Masters's Thesis

Implementation and validation of an open source model for generating wind feed-in time series

Submitted by Sabine Haas
Student ID 339856

Berlin, January 23, 2019

Reviewers Prof. Dr. rer. nat. Frank Behrendt
Technische Universität Berlin

Dipl.-Ing. Jenny Rieck
Technische Universität Berlin

External Supervisor M.Sc. Birgit Schachler
Reiner Lemoine Institute

Declaration of authorship

I hereby declare that this Master Thesis on the topic of „Implementation and validation of an open source model for generating wind feed-in time series“ and the work presented in it is my own work and the result of my research, unless stated otherwise. No other person's work has been used without acknowledgement and all references have been quoted.

Berlin, January 23, 2019

Sabine Haas

Acknowledgments

Danksagung

An dieser Stelle möchte ich mich herzlich bei allen KollegInnen am Reiner Lemoine Institut bedanken. In der dort außergewöhnlich positiven und freundschaftlichen Atmosphäre fühlte ich mich sehr wohl. Ein besonderer Dank gilt Birgit Schachler, die mich während meiner Arbeit am Institut hervorragend betreute und mir stets mit wertvollen Tipps und fachlichen Diskussionen zur Seite stand. Für die Unterstützung und Kommunikation bezüglich benötigter Daten möchte ich mich bei Jonathan Amme und ebenfalls Birgit Schachler bedanken. Des Weiteren danke ich Jenny Rieck für die Betreuung am Fachgebiet Energieverfahrenstechnik und Umwandlungstechniken regenerativer Energien an der Technischen Universität Berlin, für ihre hilfreichen Anmerkungen zu meiner Masterarbeit und für ihre Unterstützung auch auf kurzfristige Anfragen. Weiterhin danke ich vielmals meinen Freunden Sebastian Triebs, Moritz Schenker und Philipp Maskos, die mich beim Korrekturlesen der Arbeit tatkräftig unterstützten. Außerdem möchte ich mich bei meinen FreundInnen und KommilitonInnen bedanken, welche mein Studium und meine Zeit in Berlin zu etwas Besonderem machen.

Abstract

In this thesis an open source model for generating wind feed-in time series called *Windpowerlib* is implemented in Python and validated with measured feed-in time series. A basic version of the Windpowerlib that was developed at the Reiner Lemoine Institute serves as basis of this work. Functionalities like wind speed height corrections, density and power output calculations, power curve smoothing, aggregated power curves and functionalities for the consideration of wake losses are presented, implemented and their effect on simulation results evaluated. The validation is carried out with measured feed-in time series of wind parks in Schleswig-Holstein (coastal region) and Brandenburg (inland region) in Germany for the years 2015 and 2016. Moreover, the influence of weather data on feed-in time series simulations is examined by using two different weather data sets. MERRA-2 data provided by the NASA is compared with open_FRED weather data that was especially created for energy systems simulations.

Wind farm feed-in can be simulated with a deviation of 4.7 % (inland) and 3.4 % (coastal region) from the measured annual energy output (overestimation) by the Windpowerlib when using open_FRED weather data. For MERRA-2 data the deviations are about ten percentage points higher in Schleswig-Holstein (coastal region) and about 26 percentage points higher in Brandenburg (inland region). All generated time series attain strong correlations with the measured time series with Pearson correlation coefficients of about 0.7 to 0.9 while MERRA-2 data reaches slightly higher correlations compared to open_FRED data.

Kurzzusammenfassung

In dieser Arbeit wird das Open Source Modell *Windpowerlib* zur Generierung von Windeinspeisezeitreihen in Python implementiert und mit gemessenen Einspeisezeitreihen validiert. Eine frühere Implementierung der Windpowerlib, welche am Reiner Lemoine Institut entwickelt wurde, dient als Grundlage für diese Arbeit. Es werden Funktionen zur Höhenkorrektur von Windgeschwindigkeiten, Berechnung von Dichte und Leistungsabgabe, Glättung von Leistungskurven, Berechnung aggregierter Leistungskurven und Berücksichtigung von Abschattungseffekten diskutiert, implementiert und deren Effekte auf die Simulationsergebnisse ausgewertet. Die Validierung erfolgt mit Hilfe von gemessenen Einspeisezeitreihen von Windparks in Schleswig-Holstein (Küste) und Brandenburg (Inland) für die Jahre 2015 und 2016. Des Weiteren wird der Einfluss von Wetterdaten auf die Simulationen von Einspeisezeitreihen untersucht, wofür zwei verschiedene Wetterdatensätze verwendet werden. Dabei wird der MERRA-2 Datensatz der NASA mit dem open_FRED-Datensatz, der speziell zur Nutzung in Energiesystemsimulationen erstellt wurde, verglichen.

Mit der Windpowerlib kann der Ertrag eines Windparks mit einer Überschätzung der gemessenen Jahresenergiemenge von 4.7 % (Brandenburg) und 3.4 % (Schleswig-Holstein) simuliert werden, wenn der open_FRED-Datensatz verwendet wird. Diese Abweichungen vom Jahresenergieertrag fallen für MERRA-2-Daten für den Windpark in Schleswig-Holstein um circa zehn Prozentpunkte und für den Windpark in Brandenburg um circa 26 Prozentpunkte höher aus. Alle erzeugten Zeitreihen weisen mit Pearson Korrelationskoeffizienten von circa 0.7 bis 0.9 hohe Korrelationen bezüglich der gemessenen Zeitreihen auf, wobei mit MERRA-2-Daten generierte Zeitreihen etwas höhere Korrelationen besitzen als diejenigen, die mit open_FRED-Daten erzeugt wurden.

Contents

Abstract **V**

Nomenclature **XVI**

List of Acronyms **XVIII**

1 Introduction **1**

1.1 Motivation 1

1.2 Preceding work as basis of this thesis 3

1.3 Open source modelling and open data 4

1.4 Aim and structure of the thesis 5

2 Wind feed-in time series simulation **7**

2.1 Height correction and conversion of weather data 7

2.1.1 Wind speed height corrections 7

2.1.2 Density calculations 8

2.2 Power output calculations 9

2.3 Wake losses in wind farms 11

2.3.1 Definition of wind farms and clusters 11

2.3.2 Wake models 11

2.3.3 Wind farm efficiency and wind efficiency curves 12

2.4 Spatial distribution of wind speeds 15

2.4.1 Smoothed power curve 15

2.4.2 Parameters of the Gauss distribution of smoothed power curves 18

2.5 Aggregated power curves 19

2.5.1 Current research approaches 20

2.5.2 Average power-weighted hub height 21

3 Data basis for the validation of the open source model 23

3.1 Weather data . . . 23

3.2 Measured feed-in and wind speed data . . . 25

3.3 Pre-processing of measured time series . . . 27

3.3.1 Aggregation and filtering . . . 27

3.3.2 First row time series . . . 28

3.4 Wind turbine data . . . 29

4 Methodology 31

4.1 Implementation of the open source model . . . 31

4.1.1 General structure . . . 31

4.1.2 Implementation details . . . 33

4.1.3 The modelchains: an easy start into simulations . . . 37

4.1.4 Required input data . . . 42

4.2 Validation of the Windpowerlib . . . 43

4.2.1 Statistical metrics . . . 44

4.2.2 Temporal and spatial resolution of the examined time series . . . 46

4.2.3 Single functionalities . . . 47

4.2.4 Single wind turbine model . . . 52

4.2.5 Wind farm model . . . 53

4.2.6 Influence of weather data . . . 53

5 Simulation results 55

5.1 Validation of single functionalities . . . 55

5.1.1 Height correction of wind speed data . . . 55

5.1.2 Power output calculations . . . 58

5.1.3 Smoothed power curves . . . 60

5.1.4 Wake losses . . . 65

5.2 Single wind turbine simulations . . . 67

5.3 Wind farm simulations . . . 69

5.4 Simulations comparing the weather data sets . . . 70

5.4.1 Influence on wind speed height corrections . . . 71

5.4.2 Influence on single wind turbine power output simulations . . . 73

5.4.3 Influence on wind farm power output simulations . . . 74

6 Discussion **77**

6.1 Performance of the functionalities 77

6.1.1 Wind speed height correction functionalities 78

6.1.2 Power output calculation functionalities 79

6.1.3 Smoothing power curve functionality 80

6.1.4 Functionalities for the consideration of wake losses 81

6.2 Influence of the weather data 82

6.3 Trustworthiness of the results 83

6.4 Limitations of the Windpowerlib 84

7 Conclusion **85**

8 Outlook **87**

A Appendix **89**

A.1 Evaluation of wind directions and nacelle positions 89

A.2 Additional results of power output calculations 91

A.3 Additional results of power curve smoothing 93

A.4 Additional results of wake losses simulations 94

A.5 Additional results of single wind turbine model simulations 95

Bibliography **97**

List of Figures

1.1 Gross electricity generation from renewables in Germany 2

2.1 Average and extremely deviating wind efficiency curves 14
2.2 Gauss distribution 16
2.3 Smoothed power curve of a Vestas V90 17

4.1 Flow chart of the `ModelChain` 39
4.2 Flow chart of the `TurbineClusterModelChain` 40
4.3 Flow chart of the functionality `assign_power_curve()` 41
4.4 Illustration of input and output parameters of the Windpowerlib . . . 42

5.1 Statistical metrics of hourly wind speed time series 56
5.2 Correlation of hourly wind speed time series 57
5.3 Relative RMSE of hourly wind speed time series (restricted data points) 58
5.4 With different block widths smoothed power curve 61
5.5 With different wind speed ranges smoothed power curve 61
5.6 With different standard deviation methods smoothed power curve . . 62
5.7 Standard deviation of hourly feed-in time series 64
5.8 Pearson correlation coefficient of hourly time series (smoothing) . . . 64
5.9 Power and wind efficiency curves 65
5.10 Statistical metrics of hourly feed-in time series (wake losses) 67
5.11 Statistical metrics of hourly feed-in time series (single wind turbines) 69
5.12 Statistical metrics of hourly feed-in time series (wind farms) 70
5.13 Relative RMSE of hourly wind speed time series (weather data) . . . 71
5.14 Mean bias of hourly wind speed time series (weather data) 72
5.15 Correlation of hourly wind speed time series (weather data) 72

5.16 Relative RMSE of hourly feed-in time series (weather data) 75
5.17 Mean bias of hourly feed-in time series (weather data) 75
5.18 Correlation of hourly feed-in time series (weather data) 76

A.1 Pearson correlation coefficient of monthly time series (smoothing) . . 93
A.2 Statistical metrics of hourly feed-in time series (single wind turbines) 95

List of Tables

3.1 Overview over the weather data sets MERRA and open_FRED 25
3.2 Characteristics of wind farms examined in this work 25
3.3 Characteristics of the measured feed-in time series 26
3.4 Amount of available time steps in wind farm feed-in time series . . . 28
3.5 Amount of available time steps in first row time series 29

4.1 Overview of the functions of the Windpowerlib and the required data 43
4.2 Simulations run for the validation of wind speed functionalities 48
4.3 Simulations run for the validation of power output functionalities . . 48
4.4 Characteristics of simulations (smooth power curve function) 50
4.5 Simulations (wake losses) 51
4.6 Simulations (single turbine model) 53
4.7 Characteristics of simulations (wind farm model) 53
4.8 Characteristics of simulations (weather data) 54

5.1 Relative RMSE of power output simulations 59
5.2 Pearson correlation coefficient of power output simulations 59
5.3 Deviation from measured annual energy output (power output) . . . 60
5.4 Annual energy evaluation of simulations with power curve smoothing 63
5.5 Annual energy evaluation of simulations with wake losses methods . . 66
5.6 Annual energy evaluation of single wind turbine simulations 68
5.7 Annual energy evaluation of single wind turbine simulations 68
5.8 Deviation from the measured annual energy output (wind farms) . . . 70
5.9 Deviation from the measured annual energy output (power output) . 73
5.10 Evaluation of single wind turbine feed-in time series (2015) 73
5.11 Evaluation of single wind turbine feed-in time series (2016) 74

A.1 Correlation of wind directions with nacelle positions 90
A.2 Annual energy output of power output calculations (2015, open_FRED) 91
A.3 Annual energy output of power output calculations (2016, open_FRED) 92
A.4 Standard deviation of calculated and measured feed-in time series . . 93
A.5 Evaluation of feed-in time series (wake losses) 94
A.6 Evaluation of feed-in time series (wake losses) 94
A.7 Evaluation of feed-in time series (single turbine, open_FRED, 2015) . 95
A.8 Evaluation of feed-in time series (single turbine, open_FRED, 2016) . 96
A.9 Evaluation of feed-in time series (single turbine, MERRA, 2015) . . . 96
A.10 Evaluation of feed-in time series (single turbine, MERRA, 2016) . . . 96

Nomenclature

Latin letters

b	[m/s] or [MW]	bias
cp	$[-]$	power coefficient
c_{t}	$[-]$	thrust coefficient
d	$[m]$	Offset of the boundary layer
d	$[m]$	diameter
f	$[-]$	scaling factor
f	$[-]$	probability
h	$[m]$	height above ground
m	[-]	model equation
n	$[-]$	amount
p	$[Pa]$	pressure
p	$[-]$	parameter of density correction
P	$[W]$	power, capacity
Pr	$[-]$	Pearson Correlation Coefficient
$RMSE$	[m/s] or [MW]	root mean squared error
R_{s}	$[J/(molK)$	specific gas constant
T	$[K]$	temperature
TI	$[-]$	turbulence intensity
v	$[m/s]$	speed
x	$([m/s])$	function variable of gauss distribution
z_0	$[m]$	roughness length

Greek letters

α	$[-]$	hellman exponent
η	$[-]$	efficiency
μ	$([m/s])$	mean (offset of gauss function)
ϱ	$[kg/m^3]$	air density
σ	$([m/s])$	standard deviation
σ^2	$([(m/s)^2])$	variance
ω	$[^\circ]$	wind direction

Indices

0	ambient
avg	average
data	data as origin
hub	hub height of a wind turbine
i	$i^{th} element$
n	nominal or normal
p	pressure
site	site specific
std	standard
T	temperature
WF	wind farm
WT	wind turbine

List of Acronyms

Abkürzung	Bedeutung
Agg.	aggregation (simply aggregation approach)
COSMO-CLM	COSMO-Model in CLimate Mode
Cp	power coefficient
d.-c.	density corrected/correction
DWD	Deutscher Wetterdienst
FlaP	Farm Layout Program
-lib	library
Log. interp.	logarithmic interpolation
log	logarithmic wind profile
MERRA	Modern-Era Retrospective Analysis for Research and Applications
oemof	Open Energy Modelling Framework
open_FRED	Open feed-in time series based on a Renewable Energy Database
Openmod	Open Energy Modelling (initiative)
OSM	Open Street Map
pv	photovoltaic
RED	Renewable Energy Directive
RLI	Reiner Lemoine Institut
RMSE	root mean squared error
SP	Staffell-Pfenninger (method)
TI	turbulence intensity
v0.0.4	version 0.0.4
WF	wind farm

Chapter 1

Introduction

1.1 Motivation

Renewable energy sources play a key role in the combat against climate change. The necessity for the transition of energy systems from fossil-fueled power plants to renewable energy power plants and for action against climate change is established in the Sustainable Development Goals of the United Nations [2017c]. Goal 7 is striving to ”ensure universal access to affordable, reliable and modern energy services”, where the ”renewable energy share in the total final energy consumption” needs to be ”increase[d] substantially” and the urgency for taking ”action to combat climate change and its impacts” is anchored in goal 13 [United Nations, 2017a,b].

Furthermore, the Renewable Energy Directive 2009/28/EC (RED) of the European Parliament and Council determines the share of renewable energy in the gross final energy consumption in the EU to reach at least 20 % in the year 2020. The member states committed themselves to national goals which stretch from 10 % to 49 % where Germany is obliged to attain at least 18 % of renewable energy share. In 2014 the European Council agreed on a percentage of at least 27 % in the EU until the year 2030 [Bundesministerium für Wirtschaft und Energie, 2017, p. 33].

Among the different conversion technologies for renewables, such as wind, sunlight, biomass, natural heat of earth and water, wind power plants are one of the most applied technologies in Europe. According to WindEurope [2018, p. 7] an amount of 15.638 GW of wind power was installed in Europe in 2017 which is more than the newly installed capacity of any other energy conversion technology in the same year. Moreover, WindEurope [2018, p. 6] shows that since 2016 wind power has been the second largest power source in Europe concerning the total power generation capacity installed.

Also in Germany wind energy plays a crucial role in power generation. Figure 1.1 shows the progress of the gross electricity generation from renewables in Germany during the time period from 1990 to 2017 (biomass not included). Electricity generation from hydro power shows annual variations around a comparatively high level of 20 000 GWh since the 1990s. Geothermal electricity generation only shows a minor increase from 0 GWh (1990 - 2003) to 155 GWh in 2017 which can not be seen in this figure due to its scale. The electricity generation from solar and wind energy has been increasing rapidly of which the rise of onshore wind energy starts first followed by photovoltaic and offshore wind energy. In 2017 the gross electricity generation from onshore wind power plants was more than twice as high as the one from photovoltaic. This graphic and the earlier mentioned figures of installed capacity published by WindEurope [2018] show that Europe is aiming to fulfill the goals of the United Nations and the RED mentioned above and that wind energy plays a key role in this process.

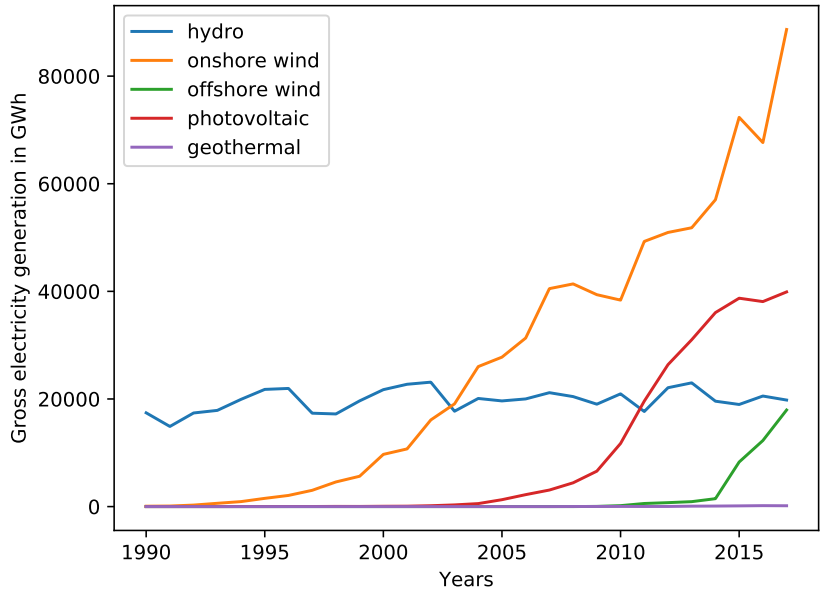


Figure 1.1: *Gross electricity generation from renewables in Germany from 1990 to 2017. Data from AGEE-Stat [2018, p. 6]. Hydro power comprises storage- and run-of-the-river power plants and pumped storage power plants with natural inflow. Biomass is not considered.*

Compared to fossil energy carriers such as coal, natural gas and oil, some renewables such as solar and wind energy, are fluctuating and differ significantly from location to location. This poses new demands on electrical grids. For a reliable electricity supply long- and short-term energy storages need to be installed and grid extensions such as in Germany a north-south connection with higher capacity have to be considered.

For the realization of these demands it is necessary to use modelling tools that can for instance optimize the implementation of storages and grid-extensions. Due to the volatile nature of the most installed renewables and the increasing share of renewables in the European electricity grid these optimization tools need models (or components) that simulate the feed-in of renewables.

1.2 Preceding work as basis of this thesis

This thesis was written at the Reiner Lemoine Institute (RLI) that mainly uses the Open Energy Modelling Framework (oemof)[1] for energy systems simulations, which was developed in a partnership between the Center for Sustainable Energy Systems (ZNES) Flensburg, the RLI and the Otto-von-Guericke-University of Magdeburg (OVGU) [Oemof Developer Group]. Oemof is written in Python and provides its users with a toolbox for energy system modelling. It obtains feed-in time series required for the simulations from standalone applications like feed-in of photovoltaic from the pvlib[2], feed-in of wind power plants from the Windpowerlib[3] and feed-in of hydro power plants from the hydrolib[4]. As wind power plants have a high share in electricity generation in both Germany and Europe and as the pvlib is already well-developed this thesis is dedicated to the further development of the wind feed-in time series generating Windpowerlib.

Version 0.0.4 (v0.0.4) of the Windpowerlib builds the basis of this thesis. It consists of a module `basicmodel` which provides a class `SimpleWindTurbine` for calculating the power output (feed-in) of a wind turbine and of a `basic_example` module showing the basic usage of the Windpowerlib. Moreover, power coefficient curves as well as example weather data is provided. The `SimpleWindTurbine` class includes functions for reading turbine data and one function each for the calculation of wind speed (logarithmic wind profile) and density (barometric height equation) at hub height of the wind turbine. The source code of Windpowerlib v0.0.4 is hosted on the software development platform Github[5].

This basic version of the Windpowerlib was created to be able to start calculations with oemof and to provide a framework for the development of an extensive library. It contains functions which are sufficient for basic feed-in calculations of a single wind

[1] https://github.com/oemof
[2] https://github.com/pvlib/pvlib-python
[3] https://github.com/wind-python/windpowerlib
[4] https://github.com/hydro-python/hydropowerlib/tree/dev
[5] https://github.com/wind-python/windpowerlib/tree/v0.0.4

turbine. However, as the Windpowerlib is aimed to become an extensive and flexibly applicable library it should provide various calculation options. Furthermore, the calculation of a single wind turbine's feed-in is not sufficient for all applications. For instance, for power grid simulations or an estimation of required energy storage capacity it is necessary to generate feed-in time series of wind farms or larger areas. As wind turbines influence each other's power output when standing close to each other the simple aggregation of single wind turbine power output is not detailed enough for all applications. The Windpowerlib is supposed to rather serve for simulations of larger areas than single wind turbines. Moreover, for the users of models like the Windpowerlib it is very important to get an idea of over- or underestimation made by the model, as well as of how it reacts on different qualities of input data. Thus, this thesis is focusing on the implementation of additional functions into the Windpowerlib and its profound validation with measured feed-in time series. The aim of this thesis is further described in the Section 1.4.

1.3 Open source modelling and open data

As remarked in Section 1.1 energy systems simulations require modelling tools that can simulate the feed-in time series of renewables. There exists a multitude of energy system and power generation models. However, a large part is commercial which means firstly that they are not available to all possible users and secondly that the source code can not be viewed nor adapted to the needs of the respective application. In many commercial softwares it is not comprehensible which assumptions are made and which models are used for the calculations. This leads to difficulties in verifying and fully understanding research results [Reiner Lemoine Institut, a].

To address this problem the Open Energy Modelling (Openmod) Initiative[6] was founded in 2014 by researchers from different institutions in Europe [Neon Neue Energieökonomik GmbH]. The Openmod strives for "open[ing] the whole energy modelling process by utilizing open data and open source code" and "promotes open energy modelling in Europe". "Open" means to publish source code and data and to make it freely available, usable and adjustable while it should be published under an open software license[7]. According to the Openmod Initiative "more openness in

[6] http://openmod-initiative.org/

[7] Open source and copyleft licenses are for instance published by the Free Software Foundation [2018].

energy modelling will increase transparency and credibility, reduce wasteful double-work and improve overall quality" [Openmod Initiative].

As the Windpowerlib is an open source library it is freely available on Github where all developers can contribute to the project while responsible persons at the RLI have to accept changes before they are included into the code. The Windpowerlib is supposed to become an open source community project following the example of oemof, to which developers of several institutions have committed. So far no developers outside the RLI have contributed to the Windpowerlib, however, employees of the RLI did apply changes which comprise mainly adding of power curves, implementing the usage of a special data structure (MultiIndex Data Frames) and adjustments in the documentation (docstrings). Although the main developer of the Windpowerlib from v0.0.4 onwards is the author of this thesis all other authors of changes are documented on Github which gives transparency about the work done in this thesis.

1.4 Aim and structure of the thesis

With the Windpowerlib v0.0.4 a basic framework for immediate use and further development was created as described in Section 1.2. The limitations of Windpowerlib v.0.0.4 are the following:

- Power output calculations only possible by power coefficient curve but not by power curve[8]
- Only one available function for each wind speed and density calculations
- Modelling of wind farms or rougher spatial resolutions is missing
- No validation has been made

The aim of this thesis is to implement functions that will abolish the first three limitations listed above and to carry out a validation with measured feed-in time series. Moreover, as the library will grow it is intended to realize a clear module structure. Furthermore, this thesis is part of the open_FRED (Open feed-in time series based on a Renewable Energy Database) project at the Reiner Lemoine Institute which strives for the creation and the publication of consistent standard data that "match[es] the needs of simulation models for fluctuating renewables" and aims at connecting them to open source simulation models [Reiner Lemoine Institut, b]. As

[8] for an explanation of the terminology see Section 2.2

the Windpowerlib is strongly connected to oemof (see Section 1.2), it is supposed to follow the same principles which includes the following points defined by the Oemof Developer Group:

1. Free of charge
2. Open source
3. Transparency
4. Clear documentation
5. Flexibility

The first three requirements are already met by an open source license and the publication of the source code on Github. Moreover, a clear documentation of the implementations during this thesis will be included into the source code. The fifth requirement, flexibility, will be met by introducing a modular structure which allows a flexible usage of single functions and classes. In addition to that, this work aims on implementing a default model, similar to the *modelchain* of the pvlib (see pvlib python [2018]), which is supposed to enable the users to understand and work with the library quickly. As the Windpowerlib will be used in scientific research a high academic standard will be ensured by inserting references for every function implemented.

As mentioned before, this thesis also focuses on validating the Windpowerlib. This implies a comparison of different calculation methods implemented in the library as well as the validation of calculated feed-in time series with measured feed-in data. Furthermore, to examine the influence of input data simulation results using two weather data sets that differ in spatial and temporal resolution are planned to be compared.

This thesis is divided into 8 parts. Chapter 2 provides an introduction into wind feed-in time series simulations and describes various approaches in the literature. Chapter 3 presents the data basis for the simulations in this thesis. The model implementation is depicted in Chapter 4 as well as the simulation cases for the validation. Chapter 5 shows the results of these simulations which are discussed in Chapter 6. A conclusion is given in Chapter 7 which is followed by an outlook in Chapter 8.

Chapter 2

Wind feed-in time series simulation

Wind is a highly fluctuating renewable energy and therefore simulation tools generating wind feed-in time series became inevitable for energy systems simulations. In this chapter fundamental equations for wind feed-in time series are presented and various approaches for different modelling problems are described.

2.1 Height correction and conversion of weather data

Weather data is usually available for a restricted amount of heights above ground. However, for wind feed-in time series calculations weather data is needed at hub height of the examined wind turbines. Thus, weather data has to undergo height corrections.

2.1.1 Wind speed height corrections

The increase of wind speed with height depends on different factors among which are roughness of the landscape, ground structure and temperature profile. For a neutral stratification, flat ground and a uniform roughness length within the boundary layer the vertical wind speed profile can be described by the logarithmic wind profile (log law) [Gasch and Twele, 2005, p. 128, 132]. Equation (2.1) shows the calculation of the wind speed at hub height $v_{\text{wind,hub}}$ with the log law from wind speed data $v_{\text{wind,data}}$ at another height h_{data} via the roughness length z_0 [Gasch and Twele, 2005, p. 131; Hau, 2014, p. 557].

$$v_{\text{wind,hub}} = v_{\text{wind,data}} \cdot \frac{\ln\left(\frac{h_{\text{hub}}}{z_0}\right)}{\ln\left(\frac{h_{\text{data}}}{z_0}\right)} \tag{2.1}$$

Quaschning [2011, p. 278] adds a variable d to this function which includes the offset of the boundary layer due to obstacles in the surroundings. This offset d is estimated by 70 % of the obstacle height. The so altered function in Equation (2.2) results in Equation (2.1) if $d = 0$.

$$v_{\text{wind,hub}} = v_{\text{wind,data}} \cdot \frac{\ln\left(\frac{h_{\text{hub}}-d}{z_0}\right)}{\ln\left(\frac{h_{\text{data}}-d}{z_0}\right)} \tag{2.2}$$

Another commonly used option for wind speed height corrections is the Hellman equation shown in Equation (2.3) which assumes that the wind profile follows a power law. Here the wind speed at hub height is calculated by using the Hellman exponent α [Quaschning, 2011, p. 279; Hau, 2014, p. 557; Sharp, 2015, p. 82 f.].

$$v_{\text{wind,hub}} = v_{\text{wind,data}} \cdot \left(\frac{h_{\text{hub}}}{h_{\text{data}}}\right)^{\alpha} \tag{2.3}$$

The Hellman exponent can adopt values between 0.1 and 0.6. In many studies a value of 1/7 is assumed for onshore wind turbines [Sharp, 2015, p. 83]. Equation (2.4) shows a way to estimate α via h_{hub} and z_0 [Hau, 2014, p. 559; Quaschning, 2011, p. 279].

$$\alpha = \frac{1}{\ln\left(\frac{h_{\text{hub}}}{z_0}\right)} \tag{2.4}$$

Sharp [2015, p. 83 f.] presents findings in the literature concerning the performance of the logarithmic wind profile and the Hellman equation. Among other things he states that Elkinton, Rogers, and McGowan [2006] found out that these equations "perform equivalently with different methods working better depending on the site". Moreover, they showed that the Hellman exponent is time-dependent and that the use of a non-time-dependent Hellman exponent can result in great errors in hourly resolution, however, performs well for a whole year. Hau [2014, p. 559] states that both functions usually underestimate the wind speeds at hub heights higher than 100 m while these underestimations are stronger with the Hellman equation.

2.1.2 Density calculations

The air density at hub height of a wind turbine is necessary in wind feed-in simulations if, for instance, a density correction is applied to power curves as explained later in Section 2.2.

One way to calculate the density at hub height is the barometric height equation which is displayed in Equation (2.5) [Hau, 2014, p. 610; Gasch and Twele, 2005, p. 150]. The density at hub height ϱ_{hub} is calulated using temperature T_{hub} and pressure p_{hub} at hub height as well as the standard temperature $T_0 = 288.15$ K, pressure $p_0 = 1013.25$ hPa and density $\varrho_0 = 1.225$ kg/m^3 [Deutscher Wetterdienst, b].

$$\varrho_{\text{hub}} = p_{\text{hub}} \cdot \frac{\varrho_0 T_0}{p_0 T_{\text{hub}}} \tag{2.5}$$

The pressure given at any height can be corrected to hub height by Equation (2.6) assuming a linear gradient of -1/8 hPa/m [Deutscher Wetterdienst, a].

$$p_{\text{hub}} = \left(\frac{p_{\text{data}}}{100} - (h_{\text{hub}} - h_{\text{p,data}}) \cdot \frac{1}{8} \frac{\text{hPa}}{\text{m}} \right) \cdot 100 \tag{2.6}$$

The change of temperature with height is defined in the standard norm atmosphere which is specified in a document of Deutscher Wetterdienst [b]. It states that temperature decreases with a linear gradient of 6.5 K/km with height. This is expressed in Equation (2.7) where the temperature T_{hub} at hub height of a wind turbine is calculated by applying the linear gradient on the temperature T_{data} at height $h_{\text{T,data}}$ of a weather data set.

$$T_{\text{hub}} = T_{\text{data}} - 0.0065 \cdot (h_{\text{hub}} - h_{\text{T,data}}) \tag{2.7}$$

An alternative option for the calculation of air density is the ideal gas equation [Ahrendts and Kabelac, 2014, p. 23]. To calculate the density at hub height this formula can be converted to Equation (2.8), where $R_{\text{s}} = 8.314$ J/(mol $\cdot$ K) is the specific gas constant. This formula is used by Knorr [2016, p. 96 f.] and Biank [2014].

$$\varrho_{\text{hub}} = \frac{p_{\text{hub}}}{R_{\text{s}} T_{\text{hub}}} \tag{2.8}$$

2.2 Power output calculations

The power output of a wind turbine can be determined via its power curve which describes the relationship between wind speed and power output. Alternatively it can be calculated by using the power coefficient (Cp-) curve of the wind turbine. In this case the power output P at wind speed v_{wind} is defined by Equation (2.9), where d_{rotor} is the rotor diameter of the wind turbine, ϱ_{hub} the density at hub height and

cp the power coefficient of the Cp-curve [Hau, 2014, p. 106, modified; Quaschning, 2011, p. 282 modified].

$$P(v_{\text{wind}}) = \frac{1}{8} \cdot \varrho_{\text{hub}} \cdot d_{\text{rotor}}^2 \cdot \pi \cdot v_{\text{wind}}^3 \cdot cp\,(v_{\text{wind}}) \tag{2.9}$$

The standard for measuring power curves is regulated in IEC 61400-12. This norm defines how power curves are corrected from a site specific air density to standard air density. To correct a standard power curve to be valid at site specific air density this proceeding can be applied reversely. According to Knorr [2016, p. 97] and Svenningsen [2010] there exist multiple equations depending on the rotor power regulation. For pitch regulated wind turbines the standard wind speeds v_{std} of the power curve are replaced by site specific wind speeds v_{site} calculated by Equation (2.10) where ϱ_{site} is the site specific air density and ϱ_0 the standard air density.

$$v_{\text{site}} = v_{\text{std}} \cdot \left(\frac{\varrho_0}{\varrho_{\text{site}}}\right)^{\frac{1}{3}} \tag{2.10}$$

In contrast, the density corrected power curve of a stall regulated wind turbine is achieved by replacing the standard power output values P_{std} of the power curve with the site specific power output values P_{site} calculated by Equation (2.11).

$$P_{\text{site}}(v) = P_{\text{std}}(v) \cdot \frac{\varrho_{\text{site}}}{\varrho_0} \tag{2.11}$$

Svenningsen [2010] explains that the density correction for pitch regulated wind turbines can result in up to 5 % overestimation of the annual energy output. Therefore, he proposes an altered method by which the errors "generally reduce to <1 %". In this method the ratio in Equation (2.10) is taken to the power of a parameter p which depends on v_{std}. This can be seen in Equation (2.12) while Equation (2.13) shows how p depends on v_{std}.

$$v_{\text{site}} = v_{\text{std}} \cdot \left(\frac{\varrho_0}{\varrho_{\text{site}}}\right)^{p(v_{\text{std}})} \tag{2.12}$$

$$p = \begin{cases} \frac{1}{3} & v_{\text{std}} \leq 7.5 \text{ m/s} \\ \frac{1}{15} \cdot v_{\text{std}} - \frac{1}{6} & 7.5 \text{ m/s} < v_{\text{std}} < 12.5 \text{ m/s} \\ \frac{2}{3} & \geq 12.5 \text{ m/s} \end{cases} \tag{2.13}$$

2.3 Wake losses in wind farms

When wind turbines are grouped to wind farms they influence each other. As a wind turbine transforms a part of the kinematic energy of the wind to electric energy the wind speed behind the rotor is reduced. Thus, a wind turbine standing in the second row of a wind farm concerning the wind direction is shadowed by wind turbines of the first row and, therefore, produces less power. These wake losses have to be considered when calculating wind feed-in time series of wind farms.

2.3.1 Definition of wind farms and clusters

As Knorr [2016, p. 22] states wind farms are often defined by ownership. However, from a computational modelling point of view wind turbines group to wind farms by the distance between each other. According to Knorr [2016] a wind farm has to be seen as a collection of wind turbines that are in sufficient distance to other wind turbines to generate negligible wake losses on them. He further states that this case normally occurs from a distance of ten rotor diameters. This is confirmed by Hau [2014, p. 784 ff.] who writes that a minimum wind turbine distance of eight to ten rotor diameters in the prevailing wind direction and of three to five rotor diameters vertical to the prevailing wind direction results in an acceptable wind farm layout. In his work Knorr [2016] groups wind turbines depending on their distance to other wind turbines and calls the result *clusters* (see Section 2.3.3). This work will use the term *wind farms* when refering to Knorr's clusters while a *cluster* or *wind turbine cluster* comprises wind turbines belonging to one weather data point. In simulations each wind turbine is assigned to the weather data point closest to its location. This means a cluster can include several wind farms as well as single wind turbines. This is useful as in simulation models usually to all wind farms and wind turbines the same wind speed of the common weather data point is applied. Section 2.5 explaines how an aggregated power curve representing all wind turbines within a cluster on which this wind speed data is applied can be calculated.

2.3.2 Wake models

In the literature a large number of models for considering wake losses can be found. Renkema [2007, p. 2] validates wake models available in WindPRO among other models "that can be used in a[n] optimization tool". He gives an overview of these

wake models and categorizes them into kinematic (explicit) wake models and field (implicit) models. Kinematic models determine the wind speed deficit behind a wind turbine whereas field models calculate a complete flow field within a wind farm. The Jensen model [Jensen, 1983], which is one of the oldest models and is widely used in the literature, the Larsen model and the Frandsen model [Frandsen, Barthelmie, Pryor, Rathmann, Larsen, Højstrup, and Thøgersen, 2006] count to the kinematic models. Often mentioned in the literature is the model by Ainslie [Ainslie, 1988] which is a two-dimensional field model [Renkema, 2007, p. 5 ff.].

These wake models have one thing in common. When applying them to wind feed-in time series simulations the layouts of the examined wind farms have to be known. When simulating large areas handling this is time intensive and requires extensive data sources as the exact locations of all wind turbines have to be available. However, this data is not freely available for all countries and areas. Thus, non of these models will be implemented into the model in this thesis and, therefore, they will not be described further. More detailed information on wake models can for instance be found in the works of Knorr [2016], Renkema [2007], Shakoor, Hassan, Raheem, and Rasheed [2015] and Barthelmie, Folkerts, Larsen, Rados, Pryor, Frandsen, Lange, and Schepers [2005].

2.3.3 Wind farm efficiency and wind efficiency curves

As explained in Section 2.3.2 wake models require the layouts of wind farms for which wake losses are aimed to be modeled. If the layouts are not known, which will supposedly be the case for Windpowerlib applications, estimated wake losses in form of a wind farm efficiency reducing the power can be used instead. A wind farm efficiency η can be understood as shown in Equation (2.14) as the ratio of the actual power output of the wind farm $P_{\text{wind farm}}$ to the hypothetical power output of this wind farm when every wind turbine would deliver a feed-in not influenced by wake losses. The latter is expressed by the sum over the power output of the single undisturbed wind turbines $P_{\text{i,single}}$ in Equation (2.14).

$$\eta = \frac{P_{\text{wind farm}}}{\sum_i P_{\text{i,single}}} \tag{2.14}$$

One way of applying this is to use a constant wind farm efficiency. However, Barthelmie, Hansen, and Pryor [2013, p. 1017] found out that the "largest changes in wind farm efficiency are those associated with changing wind speeds" when they

analyzed "the influence of wind speed, wind direction/turbine spacing, TI [(turbulence intensity)], and atmospheric stability on wind farm efficiency" using SCADA data from two large wind farms. Therefore, it would be more convenient to use an efficiency curve that determines the efficiency of a wind farm dependent on wind speed. Two studies are using efficiency curves reducing wind speeds instead of wind farm efficiencies reducing power output, which is explained in the following.

In the "dena-Netzstudie II" [Kohler, Agricola, and Seidl, 2010] and in the work of Knorr [2016] *wind efficiency curves* are used to calculated wake losses in wind farms. In contrast to a wind farm efficiency as shown in Equation (2.14), that depicts a power loss, a *wind efficiency* determines the average reduction of wind speeds within a wind farm induced by wake losses. The wind efficiency curves used and generated in these two studies define the wind efficiency depending on the wind speed and are applied on wind speeds from the weather data before power output calculations.

To attain these wind efficiency curves, at first, the "dena-Netzstudie II" [Kohler et al., 2010] and Knorr [2016] calculate so-called *wind efficiency fields*. A wind efficiency field determines the wind efficiency depending on wind speed and wind direction. The wind efficiency at one wind turbine of a wind farm η_{WT} is calculated by Equation (2.15) by the relation between the wind speed v_{wind} before interaction with the wind farm and the wind speed v_{WT} that arrives at the wind turbine.

$$\eta_{\mathrm{WT}}(v,\omega) = \frac{v_{\mathrm{wind}}(\omega)}{v_{\mathrm{WT}}(\omega)} \tag{2.15}$$

The "dena-Netzstudie II" [Kohler et al., 2010, p. 99 f.] determines wind efficiency fields for 12 reference wind farms spread over Germany by using the Jensen Model in the Farm Layout Program (FlaP), that was developed by the University of Oldenburg for dimensioning and optimizing wind farms. Knorr [2016, p. 114 ff.] uses a model that "underlies trigonometric calculations that are often [...] summarized with the terms 'Jensen-, Risø- or Ainslie-Modell'" for calculating the wind efficiency fields of 2364 wind farms. In both studies first the wind efficiency field of each turbine of a wind farm is determined and then the average wind efficiency field of the whole wind farm η_{WF} is calculated by averaging the fields of all turbines within the wind farm. Thereby, a power weighted average, like shown in Equation (2.16), is applied which gives the efficiency $\eta_{\mathrm{WT,i}}$ of turbines with a higher nominal power $P_{\mathrm{n,WT,i}}$ a greater weight.

$$\eta_{\mathrm{WF}}(v,\omega) = \sum_i \frac{\eta_{\mathrm{WT,i}}(v,\omega) \cdot P_{\mathrm{n,WT,i}}}{\sum\limits_i P_{\mathrm{n,WT,i}}} \tag{2.16}$$

In the next step wind efficiency curves $\eta_{\mathrm{WF}}(v)$ are derived from the wind efficiency fields $\eta_{\mathrm{WF}}(v, \omega_{\mathrm{i}})$ by averaging over wind directions ω_{i} while taking the frequency $h(\omega_{\mathrm{i}})$ of these wind directions into consideration (see Equation (2.17)). Knorr [2016, p. 125] uses COSMO-DE data for this and applies hourly wind direction values at 73 meters above ground of the year 2012.

$$\eta_{\mathrm{WF}}(v) = \sum_{i} \eta_{\mathrm{WF}}(v, \omega_{\mathrm{i}}) \cdot h(\omega_{\mathrm{i}}) \tag{2.17}$$

On the basis of these wind efficiency curves of 12 or respectively 2364 wind farms in both studies a mean wind efficiency curve is calculated by averaging all wind efficiency curves. Figure 2.1 illustrates the mean wind efficiency curves and extremely deviating curves of single wind farms of both studies. The individual curves deviate considerably from the mean curves which means that wind efficiency curves can differ indeed for different wind farms. However, the "dena-Netzstudie II" [Kohler et al., 2010, p. 100] explains that the area and turbine number of the 12 reference wind farms does not provide insight into the spatial distribution of the turbines. Due to that they cannot assign individual wind efficiency curves to wind farms and, therefore, use the mean wind efficiency curve for predicting the wind feed-in of 2020.

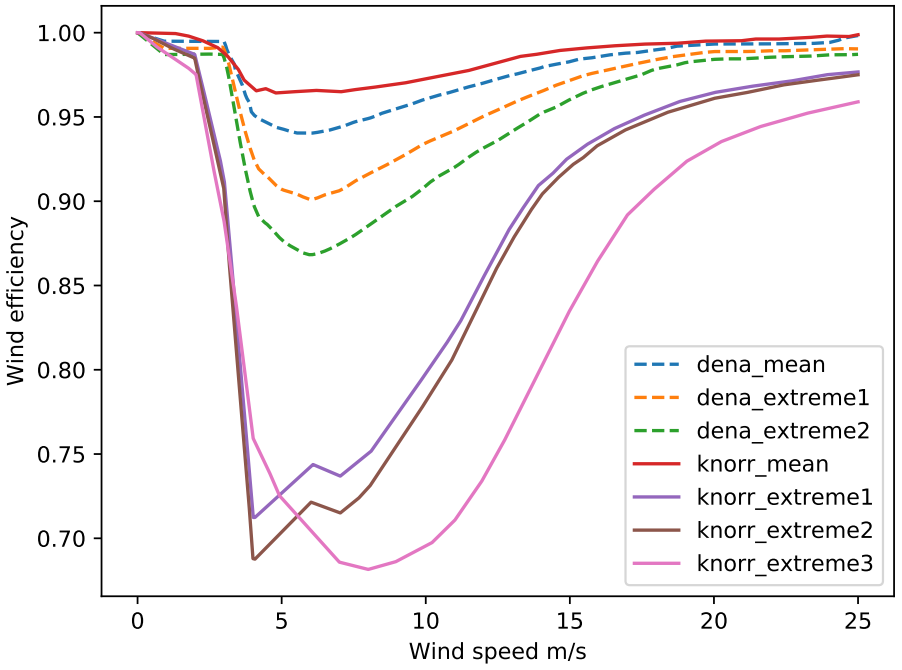


Figure 2.1: *Average and extremely deviating wind efficiency curves from the "dena-Netzstudie II" [Kohler et al., 2010, p. 101] and from the work of Knorr [2016, p. 124]*

Knorr [2016, p. 123 ff.] additionally suggests to generate wind efficiency curves by a model equation using mirrored thrust efficiency curves of wind turbines. Equation (2.18) shows his model equation where c_t is the thrust coefficient and f a scaling factor. However, he detects no clear dependency of the factor f on the following wind farm characteristics: number of turbines, wind farm area, maximum turbine distance, standard distance.

$$m_{\eta,\mathrm{WF}}(v_{\mathrm{WF}}) = 1 - f \cdot c_t(v_{\mathrm{WF}}) \tag{2.18}$$

To distinguish between wind farm efficiency (curves) and wind efficiency curves in this thesis wind farm efficiency curves, which depict the power reduction due to wake losses, will be called *power efficiency curves*.

2.4 Spatial distribution of wind speeds

Wind is a fluctuating renewable energy and differs in space and time. In weather data this can only be expressed up to a certain extent since its spatial and temporal resolution is limited. Concerning the spatial resolution this leads to errors when the feed-in of several wind turbines spread over an area is calculated. In simulation models like the Windpowerlib each turbine is assigned to the closest weather data point. This means that in these models all wind turbines within a certain area (e.g. about 50 x 50 km in MERRA-2 data or 6 x 6 km in open_FRED data, see Section 3.1) encounter the same wind speed and, therefore, produce a power output induced by this identical wind speed (or an identical power output if they are of the same turbine type). In reality wind speed differs over the area. Thus, the wind turbines produce a power output that is induced by different wind speeds (and is not identical for identical turbine types). As Nørgaard and Holttinen [2000, p. 2] explain this results in a smoothing of the single turbines' power output fluctuations and leads to the effect that simulated wind feed-in time series show higher fluctuations than measured feed-in time series.

2.4.1 Smoothed power curve

To account for the spatial distribution of wind speeds within an area Nørgaard and Holttinen [2000] assume that wind speed is distributed by a Gauss distribution over space.

Equation (2.19) shows the Gauss distribution where σ is the standard deviation and μ the mean [Berendsen, 2011, p. 37].

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{(x-\mu)^2}{2\sigma^2}\right] \tag{2.19}$$

Figure 2.2 shows an example of the Gauss distribution with a σ of 1.2 m/s and a μ of 0 m/s. The function variable x in this case was differed in steps of 0.1 m/s from -7 m/s to 7 m/s. μ creates an offset which would move the graph to the left for negative values and to the right for positive values. The Gauss function is not defined for $\sigma = 0$ and becomes a normal distribution for $\mu = 0$. Approaches for defining the parameters σ and μ in a wind feed-in simulation are depicted in Section 2.4.2.

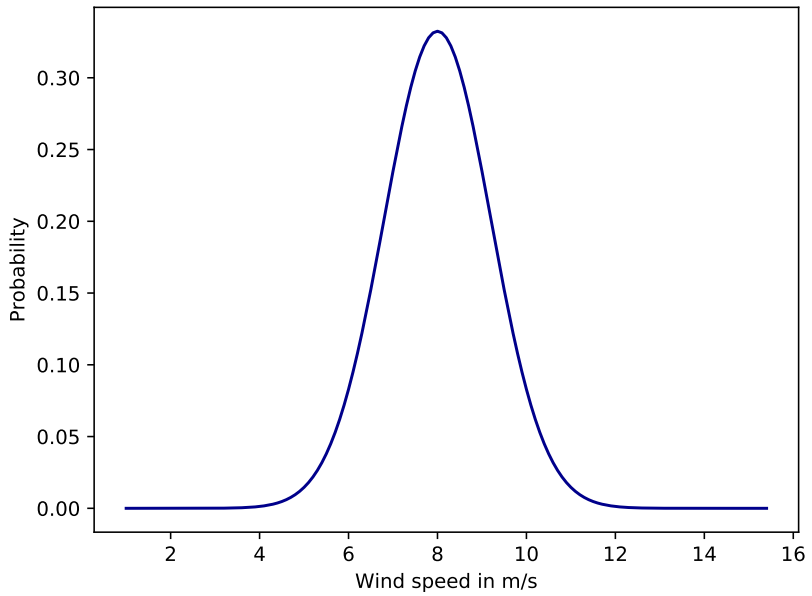


Figure 2.2: *Gauss distribution for a standard deviation of 1.2 m/s and an offset of 0 m/s [Nørgaard and Holttinen, 2000, p. 4, modified]*

By applying the distribution of wind speeds from Equation (2.19) and Figure 2.2 to a power curve a smoothed power curve as illustrated in Figure 2.3 is received.

As Nørgaard and Holttinen [2000, p. 2 f.] are aiming at creating a so-called multi-turbine power curve they apply the distribution to a power curve representing all turbines within one area. Equation (2.20) shows how they find the j^{th} element of the smoothed power curve $P_{\text{smoothed, j}}$. $P_{\text{j+i}}$ is the $(j+i)^{th}$ element of the original power curve and f_i the probability of the spatial distribution in Figure 2.2. They state that

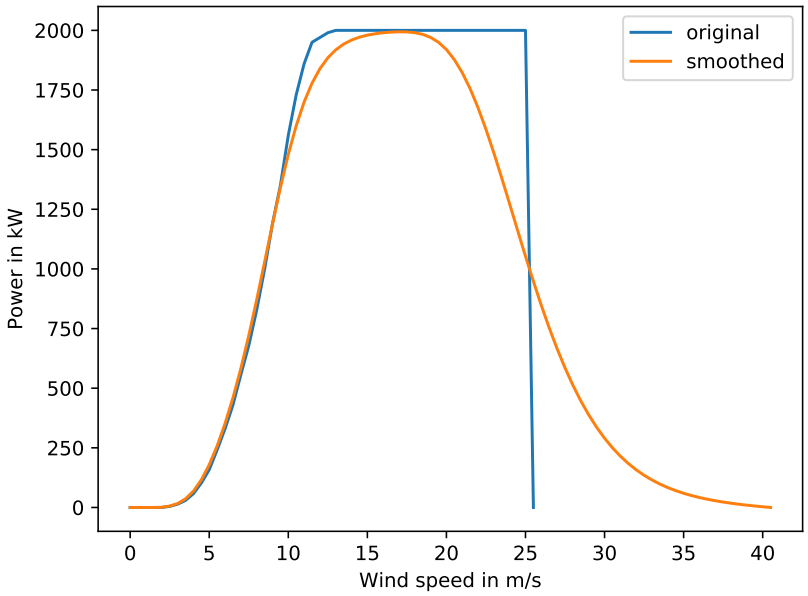


Figure 2.3: *Smoothed power curve of a Vestas V90 with the Gauss distribution of Figure 2.2*

the sum in Equation (2.20) "should as a minimum be done for a wind speed range from -5 m/s to +5 m/s around the j$^{\text{th}}$ element in the power curve".

$$P_{\text{smoothed, j}} = \sum_i P_{\text{j+i}} * f_{\text{i}} \tag{2.20}$$

This way of smoothing power curves is also used in the "dena-Netzstudie II" [Kohler et al., 2010] and in the dissertation of Knorr [2016]. In the latter Knorr [2016, p. 106 ff.] explains the procedure of creating a smoothed power curve according to Nørgaard and Holttinen [2000] in detail and makes suggestions for extensions. Equation (2.21) shows a modified version of his interpretation of how to achieve the values of a smoothed power curve P_{smoothed} for each wind speed v_{std} of the power curve [Knorr, 2016, p. 106]. The sum is done for v_{i} which are the wind speeds around the power curve wind speed v_{std} for which Nørgaard and Holttinen [2000] recommend to use a range from -5 m/s to +5 m/s (see above) and Δv_{i} is the interval length between v_{i} and $v_{\text{i+1}}$. σ and μ are the parameters of the Gauss distribution in Equation (2.19).

$$P_{\text{smoothed}}(v_{\text{std}}) = \sum_{v_{\text{i}}} \Delta v_{\text{i}} \cdot P(v_{\text{i}}) \cdot \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{(v_{\text{std}} - v_{\text{i}} - \mu)^2}{2\sigma^2}\right] \tag{2.21}$$

Figure 2.3 shows the result of smoothing the power curve of a Vestas V90 wind turbine with the Gauss distribution of Figure 2.2. It can be seen that the smoothed power curve lies slightly above the original power curve for the first section between 0 m/s and the nominal speed. This effects occurs due to the rising slope of the original power curve. Shortly before this slope starts to decrease the slope of the smoothed power curve decreases, too, and the smoothed curve's values lie below the ones of the original curve. The smoothed power curve arrives at nominal power after the nominal speed due to the lower power values before nominal wind speed that are considered in the sum of Equation (2.21). From there the smoothed power curve stays for some wind speeds at nominal power and lowers itself a great range before the cut-out wind speed. This is due to the power values of 0 MW from higher wind speeds that are included into the sum of Equation (2.21). For power curve wind speeds higher than cut-out wind speed power values of wind speeds lower than cut-out wind speed are included into the sum. Therefore, the smoothed power curve does not fall to 0 MW at cut-out speed immediately.

The smoothing of feed-in fluctuations for larger aggregation levels can be evaluated by the standard deviation or relative standard deviation (standard deviation divided by average) of the time series as Nørgaard and Holttinen [2000, p. 2] do for two areas. One of their examples is the power generation at the West coast in Finland in 2001 where the relative standard deviation of the hourly feed-in time series of a single wind turbine was 1.14, of three wind turbine gatherings containing eight wind turbines in an area of ten kilometers 1.02 and of five sites within an area of 200 km 0.93.

2.4.2 Parameters of the Gauss distribution of smoothed power curves

The crucial part of using smoothed power curves in wind feed-in simulations is to find a suitable way to describe the parameters of the Gauss distribution. This section is a summary of the usage of these parameters in different studies.

Nørgaard and Holttinen [2000] and Staffell and Pfenninger [2005] use a standard deviation that is dependent on the power curve wind speed. Nørgaard and Holttinen [2000, p. 5] multiply the normalized standard deviation σ_{n} with the power curve wind speed v_{std} as shown in Equation (2.22) to get σ for the Gauss distribution.

$$\sigma = v_{\mathrm{std}} \cdot \sigma_{\mathrm{n}} \tag{2.22}$$

According to their findings the normalized standard deviation depends on the spatial dimension of the area over which the examined wind turbines are spread and on the turbulence intensity (TI). They provide a graphic which shows the normalized standard deviation for an area dimension from 0 km to 300 km and a TI of 5 %, 10 %, 15 % and 20 %. However, they state that this is "still to be further empiric[ally] validated", which is why this approach of getting the standard deviation is not considered in this thesis. To attain μ they use an optimization by altering the value until they get an identical accumulated annual production from the original and the smoothed power curve concerning a Weibull distribution of wind speeds.

Staffell and Pfenninger [2005, p.11] use Equation (2.23) for which they determine the parameters empirically.

$$\sigma = 0.6 + 0.2 \cdot v_{\text{std}} \tag{2.23}$$

In the "dena-Netzstudie II" [Kohler et al., 2010] σ is determined independently from the wind speed in an optimization in which the optimum is found when the sum of the squared errors between the measured and simulated power output is minimal (least squares method) [Kohler et al., 2010, p. 71]. Knorr [2016, p. 107 f., appendix p. 4] optimizes σ in the same way, however, he states as well that Equation (2.22) can be used. The therefore needed normalized standard deviation he interprets as turbulence intensity, as it is a standard deviation normalized on an average wind speed. He does not consider an offset adjustment and sets μ to 0 m/s. For an estimation of the turbulence intensity TI at height h Knorr [2016, p. 88] gives the following Equation (2.24) where z_0 is the roughness length of the area.

$$TI \approx \frac{1}{\ln\left(\frac{h}{z_0}\right)} \tag{2.24}$$

2.5 Aggregated power curves

In energy systems simulations feed-in time series are usually calculated for different sized areas. The extension of these areas can differ from village size up to whole countries. The data basis of wind turbine's locations and specifications is limited for many countries. Therefore, it can be useful to utilize characteristic model power curves that represent the power curves of a collection of wind turbines within an area and are scaled to the installed power output. Even if a more detailed data basis is available aggregated power curves can be practical to save computational time. Furthermore, if power efficiency curves are used for modelling wake losses of a wind farm, like introduced in Section 2.3.3, it is practical to apply them to a wind farm

power curve or cluster power curve (see Section 2.3.1) instead of applying them to every single wind turbines' power curve. The same counts for power curve smoothing (see Section 2.4.1).

2.5.1 Current research approaches

An approach for representative power curves can be found in the work of Nørgaard and Holttinen [2000] who introduce a so-called *multi-turbine power curve* approach to simulate feed-in time series of a wind turbine cluster with similar wind turbines. The approach is limited to similar turbine types as only one power curve that is supposed to represent all wind turbines within one area is used. They smooth the representative power curve as explained in Section 2.4.1.

In the "dena-Netzstudie II" Kohler et al. [2010] generate aggregated power curves for offshore wind farms and onshore network nodes to estimate the wind feed-in time series in the year 2020. The aggregated onshore power curves result from summing up all power curves of wind turbines that are estimated to further be in operation in 2020. For the remaining predicted installed power a model power curve that was created earlier is scaled to this remaining installed power and is added to the aggregated onshore power curves. In the following these aggregated power curves are smoothed by applying a Gauss distribution and undergo a density correction. The model power curve is conceived by averaging the power curves of an Enercon E 82 and a Vestas V90 wind turbine [Kohler et al., 2010, p. 96 f.].

Knorr [2016, p. 127] depicts the generation of wind farm power curves by a summation of single power curves in his work. To account for the spatial distribution of the wind speeds he smooths the aggregated power curves as described in Section 2.4.1. Like the "dena-Netzstudie II" he uses a density correction. In addition to that Knorr [2016, p. 96] points out a proceeding that is not presented in the other studies mentioned before. Due to the different hub heights wind turbines within one wind farm might have he suggests to use a height correction for the power curve wind speeds to the mean hub height of the wind farm. The suggested Equation (2.25) calculates a power curve wind speed $v_{\mathrm{std},h_{\mathrm{WF}}}$ at the average wind farm hub height h_{WF} (see Equation (2.27)) by the logarithmic wind profile (see Equation (2.1)), where h_{WT} is

the hub height of the single wind turbine and $v_{\mathrm{std},h_{\mathrm{WT}}}$ the original power curve wind speed at turbine hub height.

$$v_{\mathrm{std},h_{\mathrm{WF}}} = v_{\mathrm{std},h_{\mathrm{WT}}} \cdot \frac{\ln\left(\frac{h_{\mathrm{WF}}}{z_0}\right)}{\ln\left(\frac{h_{\mathrm{WT}}}{z_0}\right)} \tag{2.25}$$

2.5.2 Average power-weighted hub height

Finally, before the power output can be calculated with a wind farm, cluster or model power curve the average hub height of the included wind turbines is needed. In the "dena-Netzstudie II" Kohler et al. [2010, p. 94] calculate a power weighted mean hub height $h_{\mathrm{node},2020}$ for a node in 2020 by using Equation (2.26). The first part of the equation weighs the hub height $h_{\mathrm{WT},2020}$ estimated for 2020 with the newly installed power over the estimated installed power in 2020 $P_{\mathrm{n},2020}$. The second part of the equation sums up the hub heights of the existing turbines in 2007 $h_{\mathrm{WT},k,2007}$ that are weighted with the installed power likewise.

$$h_{\mathrm{node},2020} = h_{\mathrm{WT},2020}\frac{P_{\mathrm{n},2020} - P_{\mathrm{n},2007}}{P_{\mathrm{n},2020}} + \sum_k h_{\mathrm{WT},k,2007}\frac{P_{\mathrm{n},k,2007}}{P_{\mathrm{n},2020}} \tag{2.26}$$

Knorr [2016, p. 34 f.], as well, calculates a power weighted mean hub height as shown in Equation (2.27). In contrast to the "dena-Netzstudie II" he sums up the logarithm of the hub heights $h_{\mathrm{WT},k}$ of the wind turbines after multiplying them with the ratio of the nominal power of the turbine $P_{\mathrm{n},k}$ and the total nominal power of all turbines $\sum_k P_{\mathrm{n},k}$.

$$h_{\mathrm{WF}} = \exp\left(\sum_k \ln(h_{\mathrm{WT},k})\frac{P_{\mathrm{n},k}}{\sum_k P_{\mathrm{n},k}}\right) \tag{2.27}$$

Chapter 3

Data basis for the validation of the open source model

This chapter presents two weather data sets used in this thesis in Section 3.1. Measured feed-in and wind speed time series for the validation of the model are presented in Section 3.2. Before the measured data can be used for the validation it has to be processed which is depicted in Section 3.3. This chapter ends in Section 3.4 with a short description of the wind turbine data needed for simulations.

3.1 Weather data

For the simulation of wind feed-in time series two different weather data sets are used in this thesis. They differ in temporal and spatial resolution as well as in heights above ground for which weather data is available. These different weather data sets are used to investigate the influence of different input data on the results. MERRA (and MERRA-2) is a weather data set frequently used in energy systems simulations which was in first place not created for that purpose. In contrast, the open_FRED weather data set is especially adjusted to energy systems simulations.

NASA's reanalysis weather data set MERRA-2

MERRA-2 is the second version of the *Modern-Era Retrospective Analysis for Research and Applications* (MERRA) provided by the NASA [Pawson, 2017]. It is a long-term global reanalysis weather data set which includes data that ranges from 1980 until today with a temporal resolution of one hour. The spatial resolution is

the same as in the first version of MERRA: 0.5° x 0.625° [Global Modeling and Assimilation Office, 2012, p. 6], which corresponds to about 50 km in the latitudinal direction. For wind feed-in simulations interesting data of MERRA-2 comprises wind speeds at 50 m (and two and ten meters above displacement height), temperature, pressure and roughness length. This data set will be referred to as *MERRA (weather) data* in this thesis.

Open_FRED test weather data set

The second weather data set used in this thesis is a test data set from the weather data that is created by the Helmholtz-Zentrum Geesthacht[1] for the open_FRED project which was mentioned in Section 1.4. This test data set exists for the years 2015 and 2016 and will be referred to as *open_FRED (weather) data* in this work. It was generated with the *COSMO-Model in CLimate Mode* (COSMO-CLM) of the Climate Limited-area Modelling Community [2018] using MERRA data as input data. The *COSMO-Model* is an atmospheric prediction model which was developed by the Consortium for Small-Scale Modelling [2011] from the formerly *Lokal Modell* from Deutscher Wetterdienst (DWD) [Steppeler, Doms, and Adrian, 2002]. The temporal resolution of the open_FRED weather data is 30 minutes and the spatial resolution about 6.6 km (0.0625°). At the moment the data set comprises a roughness length as well as wind speed, wind direction, pressure and temperature at the following heights: 10 m, 80 m, 100 m, 120 m, 140 m, 160 m, 200 m, 240 m. In the future further variables like density and turbulence intensity are planned to be added. Apart from these values the open_FRED weather data set contains values interesting for solar and hydro power simulations.

The MERRA and open_FRED weather data sets at one glance

Table 3.1 shows an overview over both weather data sets used in this thesis. It can be seen that the spatial as well as the temporal resolution of the open_FRED weather data is finer than the one of the MERRA data. Apart from that, wind speed, pressure and temperature is available at far more heights in the open_FRED than in the MERRA weather data set.

[1] https://www.hzg.de/index.php.de

Table 3.1: *Overview over the weather data sets MERRA and open_FRED*

Resolution	**MERRA**	**open_FRED**
Spatial resolution	$\sim$ 50 km	$\sim$ 6.6 km
Temporal resolution	60 min	30 min
Parameters	**MERRA heights**	**open_FRED heights**
Wind speed	50 m	10m, 80m, 100m, 120m, 140m, 160m, 200m, 240m
Temperature	2 m above displacement height	
Pressure	0 m	
Density	0 m	expected in Summer 2018
Roughness length	available	available

3.2 Measured feed-in and wind speed data

For the validation of the Windpowerlib measured wind feed-in time series are needed. As the open_FRED test weather data set provides data for the years 2015 and 2016 measured feed-in time series for this time period are used. Apart from that, it is convenient to use time series of wind farms at different locations. For simulations in this thesis feed-in time series of five wind farms are available. Their locations differ from close to the north sea coast in Schleswig-Holstein to flat inland country in Brandenburg. Moreover, the wind farms are of different sizes, from two to 17 wind turbines within one farm, and they include different turbine types. As names and exact locations of the wind farms are not permitted to be named alias names are given for each wind farm (WF) and locations are only named by region. Table 3.2 shows characteristics of these wind farms and Table 3.3 of the time series provided for these wind farms. The temporal resolution of the obtained time series ranges from one to ten minutes and their values were averaged from more frequent measurements according to the providers of the data.

Table 3.2: *Characteristics of the wind farms measured feed-in time series are available for in this work*

Alias	**Location**	**Amount of turbines**	**Mean hub height [m]**	**Power class(es) [MW]**	**Comments**
WF BE	Brandenburg (east)	9	105	2	add. WF in the west
WF BNW	Brandenburg (north-west)	2	60	2	surrounded by other wind turbines
WF BS	Brandenburg (south)	14	105	2	add. WF in the south and east
WF BNE	Brandenburg (north-east)	17	104.5	1,5 – 2	various turbine types
WF SH	Schleswig-Holstein	6	64	2,3	-

Wind farms BE, BNW and BS are situated in east, north-west and south Brandenburg. The data available for these wind farms comprises wind speed, power output, wind direction and nacelle position for the single wind turbines. In addition to that, error codes for each time step and wind turbine are available of which some only

Table 3.3: *Characteristics of the measured feed-in time series*

Alias	Temporal resolution [min] 2015	2016	Wind speed data corrected	Curtailment data
WF BE	10	10	yes	-
WF BS	10	10	yes	-
WF BNW	10	10	yes	-
WF BNE	-	5	no information	provided
WF SH	5	1	no information	-

represent a warning. The wind speed data originates from the nacelles but was corrected by the operator to represent wind speeds in front of the rotors. For the wind turbines of WF BE no wind directions are available. Therefore, nacelle positions of the wind turbines are used for all wind farms instead of wind directions after making sure that existing wind directions (of WF BNW and WF BS) have a strong correlation with the nacelle positions and that the average deviation between these two is low (see Appendix A.1). The surroundings of the wind farms were described by the provider of the data as follows. While in the west of WF BE only a narrow area is covered by another wind farm WF BS has high coverage in the south to east. No information like that was given for WF BNW, however, investigation on Open Street Map (OSM) showed that the turbines are surrounded by other wind turbines.

Time series for WF BNE are only available for the year 2016. This wind farm is located in north-east Brandenburg and is the only wind farm in this thesis comprising more than one wind turbine type and different hub heights. Apart from wind speed and feed-in time series curtailment data is available for this wind farm. During the simulation of the feed-in of this wind farm this curtailment data is taken into consideration by reducing the simulated power output by the curtailment at each time step. Time steps that contain total curtailment (switch off) are not considered as the concordance of simulated and measured time series would obviously be 100 % for these time steps (power output of zero) which would distort the results. As according to the provider some of the wind direction data contains an azimuth error it is not used in this thesis. No other wind turbines could be found close to WF BNE on OSM and villages lie one to two kilometers away.

Wind farm SH is situated in Schleswig-Holstein not far from the coast. The obtained data comprises power output, wind speed and wind direction of the wind turbines. However, no information about whether the wind speed data was corrected and whether measured wind directions are correct has been received. The measurement data of 2015 starts in May.

To be comparable with the calculated time series in this thesis which have the temporal resolution of the weather data (see Section 3.1) all measured feed-in time series are resampled to 30 and 60 minutes[2]. Moreover, some of the data is missing or filtered out due to error codes or other occurrences which is explained later in Section 3.3.

3.3 Pre-processing of measured time series

This section describes how obtained data is processed before using it for the validation of the Windpowerlib in this thesis.

3.3.1 Aggregation and filtering

Wind farm feed-in time series are attained from the data mentioned in Section 3.2 by aggregating the provided power output of the wind farms' wind turbines. Only if the power output for all turbines is available and if for none of the turbines an error is indicated by the error code a time step is considered in the aggregation. As wind turbines have an energy consumption at stand still the measured feed-in time series contain negative values. This is not considered in the Windpowerlib and the validation concentrates on the functionalities existent in the model. Therefore, negative power output values are filtered out.

In Section 3.2 it was mentioned that for the wind farms BE, BNW and BS error codes are available for each time step. During the pre-processing of the data all time steps belonging to error codes not indicating a warning but an error are filtered out. Moreover, duplicated time steps detected in some of the time series of WF BE, WF BNW and WF BS are dropped and one single time step is kept[3]. Apart from that, some of the time steps of the time series of the same wind farms contain zeros for all values of one wind turbine which is assumed to represent an error. Thus, these values are filtered and the wind farm power output of these time steps is not considered in the simulations. Furthermore, implausible high power output is detected in some time steps which are cleaned by excluding power outputs that are more than five percent higher than the nominal power of the wind turbines.

[2]The resampling is done by taking the arithmetic mean of the values in the respective time period.

[3]apart from one time step in WF BE of which the duplicates contain ambiguous values which is why it is not considered in the calculations

After having processed the measurement data of WF BE, BNW and BS the amount of time steps listed in Table 3.4 is available for the simulations before changing the temporal resolution.

Table 3.4: *Amount of available time steps per year in wind farm feed-in time series after the processing*

Wind farm	2015	2016
BE	30760	30432
BNW	33646	35660
BS	28072	24196

3.3.2 First row time series

For the validation of some of the functionalities (see Sections 4.2.3 and 4.2.3) and of the single wind turbine model (see Section 4.2.4) wind speed and power output time series of wind turbines not influenced by other wind turbines are necessary. These time series are attained by choosing wind turbines standing in the first row of a wind farm depending on the measured wind direction. This is done for the wind farms BE, BNW and BS as information is provided that measured wind directions are correct and as correlations with and deviations from nacelle positions were evaluated (see Section 3.2). These time series will be referred to as *first row power output* and *first row wind speed time series* (or *data*) in the remaining work. To attain the first row wind speed time series of a wind farm for each bin of 45° a wind turbine that would not be influenced by other wind turbines is found. All time steps of a measured wind speed time series of a wind turbine at which the measured wind direction fits in the wind direction bin the wind turbine stands for are used for the first row time series. As the measured wind directions of all wind turbines are not identical for each time step some time steps are created by the average wind speed of several wind turbines with fitting wind directions. First row power output time series are created in the same way.

Before this processing is done, the measured wind directions of all wind turbines are evaluated considering their correlation by using the Pearson correlation coefficient as defined in Section 4.2.1. The wind direction time series of two turbines of WF BE show weak correlations with the other wind direction time series of this wind farm and are, therefore, not utilized. Due to that, wind direction bins from 0° to 90° and from 180° to 225° are not considered for this wind farm. Furthermore, negative wind directions at one wind turbine of WF BS lead to weak correlations. It

is assumed that the measurement device wrote down negative values when the wind direction jumped from 0° to wind directions lower than 360°. Therefore, these values are adjusted by adding 360° which results in plausible correlations (see Table A.1 in the Appendix A.1). Table 3.5 lists the amount of time steps available after the processing for the simulations with first row time series before changing the temporal resolution.

Table 3.5: *Amount of available time steps per year in first row feed-in and wind speed time series after the processing*

Wind farm	Data	2015	2016
BE	power output	21998	22687
	wind speed	27265	28706
BNW	power output	35483	38183
	wind speed	46617	51336
BS	power output	31363	29434
	wind speed	40343	40606

3.4 Wind turbine data

For the simulations in this thesis the following turbine data is needed:

- Power curve or power coefficient curve
- Turbine hub height
- Rotor diameter

The Windpowerlib provides power curves of 158 and power coefficient curves of 91 turbine types. All turbine types relevant for this work can be found there. Hub heights and rotor diameters are taken from data sheets if they are not given in the master data of the measured feed-in time series.

Chapter 4

Methodology

The methodology of this work is divided into two mayor topics. Section 4.1 presents the implementation of the open source model while the methodology for the validation of the Windpowerlib is depicted in Section 4.2.

4.1 Implementation of the open source model

This section deals with the implementation of the Windpowerlib. Section 4.1.1 introduces the general structure of the model while Section 4.1.2 depicts the implementation of functionalities. Section 4.1.3 presents default models for an easy start into working with the Windpowerlib and Section 4.1.4 shows an overview of the required input data. A detailed description of all functions, classes and all their parameters can be found in the documentation which is hosted at Readthedocs[1], a web platform for documentations. The version of the Windpowerlib developed in this thesis can be found in the developer branch on Github[2].

4.1.1 General structure

As mentioned earlier in Section 1.2 the Windpowerlib is aimed to be constructed as a flexibly applicable library. Therefore, diverse functions and classes for modelling wind power plants and calculating their power output are implemented in a way that they can be utilized independently.

The `basicmodel` module of Windpowerlib v0.0.4 (see Section 1.2) is turned into the module `wind_turbine` and the functions for wind speed and density calculations

[1] http://Windpowerlib.readthedocs.io/en/latest/
[2] https://github.com/wind-python/windpowerlib/tree/9bca91f

are moved to separate modules. All functionalities are gathered within the following modules concerning topics:

- `wind_turbine`
- `wind_farm`
- `wind_turbine_cluster`
- `tools`
- `wind_speed`
- `temperature`
- `density`
- `power_output`
- `power_curves`
- `wake_losses`
- `modelchain`
- `turbine_cluster_modelchain`

The `wind_turbine` module consists of the class `WindTurbine` which models a wind turbine as well as functions that read and restructure wind turbine data like power curves or Cp-curves. Additionally, it contains the function `get_turbine_types()` which lists all turbine types of which the power curve or Cp-curve is provided within the Windpowerlib.

A wind farm can be modeled with the class `WindFarm` in the `wind_farm` module. Additionally, this module contains functions for the calculation of its mean hub height and installed power as well as the function `power_curve` for generating an aggregated wind farm power curve. Similarly the `wind_turbine_cluster` module contains the class `WindTurbineCluster` representing a cluster of several wind farms (belonging to the same weather data point) and functions for the calculation of its mean hub height, installed power and aggregated power curve. The implementation of aggregated power curves is further described in Section 4.1.2.

In the `tools` module functions providing tools for the functionalities of the Windpowerlib are gathered. The modules `wind_speed`, `temperature` and `density` contain functions to correct the respective value to be valid at hub height of a wind turbine. These, the `power_output` and the `power_curves` module containing functions for power output calculations and respectively for power curve alterations and

the `wake_losses` module containing functions for modelling wake losses are further described in Section 4.1.2.

Additional modules called `modelchain` and `turbine_cluster_modelchain` are constructed to ensure an easy start into the Windpowerlib. They contain the classes `ModelChain` and `TurbineClusterModelChain` which work like models that combine all functions provided in the library. Flow charts of these models and further explanations can be found in Section 4.1.3.

4.1.2 Implementation details

Height correction and conversion of weather data

In this section the functionalities for height corrections and conversion of weather data are presented ordered by the modules they are implemented in.

`tools`: In this module a function implementing a linear inter-/ extrapolation using Equation (4.1) can be found. This function can be used whenever values are known at two different heights. As the wind speed follows rather a logarithmic profile than a linear function a logarithmic inter-/ extrapolation as shown in Equation (4.2) is provided, as well.

$$f(x) = \frac{(f(x_2) - f(x_1))}{(x_2 - x_1)} \cdot (x - x_1) + f(x_1) \tag{4.1}$$

$$f(x) = \frac{\ln(x) \cdot (f(x_2) - f(x_1)) - f(x_2) \cdot \ln(x_1) + f(x_1) \cdot \ln(x_2)}{\ln(x_2) - \ln(x_1)} \tag{4.2}$$

`wind_speed`: The following possibilities are implemented for the height correction of wind speed to the hub height of a wind turbine.

Logarithmic wind profile	→	`logarithmic_profile()`
Hellman equation	→	`hellman()`
Logarithmic inter- or extrapolation	→	`logarithmic_interpolation_extrapolation()` (`tools`)

The logarithmic wind profile is implemented in the function `logarithmic_profile()` by Equation (2.2).

To the function `hellman()` in which the Hellman equation is implemented by Equation (2.3) a Hellman exponent can be passed. If no Hellman exponent but a roughness length is passed, the Hellman exponent is calculated via Equation (2.4). If neither a Hellman exponent nor a roughness length is provided, a Hellman exponent of 1/7 is assumed.

Alternatively, the wind speed at hub height can be determined by using the function `logarithmic_interpolation_extrapolation()` of the `tools` module if wind speeds are available for at least two different heights.

`temperature`: For temperature height correction the following functions can be used.

Linear gradient	→	`linear_gradient()`
Linar inter- or extrapolation	→	`linear_interpolation_extrapolation()` (`tools`)

The `linear_gradient()` function of the Windpowerlib assumes a standard atmosphere [Deutscher Wetterdienst, b] and calculates the temperature at hub height via Equation (2.7). To apply the `linear_interpolation_extrapolation()` function of the `tools` module input data has to provide temperature data for at least two different heights.

`density`: The following functions for the height corrections of air density are provided.

Barometric height equation	→	`barometric()`
Ideal gas equation	→	`ideal_gas()`
Linar inter- or extrapolation	→	`linear_interpolation_extrapolation()` (`tools`)

The barometric height equation is implemented in the function `barometric()` as described in Equation (2.5) and the ideal gas equation in the function `ideal_gas()` like shown in Equation (2.8). In both functions the pressure at hub height is calculated via Equation (2.6). As a third possibility the `linear_interpolation_extrapolation()` function of the `tools` module can be used. Equally to the height corrections of wind speed and temperature density data has to be provided at least at two heights to apply this function.

Power output calculations

The `power_output` module is divided into the following two main functionalities of which the `power_curve()` function can be used in combination with a density correction.

power coefficient curve	→	`power_coefficient_curve()`
power curve	→	`power_curve()`

If the power output of the examined wind turbine is calculated by its power coefficient curve Equation (2.9) is used. For the power output calculations by using a power curve the output of a certain wind speed is gained by interpolating between the wind speeds of the power curve. The density correction is implemented as defined in Equations 2.10 and 2.12 and is carried out if the parameter `density_correction` of the `power_curve()` function is set to *True*.

Wake losses

In Section 1.2 it was outlined that the Windpowerlib will rather be applied for calculating the feed-in of larger areas than for single wind farms. As explained in Section 2.3.3 using one of the explicit or implicit wake models listed in that section is very time intensive for larger nodes and often not viable due to an incomplete data basis. Therefore, wake losses are taken into consideration by implementing the following options for considering wake losses.

wind efficiency curve	→	`get_wind_efficiency_curve()`	(`wake_losses`)
	→	`reduce_wind_speed()`	(`wake_losses`)
constant efficiency	→	`wake_losses_to_power_curve()`	(`power_curves`)
power efficiency curve	→	`wake_losses_to_power_curve()`	(`power_curves`)

The Windpowerlib provides the mean wind efficiency curves of the "dena-Netzstudie II" [Kohler et al., 2010] and Knorr [2016] that depict the wind speed reduction within a wind farm (see Section 2.3.3). The functions for the usage of wind efficiency curves are located in the module `wake_losses`. A wind efficiency curve can be fetched by the function `get_wind_efficiency_curve()` and all curves can be displayed using `display_wind_efficiency_curves()`. The same module contains a function

`reduce_wind_speed()` that applies a wind efficiency curve on a wind speed time series.

In addition to that, the option of defining a constant wind farm efficiency, that is applied to the power curve of a wind farm, is implemented in the function `wake_losses_to_power_curve()` in the `power_curves` module. In this function the values of a wind farm power curve are reduced by the respective efficiency. However, as mentioned in Section 2.3.3 Barthelmie et al. [2013, p. 1017] found out that the wind farm efficiency changes significantly with the wind speed. Thus, the application of power efficiency curves (wind farm efficiency depending on the wind speed) is implemented, as well. Like the constant efficiency power efficiency curves are applied to wind farm power curves. This has the advantage that they can further be aggregated to achieve turbine cluster power curves as explained later.

Smoothing of power curves

To account for the spatial distribution of wind speeds a function for smoothing power curves (`smooth_power_curve()`) is implemented in the `power_curves` module. This function applies a Gauss distribution to power curves following the approach of Nørgaard and Holttinen [2000] like it is interpreted by Knorr [2016] (see Section 2.4.1) and shown in Equation (2.21). The wind speed range and its block width (interval length Δv_i in Equation (2.21)) are parameters of the function (`wind_speed_range` and `block_width`) with default values of 15 m/s for the wind speed range and 0.5 m/s for the block width. Further parameters are the mean μ of the Gauss distribution and a specification of the method for getting the standard deviation σ (Gauss distribution). Options for the standard deviation are calculating it via a turbulence intensity (TI method) by Equations 2.22 and 2.24 and a method proposed by Staffell and Green [2014] that is shown in Equation (2.23) (SP method). The calculation via a turbulence intensity is set as default. Decisions for the default values were made based on Section 5.1.3.

Aggregated wind farm and turbine cluster power curves

As mentioned in Section 4.1.1 wind farm and turbine cluster power curves are attained by aggregating wind turbine (or respectively wind farm) power curves. This is implemented in the `wind_farm` module for wind farms and in the `wind_turbine_cluster` module for clusters in the equally named functions `assign_power_curve()`. Depending on the parameters the aggregated power curve

is smoothed as well as a wind farm efficiency (power efficiency curve or constant efficiency) can be applied. When a turbine cluster is modeled the smoothing can either be applied to the wind farm power curves or to the turbine cluster power curve while wake losses are always applied to the wind farm power curves.

For the power output calculations with these aggregated power curves the mean hub height of the wind farm or respective the cluster has to be calculated which is implemented by Equation (2.27) in the classes `WindFarm` and `WindTurbineCluster`. For the mean hub height of clusters the wind turbine specific values (WT) are exchanged with the corresponding values of the wind farms contained in the clusters.

4.1.3 The modelchains: an easy start into simulations

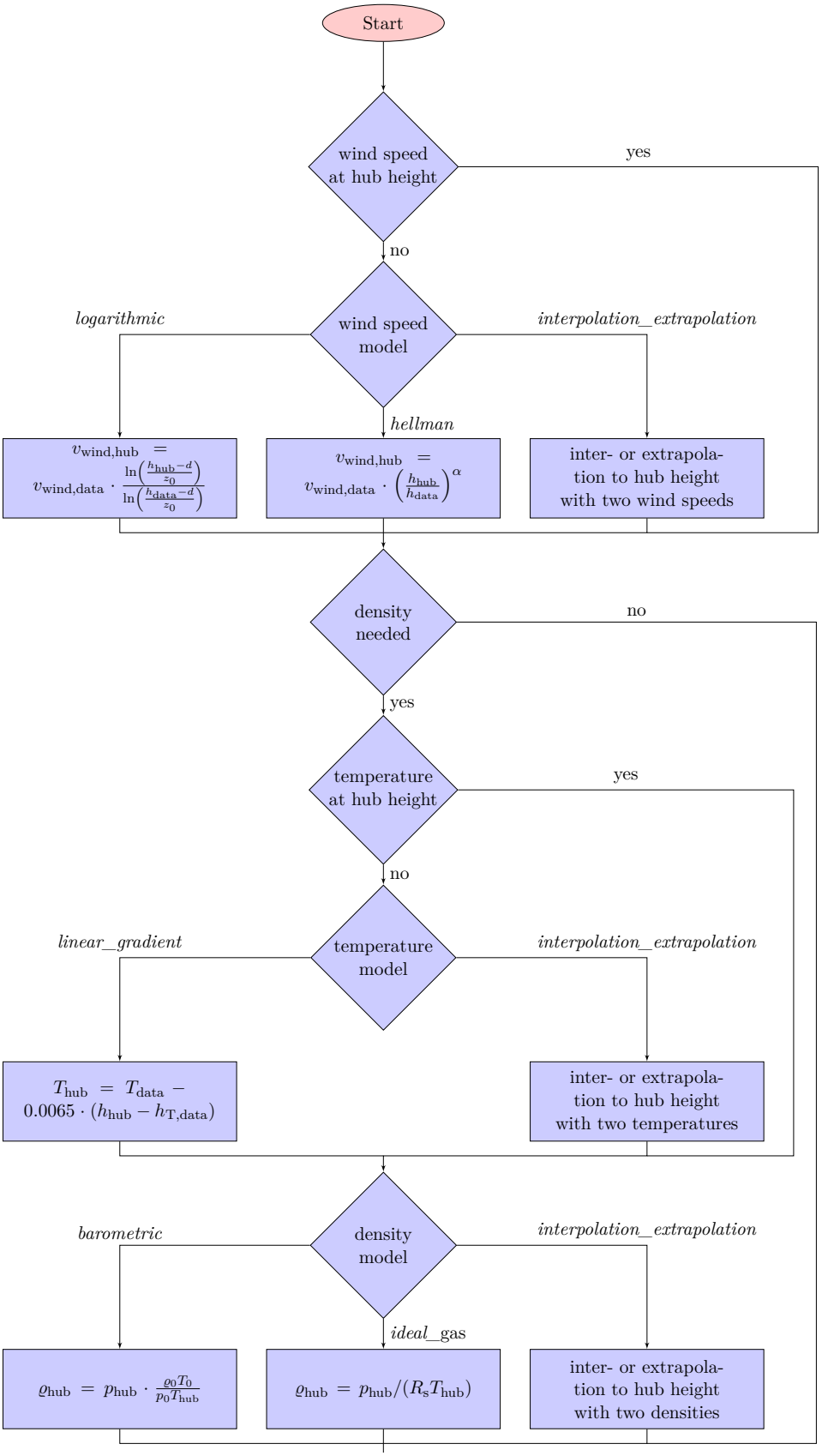
As mentioned in Section 4.1.1 the `modelchain` and the `turbine_cluster_modelchain` modules are implemented to ensure an easy start into the Windpowerlib. They work like models that combine all functions provided in the library. Their functioning is explained in this section by using flow charts.

Modelchain for single wind turbines

The `modelchain` module contains the class `ModelChain` which provides a simple model that calculates the power output of a wind turbine. The class is initialized with a `WindTurbine` object (see Section 4.1.1) and certain parameters describing the desired usage of the Windpowerlib. For parameters that are not specified default parameters are used. All parameters and their default values are described in the documentation of the Windpowerlib[3]. The model is started by carrying out the function `run_model()` which requires a MultiIndex data frame containing weather data as input parameter. How such a MultiIndex data frame can be created is explained in an example in the documentation of the function. The flow chart of the `ModelChain` presented in Figure 4.1 shows the behavior of the model depending on the parameters. When running the model it is first checked whether wind speed data is available at hub height. If the wind speed data has to be corrected to hub height a height correction is carried out depending on a parameter specifying which function to use. After that, it is checked whether density is needed for further calculations which is the case if a density corrected power curve or Cp-curve is used for power output calculations. Methods for density, temperature and power output

[3] http://Windpowerlib.readthedocs.io/en/latest/

calculations are carried out depending on the parameters as shown in the flow chart. Finally, the calculated power output is assigned to the `WindTurbine` object.



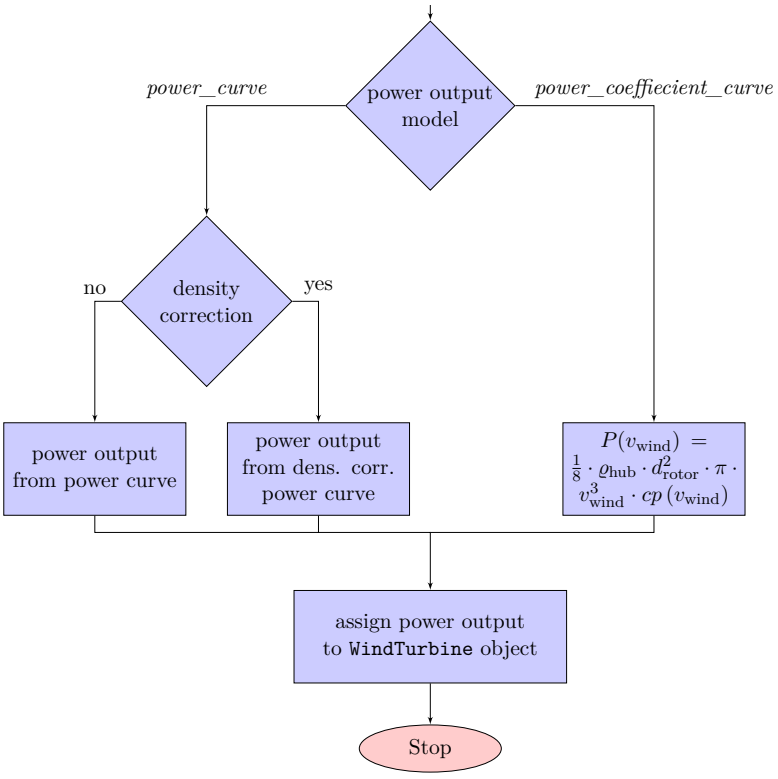


Figure 4.1: *Flow chart of the* `ModelChain`

Modelchain for wind farms and clusters

The `TurbineClusterModelChain` is a subclass of the `ModelChain` which is why it comprises the same parameters and functions to which wind farm and cluster specific parameters are added. Similarly to the `ModelChain` class an object of the class `TurbineClusterModelchain` is initialized with a `WindFarm` or a `WindTurbineCluster` object (see Section 4.1.1) and parameters for which default values are used if not specified. Like the `ModelChain` the model of the `TurbineClusterModelChain` is started by a function `run_model()` using a MultiIndex data frame containing weather data as input parameter. Figure 4.2 shows the behavior of the model in a flow chart. At first it is checked whether the power plant with which the class was initialized is a wind farm or a cluster. The difference between them is that for a cluster wind farm power curves of each wind farm are generated before aggregating them to a cluster power curve while for a wind farm its power curve is assigned to the object directly. As can be seen in the flow chart this is done by the function `assign_power_curve()` of the `WindFarm` module, which is illustrated in Figure 4.3.

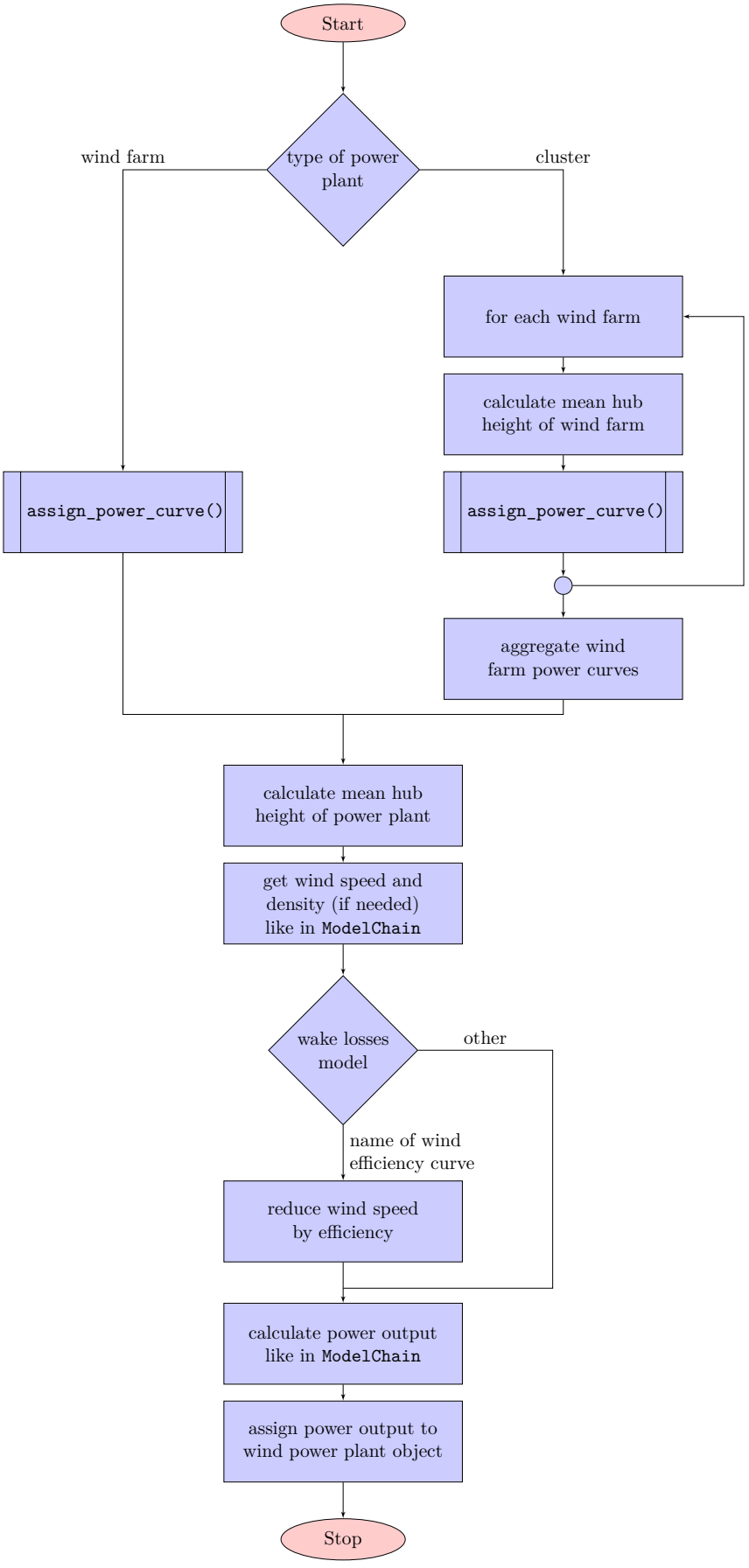


Figure 4.2: *Flow chart of the* `TurbineClusterModelChain`

Within this function wind turbine power curves are aggregated to a wind farm power curve. After that, depending on the parameters smoothing and wake losses are applied to the power curve which is then assigned to the `WindFarm` object. After the generation of a wind farm or cluster power curve the mean hub height of the power plant is calculated (see Figure 4.2). It follows a height correction of weather data like depicted for the `ModelChain` class in Figure 4.1. If the name of a wind efficiency curve is given as method for considering wake losses wind speeds are reduced by this wind efficiency curve. Then, the power output of the power plant is calculated as in the `ModelChain` (see Figure 4.1). Finally, the calculated power output is assigned to the power plant object.

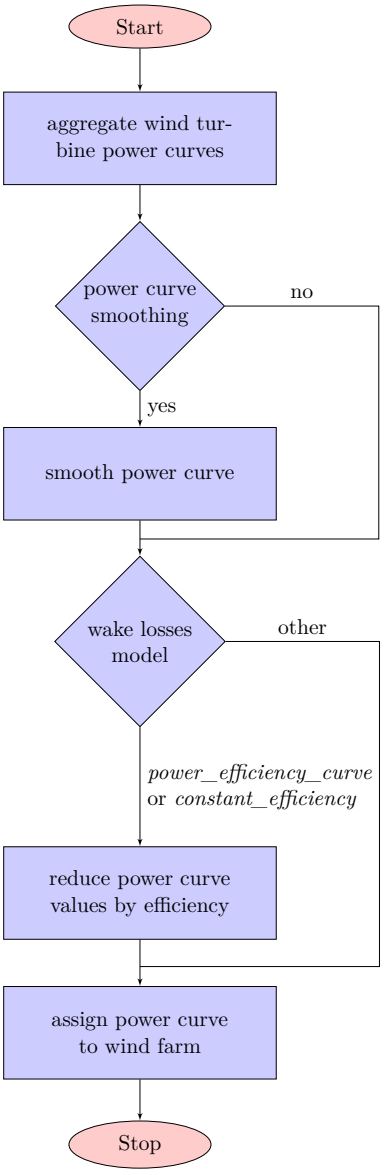


Figure 4.3: *Flow chart of the functionality* `assign_power_curve()` *of the* `WindFarm` *class*

4.1.4 Required input data

As shown in the illustration of the input and output parameters of the Windpowerlib in Figure 4.4 weather and turbine data are necessary input data for the calculation of power output time series. The weather data must include a wind speed time

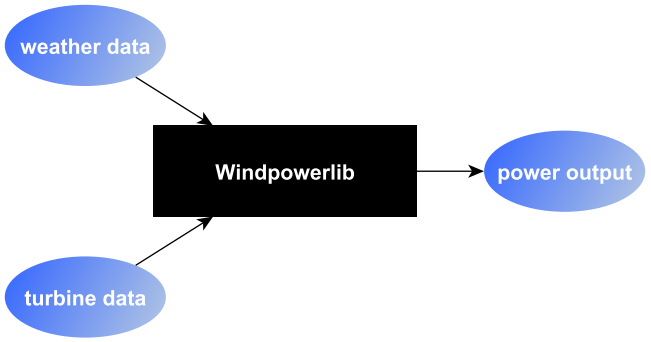


Figure 4.4: *Illustration of input and output parameters of the Windpowerlib*

series and depending on the functions being used roughness length, temperature and pressure (or density) data has to be added. In addition to the weather data the corresponding heights at which the respective data was measured or modeled has to be provided. If the `ModelChain` is used the weather data has to be stored in a MultiIndex data frame the creation of which is explained in the documentation of the Windpowerlib. If, however, the simulation is run with single functions this is not necessary as they only require time series (that can be arrays or series) of the respective weather data.

Concerning the turbine data the hub height of a simulated wind turbine (or wind farm or wind turbine cluster) is required in all cases whereas rotor diameters are only necessary for calculations with Cp-curves (see Table 4.1). A power or Cp-curve is necessary for the respective functionality. Power curves of 158 and Cp-curves of 91 turbine types are provided in the Windpowerlib. An overview of all provided turbine types can be retrieved by using the function `get_turbine_types()` of `the wind_turbine` module.

Table 4.1 shows input data that is required for the respective functions. Apart from the named functions for height corrections of wind speed, temperature and density a function carrying out linear inter- or extrapolation can be used as well. For this function the respective weather data is required for at least two heights. For wind speed height correction with the Hellman function roughness length and Hellman exponent, which can be estimated with the roughness length, are optional. If both

are not given a Hellman exponent of 1/7 is assumed. For smoothing power curves a roughness length is needed if a turbulence intensity should be calculated for the standard deviation of the Gauss function (see Equation (2.24)), however, is not necessary for a standard deviation by Equation (2.23) called Staffell_Pfenniger (SP) method.

Table 4.1: *Overview of the functions of the Windpowerlib and the required data*

Functionality	**wind speed**	**roughness length**	**temperature**	**density**	**pressure**	**rotor diameter**
wind speed						
logarithmic profile	✓	✓	-	-	-	-
hellman	✓	optional	-	-	-	-
density						
barometric	-	-	✓	-	✓	-
ideal gas equation	-	-	✓	-	✓	-
temperature						
linear gradient	-	-	✓	-	-	-
power output						
power curve	✓	-	-	-	-	-
power curve density corr.	✓	-	-	✓	-	-
Cp-curve	✓	-	-	✓	-	✓
power curve alterations						
smooth power curve	-	optional	-	-	-	-
wake losses	-	-	-	-	-	-

4.2 Validation of the Windpowerlib

For the validation of the Windpowerlib simulated feed-in time series are compared with measured feed-in time series. Apart from feed-in time series wind speed time series are validated by using measured wind speed data. Statistical metrics for the evaluation of simulation results are introduced in Section 4.2.1, Section 4.2.2 deals with the temporal and spatial resolution of simulated time series and Sections 4.2.3 to 4.2.6 focus on the description of simulations carried out for the validation. The simulations are carried out with the version of the Windpowerlib developed in this thesis which can be found in the developer branch of the Windpowerlib on Github[4].

[4]https://github.com/wind-python/windpowerlib/tree/9bca91f

4.2.1 Statistical metrics

Pearson correlation coefficient

The Pearson correlation coefficient (in the literature often: Pearson's r or Pr) is used in many studies listed by Sharp [2015, p. 59] to describe the correlation of simulated with measured time series. Fahrmeir, Heumann, Künstler, Pigeot, and Tutz [2016, p. 126 ff.] explain the nature and the applications of this statistical metric and the following explanations are taken from their work.

The Pearson correlation coefficient Pr specifies how strong the linear correlation is between two variables or in the case of this thesis between two time series. It's definition is shown in Equation (4.3). The denominator normalizes the equation whereas the numerator evolves from the sum of the products of the deviations between time series steps (x_i and y_i) and their averages ($\overline{x}$ and $\overline{y}$).

$$Pr = \frac{\sum_{i=1}^{n}(x_\mathrm{i} - \overline{x})(y_\mathrm{i} - \overline{y})}{\sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_\mathrm{i} - \overline{x})^2(y_\mathrm{i} - \overline{y})^2}} \tag{4.3}$$

The Pr can reach values between -1 and 1. A Pr of zero points out that there is no correlation between the compared time series, $Pr = 1$ indicates a maximum positive correlation and $Pr = -1$ a maximum negative correlation.

Fahrmeir et al. [2016, p. 130] propose a classification of the values between 0 and 1 as follows, however, state that this is used for rather exact measurements.

$$\begin{aligned} \text{"weak correlation"} &\rightarrow |Pr| < 0.5 \\ \text{"middle correlation"} &\rightarrow 0.5 <= |Pr| < 0.8 \\ \text{"strong correlation"} &\rightarrow 0.8 <= |Pr| \end{aligned} \tag{4.4}$$

As the Pearson correlation coefficient takes a linear correlation but not a potential bias into consideration additional metrics as the bias and the root mean squared error are as well considered in this thesis.

Bias

Two time series correlating strongly with each other can still imply a bias which results from a deviation between the values although the course of the time series is

similar. The bias of one time series concerning another time series is a time series itself. To achieve one single index per time series for a comparison of several time series in this thesis a mean bias is calculated. It is defined as an arithmetic mean like declared in Equation (4.5) where b_{avg} is the mean bias, b_i the bias of element i of the time series and n the number of time steps.

$$b_{avg} = \frac{1}{n} \sum_{i=1}^{n} b_i \tag{4.5}$$

As in the sum of Equation (4.5) positive and negative values can balance out the root mean squared error described in the next section is considered, as well, in this thesis.

Root mean squared error

In contrast to the bias, introduced in the last section, the root mean squared error (RMSE) adds up squared errors which prevents negative and positive values to balance out. The RMSE is calculated by Equation (4.6) where x_i is the i^{th} element of one time series, y_i the i^{th} element of the other time series and n the amount of time steps considered. As the root is taken of the sum of the squared errors the RMSE has the same unit as the values of the time series.

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (x_\mathrm{i} - y_\mathrm{i})^2} \tag{4.6}$$

To be able to compare the RMSE of time series from different wind farms a relative RMSE is defined in Equation (4.7) where $RMSE$ is the RMSE from Equation (4.6) and $mean$ the mean value of the time series.

$$RMSE\ [\%] = \frac{RMSE}{mean} \cdot 100 \tag{4.7}$$

Standard deviation

To measure the distribution of values of a time series around its mean value the standard deviation is used. It is obtained by the root of the variance σ^2 which is shown in Equation (4.8). The standard deviation σ is defined in Equation (4.9). In

both equations $\overline{x}$ is the mean value of the examined time series, x_i the i^{th} element of the same time series and n its amount of time steps.

$$\sigma^2 = \frac{1}{n}\sum_{i=1}^{n}(x_\mathrm{i} - \overline{x})^2 \tag{4.8}$$

$$\sigma = \sqrt{\sigma^2} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_\mathrm{i} - \overline{x})^2} \tag{4.9}$$

Like the RMSE the standard deviation has the same unit as the observed values. As the deviations from the mean value are squared they are all included as a positive value which means that like for the RMSE positive and negative values do not compensate each other. [Fahrmeir et al., 2016, p. 65]

4.2.2 Temporal and spatial resolution of the examined time series

The time series considered in this thesis are of different temporal resolutions. As known from Table 3.1 the temporal resolutions of the weather data time series are 30 and 60 minutes whereas the resolutions of the measured feed-in time series range from one to ten minutes (see Table 3.3). When it comes to the validation it makes sense to change the temporal resolutions of all time series to the lowest resolution which would be 60 minutes. However, as in this thesis the influence of the input data is evaluated, both half-hourly and hourly time series are considered while half-hourly time series are only used for simulations with open_FRED weather data. The target resolution is attained by averaging time series with higher resolutions. To get an idea about the canceling out of errors over time monthly mean values (monthly resolution) are examined, too. For the evaluation of a yearly mean the annual energy output of simulated and measured time series is compared which is identical with an annual mean power output multiplied by time. The averaging from higher to lower temporal resolutions only takes place for more than 50 % available values within the target time period. For instance to achieve a temporal resolution of 60 min from a time series with a resolution of ten minutes at least four time steps have to be available (no error, not missing). In some cases of the first row time series (see Section 3.3.2) not enough time steps are available for attaining the monthly resolution.

Concerning the spatial resolution of time series the selection is limited by the available measured feed-in time series. To validate the power output calculations for a single wind turbine first row time series are considered. Furthermore, the calculated

power output calculations of wind farms of different sizes which were shown in Table 3.2 are validated. As each of these wind farms is situated in a different region and, thus, is assigned to a different weather data point, the aggregated power output of a wind turbine cluster is not validated.

4.2.3 Single functionalities

This section describes simulations that are carried out for the validation of single functionalities of the Windpowerlib. Functionalities calculating the air density are not evaluated as appropriate data is not available.

Wind speed height correction functionalities

Researchers found out that the performance of the logarithmic wind profile and the Hellman equation depends on site conditions which was stated in Section 2.1.1. As the choice of measurement data available in this work (see Section 3.2) is restricted the performance of these methods is not specifically evaluated. It is rather focused on their performance depending on the height at which wind speed data is available. Wind speed data is often available at ten meters above ground [Sharp, 2015, p. 82], MERRA provides wind speeds at 50 m and the open_FRED weather data includes wind speeds at various heights that are calculated with a complex climate model. The wind speed height correction functionalities implemented in the Windpowerlib are less complex than calculations within climate models. It is aimed to analyze the results of these simplified calculation methods when extrapolating wind speeds from different heights (calculated in the climate model) to hub heights.

For the examination wind speed height correction functionalities of the Windpowerlib are applied on open_FRED wind speed data at 10 m, 80 m and 100 m above ground. The calculated wind speed at hub height is validated with first row wind speed data that was derived from the measurements at the wind farms BE, BS (105 m hub heights) and BNW (60 m) in east, south and north-west Brandenburg (see Table 3.2 and Section 3.3.2). In addition to that, the performance of a logarithmic interpolation between wind speeds of different data heights is compared with the other functions.

Table 4.2 shows the characteristics of the simulations that are done for these evaluations. The Hellman equation is applied with α calculated by Equation (2.4) (H) and with $\alpha = 1/7$ (H2), which is assumed in many studies (see Section 2.1.1).

Table 4.2: *Characteristics of simulations run for the validation of wind speed functionalities*

Name	Function	Special parameters	Weather data height [m]
Log 100	`logarithmic_profile()`	-	100
Log 80	`logarithmic_profile()`	-	80
Log 10	`logarithmic_profile()`	-	10
H 100	`hellman()`	-	100
H 80	`hellman()`	-	80
H 10	`hellman()`	-	10
H2 100	`hellman()`	$\alpha = 1/7$	100
H2 80	`hellman()`	$\alpha = 1/7$	80
H2 10	`hellman()`	$\alpha = 1/7$	10
Log. Interp.	`logarithmic_interpolation_extrapolation()`	-	100, 120 or 10, 80 (BNW)

Power output calculation functionalities

The power output calculation functionalities of the Windpowerlib are validated by using measured wind speeds instead of reanalysis wind speeds from the weather data sets for the calculations. This has the advantage of separating errors caused by the weather models and by height corrections from the performance evaluation of the power output functionalities. The wind speeds are taken from the measurement data of three wind farms in Brandenburg (WF BE, WF BNW and WF BS; see Table 3.2). According to the provider of this data the wind speeds were measured at the nacelles of the wind turbines and corrected to represent wind speeds in front of the rotors (see Section 3.2). This way the power output methods are validated with data from 24 turbines and two years. Table 4.3 shows an overview of these simulations that are carried out with both weather data sets. For the Cp-curve and density corrected power curve approach the density is calculated with the ideal gas equation. During tests in the Windpowerlib no great difference was detected between the ideal gas equation and the barometric height equation. The temperature at hub height necessary for the density calculations is calculated with a linear gradient (see Section 4.1.2) to use the same equation for MERRA and open_FRED data[5].

Table 4.3: *Characteristics of simulations run for the validation of power output functionalities*

Name	Function	Density correction	Weather data
P-curve	`power_curve`	no	-
Cp-curve	`power_coefficient_curve`	no	open_FRED, MERRA
P (d.-c.)	`power_curve`	yes	open_FRED, MERRA

[5]For open_FRED data a linear interpolation between two heights could be used but MERRA data only contains one data height.

Smoothing of power curves

Before validating the functionality for smoothing power curves its parameters are analyzed and appropriate values are chosen. Whenever weather data is necessary for the simulations (for instance TI method, see Section 4.1.2) they are carried out with open_FRED weather data.

Parameters At first the parameter `wind_speed_range` for which the sum in Equation (2.21) is taken and its block width (parameter `block_width`) of the function are evaluated. Then two options of calculating the standard deviation of the Gauss distribution are examined.

As mentioned in Section 2.4.1 Nørgaard and Holttinen [2000, p. 3] recommend to use a wind speed block range from at least -5 m/s to +5 m/s. Therefore, this range and additionally the following are tested on the power curves:

- -10 m/s to +10 m/s
- -15 m/s to +15 m/s
- -20 m/s to +20 m/s

For decisions about the block width power curves smoothed with values of 0.1 m/s, 0.5 m/s and 1 m/s are examined. Furthermore, power curves are smoothed with a normalized standard deviation calculated by a TI (Equations 2.22 and 2.24) and with a normalized standard deviation calculated by the SP method shown in Equation (2.23).

Validation Looking at the smoothed power curve that was shown in Figure 2.3 it might be assumed that the application of this method would reduce the calculated power output. However, for low wind speeds the power output of the smoothed power curve lies above the original power curve[6]. Therefore, it depends on the prevailing wind speeds whether power curve smoothing would rather induce over- or underestimation of power outputs. For the later validation of the wind farm model (see Section 4.2.5) it is convenient to analyze whether the application of smoothed power curves has a positive effect on the simulation results compared to simple aggregation. Furthermore, as mentioned in Section 2.4.1 the fluctuation

[6]This difference can be higher than in Figure 2.3 depending on the methods used.

(standard deviation) of simulated time series is supposed to be reduced by power curve smoothing. It is examined whether the fluctuation of measured time series are represented well by the simulated time series.

For these purposes three simulations that are illustrated in Table 4.4 are carried out for the five wind farms listed in Table 3.2. Two simulations use the standard deviation methods TI and SP. A third simulation that simply aggregates the single wind turbine power outputs is done for comparison. Wind speed data for these simulations is taken from the open_FRED weather data set to look at the fluctuations resulting from the power curve smoothing in combination with the wind speed data being used in the wind farm model later. Wind speed height corrections are done with the logarithmic wind profile in accordance with the findings of the wind speed simulations in Section 5.1.1. As wind speed data height the one closest to hub height is chosen as it is assumed that most users would decide on that. For power output calculations the simplest method of using a power curve is chosen which leads to the best results when the single wind turbine model is validated (see Section 5.2).

Table 4.4: *Characteristics of simulations run for the validation of the smooth power curve function*

Name	**Calculation method**	**Standard deviation for the Gauss function**
TI	Smoothed wind farm power curve	Turbulence intensity
SP	Smoothed wind farm power curve	Staffell-Pfenninger
Agg.	Aggregated turbine power output	-

Wake losses

In Section 4.1.2 the different ways of considering wake losses in a wind farm implemented in the Windpowerlib were explained to be the following:

- Wind efficiency curve (reduction of wind speed)
- Power efficiency curve (reduction of power output)
- Constant (power) efficiency (reduction of power output)

While mean wind efficiency curves generated from wind farms in Germany (see Figure 2.1 and Section 4.1.2) are provided as data in the Windpowerlib, no average power efficiency curves could be found in the literature. To get a first impression on the appearance and the performance of power efficiency curves such curves are generated from measured feed-in data. They are calculated only for three wind

farms (BE, BNW and BS) as wind directions measured at WF BNE contain errors and no information about the correctness of wind directions measured at WF SH is available (see Section 3.2). The methodology used is depicted later. The effect of these curves, specifically created for single wind farms, on the power output is compared with the mean wind efficiency curve that was generated in the "dena-Netzstudie II" (see Figure 2.1 and Section 2.3.3). In the remaining work this curve will be called *dena mean wind efficiency curve*. As the dena mean wind efficiency curve is applied to wind speeds and the calculated power efficiency curves to power output a comparison between them is difficult. However, although the application is different it is still expected that the power efficiency curves calculated for specific wind farms would result in lower errors.

The simulations run for the comparison between the different approaches are illustrated in Table 4.5. They are carried out with open_FRED reanalysis wind speeds. These wind speeds are used instead of measured wind speeds as the latter induce high underestimations which is also detected in the single wind turbine simulations (see Section 5.2). For the evaluation of the functionalities for considering wake losses it makes sense to use reanalysis wind speeds as the overall performance of the simulations has to be evaluated. Open_FRED wind speeds are chosen at the data height closest to hub height and are corrected with the logarithmic wind profile. For the power output calculations the power curve approach is used. Reasons for that decision were given earlier when the simulations for the evaluation of power curve smoothing were described. Apart from the wind and power efficiency curves the application of a constant efficiency is evaluated. The aim of that is to evaluate whether such a simplified consideration of wake losses would lead to acceptable results. While pre-evaluating the model performance it has become clear that the application of calculated power efficiency curves and of the dena mean wind efficiency curves will result in overestimations. Therefore, a value lying in between the lower edges (not the peaks in Figure 5.9) of the curves is chosen for the constant efficiency: 80 %.

Table 4.5: *Simulations run for the evaluation of the functionalities for considering wake losses*

Name	**Efficiency**
Calc.	Calculated power efficiency curves
Const.	Constant efficiency of 80 %
Dena	Mean wind efficiency curve (dena)
No losses	No wake losses considered

The specific power efficiency curves of the wind farms BE, BNW and BS are produced by, firstly, calculating the efficiency for each time step of the measured data using

Equation (2.14). These efficiencies are, secondly, averaged in wind speed bins of 0.5 m/s. If for an "inner" wind speed bin no value is existent it is interpolated while missing values at the edges are set to an efficiency of one to achieve a curve for wind speed values from zero to 25 m/s. The value of one is chosen as for high wind speeds also wind turbines experiencing a reduced wind speed would attain nominal power. For low wind speeds the setting to a value of one only occurs for wind speeds below cut-in wind speed which do not have any effect on the power output. The final power efficiency curves are achieved by taking the mean of the power efficiency curves of the years 2015 and 2016. Time steps for which the efficiency reaches values above one, are detected as incorrect and are not further considered in the power efficiency curves. It is suspected that these values occur due to curtailment of single turbines for which no data is provided. If for example a first row wind turbine suffers curtailment while second or third row wind turbines are not the efficiency could reach values higher than one.

4.2.4 Single wind turbine model

In Section 4.2.3 the methodology for validating the single functionalities for wind speed height corrections and power output calculations was described. In contrast to these validations the single wind turbine model is validated by full simulations from reanalysis wind speeds until the power output of a wind turbine. The data used for the validation is derived from time series of the wind farms BE, BNW and BS and was processed as described in Section 3.3 to attain a power output time series representing a single wind turbine that does not experience wake losses (first row time series).

The simulations run for the validation of the single wind turbine model are shown in Table 4.6. The model is validated for both weather data sets. Concerning wind speed and density calculations the same methods as in the validation of the power curve smoothing and wake losses functionalities (see Section 4.2.3) are chosen: for the wind speed height corrections the logarithmic wind profile is used and as open_FRED wind speed data height the height closest to hub height is selected. The density, needed for the Cp-curve and density corrected power curve, is calculated with the `ideal_gas()` function for which the temperature is derived by the `linear_gradient()` function.

Table 4.6: *Simulations run for the validation of the single turbine model*

Name	Function	Density correction
P-curve	`power_curve`	no
Cp-curve	`power_coefficient_curve`	no
P (d.-c.)	`power_curve`	yes

4.2.5 Wind farm model

The simulations for the wind farm model validation are carried out with different functionalities that are listed in Table 4.7. Power efficiency curves are only available for three wind farms (see Section 4.2.3) and during the simulation of wake losses in this work their application only leads to minor improvements compared with the dena mean wind efficiency curve (see Section 5.1.4). Therefore, only the dena mean wind efficiency curve and a constant efficiency (80 % like in Section 4.2.3) are tested here. As in the evaluation of the smoothing functionalities for power curves the SP approach does not perform well (see Section 5.1.3) this method is not applied. However, the TI method leads to an improvement for some wind farms. Therefore, a combination of the dena mean wind efficiency curve with power curves smoothed with the TI method is examined. Furthermore, in order to estimate whether the application of these new functionalities improves simulation results, in one simulation the power output of the single wind turbines is simply aggregated to a wind farm power output (Agg.). All simulations described are carried out with both weather data sets. Like in the validation of the single wind turbine model the logarithmic wind profile and power curves are used for the calculations while open_FRED wind speeds are taken from a data height closest to hub height.

Table 4.7: *Characteristics of simulations run for the validation of the wind farm model*

Name	Wake losses	Smoothing
Const.	Constant efficiency of 80 %	-
Dena	Mean efficiency curve (dena)	-
Dena-TI	Mean efficiency curve (dena)	TI
Agg.	-	-

4.2.6 Influence of weather data

The evaluation of the influence of the weather data is done by firstly evaluating differences between wind speed height corrections with the two weather data sets

and secondly and thirdly by comparing the results for single wind turbine and wind farm feed-in simulations.

Concerning the wind speed height corrections simulations from Section 4.2.3 depicted in Table 4.8 are run for both weather data sets.

Table 4.8: *Characteristics of simulations run for the evaluation of the influence of weather data on wind speed calculations*

Name	**Function**	**Special parameters**	**Weather data height [m]**	
			open_FRED	**MERRA**
Log	`logarithmic_profile()`	-	100 or 80 (BNW)	50
H	`hellman()`	-	100 or 80 (BNW)	50
H2	`hellman()`	$\alpha = 1/7$	100 or 80 (BNW)	50
Log. Interp.	`logarithmic_interpolation_extrapolation()`	-	100, 120 or 10, 80 (BNW)	-

Concerning the single wind turbine model results are taken from the simulations depicted in Section 4.2.4 (Table 4.6) while respectively results for the wind farm model are taken from simulations described in Section 4.2.5 (Table 4.7).

Chapter 5

Simulation results

This chapter presents the results of the simulations described in Chapter 4. Section 5.1 deals with results from simulations for the validation of single functionalities. Section 5.2 then focuses on the results from single wind turbine simulations while Section 5.3 presents results from wind farm simulations. This chapter ends with results of simulations evaluating the influence of the weather data on simulation results in Section 5.4.

5.1 Validation of single functionalities

As mentioned before the Windpowerlib was validated by examining single functionalities as well as simulations of a single wind turbine and a wind farm. This section presents the results of the validation of single functionalities. The results of simulations with different wind speed height correction functions are depicted in Section 5.1.1 while those of simulations with different power output calculation functions are shown in Section 5.1.2. Section 5.1.3 deals with simulation results of power curve smoothing and Section 5.1.4 with those of functionalities considering wake losses.

5.1.1 Height correction of wind speed data

In this section the results of simulations analyzing the performance of the wind speed height correction functionalities when extrapolating wind speeds from different heights to hub height are presented. The simulated wind speed time series are compared with the measured first row wind speed time series (see Section 3.3.2).

Figure 5.1 shows the statistical metrics of the hourly time series calculated with different methods from different wind speed data heights averaged over the years 2015 and 2016. When comparing these statistical metrics it can not clearly be said

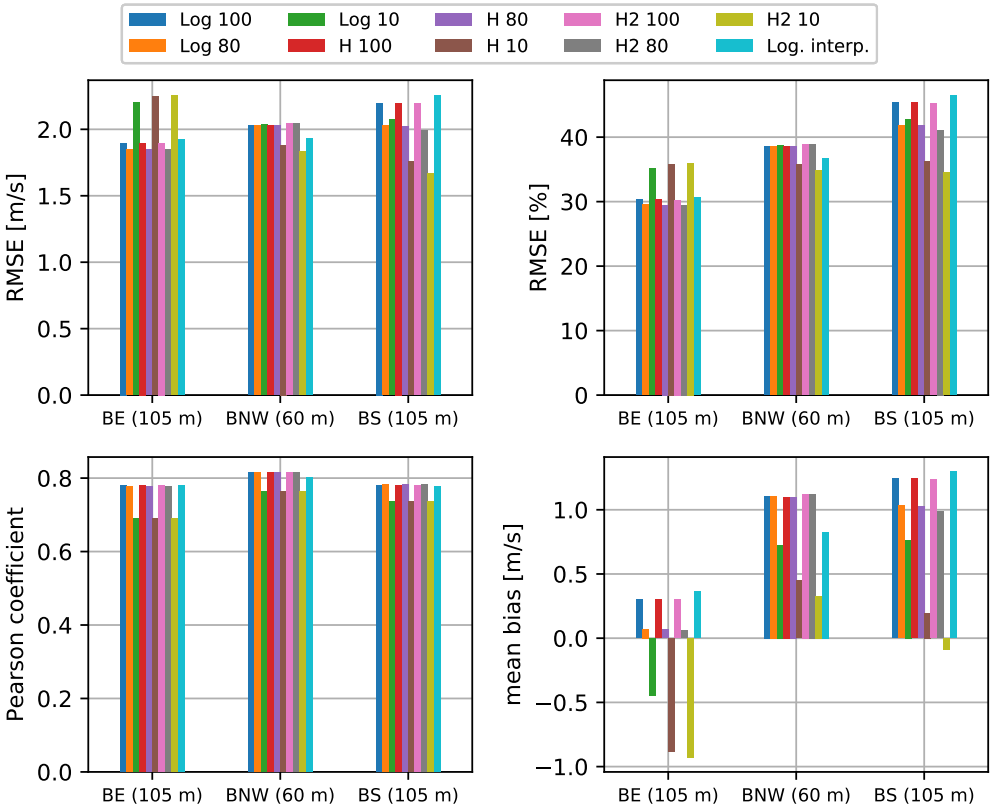


Figure 5.1: *Statistical metrics of hourly wind speed time series calculated with different methods (average of 2015 and 2016)*

that one of the wind speed heights leads to the best results. In the simulations of WF BE wind speeds from 80 m result in the lowest RMSE whereas for the other wind farms wind speeds from 10 m perform best with one exception (Log 10 of WF BS). It should be kept in mind that the hub heights of the wind turbines of WF BNW are different (60 m) from the turbine hub heights of the wind farms BE and BS (105 m). However, the calculations for the latter two wind farms do not deliver corresponding results, either. Another remarkable result is that the logarithmic interpolation (Log. interp.) between 100 and 120 m (WF BE and BS) leads to higher RMSE than other functions with weather data heights of 80 and 100 m. For wind farm BNW, that comprises turbines with hub heights of 60 m, the Log. interp. between 10 m and 80 m attains lower RMSE which might be explained by the hub heights being further away from the weather data heights.

The mean biases of the time series help to understand whether errors balance out over a period of time. In Figure 5.1 it can be seen that mean biases vary between

about -0.9 m/s and +1.3 m/s while simulations with a data height of 10 m are the only that result in negative values (WF BE and H2 in WF BS). The mean biases of WF BE are far lower than the ones of the other wind farms. This means that the errors made in simulations of WF BE vary more between negative and positive values, which can balance out over time, in comparison to the other wind farms. From Figure 5.1 it is evident that for all height correction functionalities the correlation of the time series calculated with wind speed data from a 10 m height is worse than the correlation of the time series calculated with other heights. The Pearson correlation coefficient of the simulations with a logarithmic interpolation lies in between these values or at the same level with 80 m and 100 m data height simulations with the other methods.

The correlations of the calculated hourly wind speed time series with measured time series of WF BE in 2015 are illustrated in Figure 5.2 where the logarithmic wind profile with data heights of 10 m, 80 m and 100 m and logarithmic interpolation were applied. It can be seen that the comparatively worse correlation of the simulation with a data height of 10 m origins from a underestimation of wind speeds between about two to eight m/s.

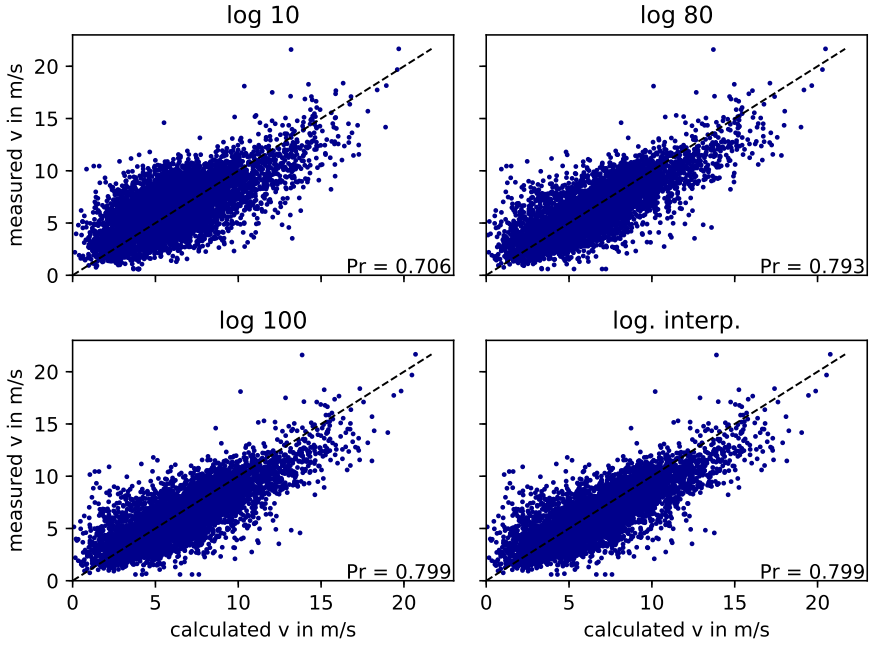


Figure 5.2: *Correlation of hourly wind speed time series calculated with different methods with measured time series for wind farm BE in 2015*

It was observed earlier that the performance of simulations with different weather data heights was ambivalent for wind farms BE and BS although the hub heights of their wind turbine are the same and they, thus, should be comparable. A possible

reason for that could be shading from other wind turbines in the surroundings of the wind farms as described in Section 3.2. Therefore, the same simulations were compared with another measured first row wind speed time series with less data points that were selected from only western (WF BS) and north-western (WF BE, WF BS) wind directions. Figure 5.3 shows the average relative RMSE from 2015 and 2016 for these simulations. It is evident that now simulations with a data height of 10 m lead to the highest RMSE for each functionality. As the available amount of time steps was strongly reduced by this selection from around 27000 (WF BE) to 40000 (WF BS) to about 8000 (WF BS) to 9000 (WF BE) (about 65 to 80 %) these time series were not used for validations in this thesis.

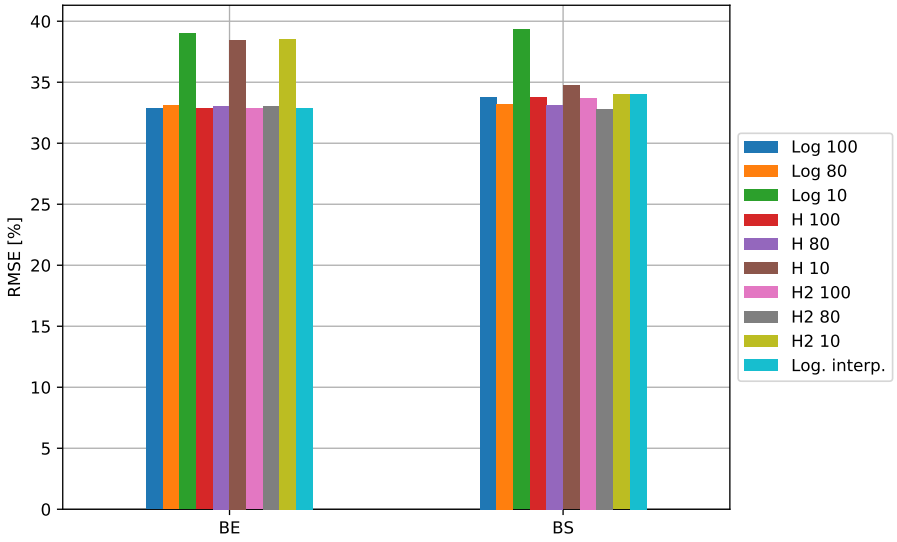


Figure 5.3: *Relative RMSE of hourly wind speed time series calculated with different methods with restricted data points averaged over 2015 and 2016*

In further simulation cases the logarithmic wind profile is used for the height correction of wind speeds as it is commonly used and the performances of the different functions are very similar. Concerning open_FRED weather data the data height closest to hub height is used as it is assumed that most users would decide on that.

5.1.2 Power output calculations

The power output calculation functionalities are validated by using measured wind speeds to exclude errors from the reanalysis wind speeds and height corrections (see Section 4.2.3). Only data needed for density calculations for the Cp-curve and density corrected power curve approach are taken from the weather data.

To compare the RMSE of the different power output calculation methods Table 5.1 shows their relative RMSE as an average taken from all 24 simulated wind turbines for hourly and monthly resolutions in the years 2015 and 2016. Simulations with Cp-curves attain the best results. The results of simulations run with density corrected power curves are slightly better than those for power curves. Compared to the hourly time series the relative RMSE of monthly time series is about eight to ten percentage points lower for the Cp-curve approach and about six to eight percentage points for the remaining approaches. This means that errors made by simulations with Cp-curves cancel out over time to a greater extend than errors of simulations with the other methods.

Table 5.1: *Relative RMSE (average of all turbines) in % of power output simulations with different functionalities*

	Temporal	Cp		P	P (d.-c.)	
Year	resolution	MERRA	open_FRED	-	MERRA	open_FRED
2015	hourly	17.126	16.556	20.965	20.808	19.908
2015	monthly	9.264	8.812	14.106	14.045	13.175
2016	hourly	19.078	18.622	23.153	22.871	22.080
2016	monthly	9.911	9.430	15.095	15.017	14.112

Table 5.2 shows the correlations of simulated with measured time series for which an average was taken from all wind turbines to get an overview. It is evident that all simulations produce strong correlations.

Table 5.2: *Pearson correlation coefficient (average of all turbines) of power output simulations with different functionalities*

	Temporal	Cp		P	P (d.-c.)	
Year	resolution	MERRA	open_FRED	-	MERRA	open_FRED
2015	hourly	0.990	0.991	0.987	0.988	0.989
2015	monthly	0.998	0.998	0.996	0.998	0.998
2016	hourly	0.988	0.989	0.986	0.986	0.987
2016	monthly	0.997	0.997	0.993	0.997	0.997

Instead of looking at the mean bias, the annual energy production is evaluated in this section. To get an overview at one glance Table 5.3 illustrates the deviations of the calculated from the measured annual energy output as an average of all wind turbines. Using power curves or density corrected power curves leads in average to an underestimation of about 13 % which is around eight percentage points higher than the average underestimation resulting from Cp-curves. Tables A.2 and A.3 in the Appendix A.2 show the measured and calculated annual energy output and

Table 5.3: *Deviation (average of all turbines) from measured annual energy output in % - negative values imply underestimation*

	Cp		P	P (d.-c.)	
Year	MERRA	open_FRED	-	MERRA	open_FRED
2015	-5.72	-4.68	-12.77	-13.10	-12.08
2016	-5.29	-4.22	-13.56	-13.75	-12.68

deviations from the measured output of all wind turbines simulated with open_FRED data.

5.1.3 Smoothed power curves

This section is subdivided into the paragraphs *Parameters* and *Validation* like Section 4.2.3 where the methodology of validating the `smooth_power_curve()` function was described.

Parameters

The parameters of the `smooth_power_curve()` function are examined by evaluating smoothed power curves of an Enercon E82 representative for all turbine types in this thesis as results are very similar.

The result of smoothing an Enercon E82 power curve with three different block widths is shown in Figure 5.4. The power curves smoothed with 0.1 m/s and 0.5 m/s are concurring whereas the one smoothed with a block width of 1 m/s falls from cut-out wind speed down to zero in steps. The same occurs for all power curves evaluated independently from the standard deviation method and wind speed range. Thus, a block width of 1 m/s is not recommended. Knorr [2016, p. 107] uses a block width of 0.1 m/s but does not give any specific reason for that. As the power curves tested here are concurrent for 0.1 m/s and 0.5 m/s and as the latter would result in a lower computational time a block width of 0.5 m/s will be used for further power curves smoothing in this thesis.

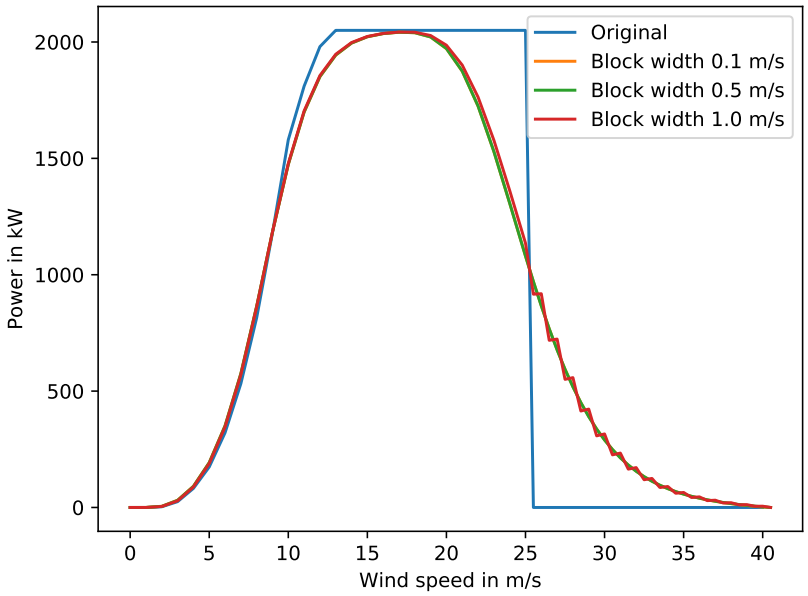


Figure 5.4: *Original and with three different block widths smoothed power curve of an Enercon E82 (wind speed range: 15 m/s, TI method)*

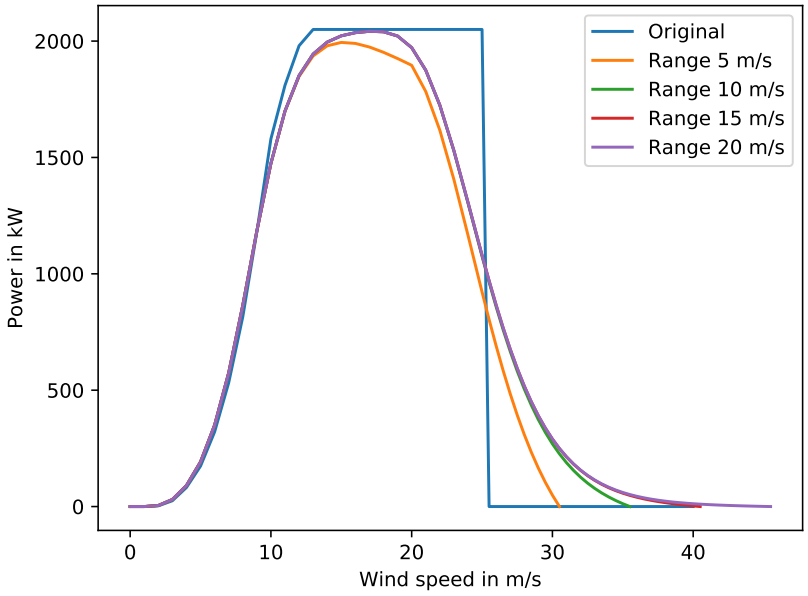


Figure 5.5: *Original and with four different wind speed ranges smoothed power curve of an Enercon E82 (block width: 0.5 m/s, TI approach)*

Figure 5.5 presents the same power curve smoothed with four different wind speed ranges and a block width of 0.5 m/s. The power curve resulting from a smoothing with a wind speed range of 20 m/s covers the curve calculated with a wind speed range of 15 m/s. The power curve smoothed with a wind speed range of 5 m/s does

not reach nominal power and shows a sharp bend at turning to zero. This value was recommended as a minimum in the work of Nørgaard and Holttinen [2000, p. 3] (see Section 2.4.2) but did not lead to convincing results here. The power curve smoothed with 10 m/s shows a less sharp bend when turning to zero than the one smoothed with 5 m/s. As smoothed power curves shown in the work of Knorr [2016, p. 107] and Nørgaard and Holttinen [2000, p. 4] do not bend sharply to zero a wind speed range of 15 m/s is selected for further simulations in this thesis.

In order to compare the two standard deviation methods Figure 5.6 illustrates the E82 power curves smoothed with TI and SP method in comparison with the original power curve. The power curve smoothed with SP method deviates for low wind speeds stronger from the original power curve than the one smoothed with TI method. Moreover, the power curve smoothed with SP method does not reach nominal power whereas the one smoothed with TI method does reach nominal power at wind speeds from around 15 to 18 m/s. For wind speeds greater than cut-out wind speed the power curve smoothed with TI falls faster (with at first higher gradient) to zero than the other curve.

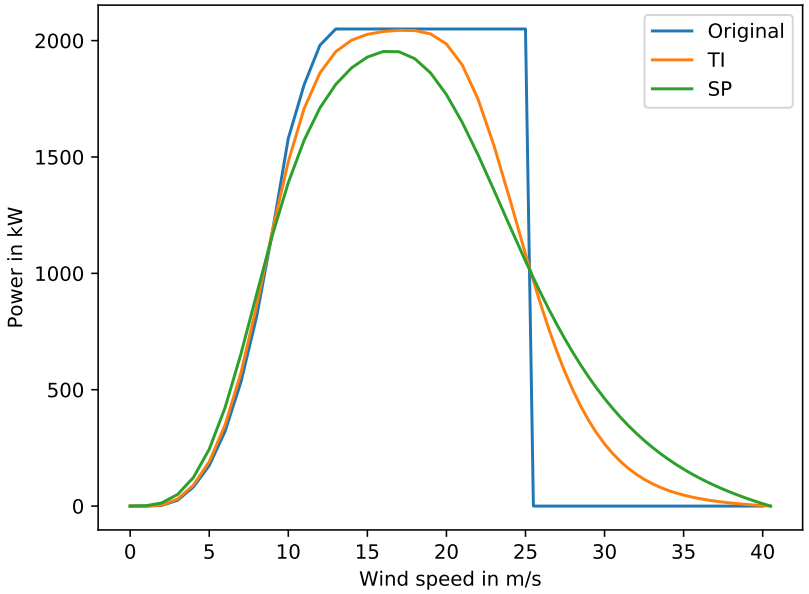


Figure 5.6: *Original and with different standard deviation methods smoothed power curve of an Enercon E82*

Validation

The aim of the simulations with smoothed power curves while neglecting wake losses is to analyze whether the application of smoothed power curves has a positive effect on the simulation results compared to simple aggregation. Moreover, it is examined whether the fluctuation (standard deviation) of measured time series can be represented well by the simulated time series.

Table 5.4 reveals that for the SP method the overestimations of annual energy outputs is increased in comparison to simple aggregation. For the TI method the overestimation is increased to a smaller extent for three wind parks while it is reduced for the wind parks BNE and SH.

Table 5.4: *Annual energy evaluation of simulations with different smoothing approaches (mean of 2015 and 2016)*

	measured	TI		SP		Agg.	
	[MWh]	deviation [%]	[MWh]	deviation [%]	[MWh]	deviation [%]	[MWh]
WF BE	27839.00	19.00	33114.69	22.35	34039.79	18.53	32987.16
WF BNE	39069.95	34.80	52664.87	43.25	55968.13	35.58	52969.36
WF BNW	4315.78	51.15	6479.63	57.86	6764.27	49.32	6402.39
WF BS	35370.39	32.69	46848.29	35.42	47790.30	32.49	46788.44
WF SH	24819.44	11.41	27631.47	12.28	27885.55	12.25	27822.07

The standard deviation of measured feed-in time series and of feed-in time series calculated with different approaches in hourly resolution is illustrated in Figure 5.7 as an average of the years 2015 and 2016. The results of the monthly time series are overall lower but the relationship stays similar. All numbers can be looked up in Table A.4 in the Appendix A.3. From Figure 5.7 it is evident that the standard deviation of the feed-in time series is reduced more by the SP method than by the TI method when being compared with simple aggregation. While in case of WF SH and SP the standard deviation of the measured time series is underestimated it is overestimated in the remaining simulations.

The correlations of simulated with measured time series are examined with the help of Figure 5.8 which presents the Pearson correlation coefficient of the calculated hourly time series in average of the years 2015 and 2016. The correlations of time series calculated with SP smoothed power curves are slightly lower than the correlations of the other time series. Time series with TI smoothed power curves attain higher correlations than with SP smoothed power curves and for WF BE reach the same correlation as aggregated time series. The correlations of monthly time series

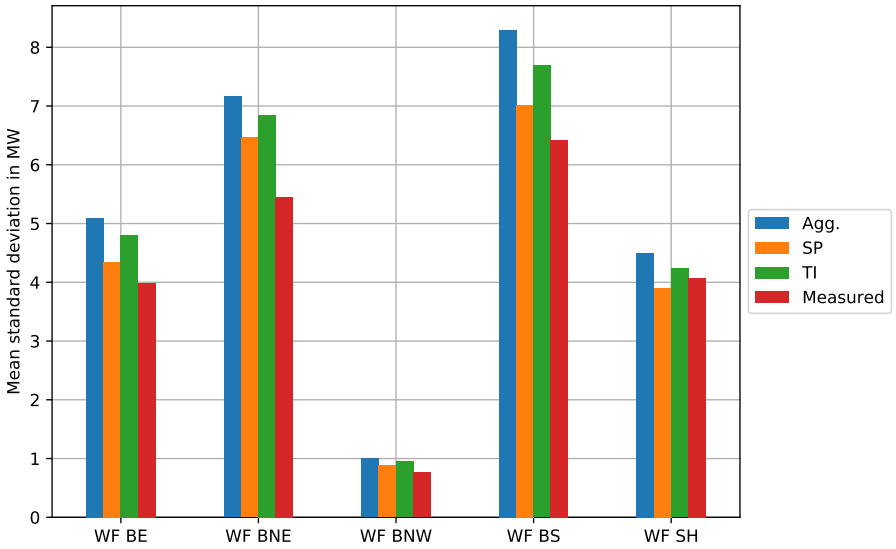


Figure 5.7: *Standard deviation of measured feed-in time series and feed-in time series calculated with smoothed power curves (different methods for standard deviation of Gauss distribution) and calculated with a simple aggregation of wind turbine feed-in in hourly resolution. (average of 2015 and 2016)*

is overall higher and the differences between the methods lower as can be recognized from Figure A.1 in the Appendix A.3.

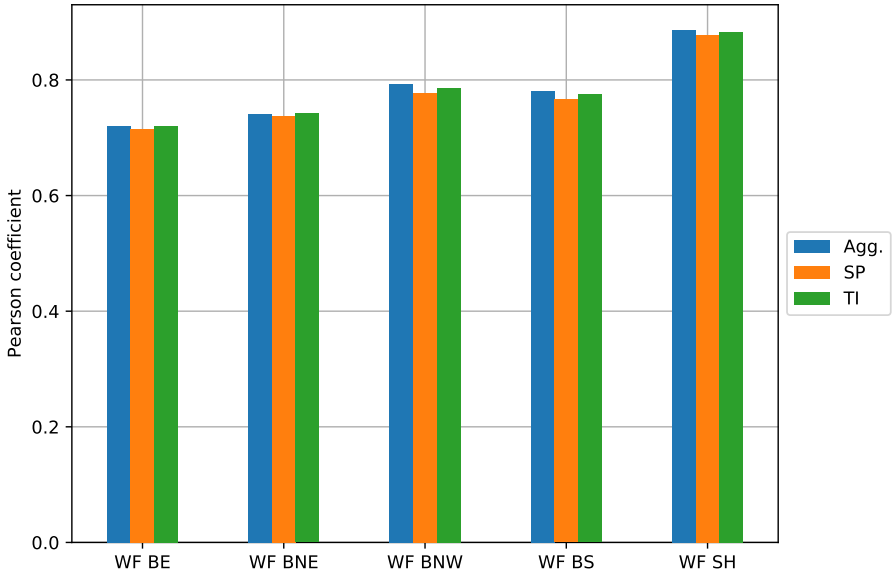


Figure 5.8: *Pearson correlation coefficient of hourly time series calculated with different smoothing approaches and with aggregation approach (average of 2015 and 2016)*

5.1.4 Wake losses

This section presents, firstly, power efficiency curves that were calculated for specific wind farms and, secondly, simulation results of the simulations that were listed in Table 4.5 in Section 4.2.3.

Calculated power efficiency curves

Figure 5.9 shows the power efficiency curves that were generated for WF BE, WF BNW and WF BS as described in Section 4.2.3. Additionally, the mean wind efficiency curve of the "dena-Netzstudie II" and an extremely deviating curve calculated by Knorr [2016] are illustrated.

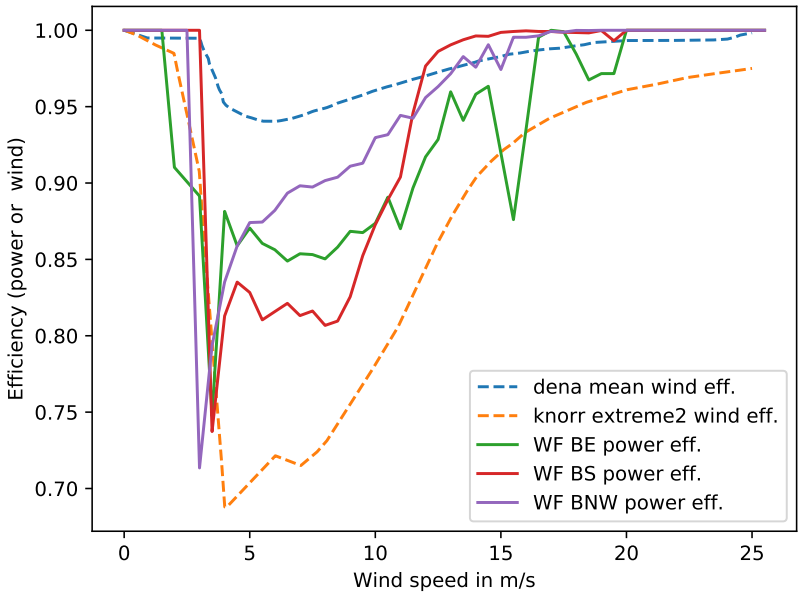


Figure 5.9: *Calculated power efficiency curves, mean wind efficiency curve of "dena-Netzstudie II" [Kohler et al., 2010] and extremely deviating wind efficiency curve of Knorr [2016]*

It should be kept in mind that wind efficiency curves are applied to wind speed time series and, therefore, have a stronger effect on the power output than power efficiency curves that are applied to the power output. Studying the different curves in Figure 5.9 it is noticed that the power efficiency curves are far more fluctuating than the wind efficiency curves. Concerning the effect of the curves it is clear that the extremely deviating wind efficiency curve of Knorr [2016] would lead to a higher reduction of the feed-in than the calculated power efficiency curves. This cannot be declared on first sight for the dena mean wind efficiency curve as its efficiency

values are lower but will have a stronger effect on the power output due to the wind speed being included to the power of three. Therefore, in the next section results of simulations with the different curves are presented.

Simulation results

The annual energy output of the simulations and deviations from the measured annual energy output are shown as an average of both years in Table 5.5. It can be seen that the deviations for all wind farms are highest for the simulations that did not consider any wake losses. This was expected beforehand and is a reason for the implementation of wake losses into the Windpowerlib. Moreover, the deviations of simulations with specifically for wind farms calculated power efficiency curves is for WF BE about one and for WF BS about three percentage points lower than the deviation of simulations with the dena mean wind efficiency curve. In contrast to that, for WF BNW deviations are about five to seven percentage points higher when using a power efficiency curve. Furthermore, the simulations with a constant wind farm efficiency of 80 % result in the lowest deviations, however, the annual energy output of WF BE is underestimated.

Table 5.5: *Annual energy evaluation of simulations with different wake losses methods (average of 2015 and 2016) - negative deviations imply underestimation, positive deviations overestimation*

	measured	Dena		Calc.		Const.		No losses	
	[MWh]	deviation [%]	[MWh]	deviation [%]	[MWh]	deviation [%]	[MWh]	deviation [%]	[MWh]
WF BE	27839.00	4.74	29151.82	3.83	28897.06	-5.18	26389.73	18.53	32987.16
WF BNW	4315.78	32.17	5669.93	37.79	5909.41	19.45	5121.91	49.32	6402.39
WF BS	35370.39	18.24	41781.22	15.87	40943.73	5.99	37430.75	32.49	46788.44

Figure 5.10 displays the statistical metrics of the hourly time series calculated with different wake losses methods averaged over 2015 and 2016. The figure reveals that the highest RMSE per wind farm results from simulations not considering wake losses at all, which was expected beforehand. Further expected was a lower RMSE for simulations with the calculated power efficiency curves in respect of simulations with the dena mean wind efficiency curve which is confirmed in Figure 5.10 for WF BE and WF BS but not for WF BNW. The lowest RMSE is reached by the constant efficiency. The Pearson correlation coefficient in Figure 5.10 is for all methods very similar. For WF BE it is about 0.7 and for the other wind farms about 0.79 which can be classified as upper middle correlations (see Section 4.2.1). The statistical metrics of all temporal resolutions are displayed in Tables A.5 and A.6 in the Appendix A.4. From there it is known that time series in half-hourly resolution have slightly lower

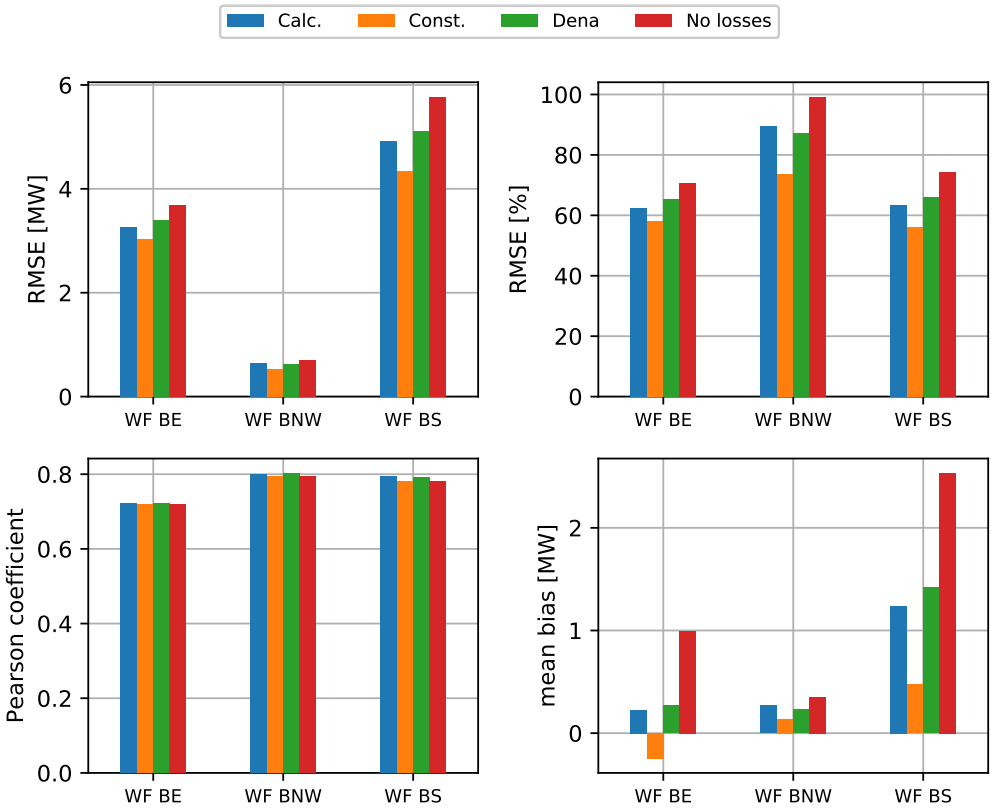


Figure 5.10: *Statistical metrics of hourly feed-in time series calculated with different wake losses methods (average of 2015 and 2016)*

correlations while monthly time series reach strong correlations of 0.9 to 1 depending on the wind farm.

5.2 Single wind turbine simulations

The single wind turbine model is evaluated by comparing the simulated power output of wind turbines with measured first row power output time series. For the evaluation of the annual energy output Tables 5.6 and 5.7 show the measured and calculated annual energy output and the deviations between them as an average of 2015 and 2016 for both weather data sets. The measured power output varies for the different weather data sets as not considered time steps (due to the threshold for the resampling to lower temporal resolutions, see Section 3.3) differ from data set to data set.

Overall high overestimation takes place. The simulations of first row wind turbines of WF BNW and WF BS lead to much higher overestimations than simulations of first row wind turbines of WF BE which might be explained by the denser surround-

Table 5.6: *Annual energy evaluation of wind turbine simulations with different approaches and open_FRED data (average of 2015 and 2016)*

	measured	P		Cp		P (d.-c.)	
	[MWh]	deviation [%]	[MWh]	deviation [%]	[MWh]	deviation [%]	[MWh]
BE	2436.95	11.43	2714.90	19.14	2901.92	12.35	2736.95
BNW	2257.27	50.35	3386.24	72.31	3880.33	52.35	3431.18
BS	2260.17	55.92	3512.23	66.86	3756.10	57.22	3540.60

Table 5.7: *Annual energy evaluation of wind turbine simulations with different approaches and MERRA data (average of 2015 and 2016)*

	measured	P		Cp		P (d.-c.)	
	[MWh]	deviation [%]	[MWh]	deviation [%]	[MWh]	deviation [%]	[MWh]
BE	2637.98	39.81	3687.91	47.90	3900.44	40.58	3707.70
BNW	2282.63	76.34	4025.62	100.36	4572.82	78.34	4070.85
BS	2355.23	113.78	5016.29	125.28	5283.33	114.56	5033.74

ings of the wind farms which were mentioned in Section 3.2. It is evident that for both weather data sets the deviation from the measured annual energy output is lowest for the power curve method, followed closely by the density corrected power curve method and the Cp-curve method. Moreover, in all cases simulations with open_FRED weather data attain lower deviations than simulations with MERRA weather data.

It can be observed from Figure 5.11 that hourly time series simulated with power curves reach RMSE of about 0.35 to 0.45 MW for open_FRED data (about 0.4 to 0.6 MW for MERRA, see Figure A.2 in the Appendix A.5) while RMSE of simulations with density corrected power curves only lie slightly above these values. The Cp-curve simulations lead to the highest RMSE with about 0.41 to 0.47 MW for open_FRED data (about 0.48 to 0.61 MW concerning MERRA). In relative numbers the RMSE for the power curve approach reaches from about 70 to 115 % for open_FRED data (about 80 to 155 % for MERRA data).

All statistical metrics of the simulations can be looked up in Tables A.7 to A.10 in the Appendix A.5. Looking at these tables it can be seen that the correlation for the simulations vary for half-hourly and hourly resolution from 0.7 to 0.8 which means that middle correlations, close to strong correlations, are attained. The monthly time series reach strong correlations with a Pr from 0.93 up to 0.97.

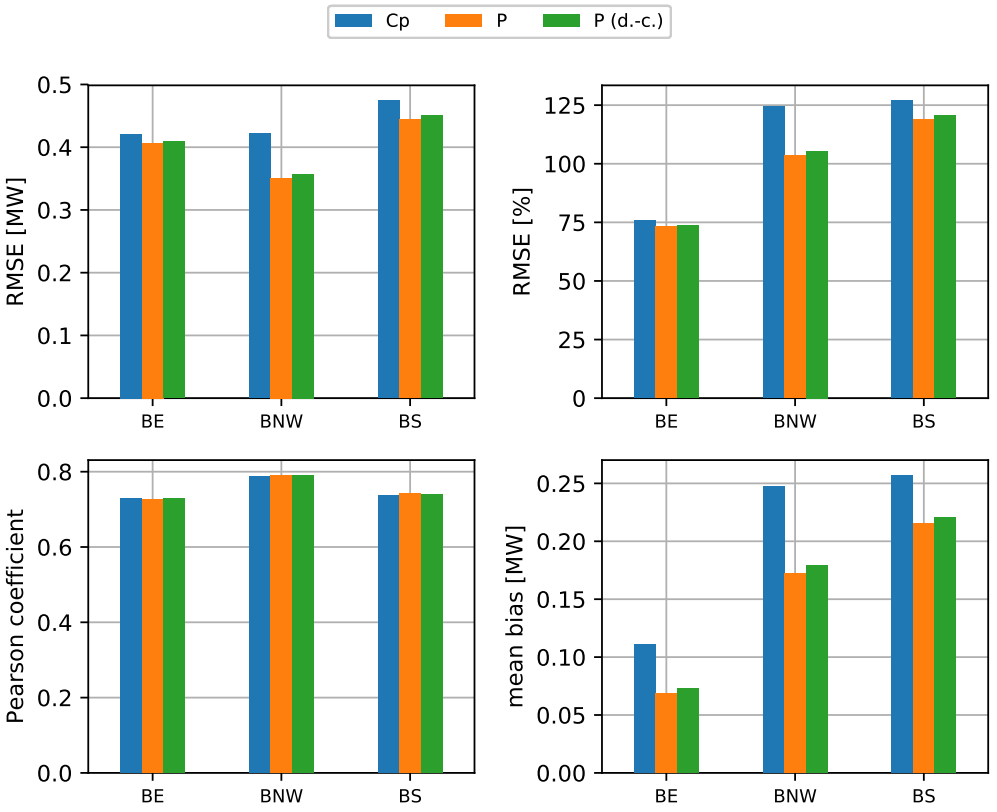


Figure 5.11: *Statistical metrics of hourly feed-in time series of single wind turbines calculated with open_FRED weather data (average of 2015 and 2016)*

5.3 Wind farm simulations

The results of the wind farm feed-in simulations depicted in Table 4.7 are presented in this section. Table 5.8 provides their deviations from the measured annual energy output averaged over the years for both weather data sets. It can be seen that the deviations of simulations with the dena mean wind efficiency curve are in most cases lower if it is applied without power curve smoothing. For WF SH the power curve smoothing improves the results which is also the case for the simulation of WF BS with MERRA data. The application of a constant efficiency of 80 % leads to lower deviations compared with the other methods. However, three simulations with constant efficiency result in underestimations which in two cases leads in absolute numbers to a higher deviation from the annual energy output. Compared to simple power output aggregation (Agg.) all methods improve the results.

The statistical metrics of the simulations with open_FRED data are illustrated in Figure 5.12 where an average over the years 2015 and 2016 is shown. Although the deviation from the annual energy output is lower for simulations with the dena mean wind efficiency curve without smoothing (Dena) hourly time series generated with

Table 5.8: *Deviations from the measured annual energy output in % of wind farm simulations (average of 2015 and 2016)*

	Agg.		Const.		Dena		Dena-TI	
	MERRA	open_FRED	MERRA	open_FRED	MERRA	open_FRED	MERRA	open_FRED
WF BE	46.18	18.53	16.94	-5.18	30.58	4.74	30.94	5.81
WF BNE	83.84	35.58	46.94	5.82	63.64	15.98	65.11	18.80
WF BNW	75.06	49.32	40.05	19.45	55.87	32.17	57.76	34.44
WF BS	75.19	32.49	40.15	5.99	58.61	18.24	58.19	19.14
WF SH	23.43	12.25	-1.25	-10.20	13.41	3.37	13.20	2.91

smoothing (Dena-TI) lead to lower RMSE. The mean biases of Dena-TI simulations are higher than those of Dena simulations which is why Dena simulations attain lower deviations from the annual energy output. The constant efficiency leads to the lowest RMSE but also induces underestimation and reaches lower correlations.

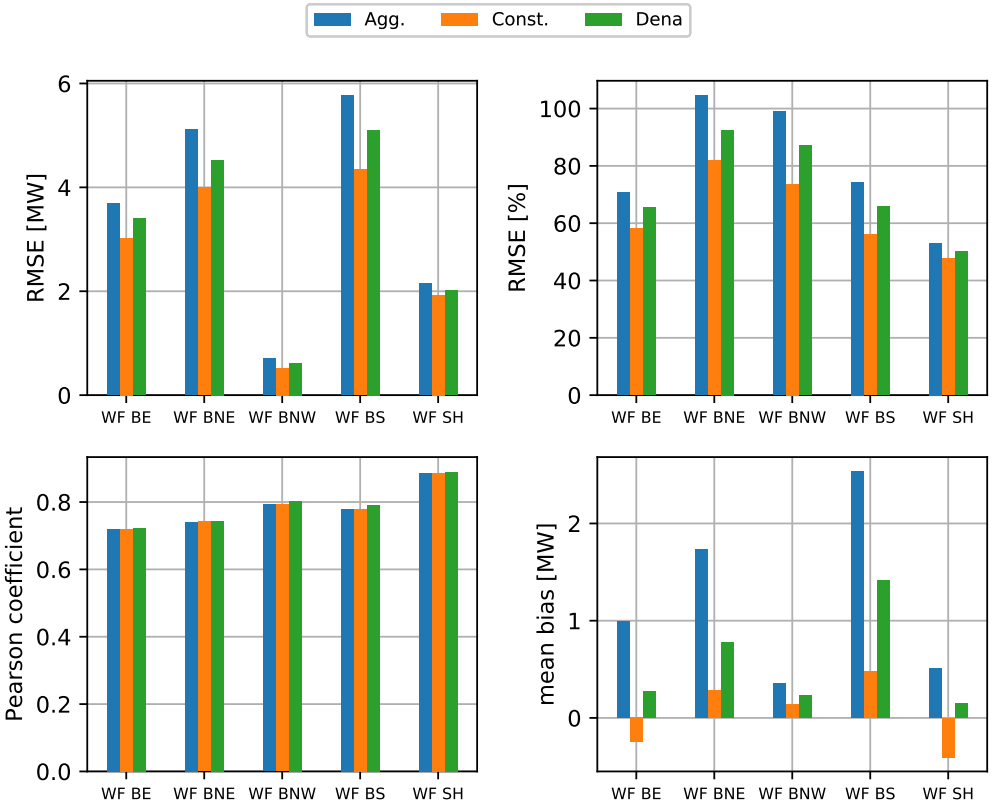


Figure 5.12: *Statistical metrics of hourly feed-in time series of wind farms calculated with open_FRED data (average of 2015 and 2016)*

5.4 Simulations comparing the weather data sets

The results of simulations concerning the influence of different weather data sets are presented in three sections. Section 5.4.1 shows results of simulations revealing the

influence on wind speed height corrections, Section 5.4.2 presents results concerning the influence on single wind turbine power output simulations and Section 5.4.3 displays results concerning the influence on wind farm power output simulations.

5.4.1 Influence on wind speed height corrections

In Section 5.1.1 the results of simulations evaluating wind speed height correction functionalities with different heights of open_FRED weather data were presented. This section also presents simulations concerning wind speed height corrections, however, it is focused on differences in the performance of the different weather data sets. In the simulations of which the results are shown in this section the closest data height of open_FRED wind speeds is used.

Figures 5.13 to 5.15 compare the relative RMSE, the mean bias and the Pearson correlation coefficient of all wind speed height correction simulations for both weather data sets where an average is taken of the two years. For all height correction methods and wind farms the relative RMSE of simulations with open_FRED weather data is lower than with MERRA weather data as is evident from Figure 5.13. However, also with open_FRED data relative RMSE from around 30 to 45 % occur.

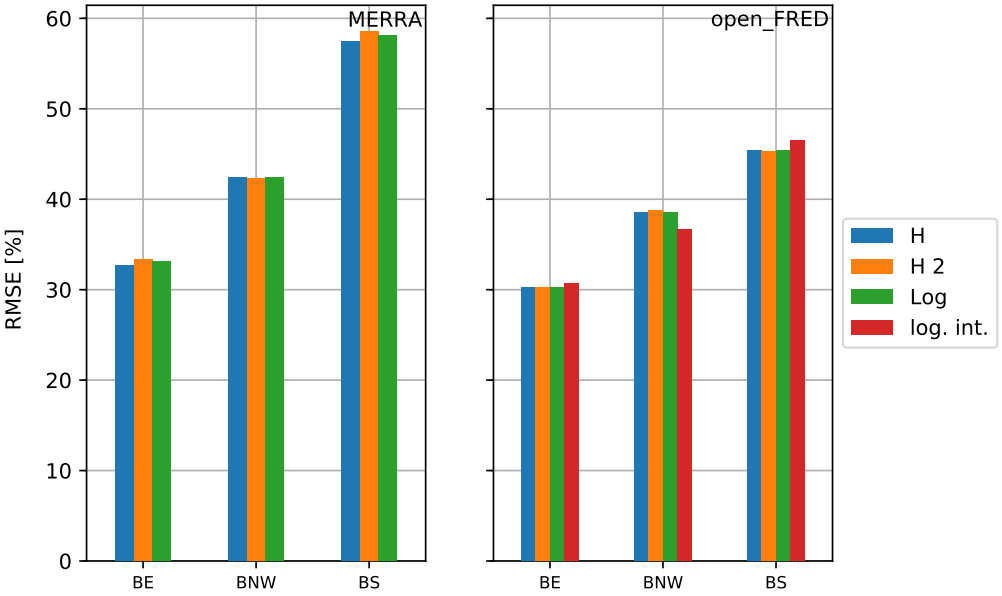


Figure 5.13: *Relative RMSE of hourly time series of wind speed simulations (average of 2015 and 2016)*

These errors result in an overestimation of the annual mean wind speeds of about 0.3 to 1.2 m/s which is indicated by the mean biases in Figure 5.14. These mean

biases of simulations with open_FRED data give hint on a lower mean overestimation comparing to MERRA data with mean biases of about one to more than 2 m/s.

Observing the Pearson correlation coefficients in Figure 5.15 it can be seen that hourly time series simulated with MERRA data attain higher correlations than time series calculated with open_FRED data.

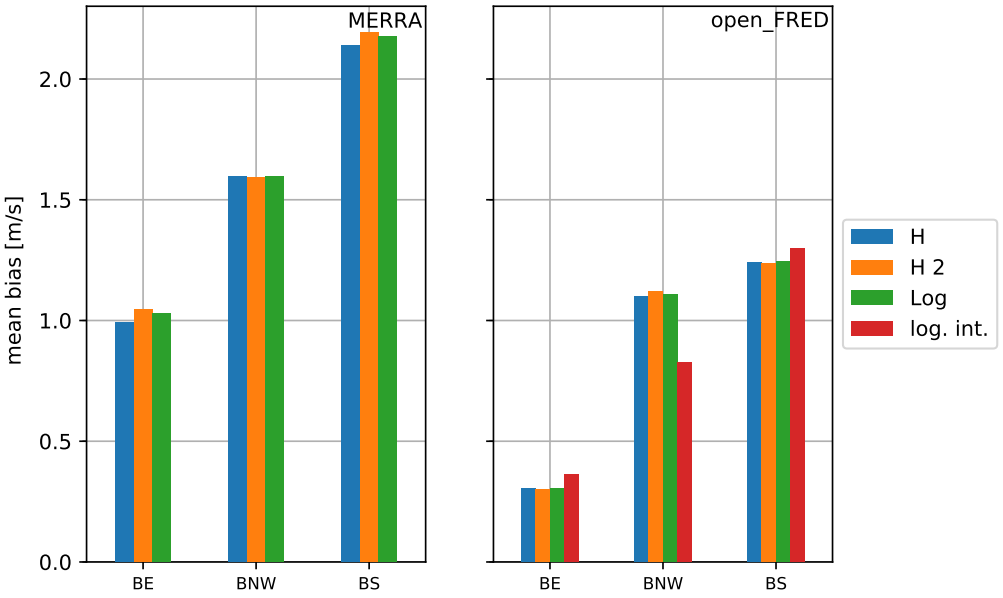


Figure 5.14: *Mean bias of hourly time series of wind speed simulations (average of 2015 and 2016)*

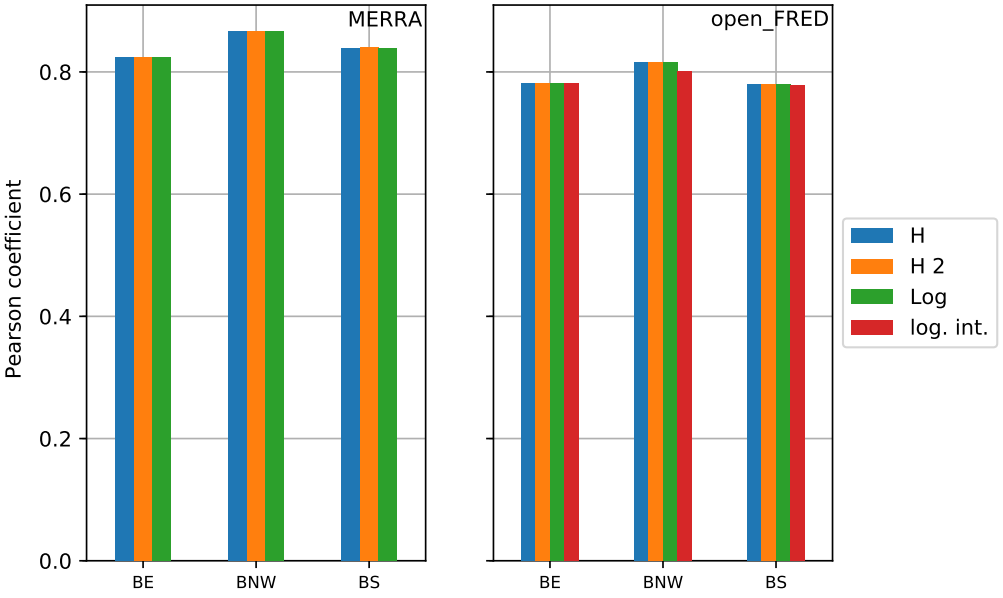


Figure 5.15: *Pearson correlation coefficient of hourly time series of wind speed simulations (average of 2015 and 2016)*

5.4.2 Influence on single wind turbine power output simulations

This section examines the results of the single wind turbine simulations depicted in Section 4.2.4 concerning the different weather data sets. The deviations from the measured annual energy output are overall lower for simulations with open_FRED than for simulations with MERRA data as was already observed in Section 4.6. All deviations averaged over the years can be seen in Table 5.9 where positive values indicate that the annual energy output is overestimated in the simulations. The overestimations by MERRA data are for the first row wind turbines of WF BE and WF BNW about 27 percentage points higher than overestimations by open_FRED and about 58 percentage points respectively for WF BS. The comparatively high

Table 5.9: *Deviation from the measured annual energy output in % of single turbine simulations with different power output methods (mean of 2015 and 2016) - positive values imply overestimation*

	Cp		P		P (d.-c.)	
	MERRA	open_FRED	MERRA	open_FRED	MERRA	open_FRED
BE	47.90	19.14	39.81	11.43	40.58	12.35
BNW	100.36	72.31	76.34	50.35	78.34	52.35
BS	125.28	66.86	113.78	55.92	114.56	57.22

overestimations resulting from simulations with MERRA data can be explained by the higher overestimation of wind speeds compared to open_FRED data (see Figure 5.14).

As the power curve approach performed best in the single wind turbine model for both weather data sets (see Section 5.2) the statistical metrics are compared for this approach only and are illustrated in Tables 5.10 and 5.11.

Table 5.10: *Evaluation of single wind turbine feed-in time series in 2015 calculated with the power curve approach*

		RMSE [MW]		RMSE [%]		Pearson coeff.		mean bias [MW]	
		MERRA	open_FRED	MERRA	open_FRED	MERRA	open_FRED	MERRA	open_FRED
BE	hourly	0.47	0.41	79.34	68.42	0.75	0.74	0.23	0.07
	monthly	0.26	0.09	41.18	14.47	0.98	0.98	0.26	0.08
BNW	hourly	0.42	0.36	114.34	96.66	0.83	0.79	0.28	0.16
	monthly	0.29	0.17	80.37	47.07	0.98	0.96	0.28	0.16
BS	hourly	0.61	0.45	146.66	108.30	0.76	0.76	0.44	0.22
	monthly	0.45	0.26	106.40	57.96	0.95	0.93	0.44	0.23

Table 5.11: *Evaluation of single wind turbine feed-in time series in 2016 calculated with the power curve approach*

		RMSE [MW]		RMSE [%]		Pearson coeff.		mean bias [MW]	
		MERRA	open_FRED	MERRA	open_FRED	MERRA	open_FRED	MERRA	open_FRED
BE	hourly	0.44	0.41	84.87	78.07	0.76	0.71	0.21	0.07
	monthly	0.22	0.14	41.36	24.57	0.95	0.90	0.21	0.08
BNW	hourly	0.37	0.35	116.56	110.28	0.83	0.79	0.24	0.19
	monthly	0.24	0.20	78.25	61.74	0.98	0.95	0.24	0.19
BS	hourly	0.58	0.44	170.23	129.39	0.76	0.72	0.41	0.21
	monthly	0.42	0.25	123.74	67.45	0.99	0.97	0.41	0.22

The RMSE of hourly time series is lower for open_FRED simulations of about five to 26 % and of monthly time series of about 17 to 65 % compared to simulations with MERRA data. The stronger deviation of the monthly time series can be explained by the lower mean bias of open_FRED simulations which stand for better balancing out over time. The correlations of open_FRED simulations are for hourly time series about one to seven percent lower and for monthly time series zero to five percent than the correction of MERRA simulations.

5.4.3 Influence on wind farm power output simulations

The influence of weather data on the simulation of the annual power output of wind farms was already shown in Table 5.8 in Section 5.3. It can be observed that MERRA data leads to far higher overestimations than open_FRED weather data. Only for the constant efficiency the MERRA data performs better at WF SH.

In Figure 5.16 it can be observed that the relative RMSE of open_FRED simulations are lower compared to MERRA simulations apart from WF SH where MERRA attains lower relative RMSE.

Simulations with open_FRED data reach far lower mean biases than simulations with MERRA data except for WF SH and WF BE with a constant efficiency where the absolute mean biases are higher at simulations with open_FRED data (see Figure 5.17). Like in the wind speed and single wind turbine simulations wind farm feed-in time series calculated with open_FRED data lead to lower correlations compared to simulations with MERRA data (see Figure 5.18).

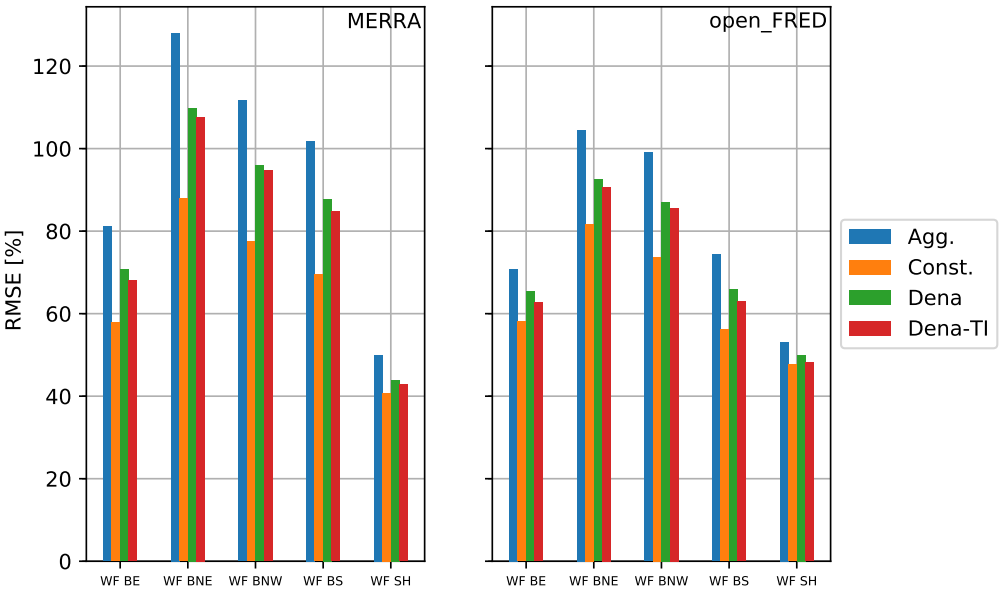


Figure 5.16: *Relative RMSE of hourly feed-in time series of wind farm simulations (average of 2015 and 2016)*

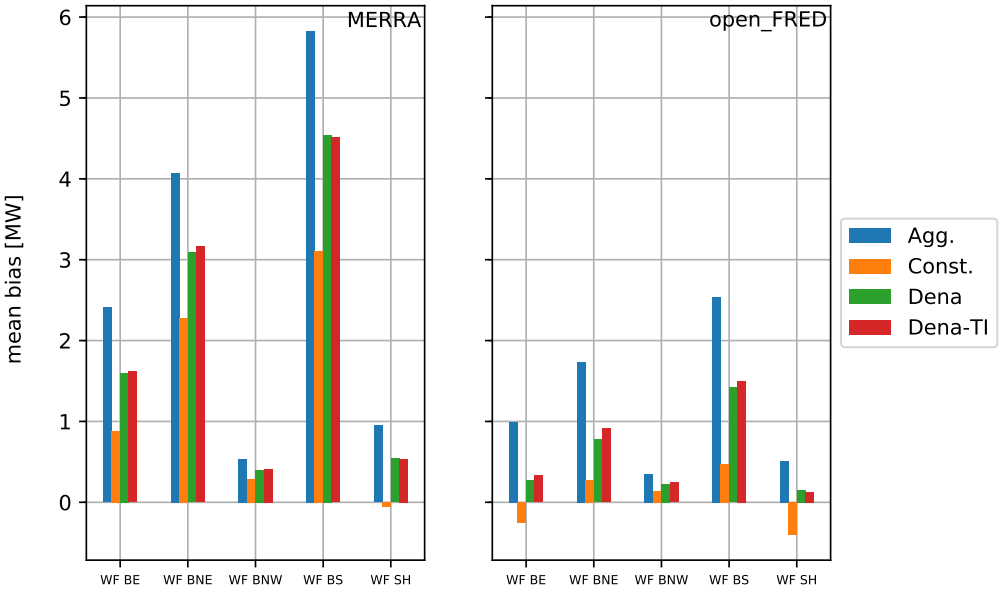


Figure 5.17: *Mean bias of hourly feed-in time series of wind farm simulations (average of 2015 and 2016)*

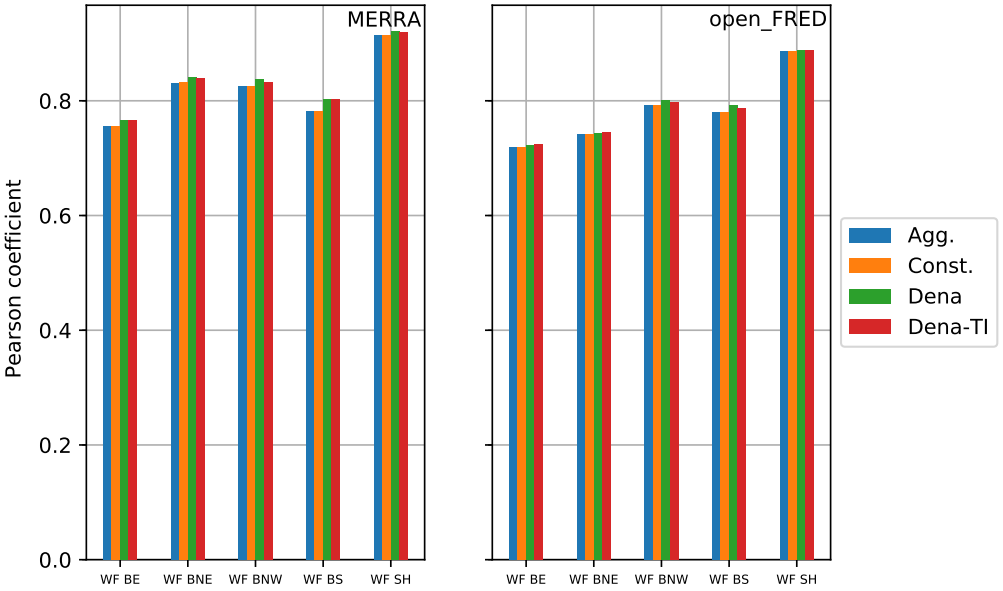


Figure 5.18: *Pearson correlation coefficient of hourly feed-in time series of wind farm simulations (average of 2015 and 2016)*

Chapter 6

Discussion

The objective of this thesis is to implement functions into the open source library Windpowerlib that abolish the limitations of Windpowerlib v0.0.4 named in Section 1.4. Moreover, it is aimed to evaluate feed-in time series simulated with different methods, to validate single functionalities and to examine the influence of weather data on simulation results. The first aim was fulfilled by the implementation of additional functionalities through which a modelling of wind farms and wind turbine clusters was made possible (see Section 4.1.2). The second aim was achieved by validating simulated time series with measured feed-in time series. As single functionalities the height correction methods for wind speed data, power output calculations, power curve smoothing and the application of wake losses were evaluated. In addition to that, the influence of weather data on the simulation results was examined. The performance of the validated functionalities is discussed in Section 6.1 and the influence of the weather data on the results in Section 6.2. Section 6.3 gives hints on the trustworthiness of the results and Section 6.4 deals with the limitations of the Windpowerlib.

6.1 Performance of the functionalities

In this section the performance of the functionalities of the Windpowerlib is discussed. Section 6.1.1 evaluates the performance of wind speed height correction functionalities, Section 6.1.2 examines power output calculations functions, Section 6.1.3 deals with the smoothing of power curves and Section 6.1.4 focuses on functionalities for considering wake losses.

6.1.1 Wind speed height correction functionalities

The functionalities for wind speed height corrections were evaluated by comparing simulated time series using open_FRED wind speeds at different heights with the first row wind speed time series of three wind farms that were obtained from measured data (see Section 3.3.2). A significant difference in performance between the logarithmic wind profile and the Hellman equation could not be detected and was not the aim of this evaluation. However, from Section 2.1.1 it is known that the Hellman equation is likely to underestimate wind speeds more strongly than the logarithmic wind profile. Therefore, it may be preferred to use the logarithmic wind profile, although it may not lead to the best results for all locations and situations. Logarithmic interpolation can be an alternative when the wind speed data is available at two heights. The results in Section 5.1.1 showed a worse performance of the logarithmic interpolation compared to the other functionalities when hub heights are close to the weather data heights (WF BE, WF BS). However, for the wind farm with hub heights further in between the data heights (WF BNW) it resulted in a lower RMSE (and mean bias) than the other functions. This seems logical as extrapolating over a shorter range implies less uncertainty. Moreover, when correcting wind speeds over a longer range the interpolation between two data heights provides certainty.

The performance of wind speed height corrections was rather influenced by the wind speed data height than by the functionality chosen for the calculations. Concerning the different heights of the weather data it was found out in Section 5.1.1 that height corrections from 80 m gave better results than height corrections from 100 m. This was also the case for hub heights of 105 m although the wind speed data height of 100 m is closer to hub height. An explanation for that could be a general overestimation of wind speeds which cancel out with the underestimation induced by the functionalities. Concerning the MERRA weather data Staffell and Pfenninger [2005, p. 31] state that it produces about 60 % overestimated capacity factors in Germany and Denmark. Capacity factors show the relation between the average (annual) feed-in and the installed power. Therefore, they should not only origin from errors in the weather data but for instance also from a technical availability lower than 100 %. However, the overestimation of capacity factors of 60 % should indicate a fairly high overestimation of MERRA wind speeds. As for the generation of the open_FRED weather data MERRA data is used in the COSMO model this high overestimation of wind speeds may also be present in the open_FRED data. This was confirmed by evaluations within the open_FRED project. The even lower

RMSE from simulations with a data height of 10 m may result from wake losses induced from wind turbines standing close to BNW and BS. This was shown in Figure 5.3 where only wind speeds not being influenced by these surroundings were considered (selected by wind direction). Thus, it is not recommended to use a data height as far away as 10 m when closer data heights are available. However, using a slightly lower height can lead to better results. In this thesis this was only tested on three wind farms which means that this finding might not be generally valid. Nevertheless, as mentioned earlier, wind speeds are usually overestimated by the weather models which leads to the expectation of similar results for other wind farms.

6.1.2 Power output calculation functionalities

The power output calculation functionalities of the Windpowerlib were validated with measured feed-in time series of 24 wind turbines in three wind farms. The power output of these wind turbines was calculated from measured wind speed time series instead of reanalysis wind speeds in order to exclude errors from the weather data or made by height corrections. The power output of the wind turbines was in average underestimated by about five percent when using Cp-curves and about 13 % when using power curves or density corrected power curves as is known from Section 5.1.2. Such a great difference of performance between Cp-curves and density corrected power curves was not expected. As both methods take the site specific air density into consideration it was contemplated as likely that they would perform similarly. This questions the density corrected power curve approach and should be further examined.

Although the results showed that Cp-curves perform better than power curves when examined as single functionalities they performed differently in the validation of the single wind turbine model (see Section 5.2). In that section the performance of calculating the power output of a single wind turbine with reanalysis wind speeds was evaluated. It was explained earlier that reanalysis wind speeds from the weather data sets are generally overestimated. From Figure 5.1 a relative RMSE of about 30 % (WF BE) of the wind speed time series can be observed. As wind speed has an effect to the power of three on the power output it is not surprising that in average the power output was overestimated in the single wind turbine simulations. As power curves led to higher underestimations than Cp-curves when applied as single functionality they performed better when they were combined with the highly

overestimated wind speeds (see Section 5.2). As wind speeds seem to be generally overestimated this should be the case for other locations, too, which is why power curves should lead to better results in wind feed-in simulations than Cp-curves.

6.1.3 Smoothing power curve functionality

For the smoothing of power curves appropriate parameters could be identified in Section 5.1.3 which were set as default values within the Windpowerlib. The simulations for the evaluation of this function were carried out with these default parameters. The objective of the simulations was to analyze the effect of smoothed power curves on the simulation results and to examine whether the fluctuation of measured time series could be represented well. The results of the simulations in Section 5.1.3 reveal a bad influence on the annual mean production. In most cases power curve smoothing led to higher overestimations of annual energy output than a simple aggregation of power outputs. The influence of the SP method was in all cases worse than the influence of the TI method. Thus, it was not used in any other simulations in this thesis. As the roughness length and wind turbine hub heights are used for calculating a turbulence intensity the TI method depends on site specific parameters. This could be detected in the results as the performance of TI depended on the wind farm. However, only in two cases (WF BNE and WF SH) it led to better results than the simple aggregation. Another explanation for a dependency on the location is a dependency on wind speeds. Looking at the smoothed power curves in Figure 5.6 it is obvious that the frequency of wind speeds below and above certain values determine whether compared to simple aggregation a higher or lower mean power output is calculated. A higher overestimation of the yearly mean with smoothed power curves means that wind speeds below about 8 m/s (in case of Figure 5.6) and above cut-out wind speed rather influenced the power output calculations. The bad effect of smoothed power curves on the annual mean detected in the results might be prevented by adjusting μ in the Gauss function, which will be proposed in Chapter 8. As was found out in Section 5.1.3 the fluctuation of simulated wind farm feed-in time series can indeed be lowered by the application of smoothed power curves. However, the overall appearance of the wind farm feed-in time series may usually be more important. Thus, the application of power curves should only be used if an offset adjustment can be done and validated.

6.1.4 Functionalities for the consideration of wake losses

For the performance of functionalities considering wake losses three approaches were compared: (dena) mean wind efficiency curve, power efficiency curves and a constant wind farm efficiency of 80 %. It should be kept in mind that wind efficiency curves are applied to wind speed time series while power efficiency curves (and constant efficiencies) are applied to power curves. Moreover, the power efficiency curves used in this thesis were calculated specifically for the wind farms on which they were applied later.

In Section 5.1.4 it was noticed that these calculated power efficiency curves are far more fluctuating than the wind efficiency curves. This can be explained by the way the curves were generated. While Knorr [2016, p. 114] and Kohler et al. [2010, p. 100 f.] used wake models in which for frequent wind speed steps efficiencies could be calculated, in this work the power efficiency curves were calculated by evaluating measured feed-in time series that do not provide values for all possible wind speeds and that could contain errors. Furthermore, the mean values of the curves comprise varying amounts of time steps which leads to a different weighing of incorrect values.

As mentioned before wind efficiency curves and power efficiency curves (and constant wind farm efficiencies) are applied in different ways. Thus, they cannot be compared directly with each other. Still, it was aimed to evaluate the effects of these different curves on the simulation results. In addition to the simulations with these curves a constant efficiency was used in one simulation to evaluate whether such a simplified consideration of wake losses would lead to acceptable results. As known from Sections 5.1.4 and 5.3 a constant efficiency of 80 % led in most cases to lower deviations from the measured annual energy output and lower RMSE than the other methods. From first sight these results are surprising. It was expected that a constant efficiency would lead to higher errors as for high wind speeds the constant efficiency would highly underestimate the power output. A possible explanation for these low errors are the high overestimations of power output (see Section 5.2). For high wind speeds the overestimation of the power output could cancel out the underestimation induced by the constant wind farm efficiency. In some cases a constant efficiency led to underestimations and in absolute values to higher deviations from the measured annual energy output. This leads to the assumption that the performance of the constant efficiency is comparatively good for wind farms for which the dena mean wind efficiency curve is overestimating the efficiency. When calculating larger regions it is probable that the different efficiencies of the wind farms are in average met well by the dena mean wind efficiency curve. Furthermore, from Section 5.3

it is known that the application of a constant efficiency influences the correlation of the time series negatively and can even lead to lower correlations than a simple aggregation of wind turbine power outputs. Thus, the dena mean wind efficiency curve should be preferred over a constant efficiency.

Comparing the results of the dena mean wind efficiency curve with the results of the calculated power efficiency curves in Section 5.1.4 it becomes clear that there is only a small difference between their performances. As the power efficiency curves were specifically calculated from measured feed-in data it was expected to find a greater difference with a better performance of the power efficiency curves. Furthermore, the results for the different wind farms are ambiguous. For two of the wind farms (WF BE and WF BS) the calculated power efficiency curves performed better while for the third wind farm (WF BNW) the dena mean wind efficiency curve attained better results. This difference in performance can be seen looking at the behavior of the power and wind efficiency curves in Figure 5.9. Compared to the other curves the power efficiency curve of WF BNW rises with a higher gradient and gets closer to the dena mean wind efficiency curve for wind speeds of about four to 11 m/s. As the wind speeds influence the power output to the power of three the power efficiency curve of WF BNW can lead to higher overestimation. The reason for the different appearance of this power efficiency curve can be found in the site specific circumstances of this wind farm. As known from Table 3.2 it only comprises two wind turbines and is by definition of Section 2.3.1 part of a wind farm comprising more wind turbines. It can be assumed that the influence of the surrounding wind turbines is stronger than the wake losses being induced from one on the other wind turbine of WF BNW.

6.2 Influence of the weather data

As known from the simulation results open_FRED weather data performed better than MERRA weather data throughout this thesis concerning the RMSE of time series and the annual energy output. Moreover, a high influence of the weather data on the results could be detected. Looking at the statistical metrics of wind farm simulations in Section 5.4.3 a difference between the weather data sets of up to about 25 % concerning the RMSE of hourly time series depending on the calculation method and wind farm can be seen. The Pearson correlation coefficient seems to be less influenced while the mean biases show great differences between the weather data sets. According to the mean biases errors of time series calculated

with open_FRED weather data cancel out way better than errors made in calculations with MERRA data. This was confirmed in the deviations from the annual energy output in Table 5.8 and means that the errors of hourly wind turbine time series calculated with open_FRED data are fluctuating more between positive and negative values than errors made by MERRA data. An explanation for that could be the higher overestimation of wind speeds with MERRA data.

Interesting are the results of wind farm simulations for WF SH where MERRA data led to lower RMSE but to higher mean biases and a higher overestimation in the annual mean compared to open_FRED data. A reason for the comparatively good performance concerning the RMSE could be that WF SH is located close to the coast in Schlewsig-Holstein where the land mass does not influence the wind as strongly as in inland regions in Brandenburg. It is possible that close to the coast the advantage of the open_FRED data having a finer spatial resolution is lower.

As discussed in Section 6.1.1 the data heights at which wind speed data is available have a stronger influence on the simulation results than the different functionalities for height corrections.

6.3 Trustworthiness of the results

Concerning the trustworthiness of the results it has to be looked at the wind farms and their measurement data used in the validation. As described in Section 3.2 some of the wind farms are surrounded by other wind turbines while others are not. In simulations of larger areas wind farms would not be defined by ownership but by the distance between the wind turbines. Therefore, to evaluate the performance of the Windpowerlib and its functionalities it should rather be looked at those wind farms being free from surrounding obstacles. For the coastal region WF SH is a good example as it is not shaded by any other wind turbines. It was mentioned in Section 3.2 that the wind farm BNW is surrounded by other wind turbines and that also WF BE and WF BS are shaded from some directions. Of these farms WF BE is the one shaded least and, therefore, the most reliable. The wind farm BNE in Brandenburg is not surrounded by other wind turbines. However, it comprises wind turbines of different hub heights while wind speeds at mean hub height are applied to the aggregated power curve. This might lead to errors as the wind speed is different at the wind turbine hub heights. In the "dena-Netzstudie II" [Kohler et al., 2010] this is neglected while Knorr [2016] suggests to correct the power curve wind speeds before the aggregation (see Section 2.5.1).

6.4 Limitations of the Windpowerlib

As the Windpowerlib is intended for generating feed-in time series of larger areas rather than single wind turbines or wind farms simplifying assumptions are made in the calculation methods. This means that the feed-in of single wind turbines or wind farms is not simulated in great detail. This is not necessary for larger areas as the uncertainty about wind turbine types and their locations is too high to go into such great detail in the calculations.

The performance of the functionalities of the Windpowerlib were validated in this thesis, however, further validation is needed for the simulation of a higher aggregation level than wind farms. Moreover, it would be useful to validate the functionalities for further regions. The locations of the wind farms used for the validation in this thesis differ from coastal region to flat inland regions, however, it would be interesting to examine locations with more distinctive topography as this might have a great influence on the results.

Chapter 7

Conclusion

In this thesis a basic version of the open source model called *Windpowerlib* was used as a basis for the implementation of a profound library for generating wind feed-in time series. The objectives of this thesis were to develop this library in a way to be able to calculate feed-in time series simulations of wind farms and larger areas and to validate the various functionalities. It was further aimed to examine the influence of different weather data sets on the simulation results.

The simulation of wind farms and larger areas was made possible by the implementation of a functionality for considering wake losses. To apply this functionality it was further necessary to implement power output calculations by a power curve which was previously only possible by power coefficient curves. Moreover, additional functions for wind speed height corrections (Hellman equation) and density calculations (ideal gas equation) and the possibility of a density correction of power curves were implemented. Furthermore, a functionality dealing with the spatial distribution of wind speeds was added (smoothing of power curves). Except for the functionalities for density and temperature calculations all functions were evaluated with measured feed-in and wind speed time series. Concerning the wind speed and power output functionalities validation time series of three different locations were used while for the validation of the other functions time series of five different locations were available.

It was found that the height of the utilized wind speed data strongly influences the results while the different height correction functionalities performed similarly in this thesis. It can be inferred that choosing a lower weather data height the general overestimation of wind speeds cancels out underestimations induced by height correction functionalities. Moreover, using logarithmic interpolation between two data heights led to the best results at one wind farm. For all available power output

functionalities underestimation of the annual energy output was detected when measured wind speed data was used. The underestimation was of about eight percent higher when using power curves or density corrected power curves compared to using power coefficient curves. In combination, the overestimated wind speeds and the underestimation by the power output calculation functionalities led to overestimations in wind turbine power output simulations. In this case the performance was best when using power curves for the power output calculations. For wind farms assessed as reliable data source in Section 6.3 the annual energy output of wind farms was overestimated by 4.7 % (in Brandenburg) and 3.4 % (in Schleswig-Holstein) when applying the dena mean wind efficiency curve (methods: power curve, logarithmic wind profile, data: open_FRED). The overestimations in the simulations of single wind turbines were slightly higher with an average of 11.4 % for a wind turbine standing in the first row of the wind farm in Brandenburg (methods: power curve, logarithmic wind profile; data: open_FRED). For both, the simulated wind turbine feed-in time series and the simulated wind farm feed-in, high Pearson correlation coefficients of about 0.7 to 0.9 were attained. In comparison with the simple aggregation of the power outputs of wind turbines an improvement could be achieved. This was observed in the RMSE and the correlation of the time series as well as in the lower overestimation of annual energy outputs. The latter was reduced by applying the dena wind efficiency curve (same methods as above) from 18.5 % to 4.7 % (Brandenburg) and from 12.3 % to 3.4 % (Schleswig-Holstein).

Concerning the different weather data sets there was found a high influence on the simulation results. For example the overestimations of annual energy output of wind farms (same methods as above) was about 26 (Brandenburg) and about ten (Schleswig-Holstein) percentage points lower with open_FRED data compared to MERRA data. The differences of the relative RMSE and correlations of the time series were smaller. This leads to the conclusion that the weather data has an effect on the canceling out of errors over time. It was further found out that the difference in performance was lower for the wind farm SH that is located close to the coast which leads to the supposition that the finer resolution of the open_FRED weather data develops its advantage primarily in inland regions.

Chapter 8

Outlook

The Windpowerlib will continue as open source community project organized at the Reiner Lemoine Institute and is likely to gain more contributing developers in the future. It will provide wind feed-in time series for energy systems simulations with oemof, which is the Open Energy Modelling Framework used at the Reiner Lemoine Institute. Besides, it can be used as a standalone application.

In this thesis different aspects concerning the simulation of wind feed-in time series were implemented into the library and evaluated. Nevertheless, there are still questions remaining that are interesting to be answered. Concerning the validation it is recommended to examine the model performance on higher aggregation levels, for instance of federal states in Germany. In connection with that it would be highly interesting to examine and validate an additional approach determining the standard deviation for the generation of smoothed power curves mentioned in Section 2.4.2. This alternative method introduced by Nørgaard and Holttinen [2000] takes into account the dimension of the area over which wind turbines are spread. This might be very useful for the simulation of wind feed-in of different sized areas. However, as they state that a further validation of this approach is still needed it should be validated before implementing it into the Windpowerlib. Moreover, functionalities for density calculations and temperature height corrections could be validated in the future. Another additional interesting research could be evaluating the performance of the functionalities depending on time periods such as seasons or times of day. Especially for wind speeds it may be advantageous, if existent, to gain knowledge about their overestimation or underestimation in specific seasons or times of day.

Concerning the development of new functionalities for the Windpowerlib temporal variability could be an aspect to consider. The temporal resolution of time series generated by the Windpowerlib is identical with the temporal resolution of the in-

put weather data. However, for some applications it might be useful to gain finer temporal resolutions than the hourly (or half-hourly) resolution that is common for weather data. Another useful functionality could be a height correction of power curve wind speeds when aggregated power curves of wind farms or clusters comprising wind turbines with varying hub heights are simulated (see Section 6.3 and 2.5.1). Moreover, it was mentioned in Section 6.1.3 that adjusting the offset (mean) of the Gauss distribution in the power curve smoothing function may lead to better results. Optimizing the offset for each wind turbine cluster power curve could be very time intensive when calculating larger areas. However, if values ofr the offset representing an area of several weather data points well could be found this could be a great achievement.

Concerning the power efficiency curves calculated in this work it would be excellent to find a dependency on wind farm characteristics. However, it should be kept in mind that for wind efficiency curves, which represent the reduced wind speed induced by wake losses in a wind farm, a dependency on wind farm characteristics could not be identified in the literature (see Section 2.3.3). Therefore, it might be sufficient to find "classes" of wind farms defined by few characteristics that could at least estimate the extend of wake losses.

For community modelling projects it is common to deliver tests with the code that have to successfully run through before changes in the code are accepted to the project. A development of these tests has already been started for some of the functionalities. For the successful continuation of the Windpowerlib as open source community project it is necessary to include tests for all functionalities and modules contained in the library.

Appendix A

Appendix

A.1 Evaluation of wind directions and nacelle positions

This section presents results of an evaluation of wind directions and nacelle positions of the wind farms BNW and BS (see Section 3.2). For WF BE measured wind directions are not available. Instead nacelle positions which can have an azimuth error are used for simulations with these three wind farms after investigating the correlation and deviation between wind directions and nacelle positions of the wind turbine of WF BNW and WF BS. Table A.1 shows the Pearson correlation coefficient (Pr) of wind directions with nacelle positions and the mean biases between these. For an explanation of the statistical metrics Pr and mean bias see Section 4.2.1. From Table A.1 it is evident that all correlations between wind direction and nacelle position are higher than 0.8 which is classified as strong correlation (see Section 4.2.1). Moreover, apart from BS 11 for all wind turbines the mean bias is lower than 3.5°. As WF BE is from the same operator it can be assumed that also for this wind farm nacelle positions and measured wind directions show similar correlations and mean biases. BS 11 was not considered in any calculation where wind directions are used. Among the measured nacelle positions of wind turbine BS 14 negative values were found. It was assumed that the measurement device wrote down negative values when the wind direction jumped from 0° to wind directions lower than 360°. Therefore, these values were adjusted by adding 360° which resulted into strong correlations and low mean biases.

Table A.1: *Pearson correlation coefficient (Pr) of wind directions with nacelle positions and mean bias between these for wind turbines of WF BNW and WF BS*

Wind turbine	**2015**		**2016**	
	Pr	**Mean bias** [°]	**Pr**	**Mean bias** [°]
BNW 1	0.90	1.71	0.88	-0.63
BNW 2	0.87	-0.19	0.90	0.57
BS 1	0.90	1.85	0.84	0.59
BS 2	0.90	2.17	0.83	2.63
BS 3	0.85	1.57	0.88	0.84
BS 4	0.92	1.04	0.87	-0.42
BS 5	0.89	1.88	0.89	0.29
BS 6	0.89	0.73	0.86	-1.23
BS 7	0.90	2.07	0.89	-0.64
BS 8	0.88	1.58	0.89	-0.30
BS 9	0.86	0.56	0.88	-0.72
BS 10	0.87	3.39	0.89	-1.69
BS 11	0.90	1.02	0.37	-45.39
BS 12	0.91	2.03	0.89	1.39
BS 13	0.91	0.80	0.89	-1.16
BS 14	0.89	0.03	0.87	-1.95

A.2 Additional results of power output calculations

Table A.2: *Annual energy output evaluation of power output calculations with measured wind speed data in 2015 (density from open_FRED) - negative deviations imply underestimation, positive deviations overestimation*

	measured	P		Cp		P (d.-c.)	
	[MWh]	deviation [%]	[MWh]	deviation [%]	[MWh]	deviation [%]	[MWh]
BE 1	3331.63	2.58	3417.45	10.61	3685.03	3.21	3438.60
BE 2	3500.55	-2.77	3403.60	5.09	3678.88	-2.04	3429.20
BE 3	3303.77	-8.99	3006.68	-1.27	3261.83	-8.29	3029.86
BE 4	3942.46	-16.60	3288.01	-9.66	3561.68	-15.95	3313.48
BE 5	3510.42	-6.25	3291.18	1.15	3550.79	-5.49	3317.85
BE 6	3303.03	-9.35	2994.22	-1.28	3260.70	-8.66	3017.10
BE 7	3344.44	0.75	3369.57	8.81	3639.07	1.50	3394.46
BE 8	3515.53	-22.47	2725.50	-15.54	2969.16	-21.89	2745.82
BE 9	3490.20	-9.95	3142.96	-2.65	3397.68	-9.28	3166.36
BNW 1	2363.42	-9.14	2147.40	6.81	2524.36	-7.81	2178.79
BNW 2	2404.17	-12.18	2111.43	3.47	2487.67	-10.82	2143.94
BS 1	3194.12	-16.38	2670.79	-8.58	2919.93	-15.73	2691.76
BS 10	3215.55	-9.18	2920.29	-1.18	3177.49	-8.48	2942.93
BS 11	3959.12	-8.95	3604.96	-1.66	3893.44	-8.29	3630.99
BS 12	3818.39	-18.45	3113.75	-11.56	3377.01	-17.83	3137.51
BS 13	4167.93	-3.47	4023.31	3.28	4304.59	-2.90	4047.19
BS 14	3545.42	-22.60	2744.32	-15.36	3001.01	-22.00	2765.45
BS 2	2976.16	-19.64	2391.58	-11.54	2632.67	-19.07	2408.55
BS 3	3389.00	-16.40	2833.15	-8.80	3090.63	-15.81	2853.09
BS 4	3519.44	-17.53	2902.51	-10.01	3167.16	-16.95	2922.94
BS 5	3340.41	-14.56	2853.95	-6.48	3123.82	-13.96	2874.15
BS 6	4362.59	-17.07	3618.06	-10.51	3904.20	-16.46	3644.55
BS 7	3747.79	-17.80	3080.60	-10.72	3345.89	-17.24	3101.69
BS 8	3442.58	-22.53	2666.95	-15.38	2913.12	-21.96	2686.63
BS 9	4612.55	-20.24	3679.16	-14.00	3967.02	-19.69	3704.21

Table A.3: *Annual energy output evaluation of power output calculations with measured wind speed data in 2016 (density from open_FRED) - negative deviations imply underestimation, positive deviations overestimation*

	measured	P		Cp		P (d.-c.)	
	[MWh]	deviation [%]	[MWh]	deviation [%]	[MWh]	deviation [%]	[MWh]
BE 1	3165.61	4.38	3304.30	13.42	3590.43	5.17	3329.14
BE 2	3297.41	-2.38	3218.94	6.53	3512.64	-1.33	3253.39
BE 3	3138.79	-6.88	2922.79	1.91	3198.74	-5.98	2950.96
BE 4	3615.93	-17.14	2996.16	-9.30	3279.60	-16.34	3024.94
BE 5	3454.18	-7.18	3206.04	1.18	3495.07	-6.35	3234.71
BE 6	3145.82	-9.58	2844.55	-0.67	3124.89	-8.68	2872.68
BE 7	3114.91	0.60	3133.45	9.79	3419.78	1.60	3164.88
BE 8	3260.03	-22.98	2510.94	-15.16	2765.92	-22.19	2536.67
BE 9	3370.69	-8.96	3068.62	-0.70	3347.23	-8.18	3094.98
BNW 1	2077.22	-13.03	1806.46	3.82	2156.56	-11.76	1833.03
BNW 2	2052.44	-8.29	1882.29	8.96	2236.42	-6.98	1909.25
BS 1	2379.17	-19.11	1924.61	-9.91	2143.39	-18.34	1942.90
BS 10	2447.14	-10.71	2185.01	-1.25	2416.59	-9.86	2205.96
BS 11	3019.73	-9.20	2741.95	-0.40	3007.76	-8.21	2771.69
BS 12	2802.59	-20.73	2221.63	-12.45	2453.58	-19.89	2245.17
BS 13	3163.12	-1.58	3113.16	6.66	3373.65	-0.72	3140.40
BS 14	2607.01	-25.21	1949.89	-16.45	2178.03	-24.37	1971.78
BS 2	2361.44	-21.65	1850.22	-12.34	2069.97	-20.92	1867.32
BS 3	2571.81	-18.05	2107.55	-8.85	2344.18	-17.18	2130.05
BS 4	2750.11	-18.77	2233.92	-9.72	2482.86	-17.91	2257.44
BS 5	2578.71	-15.55	2177.67	-6.04	2422.87	-14.77	2197.91
BS 6	3417.58	-18.92	2770.97	-11.14	3036.91	-18.13	2798.02
BS 7	2894.88	-20.14	2311.88	-11.58	2559.61	-19.27	2337.09
BS 8	2624.72	-25.37	1958.79	-16.85	2182.56	-24.66	1977.57
BS 9	3625.80	-22.45	2811.82	-14.97	3082.99	-21.68	2839.62

A.3 Additional results of power curve smoothing

Table A.4: *Standard deviation of feed-in time series calculated with smoothed power curves (method for standard deviation of gauss distribution: TI - turbulence intensity, SP - Staffell_Pfenniger) and calculated with a simple aggregation (Agg.) of turbine feed-in and of the measured time series. (average of results for 2015 and 2016)*

Weather data name	Temporal resolution	Standard deviation in MW			
		Agg.	SP	TI	Measured
WF BE	half-hourly	5.11	4.34	4.82	4.02
	hourly	5.09	4.35	4.81	3.99
	monthly	1.56	1.32	1.47	1.16
WF BNE	half-hourly	7.21	6.50	6.89	5.50
	hourly	7.16	6.46	6.84	5.45
	monthly	2.35	2.07	2.21	1.64
WF BNW	half-hourly	1.01	0.89	0.96	0.78
	hourly	1.00	0.88	0.95	0.76
	monthly	0.35	0.31	0.33	0.28
WF BS	half-hourly	8.35	7.02	7.71	6.51
	hourly	8.29	7.02	7.69	6.42
	monthly	3.22	2.65	2.94	2.28
WF SH	half-hourly	4.51	3.91	4.25	4.10
	hourly	4.50	3.91	4.24	4.07
	monthly	1.67	1.42	1.56	1.62

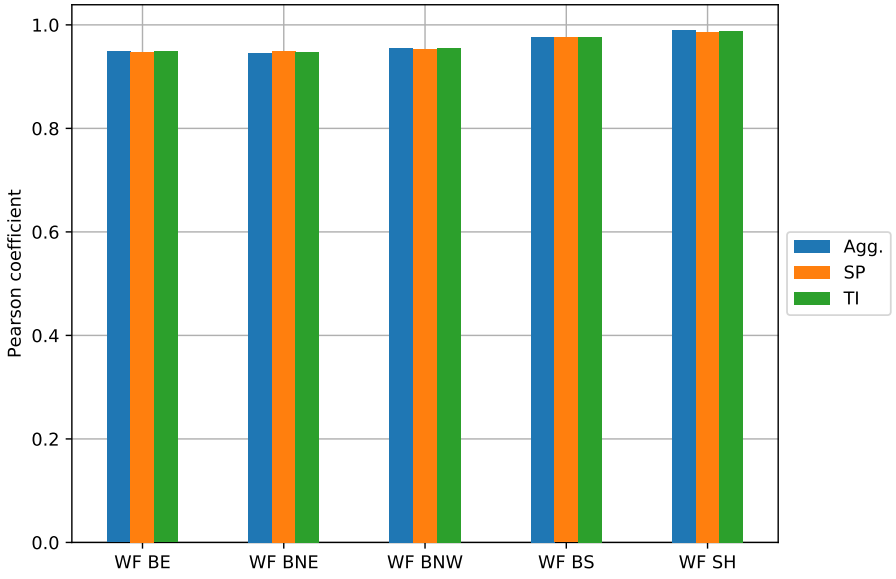


Figure A.1: *Pearson correlation coefficient of monthly time series calculated with different smoothing approaches and with simple aggregation (average of 2015 and 2016)*

A.4 Additional results of wake losses simulations

Table A.5: *Evaluation of feed-in time series calculated with different wake losses methods for the year 2015*

		RMSE [MW]				RMSE [%]				Pearson coefficient				mean bias [MW]			
		Calc.	Const.	Dena	No losses	Calc.	Const.	Dena	No losses	Calc.	Const.	Dena	No losses	Calc.	Const.	Dena	No losses
WF BE	half-hourly	3.38	3.17	3.53	3.79	59.72	56.03	62.39	67.10	0.72	0.71	0.72	0.71	0.19	-0.32	0.25	1.01
	hourly	3.24	3.03	3.39	3.66	59.15	55.30	61.84	66.87	0.74	0.73	0.74	0.73	0.22	-0.28	0.27	1.02
	monthly	0.25	0.37	0.34	1.03	4.35	6.55	6.00	18.21	1.00	1.00	1.00	1.00	0.17	-0.34	0.24	0.98
WF BNW	half-hourly	0.65	0.54	0.64	0.71	79.01	65.76	77.68	86.09	0.80	0.80	0.80	0.80	0.22	0.07	0.18	0.30
	hourly	0.62	0.51	0.61	0.68	78.06	64.38	76.58	85.47	0.82	0.81	0.82	0.81	0.21	0.08	0.17	0.29
	monthly	0.24	0.12	0.21	0.32	30.52	15.21	26.25	40.44	0.96	0.96	0.96	0.96	0.22	0.08	0.18	0.30
WF BS	half-hourly	5.12	4.54	5.30	5.94	60.68	53.82	62.85	70.44	0.80	0.78	0.79	0.78	1.26	0.38	1.45	2.59
	hourly	4.89	4.33	5.07	5.72	60.28	53.41	62.51	70.55	0.81	0.80	0.81	0.80	1.22	0.39	1.40	2.51
	monthly	1.83	0.86	2.05	3.12	20.30	9.49	22.72	34.56	0.98	0.97	0.98	0.97	1.47	0.48	1.68	2.85

Table A.6: *Evaluation of feed-in time series calculated with different wake losses methods for the year 2016*

		RMSE [MW]				RMSE [%]				Pearson coefficient				mean bias [MW]			
		Calc.	Const.	Dena	No losses	Calc.	Const.	Dena	No losses	Calc.	Const.	Dena	No losses	Calc.	Const.	Dena	No losses
WF BE	half-hourly	3.41	3.16	3.56	3.85	66.43	61.59	69.47	74.95	0.69	0.68	0.69	0.68	0.22	-0.24	0.26	0.99
	hourly	3.27	3.03	3.42	3.71	65.87	60.94	68.89	74.67	0.71	0.71	0.71	0.71	0.23	-0.21	0.27	0.97
	monthly	0.73	0.53	0.82	1.39	13.69	9.97	15.27	25.93	0.90	0.90	0.90	0.90	0.37	-0.13	0.42	1.17
WF BNW	half-hourly	0.68	0.56	0.66	0.75	101.06	83.53	97.78	112.23	0.76	0.76	0.77	0.76	0.33	0.20	0.29	0.42
	hourly	0.65	0.53	0.63	0.73	101.08	83.02	97.53	112.72	0.78	0.77	0.78	0.77	0.32	0.20	0.28	0.41
	monthly	0.34	0.22	0.30	0.43	52.83	33.35	46.42	66.07	0.96	0.95	0.96	0.95	0.33	0.20	0.29	0.41
WF BS	half-hourly	5.22	4.61	5.41	6.07	67.40	59.43	69.81	78.27	0.75	0.74	0.75	0.74	1.30	0.58	1.49	2.66
	hourly	4.96	4.37	5.15	5.81	66.79	58.83	69.26	78.19	0.78	0.76	0.77	0.76	1.26	0.56	1.43	2.56
	monthly	1.74	0.78	1.95	3.04	20.39	9.09	22.77	35.53	0.98	0.98	0.98	0.98	1.48	0.59	1.69	2.88

A.5 Additional results of single wind turbine model simulations

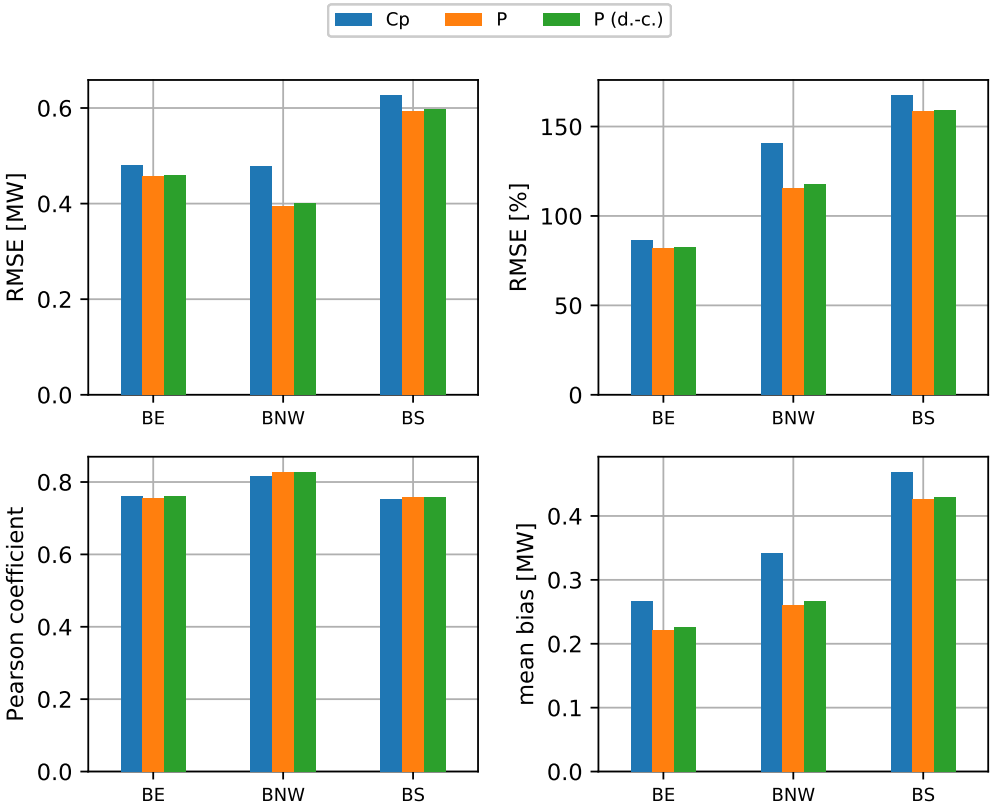


Figure A.2: *Statistical metrics of hourly feed-in time series of single wind turbines calculated with MERRA weather data (average of 2015 and 2016)*

Table A.7: *Evaluation of feed-in time series calculated for the single turbine model with open_FRED data in 2015*

		RMSE [MW]			RMSE [%]			Pearson coefficient			mean bias [MW]		
		Cp	P	P (d.-c.)	Cp	P	P (d.-c.)	Cp	P	P (d.-c.)	Cp	P	P (d.-c.)
BE	half-hourly	0.43	0.42	0.42	70.75	69.19	69.10	0.73	0.73	0.73	0.11	0.07	0.07
	hourly	0.42	0.41	0.41	70.09	68.42	68.27	0.75	0.74	0.75	0.11	0.07	0.07
BNW	half-hourly	0.44	0.37	0.37	114.85	97.23	98.31	0.77	0.78	0.78	0.24	0.16	0.17
	hourly	0.42	0.36	0.36	114.74	96.66	97.77	0.79	0.79	0.80	0.23	0.16	0.16
	monthly	0.25	0.17	0.18	67.51	47.07	48.64	0.97	0.96	0.96	0.24	0.16	0.17
BS	half-hourly	0.49	0.46	0.47	114.19	107.74	108.58	0.74	0.75	0.75	0.26	0.22	0.22
	hourly	0.48	0.45	0.45	115.15	108.30	109.17	0.76	0.76	0.76	0.26	0.22	0.22
	monthly	0.30	0.26	0.26	67.87	57.96	59.89	0.94	0.93	0.94	0.27	0.23	0.23

Table A.8: *Evaluation of feed-in time series calculated for the single turbine model with open_FRED data in 2016*

		RMSE [MW]			RMSE [%]			Pearson coefficient			mean bias [MW]		
		Cp	P	P (d.-c.)	Cp	P	P (d.-c.)	Cp	P	P (d.-c.)	Cp	P	P (d.-c.)
BE	half-hourly	0.44	0.42	0.43	81.87	78.41	79.62	0.70	0.70	0.70	0.11	0.06	0.07
	hourly	0.43	0.41	0.41	81.60	78.07	79.23	0.71	0.71	0.71	0.11	0.07	0.07
BNW	half-hourly	0.43	0.36	0.37	134.22	110.82	113.48	0.76	0.77	0.77	0.27	0.19	0.20
	hourly	0.42	0.35	0.35	134.38	110.28	113.04	0.78	0.79	0.79	0.26	0.19	0.19
	monthly	0.28	0.20	0.21	86.91	61.74	65.05	0.94	0.95	0.95	0.26	0.19	0.20
BS	half-hourly	0.49	0.46	0.47	137.27	128.12	130.50	0.70	0.70	0.70	0.26	0.22	0.22
	hourly	0.47	0.44	0.45	139.03	129.39	131.83	0.72	0.72	0.72	0.26	0.21	0.22
	monthly	0.29	0.25	0.26	80.30	67.45	70.71	0.96	0.97	0.96	0.27	0.22	0.23

Table A.9: *Evaluation of feed-in time series calculated for the single turbine model with MERRA data in 2015*

		RMSE [MW]			RMSE [%]			Pearson coefficient			mean bias [MW]		
		Cp	P	P (d.-c.)	Cp	P	P (d.-c.)	Cp	P	P (d.-c.)	Cp	P	P (d.-c.)
BE	hourly	0.49	0.47	0.47	82.93	79.34	79.22	0.76	0.75	0.76	0.28	0.23	0.24
BNW	hourly	0.51	0.42	0.43	137.58	114.34	115.77	0.82	0.83	0.83	0.37	0.28	0.29
	monthly	0.37	0.29	0.29	104.13	80.37	82.24	0.98	0.98	0.98	0.36	0.28	0.29
BS	hourly	0.64	0.61	0.61	154.42	146.66	147.15	0.76	0.76	0.76	0.48	0.44	0.44
	monthly	0.49	0.45	0.46	116.58	106.40	107.46	0.95	0.95	0.96	0.48	0.44	0.45

Table A.10: *Evaluation of feed-in time series calculated for the single turbine model with MERRA data in 2016*

		RMSE [MW]			RMSE [%]			Pearson coefficient			mean bias [MW]		
		Cp	P	P (d.-c.)	Cp	P	P (d.-c.)	Cp	P	P (d.-c.)	Cp	P	P (d.-c.)
BE	hourly	0.47	0.44	0.45	89.71	84.87	85.44	0.76	0.76	0.76	0.25	0.21	0.21
BNW	hourly	0.45	0.37	0.37	143.18	116.56	119.20	0.82	0.83	0.83	0.32	0.24	0.25
	monthly	0.32	0.24	0.25	104.59	78.25	81.30	0.98	0.98	0.98	0.32	0.24	0.24
BS	hourly	0.62	0.58	0.59	180.87	170.23	171.53	0.75	0.76	0.76	0.46	0.41	0.42
	monthly	0.46	0.42	0.42	137.27	123.74	125.61	0.98	0.99	0.99	0.45	0.41	0.41

Bibliography

Arbeitsgruppe Erneuerbare Energien-Statistik AGEE-Stat. Zeitreihen zur Entwicklung der erneuerbaren Energien in Deutschland. Technical report, Umweltbundesamt (UBA), 2018. URL https://www.erneuerbare-energien.de/EE/Navigation/DE/Service/Erneuerbare_Energien_in_Zahlen/Zeitreihen/zeitreihen.html.

J. Ahrendts and S. Kabelac. *Das Ingenieurwissen. Technische Thermodynamik*. Springer-Verlag, Berlin-Heidelberg, 34 edition, 2014.

J. F. Ainslie. Calculating the flowfield in the wake of wind turbines. *Journal of Wind Engineering and Industrial Aerodynamics*, 27:213–224, 1988. doi: https://doi.org/10.1016/0167-6105(88)90037-2.

R. J. Barthelmie, L. Folkerts, G. C. Larsen, K. Rados, S. C. Pryor, S. T. Frandsen, B. Lange, and G. Schepers. Comparison of Wake and Model Simulations and with Offshore and Wind Turbine and Wake Profiles and Measured by Sodar. *Journal of athmospheric and oceanic technology*, 23:888–901, 2005.

Rebecca J. Barthelmie, Kurt S. Hansen, and Sara C. Pryor. Meteorological Controls on Wind Turbine Wakes. *Proceedings of the IEEE*, 101(4), April 2013.

H. Berendsen. *A Student's Guide to Data and Error Analysis*, chapter Chapter 5 Processing of experimental data, pages 53–70. 2011. doi: .org/10.1017/CBO9780511921247.006.

Marcus Biank. Methodology, Implementation and Validation of a Variable Scale Simulation Model for Windpower based on the Georeferenced Installation Register of Germany. Master's thesis, Albert-Ludwigs-Universität Freiburg, #sep# 2014.

Bundesministerium für Wirtschaft und Energie. Erneuerbare Energien in Zahlen. Nationale und internationale Entwicklung im Jahr 2016. Broschüre, 2017. URL https://www.bmwi.de/Redaktion/DE/Publikationen/Energie/erneuerbare-energien-in-zahlen-2016.pdf?__blob=publicationFile&v=8.

Climate Limited-area Modelling Community. Overview. The COSMO-CLM Model System and its extensions, 2018. URL https://www.clm-community.eu/index.php?menuid=219.

Consortium for Small-Scale Modelling. General Description: General Description of the COSMO-Model, 2011. URL http://www.cosmo-model.org/content/model/general/default.htm.

Deutscher Wetterdienst. Weitere Erläuterungen zur Druckgradientkraft. a. URL http://www.dwd.de/DE/service/lexikon/begriffe/D/Druckgradient_pdf.pdf?__blob=publicationFile&v=4.

Deutscher Wetterdienst. ICAO-Standardatmosphäre (ISA). b. URL http://www.dwd.de/DE/service/lexikon/begriffe/S/Standardatmosphaere_pdf.pdf?__blob=publicationFile&v=3.

MR Elkinton, AL Rogers, and JG McGowan. An investigation of wind-shear models and experimental data trends for different terrains. *Wind Engineering*, 30(4): 341–350, 2006. doi: 10.1260/030952406779295417.

Ludwig Fahrmeir, Christian Heumann, Rita Künstler, Iris Pigeot, and Gerhard Tutz. *Statistik. Der Weg and zur Datenanalyse*. Springer-Verlag Berlin Heidelberg, 8 edition, 2016. doi: 10.1007/978-3-662-50372-0.

Sten Frandsen, Rebecca Barthelmie, Sara Pryor, Ole Rathmann, Søren Larsen, Jørgen Højstrup, and Morten Thøgersen. Research Analytical and Modelling of Wind and Speed and Article Deficit and in Large and Offshore Wind and Farms. *Wind Energy*, 9:39–53, 2006. doi: 10.1002/we.189.

Free Software Foundation. Gnu operating system - licenses, 2018. URL http://www.gnu.org/licenses/.

Robert Gasch and Jochen Twele. *Windkraftanlagen - Grundlagen, Entwurf, Planung und Betrieb*. Vieweg+Teubner Verlag, 4 edition, 2005. doi: 10.1007/978-3-322-99446-2.

Global Modeling and Assimilation Office. File Specification and for MERRA Products. Technical report, 2012. URL https://gmao.gsfc.nasa.gov/pubs/docs/Bosilovich785.pdf.

Erich Hau. *Windkraftanlagen - Grundlagen, Technik, Einsatz, Wirtschaftlichkeit*. Springer-Verlag, Berlin-Heidelberg, 5 edition, 2014.

N. O. Jensen. A note on wind generator interaction. (Risø-M; No. 2411). 1983. URL http://orbit.dtu.dk/files/55857682/ris_m_2411.pdf.

Kaspar Knorr. *Modellierung von raum-zeitlichen Eigenschaften der Windenergieeinspeisung für wetterdatenbasierte Windleistungssimulationen*. PhD thesis, Universität Kassel, 2016.

Stephan Kohler, Annegret-Cl. Agricola, and Hannes Seidl. dena-Netzstudie II. Integration erneuerbarer Energien in die deutsche Stromversorgung im Zeitraum 2015 – 2020 mit Ausblick 2025. Technical report, 2010.

Neon Neue Energieökonomik GmbH. Workshop zur offenen Energiesystemmodellierung. URL https://neon-energie.de/workshop-offene-energiesystem-modellierung/.

Per Nørgaard and Hannele Holttinen. A Multi-Turbine and Power Curve Approach. In *Nordic Wind Power Conference NWPC'04, 1.–2.3.2004*, 2000.

Oemof Developer Group. About oemof - readthedocs. URL http://oemof.readthedocs.io/en/stable/about_oemof.html.

Openmod Initiative. Openmod in a nutshell. URL http://openmod-initiative.org/manifesto.html.

Steven Pawson. Modern-Era Retrospective analysis for Research and Applications, Version 2, 2017. URL https://gmao.gsfc.nasa.gov/reanalysis/MERRA-2/.

pvlib python. Modelchain, 2018. URL http://pvlib-python.readthedocs.io/en/latest/modelchain.html.

Volker Quaschning. *Regenerative Energiesysteme Kapitel 6-12.* Hanser Verlag, München, 7 edition, 2011. ISBN 9783446427327.

Reiner Lemoine Institut. oemof - open energy system modelling framework. The Project, a. URL https://reiner-lemoine-institut.de/en/oemof/.

Reiner Lemoine Institut. open_FRED: open feed-in time series based on a Renewable Energy Database, b. URL https://reiner-lemoine-institut.de/en/open_fred-open-feed-time-series-based-renewable-energy-database/.

Douwe J. Renkema. Validation of wind turbine wake and models. Using wind farm data and wind tunnel measurements. Master's thesis, Delft University of Technology, 2007.

Rabia Shakoor, Mohammad Yusri Hassan, Abdur Raheem, and Nadia Rasheed. The Modelling and of Wind and Farm Layout and Optimization and for the Reduction and of Wake and Losses. *Indian Journal of Science and Technology*, 8(17), 2015. doi: 10.17485/ijst/2015/v8i17/69817.

Ed Sharp. *Spatiotemporal disaggregation of GB scenarios depicting increased wind capacity and electrified heat demand in dwellings* . PhD thesis, UCL Energy Institute, 2015.

Iain Staffell and Richard Green. How Does Wind Farm Performance Decline with Age. *Renewable Energy*, 2014. doi: https://doi.org/10.1016/j.renene.2013.10.041.

Iain Staffell and Stefan Pfenninger. Using Bias-Corrected Reanalysis to Simulate Current and Future Wind Power Output. 2005. URL http://www.sciencedirect.com/science/article/pii/S0360544216311811.

J. Steppeler, G. Doms, and G. Adrian. Das Lokal-Modell LM. *promet*, 27(3/4): 123–128, Juni 2002. URL https://www.dwd.de/SharedDocs/downloads/DE/modelldokumentationen/nwv/veroeffentlichungen/Promet/dwd_promet_27_3_4_k3_lm.html.

Lasse Svenningsen. Proposal of an Improved Power Curve Correction. In *European Wind Energy Conference & Exhibition 2010*, number Switzerland., 2010.

United Nations. Sustainable Development Goal 7, 2017a. URL https://sustainabledevelopment.un.org/sdg7.

United Nations. Sustainable Development Goal 13, 2017b. URL https://sustainabledevelopment.un.org/sdg13.

United Nations. Sustainable Development Knowledge Platform, 2017c. URL https://sustainabledevelopment.un.org/sdgs.

WindEurope. Wind in power 2017. Annual combined onshore and offshore wind energy statistics. Technical report, 2018. URL https://windeurope.org/wp-content/uploads/files/about-wind/statistics/WindEurope-Annual-Statistics-2017.pdf.