtime as function of total width/height of the windfield.	39
5.8	Runtime as function of gridpoints. Comparison between output format.	40
5.9	Runtime as function of gridpoints. Comparison between output format with higher grid resolutions.	40
5.10	Output file size as function of number of gridpoints.	41
5.11	Maximum memory allocation as function of gridpoints.	44
5.12	Mean standard deviation of all points in the grid during the whole time-length without scaling.	45

Abbreviations

CPU	=	Central Processing Unit
ETM	=	Extreme Turbulence Model
EWM	=	Extreme Wind Model
FF	=	Full Field
FFT	=	Fast Fourier Transform
HH	=	Hub Height
IEC	=	International Electrotechnical Commission
NREL	=	National Renewable Energy Laboratory
NTM	=	Normal Turbulence Model
NWTC	=	National Wind Technology Center
OC4	=	Offshore Code Comparison Collaboration Continuation
pRNG	=	Pseudo Random Number Generator
RAM	=	Random Access Memory
TI	=	Turbulence Intensity

Terms and definitions

Annual average

Mean value of a set of measured data of sufficient size and duration to serve as an estimate of the expected value of the quantity. The averaging time interval should be a whole number of years to average out non-stationary effects such as seasonality.

Annual average wind speed

V_{ave} wind speed averaged according to the definition of annual average.

Extreme wind speed

Value of the highest wind speed, averaged over t seconds, with an annual probability of exceedance of $1/N$ ("recurrence period": N years) Standard recurrence periods of $N=50$ years and $N=1$ year and averaging time intervals of $t=3$ seconds and $t=10$ minutes are used.

Gust

Temporary change in the wind speed.

Hub

Fixture for attaching the blades or blade assembly to the rotor shaft.

Hub height

Z_{hub} Height of the centre of the swept area of the wind turbine rotor above the terrain surface.

Mean wind speed

Statistical mean of the instantaneous value of the wind speed averaged over a given time period which can vary from a few seconds to many years.

Reference wind speed

V_{ref} Basic parameter for wind speed used for defining wind turbine classes. Other design related climatic parameters are derived from the reference wind speed and other basic wind turbine class parameters. A turbine designed for a wind turbine class with a reference wind speed V_{ref} , is designed to withstand climates for which the extreme 10 min average wind speed with a recurrence period of 50 years at turbine height is lower than or equal to V_{ref} .

Roughness length

z_0 Extrapolated height at which the mean wind speed becomes zero if the vertical wind profile is assumed to have a logarithmic variation with height.

Reynolds number

The Reynolds number can be defined as $Re = u*L/v$ where u is velocity, L is characteristic length and v is kinematic viscosity.

 $S_1(f)$

Power spectral density function for the longitudinal wind velocity.

Turbulence intensity

I or **TI** Ratio of the wind speed standard deviation to the mean wind speed, determined from the same set of measured data samples of wind speed, and taken over a specified period of time.

Turbulence standard deviation

σ_1 Standard deviation of the longitudinal component of the turbulent wind velocity at hub height.

Turbulence scale parameter

Λ_1 Wavelength where the non-dimensional, longitudinal power spectral density is equal to 0,05.

Upwind

In the direction opposite to the main wind vector.

Wind profile

Mathematical expression for assumed wind speed variation with height above ground.

Wind shear

Variation of wind speed across a plane perpendicular to the wind direction.

Wind shear exponent / Power law exponent

α Constant used in the power law wind profile.

Wind speed at hub height

V_{hub} Wind speed at hub height.

 Z_{hub}

Hub height of the wind turbine.

Chapter 1

Introduction

It is almost universally accepted throughout the scientific community that current means of producing energy like fossil fuels are environmentally destructive and release greenhouse gases causing global warming. With renewable energy like sunlight, wind, currents and rivers surrounding us, exploring and utilizing these resources should be highly prioritized.

Wind power have been increasingly popular in later years. It produces inexpensive energy, and has little to no risks of danger to people or the environment.

Offshore wind turbines provides more energy because of higher winds and does not visually disturb the view of neighbours.

An important tool for testing and design of wind turbines is aeroelastic software. This type of software is used to see how the turbine and blades behaves under different conditions.

The most common wind on turbine sites is turbulent. On a short time scale of minutes and down to seconds, turbulence has a very significant impact on the quality of the power output to the grid as well as design and performance of individual wind turbines (Burton 2011). Most important, the ever-present turbulence makes the parts, in particular the blades of the turbine subject to material fatigue. Extreme wind speeds, which is more rare and usually less of a problem must still be taken into consideration when designing the turbine (Hau 2013).

Generation of an atmospherical turbulent windfield is not an easy task, and changes in standards are still going on. It is expected that simulation of atmospherical turbulence will be solved with computational fluid dynamics in the future when computing power increases.

1.1 What is turbulence?

We all have an idea what turbulence is. It's the fast and irregular wind we feel during a storm. The definition of turbulence remains somewhat unclear. Initially it was defined as an irregular motion in fluids. Although several key characteristics is defined, a precise definition has still not been made, and is probably not due until the phenomenon is fully understood. The characteristics of turbulence is as follows (Rosa n.d.):

- **Irregularity and unpredictability:** A turbulent flow is irregular both in space and time, displaying unpredictable, random patterns.
- **Statistical order:** From the irregularity of a turbulent motion there emerges a certain statistical order. Mean quantities and correlation are regular and predictable.
- **Wide range of active scales:** A wide range of scales of motion are active and display an irregular motion, yielding a large number of degrees of freedom.
- **Mixing and enhanced diffusivity:** The fluid particles undergo complicated and convoluted paths, causing a large mixing of different parts of fluid. This mixing significantly enhances diffusion, increasing the transport of momentum, energy, heat and other advected quantities.
- **Vortex stretching:** When a moving portion of fluid also rotates transversally to its motion an increase in speed causes it to rotate faster, a phenomenon called vortex stretching. This causes that portion of fluid to become thinner and elongated, and fold and intertwine with other such portions. This is an intrinsically three-dimensional mechanism which plays a fundamental role in turbulence and is associated with large fluctuations in the vorticity field.

Chapter 2

Use and theory of turbulent wind

2.1 Overview

The assumptions, statistics, standards and formulas behind the simulation of turbulence in Ashes will be layed out in this chapter.

The directions refered to throughout the document is the same as in the IEC, and can be explained briefly:

longitudinal - along the direction of the mean wind velocity

lateral - horizontal and normal to the longitudinal direction. Can also be refered to as crosswind.

upward - normal to both the longitudinal and lateral directions, i.e. tilted from the vertical by the mean flow Inclination angle. (IEC 2005)

2.2 How turbulence is originated

The generation of turbulence in the boundary layer is caused by friction with the earth's surface and thermal effects that cause air masses to move vertically. An example of these two forces interconnecting is when warm air is forced up by a mountain and into cooler regions where it is no longer in thermal equilibrium with its surroundings (Burton 2011).

2.3 Standards

Before 1988, nations like Denmark, the US, Germany and Sweden had their own standards developed for design and rules for wind turbines. The International Electrotechnical Commission (IEC) then took over this task on an international basis. Although the IEC standards

are widely used, they have not yet gained unrestricted legal force. National standards are sometimes used as a second opinion. (Hau 2013)

My investigation, implementation and integration of the turbulence simulating software will be based on this latest standards from the IEC. The IEC 61400-1-ed3 and the IEC 61400-3-ed1. These documents describe safety and design requirements for wind turbines onshore and offshore(the latter). (IEC 2005)

2.4 IEC Turbine classes

The IEC has divided wind conditions into classes. Engineers can then measure wind and look at historical data to determine which class the site in question belongs to. The defining variable is the *reference wind speed* or V_{ref} . This is the maximum wind speed which only is exceeded one per 50 years statistically. It is obtained by measuring the ten-minute mean value, then deriving from this value the maximum gusts to be expected. (Hau 2013)

There are 3 classes defined in the IEC 61400-1-ed3; 1, 2, and 3. In addition, the turbulence intensity of the site is classified by the classes A, B and C. The values of the classes are described in the table above. 2.1 The intention of the classes is not to give a precise representation of any specific site, but rather to cover most applications. Class 1, 2 and 3 are not often used on offshore sites. The class S is defined for use when special wind or other conditions

Wind turbine class	V_{ref}	I_{ref}
1	50	A, B or C
2	42,5	A, B or C
3	37,5	A, B or C
A	1, 2 or 3	0,16
B	1, 2 or 3	0,14
C	1, 2 or 3	0,12
S	Values specified by the designer	Values specified by the designer

Table 2.1: Turbine classes defined by IEC

2.5 Statistical properties of turbulence

There are many useful statistical properties of turbulence. The turbulence intensity is a measure of the overall level of turbulence, and how violent the gusts are.

$$I = \frac{\sigma}{\bar{U}} \tag{2.1}$$

Where σ is the standard deviation of wind speed variations and \bar{U} is the mean wind speed. Turbulent wind speeds can be considered Gaussian, but the tails of the distribution is not Gaussian. This means the occurrence of larger gusts within a time period can not be reliably be estimated using this distribution. This is illustrated well by the figures from OC4 in chapter 4.

2.5.1 Wind Profile

Another basic statistical property is the speed profile of the wind speed from the ground and up in the area of the boundary layer closest to the ground. This is known as the Prandtl layer. When the turbulence is simulated, the mean wind values for each height should follow a function also known as a wind profile. The most common profile is known as the power law, and defined in the IEC-61400-1-ED3 as:

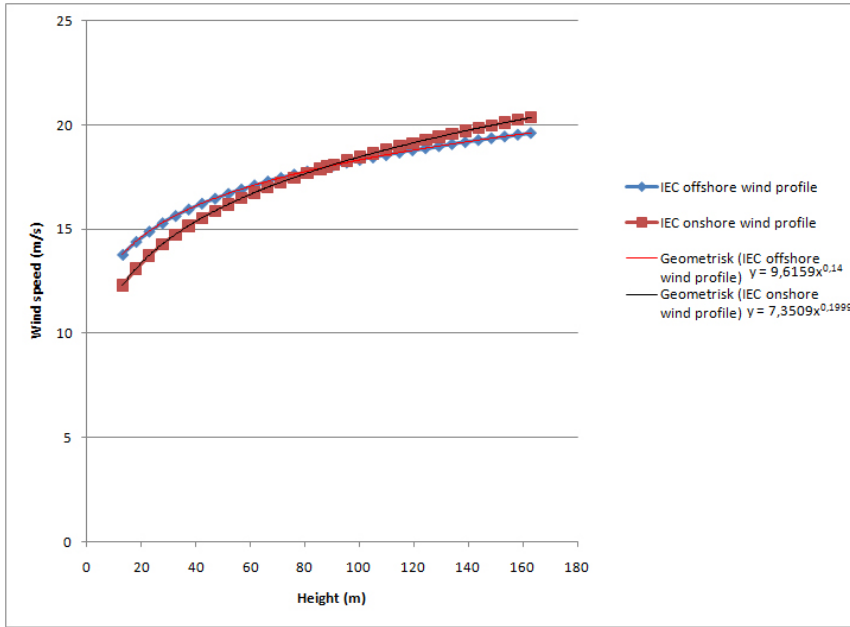
$$V(z) = V_{hub}(z/Z_{hub})^\alpha \quad (2.2)$$

The power law exponent, α , should be set so the resulting $V(z)$ function follows the measurements done on site as close as possible. If no such measurements can be obtained, α shall be assumed to be 0,2 for the normal wind turbine classes (1, 2, 3). For offshore wind turbines, α is estimated to 0,14 according to IEC 61400-3 (Offshore), but should only be used when historical wind data from the site are not available.

The power law is somewhat different under the special test EWM (Extreme wind model), where alpha is set to 0,11. All other wind cases described in IEC-61400-1-ED3 uses the normal wind profile (NWP), where alpha has a recommended value of 0,2.

An engineer who wants to simulate a specific site as correctly as possible can quite easily obtain wind speeds at different heights using an anemometer or similar tools and calculate the corresponding alpha using regression and equation 2.2. Turbsim allows the user to set the alpha to the desired value.

Figure 2.1: Comparison of wind profile for offshore and onshore turbines using IEC specifications. 88,2 m hub height and 18 m/s average wind speed at hub height.



The figure above illustrates the wind profiles proposed for both onshore and offshore wind turbines by the IEC.

2.5.2 Normal Turbulence Model

The Normal Turbulence Model, or NTM is used to describe the standard deviation of the wind in the given wind direction. From (IEC 2005):

For the normal turbulence model, the representative value of the turbulence standard deviation, σ_1 , shall be given by the 90% quantile for the given hub height wind speed. This value for the standard wind turbine classes shall be given by

$$\sigma_1 = I_{ref}(0.75 * V_{hub} + b); b = 5.6m/s \quad (2.3)$$

2.5.3 Spectral models of turbulence

The continuous fluctuations in wind speed which is defining for turbulence can best be reproduced by a statistical method also used in meteorology. The most important part of a

turbulent windfield is the longitudinal wind speed fluctuations. In most cases the longitudinal component of the wind is the only component needed. (Hau 2013)

The turbulence models discussed in this thesis will be the Kaimal(1972) spectral model and the Mann(1994) uniform shear turbulence model. These models are the only ones recommended in (IEC 2005). The Von Karman turbulence model are briefly described being a basis for the Mann model.

It's worth mentioning that IEC is not the only standard that describes these spectral models. Other standards like the Danish standard (DS 472, 1992) and Eurocode use different approximations for the length scales.

Von Karman spectral model

According to Petersen et al. (1998), the Von Karman spectrum gives a good description for turbulence in wind tunnels, although the Kaimal spectrum may give a better fit to empirical observations of atmospheric turbulence. Nevertheless the von Karman spectrum is often used for consistency with analytical expressions for the correlations (Burton 2011). An important characteristic of this model is that it produces an isotropic solution for the standard deviations for the three components. In 2005, when the third edition of the IEC-61400-1 was published, the Von Karman spectral model was removed from the suggested turbulence models. As Ashes follows this latest standard, the von Karman spectral model will not be elaborated on further in this paper. It should however be noted that the model is included in Turbsim, but currently not available in the Ashes user interface.

Kaimal spectral and exponential coherence model

The Kaimal model as defined by the IEC assumes neutral atmospheric stability (RICH_NO=0). The spectra for the wind components, K=u, v, w, are given by (Jonkman and Kilcher 2012)

$$S_K(f) = \frac{4\sigma_K^2 L_K / \bar{u}_{hub}}{(1 + 6f L_k / \bar{u}_{hub})^{5/3}} \quad (2.4)$$

where f is the cyclic frequency and L_K is an integral scale parameter.

The relationships between the standard deviations are given by (ibid.) and (IEC 2005):

$$\sigma_v = 0.8\sigma_u, \sigma_w = 0.5\sigma_u. \quad (2.5)$$

The Kaimal spectrum requires the use of length scales. Length scales are dependent on height. Proximity to the ground constrains the size of turbulent eddies and, thus, reduces the length scales. At higher altitudes the turbulence is no longer influenced by the ground and becomes isotropic. These heights are not relevant here because they are well above the height of a wind turbine. (Burton 2011).

The IEC 61400-1-ED3 standard defines the integral scale parameter to be

$$L_K = \begin{matrix} 8.10\Lambda_U & , K=u \\ 2.70\Lambda_U & , K=v \\ 0.66\Lambda_U & , K=w \end{matrix} \quad (2.6)$$

The longitudinal turbulence scale parameter Λ_U is defined as

$$\Lambda_U = \begin{matrix} 0,7z & z \leq 60m \\ 42m & z \geq 60m \end{matrix}$$

Mann uniform shear turbulence model

IEC-61400-1-ED3 recommends using the Mann(1994) uniform shear turbulence model over Kaimal. This turbulence model is based on rapid distortion theory. Although rapid distortion theory is not itself a theory of turbulence, it can be used in conjunction with turbulence theories as guide to the modelling of dynamical terms. (Hunt and Kevlahan n.d.)

Both Kaimal and the Von Karman model uses one-dimensional Fast fourier transform (FFT) to generate time histories from spectra. This makes each component an independent calculation. The Mann model does things in a different way. A three-dimensional spectrum tensor is used in combination with a three-dimensional FFT to generate all components of turbulence simultaneously (IEC 2005).

This means that the correlation between particularly the longitudinal and vertical components resulting from Reynolds stresses is included in the calculations. The spectral density for any three-dimensional wavenumber vector is derived, and all three components of turbulence are then generated simultaneously by summing a set of such wavenumber vectors, each with the appropriate amplitude and a random phase.

It is important to note that efficient numerical computation with the Mann spectral model requires the number of points in the longitudinal, lateral and vertical directions to be a power of two.

A weakness with Mann is that with a number of points that with current computers give reasonable computation time, the resulting turbulence spectra are deficient at the high frequency end (Veldkamp 2006). Mann suggests that this may be realistic, because it represents averaging of the turbulence over finite volumes of space which is appropriate for practical engineering applications. (Burton 2011)

The parameters for the Mann turbulence model are (IEC 2005):

$$\begin{matrix} \gamma = 3,9 \\ \sigma_{iso} = 0,55\sigma_1 \\ l = 0,8\Gamma_1 \end{matrix} \quad (2.7)$$

γ is the shear distortion parameter. Increased value will cause the longitudinal and lateral velocity component variances to increase while the upward velocity component variance

decreases. σ_{iso} is the unsheared, isotropic variance and l is the length scale (ibid.). Where σ_1 and l are specified by the NTM and equation (2.5.3). The equations behind the Mann turbulence model is described in detail in the IEC-61400-1-ED3. (ibid.)

The Mann model is not implemented in Turbsim. Another numerical solver, windsimu, was developed by Jakob Mann in 1999 and can in theory be used with Ashes by integrating the IEC Turbulence Simulator available at www.wasp.dk. This proved to be difficult because a lack of documentation on the software. A clear summary of the input is not provided, and the result is not correctly scaled and probably needs some post processing. This is acknowledged by Bonnie Jonkman, Senior Scientist at NWTC in a post on Turbsim's forums (*NWTC • View topic - Mann turbulence from HAWC2 using InflowWind* 2014). Further there is no mention on what the different values for turbulence models in the input file represents. A correspondence with the Technical University of Denmark where the code was developed is proposed as a future solution to clear up the questions regarding the software.

Spectral models and asymptotic limits

It is important to understand the scale and purpose of the turbulent wind field in question. Turbulence is studied from many perspectives, like how it originates, how it behaves around objects, drag and lift coefficients, multi-layer turbulence with different liquids and fully developed atmospheric turbulence like the type the software used in Ashes will simulate. Turbulence on smaller scales or in other fluids will have different length scales, different profiles and different turbulence models. This is however still an important part of current research on turbulence, and it can be useful to have a look at a theory that attempts to use the two fundamental parameters of turbulence: the rate of dissipation ϵ and the kinetic energy k to describe turbulence on all scales when the reynolds number is high.

The asymptotic limits of turbulence used in the IEC standard stems from Kolmogorov's theory of 1941. IEC standards allows for other spectra to be used, but the high frequency asymptotic behavior must tend to the following relationship to comply with the standard: (IEC 2005)

$$\frac{S_u(n)}{\sigma_u^2} n \xrightarrow{\infty} .005 \left(\frac{\Lambda_1}{U} \right)^{-2/3} n^{-5/3} \quad (2.8)$$

In edition 3, another constraint on heights above 30 m was added.

$$\frac{S_u(n)}{\sigma_u^2} n \xrightarrow{\infty} A \left(\frac{\Lambda_1}{U} \right)^{-2/3} n^{-5/3} \quad (2.9)$$

A varies with different heights and models. (Burton 2011)

The notion of turbulence has from the beginning been that the flow is composed by eddies of different sizes. Large eddies can be described with length scales, velocity scales and time scales. They eventually break up in to smaller eddies which break up in to even

smaller eddies all the way down until the viscosity in the fluid can effectively dissipate the kinetic energy into internal energy / heat (Khusnutdinova 1995). Kolmogorov's third hypothesis is

$$E(k) = C * \epsilon^{(2/3)} * k^{(-5/3)} \quad (2.10)$$

There is presently no fully deductive theory which starts from the Navier-Stokes equations and leads to the Kolmogorov's law (ibid.). However the theory has considerable experimental evidence to support it. The theory assumes that turbulence is statistically similar invariant of the scale. A question about if the constant C can be a universal value for all scales is an important area of current research on turbulence (Frisch 1995).

2.6 Extreme gusts, winds, direction changes and shears

To make sure the wind turbine can withstand all naturally occurring wind conditions, the designer must run the following special tests on the wind turbine:

- EWM50 - Extreme wind model 50 year recurrence period
- EWM01 - Extreme wind model 1 year recurrence period
- EOG - Extreme operating gust
- ETM - Extreme turbulence model
- EDC - Extreme direction change
- ECD - Extreme coherent gust with direction change
- EWS - Extreme wind shear

TurbSim only allows 3 of these: ETM, EWM50 and EWM01. Another tool, IECWind can run all the abovementioned models except ETM (Extreme turbulence model). IECWind is also produced by NREL.

2.6.1 Extreme turbulence model

The extreme turbulence model sets the standard deviation to a higher value using the following equation

$$\sigma_1 = c * I_{ref} \left(0, 072 \left(\frac{V_{avg}}{c} + 3 \right) \left(\frac{V_{hub}}{c} - 4 \right) + 10 \right) \quad (2.11)$$

It shall use 2 m/s as c and follow the normal wind profile according to the IEC (IEC 2005).

2.7 Coherence

When simulating a turbulent wind field of a given time length in a plane, coherence needs to be taken into account. A given point in the field will be influenced by the wind speed in another point, like it's closest neighbors in the grid. Actually, all the other points have some influence on the others.

2.7.1 Coherence in the Kaimal turbulence model

The coherence in Kaimal is given by the following function:

$$Coh(r, f) = exp[-12((f * r/V_{hub})^2 + (0,12r/L_C)^2)^{0,5}] \quad (2.12)$$

where $Coh(r, f)$ is described in IEC-61400-1-ED3 as

the coherence function defined by the complex magnitude of the cross spectral density of the longitudinal wind velocity components at two spatially separated points divided by the auto-spectrum function.

Coherence is a requirement for any turbulence model used when following the IEC standard (ibid.).

2.8 Turbsim

Turbsim is a stochastic, full-field, turbulent wind simulator. It is written in Fortran 90. Turbsim uses a statistical model to numerically simulate time series of three-component wind-speed vectors at points in a two-dimensional vertical rectangular grid that is fixed in space. This makes it convenient for Ashes (as opposed to a 3 dimensional solution), because all we really are interested in is how the turbulence will affect the wind turbine. Turbsim is written by researchers at the National Renewable Energy Laboratory (NREL). The code has its roots in a program called SNLWIND written by Paul Veers. This program used the Sandia method to generate full-field turbulent wind for the streamwise (u) component only. Turbsim was an extension to this program modifying it to also generate v and w components (Jonkman and Kilcher 2012). Paul Veers also wrote a paper describing the Sandia method (Veers 1988).

2.8.1 How Turbsim works

This section will briefly explain what the Turbsim code does. The Turbsim source code is open source and has been examined for this analysis. Turbsim begins with reading in all the input from the input file. After that it checks which turbulence model is used. Ashes follows the IEC standard, and will only use the Kaimal model in Turbsim, so only this

part of the code will be explained. The next thing it does is find the correct sigma for each component for the given spectral model and turbulence model. Using the IEC standard (IEC 2005), the turbulence model is NTM (Normal turbulence model) or ETM (Extreme turbulence model). Turbsim then opens the .sum file and writes the parameters given as input with the standard deviations.

After this, Turbsim generates the random numbers it needs, then goes ahead with calculating the spectral and transfer function matrices. This involves calling a function which computes the coherence between two points in the grid for all the points. This coherence should not be confused with the coherent turbulence scaling using files which include more rigorous simulations and can be downloaded separately from Turbsim. This part of Turbsim does not apply to IEC turbulence.

After this, an inverse fast fourier transform is performed to obtain the wind speeds. After the IFFT, Turbsim checks if a parameter is set to scale all the wind speeds to meet the target standard deviations and mean wind speed, and performs the scaling if the parameter is set.

At the end it writes the output files and calculates the mean wind speed across the whole grid and the turbulence intensity from the simulated data. This lets the user check if the turbulence intensity is correct according to the given input.

2.8.2 Wind speed distribution

The mean annual wind speed, understood to be the "invariable" long-term mean value of the wind speed at one location can only be determined on the basis of measurements taken over decades. (Hau 2013)

The wind speed distribution is not essential to understand how turbulence is generated, but can be beneficial for engineers to know about when dealing with wind.

The wind speed distribution should be obtained by measurements over at least several year. Meteorologists suggests up to 10 years of measurements to obtain reliable statistical data. A common problem is that insufficient data about the frequency distribution of the wind speeds at a particular location are available. In these cases a Weibull distribution can be used to provide a good approximation (ibid.). The Weibull and Rayleigh distributions are defined in (IEC 2005) as:

Reyleigh distribution:

$$P_R(V_{hub}) = 1 - \exp[-\pi(V_0/2V_{ave})^2] \quad (2.13)$$

with

$$V_{ave} = C \frac{\sqrt{\pi}}{2}$$

if $k = 2$.

and else

$$V_{ave} = C\Gamma\left(1 + \frac{1}{k}\right)$$

Weibull distribution:

$$P_W(V_{hub}) = 1 - \exp[-(V_0/C)^k] \quad (2.14)$$

where

$P(V_0)$ is the cumulative probability function, i.e. the probability that $V < V_0$. V_0 is the wind speed (limit). V_{ave} is the average value of V . C is the scale parameter of the Weibull function. k is the shape parameter of the Weibull function. Γ is the gamma function.

Chapter 3

Implementation and design

3.1 Tools

The programming was done in the C++ language in Microsoft Visual Studio 2012. On top of that, the QT-library (*Qt - Qt Product 2013*) was frequently used.

3.1.1 Qt

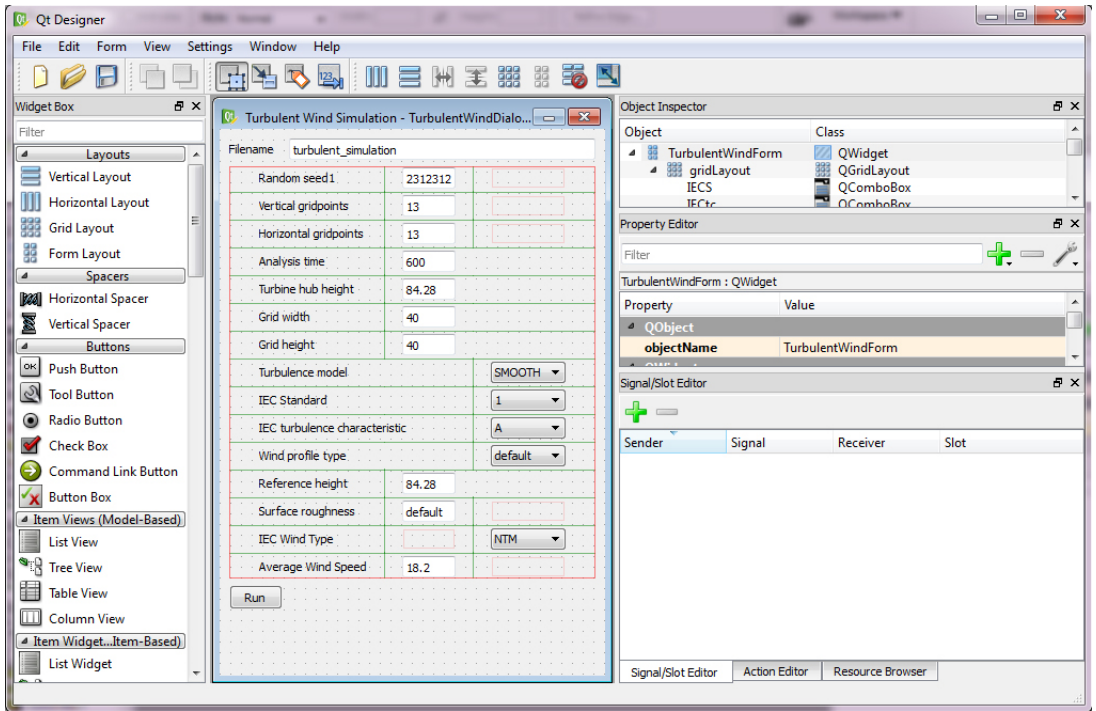
QT is a cross-platform application framework that is widely used for developing application software with a graphical user interface. In regards of software quality, using QT can increase portability. The framework works on the following platforms: Windows, Windows CE/mobile, Symbian, OS X, X11, Embedded Linux, Maemo, MeeGo, Wayland. As QT is open source, the community has also made the following ports (ibid.): QT for OpenSolaris, QT for Haiku, QT for OS/2, QT-iPhone, QT for webOS, QT for Amazon Kindle DX, Necessitas (Android), QT for BlackBerry.

The way Ashes is written, QT's classes often form a base which then is built upon. The basic classes like Qstring and Qvector provides basic functions which saves time for developers. These classes do not need a QObject as a parent. Qt also provides a framework for signals and slots.

Qt uses a parent/child tree structure, which is not the same as regular inheritance. The parent is usually passed on to any children via the constructor of the child object, but other Q-objects can be used as a parent. The good thing about this system is that it takes care of memory management automatically, in the way that when a parent is deleted from memory, the children is also deleted.

QWidget, QDialog and QMainWindow are examples of classes that makes the whole process of making a graphical user interface faster. They also relate in a hierarchical tree structure, with the same memory management mechanisms, signals and slots etc, but they

Figure 3.1: Qt Designer. Tool for creating GUI.



are also divided visually into parent/child relationships, where the children visually appears inside their parent.

QProcess was used to execute the Turbsim .exe file from within Ashes.

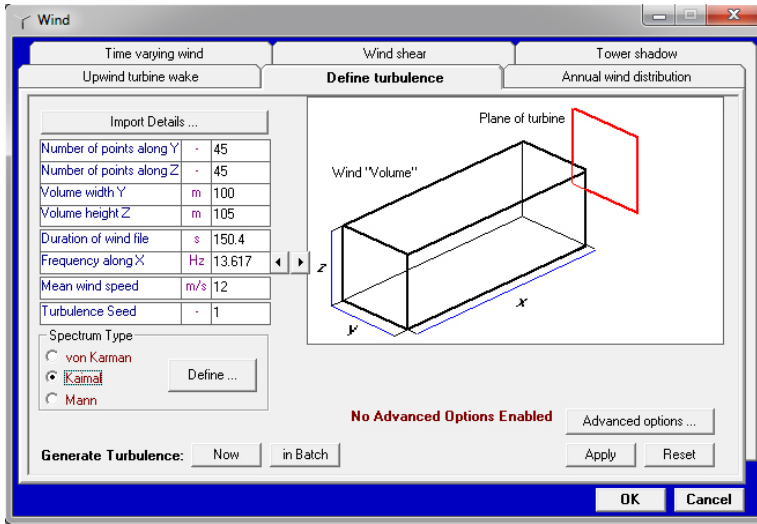
The Qt Designer is a program where the developer can draw up dialogs and input in a short time. The Qt Designer were frequently used during the development of the Turbsim/Ashes integration.

3.2 Design of the interface

Another aeroelastic software, Bladed, has a very complicated user interface intended for advanced users. One of the 6 tabs used for parameter input is shown in **Fig. 3.2**. The user should be familiar with advanced terminology describing wind and will very likely have to spend a lot of time reading documentation and taking courses. Several decisions have been made before designing the user interface of the turbulent wind simulation tool. The first major decision was to divide the user interface into a simple interface for new users, and an advanced interface for experienced users of Turbsim. The input file has 51

parameters, to make the user interface simpler, the amount of input needed from the user was shortened.

Figure 3.2: Turbulent wind user interface in Bladed. This might be overwhelming for new users.



To make the user interface as simple as possible, only the important parameters should be shown to the user. The following chapter will go through the simplifications made to the user interface. The Turbsim integration follows the IEC standard 61400-1-ED3 which makes it possible to exclude or set some parameters to a static value hidden from the user. When the turbulence model is set to IECKAI, or Kaimal, Turbsim only reads 32 parameters, simplifying the user interface. In addition, several variables have only 1 valid value for IEC turbulence, and is therefore hidden from the user.

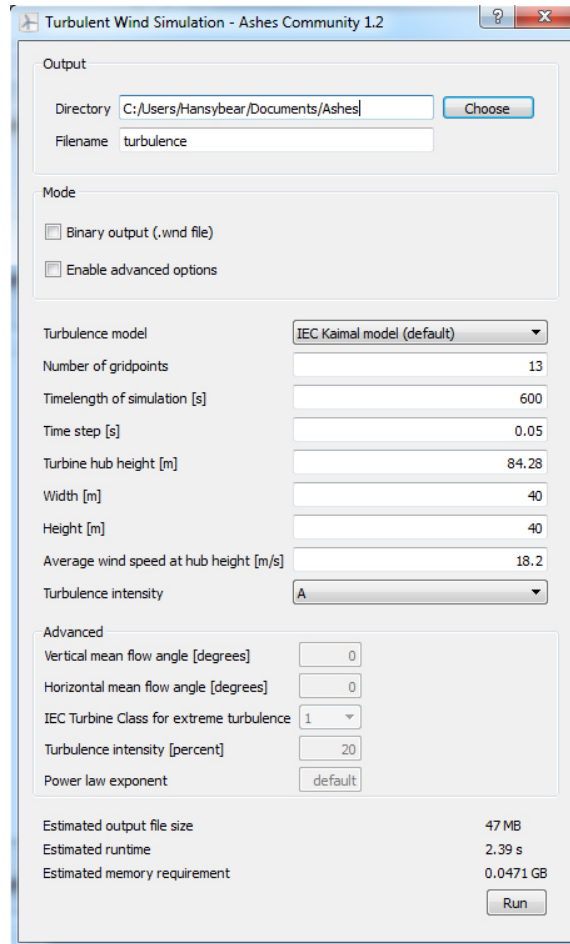
3.2.1 Runtime options

In a Turbsim input file, line 4-15 is classified under runtime options. These 12 parameters tells Turbsim what type of output file it should produce, what random seeds to use, the organization of output data (clockwise or not) and if the output should be scaled to fit the exact turbulence intensity (ScaleIEC).

The random number seeds 1 and 2 can let the user perform exactly the same simulation if these seeds are not changed. To make the user interface easier to grasp, these two seeds were both set to a random generated number and hidden from the user. This trades increased usability with a minor decrease in functionality.

The next 8 parameters are boolean values (True or False) which tells Turbsim what output files to generate. Producing only the wind data at hub height were investigated in chapter 5 and found to have the same runtime as when producing the full field output. In addition,

Figure 3.3: Turbulent wind user interface in Ashes.



the whole field is required to test the whole sweeping area under turbulent winds. The user is left with two options, outputting a single binary file or 3 readable files, one for each component.

3.2.2 Turbine/Model specifications

Line 18-27 in the input file has data about the geometry of the wind field. These parameters should generally be available for the user to edit. The resolution of the grid lets the user decide how detailed the turbulent field output should be. Although both appear white in **Fig. 3.6**, only one input is used for the number of gridpoints in each direction, because a rectangular grid would not bring much useful data to the application. The user can also

Figure 3.4: Runtime options. Blue fields are hidden from the user and set to default values. Yellow fields are available in advanced mode. Fields with a red line through are not available through the user interface. Fields with no added color are enabled for the user.

```

3 -----Runtime Options-----
4 2318573          RandSeed1
5 RANLUX          RandSeed2
6 False          WrDHHTP
7 False          WrFHHTP
8 False          WrADHH
9 False          WrADFF
10 False          WrBLFF
11 False          WrADTWR
12 True           WrFMTFF
13 False          WrACT
14 True           Clockwise
15 2              ScaleIEC

```

Figure 3.5: Options about the turbine/model. Blue fields are hidden from the user and set to default values. Yellow fields are available in advanced mode. Fields with a red line through are not available through the user interface. Fields with no added color are enabled for the user.

```

17 -----Turbine/Model Specifications
18 45              NumGrid_Z
19 45              NumGrid_Y
20 0.05           TimeStep
21 600            AnalysisTime
22 600            UsableTime
23 84.2876        HubHt
24 140.00         GridHeight
25 140.00         GridWidth
26 0              VFlowAng
27 0              HFlowAng

```

change the size of the grid, angle of mean wind, the height of the turbine, the duration or timelength of the simulation and the length of each timestep.

3.2.3 Meteorological boundary conditions

Line 30-40 describes the conditions of the site. The turbulence model is set to the Kaimal model to comply with IEC 61400-1-ED3. The user will not be able to change the input to the other turbulence models available in Turbsim. The Kaimal model is the only turbulence

Figure 3.6: Options about the meteorological boundary conditions. Blue fields are hidden from the user and set to default values. Yellow fields are available in advanced mode. Fields with a red line through are not available through the user interface. Fields with no added color are enabled for the user.

```
29 -----Meteorological Boundary Co
30 "IECKAI"          TurbModel
31 "1-ED3"          IECstandard
32 "A"              IECturbc
33 "NTM"            IEC_WindType
34 default        ETMc
35 IEC              WindProfileType
36 84.2876          RefHt
37 10               URef
38 default        ZJetMax
39 default          PLExp
40 default        Z0
```

model available in Turbsim which complies with IEC 61400-1-ED3. The IEC standard is set to 1-ED3, which is the newest edition of the IEC 61400-1. This effects a constant, the turbulence scale parameter in the Kaimal model which was updated in the latest edition. The turbulence characteristic describes the intensity of the turbulence. When it comes to the mean wind speed at hub height and the turbulence intensity, the user has the choice to use the classifications described in IEC-61400-1-ED3 or use custom values in advanced mode. The wind profile is set to IEC and is hidden from the user. The user can change the power law exponent of the wind profile in advanced mode. The reference height is set by the code to the same as the hub height. This means that the mean wind speed defined as a class (1, 2 or 3) in simple mode or a numerical value in advanced mode will be the mean wind speed at the center of the hub. Surface roughness was found to have no impact on the generation as long as the wind profile is set to PL (Power Law).

3.2.4 Extreme turbulence

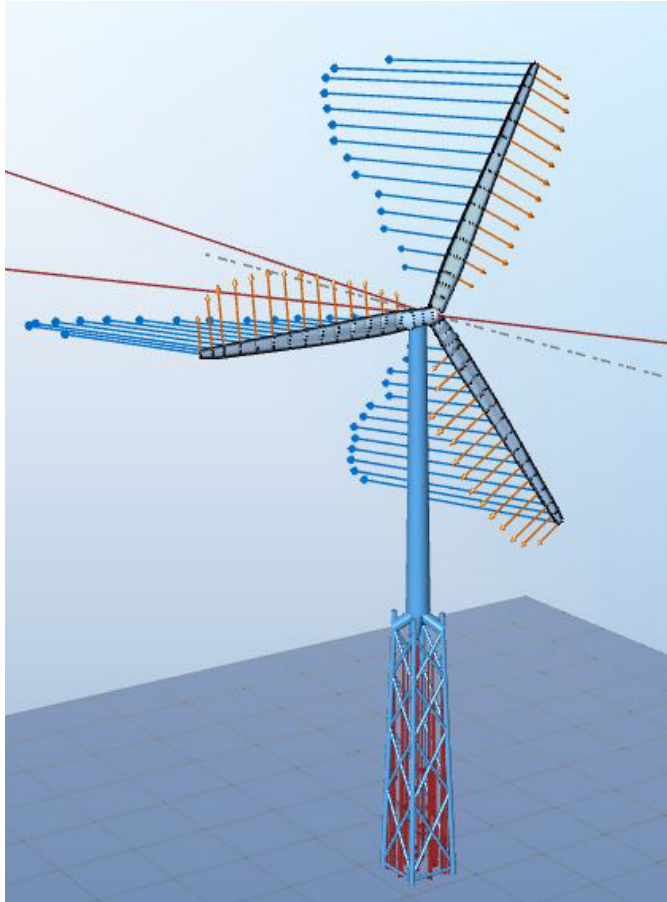
To comply with IEC specifications, the user is also able to perform a simulation of extreme turbulence with the Extreme turbulence model (ETM) described in section 6.3.2.3 of (IEC 2005). This model scales the standard deviation, or turbulence intensity using equation 2.11.

The constant c in the equation is specified to be 2 m/s by the standard (ibid.). This is a modifiable parameter in Turbsim, but is not available for the user in the turbulent wind dialog in Ashes. It is simply set to 2 by the program.

3.3 Visualization

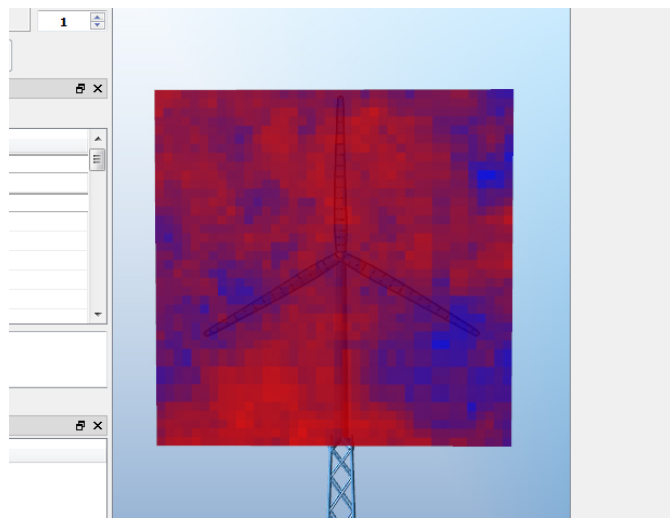
A visualization of the field was developed and can be seen in **Fig. 3.8**. This visualization was however removed because of a new method of showing the forces on the wind turbine developed by Per Ivar Bruheim. This solution is illustrated in **Fig. 3.7**.

Figure 3.7: New visualization of the wind forces in Ashes.



Arrows on the wind turbine shows the forces acting on the elements in real time. This had a cleaner look than a two dimensional color chart, and is proposed as the best way of visualizing the wind forces.

Figure 3.8: Earlier visualization of turbulent wind field in Ashes.



Chapter 4

Testing

4.1 Verification and comparison

Using Turbsim in relation with the IEC standard, the user is constrained to using just one of the many spectral models, namely the Kaimal (1972) spectrum and exponential coherence model. To make sure the calculations are correct, the data produced with Turbsim in Ashes was compared to other results in the Offshore Code Comparison Collaboration Continuation (OC4).

4.1.1 Specifications for OC4 tests

The Turbulent wind datasets generated by Risø in the OC4 report had a grid resolution of 32 x 32 points. This was used in Turbsim as well. The grid size was set to 150 m x 150 m, the size of a fairly large turbine. The dimensions result in a distance of 4,6875 m between the grid points. The simulation time was set to 3600 seconds and the time step was set to 0,1 seconds according to the specifications (Worpahl and Popko 2013). Out of 17 load cases, case 3.4a is most relevant for the generation of turbulence:

V_{hub}	σ_x	σ_y	σ_z	L_C	Wind shear (α)	Hub height
11,4 m/s	1,68 m/s	1,34 m/s	0,84 m/s	340,20 m	0,14	88,15

Table 4.1: Wind condition parameters for load case 3.4a.

In this test the α is set to 0,14 which is the suggested value for offshore wind turbines. A lower alpha value will produce lower winds above hub height and higher winds below hub height compared with a higher alpha value. A probability density function was produced and added with the other data from OC4.

4.1.2 Comparison of data for load case 3.4a

Figure 4.1: Comparison of probability density functions of longitudinal wind speed (OC4)



Figure 4.2: Comparison of probability density functions of longitudinal wind speed (OC4)

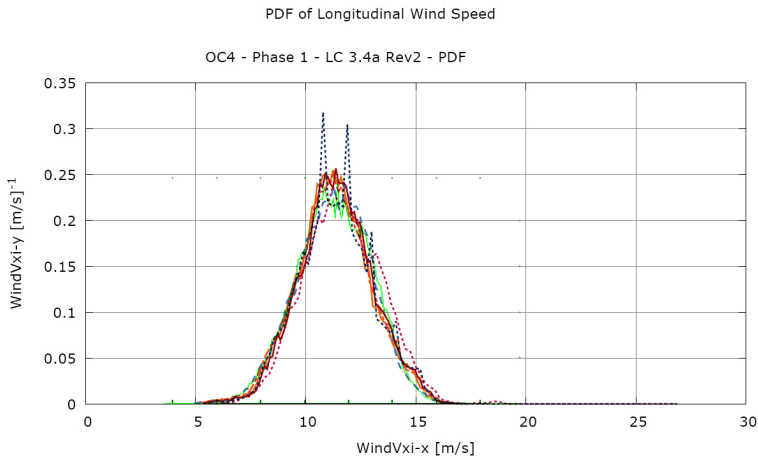
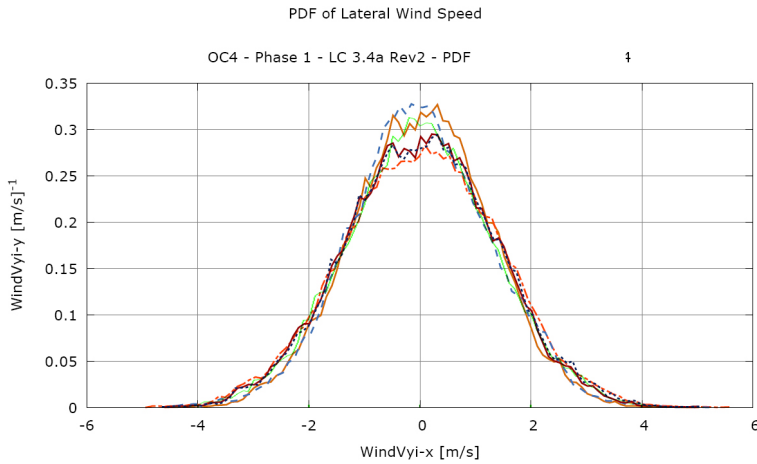
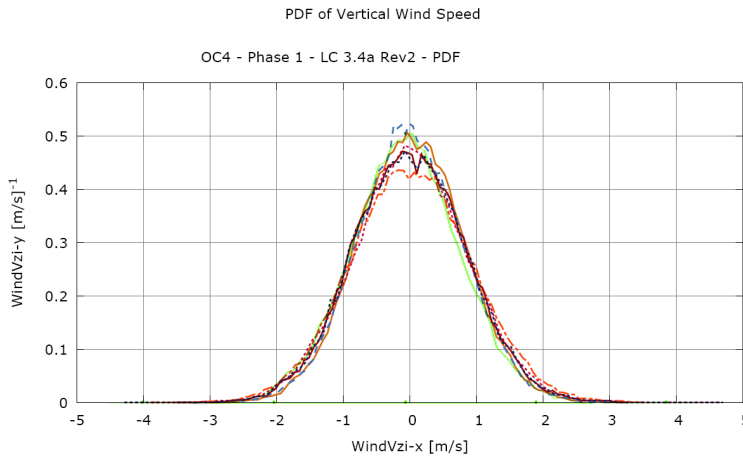


Figure 4.3: Comparison of probability density functions of lateral wind speed (OC4)**Figure 4.4:** Comparison of probability density functions of vertical wind speed (OC4)

4.1.3 Black box tests

Black box testing is where the tester can operate without knowledge of the underlying code. Testing different settings and functionality in a program is black box testing. This section contains a selection of basic tests of the integration with Turbsim in Ashes.

Nr.	1
Test description	Test running Turbsim from Ashes to see that it is set up correctly.
Input	Different input in the Turbsim input dialog. Found in Tools->Turbulent wind.
Expected output	Execution of turbsim normally through the process window.
Actual output	Execution of turbsim normally through the process window.
Comments	The program performed as expected.

Nr.	2
Test description	Test to see if the user interface updates the estimations on memory usage correctly.
Input	Kaimal model. 30 x 30 gridpoints. 1200 seconds runtime with 0,05 seconds timestep.
Expected output	Using the formulas from 5.1.10 the estimated memory requirement is 0,49 GB. With an increase of 10% for sub-optimal memory allocation described in the same chapter, the memory approximation is 0,54 GB.
Actual output	0,54 GB.
Comments	The program performed as expected.

Nr.	3
Test description	Test to see if the user interface updates the estimations on file size correctly.
Input	Kaimal model. 45 x 45 gridpoints. 600 seconds runtime with 0,05 seconds timestep.
Expected output	Using the formulas from 5.1.8 the estimated output file size for binary output is 139 MB. The output file size is slightly larger, because the formula used don't include the header in the file, just the data.
Actual output	139 MB.
Comments	The program performed as expected. Header part of the output files are variable and makes up only a small part and are not included in the prediction of output file size.

Nr.	4
Test description	Test to see if the user interface updates the estimations on file size correctly.
Input	Kaimal model. 45 x 45 gridpoints. 600 seconds runtime with 0,05 seconds timestep.
Expected output	Using the formulas from 5.1.8 the estimated output file size for binary output is 139 MB. The actual output file size is slightly larger at 145,7 MB because the formula used don't include the header in the file, just the wind speeds.
Actual output	139 MB.
Comments	The program performed as expected. Header part of the output files are variable and makes up only a small part and are not included in the prediction of output file size.

Nr.	5
Test description	Test to see if Turbsim produces the correct standard deviations for all the components.
Input	Kaimal model. 45 x 45 gridpoints. 600 seconds runtime with 0,05 seconds timestep. 10 m/s wind speed with I_{ref} , or the reference turbulence intensity at 0,16. ScaleIEC set to 2. Scaling for the whole wind field.
Expected output	Using equation for the Normal turbulence model the standard deviation for the longitudinal component or u is 2,096. Continuing with the definitions for crosswise and vertical standard deviations in the Kaimal model, the expected deviation for the v component (crosswise) was 1,677 and 1,048 for the vertical component w.
Actual output	2,096 for all points in the longitudinal direction. 1,677 for all points in the transverse or crosswise direction (v component). 1,048 for all points in the vertical component w.
Comments	The program performed as expected. The deviations were according to IEC specifications.

Nr.	6
Test description	Test to see if Turbsim produces the specified mean wind speed for the hub grid point.
Input	Kaimal model. 30 x 30 gridpoints. 1200 seconds runtime with 0,05 seconds timestep. 10 m/s wind speed for hub height.
Expected output	The expected mean wind speed for the hub grid points is 10 m/s as defined in the input.
Actual output	10 m/s.
Comments	The program performed as expected.

Nr.	6
Test description	Test to make sure the parameter z0, or surface roughness does not have any impact on power law wind profiles.
Input	Two input files were created. Both with the same random seeds. Kaimal model. 20 x 20 gridpoints. Z0 was set to 0,01 in one and z0 as 100 in the other input file.
Expected output	Exactly the same output for the two simulations.
Actual output	Exactly the same output for the two simulations.
Comments	The program performed as expected and z0 is not used in any calculations with the power law wind profile.

Validation of Extreme turbulence model

To see if Turbsim produces the correct output for the Extreme turbulence model or "ETM", a simulation using the standard input of table 5.1 with an alteration of the turbulence model from "NTM" to "1ETM" and the ScaleIEC parameter set to 2 was performed. The output .sum file was then inspected. Assigning the variables in equation 2.11 to the corresponding input in the simulation, the target sigma was calculated and compared to the result computed by Turbsim. Note that only the longitudinal component is subject to alteration by the equation. The sigmas of the other components are calculated from the longitudinal sigma with constants described in 2.

$$c = 2m/s \quad (4.1)$$

$$I_{ref} = 0,16 \quad (4.2)$$

$$V_{avg} = 10m/s \quad (4.3)$$

$$V_{hub} = 10m/s \quad (4.4)$$

$$\sigma_1 = 2 * 0,16 * (0,072 * (\frac{10m/s}{2} + 3m/s)(\frac{10m/s}{2} - 4m/s) + 10m/s) \quad (4.5)$$

$$\sigma_1 = 3,384m/s \quad (4.6)$$

This result is found to be the exact same as the sigma for the longitudinal component generated by Turbsim.

Chapter 5

Performance

5.1 Turbsim parameter study

To investigate the impact on CPU time, memory requirement and output file size when changing input, a series of parameter tests was performed. The purpose of this investigation is to get a good overview on the performance and limitations of Turbsim and answer the following questions.

- How does CPU time vary with different computers?
- How does CPU time vary with no parameter change?
- How many gridpoints can be calculated within reasonable time?
- Is there a limit to the amount of gridpoints so that the program don't start?
- How is CPU time dependent on the different parameters?
- Is Turbsim written to make use of multiple CPU cores?
- Can a minimum hardware requirement be established?

5.1.1 Input

Throughout the study different parameters was changed while the rest was kept steady. Unless mentioned, the input is set to the default values described in table 5.1.

5.1.2 Equipment

The measurements were done with the following two computers:

Parameter	Value
Random seed	2318573
File output	Readable full field u,v,w files
ScaleIEC	1
Gridpoints in horizontal direction	20
Gridpoints in vertical direction	20
Timestep	0,05 s
Timelength	600 s
Hubheight	84,2876 m
Width	140 m
Height	140 m
Flow angle	0 degrees vertically and horizontally
Turbulence model	IEC Kaimal following IEC 1-ED3
IEC Turbulence characteristic	A
Windtype	Normal turbulence model
Wind profile type	default
Mean total wind speed at hubheight	10 m/s

Table 5.1: Default values for the cases.

- HP Laptop with Intel i7 Q720 @ 1.60 GHZ and 4 GB RAM running Windows 7 (64bit) referred to as the laptop
- Stationary with Intel i7-3770K @ 3.50 GHZ and 8 GB RAM running Windows 7 (64bit) referred to as the stationary

The first question was investigated with a quick look at variation with each run on the two computers. The input file consisted of a 10 point squared grid. The turbulence were set to standard IEC kaimal. Output produced were set to readable u,v,w files. The simulation length were set to 10 minutes with 0,05 seconds intervals.

Laptop	Stationary
5,3664	0,96721
5,1012	1,0296
5,304	0,95161
6,0372	0,96721
5,0232	0,96721
4,8984	0,96721
6,0372	0,96721
5,0544	0,96721
5,0856	0,98281
5,8968	0,96721

Table 5.2: Variation in runtime with no input change.

The result in table 5.1 shows that the newer, faster more steady computer has a significantly lower variation with each test. The standard deviation on the laptop were 0,42 seconds of

calculation time, or 7,81%.

Meanwhile, the stationary gave the same calculation time on several runs, and had a lower standard deviation of around 0,02 seconds or 2,05%.

To minimize noise on the other tests, the stationary was selected as the primary testing computer. Another interesting comparison between the two computers is how much faster the stationary performs, over 12 times faster on the 50x50 grid. The data also suggests that the more gridpoints the simulation has, the benefit of using the faster computer also grows. In comparison, on a 20x20 grid the stationary is only 5,4 times faster.

Figure 5.1: Comparison of laptop and stationary. This figure illustrates the variance of the runtime.

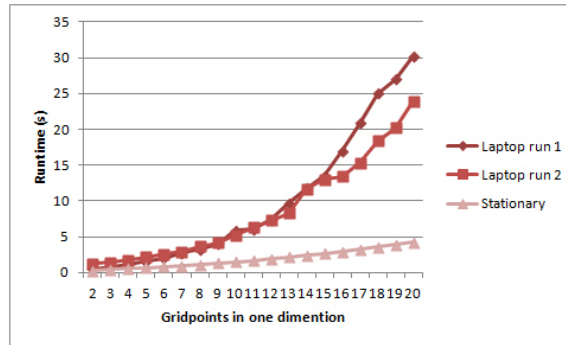


Figure 5.2: Comparison of laptop and stationary. This figure illustrates how the runtime develops with higher finer grids.

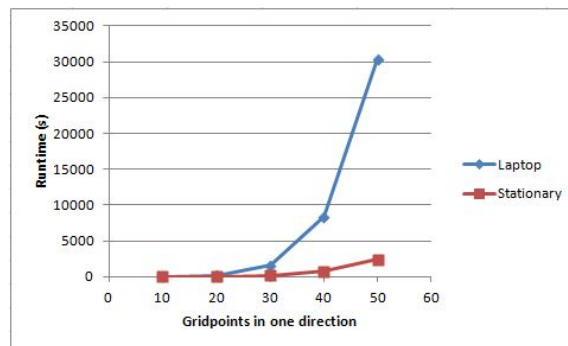


Figure **Fig. 5.1** is added to show the difference in noise on the two different computers.

5.1.3 Increase in runtime with number of gridpoints

In the tests, two measurements were performed and the average were used in the analysis to minimize noise.

In Ashes, only a square grid should be necessary to generate. A study where the number of points both vertically and horizontally were increased by 1 for each step were performed. This was done to see how runtime increases as the resolution of the grid increases. The input file is set to IEC Kaimal turbulence with 10 min length and 0,05 s as the time step. The output generated is readable u,v,w-files.

Figure 5.3: Runtime as function of gridpoints from 25 to 400 gridpoints.

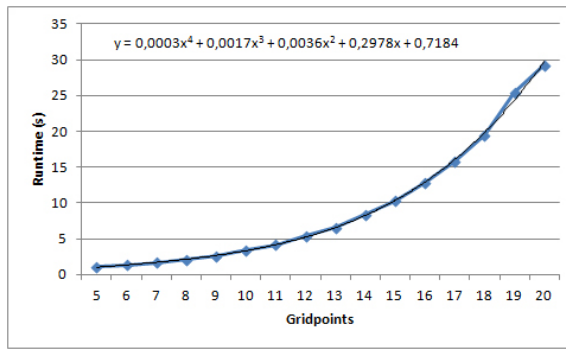
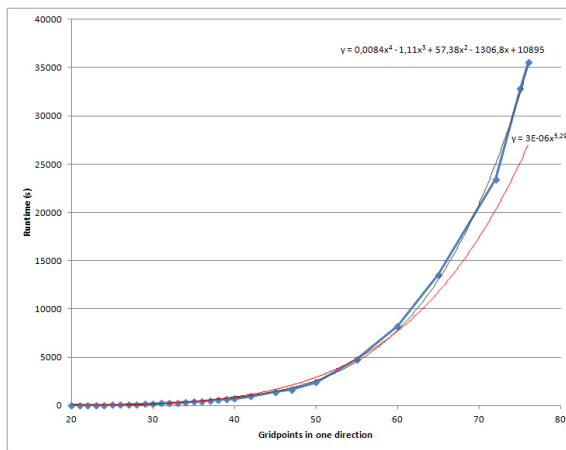


Figure 5.4: Runtime as function of gridpoints from 400 to 5184 gridpoints with curves.



It's clear from both the measurements and reasoning that the number of points in total is the biggest factor in runtime. When we look at the runtime as function of gridpoints in **Fig. 5.3** and **Fig. 5.4**, the growth looks to be smooth. This makes it more likely that the

runtime can be predicted by a function. Functions was rendered in **Fig. 5.3** and **Fig. 5.4** with the curve fit tool in Microsoft Excel. **Fig. 5.3** shows the lower grid resolutions of 5x5 to 20x20 while **Fig. 5.4** captures the bigger grid resolutions by showing grid resolutions of 20x20 up to 76x76 gridpoints.

Fig. 5.3 shows how the runtime grows with a polynomial 4th degree function. An exponential function did not fit the data well. The polynomial function used in **Fig. 5.3** did not fit well when the cases with more than 40x40 gridpoints was added. **Fig. 5.4** shows the data with both an exponential curve and a polynomial curve. In this case an exponential function was a better fit, but the polynomial curve seems to follow the data points better. Note that this polynomial function is not the same as in **Fig. 5.3**.

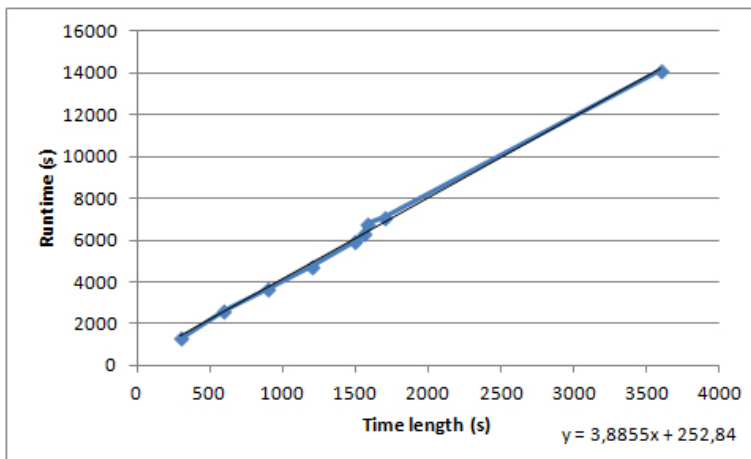
5.1.4 Runtime variation with different timelength input.

To see if the runtime develops different when keeping the grid resolution steady at 50x50 points and only changing timelength, a series of tests was performed.

Timelength	Runtime in seconds
300	1303,5
600	2631,4
900	3690,5
1200	4729,4
1500	5938,8
1560	6299,3
1700	7060,9

Table 5.3: Variation in runtime with different timelengths.

Figure 5.5: Runtime as function of timelength.



The data plotted in **Fig. 5.5** suggests a linear relationship following the function noted in the corner. This makes room for an interesting comparison. If the user increases the time-length of the simulation so that the total number of gridpoints n increases up to the point where the memory limit is, the runtime can be significantly lower than if the number of gridpoints is increased until the same number n and memory limit is reached. This means the user should try and keep the number of gridpoints low, while the timelength can be increased with a lesser impact on runtime. In addition, the total number of points is a good measurement on memory requirements, but should not be used for runtime calculations.

5.1.5 64bit CPU performance

Turbsim now comes with both a 32bit and a 64bit version. The 64bit was compared to the 32bit to see if it performed simulations faster.

64bit	32bit	Improvement in runtime
161,48	165,75	3%
170,38	165,63	-3%
186,62	194,28	4%
190,76	196,55	3%
166,56	181,32	9%
182,69	178,03	-3%
176	180,79	3%
181,94	182,69	0%
176,5	183,66	4%
184,94	184,85	0%
185,34	195,34	5%
213,92	214,66	0%
210,66	217,76	3%
254,02	262,67	3%

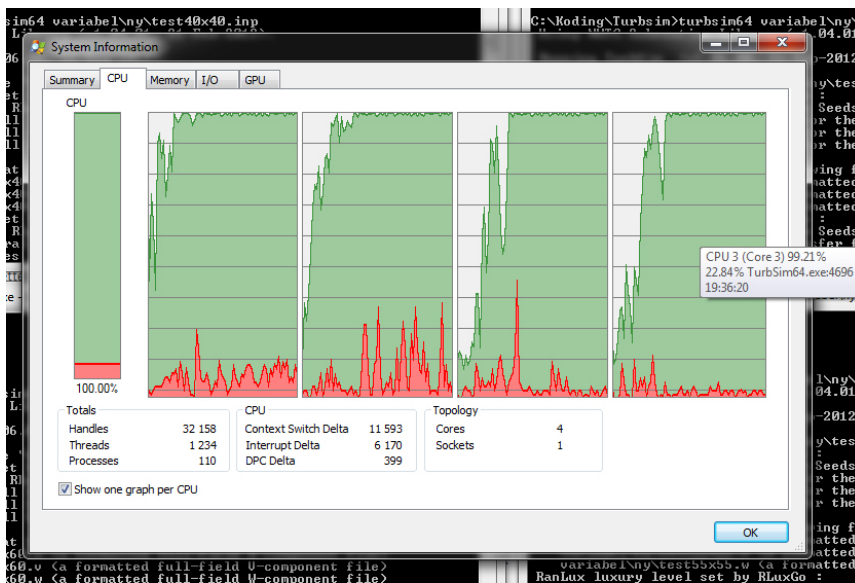
Table 5.4: Variation in runtime between 32bit and 64bit Turbsim.

From the data presented in table 5.4 it is hard to draw a definitive conclusion. On average the 64bit performs slightly faster, but the difference in runtime is not big.

5.1.6 CPU usage

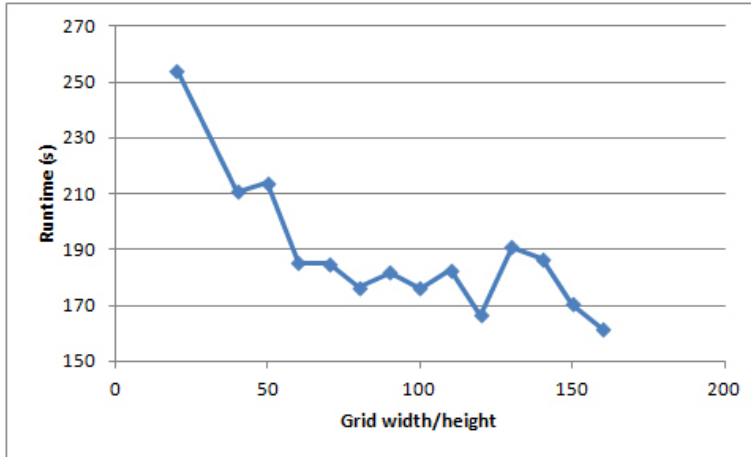
While the tests were running, the software "Process explorer" (*Process Explorer* 2013) was used to analyze the CPU load. The stationary computer has 4 cores and 8 logical CPUs with hyper threading. Hyper threading allows each physical core to be divided in two logical cores. Each thread-load were stable at around 12,3% to 12,4% during runtime. The reason it is not higher is that Turbsim is not written to work in parallel. It becomes clearer when we divide 100% with 8, resulting in 12,5%. One instance of Turbsim uses almost 100% of one logical CPU, which is reasonable. The question of why the process explorer showed the CPU use distributed around the 8 CPU's and not just 1 CPU working at 100% was investigated. Windows will receive a task from Turbsim one at a time since it is a single thread. It will then distribute the task to a CPU with the lowest load. In a parallel program, multiple threads are used, such that windows receives 8 tasks and distributes them evenly across the CPU's. To support these findings, 4 instances of Turbsim were executed simultaneously. I also turned off hyper threading on my computer so that only the 4 physical CPU cores was available. With process explorer, the CPU use of each core were stable at almost 100% CPU load also shown in **Fig. 5.6**. Each Turbsim instance were stable at around 22% CPU load. This translates to 88% of total CPU power used between the 4 instances of Turbsim. The decrease in CPU efficiency as opposed to 99,2% when only once instance ran using 12,4% of the total when the system had 8 logical CPUs can be explained by higher load on the processor by the 4 Turbsim instances with other processes on the computer also needing resources, forcing the Turbsim instances to share.

Figure 5.6: 4 Turbsim processes.



5.1.7 Runtime variation with different distances between gridpoints

Figure 5.7: Runtime as function of total width/height of the windfield.



To explore if the distance between the gridpoints have any impact on runtime, the number of gridpoints was kept steady at 30x30, while the width and height of the wind field was increased from 20 to 160. The data is presented in **fig. 5.7**.

The data suggests that longer distance between the points will give a shorter runtime. The data

5.1.8 Output files

Turbsim offers several choices in output file formats. A comparison was made to see the difference in output size and runtime. A test was performed where the number of gridpoints were increased by 1 from 10x10 to 20x20. The filetypes marked with "HH" only outputs data for the point in the center of the hub. The total number of points in the whole timelength, n can be given as

$$n = \frac{v \times h \times t}{dt} \tag{5.1}$$

where v is the number of vertical points, h is the number of horizontal points, t is the full time of the simulation and dt is the timestep. This formula can be used to get an approximation of the output file size, as the main part of the output files is the components in each gridpoint.

Figure 5.8: Runtime as function of gridpoints. Comparison between output format.

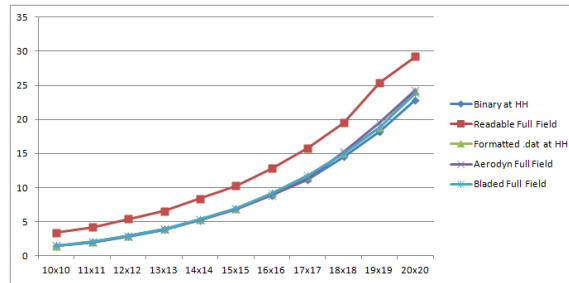
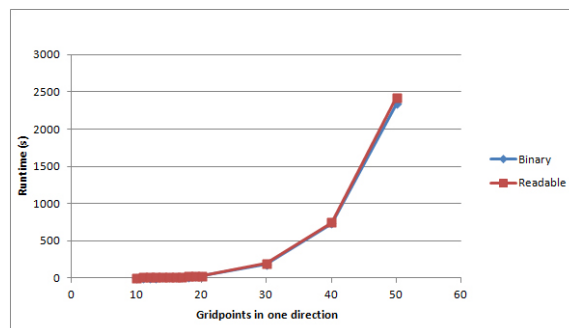


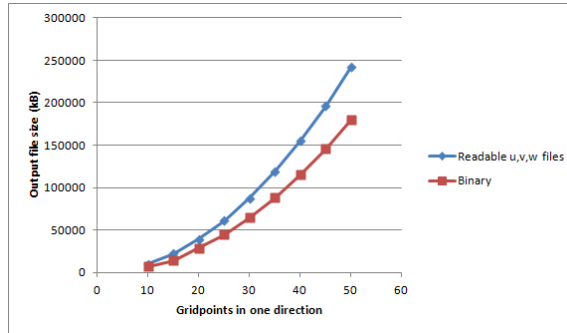
Figure 5.9: Runtime as function of gridpoints. Comparison between output format with higher grid resolutions.



The results in **Fig. 5.8** shows that the readable file format with u,v and w files causes a

slight increase in runtime compared to the other filetypes. **Fig. 5.9** reveals that the slight increase in runtime becomes only a fraction of the total runtime as the number of gridpoints gets larger. This leads to the probable conclusion that Turbsim with readable file format as output will cause an increase in runtime where it will spend more time writing the output files, but as this extra runtime only grows with filesize, it becomes negligible to the time used by the main Turbsim process: the calculation.

Figure 5.10: Output file size as function of number of gridpoints.



It becomes clear from the data presented in **Fig. 5.8** that a binary file format like the aerodyn .bts and the bladed .wnd files are preferable when it comes to the output size. Small output size means that the program will perform faster when reading in the output for visualisation and animation.

5.1.9 Other factors in runtime

A test to see if other parameters had a considerable impact on runtime were made. The first one that was considered a potential factor was the `scaleIEC` parameter. This parameter tells Turbsim to scale the distribution so that it fits the classes of turbulence intensity specified in (IEC 2005). For numerical reasons, the turbulence intensity without this scaling becomes slightly less than the specified value. There is 3 valid input values: 0, 1 and 2. 0 tells Turbsim that it should not scale the time series. 1 is "scaling by HH value". This means that all time series will be modified, using the same scaling factor for each point, the 3 components still has separate scales. The hub height will have the exact specified turbulence intensity while other points will not. 2 will tell Turbsim to scale each point individually making all the points match the exact standard deviation specified for the given turbulence intensity.

Scale parameter	Run 1	Run 2	Run 3	Run 4	Average
0	28,548	28,517	28,517	29,609	28,79775
1	31,341	29,016	28,517	28,579	29,36325
2	33,041	29,297	29,437	29,5	30,31875

Table 5.5: Variation in runtime with different Scale IEC parameters.

The measurements suggests a slight increase in runtime with the scaling set to 1 or 2. This is no suprise as the operation of multiplying only the hub height points with a constant for option 1 or multiplying every point with a constand as in option 2 will take some extra runtime.

The surface roughness `z0` was also tested, but found to have no impact on either output nor runtime because it's not

5.1.10 Memory requirements

The new stationary computer would not run 10 min simulations over 80x80 gridpoints, and Turbsim gives an error before exiting, telling the user that it can not locate the necessary amount of memory. The reason for this is that 32-bit applications in windows will have a memory limit of 2 GB (*Memory Limits for Windows Releases (Windows) 2013*). A central person in the writing of Turbsim, Marshall Buhl, explained in a forum post how one can calculate the amount of memory needed for each simulation.

$$SpectralGridElements = (NYGrids * NZGrids)^2 \quad (5.2)$$

$$SpectralMatrixElements = 3 * NZgrids * NumTimeSteps \quad (5.3)$$

$$ComponentElements = SpectralMatrixElements * NYGrids \quad (5.4)$$

$$\begin{aligned} TotalElements &= SpectralGridElements + SpectralMatrixElements \\ &+ 2 * (ComponentElements + NZGrids) + NYGrids \end{aligned} \quad (5.5)$$

$$BytesRequired = 4 * TotalElements \quad (5.6)$$

$$GBytesRequired = \frac{BytesRequired}{(1024)^3} \quad (5.7)$$

To see how much memory the 72x72 grid simulation needs, we can use the parameters as input in these equations.

$$\begin{aligned} SpectralGridElements &= (72 * 72)^2 = 26873856 \\ SpectralMatrixElements &= 3 * 72 * (600/0,05) = 2592000 \\ ComponentElements &= 2592000 * 72 = 186624000 \\ Totalelements &= 26873856 + 2592000 + 2 * (186624000 + 72) + 72 = 402714072 \\ Bytesrequired &= 4 * 402714072 = 1610856288 \\ Gigbytesrequired &= 1610856288 / (1024^3) = 1,5 \end{aligned}$$

A small addition to these formulas were made by Bonnie Jonkman in the same forum thread. Jonkman is also a contributor to Turbsim working at NWTC. The time specified by the user differs slightly from the time Turbsim needs for the simulation.

$$Turbsimtime = Specifiedtime + (gridwidth/windspeedathubheight) \quad (5.8)$$

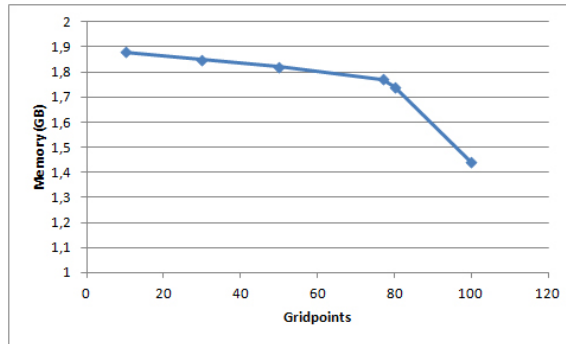
This will change the 600 seconds of input time to 609 seconds in the previous example resulting in a small increase in 0,02 GB extra memory. So with these calculations for memory use, an analysis of memory use in Turbsim was performed.

A input with 77x77 gridpoints and a 600 second timelength with 0,05 seconds in timestep size were saved to an input file. Turbsim would not run with this input because of the memory limit, although the same input with a 76x76 grid ran ok. With the same approach as the previous example, the calculated memory requirement requirement for the 77x77 grid case is 1,77 GB. This is well below the 2GB limit, and the reason for this is that memory allocation is not very efficient unless the chunks to be allocated are very small. The large arrays in Turbsim: coherence, velocity spectra and velocity components will cause the program to go over the limit well before 2 GB.

Further, several tests where the grid was kept steady at 50x50 points and the time increased from 1200 was performed. The timestep was 0,05 s. The first test that failed because of the memory limit was at 1580 seconds in timelength. This test was calculated to use 1,82 GB of ram. After this test, further tests was performed to see if a smaller grid with a long timelength was getting closer to the 2GB limit. An inputfile with 30x30 points and a 0,05 s timestep was found to run up to a timelength of 4470 seconds. This input is calculated

to need 1,85 GB of memory. Investigating this further, an inputfile with 10x10 gridpoints and a 0,05 timestep was found to run up to a timelength of 40100 seconds, requiring 1,88 GB of memory. To describe the possible relationship with number of gridpoints and the memory limit, these data were plotted in a diagram.

Figure 5.11: Maximum memory allocation as function of gridpoints.



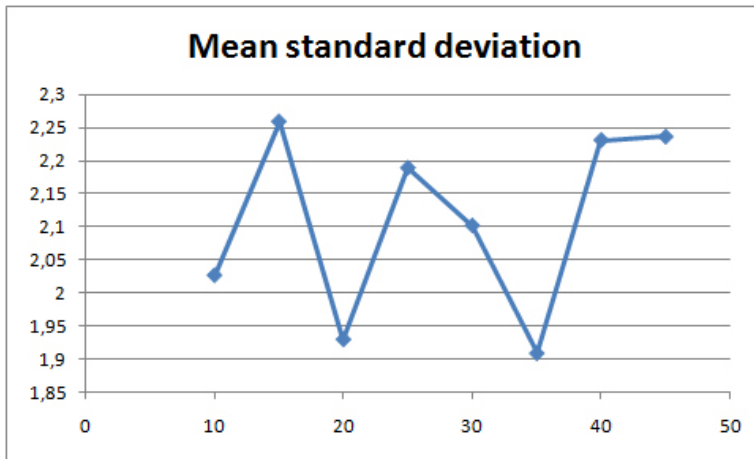
Turbsim also includes Turbsim64. To run this program, the user has to have a 64-bit operating system. This is a win64 application, which has a memory limit of 8 TB. The stationary computer used to test only had 8 GB of physical memory, so to see if the physical memory was the upper limit, a test was performed where the input was set to require 8 GB, and moving gradually down with the amount of gridpoints until Turbsim could run without the memory error. The first successful execution was a 143x143 point grid with the usual 600 second timelength and 0,05 timestep requiring about 7,31 GB.

5.1.11 Validation

A closer inspection of the output files was performed to see if the output was any different when the various input parameters was changed. Each simulation in Turbsim outputs a .sum file which is a summary of the statistics of the total output.

The standard deviation of all points during the whole timelength was plotted in **Fig. 5.12**. It is clear how it varies because of what the authors refer to as numerical reasons (Jonkman and Kilcher 2012). The tests was performed with the scale IEC parameter set to 1, which means that only the standard deviation at hub height is scaled to fit the exact turbulence intensity. When the scale IEC was set to 2, the standard deviations for all the points were the same for the different simulations, at 2,096 which is correct for the turbulence intensity input: 20,96% with 10 m/s as mean V_{hub} .

Figure 5.12: Mean standard deviation of all points in the grid during the whole timelength without scaling.



Chapter 6

Conclusion

6.1 Turbulent wind in aeroelastic software

Several software solutions for aero-elastic solutions for wind turbines like AeroDyn (*NWTC Computer-Aided Engineering Tools (AeroDyn) 2014*) and Bladed (*Bladed - DNV GL - RE-NEWABLES ADVISORY - SOFTWARE 2014*) have methods of simulating turbulence on the wind turbine to perform tests. The methods of simulating turbulence in these software applications are always a statistical approach using different turbulence models. Even though several standards exist for horizontal axis wind turbines, the IEC standard is the only standard referred to repeatedly in most of the literature about wind simulation. This is reflected in the choice to include IEC compliant turbulence models and parameters in these software solutions.

Extending Ashes with a tool for turbulence provides an important tool to the user. It is advantageous to have a easy and convenient integration to generate turbulence with IEC parameters. Several solutions for a IEC compatible tool for turbulence were investigated.

Turbsim is very well documented and already used as the simulation tool for turbulence in AeroDyn. The scientists behind the program answer questions regularly on the NWTC forums. Turbsim is free and can be included in other software according to the licence. It provides the user with several options for turbulence modelling. The most important model is the Kaimal turbulence model as this model is recommended by the IEC. Other non site-specific models like the Von Karman and Smooth models are available in Turbsim. They are not recommended in the IEC, but might still be useful for some. Turbsim also provides some turbulence models based on measurements around the NWTC.

Turbsim lets the user control a range of useful parameters to generate wind customized to measurements from a site. The measurements are often made with anemometers located at the desired location for the wind turbine. From the measurements, the average wind, wind profile and turbulence intensity can be obtained. The parameters V_{hub} , power law exponent(α) and turbulence intensity, TI can be set to the measured values. With the

Kaimal turbulence model, a wind field can be generated with Turbsim which follows the guidelines of the IEC. It is also useful to have the options to set different inflow angles in Turbsim to perform different tests. IEC requires the wind turbine to be tested with an influence of inclination of the mean flow with respect to a horizontal plane of up to 8°. (IEC 2005) Extreme turbulence is also a test requirement in the IEC standard (ibid.). Turbsim supports generation of extreme turbulence with all the required parameters.

Even though Turbsim supports IEC turbulence with the Kaimal spectral model, another turbulence model is also suggested in the IEC standard, namely the Mann uniform shear turbulence model. This model assumes an initial condition with a isotropic Von Karman spectrum being rapidly distorted by a uniform, mean velocity shear (ibid.). Implementations of this model was investigated to see if integration with Ashes was a possibility. The program windsimu was potentially a good candidate, but a lack of documentation and unstable behavior made the task difficult. Both the input parameters, inner workings and output file format lacks enough documentation to be used.

Another way of integrating was through Matlab. Francesco Perrone wrote a matlab script in 2012 to generate turbulent wind based on the Mann turbulence model (*3D Turbulent Wind Field by means of the Mann Model: MATLAB Central* 2014). By using the matlab compiler, an exe file for use with Turbsim can be created. This script was a better candidate than windsimu, being open source and producing output consistently. Even though it had some desirable features, it has no documentation other than a few lines about the input. The author is a load calculation engineer at Areva Wind in Germany.

Although the scripts for generating turbulence with the Mann model might work with some more documentation, Turbsim's user guide (Jonkman and Kilcher 2012) states that the Mann turbulence model is on a list of future development. This would be the best way to make the Mann turbulence model available in Ashes given the way Turbsim is well integrated and Turbsim is documented.

6.2 Implementation

Turbsim was chosen as the most solid choice for generation of turbulence in Ashes. The software licence allows bundeling it directly with Ashes (Buhl 2014). The implementation was focused on a software integration with high usability and flexibility for both new users and advanced users. Some features like the scaling to fit IEC target deviation, setting the wind profile to power law, setting the turbulence model to NTM and the IEC standard to the latest edition were removed from the interface to make it more user friendly with little to no sacrifice of functionality. The power law exponent α were included to provide users to produce offshore turbulence and fit the wind profile to measurements if needed.

Observations and knowledge from the parameter study was used to implement auto-updated approximations for memory requirement, run time and file size. When the user changes a parameter that has an impact on these estimates, the approximations gets updated and are visible for the user. This feature will let the user know if the simulation is taking too much memory to perform or how long it is going to run before the execution of Turbsim.

The approximation on run time was written based on the data of the computer used for the testing, and will most likely be wrong in most cases unless the user has the exact same hardware. There is no known way of making an approximation without altering the Turbsim source that will be correct for every user because of difference in hardware and also the impact of other programs running on the system. Still, an approximation should still be more beneficial to the user than no prediction at all. The memory requirement and file size is correct irregardless of the users hardware.

A visualization of the field developed earlier was not used because a new method of showing the forces on the wind turbine developed by Per Ivar Bruheim. Arrows on the wind turbine shows the forces acting on the elements in real time. This had a cleaner look than a two dimensional color chart.

6.3 Testing

A test from the OC4 specifications were performed to check that the output generated with Turbsim in Ashes produced reliable results (Popko et al. n.d.). The probability density function of the wind speeds at hub height produced results that was very consistent with the other software in the test. This suggests a realistic generation of the turbulence.

6.4 Offshore turbines

For normal conditions, offshore turbines can use the recommended parameters for onshore turbines with only modifying the power law exponent α to 0,14. It is however required to do measurements and use the observed wind profile exponent, turbulence intensity and mean wind speed at hub height if an offshore wind turbine is to be tested and constructed. Offshore wind conditions shall be tested with wave conditions, but no correlation is required by IEC (IEC 2009).

6.5 Parameters and performance

The output and performance analysis in chapter 5 was performed on a 3 year old mid end laptop and a high end stationary. The idea was to measure how much the hardware can impact runtime. The results shows that the newer computer performs much faster than the laptop, and the improvement in runtime is bigger at higher grid resolutions. This is important information because it can give an indication on minimum or recommended hardware requirements and recommended maximum number of gridpoints. A requirement that the user should be able to run a 1 hour simulation with 50x50 gridpoints with a 0,05 second timestep can be established as a reasonable requirement. Using equation 5.5 (total elements) in combination with the byte calculations gives a memory requirement of 4,06 GB. This means that the user needs over 5GB of ram. The recommended memory requirement

can be set to 6GB, given that 5GB ram is not very common in computer hardware. The runtime on this simulation for the new stationary computer was 14118 seconds, almost 4 hours. With the aquired knowledge that the laptop can use over 12 times as long performing certain simulations, it could use as much as 47 hours on this simulation if it had the memory to run it. Keeping this in mind, the recommended CPU should be the newer i5 and i7 series from Intel or a corresponding processor.

The software will give an estimation of the runtime. This will let the user have some idea if this is a job that can be performed while waiting in front of the computer, or if the simulation should be performed overnight or in the background.

Turbsim does not run in parallell. This might be updated in the future, as Turbsim is written in Fortran, and can make us of the parallell library openMP to utilize multiple logical processors (*OpenMP - The GNU Fortran Compiler* 2013).

Turbsim can get closer to the memory limit by having a smaller amount of gridpoints and a longer time-series. Big grid arrays means less optimized memory allocation, see **Fig. 5.11**. The user will be able to see the required memory beforehand. The original formula is multiplied with 1,1 to account for overhead memory illustrated in **Fig. 5.11**. This will provide the estimation of memory requirement with an extra buffer to handle the additional memory needed for large grids. The constant 1,1 is calculated from the test on the stationary computer where the calculated memory was 7,31 GB and the memory needed was 8 GB. Users on computers with 8 GB ram will with this have to perform a simulation with over 143x143 gridpoints before getting a conservative/too low estimation.

Users with only 2 GB of ram available will be able to run a simulation up to around 62x62 gridpoints before the estimation will be too low. If the user has more gridpoints and wants to run the simulation up to a timelength where exactly all the memory is used, the estimated memory need displayed to the user will be slightly below the memory actually needed.

The finest grid on a computer with 8 GB RAM in Turbsim64 when simulating a 10 min or 600 second timelength with 0,05 seconds in timestep is 143x143 points, although this is not practical to do with a normal computer without waiting weeks or more for it to finish. The finest grid test performed in the analysis was an 82x82 grid in the Turbsim64 program. A 600 second timelength with 0,05 second timestep at this resolution took 72888 seconds to perform which is over 20 hours.

Turbsim's 64-bit edition was used in Ashes. This will allow the user to perform longer simulations and use finer grids.

6.6 Further development

More turbulence models can be added in Ashes. Turbsim might include the Mann turbulence model in future updates, which will be very convenient to add in to the turbulent wind dialog. The software IECwind can be added, but might not be right for the turbulent wind dialog since it technically is not turbulence, but the tests should be available in Ashes nevertheless. Questions about when we can expect the Mann turbulence model in Turbsim

have remained unanswered, but if the addition of this model is to be implemented by other methods, the program IECwind should be further investigated. The main objective should be to acquire documentation about input and output. With this in hand, the software can be tested correctly and included in the turbulent wind dialog in Ashes. The power spectral density function for the OC4 test 3.4a could not be correctly produced. Some information about the parameters and method of producing this function for the other software is needed.

Bibliography

- 3D Turbulent Wind Field by means of the Mann Model: MATLAB Central* (2014). URL: http://www.mathworks.com/matlabcentral/fileexchange/35722-3d-turbulent-wind-field-by-means-of-the-mann-model/content/IEC_MANN.m (visited on 01/10/2014).
- Bladed - DNV GL - RENEWABLES ADVISORY - SOFTWARE* (2014). URL: <http://www.gl-garradhassan.com/en/GHBladed.php> (visited on 01/10/2014).
- Buhl, M. (2014). *NWTC Computer-Aided Engineering Tools (Disclaimer)*. URL: <http://wind.nrel.gov/designcodes/disclaimer.html> (visited on 01/11/2014).
- Burton, T. (2011). *Wind energy handbook*. Chichester: Wiley. ISBN: 978-1-119-99271-4.
- Frisch, U. (1995). *Turbulence: The Legacy of A. N. Kolmogorov*.
- Hau, E. (2013). *Wind Turbines: Fundamentals, Technologies, Application, Economics*. Berlin, Heidelberg: Springer Berlin Heidelberg. ISBN: 9783642271519.
- Hunt, J. and N. Kevlahan. *Rapid Distortion Theory and the structure of turbulence*. English. URL: http://ms.mcmaster.ca/kevla/pdf/articles/Monte_Verita93.pdf.
- IEC, I. electrotechnical commision (2005). *IEC 61400-1*. 3rd ed. Norsk elektroteknisk komite. ISBN: 2-8318-8161-7.
- (2009). *IEC 61400 3*. 1st ed. Norsk elektroteknisk komite.
- Jonkman, B. and L. Kilcher (2012). *TurbSim User's Guide*.
- Khusnutdinova, K. (1995). *Kolmogorov's 5/3 law*. URL: http://homepages.lboro.ac.uk/~makk/mathrev_kolmogorov.pdf.
- Memory Limits for Windows Releases (Windows)* (2013). URL: [http://msdn.microsoft.com/en-us/library/windows/desktop/aa366778\(v=vs.85\).aspx](http://msdn.microsoft.com/en-us/library/windows/desktop/aa366778(v=vs.85).aspx) (visited on 11/19/2013).
- NWTC Computer-Aided Engineering Tools (AeroDyn)* (2014). URL: <http://wind.nrel.gov/designcodes/simulators/aerodyn/> (visited on 01/10/2014).
- NWTC • View topic - Mann turbulence from HAWC2 using InflowWind* (2014). URL: <https://wind.nrel.gov/forum/wind/viewtopic.php?f=4&t=714> (visited on 01/09/2014).
- OpenMP - The GNU Fortran Compiler* (2013). URL: <http://gcc.gnu.org/onlinedocs/gfortran/OpenMP.html> (visited on 12/29/2013).

-
- Popko, W. et al. *Offshore Code Comparison Collaboration Continuation (OC4), Phase I – Results of Coupled Simulations of an Offshore Wind Turbine with Jacket Support Structure*.
- Process Explorer* (2013). URL: <http://technet.microsoft.com/en-us/sysinternals/bb896653.aspx> (visited on 11/22/2013).
- Qt - Qt Product* (2013). URL: <http://qt.digia.com/product/>.
- Rosa, R. M. S. *TURBULENCE THEORIES*.
- Veers, P. S. (1988). *Three-Dimensional wind Simulation*. English.
- Veldkamp, D. (2006). *Chances in Wind Energy*. DUWIND Delft University Wind Energy Research Institute. ISBN: 90-76468-12-5.
- Worpahl, F. and W. Popko (2013). *Description of the load cases and output sensors to be simulated in the OC4 project under IEA Wind Annex 30*.

Appendix

.1

Appendix A: Runtimes for 4-1600 gridpoints

Dimension	Average runtime	Runtime measurement 1	Runtime measurement 2
2x2	below 1	below 1	below 1
3x3	below 1	below 1	below 1
4x4	below 1	below 1	below 1
5x5	1,0608	1,092	1,0296
6x6	1,365	1,326	1,404
7x7	1,638	1,638	1,638
8x8	2,0826	2,106	2,0592
9x9	2,5896	2,6052	2,574
10x10	3,4008	3,4632	3,3384
11x11	4,1808	4,1184	4,2432
12x12	5,3898	5,4444	5,3352
13x13	6,5286	6,552	6,5052
14x14	8,3773	8,4241	8,3305
15x15	10,2335	10,343	10,124
16x16	12,831	12,73	12,932
17x17	15,795	15,99	15,6
18x18	19,4845	19,703	19,266
19x19	25,4285	26,692	24,165
20x20	29,242	29,281	29,203
21x21	34,125	34,273	33,977
22x22	42,214	41,855	42,573
23x23	51,9795	51,059	52,9
24x24	64,3345	64,631	64,038
25x25	78,0005	78,078	77,923
26x26	96,9855	96,19	97,781
27x27	120,805	122,07	119,54
28x28	138,55	138,23	138,87
29x29	165,69	164,99	166,39
30x30	194,915	194,97	194,86
31x31	229,79	233,16	226,42
32x32	269,515	272,38	266,65
33x33	304,725	306,07	303,38
34x34	351,565	357,74	345,39
35x35	399,995	403,84	396,15
36x36	459,71	468,02	451,4
37x37	530,94	548,2	513,68
38x38	582,51	593,4	571,62
39x39	663,285	661,29	665,28
40x40	748,625	749,18	748,07

Table 1: Measurements of runtime with increase in gridpoints. Performed on i7-3770K @ 3.50GHz.

.2

Appendix B: A standard IEC Kaimal input file

TurbSim Input File. Valid for TurbSim v1.06.00, 21-Sep-2012

```
-----Runtime Options-----
2318573      RandSeed1
RANLUX      RandSeed2
False      WrBHHTP
False      WrFHHTP
False      WrADHH
False      WrADFF
False      WrBLFF
False      WrADTWR
True      WrFMTEFF
False      WrACT
True      Clockwise
1      ScaleIEC

-----Turbine/Model Specifications-----
20      NumGrid_Z
20      NumGrid_Y
0.05      TimeStep
600      AnalysisTime
600      UsableTime
84.2876      HubHt
140.00      GridHeight
140.00      GridWidth
0      VFlowAng
0      HFlowAng

-----Meteorological Boundary Conditions-----
"IECKAI"      TurbModel
"1-ED3"      IECstandard
"A"      IECturbc
"NTM"      IEC_WindType
default      ETMc
default      WindProfileType
84.2876      RefHt
10      URef
default      ZJetMax
default      PLExp
default      Z0

-----Non-IEC Meteorological Boundary Conditions-----
default      Latitude
0.05      RICH_NO
default      UStar
default      ZI
default      PC_UW
default      PC_UV
default      PC_VW
default      IncDec1
default      IncDec2
default      IncDec3
default      CohExp

-----Coherent Turbulence Scaling Parameters-----
"M:\coh_events\eventdata"  CTEventPath
"Random"      CTEventFile
true      Randomize
1.0      DistSc1
0.5      CTLy
0.5      CTLz
30.0      CTStartTime
```

=====
NOTE: Do not add or remove any lines in this file!
=====

Appendix C: Quickstart guide for turbulent wind in Ashes

- 1. **Choose the appropriate folder and filename for your output files.** The software will produce an input file with the parameters used, a sum file with useful statistics and a binary wind file with the .wnd extension or 3 readable files with the .u, .v and .w extensions if binary is turned off.
- 2. **Leave turbulence model at IEC Kaimal, the default.** This model is suggested for use in IEC 61400-1 third edition. Used for normal non-extreme turbulence.
- 3. **Set the appropriate width and height of the grid.** Before setting the resolution, the user should set the width/height of the wind field. This value should be set using : $2 \times (\text{Rotor radius} + \text{Shaft length})$. So with a rotor radius of 40 meters and a shaft of 1 meter, the width of the wind field should be above 82 meters. The height should be the same as the width, resulting in a square wind field.
- 4. **Set the appropriate resolution (number of grid points) of the wind field.** Depending on the turbine being tested, an appropriate resolution should be set. With the wind field width of 82 meters, a resolution of 50 grid points will result in 1,64 meters between each grid point. Increasing the resolution further can be useful, but will also increase the runtime.
- 5. **Set the time length and time step of the simulation.** The length can be set to 3600 seconds, or 1 hour. A time step of 0,05 is recommended for most simulations. The time step determines the maximum frequency f_{max} .

$$f_{max} = \frac{1}{\Delta t} \quad (1)$$

- 6. **Set the height of the hub.** The hub height is the height from the ground to the center of the hub.
- 7. **Set the average wind speed at hub height.** This will tell Turbsim what the average wind shall be at hub height.
- 8. **Choose the turbulence intensity.** If the IEC classes are used, choose A, B or C from the simple interface, or click on the advanced options checkbox to enable advanced options if a user defined value for the turbulence intensity is preferred.
- 9. **Onshore or offshore.** If the turbine is located onshore, go to the next step. If the turbine is located offshore, the power law exponent needs to be changed from "default" to 0,14.
- 10. **Check estimated runtime, memory and output file size.** Make sure the computer running the simulation has more memory than the required amount. Also see that there is enough disk space to save the output. Finally, if the estimated runtime is too high, consider lowering the amount of grid points or lower time length of simulation.
- 11. **Run the simulation.**