TECHNISCHE UNIVERSITÄT WIEN

Institute of Energy Systems and Electrical Drives

Modelling Renewable Energy Sources into a Grid Simulation

Master Thesis

for the acquisition of the academic degree

Master of Science

Faculty of Electrical Engineering and Information Technology
Vienna University of Technology

submitted by

Mathias Hofschweiger, 11810761

supervised by

Main Advisor: Univ.Prof. Dipl.-Ing. Dr.sc.techn. Bernd Klöckl

Co-advisor: Ao. Univ.Prof. i.R. Dipl.-Ing. Dr.techn. Helmut Leeb

Vienna, July 29, 2024

Studienkennzahl: 066 460
Studienbezeichnung: Masterstudium Physikalische Energie- und Messtechnik

Acknowledgements

I want to thank my supervisor Brixius, for his support, patience, and scientific input. Furthermore, I want to thank Prof. Leeb and Prof. Klöckl for making this thesis possible.

Special thanks also go to my family and my girlfriend, who encouraged me when things were not going so well and always stood behind me.

I am also in debt to my friends for their time and for always being there for me when I need them.

This accomplishment would not have been possible without them.

Thank you,

Mathias Hofschweiger

Eidesstattliche Erklärung

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig und ohne fremde Hilfe verfasst, andere als die angegebenen Quellen und Hilfsmittel nicht benutzt bzw. die wörtlich oder sinngemäß entnommenen Stellen als solche kenntlich gemacht habe. Zudem bestätige ich, dass keine künstliche Intelligenz (KI) für die Verfassung der Arbeit bzw. für Teile der Arbeit zum Einsatz gekommen ist.

Wien, am .....

[redacted]
Mathias Hofschweiger

Einverständniserklärung zur Plagiatsprüfung

Ich nehme zur Kenntnis, dass die vorgelegte Arbeit mit geeigneten und dem derzeitigen Stand der Technik entsprechenden Mitteln (Plagiat-Erkennungssoftware) elektronisch-technisch überprüft wird. Dies stellt einerseits sicher, dass bei der Erstellung der vorgelegten Arbeit die hohen Qualitätsvorgaben im Rahmen der ausgegebenen der an der TU Wien geltenden Regeln zur Sicherung guter wissenschaftlicher Praxis - „Code of Conduct“ (Mitteilungsblatt 2007, 26. Stück, Nr. 257 idgF.) an der TU Wien eingehalten wurden. Zum anderen werden durch einen Abgleich mit anderen studentischen Abschlussarbeiten Verletzungen meines persönlichen Urheberrechts vermieden.

Wien, am .....

[redacted] .....
Mathias Hofschweiger

Kurzfassung

Eine zuverlässige Versorgung mit elektrischer Energie ist essentiell für eine moderne, florierende Gesellschaft und Industrie. Durch ein flächendeckendes Netzwerk, bestehend aus mehreren Spannungsebenen, werden die einzelnen Kunden mit Elektrizität versorgt. Angesichts des Klimawandels und der Notwendigkeit den globalen CO_2-Ausstoß zu reduzieren, nimmt die Bedeutung erneuerbarer Energien wie Wind- und Solarenergie stetig zu. Diese bieten erhebliche Vorteile in Bezug auf Umweltfreundlichkeit und Ressourcenschonung, bringen jedoch auch neue Herausforderungen mit sich. Insbesondere die volatile und wetterabhängige Erzeugung stellen hohe Anforderungen an die Stabilität und Zuverlässigkeit des Netzwerks. Deshalb gewinnt auch die Thematik der Energiespeicherung immer mehr an Bedeutung, um die fluktuierende Erzeugung auszugleichen. Da der aktuelle Bedarf jederzeit von Erzeugern ausgeglichen werden muss, kann es bei einer schlagartigen Änderung zur Instabilität des Netzes kommen. Neben der Spannungsstabilität gehört das Gewährleisten einer konstanten Frequenz, welche in Österreich 50 Hz beträgt, zu den wichtigsten Aufgaben der Netzbetreiber. Während kleine Abweichungen vom System verkraftbar sind, kann es bei größeren Abweichungen zu einem Ausfall des Netzes kommen.

Aus genannten Gründen ist es daher im Interesse der einzelnen Marktteilnehmer und insbesondere der Netzbetreiber, über zuverlässige Netzmodelle zu verfügen. Abhängig von den zu untersuchenden Phänomenen, wird entschieden welche Art von Netzmodell verwendet wird. Soll beispielsweise das Verhalten des Netzes bei Störungen (z.B. Kurzschlüssen) untersucht werden, müssen dynamische Modelle herangezogen werden. Beim störungsfreien Betrieb reicht es stationäre Modelle zu verwenden. Diese zeichnen sich durch ihre, verglichen mit dynamischen Modellen, geringere Komplexität aus. Stationäre Modelle bilden eine solide Grundlage und können gegebenenfalls zu einer dynamischen Simulation ausgebaut werden.

Ziel dieser Arbeit ist es Windkraftwerke, Photovoltaik-Anlagen und Batteriespeicher, welche in ein stationäres Netzmodell eingebunden werden, zu modellieren. Die Modelle bestehen wiederum aus mehreren Teilmodellen, die die wichtigsten Komponenten der einzelnen Erzeuger darstellen. So besteht beispielsweise das Windradmodell (Typ D) aus Rotor, Generator, generatorseitigen Konverter und netzseitigen Konverter. Sämtliche Modelle beschränken sich auf den stationären Zustand, wodurch Störungen im Betrieb nicht dargestellt werden können. Als Input für die Modelle dienen Parameter aus der Literatur, Datenblättern und Wetterdaten (z.B. Einstrahlungsdaten und Windgeschwindigkeiten), welche aus verschiedenen Datenbanken bezogen werden können. Anschließend werden die Modelle validiert und Vorschläge für Erweiterungen angeführt.

Die Arbeit ist gegliedert in die Bereiche: 1) Theoretische Grundlagen der erneuerbaren Erzeuger Windkraft, Photovoltaik und Batteriespeicher, 2) Vorstellung der einzelnen Modelle und ihrer Sub-Modelle, 3) Validierung der Modelle und 4) Schlussfolgerung und Ausblick für Erweiterungen.

Abstract

Nowadays, a reliable supply of electrical energy is essential for a modern, flourishing society and industry. Individual customers are supplied with electricity through a nationwide network consisting of several voltage levels. In view of climate change and the need to reduce CO_2 emissions, the importance of renewable energies such as wind and solar energy is constantly increasing. These offer considerable advantages in terms of environmental friendliness and resource conservation, but also bring new challenges for the energy supply. In particular, the volatile and weather-dependent generation of these energy sources places high demands on the stability and reliability of the network. For this reason, the issue of energy storage is also becoming increasingly important in order to balance out the fluctuating generation of renewable energy sources. As the current demand must be balanced at all times by the power plants feeding into the grid, a sudden change in demand or feed-in can lead to grid instability. In addition to voltage stability, ensuring a constant frequency, which in Austria is 50 Hz, is one of the most important tasks of the grid operators. While small deviations can be tolerated by the system, excessive deviations can lead to grid failure.

For these reasons, it is therefore in the interest of the individual market participants and in particular the grid operators to have reliable grid models. The type of grid model to be used depends on the phenomena to be investigated. For example, if the behaviour of the grid is to be investigated in the event of faults, e.g., short circuits, dynamic models must be used. For fault-free operation, it is sufficient to use stationary models. These are characterized by their lower complexity compared to dynamic models. Stationary grid models form a solid basis and can be expanded into a dynamic simulation if necessary.

The aim of this work is to model wind power plants, photovoltaic systems and battery storage systems that are integrated into a stationary grid model. The models in turn consist of several sub-models, which represent the most important components of the individual power plants. For example, the wind turbine (Type D) consists of a rotor, generator, generator-side converter and grid-side converter. All models are limited to the stationary state, which means that disturbances during operation cannot be represented. Parameters from the literature, data sheets and weather data, e.g., irradiation data and wind speeds, which can be obtained from various databases, serve as input for the models. The models are then validated and suggestions for extensions are made.

The work is divided into the following sections: 1) Theoretical foundations of the renewable generators wind power, photovoltaic and battery storage, 2) Presentation of the individual models and their sub-models, 3) Validation of the models and 4) Conclusion and outlook for extensions.

Contents

Abbrevations

AC	Alternating Current
AM	Air Mass
BESS	Battery Energy Storage System
BHI	Beam Horizontal Irradiance
DC	Direct Current
ECD	Equivalent Circuit Diagram
EoT	Equation of Time
GHI	Global Horizontal Irradiance
GSC	Grid Side Converter
HAWT	Horizontal Axis Wind Turbine
ISC	Short Circuit Current
MAPE	Mean Absolute Percentage Error
MPP	Maximum Power Point
MPPT	Maximum Power Point Tracker
MSC	Machine Side Converter
NOCT	Nominal Operating Cell Temperature
OCV	Open Circuit Voltage
PMSG	Permanent Magnet Synchronous Generator
p.u.	Per Unit
PV	Photovoltaic
PWM	Pulse Width Modulation
SCIG	Squirrel Cage Induction Generator
SOC	State of Charge
SP	Single Phase
STC	Standard Testing Condition
TPN	Two-port Network
TSR	Tip Speed Ratio
UTC	Coordinated Universal Time
VAWT	Vertical Axis Wind Turbine
WECS	Wind Energy Conversion System
WRIG	Wound Rotor Induction Generator
WRSG	Wound Rotor Synchronous Generator

List of Symbols

Wind Turbine

Symbol	Name	Unit
β	Pitch angle	°
c_p	Power coefficient	-
η	Generator efficiency	-
η_{GSC}	GSC efficiency	-
η_{MSC}	MSC efficiency	-
I_{DC}	DC-link current	A
i_{ds}	Direct stator current	p.u.
I_g	Grid current	p.u.
i_{qs}	Quadrature stator current	p.u.
λ	Tip-speed-ratio	-
L_{ds}	Direct stator inductance	p.u.
L_{qs}	Quadrature stator inductance	p.u.
n_{max}	Maximal rotational speed	rpm
ω	Angular velocity	rad/s
ω_e	Electrical speed	p.u.
p	Number of pole pairs	-
P_{GSC}	Active power GSC	W
P_{mech}	Mechanical power	W
P_{MSC}	Active power MSC	W
P_{rated}	Rated power	W
ψ_{pm}	Magnetic flux	p.u.
P_s	Active generator power	p.u.
Q_s	Reactive generator power	p.u.
r	Rotor blade length	m
ρ	Air density	kg/m^3
R_s	Stator resistance	p.u.
T_E	Electrical torque	p.u.
T_W	Mechanical torque	p.u.
U_{DC}	DC-link voltage	V
u_{ds}	Direct stator voltage	p.u.
U_g	AC grid voltage	V
u_{qs}	Quadrature stator voltage	p.u.
U_s	Stator voltage	p.u.
v_{cut-in}	Cut-in wind speed	m/s
$v_{cut-out}$	Cut-out wind speed	m/s
v_{rated}	Rated wind speed	m/s
v_w	Wind speed	m/s

Photovoltaic

Symbol	Name	Unit
α_{isc}	Temperature coefficient of I_{osc}	$\%/°C$
α_s	Solar altitude angle	°
α_{voc}	Temperature coefficient of V_{oc}	$\%/°C$
β	Slope of modules	°
δ	Declination	°
G_{pu}	Relative irradiance	-
γ	Surface azimuth of modules	°
γ_s	Solar azimuth angle	°
G_b	Horizontal beam irradiance	W/m^2
G_{bT}	Tilted beam irradiance	W/m^2
G_{dT}	Tilted diffuse irradiance	W/m^2
G_{on}	Extraterrestrial irradiance	W/m^2
G_{rT}	Tilted reflected irradiance	W/m^2
G_{sc}	Average extraterrestrial irradiance	W/m^2
G_T	Tilted global irradiance	W/m^2
I_{str}	String current	A
I_0	Dark saturation current	A
I_{ompp}	Maximum power current at STC	A
I_{osc}	Short circuit current at STC	A
I_{PV}	Photo current	A
kt	Clearness index	-
kt'	Modified clearness index	-
L_{loc}	Longitude of location	°
L_{st}	Standard meridian	°
m_a	Modulation index	-
n	Number of day in year	-
N_p	Number of parallel strings	-
N_s	Number of modules in series	-
ω	Hour angle	°
Φ	Latitude of position	°
R_L	Ohmic resistance of inverter	Ω
ρ	Ground albedo	-
$R_{s,c}$	Module serial resistance	Ω
$R_{sh,c}$	Module shunt resistance	Ω
T_a	Air temperature	°C
T_c	Cell temperature	°C
θ	Angle of incident	°
θ_z	Zenith angle	°
V_{LN}	Line-to-neutral voltage	V
V_{str}	String voltage	V
V_{ompp}	Maximum power voltage at STC	V
V_{ooc}	Open circuit voltage at STC	V

Battery Energy Storage System

Symbol	Name	Unit
α_{NC}	Alpha cycle parameter	-
α_t	Alpha time parameter	-
β_{NC}	Beta cycle parameter	K^{-1}
β_t	Beta time parameter	K^{-1}
$C_{fade,cal}$	Calendar capacity fade	%
$C_{fade,cyc}$	Cycle capacity fade	%
C_{init}	Initial capacity	Ah
C_{usable}	Usable capacity	Ah
C_{CF}	Remaining capacity	%
I_{batt}	Battery current	A
M_{charge}	Maximum charge rate	-
$M_{discharge}$	Maximum discharge rate	-
N_p	Battery cells in parallel	-
N_s	Battery cells in series	-
NC	Number of equivalent cycles	-
P_{Grid}	Power surplus / shortage of grid	W
Q_{charge}	Maximum charge	Ah
$Q_{discharge}$	Maximum discharge	Ah
R_L	Ohmic resistance of rectifier	Ω
R_S	Ohmic resistance battery cell	Ω
SOC	State of charge	%
SOC_{max}	Maximum state of charge	%
SOC_{min}	Minimum state of charge	%
T	Cell temperature	K
t_{month}	Battery age	-
V_{batt}	Battery voltage	V
V_c	Cell voltage	V
V_d	Full bridge rectifier voltage	V
V_{LL}	Line-to-line voltage	V
V_{OC}	Open circuit voltage battery cell	V

1. Introduction

The ongoing transformation of the energy sector towards a sustainable and low-carbon future is of major importance in response to climate change. One of the driving forces behind this transformation is the increased use of renewable energy sources such as wind and solar power. Their increasing integration into the electrical grid represents one of the most significant challenges of modern energy supply. In contrast to conventional energy generators such as coal, gas and nuclear power plants, renewable generators are highly dependent on environmental conditions such as wind speed and solar radiation. This volatility can lead to instability in the grid and therefore requires careful planning and control. During periods of low output, the lack of power must be compensated with the help of conventional power plants, such as gas-fired power plants and various storage technologies, such as pumped storage power plants and battery energy storage systems. In times of high output, the surplus power must be stored.

The modelling and simulation of renewable energy generators and storages within the electricity grid is therefore crucial to understanding their impact on grid stability, reliability and performance. By developing accurate and realistic grid models, grid operators and planners can better respond to the challenges associated with the integration of renewable energy. To do this, all components, such as power plants, power lines, transformers, switchgear and consumers, e.g., industry and households, must be modelled with sufficient accuracy. It is important to find a balance between the complexity of the model and the associated computing time.

The aim of this thesis is to investigate how wind power, solar energy and battery energy storage systems can be modelled realistically and with sufficient accuracy and which simplifications can be made for use in a stationary grid simulation. As part of this work, it was decided to pursue a modular approach, which divides the models into sub-models. These models reflect the most important components of the individual power plants. This makes it possible to facilitate any subsequent changes, as well as the expansion or implementation of other topologies.

A variable-speed, gearless model with a full converter was selected for the wind power model. The model consists of four sub-models: rotor, permanently excited synchronous generator, a machine side converter and a grid side converter. A time series with wind speeds and parameters from data sheets and literature are used as input.

The photovoltaic model in turn consists of the PV module and the inverter sub-models. Here, a time series with irradiation data on a horizontal surface serves as input. These are converted to an tilted surface using extensive irradiation model. Specific data sheet values of the PV module are also used as parameters.

The inverter, rectifier and battery module sub-models in turn form the battery energy storage model. The input here is a time series with electricity that needs to be stored or withdrawn or, again, component-specific parameters. Furthermore a model is used to take battery ageing and the associated loss of capacity into account.

The weather-specific data is available from various freely usable databases, such as Solcast or Copernicus. The models are implemented in MATLAB.

The work is divided into five chapters. After the introduction, the second chapter deals with the structure and the different topologies of wind power, solar systems and battery energy storage systems. This is followed by a discussion of the physical and electro-technical phenomena and fundamentals.

After the theoretical basics, the implemented models are described in the third chapter. The three main renewable models are composed of several sub-models. The input variables and the resulting output variables of the sub-models, as well as the required parameters, are discussed. In the sub-chapter on photovoltaics, the calculation of irradiation on an tilted surface and the steps required for this are explained. Furthermore, the different assumptions and simplifications made in the modelling are explained.

In the fourth chapter, the three models wind, solar power and battery energy storage systems are validated. This is achieved by comparing the results with measurement data from existing systems, other scientific studies and freely available data sets. Based on this, the applicability and possible limits of the applicability of the implemented models are discussed.

The final chapter summarizes the results of the work and recapitulates the process. The most important findings and difficulties are highlighted. Last but not least, an outlook is given on possible areas for improvement and further development of the models.

2. Theory

This chapter covers the theoretical foundations of wind power, photovoltaic and battery energy storage systems. The basic principles presented serve as a starting point for the modelling of the renewables in the next chapter. The structure of the systems, the underlying physical phenomena and equivalent circuits are presented. Chapter 2.1 covers the basics of wind energy conversion systems, Chapter 2.2 deals with the fundamentals of photovoltaic and Chapter 2.3 describes essentials for battery energy storage systems.

2.1. Wind Energy Conversion Systems

Wind Energy Conversion Systems (WECS) have witnessed a steady growth in power generation over the last decades. WECS transfer the kinetic energy of the wind into mechanical energy with the help of a wind turbine. The mechanical energy is then converted into electrical energy with a generator. In Figure 2.1 the basic concept of power generation is illustrated.

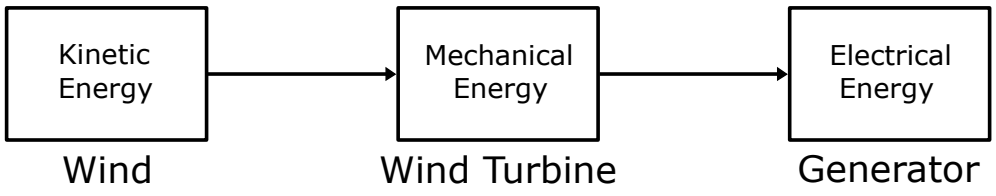


Figure 2.1.: Stages of Wind Energy Conversion.

The following sub-chapters provide a rough overview of the fundamentals of WECS.

2.1.1. Wind

Wind is caused by pressure gradients and regional differences in the warming of air masses by the sun. Solar radiation creates local and regional variations in the heating of the Earth's surface depending on the surface structure. These temperature differences lead to buoyancy effects, which in turn cause air currents and pressure differences. Wind speeds highly depend on seasonal, daily and local differences. To find a suitable location for a WECS, it is therefore important to know exactly what the local wind conditions are like, p. 410 [1].

The kinetic energy contained in the wind is calculated as, p. 412 [1]:

$$E = \frac{1}{2} m v_w^2 \quad , \tag{2.1}$$

where m is the mass of the moving air and v_w is the wind velocity. The mass flow rate of the air through the turbine's swept area can be expressed as, p. 412 [1]:

$$\dot{m} = \rho A v_w \quad , \tag{2.2}$$

where ρ is the air density and A is the area where the air flow passes through. The power contained in the wind is therefore determined as followed, p. 33 [2]:

$$P = \frac{1}{2}\rho A v_w^3 \quad . \tag{2.3}$$

This equation shows that the available power in the wind is proportional to the cube of the wind speed, highlighting the significant impact of wind speed on the potential power output of a wind turbine.

2.1.2. Betz Law

WECS can not completely convert the power in the wind, because the air masses cannot be slowed down to zero as some kinetic energy has to remain to move the air away to avoid congestion, p. 34 [2]. Betz Law defines the maximum possible energy that can be extracted from the wind by a wind turbine. According to Betz Law, no wind turbine can capture more than 59.3 % (the Betz limit) of the kinetic energy in the wind, p. 34 [2]. The theoretical maximum efficiency is given by, p. 34 [2]:

$$P_B = \frac{1}{2}\rho A v_w^3 c_{pBetz} = 0.593 \cdot \frac{1}{2}\rho A v_w^3 \quad . \tag{2.4}$$

2.1.3. Power Coefficient

The power coefficient is determined as the fraction of the extracted mechanical power and the power in the wind, p. 6 [3]:

$$c_p = \frac{P_{mech}}{P_w} \quad . \tag{2.5}$$

The power coefficient is crucial for assessing the performance of a wind turbine. The already introduced theoretical limit for $c_p = 0.593$ can not be reached by real turbines. Due to aerodynamic losses, mechanical inefficiencies and friction, the real power coefficient is significant lower. Maximizing the power coefficient involves optimizing blade design, turbine control systems, and site selection to make the best possible use of wind energy. Modern wind turbines usually reach a power coefficient of up to 0.5, p. 166 [4].

To describe the power coefficient mathematically, it is usually specified as a function of the blade pitch angle β and the tip speed ratio λ in literature. For stall regulated wind turbines, β is a fixed value and c_p is only a function of λ. Many different approaches exist in the literature to describe the power coefficient. However manufacturer documentation shows that the individual power curves are similar. Therefore, scientific papers often use general approximations to describe c_p, p. 562 [2].
The necessary equations can be taken from [5] for example. A general approach is shown below, p. 563 [2]:

$$c_p = c_1 \left(\frac{c_2}{\lambda_i} - c_3\beta - c_4 \right) \exp\left(-\frac{c_5}{\lambda_i} \right) + c_6\lambda \tag{2.6}$$

where

$$\frac{c_7}{\lambda_i} = \frac{c_9}{\lambda + c_8\beta} - \frac{c_{10}}{\beta^3 + 1} \tag{2.7}$$

The tip speed ratio λ is determined by, p. 21 [2]:

$$\lambda = \frac{\omega r}{v_w} \quad , \tag{2.8}$$

where ω is the angular velocity of the rotor. c_1-c_9 are constants and have different values for different rotor blades. The relationship between the power coefficient c_p and λ for different blade pitch angles β is illustrated in Figure 2.2.

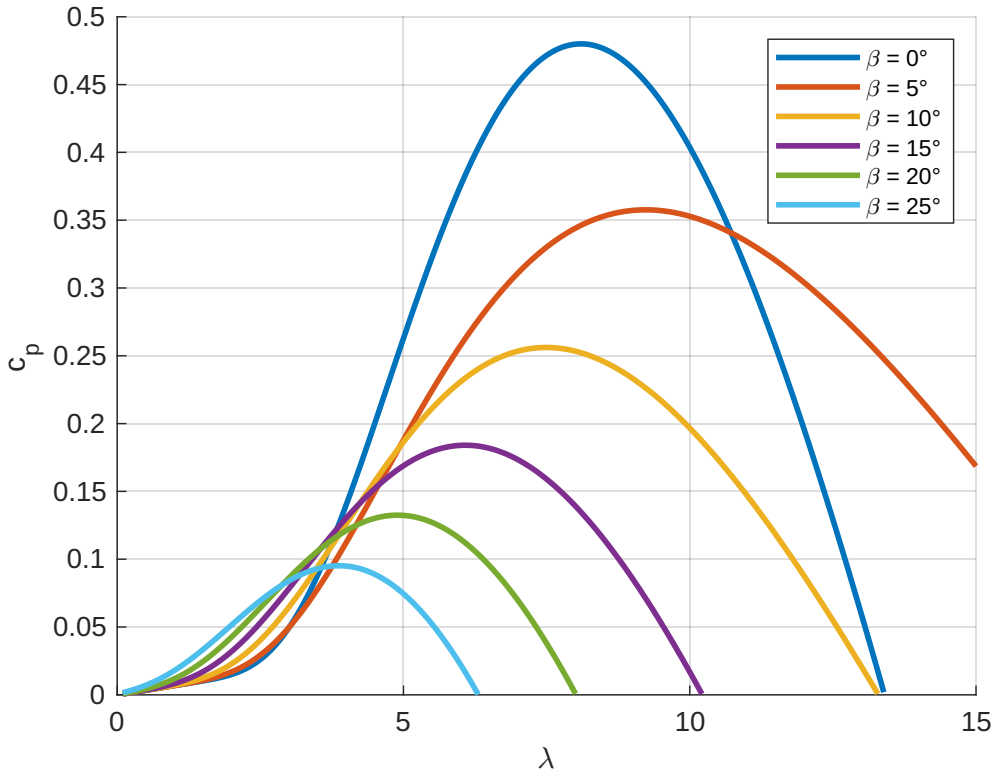


Figure 2.2.: Power coefficient c_p vs tip speed ratio λ for different blade pitch angles β

2.1.4. Power Curve

As already mentioned, the power contained in the wind, increases with the cube of the wind velocity. With the previously introduced power coefficient, the power extracted by the rotor can be calculated as:

$$P_m = \frac{1}{2}\rho\pi r^2 c_p v_w^3 \quad . \tag{2.9}$$

The power curve of a wind turbine illustrates the relationship between the wind speed and the power output of the turbine. It typically consists of four key regions, p. 35 [2]:

- **Cut-in speed:** The speed where the wind turbine start to deliver power. The cut-in wind speed is typically between 2-4 m/s.

- **Cubic region:** In the cubic region, the power curve approximately follows (2.9).
- **Rated Wind Speed and Plateau:** At the rated wind speed, the wind turbine reaches its maximum or rated power output. The rated capacity is usually reached at wind speeds between 12-16 m/s. The exact value depends on the design of the system. Above the rated speed, the power production is limited, to the rated capacity. In modern wind turbines this is typically done by adjusting the blade pitch angle.
- **Cut-out speed:** The cut-out speed is the wind speed, at which the wind turbine stops operating. The so called cut-out wind speed is typically between 20-25 m/s.

Figure 2.3 illustrates a typical power curve of a pitch regulated wind turbine with a cut-in speed of 3 m/s, a rated speed of 13 m/s and a cut-out speed of 25 m/s.

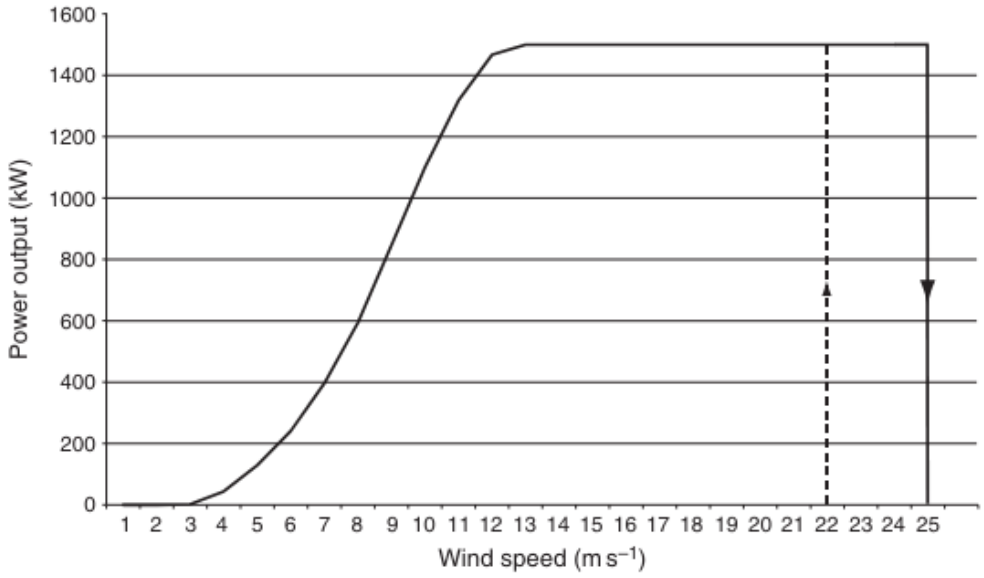


Figure 2.3.: Power curve of a pitch regulated wind turbine with a cut-in speed of 3 m/s, a rated speed of 13 m/s and a cut-out speed of 25 m/s, p. 35 [2].

2.1.5. Classification of WECS

WECS can be categorized based on many different factors. The most common classifications are based on the rotational axis, the turbine, the power control and rotational speed control criteria. In the following sub-chapters the classifications of WECS are discussed. The categorization and content follows that of Chaudhuri et al., p. 4-7 [3].

Rotational Axis

In this classification, a distinction is made between the axis of rotation of the WECS. There are wind turbines with a vertical axis (VAWT) and wind turbines with a horizontal axis (HAWT).

- **VAWT:** In VAWT the rotation axis is perpendicular to the ground. The gearbox and the generator are placed at the base of the tower on the ground, thus they have lower

installation and maintenance costs compared to HAWT. Another advantage is that VAWT do not need to adjust their orientation as they are independent of the wind direction and can operate at low wind speed. On the downside the energy conversion efficiency is lower. VAWTs are preferably used in locations where space is limited, such as urban areas. Typical designs of VAWT are Savonius and Darrieus rotor turbines.

- **HAWT:** HAWTs are the most common type of WECS. They have a higher energy conversion efficiency and can be equipped with different features for power regulation, e.g., pitch control. On the downside the installation and maintenance costs are higher. In HAWT the shaft of the turbine and the generator are placed parallel to the ground at the top of the tower inside the nacelle. The majority of commercial available HAWT's operate with three blades, as it is more economical and less stressful for the structure of the wind turbine.

Turbine

Another way to classify WECS one could use the electrical output of the turbines, p. 5 [3].

- **Low power turbines:** This includes turbines with a maximum output of 30 kW. They are mostly used for remote locations to cover electrical energy demand and to charge batteries.
- **Medium power turbines:** Turbines with an output between 30 kW and 300 kW are classified in this category. They are used for small communities in combination with other forms of electricity production.
- **High power turbines:** This refers to systems in the MW range or large wind farms that generate considerable amounts of electrical energy. They can be used to supply entire cities and industrial plants.

Power Control

The converted wind energy must be controlled appropriately to ensure safe operation and an optimal power production. They can be classified in, p. 5 [3]:

- **Passive stall controller:** Passive stall regulators control the rotor speed of wind turbines by using the natural aerodynamic properties of the blades to limit power output at high wind speeds. As wind speeds increase, the blades are designed to create turbulence and stall, which reduces the lift and slows the rotor. This is the cheapest and simplest control system, as no sensors and actuators are needed.
- **Active stall controller:** The principle is similar to the passive stall control. Here sensors are used to measure the wind speed and actuators change the pitch of the blades to generate turbulence flow and stall. This principle is often used in high power turbines.
- **Pitch controller:** Like the active stall control but with a different effect. In contrast to the previous type, the purpose of the blade angle adjustment is not to achieve a turbulent flow but to reduce the lift and thus the interaction with the wind flow. This system has the advantage of having a faster response time and is therefore widely used in modern wind turbines.

Rotational Speed Control Criteria

WECS can also be classified into fixed speed and variable speed drive train, p. 6 [3]:

- **Fixed speed WECS:** They compose the first set of commercially grid connected WECS. The rotor speed is fixed regardless of the wind speed and only determined by the grid frequency, the gear ratio and the generator design. They use a directly to the grid connected induction generator. Their design allows maximum efficiency only at one specific wind speed. Advantages include simplicity, robustness, reliability, and low electrical component costs. Disadvantages are uncontrollable reactive power consumption, mechanical stress, and limited power quality control. Wind speed fluctuations cause mechanical and electrical power fluctuations, which can lead to voltage variations in weak grids, p. 53-54 [2].
- **Variable speed WECS:** In recent years, variable-speed wind turbines have become the dominant type of installed turbines. These turbines are designed to maximize aerodynamic efficiency across a wide range of wind speeds by continuously adjusting their rotational speed to match the wind speed, maintaining a constant tip speed ratio for to reach the maximum power coefficient. Their electrical systems are more complex, typically involving an induction or synchronous generator connected to the grid via a power converter, which manages speed fluctuations. Advantages of variable-speed turbines include increased energy capture, improved power quality, and reduced mechanical stress. Disadvantages include power electronics losses, more components, and higher equipment costs, p. 54 [2].

2.1.6. Wind Turbine Configuration

As the classification in Chapter 2.1.5 suggests, various configurations for WECS exist. Figure 2.4 illustrates the four standard wind turbine types. The following section provides a brief overview of the structure of type A, type B and type C wind turbines. Type D will be discussed in more detail in Chapter 2.1.7, as only this configuration was modelled as part of the work, to keep the scope of this thesis feasible.

Type A: Fixed Speed

The type A configuration is a fixed speed wind turbine. As can be seen in Figure 2.4, the main components are the rotor, gearbox and a squirrel cage induction generator (SCIG). The SCIG is directly connected to the grid via a transformer. A capacitor bank is typically used for reactive power compensation. With a soft-starter a better grid connection can be achieved. A disadvantage of a type A wind turbine is, that wind fluctuations lead to electrical fluctuations due to the topology. In weak grids, this can cause voltage fluctuations. Both stall control, pitch control and active stall control concepts are popular for this type of wind turbine, p. 57 [2].

Type B: Limited Variable Speed

This configuration consists of a rotor, gearbox, wound rotor induction generator (WRIG), see Figure 2.4. The WRIG is directly connected to the grid via a transformer. A capacitor

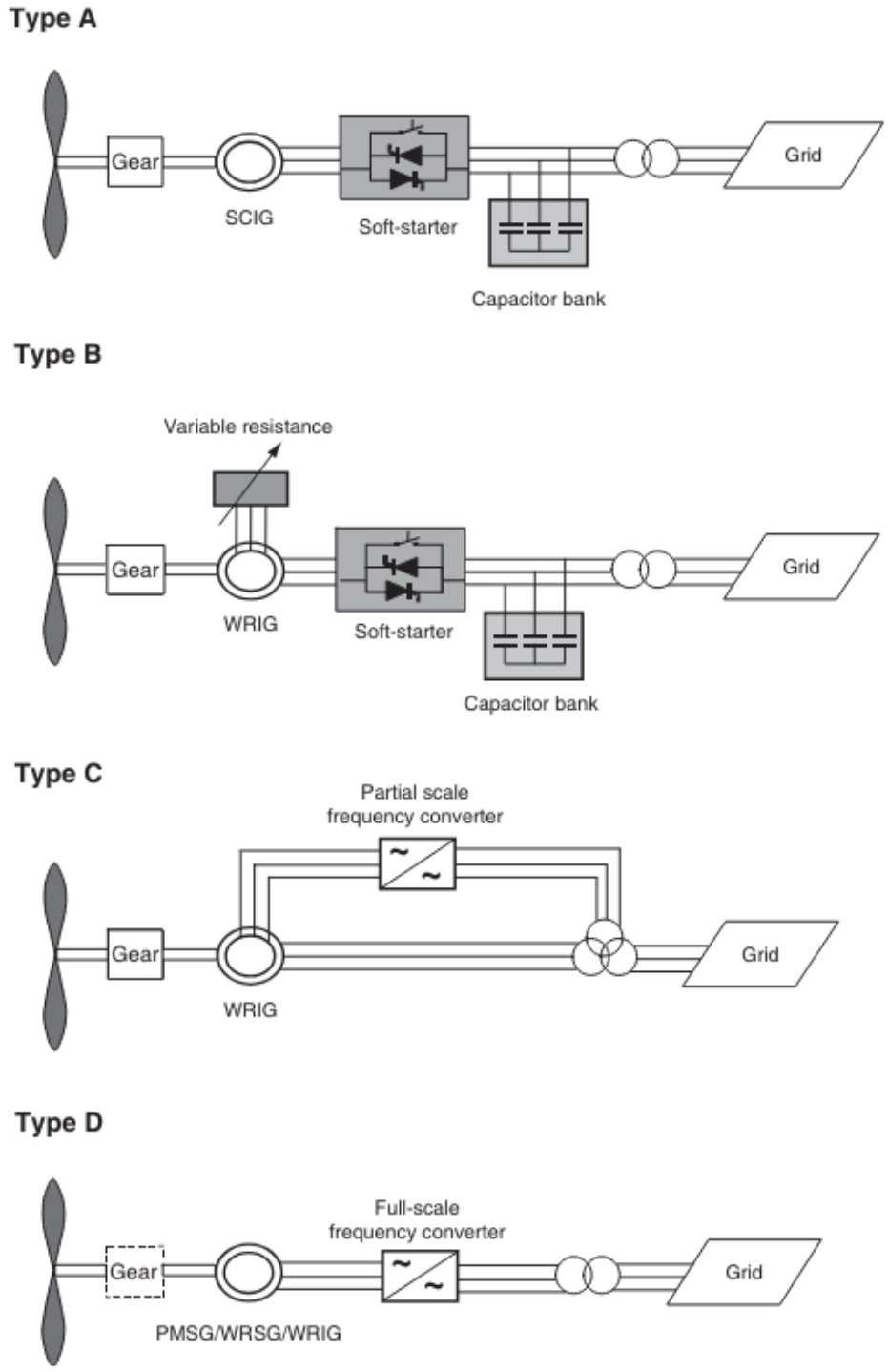


Figure 2.4.: Typical configuration of wind turbines. SCIG: squirrel cage induction generator; WRIG: wound rotor induction generator; PMSG: permanent magnet synchronous generator; WRSG: wound rotor synchronous generator. The dashed line around the gearbox in the Type D wind turbine indicates that it is optional, p. 56 [2].

bank is used to compensate reactive power. A soft-starter allows better grid connection. To allow variable speed, this concept uses an additional variable rotor resistance. The resistance controls the slip, which regulates the power. Typically, 0-10 % above synchronous speed can be achieved, p. 58 [2].

Type C: Variable Speed

The doubly fed induction generator topology allows operation with variable speed. As shown in Figure 2.4, the main components are the rotor, gearbox, WRIG and a partial scale power converter. The stator of the WRIG is directly connected to the grid via a transformer. The rotor of the WRIG is connected to the partial scale power converter, who is responsible for reactive power compensation and smooth grid connection. The converter is typically rated with 30 % of rated generator power, which is an economical advantage. A disadvantage is the use of slip rings which lead to regular maintenance, p. 58-59 [2].

2.1.7. Type D: Variable Speed

The type D wind turbine is a full variable speed wind turbine. It consists of the rotor, the generator and a full scale converter. The generator is connected to the grid via a full scale converter, which consists of a machine side converter (MSC) and a grid side converter (GSC). Figure 2.4 illustrates the configuration of this topology. The converter is responsible for reactive power compensation and smooth grid connection. Either electrically excited generators, such as wound rotor synchronous generator (WRSG) and wound rotor induction generator (WRIG), or permanently magnetically excited generators, such as permanent magnet synchronous generators (PMSG) can be used. An additional gearbox is optional. In case of no gearbox, a multi-pole generator with huge diameter is used, p. 109 [2]. The components in the case of a PMSG are discussed below.

Permanent Magnet Synchronous Generator

PMSG are popular in the field of variable speed wind turbines, as they have a higher efficiency than induction machines because the excitation is provided without any external energy supply. Furthermore PMSG allow operation at a higher power factor and have significantly lower maintenance costs, if the gearbox is absent. On the other hand the needed materials for permanent magnets are expensive and difficult to work with. The magnets are also sensitive to heat and can lose their magnetic properties, e.g., during a fault. Therefore a cooling system must be implemented. Wind turbines with PMSG require a full scale converter to regulate voltage and frequency, which is expensive but allows operation at any wind speed. This type of generator uses a wounded stator and the rotor is equipped with a system of permanent magnets, which are either cylindrical or of salient type. PMSG with low speed are typically equipped with salient type poles and have many poles, p. 70 [2]. The equivalent circuit of a synchronous generator is described in Chapter 2.1.8.

Full Scale Converter

As mentioned previously a wind turbine using a PMSG needs a full scale converter to control its terminal voltage and frequency. Power converters for PMSG wind turbines can be

classified in single-stage AC/AC , two-stage AC/DC + DC/AC and three-stage AC/DC + DC/DC + DC/AC power converters. The power converters are further categorized in low-voltage and medium-voltage, p. 957 [6]. Figure 2.5 illustrates the topology of a three-stage power converter consisting of machine side converter (MSC), DC-link and grid side converter (GSC).

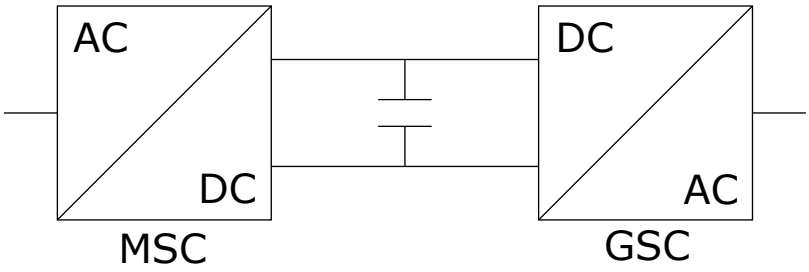


Figure 2.5.: Topology of a three-stage full scale converter, consisting of machine side converter (MSC), DC-link and grid side converter (GSC).

The MSC acts as a rectifier and converts the low frequency AC from the generator to a controlled DC voltage. By converting the PMSG output to DC, the MSC decouples the generator dynamics from the grid, allowing more flexible and stable turbine operation. The MSC plays a vital role in ensuring that the turbine operates at its maximum power point (MPPT), thereby maximizing energy capture from the wind. Various control strategies can be used for the MSC. Common approaches are: minimum stator current control, unity power factor control and the constant stator voltage control, p. 1312 [7]. The active power generated by the PMSG is supplied to the GSC via the DC link. The DC link consists of a capacitor which is charged and discharged by the currents from MSC and GSC. The GSC acts as an inverter and converts DC-link voltage to AC voltage. It controls the DC-link voltage capacitor voltage to a reference value. This ensures the active power supply to the grid. Another control objective, is the reactive power exchange with the grid, to support grid voltage or to achieve the required power factor, which is essential for maintaining stability in the grid, p. 1313 [7].

2.1.8. PMSG Equivalent Circuit

In contrast to the electrically excited synchronous generator, the PMSG is excited with permanent magnets. Because the permanent magnets do not consume electric energy, PMSG achieve higher efficiencies. On the drawback, the excitation and power factor can not be adjusted. Therefore PMSG are always used in combination with converters, which control the generator, p. 617 [8]. Figure 2.6 illustrates the equivalent circuit of an synchronous generator. It consists of an AC voltage source E_g, a serial reactance X_d and a stator resistance R_S.

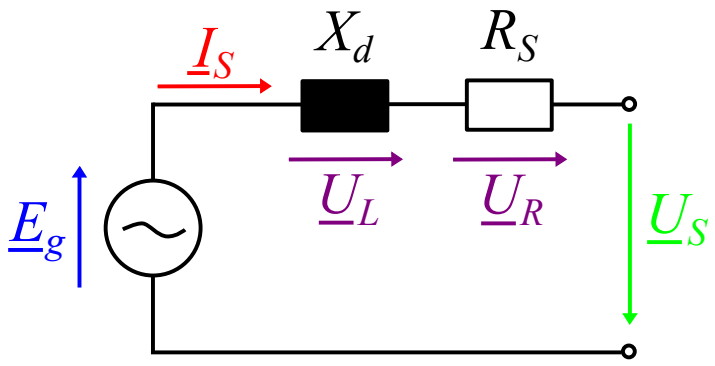


Figure 2.6.: Equivalent circuit of a PMSG, consisting of an AC voltage source E_g, a serial reactance X_d and a stator resistance R_S.

The voltage at the output of the generator can be determined as:

$$\underline{U}_S = \underline{E}_g - \underline{U}_L - \underline{U}_R \quad , \tag{2.10}$$

where

$$\underline{U}_L = jX_d \cdot \underline{I}_S \quad , \tag{2.11}$$

$$\underline{U}_R = R_S \cdot \underline{I}_S \quad . \tag{2.12}$$

The excitation voltage depends on the magnetic flux ψ_{PM} induced in the stator windings and electrical speed ω_e, p. 1311 [7]:

$$\underline{E}_g = \omega_e \cdot \psi_{PM} \quad . \tag{2.13}$$

2.2. Photovoltaic

Photovoltaic (PV) systems convert sunlight directly into electricity using semiconductor materials. As technological advancements continue to drive down costs and improve efficiency, the adoption of PV systems is accelerating globally. In Figure 2.7 the basic concept of power generation is illustrated.

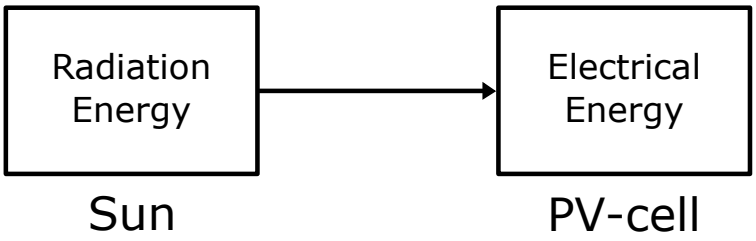


Figure 2.7.: Stages of Solar Energy Conversion.

The following sub-chapters provide a rough overview of the fundamentals of photovoltaic.

2.2.1. Solar Radiation

The energy, released during the nuclear fusion of hydrogen into helium, is emitted into space in the form of radiation. The overall reaction equation is, p. 61 [4]:

$$4\,{}^{1}_{1}p \rightarrow {}^{4}_{2}\,\alpha + 2e^{+} + 2\nu_e + \Delta E \quad . \tag{2.14}$$

The energy released, as a result of the mass defect, per fusion of four protons into one helium nucleus, is approximately $\Delta E = 26.7\ MeV = 4.28 \cdot 10^{-12}\ J$. This results in a total power output of about $3.85 \cdot 10^{26}\ W$, which is emitted in the core of the sun and radiated at the surface, p. 61-62 [4].
With a solar surface area of $6.087 \cdot 10^{12}\ km^2$, the specific radiant power is therefore, p. 62 [4]:

$$\dot{E}_s = \frac{\dot{Q}_s}{A_s} = 63.1\ \frac{MW}{m^2} \quad . \tag{2.15}$$

Solar Constant

The specific radiant power decreases with the square of the distance. With a sun diameter $r_s = 1.39 \cdot 10^6\ km$ and a distance from earth to sun, varying between $1.47 \cdot 10^8\ km$ and $1.52 \cdot 10^8\ km$, a mean value for the extraterrestrial radiation outside the earth's atmosphere can be calculated by, p. 63 [4]:

$$G_{sc} = \left(\frac{r_s}{r_{se}}\right)^2 \cdot \dot{E}_s = 1367\ \frac{W}{m^2} \quad . \tag{2.16}$$

This quantity is called the solar constant. It indicates the radiant power outside the earth's atmosphere. Due to the already mentioned varying distance between sun and earth the true value varies between. The actual value can be determined approximately with, p. 9 [9]:

$$G_{on} = G_{sc}\left(1 + 0.033\cos\left(\frac{360n}{365}\right)\right) \quad . \tag{2.17}$$

Figure 2.8 illustrates the variation of the extraterrestrial radiation over the year. It varies from approximately 1415 W/m^2 on first of January, to 1325 W/m^2 on first of July.

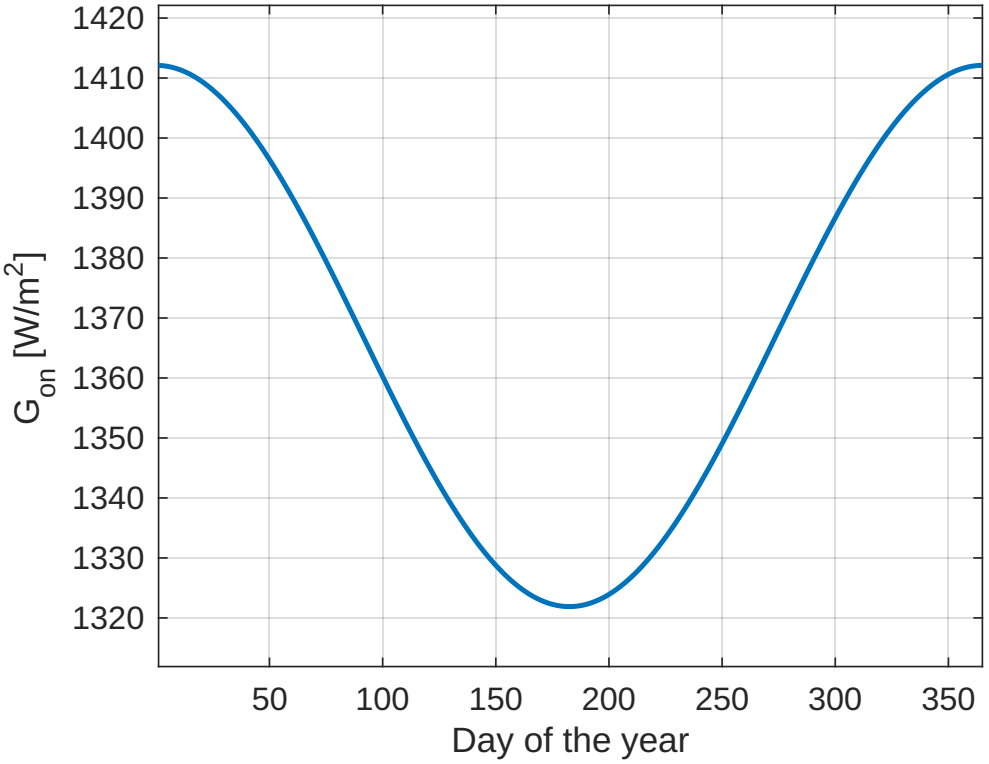


Figure 2.8.: Variation of extraterrestrial radiation over the year.

Air Mass

The spectrum of the emitted sun light changes as it passes through the atmosphere. The underlying phenomena are, p. 38-39 [10]:

- Reflection,
- Absorption,
- Rayleigh-scattering,
- Scattering on aerosol and dust particles.

The air mass index (AM) describes the reduction in irradiance due to the atmosphere. It is defined as the ratio between the mass of atmosphere through which the beam passes, to the mass the light would pass if the sun is at the zenith. For example an AM = 1.5 means, the light needs to pass 1.5-times the way it would as compared to the vertical passage. For a zenith angle θ_z from 0° to 70°, AM can be approximately calculated as, p. 10 [9]:

$$AM = \frac{1}{\cos\theta_z} \quad . \tag{2.18}$$

Global Radiation

Global radiation is the total radiation, incident on a surface. Most measurements of solar radiation refer to the global radiation on a horizontal surface. The global radiation G is the sum of the beam radiation G_b and the diffuse radiation G_d :

$$G = G_b + G_d \quad . \tag{2.19}$$

- **Beam radiation:** This refers to the amount of solar radiation that has not been scattered in the atmosphere.
- **Diffuse radiation:** This refers to the proportion of solar radiation, whose direction was changed due to scattering in the atmosphere.

2.2.2. Sun Position

Determining the position of the sun relative to a reference point is essential for radiation modelling. Therefore accurate models for the sun position are needed in the field of solar energy systems. The next sub-chapters will explain how the position of the sun can be determined in relation to an arbitrarily oriented surface. The model from the book "Solar Engineering of Thermal Processes" by John A. Duffie is used for this purpose [9].

Solar Time

The solar time is used for calculating the position of the sun and and refers to the apparent movement of the sun across the sky, as seen from a specific location on earth, p. 11 [9]. Local time differs from solar time because time zones follow political boundaries. For example, if it is 12 a.m. according to local time, it is possible that the sun has not reached its highest point yet or has already reached it, depending on where the observer is located. Therefore a correction regarding the difference in longitude between the meridian of the time zone and the desired location is needed. Another correction is the equation of time (EoT) which corrects the eccentricity of the earths orbit and the axial inclination of the earth, impacting the moment when the sun passes directly over an observers meridian. The following equation determines the difference between solar time and standard time in minutes, p. 11 [9]:

$$\text{Solar time} - \text{standard time} = 4\,(L_{st} - L_{loc}) + \text{EoT} \quad , \tag{2.20}$$

where L_{st} is the longitude of the time zone and L_{loc} is the observers longitude. The EoT is calculated as, p. 11 [9]:

$$\begin{aligned} \text{EoT} = 229.2\,(&0.000075 + 0.001868\cos(B) - 0.032077\sin(B) \\ &- 0.014615\cos(2B) - 0.04089\sin(2B)) \quad , \end{aligned} \tag{2.21}$$

$$B = (n-1)\,\frac{360}{365} \quad , \tag{2.22}$$

where n is the number of the day. Figure 2.9 illustrates the time correction due to the EoT over the year in minutes.

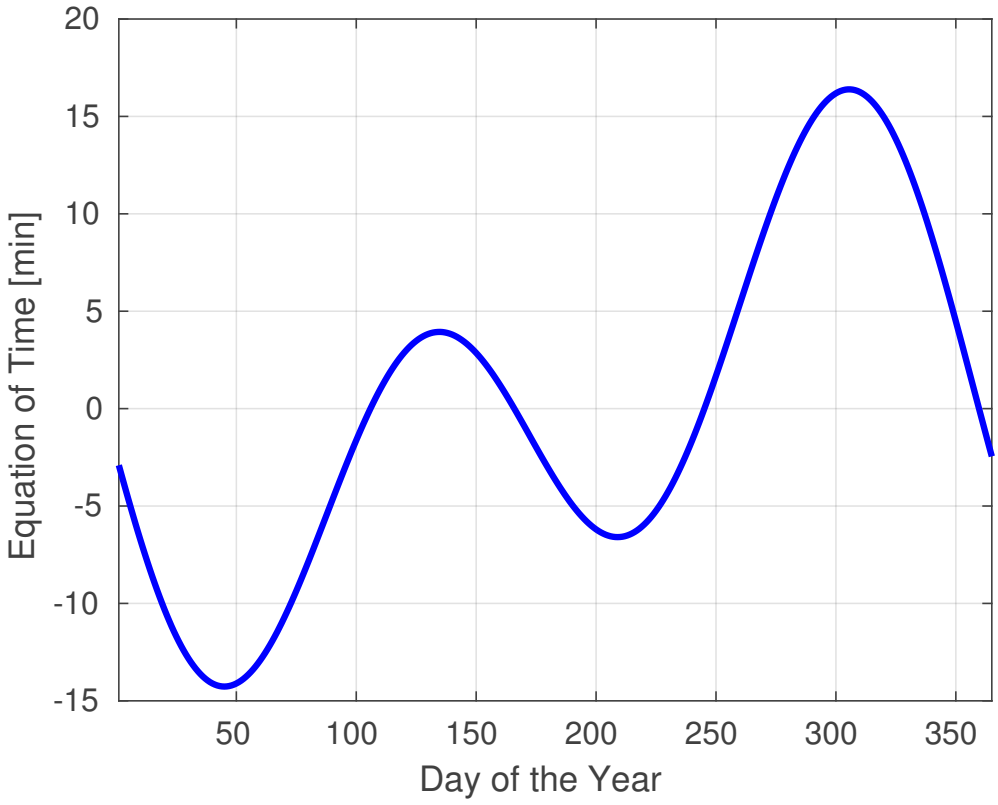


Figure 2.9.: Time correction due to the Equation of Time (EoT) over the year in minutes.

Direction of Beam on Tilted Surface

In order to determine the position of the sun in relation to an inclined surface at any given time, a number of geometric relationships must be introduced. Figure 2.10 illustrates most of the angles. A description of the relevant angles and their range of values is listed below, p. 12-13 [9].

- **Hour angle ω:** The hour angle describes the position of the sun, east or west of the standard meridian of the desired position. The earth rotates 15° per hour around its axis. The sign convention follows: morning negative and afternoon positive, $-180° \leq \omega \leq 180°$.
- **Latitude Φ:** The Latitude describes the displacement north or south of the equator, with north positive and south negative, $-90° \leq \Phi \leq 90°$.
- **Declination δ:** The position of the sun at solar noon, in relation to the equatorial plane, $-23.45° \leq \delta \leq 23.45°$.
- **Slope β:** The contained angle between the tilted surface and the horizontal, $0° \leq \beta \leq 180°$.
- **Surface azimuth angle γ:** Orientation of the projected plane, normal to the surface from the local meridian, where orientation to the south equals zero, west is positive and east is negative. $-180° \leq \delta \leq 180°$.
- **Angle of incidence θ:** Angle between the normal of the surface and the beam.

- **Zenith angle θ_z:** Describes the angle contained between the vertical and the line of the sun.
- **Solar altitude angle α_s:** Describes the angle between the horizontal and the line to the sun.
- **Solar azimuth angle γ_s:** The angular deviation from the south direction of the projection of beam radiation on the horizontal plane, with east of south is negative and west of south is positive.

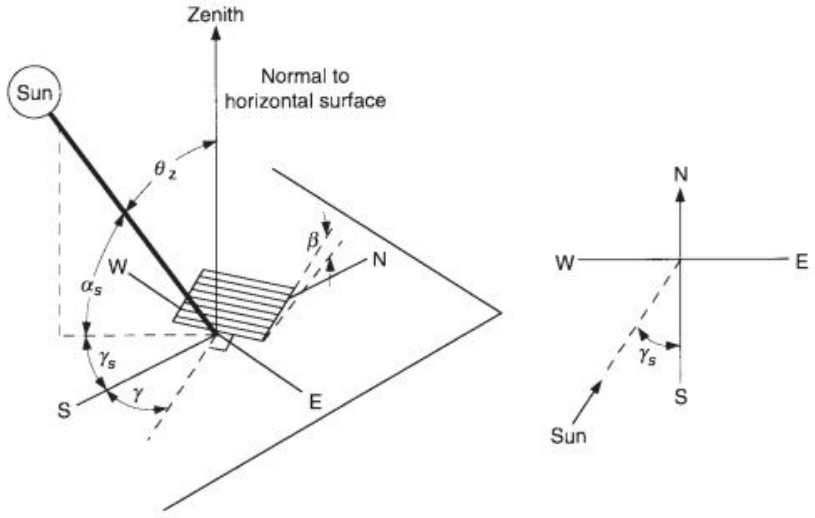


Figure 2.10.: left: Zenith angle θ_z, surface azimuth angle γ, solar azimuth angle γ_s and slope β for an angled surface. right: Plan view of solar azimuth angle γ_s , p. 13 [9].

2.2.3. Working Principle of a Solar Cell

In this sub-chapter, the function of a solar cell will be explained in simplified form using a p-n junction.

Solar cells consist of a p-type and a n-type silicon layer. The n-type layer contains atoms, which have one more electron than silicon in their outer shell, e.g., phosphorus. Silicon has four outer valence electrons and phosphorus has five, so that one electron is not involved in the bond and can move freely in the silicon structure. The p-type layer contains atoms, which have one less electron than silicon in their outer shell, e.g., boron and gallium. Boron has three valence electrons, which means a "hole" remains. If a p-type and a n-type layer are brought together, free electrons diffuse from the n-side to the p-side, where they recombine with the holes. The holes also diffuse towards the n-side, where they recombine with the free electrons. Therefore, there are almost no free electrons and holes near the junction that could fill the charged dopant atoms. The result is an electric field, that tries to repel the free charges from the p-n junction. A new thermal equilibrium is formed, where the diffusion and field current cancel each other out and a depletion zone is created at the p-n junction, p. 68-71 [10].

When photons arrive at the p-n junction, they can be absorbed. The solar cells are designed in a way, which maximizes the probability of absorption in the depletion zone, p. 80 [10]. The absorption of one photon leads to the creation of electron-hole pairs. The now free electrons at the junction are shifted to the n-type side and the holes to the p-type side, due to the electric field. If the n-type layer and the p-type layer are connected to each other, e.g., via a conductor, an electrical current flows, the so-called photocurrent. The phenomena described are shown in Figure 2.11 for a better understanding.

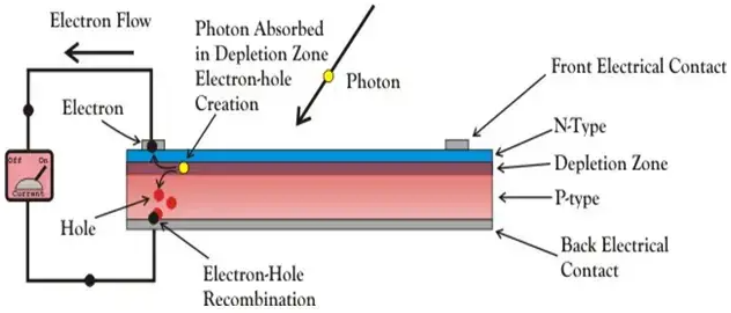


Figure 2.11.: Lighted p-n junction. The electron-hole pairs are separated by the electric field, creating a current when the layers are connected [11].

2.2.4. Equivalent Circuit

To describe real solar cells, it is important to have adequate equivalent circuits for PV cells to predict their performance. The models can be used to determine the electrical output of PV modules and further more whole PV arrays. A wide variety of equivalent circuits can be found in literature. The model approaches differ in the number of parameters, which are required to characterize the PV cell. The more parameters are used, the more accurately the physical phenomena within the PV cell can be described. However, this makes the model more complex and increases the computational effort. The five parameter PV equivalent circuit model represents a good compromise between accuracy and complexity, p. 86 [12].

The five parameter model consists of a current source, a diode and two resistances. The diode and the cell shunt resistance $R_{sh,c}$ are connected in parallel to the current source, whereas the cell serial resistance $R_{s,c}$ is connected in series to the terminal (see Figure 2.12).

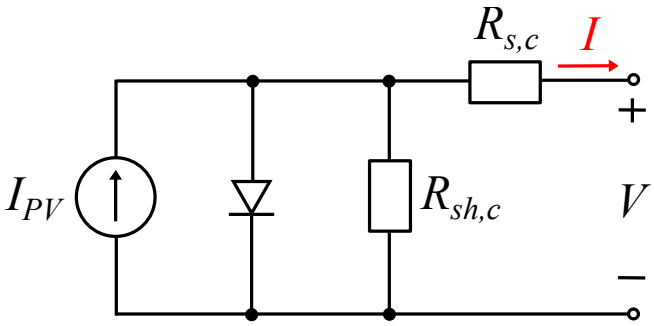


Figure 2.12.: Five parameter equivalent circuit consisting of a current source, a diode and two resistances.

The current-voltage characteristic of the cell is given by, p. 86 [12]:

$$I = I_{PV} - I_0 \cdot \left(e^{\frac{V+R_{s,c}I}{nV_t}} - 1 \right) - \frac{V + R_{s,c}I}{R_{sh,c}} \quad , \tag{2.23}$$

where I_{PV} is the light generated current, I_0 is the reverse saturation current of the p-n junction and n is the diode ideality factor. The serial and shunt resistance highly affect the I-V characteristic of the solar cell. The influence of the shunt resistance is particularly dominant at low cell voltage, whereas the serial resistance is relevant at high cell voltages, p. 86 [12].

2.2.5. Solar Cell Characteristic

The voltage-current characteristic is almost the same as of a photodiode. In solar cells, only the quadrant in which the energy is generated plays a role, p. 92 [10]. Figure 2.13 shows the typical pattern of the characteristic curve and important parameters of the cell.

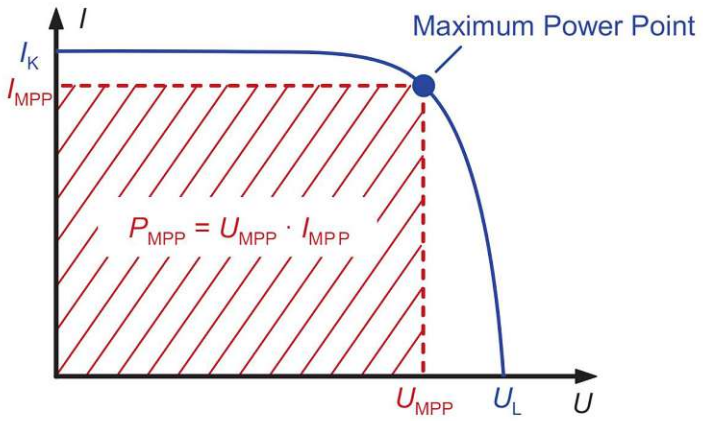


Figure 2.13.: Voltage-current characteristic and important parameters of a solar cell, p. 92 [10].

- **Open circuit voltage U_L:** Is the maximum possible voltage and occurs when the current equals zero. U_L is depending on irradiation and temperature (approx. $-0.4\ \%/K$, p. 95 [10]).
- **Short circuit current I_K:** Is the maximum current and occurs when the voltage is zero. I_K is depending on irradiation and temperature (approx. $+0.06\ \%/K$, p. 95 [10]).
- **Maximum Power Point (MPP):** The solar cells power output depends on the operation point. The MPP is the maximum power that is possible due to the irradiation and temperature. It is defined as the product of I_{MPP} and U_{MPP}.

2.3. Battery Energy Storage System

Battery Energy Storage systems (BESS) are playing an increasingly important role in the growing penetration of the market by renewable energy producers. They have the task to balance the electric grid, by storing the surplus electrical energy as chemical energy when generation is too high, and deliver electrical energy when power production is too low (see Figure 2.14).

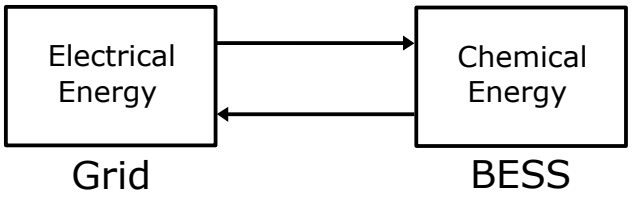


Figure 2.14.: Stages of BESS Energy Conversion.

The following sub-chapters provide a rough overview of the fundamentals of BESS.

2.3.1. Operating Principle of a Battery

In this chapter the basic working principle based on a lithium ion battery is described. A battery consists of two electrodes, called anode and cathode, and an electrolyte. The most common used negative electrode is graphite, p. 96 [13]. For the positive electrode different materials are popular, such as LiFePO4 or $LiCoO_2$, p. 98 [13]. Typically a liquid organic electrolyte is used. Figure 2.15 illustrates the schematic of an lithium ion battery.

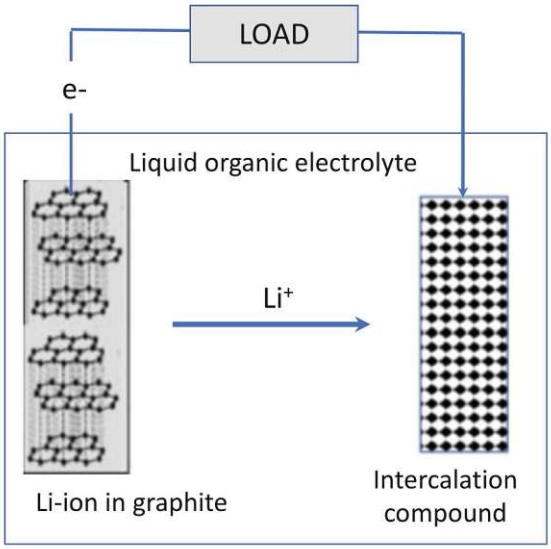


Figure 2.15.: Structure of a lithium ion battery with a graphite anode and a cathode made of intercalate material, p. 97 [13].

The basic principle of a battery is the oxidation and reduction reaction of an electrolyte

with metals. The use of different metals causes a potential difference. During discharge, the negatively charged electrode detaches lithium. The lithium is oxidized to positively charged lithium ions and electrons:

$$Li_xC_6 \rightarrow xLi^+ + xe^- + C_6 \tag{2.24}$$

The lithium ions move through the electrolyte, while the electrons move through the external cycle to the cathode and recombine with the cathode (Reduction half reaction):

$$Li_xMO_2 + xLi^+ + xe^- \rightarrow LiM_{1-x}O_2 \tag{2.25}$$

During charging the mentioned reactions are reversed. An external circuit must provide electrical energy. $LiM_{1-x}O_2$ is oxidized and releases positively charged lithium ions, which move through the electrolyte and reduce on the graphite. The presented reaction equations apply for the use of a general metal oxide cathode, p. 96-97 [13].

2.3.2. Battery Aging

Battery ageing refers to the decline in capacity of a battery over time due to different complex mechanisms. There are two types of battery aging. The first one is calendar ageing and refers to all phenomena, that cause capacity fade over time, irrespective of the usage. The second one is cycle ageing, which corresponds with the utilization of the battery, p. 682 [14].

Calendar Aging

Calendar aging refers to the irreversible loss of battery capacity during storage. It is the degradation that occurs while the battery is not in use. The self-discharge rate can vary significantly depending on storage conditions, which can either accelerate or slow down the processes occurring within the battery. The primary condition affecting calendar aging and self-discharge is the storage temperature. High temperatures favor reactions such as corrosion, leading to lithium loss and capacity fade. Conversely, low temperatures can limit these reactions but may cause issues related to material diffusion and alter the batteries chemistry. Another influence is the state of charge (SOC) level during storage. Batteries do not degrade uniformly at different SOC levels, even at the same temperature. A higher SOC results in higher degradation due to increased potential imbalance at the electrode/electrolyte interface, favoring chemical reactions. Capacity fade increases non linearly over time, p. 682 [14].

Cycle Aging

As mentioned cycle aging occurs if a battery is either charging or discharging. This aging process is influenced by several factors, such as the SOC, how the battery is used and temperature conditions. Many factors affecting calendar aging also play a role in cycle ageing. A battery in use is generally vulnerable to exothermic effects, which are exacerbated by high temperatures and accelerate battery aging. Apart from temperature, the mode of battery utilization significantly impacts the capacity fade. Another crucial factor in lithium-ion battery aging is the charging and discharging voltage throughout its life. High charging voltages accelerate the aging process. Discharge voltage also impacts battery aging through increased impedance. Current peaks also play a role in the cycle aging because significant

current peaks involve high levels of energy being delivered or released from the battery, p. 682-683 [14].

2.3.3. Battery Voltage Behaviour

The battery voltage behaviour is important, in order to effectively manage and utilize batteries. The behaviour of the voltage depends on several factors including the type of battery, the SOC, the load and the environmental conditions, such as temperature. The following overview covers the points listed:

- **Type of battery**: Different types of batteries all have their unique voltage characteristics. A curve that is characteristic for most batteries includes three regions. At the start of discharge the voltage drops, followed by an rather linear region and finally an abrupt drop near the end. Lithium ion based batteries for example maintain a relatively flat voltage profile, usually ranging from 4.2 V when fully charged to 3.0 V when fully discharged, p. 504 [15]. Other batteries normally have a sharper decline as the battery nears full discharge.

- **State of Charge (SOC):** The state of charge significantly impacts the battery voltage. When a battery is fully charged, it exhibits its highest voltage. As the battery discharges, the voltage drops. Monitoring the voltage can give a rough estimate of the SOC, although this method is more accurate for some batteries, p. 24 [13]. For lithium ion batteries, precise SOC estimation often requires other techniques due to their flat discharge curve, p. 24 [13]. Batteries are operated in a way that they do not fall below a certain SOC when discharging, as this can damage the battery. Similarly, there is often an upper SOC to avoid overcharging, p. 23-24 [13].

- **Temperature:** Temperature also plays a crucial role. At lower temperatures, the chemical reactions inside the battery are slower, leading to a higher internal resistance and lower capacity. Conversely, higher temperatures cause higher reaction rates and a higher available capacity, p. 81 [13].

- **Load Effects:** The load or the rate a battery gets discharged highly influences the voltage. Under a high load, the voltage will drop due to internal resistance (I^2R losses), p. 105 [13]. Figure 2.16 illustrates the voltage behaviour for different values of discharging rate C for a specific type of lithium ion battery. A discharge rate of 1 C means that the load will fully discharge the battery in one hour (2 C = 30 min).

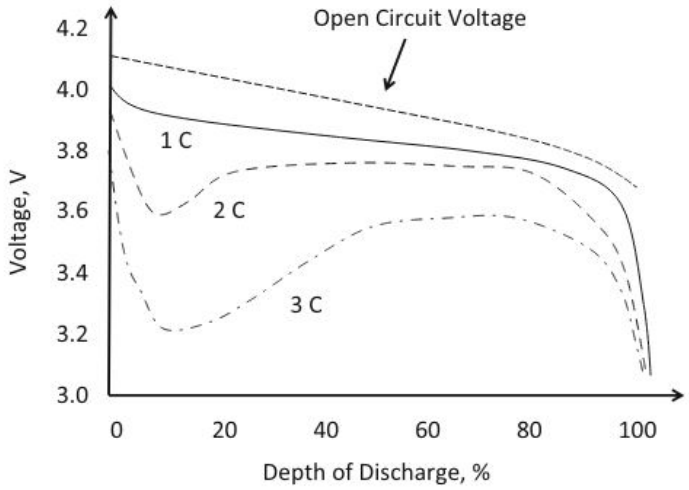


Figure 2.16.: Voltage behaviour of a specific lithium ion battery under different discharging rates C, p. 105 [13].

2.3.4. Equivalent Circuit

To describe real batteries, it is crucial to have adequate equivalent circuits to predict their performance. A wide variety of equivalent circuits can be found in literature. The model approaches differ in the number of parameters, which are required to characterize the battery cell. The more parameters are used, the more accurately the physical and chemical phenomena can be described. However, this makes the model more complex and increases the computational effort. The Rint model is one of the simplest models and consists of a voltage source, that represents the voltage without load, and a serial resistance. With this model rapid dynamics can not be captured but it is sufficient enough for stationary events. Another popular equivalent model is the Thevenin model. It consists of a voltage source, a serial resistance and varying numbers of RC pairs. The RC pairs capture dynamic phenomena within the battery, such as sudden load changes p. 506 [15]. Figure 2.17 illustrates the Thevenin equivalent circuit with one RC pair.

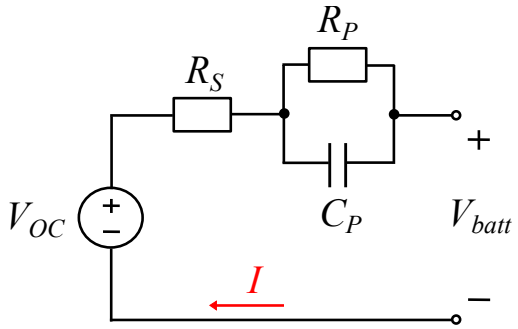


Figure 2.17.: Thevenin equivalent circuit with a serial resistance R_S and one RC pair (R_P and C_P).

All models can be improved if the SOC and temperature dependency of the open circuit voltage (OCV), serial resistance and the RC pairs are included. However, the literature research conducted for this thesis has shown that there are no empirical formulas. Only formulas for specific batteries, based on curve fitting, were found. Most of them ignored the temperature dependency, because the error lies in a certain tolerance and is therefore neglected, p. 507 [15].

3. Proposed Models

This chapter describes the modelling process of the renewable producers, based on the fundamentals described in Chapter 2. The models are implemented in MATLAB. The relevant parameters, mathematical relationships, and assumptions made are explained. Chapter 3.1 gives a brief introduction into BrixGrid, the grid simulation for which the models are implemented. Chapter 3.2 covers the modelling of the variable speed wind turbine, Chapter 3.3 deals with the photovoltaic (PV) modelling and Chapter 3.4 describes the modelling process of the battery energy storage system (BESS). The MATLAB code for the sub-models and combined models can be found in Appendix C.

3.1. Introduction "BrixGrid"

BrixGrid aims to be an Open Source simulation tool for the electrical grid. Only a stationary scenario shall be considered. It shall be modular in its structure to allow a step-by-step developing cycle and enabling multiple people with different expertise to work on the project simultaneously. Additionally, such an approach allows for an iterative improvement.

All electrical calculations will be done in the complex domain; this allows one to avoid solving differential equations. As a back draw, transient simulation is not possible in such a case. For the first version, a symmetrical load is implied and hence a single-phase (SP) equivalent circuit diagram (ECD) used. On a later stage either a $dq0$, $\alpha\beta0$ or a $+-0$ sequence shall be used as this makes it possible to analyse non-symmetrical loads, especially failures in the grid.

The simulation is set up in the following way: real-world elements like power plants, transmission lines, transformers etc. are the so called Brixs. Each Brix contains one or even several nodes that are used to connect two or more Brix. Each node contains several properties, most importantly a complex voltage and current. A node connection is considered as a two-port network (TPN) with its contents depending on what the Brix is representing. The solver now can solve for the unknown currents and voltages by solving the matrix equations of the TPNs.

As an example, consider a set-up where a power plant generates electricity that is consumed by a factory. To model this scenario, we use five Brix: the power plant, one transformer, one transmission cable, another transformer and the factory. The power plant and factory have one node each, while the transmission line and the transformers have two nodes, representing beginning and ending of the cable and transformer primary and secondary side respectively. Figure 3.1 visualizes this set-up.

Calculations are performed in the following way: a time constant is determined, e.g., 1

h. For each instance, a set of equations is solved to determine the power infeed and consumption in that period, losses and the path of the power flow. For the electrical part of the simulation, a linear equation system is produced, similar to the concept of nodal analysis. Each node contains several unknowns: the imaginary and real part of its voltage and the outgoing currents for each adjunct branch. One needs to make sure to generate enough equations to solve said unknowns. Aside from Kirchhoff's law this can include simplifications, valid for a stationary analysis, like assuming that the voltage levels remain constant in the respective hierarchy levels of the grid.

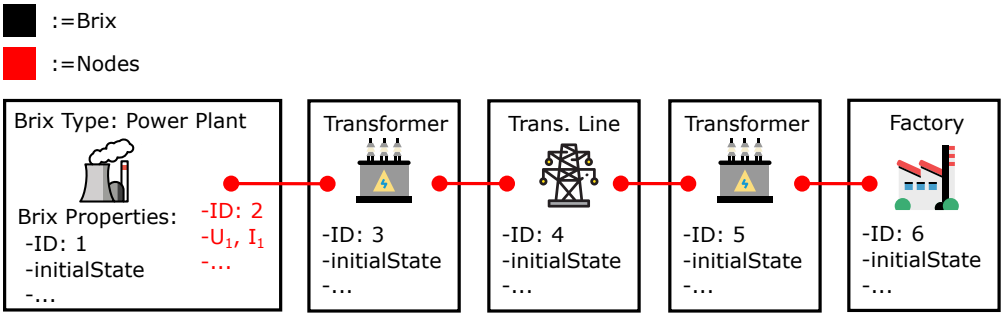


Figure 3.1.: Configuration of a Simulation in BrixGrid. The main building blocks are the Brix, each containing one or several nodes. Nodes are used by the solver. For a better overview, only one node's specification are written out.

3.2. Variable Speed Wind Turbine

The Wind Turbine Brix consists of the four sub-models: rotor, permanent magnet synchronous generator (PMSG), machine side converter (MSC) and grid side converter (GSC). The following sub-chapters describe the modelling process of the sub-models. At the end the overall wind turbine model is presented.

3.2.1. Rotor

The sub-model rotor has the task of determining the mechanical energy, which is extracted from the kinetic energy of the wind. Table 3.1 shows the input parameters that define the rotor model. The input parameters are blade radius r, cut-in wind speed v_{cut-in}, the rated wind speed v_{rated}, the cut-out wind speed $v_{cut-out}$ and the maximal rotational speed n_{max}, and can be taken from data sheets of the corresponding real-life wind turbine. The other two input parameters are the air density ρ and a wind speed time series v_w The output parameters are the rated power P_{rated}, the mechanical power P_{mech} and the angular velocity ω, see Table 3.1. These are later used as input parameters in the PMSG sub-model. The following sections describe the mathematical description of the rotor.

Input Parameters		
Symbol	**Name**	**Unit**
v_w	Wind speed	m/s
ρ	Air density	kg/m^3
r	Blade radius	m
v_{cut-in}	Cut-in wind speed	m/s
v_{rated}	Rated wind speed	m/s
$v_{cut-out}$	Cut-out wind speed	m/s
n_{max}	Maximal rotational speed	rpm
Output Parameters		
Symbol	**Name**	**Unit**
P_{mech}	Mechanical power	W
P_{rated}	Rated power	W
ω	Angular velocity	rad/s

Table 3.1.: Input and output parameters of the rotor.

Mathematical Model

As described in Chapter 2.1.4 the mechanical power extracted by the rotor is determined as:

$$P_{mech} = \frac{1}{2}\rho\pi r^2 c_p(\beta, \lambda) v_w^3 \quad , \tag{3.1}$$

where c_p is the power coefficient. c_p is a function of the blade pitch angle β and the tip speed ratio λ. A precise description of c_p for a specific rotor blade is essential for the accuracy of the simulation. However, the literature research carried out as part of this work revealed that the manufacturers do not publish these. Therefore individual c_p-curves cannot be reproduced and general approximations have to be used. Typical wind turbines have an maximum c_p between 0.4 and 0.5, p. 166 [4]. For this reason, a curve for c_p is selected in which the maximum c_p lies approximately in the middle of this range, in order to ensure a satisfactory description for various wind turbines. The following c_p formula is taken from p. 1842 [5] and reaches its maximum for $\beta = 0°$ at approximately 0.44 and is therefore perceived as a satisfactory compromise:

$$c_p = 0.22 \left(\frac{116}{\lambda_i} - 0.4\beta - 5 \right) \exp(-\frac{12.5}{\lambda_i}) \quad , \tag{3.2}$$

where

$$\frac{1}{\lambda_i} = \frac{1}{\lambda + 0.08\beta} - \frac{0.035}{\beta^3 + 1} \quad , \tag{3.3}$$

$$\lambda = \frac{\omega r}{v_w} \quad . \tag{3.4}$$

Figure 2.2 illustrates 3.2 for different values of β.

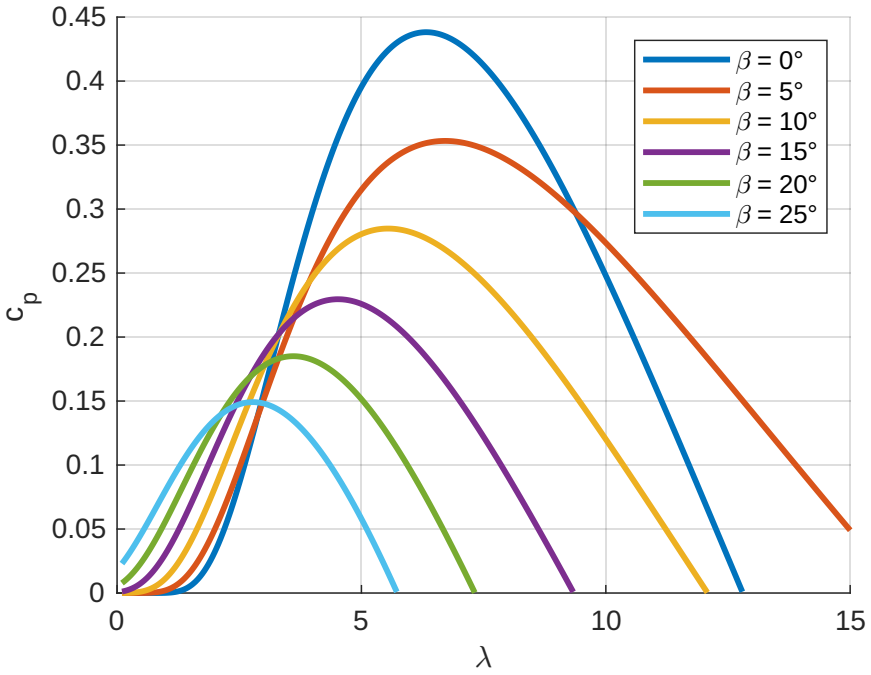


Figure 3.2.: Power coefficient c_p vs tip speed ratio λ for different blade pitch angles β

The rated power P_{rated} of the wind turbine is calculated as:

$$P_{rated} = \frac{1}{2}\rho\pi r^2 c_{p,max}(\beta,\lambda)v_{rated}^3 \quad , \tag{3.5}$$

where $c_{p,max}$ is the maximum power coefficient.

Maximal Power Coefficient Tracking

Manufacturers offer different technologies to track the maximum power coefficient in the cubic area of the power curve, p. 3 [16]. One of these techniques is the tip-speed-ratio (TSR) control method and was chosen for this thesis because it can be easily implemented. The aim of the TSR control method is to adjust the TSR of the wind turbine in order to maximize c_p. This is done by adjusting the angular velocity ω of the shaft, p. 895 [17].

The TSR control, as implemented in this thesis, can be mathematically formulated as an optimization problem:

$$max \quad c_p = 0.22\left(\frac{116}{\lambda_i} - 0.4\beta - 5\right)\cdot\exp(-\frac{12.5}{\lambda_i}) \tag{3.6}$$

subject to (3.3), (3.4) and:

$$0 \leq \omega \leq 2\pi\cdot\frac{n_{max}}{60} \quad , \tag{3.7}$$

$$\beta = 0 \quad . \tag{3.8}$$

For an implementation of the formulated problem, the function $fmincon()$ from the MATLAB optimization toolbox is used [18]. $fmincon()$ is used for solving constrained non-linear optimization multi variable problems, p. 1155 [18].

Power Limiting

If the wind speed v_w exceeds the rated wind speed v_{rated}, the power must be limited to not exceed P_{rated} to avoid damage in the wind turbine components, e.g., to minimize fatigue of the shaft or rotor blades, p. 370 [19]. This is done by varying the blade pitch angle β. The pitching control, as implemented in this thesis, can be mathematically formulated as an optimization problem. The power limiting is reached by minimizing the difference between the rated power P_{rated} and the actual power:

$$min\left(P_{rated} - \frac{1}{2}\rho\pi r^2 v_w^3 c_p\right) \tag{3.9}$$

subject to (3.2), (3.3), (3.4) and:

$$0 \leq \beta \leq 90° \quad , \tag{3.10}$$

$$\omega = 2\pi\cdot\frac{n_{max}}{60} \quad . \tag{3.11}$$

For an implementation of the formulated problem, the function $fsolve()$ from the MATLAB optimization toolbox is used [18]. $fsolve()$ solves a system of equations by minimizing the sum of squares of the components, p. 976 [18]. $fsolve()$ does not use any constraints but is chosen because $fmincon()$ leads to deviations from P_{rated}. However, the tests carried out have shown that the pitch angle β remains within the limits, as specified in (3.10).

Rotor Model

The calculation of the final power output of the rotor depends on the current wind speed. A distinction between four cases can be made:

1) $\boldsymbol{v_w < v_{cut-in}}$:
The wind turbine does not operate below the cut-in speed.
   - $P_{mech} = 0$
   - $\omega = 0$

2) $\boldsymbol{v_w \geq v_{cut-in}}$ & $\boldsymbol{v_w \leq v_{rated}}$:
In this region the TSR control is active and maximizes the power output.
   - P_{mech} and ω with (3.6)

3) $\boldsymbol{v_w > v_{rated}}$ & $\boldsymbol{v_w \leq v_{cut-out}}$:
In this region the pitching control is active and limits the power to P_{rated}.
   - P_{mech} with (3.9)
   - $\omega = 2\pi \cdot n_{max}/60$

4) $\boldsymbol{v_w \geq v_{cut-out}}$:
The rotor stands still, to avoid the risk of damage in the components of the wind turbine.
   - $P_{mech} = 0$
   - $\omega = 0$

3.2.2. PMSG

The second sub-model of the wind turbine is the permanent magnet synchronous generator (PMSG). The PMSG converts the mechanical power P_{mech} from the rotor, into electric AC power. P_{mech}, P_{rated} and ω_e are input parameters from the rotor. The other input parameters must be extracted from scientific papers, as the manufacturers typically do not publish them. The input parameters are listed in Table 3.2. The output parameters are the active power P_s and the reactive power Q_s, see Table 3.2. The following section describes the calculation of the output quantities.

Input Parameters		
Symbol	**Name**	**Unit**
P_{mech}	Mechanical power	W
P_{rated}	Rated power	W
ω	Angular velocity	rad/s
V_{rated}	Rated stator voltage	V
L_{ds}	Direct stator inductance	$p.u.$
L_{qs}	Quadrature stator inductance	$p.u.$
ψ_{pm}	magnetic flux	$p.u.$
p	Number of pole pairs	–
η	Generator efficiency	–
Output Parameters		
Symbol	**Name**	**Unit**
P_s	Active power	$p.u.$
Q_s	Reactive power	$p.u.$

Table 3.2.: Input and output parameters of the PMSG.

Electrical Model

The mathematical relationships are taken from Fernandez et al. [7]. Several assumptions were made in the PMSG model from Fernandez et al., e.g., neglect of magnetic saturation, sinusoidal flux distribution, only copper losses and sinusoidal stator voltage and current, p. 1310 [7]. Figure 3.3 illustrates the equivalent circuit of the PMSG. The calculation is carried out in the per unit system (p.u.).

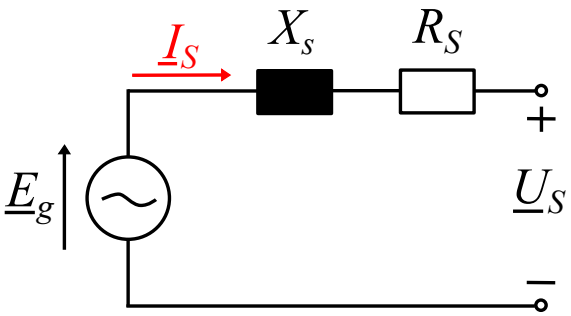


Figure 3.3.: Equivalent curcuit of the PMSG.

The mechanical torque is determined as:

$$T_W = P_{mech} \cdot \omega \quad , \tag{3.12}$$

where P_{mech} is the mechanical power from the rotor and ω is the angular velocity of the shaft. The relationship between T_W and the electrical torque T_E is:

$$T_W - T_E = 2H\frac{d\omega_e}{dt} \quad , \tag{3.13}$$

where $\omega_e = p \cdot \omega$ is the electrical speed, p is the number of pole pairs of the generator. Because the simulation uses time-steps of 1 h and the wind speed is assumed to be constant, (3.13) is simplified to:

$$T_E = T_W \quad . \tag{3.14}$$

The model is described in a dq-reference frame, where the d-axis is aligned to the permanent magnet flux position, p. 1311 [7]. In the steady-state condition the voltage in the d- and q-axis are defined as:

$$u_{ds} = -R_s i_{ds} - \omega_e L_{qs} i_{qs} \quad , \tag{3.15}$$

$$u_{qs} = \omega_e L_{ds} i_{ds} - R_s i_{qs} + \psi_{pm} \quad , \tag{3.16}$$

where R_s is the stator resistance, L_{qs} and L_{ds} are the inductance components, ψ_{pm} is the flux induced by the permanent magnets in the stator phases and i_{ds} and i_{qs} are the current components. The stator voltage U_s is therefore:

$$U_s = \sqrt{(u_{ds})^2 + (u_{qs})^2} = \sqrt{(-R_s i_{ds} - \omega_e L_{qs} i_{qs})^2 + (-R_s i_{qs} + \omega_e L_{ds} i_{ds} + \omega_e \psi_{pm})^2} \quad . \tag{3.17}$$

The electrical torque is given by:

$$T_E = \omega_e \psi_{pm} i_{qs} \quad , \tag{3.18}$$

under the assumption that L_{ds}=L_{qs}, which is given for a PMSG, that uses a surface magnet generator that exhibits negligible saliency, p. 1311 [7].

In order to determine active and reactive power the equation system has to be solved for u_{ds}, u_{qs}, i_{ds} and i_{qs}. The quadrature current i_{qs} can be obtained by transforming (3.18):

$$i_{qs} = \frac{T_E}{\omega_e \psi_{pm}} \quad . \tag{3.19}$$

The equations of the machine side converter (MSC) PI controller presented by Fernandez et al., see p. 1313 [7], are used to solve for u_{ds} and u_{qs}. The MSC controller works in a way that cancels the cross-coupling effect in (3.17). It is assumed in this thesis, that the MSC controller is perfectly regulated, so that:

$$u_{ds} = -\omega_e L_{qs} i_{qs} \quad . \tag{3.20}$$

With the constant stator voltage control, the stator voltage U_s is kept to a fixed value. Therefore the quadratic voltage u_{qs} can be determined as:

$$u_{qs} = \sqrt{(U_s)^2 - (u_{ds})^2} \quad . \tag{3.21}$$

Finally the direct component of the stator current i_{ds} is determined as:

$$i_{ds} = \frac{u_{qs} - \omega_e \psi_{pm}}{\omega_e L_{ds}} \quad . \tag{3.22}$$

The stator active power P_s is calculated as:

$$P_s = \eta \cdot (u_{ds} i_{ds} + u_{qs} i_{qs}) \quad , \tag{3.23}$$

The stator reactive power Q_s is calculated as:

$$Q_s = \eta \cdot (u_{qs} i_{ds} - u_{ds} i_{qs}) \quad , \tag{3.24}$$

where η is the generator efficiency. η is introduced to simulate losses in the conversion process, because the stator resistance is eliminated in the controller equations, see p. 1313 [7], to solve the equation system.

3.2.3. Machine Side Converter

The machine side converter (MSC) acts as a rectifier. The MSC has the task to regulate the PMSG output AC voltage U_s and to convert the AC to DC. Through the DC-link the active power from the PMSG is supplied to the grid side converter (GSC), p. 1311 [7]. The DC voltage U_{DC} is stabilized by a DC link capacitor. It is assumed that the voltage is perfectly balanced at any time and therefore constant. In the following modelling process, only the transfer of the active power between the PMSG and the grid is considered because the transfer of the reactive power requires knowledge of the controller system. Figure 3.4 illustrates the MSC model.

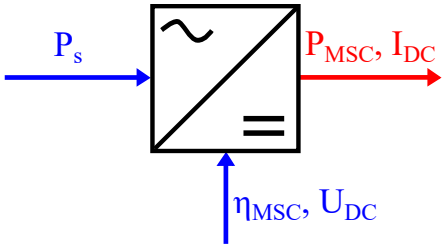


Figure 3.4.: Model of the MSC.

The MSC transfers the active power from the PMSG to the GSC. The resulting losses during the conversion to DC are taken into account by the efficiency η_{MSC}. The power on the DC side is therefore calculated as:

$$P_{MSC} = \eta_{MSC} \cdot P_s \quad , \tag{3.25}$$

where P_s is the active power from the PMSG. The current in the DC-link is determined by:

$$I_{DC} = \frac{P_{MSC}}{U_{DC}} \quad , \tag{3.26}$$

where U_{DC} is the constant voltage in the DC-link, that is controlled by the GSC. Table 3.3 summarizes the input and output parameters of the MSC model.

Input Parameters		
Symbol	**Name**	**Unit**
P_s	Active power	W
η_{MSC}	MSC efficiency	–
U_{DC}	DC-link voltage	V
Output Parameters		
Symbol	**Name**	**Unit**
P_{MSC}	Active power	W
I_{DC}	DC-link current	A

Table 3.3.: Input and output parameters of the MSC.

3.2.4. Grid Side Converter

The grid side converter (GSC) acts as an inverter. The GSC has the task to convert the DC voltage from the DC-link into 50 Hz AC voltage. Furthermore the GSC has to maintain the DC-link voltage to the set level, p. 1313 [7]. As mentioned before, only the active power transfer is considered. Therefore only the real component of the grid current can be estimated. Figure 3.5 illustrates the GSC model.

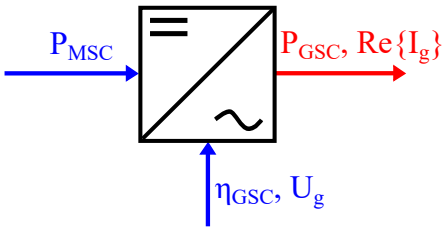


Figure 3.5.: Model of the GSC.

The resulting losses during the conversion to AC are taken into account by the eff. factor η_{GSC}. The active power injected to the grid is therefore:

$$P_{GSC} = \eta_{GSC} \cdot P_{MSC} \quad , \tag{3.27}$$

where P_{MSC} is the power output of the MSC. Power conversion gives:

$$P_{GSC} = Re\left\{3\underline{U}_g\underline{I}_g^*\right\} = 3 \cdot \left(Re\left\{\underline{U}_g\right\} \cdot Re\left\{\underline{I}_g\right\} + Im\left\{\underline{I}_g\right\} \cdot Im\left\{\underline{U}_g\right\}\right) \quad , \tag{3.28}$$

where $\underline{U}_g$ is the grid voltage and $\underline{I}_g$ is the grid phase current. The grid voltage is used as the reference point, therefore (3.28) is simplified to, p. 1312 [7]:

$$P_{GSC} = 3 \cdot \left(Re\left\{\underline{U}_g\right\} \cdot Re\left\{\underline{I}_g\right\}\right) \quad . \tag{3.29}$$

Thus the real component of the phase current is determined as:

$$Re\left\{\underline{I}_g\right\} = \frac{P_{GSC}}{3 \cdot Re\left\{\underline{U}_g\right\}} \quad . \tag{3.30}$$

Table 3.4 summarizes the input and output parameters of the GSC model.

Input Parameters		
Symbol	**Name**	**Unit**
P_{MSC}	Active power	W
η_{GSC}	GSC efficiency	–
$\underline{U}_g$	AC grid voltage	V
Output Parameters		
Symbol	**Name**	**Unit**
P_{GSC}	Active power	W
$\underline{I}_g$	Real current	A

Table 3.4.: Input and output parameters of the GSC.

3.2.5. Combined Wind Turbine Model

The overall Wind Turbine Brix is formed by connecting the rotor, PMSG, MSC and GSC. Figure 3.6 illustrates the overall model. The overall model has no input parameters because the wind speed data is included in the rotor sub-model. The final outputs of the Wind Turbine Brix are the active power P and the real component of the grid current $\underline{I}_g$, which are determined by (3.29) and (3.30). The output parameters are shown in Table 3.5.

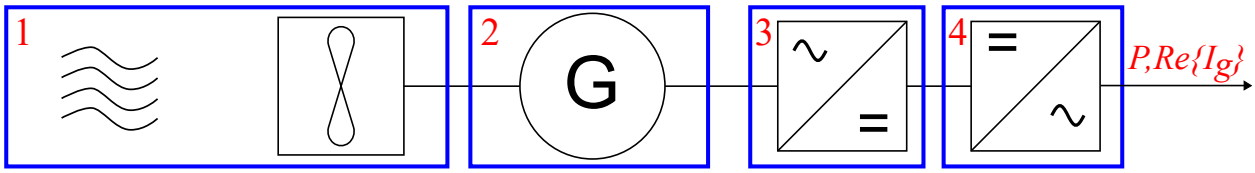


Figure 3.6.: Overall Wind Turbine model: **1.** Rotor; **2.** PMSG; **3.** MSC; **4.** GSC.

Output Parameters		
Symbol	**Name**	**Unit**
P	Active power	W
$\underline{I}_g$	Real current	A

Table 3.5.: Output parameters of the Wind Turbine model.

3.3. Photovoltaic

The PV Brix consists of two sub-models. The first one is the PV array, which consists of multiple PV modules. The second sub-model is the inverter which converts the DC from the PV array into AC to feed into the grid. The following sub-chapters, describe the modelling process of the PV array and the inverter. At the end the overall PV model is presented.

3.3.1. PV Array

The sub-model PV array essentially consists of two parts. The first part is the determination of the irradiation on the PV modules. The second part is the estimation of the PV modules IV-characteristic, to determine the maximum power point in order to calculate the modules power output. As this is the most complex model in this work, only an overview of the input and output parameters will be given here. The detailed description of the modelling process and assumptions is divided into the two following sub-chapters 3.3.2 "Estimating Irradiance on PV Module" and 3.3.3 "PV module" to provide a clearer explanation.

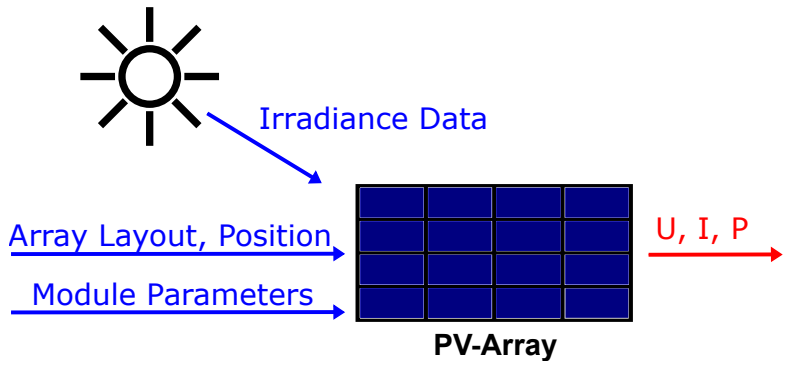


Figure 3.7.: Overview of the input and output parameters of the PV array. The input parameters are classified in three groups: **1)** Irradiance data: consisting of air temperature, ground albedo, time and date in coordinated universal time (UTC) and the irradiation on a horizontal surface. **2)** Array layout and position parameters **3)** Module parameters. Output parameters are direct voltage, direct current and power.

The input parameters for the PV array model can be divided into three main groups, as can be seen in Figure 3.7. The irradiance data, which includes the air temperature, the reflection coefficient of the ground (ground albedo), the time and date in coordinated universal time (UTC) and the irradiation on a horizontal surface. These parameters can be extracted from the free to use solar database Solcast [20]. The next important group of input parameters includes the array layout and the geographical position of the PV array. The last group of input parameters contains data sheet values, to define the module characteristic curve, these values are found on every module data sheet. The three input parameter groups and their quantities are listed in Table 3.6. The output variables for the PV array model are direct voltage, direct current and power, see Table 3.6. These variables in turn serve as input variables for the inverter model.

Input Parameters		
Irradiance Input Parameters		
Symbol	**Name**	**Unit**
GHI	Global irradiance on horizontal surface	W/m^2
BHI	Beam irradiance on horizontal surface	W/m^2
ρ	Ground albedo	-
T_a	Air temperature	°C
timestamp	year, month, day and clocktime	-
Layout and position parameters		
Symbol	**Name**	**Unit**
β	Slope of modules	°
γ	Surface azimuth of modules	°
L_{loc}	Longitude	°
L_{st}	Standard meridian	°
ϕ	Latitude	°
N_s	Modules in series	-
N_p	Modules in parallel	-
PV module parameters		
Symbol	**Name**	**Unit**
I_{ompp}	Maximum power current at STC	A
V_{ompp}	Maximum power voltage at STC	V
V_{ooc}	Open circuit voltage at STC	V
I_{osc}	Short circuit current at STC	A
α_{isc}	Temperature coefficient of I_{osc}	% / °C
α_{voc}	Temperature coefficient of V_{oc}	% / °C
$NOCT$	Nominal operating temperature	°C
Output Parameters		
Symbol	**Name**	**Unit**
I	Direct current	A
U	Direct voltage	V
P	Power	W

Table 3.6.: Input and output parameters for the PV array.

3.3.2. Estimating Irradiance on PV Module

To calculate the power output of a PV module, one has to determine the global irradiance on the module surface first. Irradiance data for randomly oriented surfaces at arbitrary locations are not available in databases and must be estimated from horizontal irradiance data. A few steps are required to apply this data to an arbitrarily oriented PV module:

- determine the solar time,
- determine the sun position and
- determine the irradiance on the tilted surface.

Solar Time

The solar time is used to calculate the "actual time" in order to determine the exact position of the sun. The input clock time must be entered in coordinated universal time (UTC), as the used corrections do not take into account the difference in summer and winter time. The time correction is calculated as described in Chapter 2.2.2.

Sun Position

The solar time can now be used to determine the exact position of the sun in relation to the PV module. Therefore, the angles introduced in Chapter 2.2.2 have to be calculated. The formulas are taken from the book "Solar Engineering of Thermal Processes" by John A. Duffie [9]. For visualization see Figure 2.10.

The earth rotates 15° per hour around its axis, see p. 13 [9]. The hour angle ω can be determined by:

$$\omega = \left(-180 + 15h + \frac{1}{4}min + \frac{1}{240}sec\right) \cdot \frac{\pi}{180} \quad , \tag{3.31}$$

where the values for h, min and sec are extracted from the previously determined solar time. The declination δ is approximately given by:

$$\delta = 23.45 \cdot \sin\left(360 \cdot \frac{284 + n}{365}\right) \quad . \tag{3.32}$$

The angle of incident θ can now be calculated using the hour angle ω, the declination δ, the slope of the PV modules β and the latitude of the desired position ϕ:

$$\begin{aligned} \cos\theta = \sin\delta\sin\Phi\cos\beta - \sin\delta\cos\Phi\sin\beta\cos\gamma \\ + \cos\delta\cos\Phi\cos\beta\cos\omega + \cos\delta\sin\Phi\sin\beta\cos\gamma\cos\omega \\ + \cos\delta\sin\beta\sin\gamma\sin\omega \quad . \end{aligned} \tag{3.33}$$

Next the zenith angle θ_z is calculated using:

$$\cos\theta_z = \cos\Phi\cos\delta\cos\omega + \sin\Phi\sin\delta \quad , \tag{3.34}$$

with whom the solar altitude angle α_s is estimated as:

$$\alpha_s = \frac{\pi}{2} - \theta_z \quad . \tag{3.35}$$

The last important angle is the solar azimuth angle γ_s.

$$\gamma_s = \text{sgn}(\omega) \left| \arccos \left(\frac{\cos\theta_z \sin\Phi - \sin\delta}{\sin\theta_z \cos\Phi} \right) \right| \quad , \tag{3.36}$$

where sgn is the signum function. With this set of angles, the position of the sun relatively to the PV module, is determined for any time.

Irradiance on Tilted Surface

Many different approaches for the estimation of the global irradiance on a tilted surface exist in the literature. From the different models the approach from Muneer et al. [21] was chosen, which proved to have the best overall performance as suggested by Gracia et al., p. 1 [22].

The global irradiance G_T $[W/m^2]$ on a tilted surface is the sum of the beam irradiance G_{bT} $[W/m^2]$, the diffuse irradiance G_{dT} $[W/m^2]$ and the irradiance received from ground reflections G_{rT} $[W/m^2]$:

$$G_T = G_{bT} + G_{dT} + G_{rT} \quad . \tag{3.37}$$

Since the beam irradiance is directly coming from the sun without scattering, G_{bT} can be calculated purely on geometric dependencies:

$$G_{bT} = G_b \cdot \frac{\cos\theta}{\cos\theta_z} \quad , \tag{3.38}$$

where G_b is the beam irradiance on a horizontal surface, θ is the angle of incident of the sun and θ_z is the solar zenith angle of the sun.

The reflected irradiance G_{rT} is assumed as purely isotropic, p. 1 [22], which means that the beam and diffuse irradiation are reflected identically and the horizon is free from barriers and uniform. In this approach, the reflected irradiance depends solely on the fraction of the incident sunlight that is reflected by the surface, also known as ground albedo ρ $[-]$, p. 51 [10], the global irradiance on the horizontal surface G and the slope of the PV modules. Therefore the reflected irradiance is determined as:

$$G_{rT} = \rho \cdot G \cdot \frac{1 - \cos\beta}{2} \quad . \tag{3.39}$$

As Gravia et al. stated, most models calculate the beam and reflected irradiance on the inclined surface as shown in the previous equations, p. 2 [22]. However, for the diffuse component different approaches exist. The diffuse irradiance is the share of global irradiance that has been scattered in the atmosphere. The scattering processes within the atmosphere are very complex and can therefore only be calculated as a rough approximation. Isotropic models are therefore associated with major errors and anisotropic approaches are favoured. Muneer et al. [21] uses an approach which considers two components for the diffuse irradiance, p. 4-5 [22]. The two components differentiate between clear and overcast conditions.

For overcast weather following equation is used:

$$G_{dT} = G_d \cdot \left[\left(\frac{1 + \cos\beta}{2} \right) + 0.25227 \left(\sin\beta - \beta\cos\beta - \pi \left(\sin\frac{\beta}{2} \right)^2 \right) \right] \quad . \tag{3.40}$$

In clear weather, a distinction is made depending on the solar elevation angle α_s. If $\alpha_s < 0.1$ rad then:

$$G_{dT} = G_d \cdot \left[\left(\left(\frac{1+\cos\beta}{2} \right) + \left(\sin\beta - \beta\cos\beta - \pi \left(\sin\frac{\beta}{2} \right)^2 \right) \right. \right. \\ \cdot \left(0.00263 - 0.712\frac{G_b}{G_{on}} - 0.6883\left(\frac{G_b}{G_{on}}\right)^2 \right) \Bigg) \\ \left. \left. \cdot \left(1 - \frac{G_b}{G_{on}} \right) + \left(\frac{G_b}{G_{on}} \cdot \frac{\sin\beta\cos(\gamma - \gamma_s)}{0.1 - 0.008\alpha_s} \right) \right] \quad . \tag{3.41}$$

For $\alpha_s \geq 0.1$ rad:

$$G_{dT} = G_d \cdot \left[\left(\left(\frac{1+\cos\beta}{2} \right) + \left(\sin\beta - \beta\cos\beta - \pi \left(\sin\frac{\beta}{2} \right)^2 \right) \right. \right. \\ \cdot \left(0.00263 - 0.712\frac{G_b}{G_{on}} - 0.6883\left(\frac{G_b}{G_{on}}\right)^2 \right) \Bigg) \\ \left. \left. \cdot \left(1 - \frac{G_b}{G_{on}} \right) + \left(\frac{G_b}{G_{on}} \cdot \frac{\cos\theta}{\cos\theta_z} \right) \right] \quad , \tag{3.42}$$

is used.

The model decides whether the sky is clear or overcast on the basis of the so called modified clearness index kt' $[-]$. The modified clearness index is defined as:

$$k_t^{'} = \frac{k_t}{0.1 + 1.031 \cdot \exp\left(\frac{-1.4}{0.9 + \frac{9.4}{AM}} \right)} \quad , \tag{3.43}$$

where k_t $[-]$ is the normal clearness index defined as the fraction of global irradiance on the horizontal surface G and the extraterrestrial irradiance G_{on} $[W/m^2]$. AM symbolizes the air mass described in Chapter 2.2.1. Overcast weather is defined as $kt' < 0.3$, p. 5 [22].

3.3.3. PV module

The second part of the sub-model "PV Array" has the task of simulating the characteristic voltage-current curve of the PV in order to determine the power output of the system. The aim was to implement a model, that is only based on environmental data, such as the air temperature and the received global irradiance on the tilted surface, and parameters that can be extracted from any PV module data sheet. Furthermore, it was important to keep the computation time as short as possible, in order to obtain quantitative results quickly for larger grid simulations. One approach that meets these requirements is that of Vergura et al. [23], whose model is used for this thesis.

Estimate Panel Parameters

The used approach is based on the five parameter equivalent circuit (see Figure 3.8), but has the advantage of calculating the parameters directly from PV module data sheet values. The unknown parameters are $R_{s,c}$, $R_{sh,c}$, n, I_0 and I_{PV}, which must be determined for

the actual global irradiance and air temperature. These parameters are not available in manufacturer data sheets. Furthermore data sheet values are related to the whole PV module and not a single cell as these parameters, p. 2 [23]. The five parameter equivalent circuit is mathematically described as:

$$I = I_{PV} - I_0 \cdot \left(\exp\left(\frac{V + R_{s,c}I}{nV_t}\right) - 1\right) - \frac{V + R_{s,c}I}{R_{sh,c}} \quad . \tag{3.44}$$

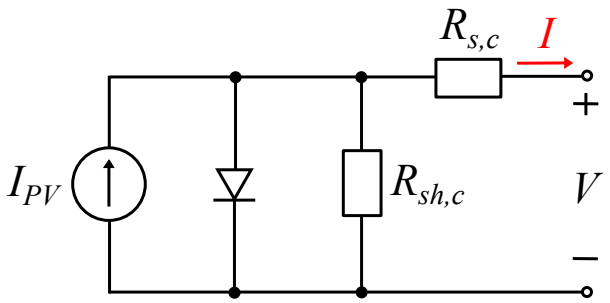


Figure 3.8.: Five parameter equivalent circuit of a PV with the unknown parameters $R_{s,c}$, $R_{sh,c}$, n, I_0 and I_{PV}.

So instead the following parameters are used to estimate the IV-curve of the PV module, p. 2-3 [23]:

- I^o_{mpp}: maximum power current at STC
- V^o_{mpp}: maximum power voltage at STC
- V^o_{oc}: open circuit voltage at STC
- I^o_{sc}: short circuit current at STC
- α_{isc}: temperature coefficient of I^o_{sc} $[\%/°C]$
- α_{voc}: temperature coefficient of V^o_{oc} $[\%/°C]$

All parameters are defined in the standard test conditions (STC), p. 3 [23].

As mentioned, the environmental conditions highly affect the power output of the PV module. The cell temperature can be approximately calculated as:

$$T_c = T_a + \frac{G}{800} \cdot (NOCT - 20) \quad , \tag{3.45}$$

where T_a is the air temperature and $NOCT$ is the nominal operating cell temperature. G is the global irradiance incident on the PV module. Because the shunt resistance $R_{sh,c}$ is responsible for the slope of the IV-curve at low voltage, p.86 [12]. It is approximated as:

$$R_{sh,c} = \frac{V^o_{mpp} - \alpha_{Voc} \cdot \Delta T \cdot V_{ooc}}{G_{pu} \cdot \frac{\left(I^o_{sc} - I^o_{mpp}\right)}{2}} \quad , \tag{3.46}$$

where ΔT is the difference of the cell temperature to the STC temperature and G_{pu} is the relative irradiance compared to the STC value, see (3.47) and (3.48).

$$\Delta T = T_c - 25 \quad , \tag{3.47}$$

$$G_{pu} = \frac{G}{1000} \quad . \tag{3.48}$$

On the other hand the serial resistance $R_{s,c}$ is dominant at high voltages, p.86 [12], and is approximated as:

$$R_{s,c} = \frac{\frac{V_{oc}^o - V_{mpp}^o}{4}}{G_{pu} \cdot \left(I_{mpp}^o + \alpha_{I_{sc}} \cdot \Delta T \cdot I_{sc}^o\right)} \quad . \tag{3.49}$$

The photo current, see first term from (3.44), is directly proportional to the irradiance, p. 3 [23]:

$$I_{PV} = G_{pu} \cdot (I_{sc}^o + \alpha_{I_{sc}} I_{sc}^o \Delta T) \quad . \tag{3.50}$$

The second term from (3.44) ranges from zero for $V = 0$ and I_0 for $V = V_{oc}$. In the approach from Vergura et al., this term is approximated as, p. 3 [23]:

$$\beta \cdot e^{\gamma \cdot (V + \alpha_{V_{oc}} \Delta T V_{oc}^o - V_{oc}^o)} \quad , \tag{3.51}$$

where β and γ are parameters to be estimated. They are defined as:

$$\beta = G_{pu} \cdot (I_{sc}^o + \alpha_{I_{sc}} \cdot \Delta T \cdot I_{sc}^o) - \frac{V_{oc}^o - \alpha_{V_{oc}} \cdot \Delta T \cdot V_{oc}^o}{R_{sh}} \quad , \tag{3.52}$$

$$\gamma = \frac{1}{V_{mpp}^o - V_{oc}^o} \cdot \ln \frac{G_{pu} \cdot \left(pI_{sc}^o - I_{mpp}^o\right) - (1-p) \cdot \alpha_{I_{sc}} \cdot \Delta T \cdot I_{sc}^o - \frac{p \cdot \left(V_{mpp}^o - \alpha_{V_{oc}} \cdot \Delta T \cdot V_{oc}^o\right)}{R_{sh}}}{p \cdot \left(G_{pu} \cdot (I_{sc}^o + \alpha_{I_{sc}} \cdot \Delta T \cdot I_{sc}^o) - \frac{V_{oc}^o - \alpha_{V_{oc}} \cdot \Delta T \cdot V_{oc}^o}{R_{sh}}\right)} \quad , \tag{3.53}$$

where the parameter p is defined as:

$$p = \frac{\left(V_{mpp}^o - \alpha_{V_{oc}} \cdot \Delta T \cdot V_{oc}^o\right) \cdot G_{pu} \cdot \left(I_{mpp}^o + \alpha_{I_{sc}} \cdot \Delta T \cdot I_{sc}^o\right)}{\left(V_{mpp}^o - \alpha_{V_{oc}} \cdot \Delta T \cdot V_{oc}^o\right) \cdot G_{pu} \cdot \left(I_{mpp}^o + \alpha_{I_{sc}} \cdot \Delta T \cdot I_{sc}^o\right) + \left(V_{oc}^o - V_{mpp}^o\right) \cdot G_{pu} \cdot \left(I_{sc}^o - I_{mpp}^o\right)} \quad . \tag{3.54}$$

The full IV characteristic for given environmental conditions, i.e., global irradiance G and air temperature T_a, is derived by substituting (3.51), (3.50), (3.49) and (3.46) into (3.44):

$$I \cdot \left(1 + \frac{R_s}{R_{sh}}\right) = G_{pu} \cdot (I_{sc}^o + \alpha_{I_{sc}} \cdot \Delta T \cdot I_{sc}^o) - \beta \cdot e^{\gamma (V + \alpha_{V_{oc}} \cdot \Delta T \cdot V_{oc}^o - V_{oc}^o)} - \frac{V}{R_{sh}} \quad . \tag{3.55}$$

Figure 3.9 recapitulates the presented process to determine the relationship between current and voltage of the PV module.

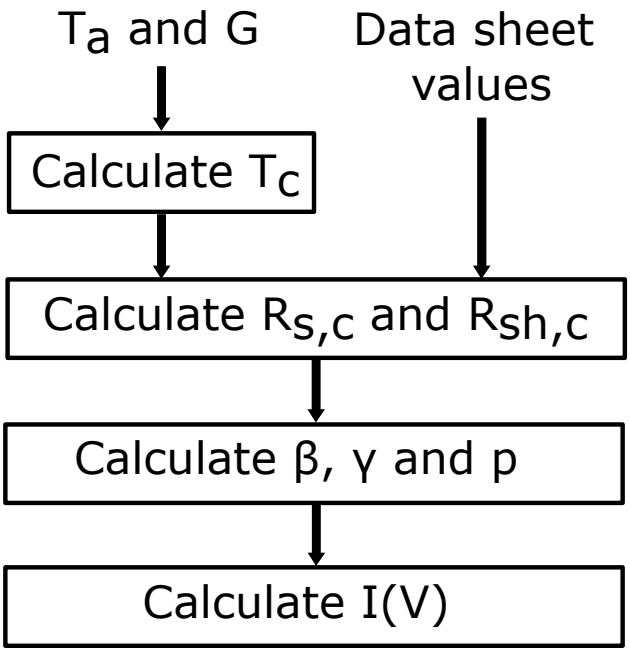


Figure 3.9.: Scheme to estimate the IV-curve of a PV module.

Maximum Power Point

The next step, to calculate the power output of the sub-model "PV Array", is to determine the maximum power point (MPP). For this purpose an algorithm was implemented which maximizes the function:

$$P = I(V) \cdot V \quad , \tag{3.56}$$

by finding the point where:

$$\frac{dP}{dV} = 0 \quad . \tag{3.57}$$

The MPP is defined by the maximum current I_{mpp} and the maximum voltage V_{mpp}. It is assumed that the individual PV modules of the array operate at the MPP at all times under the given environmental conditions. In real PV-systems, maximum power point trackers (MPPT) are used for this purpose, p. 181 [10]. For a visualization of the MPP see Figure 2.13.

For an implementation of the formulated problem, the function $fmincon()$ from the MATLAB optimization toolbox is used [18]. $fmincon()$ is used for solving constrained non-linear optimization multi variable problems, p. 1155 [18].

System Output

The final step is to calculate the total output. Figure 3.10 shows a potential layout for a PV array.

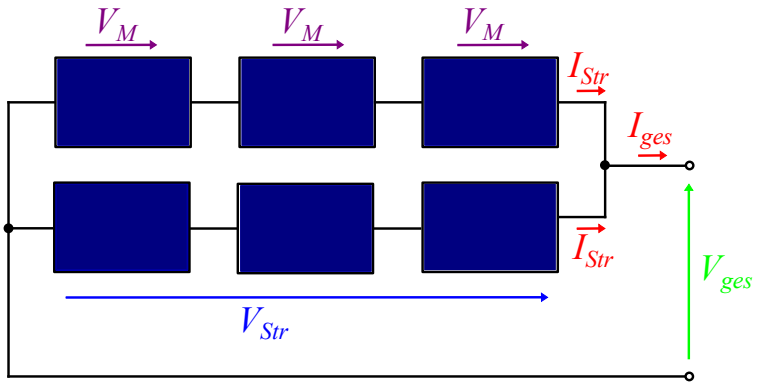


Figure 3.10.: Potential layout for a PV array, with two parallel strings, each with three modules in series.

The overall output of the sub-model "PV Array" is determined by applying Kirchhoff's laws.

$$V_{ges} = V_{Str} = V_M \cdot N_S \quad , \tag{3.58}$$

$$I_{ges} = I_{Str} \cdot N_P, \tag{3.59}$$

$$P_{ges} = V_{ges} \cdot I_{ges} \quad , \tag{3.60}$$

where N_S is number of PV modules in series and N_P is the number of parallel PV strings.

3.3.4. Inverter

The inverter is the second sub-model of the PV model. The task of the inverter is to convert DC into AC so that the power can be fed into the grid. It is assumed that only active power is injected into the grid. However, this does not apply to real inverters, as they also have to provide reactive power in order to avoid impermissible excess voltage in the grid, p. 201 [10]. The inner dynamics of an inverter are complex and are not considered in this thesis, to keep the scope reasonable. Therefore a simple approach consisting of a DC circuit, that represents the input from the PV array, and an AC circuit is used, see Figure 3.11. R_L represents the losses inside the inverter.

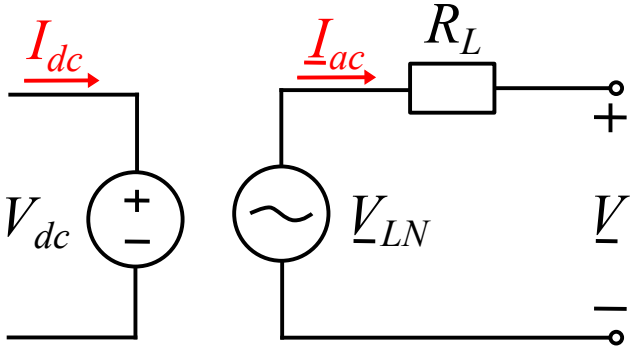


Figure 3.11.: Equivalent circuit of the inverter. The left side represents the DC input, the right side represents the AC side.

The AC voltage source represents a self-commutated, pulse-width-modulated three-phase inverter. Therefore the line-to-neutral voltage is calculated as, p. 238 [24]:

$$\underline{V}_{LN} = \frac{1}{2\sqrt{2}} \cdot m_a V_{dc} \quad , \tag{3.61}$$

where V_{dc} is the DC voltage and m_a is the modulation index. The inverter is assumed to adjust the modulation index in a way that the output voltage $\underline{V}$ remains constant. Applying the conversion of power results in:

$$V_{dc} I_{dc} = 3\underline{V}_{LN}\underline{I}_{ac} \quad . \tag{3.62}$$

The output voltage on the AC side is determined using Kirchhoff's laws:

$$\underline{V} = \underline{V}_{LN} - \underline{I}_{ac} R_L \quad , \tag{3.63}$$

where $\underline{V}$ is the grid voltage. Because the grid is assumed as rigid, V is a constant value. So inserting (3.62) in (3.63) can be solved for $\underline{I}_{ac}$ by solving the quadratic equation, where only the first solution is physically correct:

$$\underline{I}_{ac} = \frac{-\underline{V} + \sqrt{\underline{V}^2 + 4 \cdot \frac{R_L V_{dc} I_{dc}}{3}}}{2R_L} \quad . \tag{3.64}$$

The active power injected to the grid is determined as:

$$P_{ac} = 3\underline{V}\underline{I}_{ac} \tag{3.65}$$

Table 3.7 summarizes the input parameters and output parameters of the inverter model.

Input Parameters		
Symbol	**Name**	**Unit**
I_{dc}	Direct current	A
V_{dc}	Direct voltage	V
V	AC Grid voltage	V
R_L	Ohmic resistance	Ω
Output Parameters		
Symbol	**Name**	**Unit**
I_{ac}	AC current	A
P_{ac}	Active power	W

Table 3.7.: Input and output parameters of the Inverter.

3.3.5. Combined PV Model

The overall PV Brix is formed by connecting the two previously described sub-models PV array and inverter. Figure 3.12 shows the overall model. The parameters that define the

sub-models are listed in Table 3.6 and Table 3.7. The overall model has no input parameters because the irradiance data is included in the PV array sub-model. The final outputs of the PV Brix are the alternating current $\underline{I}$ and active power P, see Table 3.8, which are determined by (3.64) and (3.65).

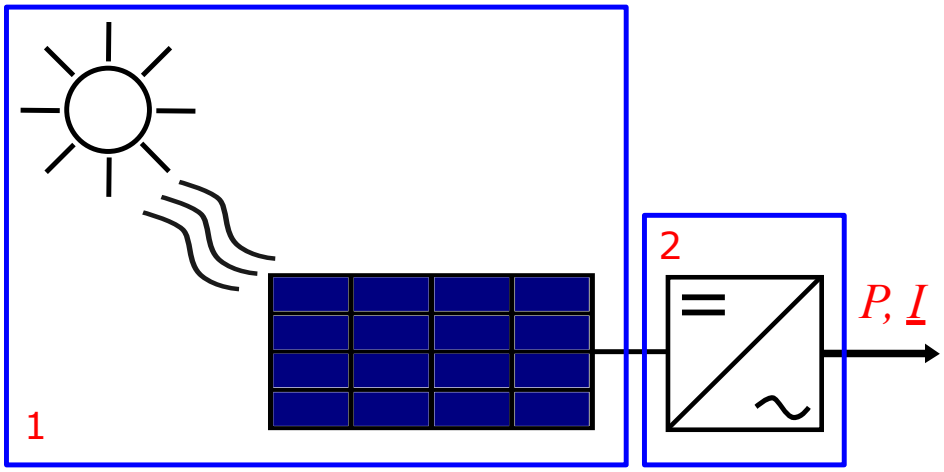


Figure 3.12.: Overall PV model: **1.** PV array, consisting of the sub-model irradiance and PV module; **2.** Inverter.

Output Parameters		
Symbol	**Name**	**Unit**
$\underline{I}$	Alternating current	A
P	Power	W

Table 3.8.: Output parameters of the overall PV model.

3.4. Battery Energy Storage System

The BESS Brix consists of the three sub-models Battery, Inverter and Rectifier. The BESS Brix has the task of storing energy when there is a surplus of electricity and discharging the battery when there is a power shortage in the grid. The following sub-chapters, describe the modelling process of the Battery, Rectifier and Inverter. At the end the overall BESS model is presented.

3.4.1. Battery

The sub-model Battery is described by an electrical model. Furthermore a simple approach to simulate the capacity decline, due to battery aging, was implemented. Table 3.9 shows the input parameters that define the battery model. The output parameters are the battery voltage V_{batt} and the battery current I_{batt}, see Table 3.9. In the following sections the necessary calculation steps for charging and discharging, as well as the parameters are explained.

Input Parameters		
Symbol	**Name**	**Unit**
SOC	State of charge	%
SOC_{min}	Minimum state of charge	%
SOC_{max}	Maximum state of charge	%
C_{init}	Initial capacity of battery	Ah
N_s	Battery cells in series	-
N_p	Battery cells in parallel	-
$M_{discharge}$	Maximum discharge rate	-
M_{charge}	Maximum charge rate	-
T	Battery cell temperature	K
t_{month}	Battery age in months	$months$
NC	Number of loading cycles	-
Output Parameters		
Symbol	**Name**	**Unit**
V_{batt}	Battery voltage	V
I_{batt}	Battery current	A

Table 3.9.: Input and output parameters of the battery.

Electrical Model

The Rint model was chosen because only static processes were considered, as the time steps within the simulation are an hour long. Therefore the RC pairs, as in the Thevenin equivalent circuit, see Figure 2.17, were cancelled because they represent the dynamic behaviour under rapid load changes, p. 506 [15]. The Rint model consists of a voltage source V_{OC} and a serial resistance R_S, see Figure 3.13.

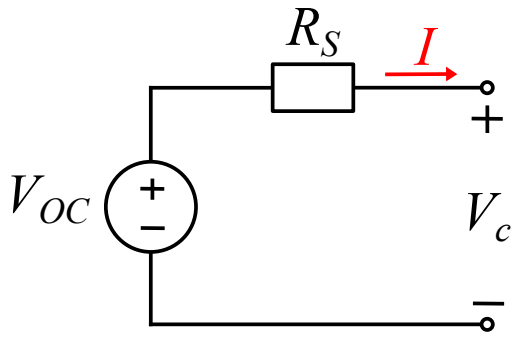


Figure 3.13.: Equivalent circuit of the Rint battery model, consisting of a voltage source V_{OC} and a serial resistance R_S.

It has to be noted once again, that no empirical formulas were found during the literature research. Only curve fitting based approaches for specific batteries were found. The formulas, that describe V_{OC} and R_S, were taken from p. 508 [15] and describe the voltage behaviour of an polymer lithium ion battery. Typically LiFePO4 based lithium batteries are used for power applications, p. 98 [13]. However, the formulas were taken due to a lack of alternatives.

$$V_{OC} = -1.031 \cdot \exp\left(-35 \cdot SOC\right) + 3.685 + 0.2156 \cdot SOC - 0.1178 \cdot SOC^2 + 0.321 \cdot SOC^3 \quad , \tag{3.66}$$

$$R_S = 0.1562 \cdot \exp\left(-24.37 \cdot SOC\right) + 0.07446 \quad . \tag{3.67}$$

Both V_{OC} and R_S are functions of the SOC. The temperature dependency was neglected because the error is negligible in a nearly constant-temperature application, p. 507 [15]. The SOC can be determined as:

$$SOC_t = SOC_{t_0} + \frac{1}{C_{usable}} \cdot \int_{t_0}^{t} I \, dt \quad , \tag{3.68}$$

where SOC_t is the new SOC, SOC_{t_0} is the old SOC, C_{usable} is the usable battery capacity, I is the current. The SOC increases when the battery is charged and decreases if the battery is discharged. The input parameters SOC_{min} and SOC_{max} ensure that the SOC lies between certain limits so that no depth discharge or overcharging of the battery occurs.

Two other characteristic parameters of the battery model are the maximum charge Q_{charge} and the maximum discharge $Q_{discharge}$ per time step:

$$Q_{charge} = M_{charge} \cdot C_{usable} \quad , \tag{3.69}$$

$$Q_{discharge} = M_{discharge} \cdot C_{usable} \quad , \tag{3.70}$$

where M_{charge} and $M_{discharge}$ are the allowed charge/discharge rate of the battery and are given as multiples of C. For example $M_{charge} = 1$ means that the battery can be fully charged in one hour. The computation process for charging and discharging is described in the next two sections.

The overall battery layout is described by the number of cells in series N_S and the cells in parallel N_P. Figure 3.14 illustrates a potential battery layout. The battery voltage V_{batt} and current I_{batt} are determined by using Kirchhoff's laws:

$$V_{batt} = N_s \cdot V_c \quad , \tag{3.71}$$

$$I_{batt} = N_p \cdot I_{Str} \quad . \tag{3.72}$$

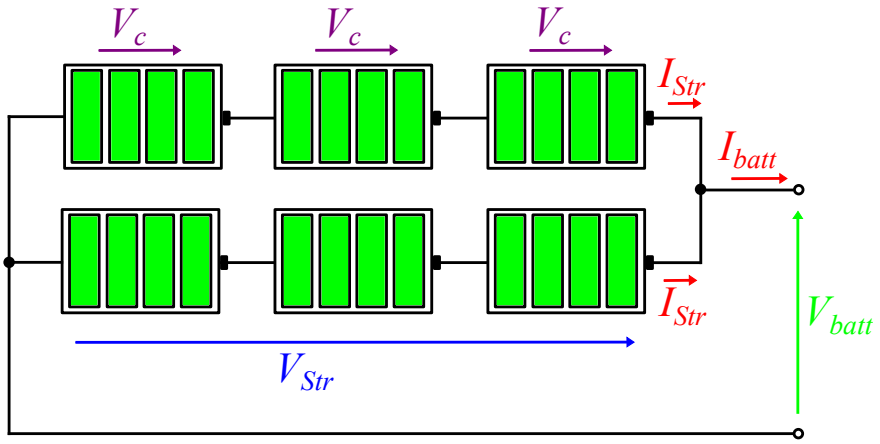


Figure 3.14.: Potential battery layout with two strings (N_P=2) and three cells in series (N_S=3).

Charging of Battery

The battery is charged if there is an electrical energy surplus in the grid ($P_{Grid} > 0$). In case of charging the input parameter of the battery is the direct current I of the rectifier, which represents the maximum possible current. The actual current over the period depends on, if the charge $Q = I \cdot \Delta t$ is smaller than the maximum possible charge per time step Q_{charge} and how much capacity is actually available. The principle of the algorithm is illustrated in Figure 3.15. For a better understanding the individual steps of the algorithm are listed below:

1) Calculate maximum possible charge per time step Q_{charge}.
2) Check if charge Q is smaller than maximum possible charge Q_{charge}.
3) Calculate possible new SOC.
4) Check if new SOC is within the boundaries.
5) Calculate the actual current.

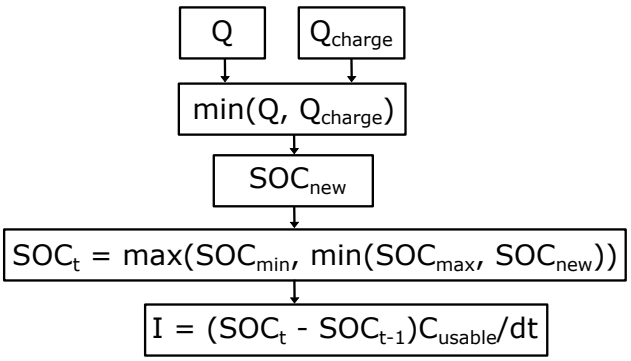


Figure 3.15.: Steps of the algorithm that estimates the actual current and ensure that the SOC stays between the boundaries.

After the actual current the cell voltage is calculated by applying Kirchhoff's laws on Figure 3.16:

$$V_c = V_{OC} + R_S I_{Str} \quad , \tag{3.73}$$

where $I_{Str} = I/N_P$ is the current per string. The average cell voltage is determined by calculating the voltage at the beginning and at the end of the charging circle.

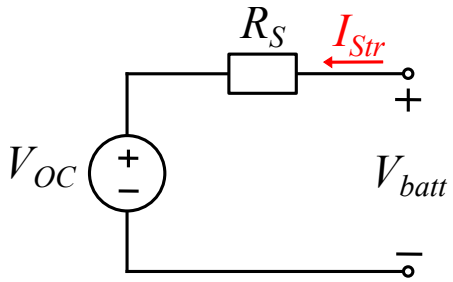


Figure 3.16.: Battery equivalent circuit when the battery gets charged.

Finally the battery voltage V_{batt} and current I_{batt} are determined by using Kirchhoff's laws on Figure 3.14.

Discharging of Battery

The battery is discharged if there is an electrical energy shortage in the grid ($P_{Grid} < 0$). In case of discharging, the input parameter of the battery is the needed power of the grid. The current per string at the beginning of the discharge process is estimated by applying Kirchhoff's laws on Figure 3.17.

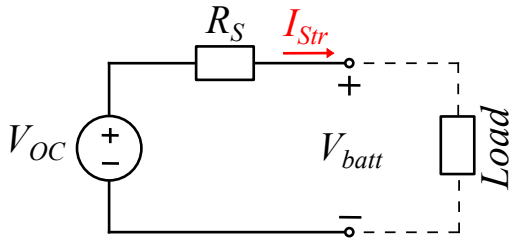


Figure 3.17.: Battery equivalent circuit when the battery gets discharged.

This leads to a quadratic equation, where only the second solution is physically stable:

$$I_{Str} = \frac{V_{OC}}{2R_S} - \sqrt{\left(\frac{V_{OC}}{2R_S}\right)^2 - \frac{P}{N_S N_P R_S}} \quad , \tag{3.74}$$

where N_S is the cells in series and N_P is the cells in parallel. The derivation can be seen in Appendix B. After that an algorithm calculates if the battery can actually deliver the charge, by checking if the charge $Q = I \cdot \Delta t$ is smaller than the maximum discharge per time step $Q_{discharge}$ and how much capacity is actually available. The principle of the algorithm is illustrated in Figure 3.18. For a better understanding the individual steps are listed below:

1) Calculate current using (3.74).
2) Calculate Q and maximum possible discharge per time step $Q_{discharge}$.
3) Check if charge Q is smaller than maximum possible charge $Q_{discharge}$.
4) Calculate possible new SOC.
5) Check if new SOC is within the boundaries.
6) Calculate the actual power.

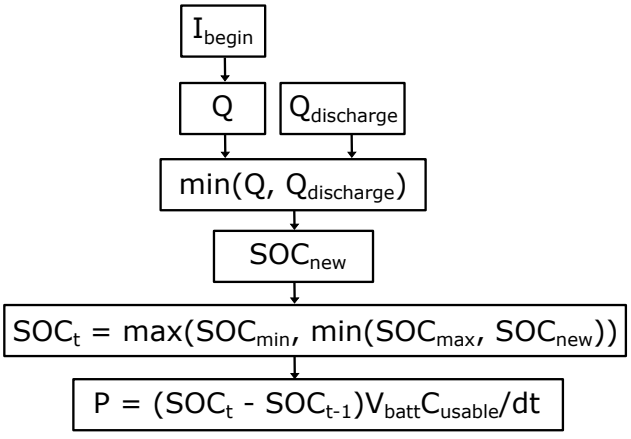


Figure 3.18.: Steps of the algorithm that estimates the actual power output and ensures that SOC stays between the boundaries.

The average string current I_{Str} and the cell voltage V_c are determined by calculating them at the beginning and at the end of the discharge process. The battery voltage V_{batt} and current I_{batt} are determined by using Kirchhoff's laws on Figure 3.14.

Battery Aging

Another feature of the battery model is a simple model to estimate the capacity fade due to battery aging. The aging effects differ for various chemicals of batteries. Furthermore the aging process depends on the utilization of the battery, p. 3 [25]. The following semi-empirical formulas were taken from Stroe et al., see p. 3 [25], and are suitable for a LiFePO4 battery. This type of battery is popular for stationary power applications, p. 98 [13].

As mentioned in Chapter 2.3.2, the battery aging processes can be categorized into calendar aging and cycle aging. The calendar aging can be described by following equation:

$$C_{fade,cal} = \alpha_t \cdot \exp\left(\beta_t \cdot T\right) \cdot t^n \quad , \tag{3.75}$$

where t is the time in months $[-]$ and T is the temperature in $[K]$. The coefficients have following quantities in this approach: $\alpha_t = 3.087{\cdot}10^{-7}$ $[-]$, $n = 0.5$ $[-]$ and $\beta_t = 0.05146$ K^{-1}.

The cycle aging in % is described by the following semi-empirical formula:

$$C_{fade,cyc} = \alpha_{NC} \cdot \exp\left(\beta_{NC} \cdot T\right) \cdot NC^n \quad , \tag{3.76}$$

where NC is the number of equivalent reference cycles. The equivalent reference cycle indicates the number of charging cycles. It has to be noted that also "half-cycles" count, p. 210 [26]. For example if the battery is charged from SOC = 0 to SOC = 0.5, then fully discharged and again charged to SOC = 0.5, would equal an additional equivalent reference cycle NC of one. The coefficients have following quantities in this approach: $\alpha_{NC} = 6.87 \cdot 10^{-5}$ $[-]$, $\beta_{NC} = 0.027$ K^{-1} and $n = 0.5$ $[-]$.

The overall capacity loss is the sum of calendar aging and cycle aging. Therefore the remaining capacity in % is:

$$C_{CF} = 1 - \left(C_{fade,cyc} + C_{fade,cal}\right) \quad . \tag{3.77}$$

Finally the usable capacity in $[Ah]$ is determined as:

$$C_{usable} = C_{init} \cdot C_{CF} \quad . \tag{3.78}$$

3.4.2. Inverter

The inverter is the second sub-model of the BESS. It is the same model as is used for the PV model. The task of the inverter is to transform the DC output of the battery into AC, when the battery discharges. The input of the inverter model are the battery voltage and current. The output variables are the alternating current and the power. For a detailed explanation see Chapter 3.3.4.

3.4.3. Rectifier

The rectifier is the third sub-model of the BESS. The rectifier has the task to convert the AC from the grid into DC when the battery gets charged. As for the inverter, the inner dynamics of the rectifier are complex and out of interest in this thesis, to keep the scope reasonable. A simple approach consisting of a three phase full bridge rectifier was chosen

as there are analytical relationships between the input and output variables, p. 139 [24]. Figure 3.19 illustrates the equivalent circuit of the rectifier. The left side is the grid side and represents the AC input, while the right side is the battery facing DC side. The voltage source stands for the three phase full bridge rectifier and the serial resistance R_L represents the losses inside the rectifier.

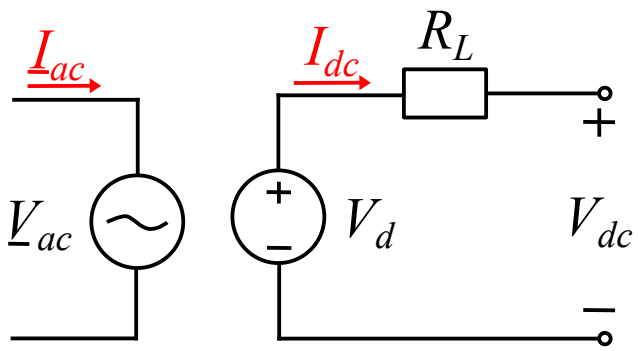


Figure 3.19.: Equivalent circuit of the rectifier. The left side represents the AC input, the right side stands for the towards the battery facing DC side.

The average DC output voltage of a three phase full bridge rectifier is determined as, p. 139 [24]:

$$V_d = \frac{3\sqrt{2}}{\pi} \cdot V_{LL} \quad , \tag{3.79}$$

where V_{LL} is the line-to-line voltage. The direct current I_{dc} can be calculated by applying the power conversion:

$$I_{dc} = \frac{P_{ac}}{V_d} \quad , \tag{3.80}$$

where P_{ac} is the active power that has to be stored in the battery. The final DC output voltage of the rectifier is determined by applying Kirchhoff's laws on :

$$V_{dc} = V_d - R_L \cdot I_{dc} \quad . \tag{3.81}$$

The input and output variables are summarized in Table 3.10.

Input Parameters		
Symbol	**Name**	**Unit**
V_{ac}	Line-to-line AC voltage	V
P_{ac}	Power	W
R_L	Ohmic resistance	Ω
Output Parameters		
Symbol	**Name**	**Unit**
V_{dc}	Direct voltage	V
I_{dc}	Direct current	A

Table 3.10.: Input and output parameters of the rectifier.

3.4.4. Combined BESS Model

The overall BESS Brix is formed by connecting the sub-models battery, inverter and rectifier. Figure 3.20 illustrates the structure of the overall model. The input parameter is the grid power P_{Grid}. It is negative if the grid needs power and positive when electrical energy has to be stored. The rectifier is only active if there is an electrical energy surplus ($P_{Grid} > 0$), while the inverter is only active if there is an electrical energy shortage ($P_{Grid} < 0$) in the grid. The output parameters are the alternating current I and the actual power P_{out}, see Table 3.11. The power is calculated either as the battery power:

$$P_{out} = V_{batt} \cdot I_{batt} \quad , \tag{3.82}$$

if the battery charges ($P_{Grid} > 0$), or as the output of the inverter with (3.65) if the battery discharges ($P_{Grid} < 0$).

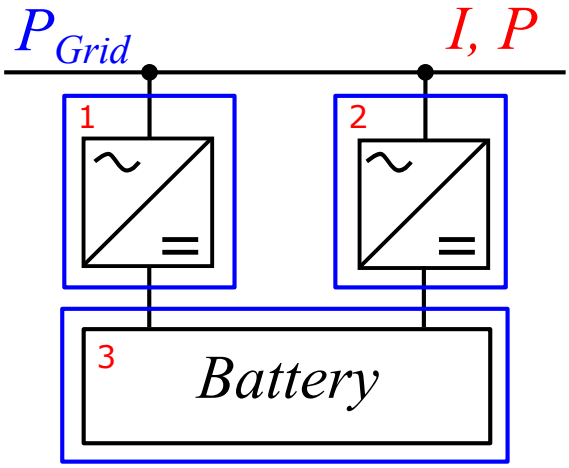


Figure 3.20.: Overall BESS model: **1.** Rectifier; **2.** Inverter; **3.** Battery.

Input Parameters		
Symbol	**Name**	**Unit**
P_{Grid}	Power surplus / shortage of grid	W
Output Parameters		
Symbol	**Name**	**Unit**
I	Alternating current	A
P	Power	W

Table 3.11.: Input and output parameters of the BESS model.

4. Validation

In this chapter the photovoltaic and wind turbine are validated. This is achieved by comparing the results with measurement data from existing systems and freely available data sets. The results will be used to draw conclusions about the accuracy and applicability of the models. Unfortunately, no suitable data could be found for the battery energy storage system (BESS) model. Therefore the BESS model can not be validated.

The mean absolute percentage errors (MAPE) are provided in order to better assess the quality of the models. The MAPE is determined as:

$$M = 100 \cdot \frac{1}{n} \sum_{i=1}^{n} \left| \frac{S_i - A_i}{A_i} \right| \quad , \tag{4.1}$$

where S_i is the simulated value, A_i is the measured or actual value and n is the number of data points.

4.1. Wind Turbine

The Wind Turbine model is tested setting the simulation parameters equal, or as close as possible to a real turbine and comparing the estimate of the power curve with the one provided from the wind turbine manufacturer. The data for the power curves are extracted from $wind-turbine-models.com$, a portal for wind energy that is supported by the wind power industry [27].

Unfortunately, no data on the characteristics of the generator and the converters are available, as these are not publicly accessible. Instead, the efficiencies are assumed based on literature and commercially available components. The efficiency of the PMSG is assumed as 97 %, p. 2 [28]. The efficiency of the MSC and GSC is assumed to be 98.5 % each, e.g., ABB ACS880 Full Converter with overall efficiency of approximately 97 % at rated power, p. 3 [29]. In the following sections the power curve for two different variable speed wind turbines is estimated and tested against manufacturer data.

4.1.1. Lagerwey LW-72/2.0

The Lagerwey LW-72/2.0 is a variable speed onshore wind turbine with a PMSG and a full scale converter. The rated power for this configuration is 2 MW. The manufacturer power curve and parameters are extracted from [30]. The extracted parameters are shown in Table 4.1.

Symbol	Name	Value
r	Blade length	35.6 m
v_{cut-in}	Cut-in wind speed	2.5 m/s
v_{rated}	Rated wind speed	13.0 m/s
$v_{cut-out}$	Cut-out wind speed	25.0 m/s
n_{max}	Maximum rotor speed	24 rpm

Table 4.1.: Parameters of the Lagerwey LW-72/2.0 wind turbine.

Figure 4.1 illustrates the comparison of the calculated power curve with the manufacturer power curve.

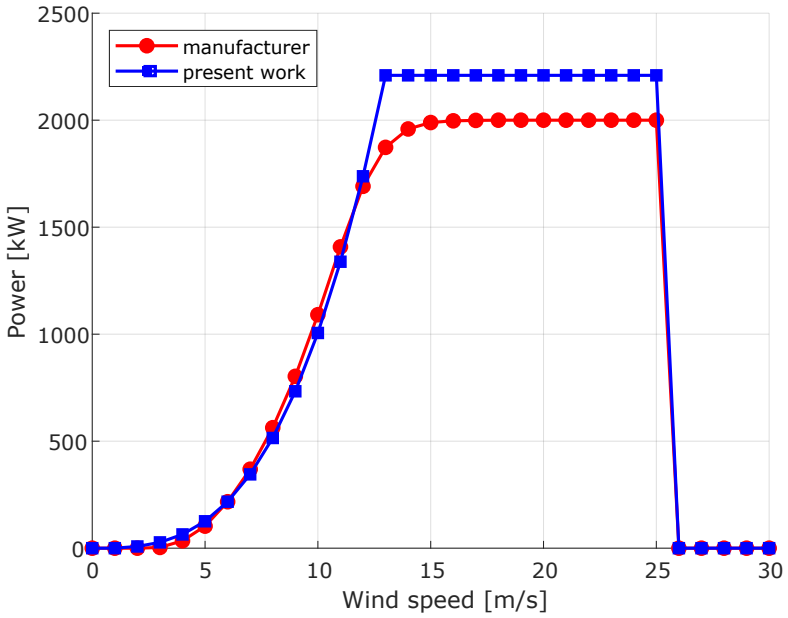


Figure 4.1.: Comparison between the simulated power curve and the power curve provided by the manufacturer for the Lagerwey LW-72/2.0 wind turbine.

Overall, the simulation shows good qualitative agreement with the manufacturer power curve. However, there are significant deviations at rated wind speed and above. As can be seen in Figure 4.1, the power limitation starts before the rated speed wind speed of $v_{rated} = 13\ m/s$, whereas in the simulation the power reduction starts at v_{rated}. One reason for this discrepancy is that the used power control technology is not known. Another reason for deviations is that the formulas used to estimate the course of the power coefficient may not apply for the used rotor blades. Furthermore the used efficiencies for the PMSG, MSC and GSC may be different. The MAPE for the simulation equals 31.31 %.

4.1.2. Enercon E-40/5.40

The Enercon E-40/5.40 is a variable speed onshore wind turbine with a PMSG and a full scale converter. The rated power for this configuration is 500 kW. The manufacturer power curve and parameters are extracted from [31]. The extracted parameters are shown in Table 4.2.

Symbol	Name	Value
r	Blade length	20.15 m
v_{cut-in}	Cut-in wind speed	2.5 m/s
v_{rated}	Rated wind speed	12.0 m/s
$v_{cut-out}$	Cut-out wind speed	25.0 m/s
n_{max}	Maximum rotor speed	38 rpm

Table 4.2.: Parameters of the Enercon E-40/5.40 wind turbine.

Figure 4.2 illustrates the comparison of the calculated power curve with the manufacturer power curve.

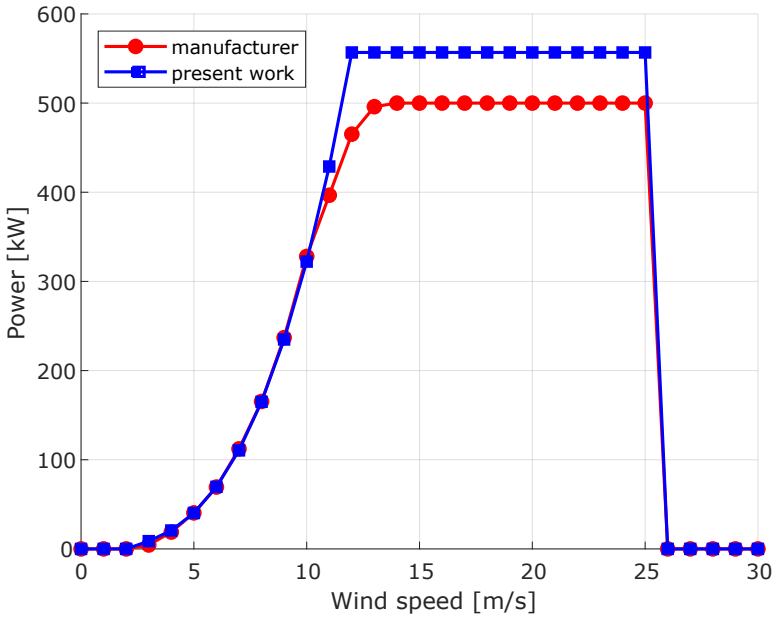


Figure 4.2.: Comparison between the simulated power curve and the power curve provided by the manufacturer for the Enercon E-40/5.40 wind turbine.

Again, the simulated power curve and the manufacturer curve demonstrate good agreement. However, significant deviations are also observed at and above the rated wind speed in this case. As can be seen in Figure 4.2, the power limitation starts before the rated speed wind speed of $v_{rated} = 12\ m/s$, whereas in the simulation the power reduction starts at v_{rated}. The reasons for the deviation mentioned in the previous test also apply in this case. The

MAPE for the simulation equals 9.34 %. The smaller error can possibly be explained by the fact that the c_p-curve fits the E-40/5.40 better than the LW-72/2.0 wind turbine. Another explanation would be that the implemented control corresponds approximately to that of the real turbine.

4.1.3. Wind Turbine Applicability

Overall, the power curves can be reproduced to a good qualitative approximation. The results show that the power curves of different wind turbines can be simulated by changing the parameters. It should be noted that no statement can be made to assess the quality of the model, as this requires further tests, e.g., comparison with a real life wind turbine. In order to improve the results, more information on the used power control must be collected. In general, more details about the individual components should be obtained. This would require contacting the manufacturers, as this data is not publicly available. Furthermore, a power coefficient adapted to the wind turbine should be used. Due to the scarcity of comparable data, other aspects of the model could not be tested. Most datasets found, either focus on dynamic phenomena or are chargeable.

Since the model does not require any iterations except for the pitch and tip-speed-ratio control, the designed model is suitable for use in a stationary grid simulation due to its computation speed. In order to be able to use the model for dynamic simulations, the model would have to be extended accordingly. For example by implementing a control system or more complex models for PMSG, MSC and GSC.

4.2. Photovoltaic

The photovoltaic model is validated by setting the simulation parameters equal, or as close as possible to a real PV system and comparing the results with measured data. The used dataset was taken from Theristis et al. [32]. The dataset is linked on p. 1146 [32]. The dataset includes hourly averaged weather data, PV module manufacturer data sheets, system layout and the hourly DC power output. Therefore only the sub-model PV array can be tested, as no comparative data for the AC power output is included in the dataset.

Theristis et al. includes six different scenarios at two different locations. The different scenarios include different solar cell technologies as well as fixed-angle and tracking system PV systems. The validation process is carried out for the location Roskilde (Denmark), where scenario 5 and 6 is tested for two different days.

- **Scenario 5** is a PV system that uses monofacial Allmax+ 305 Wp PV modules. The system consists of 88 modules (22 in series and 4 strings in parallel), with orientation to the south and a tilt of 25°.

- **Scenario 6** is a PV system that uses bifacial Duomax Twin 295 Wp PV modules. The system consists of 88 modules (22 in series and 4 strings in parallel), with orientation to the south and a tilt of 25°.

These scenarios were chosen because the layouts of the systems are the same and only the type of used PV module differs. This makes it possible to investigate whether the error differs significantly for different type of PV modules. The layout and positional parameters of the PV system are shown in Table 4.3 and are identical for both scenarios. The validation process is carried out in three steps. First, the conversion of horizontal irradiance data into irradiance on a tilted surface is tested. The next step is to test the reproduction of the characteristic IV- and PV-curve for different type of PV modules. The final step includes the verification of the DC power output. Due to time reasons, the influence of the layout could not be validated.

Symbol	Name	Value
L_{loc}	Longitude	12.104°
L_{st}	Standard meridian	15°
ϕ	Latitude	55.696°
β	Slope of modules	25°
γ	Surface azimuth of modules	0° (south)
N_s	Modules in series	22
N_p	Modules in parallel	4

Table 4.3.: Layout and positional parameters of the PV system in Roskilde, Denmark. The dataset is linked on p. 1146 [32].

4.2.1. Irradiance Model

For the validation of the irradiance model, 5th March 2019 and 1st July 2019 are selected. The dates are chosen randomly. It has to be noted, that more days should be compared, in order to better assess the quality of the simulation. For the determination of the irradiance on the inclined PV modules, the date and time, the global horizontal irradiance (GHI), the direct horizontal irradiance (DHI), ground albedo ρ and the air temperature T_a are extracted from the dataset. The geographical position of the PV system in Roskilde (Denmark) is shown in Table 4.3.

Figure 4.3 illustrates the comparison of the calculated global irradiance with measured data on the tilted PV modules for the simulation on 5th March 2019.

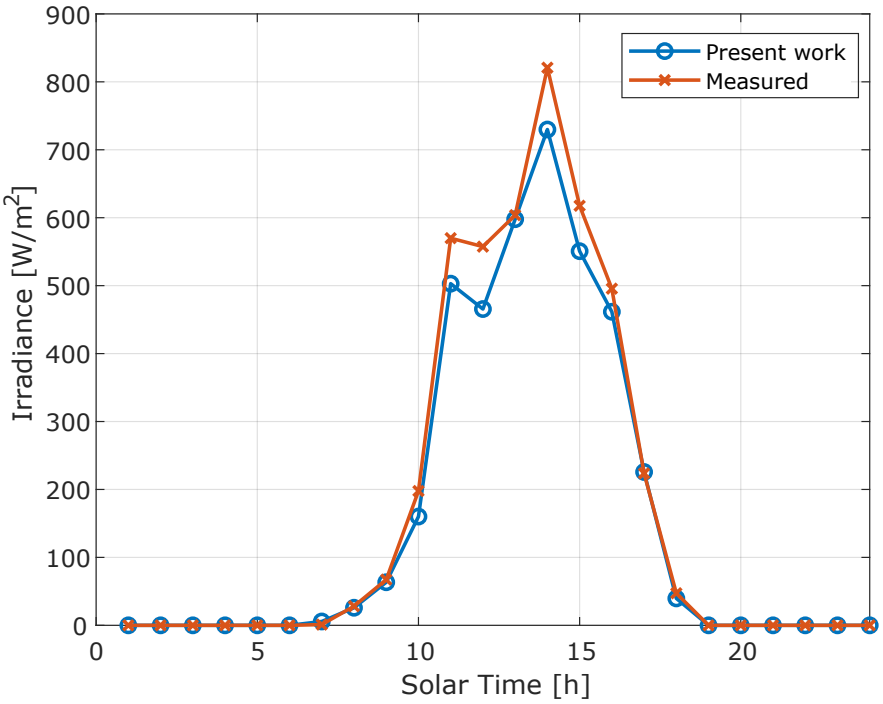


Figure 4.3.: Comparison of the calculated global irradiance with measured data vs solar time on the PV modules with tilt angle $\beta = 25°$ on 5 March 2019.

The simulation achieves good agreement, and the course of the measured irradiance can be reproduced approximately. The MAPE equals 22.08 %. The magnitude of the error is due to the values at 7 a.m. solar time, where the simulated irradiance is $G_{sim} = 5.66\ W/m^2$ and the measured value is $G_{mea} = 1.08\ W/m^2$. An explanation could be the possible shading of the PV modules at this time. If this data point is excluded the MAPE is significantly lower and equals 4.68 %.

Figure 4.4 illustrates the comparison of the calculated global irradiance with measured data on the tilted PV modules for the simulation on 1st July 2019.

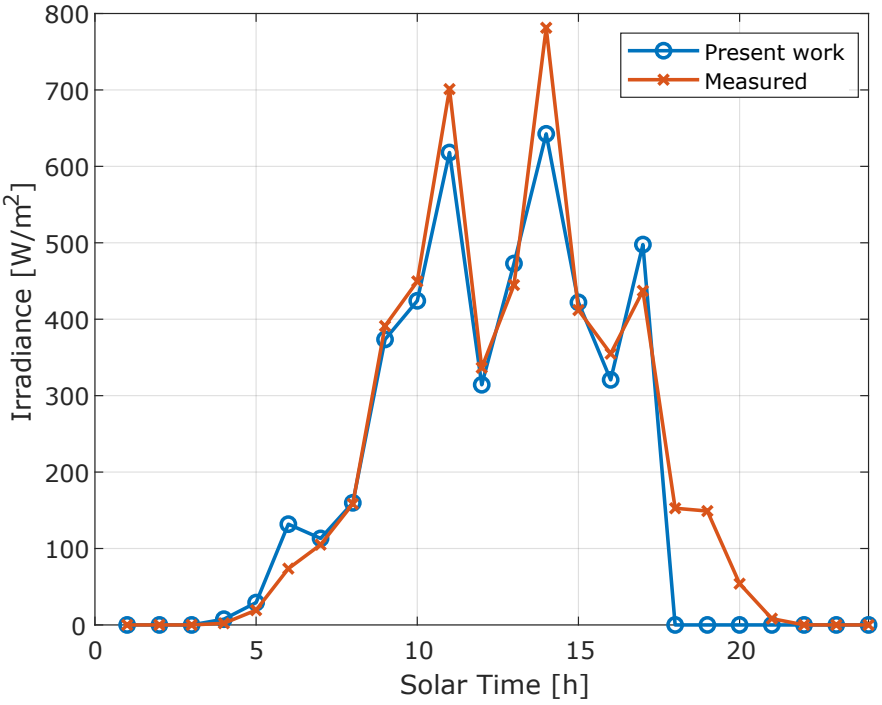


Figure 4.4.: Comparison of the calculated global irradiance with measured data vs solar time on the PV modules with tilt angle $\beta = 25°$ on 1st July 2019.

Again, the course of the simulated irradiance shows good agreement with the measured values. The area between 6 and 10 p.m. solar time is particularly interesting, as irradiance values of up to 150 W/m^2 are measured here, while the simulation determines an irradiance of 0 W/m^2. This deviation is due to the data set, as the GHI is given as 0 W/m^2 at this time, but irradiance was still measured on the elevated modules. The MAPE equals 40.23 % on this day. If the data points between 6 and 10 p.m. solar time are excluded, MAPE equals 24.47 %.

4.2.2. Characteristic Curves

In this section it is tested whether the model can reproduce the characteristic curves of two different PV modules. The PV system in Scenario 5 uses monofacial Allmax+ 305 Wp modules, while Scenario 6 uses bifacial Duomax Twin 295 Wp modules, p. 1146 [32]. However, in the Duomax Twin manufacturer data sheet only the characteristic curves of the 310 Wp module are illustrated. In the following, the characteristic IV- and PV-curve are simulated and compared with the manufacturer product data sheets. The link to the manufacturer data sheets can be found on p. 1146 [32]. The characteristic curves are reproduced at standard testing conditions (STC), i.e., 25 $°C$ cell temperature, for five different levels of irradiance, with the highest being 1000 W/m^2 and the lowest 200 W/m^2.

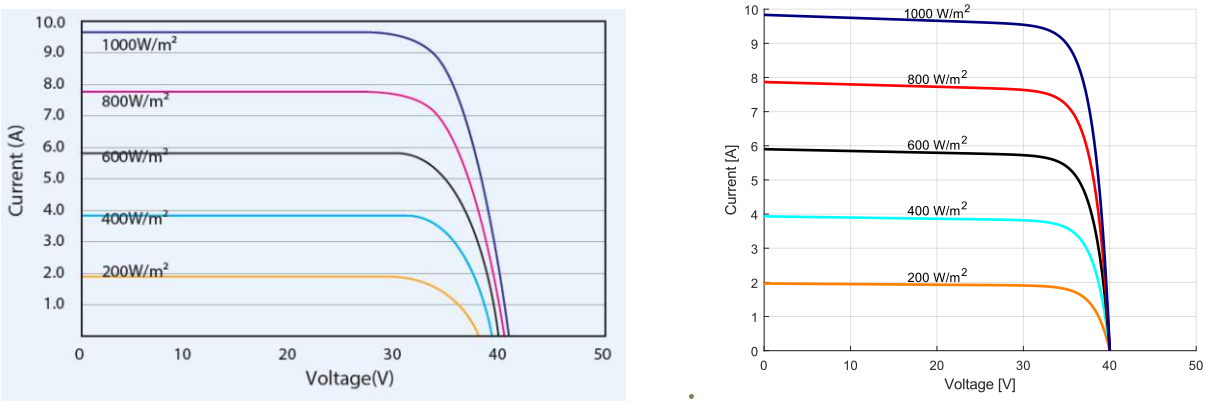
Allmax+ 305 Wp

The Allmax+ 305 Wp is a monofacial PV module. Table 4.4 shows the extracted parameters from the manufacturer data sheet, which are used to reproduce the modules characteristic curves.

Symbol	Name	Value
I_{ompp}	Maximum power current at STC	9.28 A
V_{ompp}	Maximum power voltage at STC	32.9 V
V_{ooc}	Open circuit voltage at STC	40.0 V
I_{osc}	Short circuit current at STC	9.85 A
α_{isc}	Temperture coefficient of I_{osc}	0.05 $\%/°C$
α_{voc}	Temperture coefficient of V_{ooc}	-0.29 $\%/°C$
$NOCT$	Nominal operating temperature	44 $°C$

Table 4.4.: Data sheet parameters of monofacial Allmax+ 305 Wp PV module.

Figure 4.5 illustrates the results for the comparison of the IV-characteristic. The currents for low voltages can be reproduced to a good approximation. However, at high voltages, there are deviations. While the manufacturers specifications clearly show that the voltages are different at currents towards 0 A, in the model presented, they all end at the same point at approximately 40 V. One possible explanation could be that the cell temperature T_c is not 25 $°C$ for all irradiation values, as assumed in the simulation. However, the module data sheet does not provide any information on this.

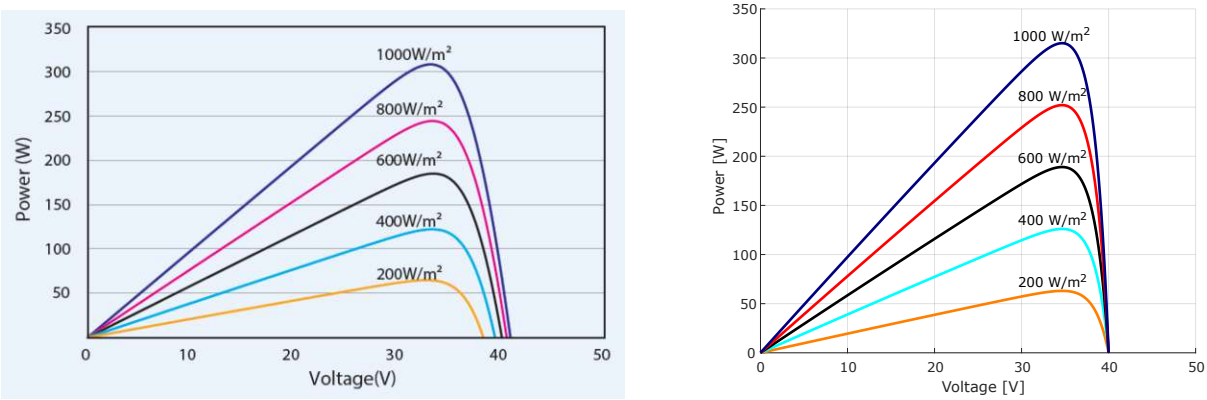


(a) (b)

Figure 4.5.: IV-curve for the Allmax+ 305 Wp PV module. a) Manufacturer data. b) Estimate in this thesis.

Figure 4.6 illustrates the results for the comparison of the PV-characteristic. The same applies here as before. While the curve can be reproduced well for small voltages, there are

deviations for larger voltages.



(a) (b)

Figure 4.6.: PV-curve for the Allmax+ 305 Wp PV module. a) Manufacturer data. b) Estimate in this thesis.

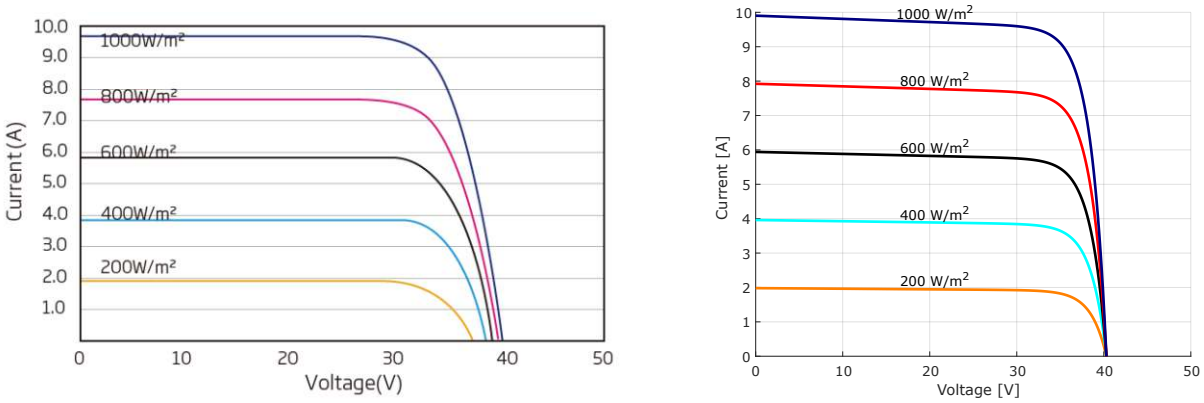
Duomax Twin 310 Wp

The Duomax Twin 310 Wp is a bifacial PV module. Table 4.5 shows the extracted parameters from the manufacturer data sheet, which are used to reproduce the modules characteristic curves.

Symbol	Name	Value
I_{ompp}	Maximum power current at STC	9.30 A
V_{ompp}	Maximum power voltage at STC	33.3 V
V_{ooc}	Open circuit voltage at STC	40.3 V
I_{osc}	Short circuit current at STC	9.92 A
α_{isc}	Temperture coefficient of I_{osc}	0.05 $\%/°C$
α_{voc}	Temperture coefficient of V_{ooc}	-0.29 $\%/°C$
$NOCT$	Nominal operating temperature	44 $°C$

Table 4.5.: Data sheet parameters of bifacial Duomax Twin 310 Wp PV module.

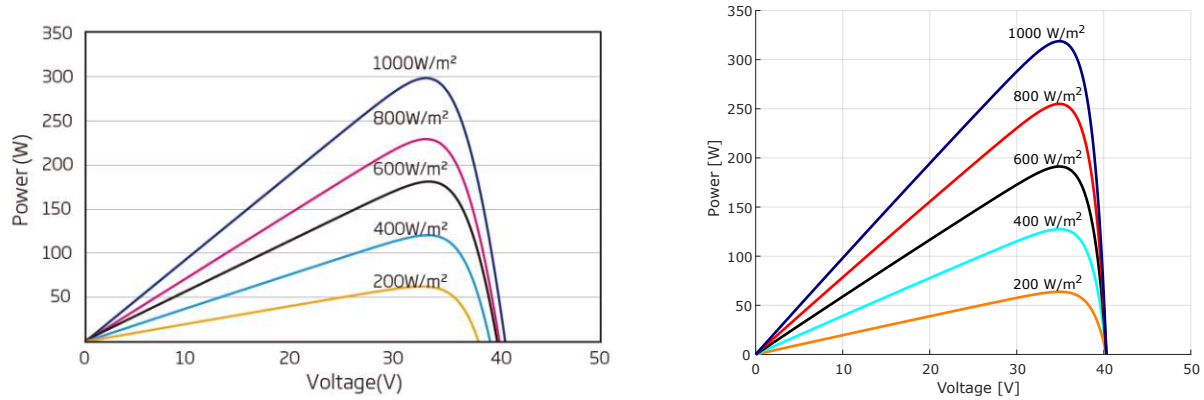
Figure 4.7 illustrates the results for the comparison of the IV-characteristic. The currents for low voltages can be reproduced to a good approximation. Again, the different IV-curves end at the same point at approximately 41 V, while the manufacturers specifications clearly show that the voltages are different at currents towards 0 A. As mentioned in the previous test, this could be due the assumption that the cell temperature T_c is 25 $°C$ at each irradiation value.



(a) (b)

Figure 4.7.: IV-curve for the Duomax Twin 310 Wp PV module. a) Manufacturer data. b) Estimate in this thesis.

Figure 4.8 illustrates the results for the comparison of the PV-characteristic. The same applies here as before. While the curve can be reproduced well for small voltages, there are deviations for larger voltages. Furthermore the maximum power is slightly higher for the individual curves. For example for $G = 1000\ W/m^2$ the maximum power point is at 300 W according to the manufacturer, while the model estimates 318 W. In the used model from Vergura et al., this is justified by the dependence of the parameter $\gamma\ [V^{-1}]$ on the constraint $\frac{dP}{dV} = 0$ in MPP, p. 11 [23].



(a) (b)

Figure 4.8.: PV-curve for the Duomax Twin 310 Wp PV module. a) Manufacturer data. b) Estimate in this thesis.

In conclusion, it can be said that the characteristic curves can be reproduced to a good approximation. There is no significant difference in accuracy between monofacial and bifacial modules.

4.2.3. DC Power Output

In this section, the DC power output for the two PV modules from Scenario 5 and 6 is estimated using the previously determined irradiation data for 5th March 2019 and 1st July 2019 from Section 4.2.1. In order to better assess the accuracy of the simulation, more days should be simulated. The dataset with the measured DC power output is linked on p. 1146 [32].

Table 4.6 shows the comparison of the calculated DC power with the measured data on the 5th March 2019 for the Scenarios 5 and 6, including total power per day, average power per day, absolute error and MAPE.

	Pres. Work S5	**Mes. S5**	**Pres. Work S6**	**Mes. S6**
Total power per day [kW]	106.906	108.40	102.975	108.496
Average power per day [kW]	4.454	4.517	4.291	4.521
Abs. error [kW]	1.494	-	5.521	-
MAPE [%]	23.88	-	20.77	-

Table 4.6.: Comparison of simulated and measured DC power for 5th March 2019 for Scenario 5 and 6, including total power per day, average power per day, absolute error and MAPE.

The measured total and average DC power can be approximately reproduced in the simulation. For Scenario 5, the absolute error equals 1.494 kW and the MAPE equals 23.88 %. For Scenario 6, the absolute error equals 5.521 kW. The MAPE is slightly lower and equals 20.77 %. As mentioned in Section 4.2.1 the magnitude of the error is mainly due to the values at 7 a.m. solar time and could be due to possible shading of the PV modules at this time. If this data point is excluded the MAPE for Scenario 5 equals 6.59 % and for Scenario 6 MAPE equals 8.13 %.

Table 4.7 illustrates the comparison of the calculated DC power with the measured data on 1st July 2019 for the Scenarios 5 and 6. The table includes the total power per day, the average power per day, the absolute error and the MAPE.

	Pres. Work S5	**Mes. S5**	**Pres. Work S6**	**Mes. S6**
Total power per day [kW]	123.179	125.601	118.67	132.215
Average power per day [kW]	5.132	5.233	4.944	5.509
Abs. error [kW]	2.422	-	13.545	-
MAPE [%]	31.84	-	26.4	-

Table 4.7.: Comparison of simulated and measured DC power for 1st July 2019 for Scenario 5 and 6, including total power per day, average power per day, absolute error and MAPE.

Again, the simulated and measured values show good qualitative agreement. For Scenario

5, the absolute error is 2.422 kW and the MAPE equals 31.84 %. For Scenario 6, the absolute error is 13.545 kW and MAPE equals 26.4 %. Same as in Section 4.2.1 there is a big discrepancy around 6 to 10 p.m. solar time. As mentioned this is mainly due to the difference in measured and simulated irradiance at this time. If the data points between 6 to 10 p.m. solar time are excluded in the determination, the MAPE for the two scenarios is significantly lower. For Scenario 5 the MAPE now equals 18.2 % and for Scenario 6, MAPE equals 11.68 %.

4.2.4. PV Applicability

The tests carried out show that the implemented model can reproduce the operating behaviour of real PV systems well. Both monofacial and bifacial PV models can be simulated to a good approximation by changing the parameters. The model could be further refined with a more complex irradiation model. Further optimization is also necessary for the IV-characteristic to better reflect the behaviour at high module voltages. Subsequent works should also focus on testing the model on different cell technologies, e.g., multicrystalline cells. Since the model does not require iterations, except for the MPP, the computation time is fast, which makes the model suitable for stationary grid simulations.

5. Conclusion and Outlook

The aim of this thesis was to develop and validate models for wind turbines, photovoltaic (PV) systems, and battery storage systems (BESS) for use in a stationary grid simulation. This effort is critical in addressing the increasing reliance on renewable energy sources and the associated challenges in maintaining grid stability and reliability. The models were implemented in MATLAB. The stationary grid model itself was not part of this thesis, but it determined the boundary conditions of the models.

During modelling, care was taken to avoid iterative algorithms in order to ensure a fast simulation runtime. Still, the determination of the maximum power point (MPP) for the PV-array sub-model and for power limitation (pitching) and tip-speed-ratio control for the rotor sub-model required iterations. The modular design of the models enables future studies to simulate different configurations with a manageable set of sub-models. This has the advantage that only smaller parts have to be re-implemented. For example, a type A wind turbine (see Figure 2.4) could be realized by using the existing rotor model and implementing two new sub-models for asynchronous generator and for gearbox.

A type D wind turbine, consisting of a rotor, permanent magnet synchronous generator (PMSG), machine side converter (MSC), and grid side converter (GSC), was implemented as part of this work. During the process of the thesis, it turned out that the reactive power exchange between the PMSG and the grid could not be directly linked without implementing a model of the converter control. In order to keep the scope of the work reasonable, the reactive power exchange was excluded, and only the active power exchange was modelled. However, this has the disadvantage that only the real component of the current can be resolved. The PV model consists of the PV-array and the inverter. The PV-array model is the most complex one, as several steps are required before the input data can be used. The BESS model consists of the sub-models battery, inverter and rectifier. The battery sub-model is split into two parts: the electrical model and the battery aging model. The literature research conducted as part of this thesis showed that for both the electrical model and the battery aging model, only curve fitting approaches for certain battery types exist. Here, the equations for a polymer lithium ion battery are used for the electrical model. These should be replaced by formulas that describe a LiFePO4 based lithium battery, as these are typically used in stationary power applications. For battery aging, formulas for this type of battery are already implemented.

The models were validated with publicly available data sets and information from manufacturers. The PV array model was tested extensively with real measured values from an existing PV system located in Denmark. The tests included verifying the irradiation determination on a tilted surface for two different days, replicating the current-voltage and power-voltage characteristics for a monofacial and bifacial PV module and estimating the DC power output of the PV system. No comparative and freely accessible data of an existing

wind turbine could be found for validation, which is the reason why the test was limited to simulating the power curve. In general, the available data for generator and converter specifications is rare, which is the reason why values from the literature had to be used. In order to validate the wind turbine model in more detail, the relevant operating and component data of the PMSG, MSC, GSC, etc. must be acquired, e.g., by means of industrial cooperation. No suitable comparative data could be found for the BESS model. A suggestion for future work would be to test a LiFePO4 battery and implement the corresponding formulas.

The final PV and wind turbine models are able to simulate weather-dependent power generation. Overall, the tests carried out have shown that the models are suitable for a stationary grid simulation with a time resolution of one hour. If the time resolution of the grid simulation is increased in the future, dynamic phenomena must also be taken into account. This includes, for example, the transient behaviour of the battery voltage in the case of sudden load changes. Another example would be considering the mass inertia of the rotor shaft and blades, when simulating the effects of wind speed fluctuations in the range of seconds. Although a consideration of the dynamic effects only makes sense if the models are used in a dynamic grid model, it enables an even wider range of simulation possibilities.

Bibliography

[1] R. Zahoransky, *Energietechnik: Systeme zur konventionellen und erneuerbaren Energieumwandlung. Kompaktwissen für Studium und Beruf*, 9th ed. Wiesbaden: Springer Vieweg Wiesbaden, 2022.

[2] T. Ackermann, *Wind Power in Power Systems*, 2nd ed. Stockholm: John Wiley & Sons, Ltd, 2012. [Online]. Available: https://www.wiley.com/en-us/Wind+Power+in+Power+Systems2C+2nd+Edition-p-9780470974162

[3] A. Chaudhuri, R. Datta, M. P. Kumar, J. P. Davim, and S. Pramanik, "Energy conversion strategies for wind energy system: Electrical, mechanical and material aspects," *Materials (Basel, Switzerland)*, vol. 15, no. 3, p. 1232, 2022. [Online]. Available: https://doi.org/10.3390/ma15031232

[4] G. Reich and M. Reppich, *Regenerative Energietechnik: Überblick über ausgewählte Technologien zur nachhaltigen Energieversorgung*, 2nd ed. Wiesbaden: Springer Vieweg Wiesbaden, 2018.

[5] M. Carpintero-Rentería, D. Santos-Martin, A. Lent, and C. Ramos González, "Wind turbine power coefficient models based on neural networks and polynomial fitting," *IET Renewable Power Generation*, vol. 14, pp. 1841–1849, 2020. [Online]. Available: https://doi.org/10.1049/iet-rpg.2019.1162

[6] V. Yaramasu, A. Dekka, M. J. Durán, S. Kouro, and B. Wu, "Pmsg-based wind energy conversion systems: survey on power converters and controls," *IET Electric Power Applications*, vol. 11, no. 6, pp. 956–968, 2017. [Online]. Available: https://doi.org/10.1049/iet-epa.2016.0799

[7] L. Fernandez, C. Garcia, and F. Jurado, "Operating capability as a pq/pv node of a direct-drive wind turbine based on a permanent magnet synchronous generator," *Renewable Energy*, vol. 35, no. 6, pp. 1308–1318, 2010. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0960148109005369

[8] A. Binder, *Elektrische Maschinen und Antriebe: Grundlagen, Betriebsverhalten*, 2nd ed. Berlin: Springer Vieweg Berlin, 2018. [Online]. Available: https://doi.org/10.1007/978-3-662-53241-6

[9] J. A. Duffie, *Solar Engineering of Thermal Processes*, 4th ed. Wisconsin-Madison: John Wiley & Sons, Ltd, 2013.

[10] K. Mertens, *Photovoltaik: Lehrbuch zu Grundlagen, Technologie und Praxis*, 2nd ed. München: Carl Hanser Verlag GmbH & Co. KG, 2013.

[11] Electrical4U, "Working principle of solar cell," 2024, accesed: 2024-06-28. [Online]. Available: https://www.electrical4u.com/solar-cell/#Working-Principle-of-Solar-Cell

[12] A. Dolara, S. Leva, and G. Manzolini, "Comparison of different physical models for pv power output prediction," *Solar Energy*, vol. 119, pp. 83–99, 2015. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0038092X15003254

[13] S. Petrovic and S. Petrovic, *Battery Technology Crash Course*, 1st ed. Springer International Publishing, 2021.

[14] A. Barré, B. Deguilhem, S. Grolleau, M. Gérard, F. Suard, and D. Riu, "A review on lithium-ion battery ageing mechanisms and estimations for automotive applications," *Journal of Power Sources*, vol. 241, pp. 680–689, 2013. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0378775313008185

[15] M. Chen and G. Rincon-Mora, "Accurate electrical battery model capable of predicting runtime and i-v performance," *IEEE Transactions on Energy Conversion*, vol. 21, no. 2, pp. 504–511, 2006. [Online]. Available: https://doi.org/10.1109/TEC.2006.874229

[16] Y. Li and L. Chengxin, "Overview of maximum power point tracking control method for wind power generation system," *IOP Conference Series: Materials Science and Engineering*, vol. 428, no. 1, p. 012007, 2018. [Online]. Available: https://doi.org/10.1088/1757-899X/428/1/012007

[17] M. Nasiri, J. Milimonfared, and S. Fathi, "Modeling, analysis and comparison of tsr and otc methods for mppt and power smoothing in permanent magnet synchronous generator-based wind turbines," *Energy Conversion and Management*, vol. 86, pp. 892–900, 2014. [Online]. Available: https://doi.org/10.1016/j.enconman.2014.06.055

[18] The MathWorks, Inc., *MATLAB Optimization Toolbox User's Guide*, 2023, mATLAB R2023b. [Online]. Available: https://de.mathworks.com/help/releases/R2023b/pdf_doc/optim/optim.pdf

[19] A. L. R. James F. Manwell, Jon G. McGowan, *Wind Energy Explained: Theory, Design and Application*, 2nd ed. Massachusetts: John Wiley & Sons, Ltd, 2009. [Online]. Available: https://www.wiley.com/en-dk/Wind+Energy+Explained%3A+Theory%2C+Design+and+Application%2C+2nd+Edition-p-9780470015001

[20] Solcast, "Global solar irradiance data and pv system power output data," 2024, accessed: 2024-07-27. [Online]. Available: https://www.solcast.com

[21] T. Muneer, "Solar radiation model for europe," *Building Services Engineering Research and Technology*, vol. 11, no. 4, pp. 153–163, 1990. [Online]. Available: https://doi.org/10.1177/014362449001100405

[22] G. Forschungsstelle, I. für Energie und Verkehr, A. Gracia, and T. Huld, *Performance comparison of different models for the estimation of global irradiance on inclined surfaces : validation of the model implemented in PVGIS*. Publications Office, 2013.

[23] S. Vergura, "A complete and simplified datasheet-based model of pv cells in variable environmental conditions for circuit simulation," *Energies*, vol. 9, no. 5, 2016. [Online]. Available: https://www.mdpi.com/1996-1073/9/5/326

[24] N. Mohan, *Power Electronics: Converters, Applications, and Design, 3rd Edition*, 3rd ed. Minnesota-Minneapolis: John Wiley and Sons, 2002.

[25] H. Beltran, I. Tomás García, J. C. Alfonso-Gil, and E. Pérez, "Levelized cost of storage for li-ion batteries used in pv power plants for ramp-rate control," *IEEE Transactions on Energy Conversion*, vol. 34, no. 1, pp. 554–561, 2019. [Online]. Available: https://doi.org/10.1109/TEC.2019.2891851

[26] S. N. Motapon, E. Lachance, L.-A. Dessaint, and K. Al-Haddad, "A generic cycle life model for lithium-ion batteries based on fatigue theory and equivalent cycle counting," *IEEE Open Journal of the Industrial Electronics Society*, vol. 1, pp. 207–217, 2020. [Online]. Available: https://doi.org/10.1109/OJIES.2020.3015396

[27] Wind-turbine-models.com, "Wind turbine models," n.d., accessed: 2024-07-14. [Online]. Available: https://www.wind-turbine-models.com

[28] S. Thirumalai and S. Chenniappan, "Performance analysis of wind driven generators supplying an isolated load," *Journal of Computational and Theoretical Nanoscience*, vol. 14, 10 2017. [Online]. Available: https://doi.org/10.1166/jctn.2017.7050

[29] ABB, *ABB Wind Turbine Converters ACS880, 800 kW to 8 MW*, 2018-09-20, brochure. [Online]. Available: https://search.abb.com/library/Download.aspx?DocumentID=3AUA0000231755&LanguageCode=en&DocumentPartId=&Action=Launch

[30] Wind-turbine-models.com, "Lagerwey lw-72/2.0," n.d., accessed: 2024-07-14. [Online]. Available: https://www.wind-turbine-models.com/turbines/1234-lagerwey-lw-72

[31] ——, "Enercon e-40/5.40," n.d., accessed: 2024-07-14. [Online]. Available: https://www.wind-turbine-models.com/turbines/67-enercon-e-40-5.40

[32] M. Theristis, N. Riedel-Lyngskær, J. S. Stein, L. Deville, L. Micheli, A. Driesse, W. B. Hobbs, S. Ovaitt, R. Daxini, D. Barrie, M. Campanelli, H. Hodges, J. R. Ledesma, I. Lokhat, B. McCormick, B. Meng, B. Miller, R. Motta, E. Noirault, M. Parker, J. Polo, D. Powell, R. Moretón, M. Prilliman, S. Ransome, M. Schneider, B. Schnierer, B. Tian, F. Warner, R. Williams, B. Wittmer, and C. Zhao, "Blind photovoltaic modeling intercomparison: A multidimensional data analysis and lessons learned," *Progress in Photovoltaics*, vol. 31, no. 11, 7 2023. [Online]. Available: https://www.osti.gov/biblio/1995463

List of Figures

List of Tables

A. PC Specs

Hardware, OS and IDE used for running the code from Appendix C. Theoretically this should not influence the results.

- **Manufacturer:** ASUSTeK COMPUTER INC.
- **System Model:** X406UAR
- **System Type:** x64-based PC
- **Operating System:** Microsoft Windows 10 Home, Version 10.0.19045 Build 19045
- **CPU:** Intel(R) Core(TM) i5-8250U CPU @ 1.60GHz, 1800 MHz, 4 Core(s), 8 logical Processor(s)
- **GPU:** Intel(R) UHD Graphics 620
- **RAM:** DDR3, 8 GB, Channel: Dual, DRAM Frequency: 931.1 MHz
- **Hard Drive:** SanDisk SD8SN8U256G1002 238.47 GB
- **IDE:** MATLAB R2023b

B. Transformation BESS Discharge

$$P = V_{Str} \cdot I_{batt} \tag{2.1}$$

$$V_{Str} = V_c \cdot N_S = (V_{OC} - I_{Str} \cdot R_S) \cdot N_S \tag{2.2}$$

$$I_{Str} = I_{batt} \cdot N_P \tag{2.3}$$

Insert 2.2 and 2.3 in 2.1:

$$P = N_P N_S V_{OC} I_{Str} - N_P N_S I_{Str}^2 R_S \tag{2.4}$$

Transforming 2.4 to I_{Str}:

$$I_{Str}^2 - I_{Str} \cdot \frac{V_{OC}}{R_S} + \frac{P}{N_P N_S R_S} = 0 \tag{2.5}$$

$$I_{Str} = \frac{V_{OC}}{2R_S} - \sqrt{\left(\frac{V_{OC}}{2R_S}\right)^2 - \frac{P}{N_S N_P R_S}} \tag{2.6}$$

C. MATLAB Models

This section lists the MATLAB code of the implemented models. To ensure good readability, the comments have been reduced to a minimum. The sub-models are listed first, followed by the combined models.

Rotor

```
classdef Rotor < handle %Rotor Sub-Brix of Wind Turbine
    properties (SetAccess = public)
        rho (1,1) double                    % Air density [kg/m^3]
        v_w (1,:) double                    % Wind speed [m/s]
        r (1,1) double                      % Blade radius [m]
        v_cutin (1,1) double                % Cut-in wind speed [m/s
           ]
        v_nenn (1,1) double                 % Rated wind speed [m/s]
        v_cutout (1,1) double               % Cut-out wind speed [m/
           s]
        n_max (1,1) double                  % max. rotational speed
           [rpm]
        P_nenn (1,1) double                 % Rated power [W]
        P_mech (1,:) double                 % mechanical/aerodynamic
            Power [W]
        omega (1,:) double                  % angular velocity [rad/
           s]
        beta (1,:) int16                    % Pitch angle [degree]
        c_p (1,:) double                    % Power coefficient
    end

    methods
        %Constructor Rotor
        function obj = Rotor(r, rho, v_w, v_cutin, v_nenn, v_cutout,
            n_max)
            obj.r = r;
            obj.rho = rho;
            obj.v_w = v_w;
            obj.v_cutin = v_cutin;
            obj.v_nenn = v_nenn;
            obj.v_cutout = v_cutout;
            obj.n_max = n_max;
        end
        %Calculation of outputs
        function calculateP(obj)
            A = pi * obj.r^2;
```

```
        rated_cp = Opt_TSR(0, obj.v_nenn, obj);     %Calculate
           the RATED POWER
        obj.('P_nenn') = 0.5 * obj.rho * A * (obj.v_nenn^3) *
           rated_cp;
        for i = 1:numel(obj.v_w)
            if obj.v_w(i) < obj.v_cutin || obj.v_w(i) > obj.
               v_cutout
                obj.('P_mech')(i) = 0;
                obj.('omega')(i) = 0;
                obj.('beta')(i) = 0;
                obj.('c_p')(i) = 0;
            elseif obj.v_cutin <= obj.v_w(i) && obj.v_w(i) <=
               obj.v_nenn
                beta_TSR = 0;
                [max_cp, optimal_omega] = Opt_TSR(beta_TSR, obj.
                   v_w(i), obj);
                obj.('P_mech')(i) = 0.5 * obj.rho * A * (obj.v_w
                   (i)^3) * max_cp;
                obj.('omega')(i) = optimal_omega;
                obj.('beta')(i) = beta_TSR;
                obj.('c_p')(i) = max_cp;
            else
                omega_max = 2 * pi * obj.n_max / 60;
                beta_pitched = Pitch_Mode(obj.v_w(i), omega_max,
                    obj);
                cp = calculateCP(beta_pitched, omega_max, obj.
                   v_w(i), obj);
                obj.('P_mech')(i) = 0.5 * obj.rho * A * (obj.v_w
                   (i)^3) * cp;
                obj.('omega')(i) = 2*pi*obj.n_max / 60;
                obj.('beta')(i) = beta_pitched;
                obj.('c_p')(i) = cp;
            end
        end
    end

    %Function to calculate cp
    function c_p = calculateCP(beta, omega, v_w, obj)
        TSR = (omega * obj.r)/v_w;
        lambda_i = ((1/(TSR + 0.08 * beta)) - (0.035/(beta.^3 +
           1))).^(-1);
        c_p = 0.22 * (((116 / lambda_i) - 0.4 * beta - 5) .* exp
           (-12.5 / lambda_i));
    end

    %Function to calculate optimal TSR mode. Maximizes Cp by
       varying angular velocity
    function [max_cp, optimal_omega] = Opt_TSR(beta, v_w, obj)
    obj_func = @(x) -calculateCP(beta, x, v_w, obj);
```

```
        initial_guess = 0;
        lb = 0;
        ub = 2 * pi * obj.n_max / 60;
        [optimal_omega, max_cp] = fmincon(obj_func, initial_guess,
           [], [], [], [], lb, ub);
        max_cp = -max_cp;
        end

        % Function to calculate optimal Pitch mode
        function beta_pitched = Pitch_Mode(v_w, omega, obj)
            obj_func = @(x) 0.5 * (pi*obj.r^2) * obj.rho * (v_w^3) *
                calculateCP(x, omega, v_w, obj) - obj.P_nenn;
            initial_guess = 80;
            options = optimoptions('fsolve', 'TolFun', 10^-18, 'TolX
               ', 10^-18);
            beta_pitched = fsolve(obj_func, initial_guess, options);
        end
    end
end
```

PMSG

```
%PMSG Sub-Brix of Wind Turbine
classdef PMSG < handle
    properties (SetAccess = public)
        V_rated (1,1) double            %Rated Stator Voltage [V]
        I_rated (1,1) double            %Rated Current [A]
        p (1,1) double                   %Numper of pole pairs
        L_ds (1,1) double               %Dir. stator induct. [p.u.]
        L_qs (1,1) double               %Quad. stator induct. [p.u.]
        psi_pm (1,1) double             %Induced flux [p.u.]
        efficiency (1,1) double         %Efficiency of PMSG
        %Calculated Quantities
        i_ds double                %Direct Current [p.u.]
        i_qs double                %Quadrature Current [p.u.]
        u_ds double                %Direct Stator Voltage [p.u.]
        u_qs double                %Quadrature Stator Voltage [p.u.]
        P_s double                 %Stator Active Power [p.u.]
        Q_s double                 %Stator Reactive Power [p.u.]
    end

    methods
        %Constructor PMSG
        function obj = PMSG(V_rated, p, L_ds, L_qs, psi_pm,
           efficiency)
            obj.V_rated = V_rated;
            obj.p = p;
            obj.L_ds = L_ds;
            obj.L_qs = L_qs;
            obj.psi_pm = psi_pm;
            obj.efficiency = efficiency;
        end
        %Calculation of outputs
        function outputPMSG = calculatePMSG(obj, P_mech, P_nenn,
           omega_mech, n_max)
            obj.('I_rated') = P_nenn / obj.V_rated;
            if P_mech > 0
                %Transforming input in Per Unit System
                P_mech_pu = (obj.efficiency * P_mech) / P_nenn;
                omega_mech_nenn = 2*pi*n_max/60;
                omega_enenn = obj.p * omega_mech_nenn;
                omega_e = obj.p * omega_mech;
                omega_ep = omega_e / omega_enenn;
                omega_mp = omega_mech / omega_mech_nenn;
                T_w = P_mech_pu * omega_mp;
                T_e = T_w;
                quadratureCurrent = T_e / (omega_ep * obj.psi_pm);
                directVoltage = -obj.L_qs*quadratureCurrent*omega_ep
                   ;
```

```
                %MSC controls terminal voltage to rated voltage
                quadratureVoltage = sqrt(1^2-directVoltage^2 );
                directCurrent = (quadratureVoltage-obj.psi_pm)/(
                    omega_ep*obj.L_ds);
                realPower = directVoltage*directCurrent +
                    quadratureVoltage*quadratureCurrent;
                reactivePower = quadratureVoltage*directCurrent -
                    directVoltage*quadratureCurrent;
            else
                directVoltage = 0;
                quadratureVoltage = 0;
                directCurrent = 0;
                quadratureCurrent = 0;
                realPower = 0;
                reactivePower = 0;
            end
            outputPMSG = [directVoltage, quadratureVoltage,
                directCurrent, quadratureCurrent, realPower,
                reactivePower];
        end
    end
end
```

MSC

```
classdef MSC < handle   %MSC Sub-Brix of Wind Turbine
    properties (SetAccess = public)
        U_DC_rated (1,1) double           %Rated DC-link Voltage [V]
        efficiency (1,1) double     %Efficiency
        U_DC (1,:) double           %DC-link voltage [V]
        I_DC (1,:) double           %DC-link current [A]
        P_MSC (1,:) double          %P_msc [W]
    end

    methods
        %Constructor MSC
        function obj = MSC(U_DC_rated, efficiency)
            obj.U_DC_rated = U_DC_rated;
            obj.efficiency = efficiency;
        end
                %Calculation of outputs
        function outputMSC = calculateMSC(obj, P_s)
            P_msc = obj.efficiency * P_s;
            U_dc = obj.U_DC_rated;
            I_dc = P_msc / U_dc;
            outputMSC = [U_dc, I_dc, P_msc];
        end
    end
end
```

GSC

```
classdef GSC < handle   %GSC Sub-Brix of Wind Turbine
    properties (SetAccess = public)
        U_grid_ref (1,1) double          %Grid Voltage [V]
        efficiency (1,1) double      %Efficiency
        I_real (1,:) double          %DC-link current [A]
        U_grid (1,:) double          %Grid Voltage [V]
        P_GSC (1,:) double           %P_msc [W]
    end

    methods
        %Constructor GSC
        function obj = GSC(U_grid_ref, efficiency)
            obj.U_grid_ref = U_grid_ref;
            obj.efficiency = efficiency;
        end
                %Calculation of outputs
        function outputGSC = calculateGSC(obj, P_msc)
            P_gsc = obj.efficiency * P_msc;
            U_g = obj.U_grid_ref;
            I_r = P_gsc / real(3*U_g);
            outputGSC = [U_g, I_r, P_gsc];
        end
    end
end
```

PV Array

```
classdef PV_Array < handle      %PV-Array Sub-Brix of PV-System
    properties (SetAccess = public)
    I_ompp (1,1) double         %Maximum Power Current at STC [A]
    V_ompp (1,1) double         %Maximum Power Voltage at STC [V]
    I_osc (1,1) double          %Short Circuit Current at STC [A]
    V_ooc (1,1) double          %Open Circuit Voltage at STC [V]
    alpha_isc (1,1) double      %Temp. Coefficient of Isc [\%/C]
    alpha_voc (1,1) double      %Temp. Coefficient of Voc [%/C]
    NOCT (1,1) double           %Nominal operating temperature [C]
    Ns (1,1) double             %Modules in Series
    Np (1,1) double             %Modules Parallel
    I_Str (1,:) double          %String Current [A]
    V_Str (1,:) double          %String Voltage [V]
    P_Str (1,:) double          %String Power [W]
    datapath char               %Name of Dataset
    GHI (1,:) double            %Global Irrad. on a horiz. sur.[W/m
       ^2]
    DHI (1,:) double            %Diffuse Irrad. on a horiz. sur.[W/m
       ^2]
    BHI (1,:) double            %Beam Irrad. on a horiz. sur.[W/m^2]
    rho (1,:) double            %Surface albedo
    T_a (1,:) double            %Air temperature [C]
    ST_data (:,:) double        %Real Solar Time
    omega (1,:) double          %Hour angle [rad]
    delta (1,:) double          %Declination [rad]
    theta (1,:) double          %Angle of incident [rad]
    theta_z (1,:) double        %Zenith angle [rad]
    alpha_s (1,:) double        %Altitude angle [rad]
    gamma_s (1,:) double        %Solar azimuth angle [rad]
    lon (1,1) double            %Longitude [degree]
    lat (1,1) double            %Latitude [degree]
    lon_std (1,1) double        %Long. of Standard Meridian [degree]
    beta_m (1,1) double         %Slope of modules [degree]
    gamma_m (1,1) double        %Surface azimuth of modules [degree]
    G_bT (1,:) double           %Beam Irrad. on tilted sur. [W/m^2]
    G_rT (1,:) double           %Reflected Irrad. on tilted sur. [W/
       m^2]
    G_dT (1,:) double           %Diffuse Irrad. on tilted sur. [W/m
       ^2]
    G_T (1,:) double            %Total Irrad. on tilted sur. [W/m^2]
    G_on (1,:) double           %Extraterrestrial Radiation [W/m^2]
    G_pu (1,:) double           %Relative Irradiance
    R_sh (1,:) double           %Shunt Resistance [Ohm]
    R_s (1,:) double            %Series Resistance [Ohm]
    beta (1,:) double           %Beta Coefficient [A]
    gamma (1,:) double          %Gamma Coefficient [V^-1]
    I_mpp (1,:) double          %Maximum Power Current [A]
```

```
V_mpp (1,:) double          %Maximum Power Voltage [V]
P_mpp (1,:) double          %Maximum Power [W]
end

methods
    %Constructor PV
    function obj = PV_Array(datapath, lon, lat, lon_std, beta_m,
        gamma_m, Ns, Np, I_ompp, V_ompp, I_osc, V_ooc, alpha_isc
       , alpha_voc, NOCT)
        obj.datapath = datapath;
        obj.lon = lon;
        obj.lat = lat * pi / 180;
        obj.lon_std = lon_std;
        obj.beta_m = beta_m * pi / 180;
        obj.gamma_m = gamma_m * pi / 180;
        obj.Ns = Ns;
        obj.Np = Np;
        obj.I_ompp = I_ompp;
        obj.V_ompp = V_ompp;
        obj.I_osc = I_osc;
        obj.V_ooc = V_ooc;
        obj.alpha_isc = alpha_isc;
        obj.alpha_voc = alpha_voc;
        obj.NOCT = NOCT;
    end
    %Calculation of outputs
    function calculate(obj)
        Calculate_Sun(obj)
        Calculate_G_T(obj)
        for i = 1:numel(obj.G_pu)
            if obj.G_pu(i) > 0
                %Get Parameters
                [est_R_sh, est_R_s, est_beta, est_gamma] =
                   Estimate_Parameter(obj.G_pu(i), obj.T_a(i),
                   obj.G_T(i), obj);
                obj.('R_sh')(i) = est_R_sh;
                obj.('R_s')(i) = est_R_s;
                obj.('beta')(i) = est_beta;
                obj.('gamma')(i) = est_gamma;
                %Find Maximum Power Point
                [est_V_mpp, est_I_mpp, est_P_mpp] = Find_MPP(obj
                   .R_sh(i),obj.R_s(i), obj.beta(i), obj.gamma(i
                   ), obj.G_pu(i), obj.T_a(i), obj.G_T(i), obj);
                obj.('V_mpp')(i) = est_V_mpp;
                obj.('I_mpp')(i) = est_I_mpp;
                obj.('P_mpp')(i) = est_P_mpp;
            else
                obj.('V_mpp')(i) = 0;
                obj.('I_mpp')(i) = 0;
```

```
                obj.('P_mpp')(i) = 0;
                obj.('R_sh')(i) = 0;
                obj.('R_s')(i) = 0;
                obj.('beta')(i) = 0;
                obj.('gamma')(i) = 0;
            end
            obj.('I_Str')(i) = obj.I_mpp(i) * obj.Np;
            obj.('V_Str')(i) = obj.V_mpp(i) * obj.Ns;
            obj.('P_Str')(i) = obj.I_mpp(i) * obj.Np * obj.V_mpp
               (i) * obj.Ns;
        end
    end
    %Estimates the Modules Parameters
    function [R_sh, R_s, beta, gamma] = Estimate_Parameter(G_pu,
        T_a, G_T,obj)
        T_c = T_a + (G_T / 800) * (obj.NOCT -20);
        dT = T_c - 25;
        R_sh = (obj.V_ompp - obj.alpha_voc * dT * obj.V_ooc) / (
           G_pu * 0.5 * (obj.I_osc - obj.I_ompp));
        R_s = (0.25 * (obj.V_ooc - obj.V_ompp)) / (G_pu * (obj.
           I_ompp + obj.alpha_isc * dT * obj.I_osc));
        beta = G_pu * (obj.I_osc + obj.alpha_isc * dT * obj.
           I_osc) - ((obj.V_ooc - obj.alpha_voc * dT * obj.V_ooc
           ) / R_sh);
        zaehler = (obj.V_ompp - obj.alpha_voc * dT * obj.V_ooc)
           * G_pu * (obj.I_ompp + obj.alpha_isc * dT * obj.I_osc
           );
        nenner = (obj.V_ompp - obj.alpha_voc * dT * obj.V_ooc) *
            G_pu * (obj.I_ompp + obj.alpha_isc * dT * obj.I_osc)
            + (obj.V_ooc - obj.V_ompp) * G_pu * (obj.I_osc - obj
           .I_ompp);
        p = zaehler / nenner;
        top = obj.G_pu * (p * obj.I_osc - obj.I_ompp) - (1 - p)
           * obj.alpha_isc * dT * obj.I_osc - ((p * (obj.V_ompp
           - obj.alpha_voc * dT * obj.V_ooc)) / R_sh);
        bottom = p * (obj.G_pu * (obj.I_osc + obj.alpha_isc * dT
            * obj.I_osc) - ((obj.V_ooc - obj.alpha_voc * dT *
           obj.V_ooc)/ R_sh));
        gamma = (1 / (obj.V_ompp - obj.V_ooc)) * log(top /
           bottom);
    end

    function I = Current(R_sh, R_s, beta, gamma, G_pu, T_a, V,
       G_T, obj)
        T_c = T_a + (G_T / 800) * (obj.NOCT -20);
        dT = T_c -25;
        I = (G_pu * (obj.I_osc + obj.alpha_isc * dT * obj.I_osc)
            - beta * exp(gamma*(V + obj.alpha_voc * dT * obj.
           V_ooc - obj.V_ooc)) - (V / R_sh)) * ((1 + R_s/R_sh)
```

```
        .^(-1));
end
%Calculates MPP
function [V_mpp, I_mpp, P_mpp] = Find_MPP(R_sh, R_s, beta,
   gamma, G_pu, T_a, G_T, obj)
    obj_func = @(V) - V * Current(R_sh, R_s, beta, gamma,
       G_pu, T_a, V, G_T, obj);
    initial_guess = 0;
    lb = 0;
    ub = obj.V_ooc;
    [V_mpp, P_mpp] = fmincon(obj_func, initial_guess, [],
       [], [], [], lb, ub, []);
    P_mpp = -P_mpp;
    I_mpp = Current(R_sh, R_s, beta, gamma, G_pu, T_a, V_mpp
       , G_T, obj);
end
%Calculates Solar Time
function Calculate_Sun(obj)
    data = readtable(obj.datapath, 'Delimiter',',','
       VariableNamingRule','preserve');
    timestep = data.('period_end');
    obj.('GHI') = data.('ghi');
    obj.('DHI') = data.('dhi');
    obj.('BHI') = obj.GHI - obj.DHI;
    obj.('T_a') = data.('air_temp');
    obj.('rho') = data.('albedo');
    timestep_arr = datetime(timestep,'InputFormat','yyyy-MM-
       dd''T''HH:mm:ss''Z').';
    dateVector = datevec(datetime(timestep,'InputFormat','
       yyyy-MM-dd''T''HH:mm:ss''Z'));
    dateVector = [dateVector, day(datetime(timestep,'
       InputFormat','yyyy-MM-dd''T''HH:mm:ss''Z'),"dayofyear
       ")];
    ST = datetime();
    for i = 1:numel(timestep)
        B = (((360/365) * (dateVector(i,7) - 1)))*pi/180;
        EOT = 229.2 * (0.000075 + 0.001868*cos(B) -
           0.032077*sin(B) - 0.014615*cos(2*B) - 0.04089*sin
           (2*B));
        ST(i) = timestep_arr(i) + minutes(4 * (obj.lon - obj
           .lon_std) + EOT);
    end
    ST = ST.';
    obj.ST_data = [datevec(datetime(ST,'InputFormat','yyyy-
       MM-dd''T''HH:mm:ss''Z')), day(datetime(ST,'
       InputFormat','yyyy-MM-dd''T''HH:mm:ss''Z'),"dayofyear
       ")];
end
%Calculates the irradiance on tilted surface
```

```
function Calculate_G_T(obj)
    for i = 1:numel(obj.GHI)
        obj.('omega')(i) = (-180 + 15*obj.ST_data(i,4) +
           0.25*obj.ST_data(i,5) + 1/240 * obj.ST_data(i,6))
           *pi/180;
        obj.('delta')(i) = (23.45 * sin((360/365) * (284 +
           obj.ST_data(i,7)) *pi/180)) * pi/180;
        obj.('theta')(i) = acos(sin(obj.delta(i))*sin(obj.
           lat)*cos(obj.beta_m) - sin(obj.delta(i))*cos(obj.
           lat)*sin(obj.beta_m)*cos(obj.gamma_m) + cos(obj.
           delta(i))*cos(obj.lat)*cos(obj.beta_m)*cos(obj.
           omega(i)) ...
            + cos(obj.delta(i))*sin(obj.lat)*sin(obj.beta_m)
               *cos(obj.gamma_m)*cos(obj.omega(i)) + cos(obj
               .delta(i))*sin(obj.beta_m)*sin(obj.gamma_m)*
               sin(obj.omega(i)));
        obj.('theta_z')(i) = acos(cos(obj.lat)*cos(obj.delta
           (i))*cos(obj.omega(i)) + sin(obj.lat)*sin(obj.
           delta(i)));
        obj.('alpha_s')(i) = (pi/2) - obj.theta_z(i);
        obj.('gamma_s')(i) = sign(obj.omega(i)) * abs(acos((
           cos(obj.theta_z(i))*sin(obj.lat)-sin(obj.delta(i)
           )) / (sin(obj.theta_z(i))*cos(obj.lat))));
        if obj.alpha_s(i)   < 0
            obj.alpha_s(i) = 0;
            obj.gamma_s(i) = 0;
        end
        obj.('G_on')(i) = 1367 * (1 + 0.033*cos(360*obj.
           ST_data(i,7)/365));
        obj.('G_bT')(i) = obj.BHI(i) * abs((cos(obj.theta(i)
           ) / cos(obj.theta_z(i))));
        obj.('G_rT')(i) = 0.5 * obj.GHI(i) * obj.rho(i) * (1
            - cos(obj.beta_m));
        AM = abs(1/cos(obj.theta_z(i)));
        k_t = (obj.GHI(i) / obj.G_on(i)) / (0.1 + 1.031 *exp
           (-1.4/(0.9+9.4/AM)));
        if k_t < 0.3 || obj.beta_m == 0
            obj.('G_dT')(i) = obj.DHI(i) * ( (1+cos(obj.
               beta_m))/2 + 0.25227 * (sin(obj.beta_m) -obj.
               beta_m *cos(obj.beta_m) -pi*(sin(0.5*obj.
               beta_m)).^2 )  );
        else
            if obj.alpha_s(i) < 0.1
                obj.('G_dT')(i) = obj.DHI(i) * ( (1+cos(obj.
                   beta_m))/2 + (sin(obj.beta_m) - obj.
                   beta_m*cos(obj.beta_m) - pi*(sin(obj.
                   beta_m/2)).^2 ) * (0.00333 - 0.415*(obj.
                   BHI(i)/obj.G_on(i)) -0.6987*(obj.BHI(i)/
                   obj.G_on(i))^2)) *(1-obj.BHI(i)/obj.G_on(
```

```
                                i)) + ((obj.BHI(i)/obj.G_on(i))*((sin(obj
                                .beta_m)*cos(obj.gamma_m-obj.gamma_s(i)))
                                /(0.1-0.008*obj.alpha_s(i)))   );
                            else
                                obj.('G_dT')(i) = obj.DHI(i) * ( (1+cos(obj.
                                beta_m))/2 + (sin(obj.beta_m) - obj.
                                beta_m*cos(obj.beta_m) - pi*(sin(obj.
                                beta_m/2)).^2 ) * (0.00333 - 0.415*(obj.
                                BHI(i)/obj.G_on(i)) -0.6987*(obj.BHI(i)/
                                obj.G_on(i))^2)) *(1-obj.BHI(i)/obj.G_on(
                                i)) + (obj.BHI(i)*cos(obj.theta(i)))/(obj
                                .G_on(i)*cos(obj.theta_z(i))    );
                            end
                        end
                        obj.('G_T')(i) = obj.G_bT(i) + obj.G_rT(i) + obj.
                           G_dT(i);
                        obj.('G_pu')(i) = obj.G_T(i) / 1000;
                    end
                end
            end
        end
    end
```

Inverter

```
%Inverter Sub-Brix of PV and BESS
classdef Inverter < handle
    properties (SetAccess = public)
        R_0 (1,1) double            %Ohmic Resistance [Ohm]
        V_grid (1,1) double         %Desired Output Voltage [V]
        V_ph (1,:) double           %Line-to-Line AC Voltage [V]
        I_ph (1,:) double           %Current [A]
    end

    methods
        %Constructor of Inverter
        function obj = Inverter(V_grid, R_0)
            obj.R_0 = R_0;
            obj.V_grid = V_grid;
        end
        %Calculation of outputs
        function outputInverter = calculateInverter(obj, V_dc, I_dc)
            a = 3*obj.R_0;
            b = 3*obj.V_grid;
            c = -V_dc * I_dc;
            discriminant = b^2 - 4*a*c;
            I_0 = (-b + sqrt(discriminant)) / (2*a);
            outputInverter = I_0;
        end
    end
end
```

Rectifier

```
%Rectifier Sub-Brix of BESS
classdef Rectifier < handle
    properties
        V_grid (1,1) double      %Grid Line-to-line Voltage [V]
        R (1,1) double           %Ohmic Resistance [Ohm]
        V_dc (1,:) double        %DC Voltage [V]
        I_dc (1,:) double        %Direct Current [A]
        P (1,:) double            %Power [W]
    end

    methods
        %Constructor Rectifier
        function obj = Rectifier(V_grid ,R)
            obj.V_grid = V_grid;
            obj.R = R;
        end
        %Calculation of outputs
        function outputRectifier = calculateRectifier(obj, P)
            V_d = 3 * sqrt(2) / pi * obj.V_grid;
            direct_current = P /V_d;
            direct_voltage = V_d - obj.R * direct_current;
            outputRectifier = [direct_current, direct_voltage];
        end
    end
end
```

Battery

```
%Sub-Brix Battery of BESS
classdef Battery < handle
     properties
        %Battery Layout
        n_series (1,1) double           %Battery cells in series
        n_parallel (1,1) double         %Battery cells in parallel
        %Battery Values
        SOC (1,1) double                %State of Charge
        SOC_min (1,1) double            %Min. State of Charge
        SOC_max (1,1) double            %Max. State of Charge
        V_bat (1,:) double              %Battery Cell Voltage [V]
        I_bat (1,:) double              %Battery Cell Current [A]
        V_ges (1,:) double              %Total Battery Voltage [V]
        I_ges (1,:) double              %Total Battery Current [A]
        C_usable (1,:) double           %Usable Battery Capacity [Ah
           ]
        C_init (1,1) double             %Initial Battery Cap.[Ah]
        stored_energy (1,:) double      %Actual stored/released
           Energy [Wh]
        discharge_rate (1,1) double     %Discharge Rate
        charge_rate (1,1) double        %Charge Rate
        T (1,1) double                  %Temperature [C]
        t_month (1,1) double            %Age of BESS in months
        NC (1,1) double                 %Loading cycles
        dt (1,1) double                 %[1hr]
    end

    methods
        %Constructor Battery
        function obj = Battery(SOC, SOC_min, SOC_max, C_init,
           n_series, n_parallel, discharge_rate, charge_rate, T,
           t_month, NC, dt)
            obj.SOC = SOC;
            obj.SOC_min = SOC_min;
            obj.SOC_max = SOC_max;
            obj.C_init = C_init;
            obj.n_series = n_series;
            obj.n_parallel = n_parallel;
            obj.discharge_rate = discharge_rate;
            obj.charge_rate = charge_rate;
            obj.T = T + 273.15;
            obj.t_month = t_month;
            obj.NC = NC;
            obj.dt = dt;
        end
        %Calculation of outputs when charging
        function outputBESS = calculateBESS_charge(obj, current)
```

```
    [obj.('C_usable')] = Fade(obj);
    [obj.('SOC'), obj.('stored_energy'), obj.('V_bat'), obj
       .('I_bat')] = ChargeBattery(obj.SOC, current, obj.
       C_usable, obj);
    V_pack = obj.V_bat * obj.n_series;
    I_pack = obj.I_bat * obj.n_parallel;
    obj.('NC') = UpdateCycles(I_pack, obj.C_usable, obj);
    outputBESS = [V_pack, I_pack];
end
%Calculation of outputs when discharging
function outputBESS = calculateBESS_discharge(obj, power)
    [obj.('C_usable')] = Fade(obj);
    [obj.('SOC'), obj.('stored_energy'), obj.('V_bat'), obj
       .('I_bat')] = DischargeBattery(obj.SOC, power, obj.
       C_usable, obj);
    V_pack = obj.V_bat * obj.n_series;
    I_pack = obj.I_bat * obj.n_parallel;
    obj.('NC') = UpdateCycles(I_pack, obj.C_usable, obj);
    outputBESS = [V_pack, I_pack];
end
%Calculates capacity fade
function [C_usable] = Fade(obj)
            C_fadecal = ((3.087 * 10.^-7 ) * exp(0.05146 *
               obj.T) * obj.t_month.^0.5) / 100;
            C_fadecyc = ((6.87*10.^-5) * exp(0.027 * obj.T)
               * obj.NC.^0.5) / 100;
            CCF = 1 - (C_fadecal + C_fadecyc);
            C_usable = obj.C_init * CCF;
end
%Updates number of reference cycles
function NC = UpdateCycles(current, C_usable, obj)
    if current > 0
        new_cycles = current * obj.dt / C_usable;
        NC = obj.NC + new_cycles;
    else
        NC = obj.NC;
    end
end
%Calculates SOC, etc. when charging
function [SOC, energy_stored, V_m, I_m] = ChargeBattery(
   old_SOC, current, battery_capacity, obj)
    max_charge = battery_capacity * obj.charge_rate;
    actual_amount = min(max_charge, current*obj.dt);
    new_SOC = old_SOC + (actual_amount / battery_capacity);
    SOC = max(obj.SOC_min, min(obj.SOC_max, new_SOC));
    actual_current = (SOC - old_SOC) * battery_capacity /
       obj.dt;
    [V_begin, I_begin] = EquivalentCircuit_charge(old_SOC,
       actual_current, obj);
```

```
            [V_end, I_end] = EquivalentCircuit_charge(SOC,
               actual_current, obj);
            V_m = (V_begin + V_end) / 2;
            I_m = (I_begin + I_end) / 2;
            energy_stored = V_m*I_m * obj.dt;
        end
        %EC when charging
        function [V_batt, I_batt] = EquivalentCircuit_charge(SOC,
           current, obj)
            V_OC = -1.031*exp(-35*SOC) + 3.685 + 0.2156*SOC -
               0.1178*SOC.^2 + 0.3201*SOC.^3;
            R_series = 0.1562*exp(-24.37*SOC) + 0.07446;
            current_per_string = - current / obj.n_parallel;
            I_batt = current_per_string;
            V_batt = V_OC - R_series * current_per_string;
        end
        %Calculates SOC, etc. when discharging
        function [SOC, energy_stored, V_m, I_m] = DischargeBattery(
           old_SOC, power, battery_capacity, obj)
            max_discharge = battery_capacity * obj.discharge_rate;
            [V_begin, I_begin] = EquivalentCircuit_discharge(old_SOC
               , power, obj);
            actual_amount = max(I_begin*obj.dt,-max_discharge);
            new_SOC = old_SOC + (actual_amount / battery_capacity);
            SOC = max(obj.SOC_min, min(obj.SOC_max, new_SOC));
            [V_act] = EquivalentCircuit_discharge(old_SOC, power,
               obj);
            actual_output_power = (SOC - old_SOC) * battery_capacity
               *((V_begin + V_act )/2)*obj.n_series/obj.dt;
            [V_begin, I_begin] = EquivalentCircuit_discharge(old_SOC
               , actual_output_power, obj);
            [V_end, I_end] = EquivalentCircuit_discharge(SOC,
               actual_output_power, obj);
            V_m = (V_begin + V_end) / 2;
            I_m = (I_begin + I_end) / 2;
            energy_stored = V_m*I_m * obj.dt;
        end
        %EC when discharging
        function [V_batt, I_batt] = EquivalentCircuit_discharge(SOC,
            power, obj)
            V_OC = -1.031*exp(-35*SOC) + 3.685 + 0.2156*SOC -
               0.1178*SOC.^2 + 0.3201*SOC.^3;
            R_series = 0.1562*exp(-24.37*SOC) + 0.07446;
            I_batt = (V_OC/(2*R_series)) - sqrt((V_OC/(2*R_series))
               ^2 - (power/(obj.n_series*obj.n_parallel*R_series)));
            V_batt = V_OC - R_series*I_batt;
        end
    end
end
```

Wind Turbine

```
classdef WindTurbine < handle
    %Initialize Rotor, PMSG, MSC and GSC
    properties
        rotor = Rotor(20.15,1.225,xlsread('testwind.xlsx','B:B')
           ,2.5,12,25,38);
        pmsg = PMSG(4000, 32, 0.75, 0.75, 1.16, 0.97);
        msc = MSC(800, 0.985);
        gsc = GSC(20000, 0.985);
    end

    methods
        %Calculation of overall output:
        function outputValues(obj)
            calculateP(obj.rotor); %Rotor output
            %PMSG, MSC and GSC output:
            for k = 1:numel(obj.rotor.v_w)
                outputPMSG = calculatePMSG(obj.pmsg, obj.rotor.
                   P_mech(k), obj.rotor.P_nenn, obj.rotor.omega(k),
                   obj.rotor.n_max);
                obj.pmsg.u_ds(k) = outputPMSG(1);
                obj.pmsg.u_qs(k) = outputPMSG(2);
                obj.pmsg.i_ds(k) = outputPMSG(3);
                obj.pmsg.i_qs(k) = outputPMSG(4);
                obj.pmsg.P_s(k) = outputPMSG(5);
                obj.pmsg.Q_s(k) = outputPMSG(6);
                outputMSC = calculateMSC(obj.msc, obj.pmsg.P_s(k)*
                   obj.rotor.P_nenn);
                obj.msc.U_DC(k) = outputMSC(1);
                obj.msc.I_DC(k) = outputMSC(2);
                obj.msc.P_MSC(k) = outputMSC(3);
                outputGSC = calculateGSC(obj.gsc,obj.msc.P_MSC(k));
                obj.gsc.U_grid(k) = outputGSC(1);
                obj.gsc.I_real(k) = outputGSC(2);
                obj.gsc.P_GSC(k) = outputGSC(3);
            end
        end
    end
end
```

PV System

```
classdef PV_System < handle
    %Initialize PV-Array and Inverter
    properties
        pv_string = PV_Array('PV_data20032024.csv'
           ,16,48,15,30,0,10,1,4.87,36.8,5.25,45,0.001,0.0037,48);
        inverter = Inverter(230, 0.1);
        V_out (1,:) double           %AC Voltage Line to line [V]
        I_out (1,:) double           %AC Current [A]
        P_out (1,:) double           %Real Power Output [W]
    end

    methods
        %Calculate overall output:
        function outputValues(obj)
            calculate(obj.pv_string);   %PV-Array output
            %Inverter output
            for i = 1:numel(obj.pv_string.V_Str)
                if obj.pv_string.V_Str(i) == 0
                    obj.inverter.V_ph(i) = 0;
                    obj.inverter.I_ph(i) = 0;
                    obj.('P_out')(i) = 0;
                else
                    outputInverter = calculateInverter(obj.inverter,
                        obj.pv_string.V_Str(i), obj.pv_string.I_Str(
                       i));
                    obj.inverter.V_ph(i) = obj.inverter.V_grid;
                    obj.inverter.I_ph(i) = outputInverter;
                    obj.('P_out')(i) = 3 * obj.inverter.V_ph(i) *
                       obj.inverter.I_ph(i);
                end
            end
            obj.('V_out') = obj.inverter.V_ph;
            obj.('I_out') = obj.inverter.I_ph;
        end
    end
end
```

BESS

```
classdef BESS < handle
        %Initialize Inverter, Rectifier and Battery
    properties
        inverter = Inverter(230, 0.1);
        rectifier = Rectifier(400, 0.1);
        bess = Battery(1,0.1,1,10,100,1,1,1,20,1,1000,1);
        V_out (1,:) double          %AC Voltage [V]
        I_out (1,:) double          %AC Current [A]
        P_out (1,:) double          %Real Power Output [W]
    end
    methods %Calculate overall output:
        function outputValues(obj, P)
                    if P >= 0   %Charging: -> Rectifier active
                        outputRectifier = calculateRectifier(obj.
                           rectifier,P);
                        obj.rectifier.V_dc = outputRectifier(2);
                        obj.rectifier.I_dc = outputRectifier(1);
                        obj.rectifier.P = outputRectifier(1)*
                           outputRectifier(2);
                        outputBESS = calculateBESS_charge(obj.bess,
                           obj.rectifier.I_dc);
                        obj.bess.V_ges = outputBESS(1);
                        obj.bess.I_ges = outputBESS(2);
                        obj.P_out = -outputBESS(1)*outputBESS(2);
                        obj.I_out = 0;
                        obj.V_out = 0;
                    else        %Discharging -> Inverter active
                        obj.rectifier.V_dc = 0;
                        obj.rectifier.I_dc = 0;
                        obj.rectifier.P = 0;
                        outputBESS = calculateBESS_discharge(obj.
                           bess, P);
                        obj.bess.V_ges = outputBESS(1);
                        obj.bess.I_ges = outputBESS(2);
                        outputInverter = calculateInverter(obj.
                           inverter,obj.bess.V_ges,abs(obj.bess.
                           I_ges));
                        obj.inverter.V_ph = obj.inverter.V_grid;
                        obj.inverter.I_ph = outputInverter;
                        obj.P_out = real(-3*outputInverter*obj.
                           inverter.V_ph.');
                        obj.I_out = outputInverter;
                        obj.V_out = obj.inverter.V_grid;
                    end
        end
    end
end
```