Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Electric Power Engineering

Martin Wirak Onsrud

Artificial Neural Network Based Power Management in Microgrid

Master's thesis in Energy and the Environment
Supervisor: Mohammad Amin
June 2022

NTNU
Norwegian University of
Science and Technology

Martin Wirak Onsrud

Artificial Neural Network Based Power Management in Microgrid

Master's thesis in Energy and the Environment
Supervisor: Mohammad Amin
June 2022

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Electric Power Engineering

NTNU
Norwegian University of
Science and Technology

Preface

This thesis is the final product of my time in the Energy and Environmental Engineering Masters of Science programme at NTNU (*The Norwegian University of Science and Technology*). The project is a continuation of the work done in the specialization project preceding this and accounts for 30 credits.

The completion of such a project has been a great challenge, with several difficulties and obstacles arising during the semester. During the process of working on this project, I have learned an immense amount while getting an enhanced interest in the field of AI-implementation in our power grid.

I would like to express my appreciation and gratitude to Associate Professor Mohammad Amin for his help and guidance through the past year of work. In addition to this I would like to thank Postdoctorate Fellow Yusuf Shankar Gupta for his contribution and help with the simulations.

Several other people have contributed to the completion of this thesis, thus deserving recognition. A big thanks to my classmates for providing both social and academic input, and my family and Julie for their continuous support.

Trondheim, 03.06.2022

Abstract

Climate change, and the resulting focus on the green transition is rapidly changing the structure and characteristics of the power systems around the world. The implementation of microgrids are facilitating renewable power production in the power system, by allowing for smaller components of local production and storage such as solar panels and batteries. These microgrids need intelligent control schemes in order to regulate the power, frequency, voltage and currents within the system. Artificial neural networks (ANNs) are proposed as one option for microgrid control with the use of machine learning.

The objective of this thesis is to develop and test a simulation model of a hybrid microgrid with an artificial neural network based centralised controller, and compare the performance to a more traditional power management based power flow algorithm. This was to be done with the overarching goal being the assessment and identification of future possibilities as well as challenges around the use of ANNs in microgrid controls.

The research started with the development of a Simulink model of a microgrid system consisting of solar panels, a battery, an electric vehicle, constant and variable loads. The power flow algorithm was produced, and the microgrid was simulated with a base case consisting of standardised solar and load curves. In parallel, an ANN was developed with the results from the simulation of the base case being used for the training. The two control systems were simulated for three cases each: a base case, a case with irregular irradiance and a case with irregular load.

The results indicated that when provided the same previously unseen input, the ANN based control system managed to adjust the output values towards a more optimal solution compared to the power flow algorithm. The complex structure of the ANN creates and identifies its own patterns that is able to provide expected output values even if the situation is different from the training data. However, as this project only tested a few cases, the ultimate usability of ANN as a centralised controller cannot be concluded. Nevertheless, the result indicate that this may be a viable option for a more secure and effective control system for power management in the future.

Sammendrag

Klimaforandringer, og det resulterende fokuset på fornybar energi forandrer stadig strukturen og karakteristikken til kraftsystem rundt om i verden. Implementeringen av mikronett fasiliterer for fornybar energi ved å tilrettelegge for lokal energiproduksjon og lagring med blant annet solceller og batterier. Disse mikronettene krever intelligente kontrollsystemer for å kunne regulere kraftflyt, frekvens, spenning og strøm i systemet. Kunstige nevrale nett er foreslått som en mulighet for mikronett kontroll ved bruk av maskin læring.

Målet med denne oppgaven er å utvikle og teste en simuleringsmodell av et hybrid mikronett med et kunstig nevralt nett-basert sentralisert kontrollsystem, og sammenlikne den med en mer tradisjonell kraftflyt-basert algoritme. Dette blir gjort med det overværende målet å vurdere og analysere systemet for å identifisere både fordeler og utfordringer rundt bruken av nevrale nett for mikronett kontroll.

Prosjektet startet med utviklingen av en Simulink-modell av et mikronett bestående av solceller, et batteri og en elektrisk bil i tillegg til variabel og konstant last. En kraftflytsalgoritme ble laget, og mokronettet ble simulert med et basistilfelle som bestod av normale sol- og lastforhold. Parallelt ble det utviklet et nevralt nett der resultatene fra simuleringen av basistilfelle ble brukt til opplæringen. De to kontrollsystemene ble simulert for tre tilfeller hver: et basistilfelle, et tilfelle med uregelmessig innstråling og et tilfelle med uregelmessig belastning.

Resultatene indikerte at når gitt den samme tidligere usynlige input dataen, klarte det nevrale nett-baserte kontrollsystemet å justere utgangsverdiene mot en mer optimal løsning sammenlignet med kraftflytsalgoritmen. Den komplekse strukturen til det nevrale nettet skaper og identifiserer sine egne mønstre som er i stand til å gi forventede utgangsverdier selv om situasjonen er forskjellig fra treningsdataene. Siden dette prosjektet bare testet noen få tilfeller, kan den endelige brukbarheten av nevrale nett som et sentralisert kontrollsystem ikke konkluderes. Resultatet tyder likevel på at dette kan være et brukbart alternativ for et sikrere og mer effektivt kontrollsystem for kraftstyring i fremtiden.

Contents

List of Figures

List of Tables

List of Abbreviations

AI Artificial Intelligence

ANN Artificial Neural Network

BDC Bidirectional Converter

BESS Battery Energy Storage System

CSI Current Source Inverter

ESS Energy Storage Systems

EV Electric Vehicle

HV High Voltage

IGBT Insulated-Gate Bipolar Transistor

INC Incremental Conductance

LV Low Voltage

MPP Maximum Power Point

MPPT Maximum Power Point Tracking

NBDC Non-Isolated bidirectional Converter

PWM Pulse Width Modulation

ReLu Rectified Linear Unit

RES Renewable Energy Sources

RMSE Root Mean Squared Error

VSI Voltage Source Inverter

VSC Voltage Source Converter

1 Introduction

1.1 Evolution of the Power System

The world is in constant change, with new challenges every year affecting the way the worlds population live their everyday lives. The largest and most demanding challenge of recent time has been climate change and how the power demand can be supplied by renewable energy sources in order to reduce emissions. Figure 1.1 shows how the development of renewable energy sources (RES) has been for the last 10 years in addition to how large the share of coal- and renewable power production has developed as a percentage of total power production in the world.[1]

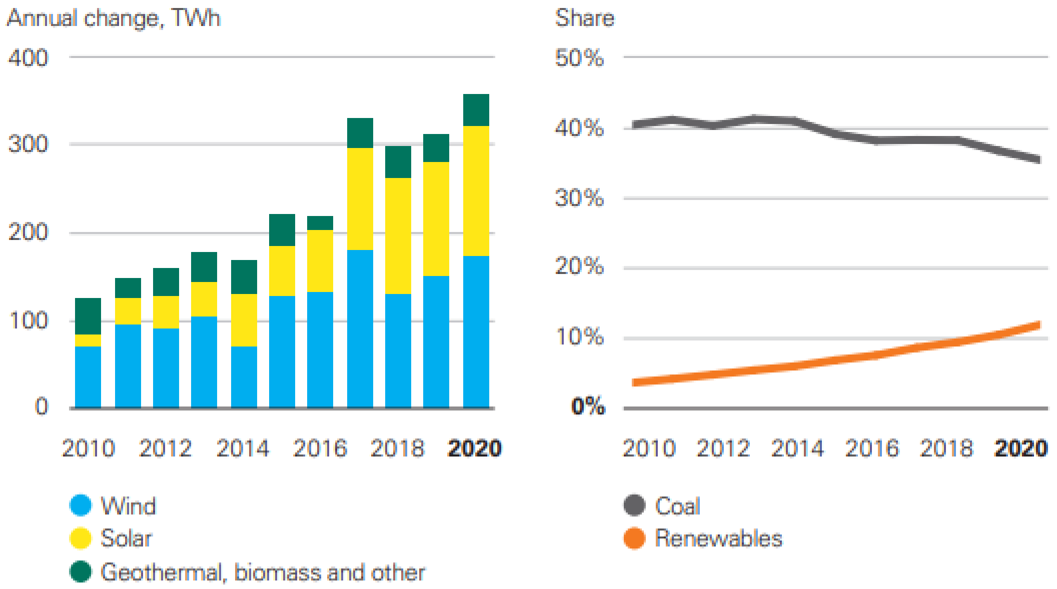


Figure 1.1: Renewable power development over the last 10 years.[1]

In contrast with the large centralised non-renewable power production of old, solar-, wind- and hydropower is dependent on nature in order to produce power. These natural resources are not always available on demand compared to the fossil fuel power plants. Although hydropower can be regulated quite easily, it is still dependent on sufficient water in the reservoirs which with more extreme weather trends is not always a guarantee. As more RES are implemented in the power system, a larger focus on proper storage and regulation is needed in order to maintain a stable power system with sufficient power to feed the demand at all times.[1, 2]

In addition to the challenges associated with the increase of RES in the power system is the fact that most of these also are distributed energy resources (DERs). The most common placement of a solar panel is on the roof of buildings, providing power directly to the local area. This results in decentralised power production, which at the rate of implementation is shifting the dynamic of the power system as a whole at a rapid pace. Figure 1.2 shows an example of how these DERs can be connected, and how a centralised controller is connected to all the elements.[1, 2]

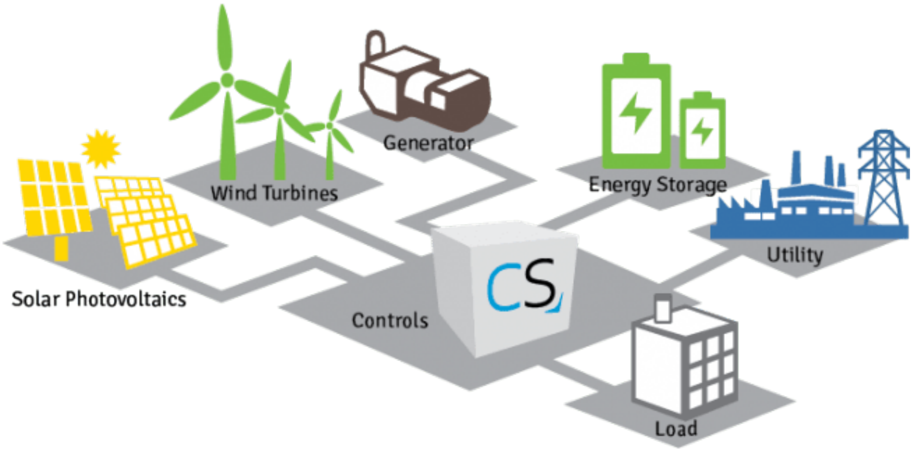


Figure 1.2: Commonly used microgrid components.[3]

As a the power system dynamic has been changing, microgrids has risen in popularity. As the microgrids often are able to operate either by itself or connected to a main grid, the system needs to be able to regulate the internal power flow to provide stability in the system. Achieving such a balance can be a challenge, especially with intermittent power sources, and warrants the need for a centralised controller uses measured values from the system, processes them and provides an output for the regulation of the system. If proper stability can be achieved, the implementation of microgrids can provide energy security from central system faults, lower electricity prices and integration of sustainable energy production.[4, 5, 6]

Over the past 5-10 years, artificial intelligence (AI) and machine learning has been taken to new heights, with a rapid rate of research and development. While artificial neural networks have existed for over 60 years, the research, development and applications of such system have advanced significantly in recent years. The use of different types of AI allows for processing complicated calculations with the use of the high computational power of the computers available. Figure 1.3 shows how a small ANN with three inputs, two hidden layers and one output could look like.[6, 7]

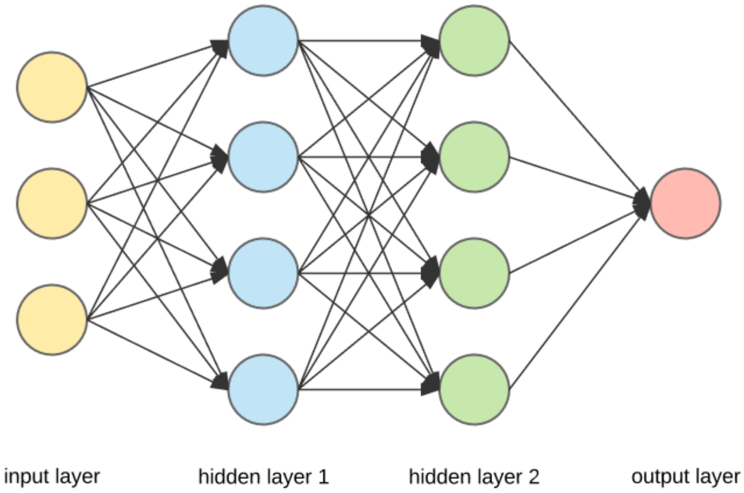


Figure 1.3: The structure of a basic ANN.[8]

The ANN uses a complex network of simple processing units which are trained by a database of training data. The structure of a neural network is quite different to traditional calculation methods, by having many simple calculations in a mesh of nodes. The structure of the neural network in addition to its learning process makes the path between input and output significantly different to traditional calculations. These factors make ANNs quite interesting and may result in a method of calculation that can be more reliable in a wide spectrum of calculations. Examples of this may be situations with corrupt or missing data, in addition to values from unforeseen circumstances that the programmers may not have taken into consideration.[6, 7]

1.2 Motivation and Purpose

With the increasing complexity of the power system, the need to look for ways to reinvent or improve the performance of the control systems becomes apparent. As there is a rapid rate of research and interest in AI-technology and how it can be used, this becomes a point of interest when looking at some of the challenges the future power system will encounter. If traditional methods of control can be replaced by various types of AI systems, the transition from a centralised grid structure to a greener, more sustainable decentralised structure may transpire at a faster rate. With climate change as the main driving factor, any technological advancements that may help drive this green transition forward should be given extensive attention.

On top of facilitating the implementation of renewable energy sources, the AI based control system can address several other challenges within the current system. The possibility of producing usable results with missing, or vastly different input data than expected can increase the resilience of cyber-attacks, natural disasters or poor signals which may lead to loss of power in the present system. The use of microgrids in general can also allow for houses, neighbourhoods or whole cities to become largely self-sufficient, reducing the dependency on transfer of power over large distances. This may reduce the need for developing larger power transfer capacities across countries, as more power can be produced locally.

These future microgrids needs to be able to adjust local production and storage in order to maintain the microgrids stability when disconnected from the grid, while also seamlessly connecting to the grid and contributing to the stability of the main grid when needed. Properly researched, stable, predictable and reliable local control system will be imperative for the wide implementation of such microgrids. Even with the promising characteristics of ANNs, a substantial amount of research and testing is needed before such systems can be implemented, highlighting the motivation for this thesis.

1.3 Project Description and Objectives

This masters thesis is the result of the work done in the final semester of the Energy and Environmental Engineering degree, and serves as a project going more deeply into one specific research area. The thesis is a continuation of the specialization project completed in the autumn of 2021, where more in depth theory and state of the art research was done as a preparation for this thesis. In addition to this, a thesis with similar scope from 2021 was used as a basis for this project with the goal of improving on the existing work and moving the research on this area further.

The overall goal of this thesis can be defined as the following:

Developing and testing the performance of a hybrid microgrid system with an Artificial Neural Network based centralised power control

Several course objectives were identified in order to achieve the goal of the thesis:

- Obtain the required theoretical knowledge to develop a model of a microgrid with the needed power electronic components in order to carry out simulations with different centralised controllers.
- Develop a power flow algorithm as a centralised controller in order to obtain training data for an artificial neural network.
- Develop, train and test an ANN to function as the centralised controller of the microgrid, and compare the performance of the two controllers with different scenarios.

By completing these objectives, this thesis hopes to contribute to the research field of AI in future power systems. With simulations of the general concept of ANN as a centralised controller, positive trends and results can be identified, while challenges and problems that needs to be addressed in future work can be determined.

1.4 Related Research

Previous research on the subject of ANN based power management control has been completed, with varying areas of focus. The general concept and applications of AI-techniques in smart grids and renewable energy systems were outlined in [9], including ANNs, fuzzy logic, expert systems and genetic algorithm.

In [10], a simple hybrid microgrid centralised controller was created and tested. The thesis concluded with the system partially working, but with poor power quality and several proposed improvements. An ANN based neural network control of a standalone DC microgrid was researched in [11], where the proposed controller was able to maintain the voltage stability, and manage the power sharing of the DC microgrid. [12] has reviewed DC microgrid control with ANN and a fuzzy logic controller, resulting in successful voltage regulation of the simulated system.

Other parts of the microgrid system have been assessed with regards to areas of use for Artificial Intelligence. One example of this is study on solar power forecasting in [13] where the ANN was used to mitigate the uncertainty of solar power production. Lastly, [14] studied frequency control in microgrid communities, using the MATLAB's Deep Learning Toolbox$^{\text{TM}}$ to develop a feed forward neural network which was successfully used for determining the optimal PID controller parameters.

1.5 Structure of Thesis

This thesis is separated into the five following chapters:

Chapter one describes the general introduction of the thesis, project description and objectives. Additionally, the motivation and purpose of the project is included with a brief review of existing literature on the subject.

Chapter two gives an overview of a general microgrid. This includes background and theory of the configuration of a microgrid along with the most used power electronic components and their controls. In addition to this, the simulation model and simulation cases used in this thesis are presented.

Chapter three explains the development of the power flow algorithm needed for acquiring training data for the ANN including all assumptions and specifications serving as the basis for the algorithm. Furthermore, the end of the chapter contains the results from simulating the microgrid with the power flow algorithm as the centralised controller with a brief discussion on the controllers performance.

Chapter four gives the background and theory on artificial neural networks, in addition to explaining how the nerual net used in this thesis was developed. The simulation results from simulating the microgrid with the ANN as the centralised controller is included in this chapter, along with a discussion of the results in comparison to the simulations in chapter three.

Chapter five includes the concluding remarks from the work depicted in this thesis accompanied with reflections on possible further work.

The state of art and related work were reviewed in the project preceding this thesis [6]. As the identification of relevant background material was carried out last semester, there are areas where no new relevant material was found during the work on this thesis. Chapter 2.1 and 2.2 is presented as presented in the previous project. As this thesis is a continuation of the previous project, some sections may contain parts of the previous project along with amendments and additions based on the continued work. These will be cited appropriately.

2 Microgrid Systems

There are several aspects of theory that needs to be taken into consideration when constructing, controlling and simulating a microgrid. To properly get an understanding of what the challenges are, a comprehensive explanation of the system topology, with energy production, storage and loads, is needed. In addition to this, an overview of which power electronic components are needed and how this all can be controlled is required. The following chapter will explain these areas of research and give a wider background for the problems tackled in this thesis.

2.1 System Topology

A microgrid is a combination of distributed energy resources, energy storage systems (ESS) and a variety of loads. There are several ways to set up a microgrid, and there can be large variations in which components it consists of. Some of the most common components in microgrids are battery energy storage systems (BESS), solar panels, wind turbines, hydrogen production and electrolysis in addition to a large variety of constant and variable loads. The use of components that previously only have been considered loads as flexible energy sources is also an area being researched and developed. One example of this is an electric vehicle (EV) which can function as both a load and a storage system. A microgrid can operate by itself (Island mode) or connected to a power grid (Grid connected mode). The goal of a microgrid is to internally regulate the power flow and deliver energy in a smart, sustainable and intelligent way with the use of intelligent monitoring and control, while functioning as a single component with regards to the main grid.[15, 16]

2.1.1 Types of Microgrid

The power structure of a microgrid can be set up in several different ways. A system with only DC power systems can work in a DC microgrid. This can also be done with AC systems in an AC power grid. The most common solution is a hybrid AC/DC microgrid which can have loads, storage and production resources of both DC- and AC-type. The hybrid microgrid can have a wide variety of configurations, but is usually consisting of one DC subgrid and one AC subgrid connected by an inverter. Illustrations of how the different types of microgrid can be set up is shown in Figure 2.1, 2.2 and 2.3. A hybrid microgrid system will often require a smaller amount of converters which can allow for a cheaper and less complex system.[15]

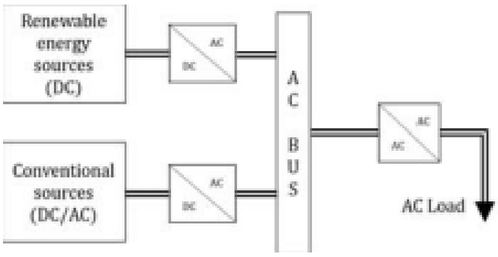


Figure 2.1: AC microgrid.[15]

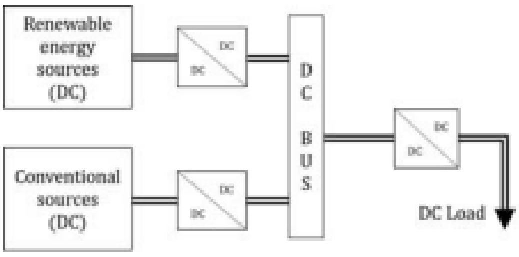


Figure 2.2: DC microgrid.[15]

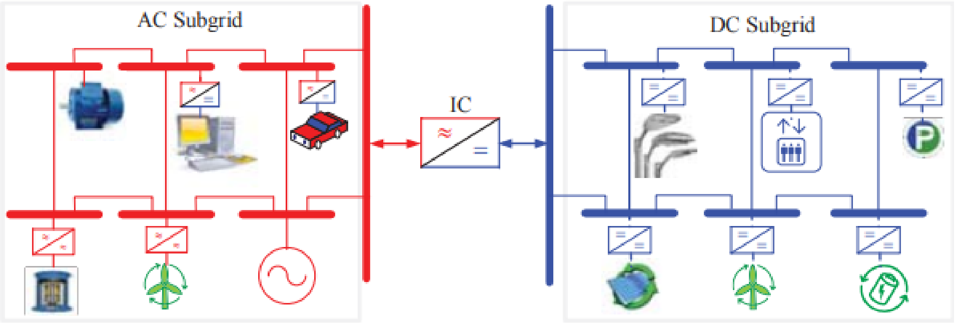


Figure 2.3: A hybrid microgrid.[17]

2.1.2 Island Mode

When a microgrid is operating in island mode, the system has to fully provide the power needed by the loads at all times while regulating the frequency and voltage levels which can be challenging at times. Having enough storage capacity to guarantee power supply when power production is reduced is one of the challenges of an islanded system. Modern microgrids often consist of renewable and variable power sources, which can challenge the stability of the system. An islanded microgrid can be both a standalone system in a remote area with no option of connecting to a main grid, or a system that has the option of being connected to or disconnected from the main grid. A system that disconnects from the main grid may do so for several reasons spanning from economic gain, the desire for self sufficiency, planned maintenance, or poor power quality of the main grid. A microgrid can also be disconnected from the main grid due to unplanned faults in the system.[15, 16]

2.1.3 Grid Connected Mode

A grid connected microgrid is connected to the main power grid, but may operate in several different ways depending on the settings of the control system. A microgrid can either be set up to only receive additional power from the grid or to be able to sell excess power back to the grid. When connecting to the main grid, the microgrid needs to synchronise the frequency to the grid to be able to transfer power without technical difficulties. This can be done in several ways, like using a droop controller or artificial intelligence methods [18]. The power sharing inside the microgrid while connected to the main grid can also be varied based on different control system priorities.[9]

2.2 Energy Production, Storage and Loads

A microgrid usually consists of a combination of loads, production and storage, and this combination can vary largely from microgrid to microgrid. However, there are some components that often are included.

2.2.1 Distributed Energy Resources

The definition of distributed energy resources are power producing units located closer to the consumer. These are often smaller in scale compared to the power producing units traditionally connected to the main power grid. Renewable energy sources are being increasingly used around the world and are becoming an integral part of the microgrid systems in place today, and likely in the future.[4]

The most used energy resources in modern microgrids are solar panels, which have had a rapid price reduction in the past few decades[19]. Solar panels are easy to place on buildings where there usually is available space. Hydropower, wind power and hydrogen electrolysis are also often used.[4]

2.2.2 Storage Solutions

Storage solutions are often implemented in order to increase the power balance in a system. This is an important part of modern microgrids which usually have RES with intermittent and unpredictable power production. A storage system can store excess power when the demand is lower and contribute with power when there is a power need in the system. The most common storage system is a BESS, but many other technologies can also be used. Examples of these are production of hydrogen, kinetic, thermal and chemical storage.

The cost of the system is usually substantially increased by implementing storage devices. This highlights the importance of proper scaling of components to reduce costs as much as possible while still working as intended.[15]

2.2.3 Loads

Loads can vary greatly from microgrid to microgrid depending on what applications the power is needed for. These loads can be constant, variable or follow certain patterns and can be on both the AC and DC side. Having an understanding of what types of loads, what load profile they have and the scale of them is important to size a microgrid properly and to get the proper performance without oversizing components. The use of variable storage and load solutions is also possible. One example of this is an electric vehicle that usually charges and functions as a load, but can be utilised for discharging power if needed.[15]

2.3 Power Electronic Components

One essential demand seen with the increasing number of distributed power sources is the need for effective power electronic components. These components are needed to control voltage and frequency, ensuring stability in the grid. For a hybrid microgrid system, a combination of DC-DC converters and DC-AC inverters are needed and each one with internal control systems tuned to unify and stabilise the system. This subject was more extensively covered in the specialisation project as part of the research for the development of the microgrid, and is therefore a smaller part of this thesis.[6]

2.3.1 DC-DC Converters

In DC systems, the different components often have a substantial difference in rated voltage levels. With batteries, loads and local energy production, DC-DC converters make sure the voltage levels can be maintained at the desired levels. The most common types of DC-DC converters are boost-converters, buck-converters and bidirectional boost-buck converters. The boost converters increase the voltage level from a low value to a larger one, while the buck converter steps the voltage level down from a large value to a low value. The bidirectional boost-buck converters can be used both to step up and down the voltage level, and can also switch the current flow direction. The main applications of boost converters are in regulated DC power supplies, renewable energy sources and electric vehicles.[20, 21]

Each DC-DC converter needs a control system tuned to the desired function, which can vary greatly depending on the type of system. For a solar PV system with a boost converter, a fitting control system can be one using maximum power point tracking (MPPT) which regulates the voltage levels for optimising the power output. In addition to this there may be a need to control the voltage level of a system. This is especially important in an islanded microgrid where the voltage level is more volatile. In this case a voltage controlled DC-DC converter is appropriate. In other cases such as the connection of an electric vehicle, a current controlled DC-DC converter may be more fitting.[21]

2.3.2 DC-AC Inverters

When connecting a microgrid with DC elements to the main grid, there is a need for a DC-AC inverter. This inverter takes a DC signal and turns it into an AC signal. Depending on what type of DC supply is provided, the inverter can function as a current source inverter (CSI) or voltage source inverter (VSI). The way the inverter is built, and the number of bridges determines the output.[6, 20]

The inverter consists of a three bridges of IGBTs, which turn on and off based on a signal provided from a control system fed through a Pulse Width Modulator (PWM) which compares a desired sinusoidal signal with a triangular wave with frequency equal to the switching frequency.[6, 20]

In addition to the inverter and the inverter control, an LCL filter is needed to work as a low pass filter. This eliminates the harmonics from the output signal from the inverter and consists of a two inductors in series with the inverter on the AC side, and one capacitor bank in parallel between the two inverters.[6, 22]

2.4 Microgrid Control

With an increasing number of renewable energy systems, decentralised energy resources and intermittent production comes a new set of requirements for the control structure of the power system. At a certain point, the amount of smaller components in the power system will become large enough to be required to take part in the frequency and voltage regulation of the main grid as well as microgrids. As a result, many power inverters will be connected to the power grid with each requiring a decentralised control structure in order to properly regulate the power flow between renewable energy sources and the power grid.[23, 24]

There are several layers of control structure needed for a microgrid. Both DC-AC and DC-DC converters need local control systems ensuring the intended operation. The DC-DC converters are often tuned with a simple control scheme which is set up to provide a certain output current or voltage. They can also be configured to control power output or ensure a stable voltage level at the DC bus.[23, 24]

In general, an AC-coupled microgrid can have two main states that affect how the control system functions: grid connected mode and island mode. For a grid connected microgrid, the microgrid can function in dispatched power mode, and undispatched power mode. When working in dispatched output power mode, the microgrid behaves like a controllable power source or load. This can provide the grid with support or load management. In this operating mode, the voltage and frequency is provided by the grid, and the current is controlled to regulate the input or output power. When on undispatched power output mode, the storage systems are usually charged and the distributed energy resources are operated in maximum power point (MPP) mode.[6, 24]

When a microgrid is operated in island mode, the stability of the system must come from internal control schemes. Options for control of such a system includes the droop method, master-slave method and power balancing using a model predictive method. The droop method is the most common, and functions by emulating the behaviour of a synchronous generator. In a synchronous generator, the voltage and frequency output varies with the real and reactive power output.[6, 24]

2.4.1 Main Controller

A grid connected microgrid will always need a main controller in order to connect and disconnect from the grid based on the state of the production and demand of the local system. The main controller can both include automated processes in addition to implementing software allowing to manually control the system in maintenance cases. The main controller will always rely on some type of data from the power system components, and will therefore have some sort of communication system with different sensors in the microgrid. These sensors may be located far apart, leading to the system being exposed to faults in the communication lines from situations like natural causes and cyber attacks. As neural networks are trained black box models, they may not be as affected by a loss of signal as traditional systems which use power flow equations for microgrid control.

There are are also several other elements that can be included in a main controller. Any lower level control system that needs a situation-dependent reference value may also receive this from the main controller. Examples of uses for this can be to specify the power reference for a VSI with droop control, or the current reference for a buck-boost converter between an EV and a DC bus. The main controller may also have a variety of priorities, such as enabling and disabling storage components at certain states of charge, or times of day in addition to controlling the power flow both in the system and from one system to the grid.[25]

2.4.2 Self-synchronised Universal Droop Controller

The objective of a grid-connected inverter is to properly regulate the power flow between the two sides of the inverter which often is the local microgrid and the main grid. When operated in set mode, the inverter sends the desired amount of real and reactive power to the grid. The inverter can also be operated in droop mode, which adjusts the amount of exchanged real and reactive power based on the voltage and frequency of the grid. By being able to operate in set and droop mode for both real and reactive power, the inverter can contribute fully in both voltage and frequency regulation of the grid. This is an element which will be more important with the increased decentralisation of power production, ensuring the grid can remain within its operating limits even with more volatile local production.[18]

A synchronisation unit is required in order for an inverter to be grid connected. This is to ensure that the output voltage is equal to the grid voltage, or at least close enough to prevent large inrush currents when connecting to the grid. A widely used synchronisation component is the Phase Locked Loop (PLL). Unfortunately, having many PLLs in a system may lead to competition between the units, decreasing the overall performance. In addition to this, as the PLLs are non-linear, these systems often tend to be more complicated and can cause instability.[26]

In order to forego the use of a PLL, a new type of droop controller called the self-synchronised universal droop controller (SUDC) has been proposed in [23]. The SUDC uses the basic established structure of a universal droop controller, and also functions in the same way when in island mode. Figure 2.4 shows the structure of the SUDC.[23]

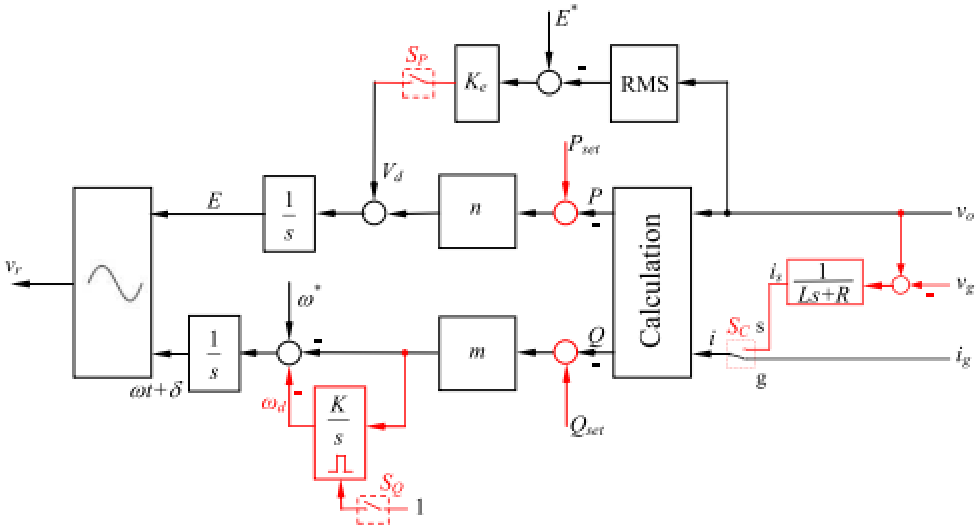


Figure 2.4: Block diagram of the proposed SUDC.[23]

Table 2.1 shows how the switches are set in the different operational modes. In the self-synchronisation mode, the terminal voltage is synchronised with the grid voltage. This is done by introducing a virtual impedance which generates a virtual current. The virtual current is related to the difference between the terminal voltage and grid voltage, and will be zero when the two voltages are equal.[23]

P-mode and Q-mode are set modes for the real and reactive power respectively. In these modes the real grid current is fed through the control system, and the desired power can be set. The voltage level will settle at a constant value in the steady state when P and P_{set} are equal, while the frequency settles at a constant value in the steady state resulting in $Q = Q_{set}$.[23]

P_D-mode and Q_D-mode are the droop modes for real and reactive power. These modes allow for the real and reactive power sent to the grid to be automatically adjusted based on the frequency and voltage of the grid. An additional resynchronisation mechanism has been proposed in [27], which reduces the amount of transient overcurrents when reconnecting to the grid after operating in island mode.[23]

Table 2.1: System operation modes of the SUDC.[23]

Mode	**Switch** S_C	**Switch** S_P	**Switch** S_Q
Self-sync mode	s	OFF	OFF
P-mode, Q-mode	g	OFF	OFF
P_D-mode, Q-mode	g	ON	OFF
P-mode, Q_D-mode	g	OFF	ON
P_D-mode, Q_D-mode	g	ON	ON

3 Simulation Model

As one of the main goals of this thesis was to test the control system of a microgrid, a simulation model was needed. Both MATLAB® and Simulink® has advanced functionality for simulating electric power systems and implementing custom code while having several neural network toolboxes. Initially, the ANN was to be developed with python and integrated in the Simulink® model. However, upon further research the Deep Learning Toolbox^{TM} for MATLAB® proved sufficient, which also allowed for a more streamlined simulation process.

In addition to including the most necessary elements of the microgrid such as the power production and storage, the choice of also including some smaller DC-DC converters and their controls was made. This was done for the added learning benefit of having more realistic components compared to making ”perfect” elements, while avoiding the creation of a system that only works in ideal conditions.

As this thesis is within the same scope of a thesis from 2021, many of the elements and their respective specifications have been continued from the previous work with a goal of easier continuation of the research subject with regards to this thesis, last years and any future research. However all specifications have been reconsidered carefully in order to assess if more optimal components and configurations were available.

3.1 Model Overview

The model consists of a hybrid AC-DC microgrid connected to the main grid. The microgrid contains one AC bus and one DC bus separated by an DC-AC inverter and a circuit breaker. The DC side of the microgrid includes a Solar PV array, a stationary battery and an electric vehicle. The solar panels is the main power provider of the microgrid, with the two batteries adding the flexibility of having both added loads or added production in the system. As the solar panels only produce power in the day, some sort of storage component is essential in order to increase the self sufficiency of the microgrid system.

With the increased acquisition of electric vehicles around the world, the possibility of using EVs as an added power source is becoming increasingly relevant. This is in contrast to the current situation where EVs mostly act as an added load.

A line diagram of the model is shown in Figure 3.1, showing how the EV, BESS and PV is connected to the DC-bus and the loads and grid is connected to the AC-bus. In addition to this, an inverter is connected to the line between the DC- and AC-buses. The full Simulink® model is shown in Appendix A.

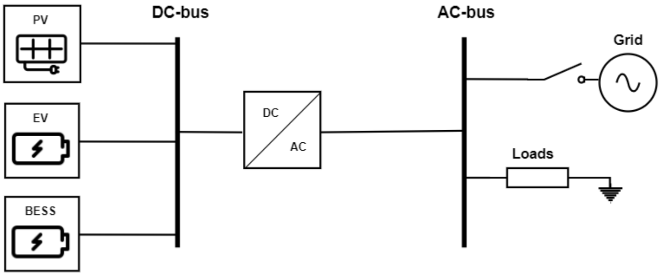


Figure 3.1: Line diagram of the hybrid microgrid.

3.2 Power Production, Storage and Loads

On the DC side of the microgrid, the components are all connected in parallel to the DC bus. The solar PV array has a power rating of 17.5 kW at 25°C, while the two batteries have a power rating of 9 kW each. The specifications for the DC components are given in Table 3.1. The irradiance input for the solar panels in the different cases can be found in Appendix B.

Table 3.1: Specifications of DC-components

	Parameters	**Value**
Solar PV	Power per panel	220 W
	Parallel	8
	Series	10
	V_{max}	390 V
	I_{max}	45 A
	P_{max}	17.5 kW
BESS	Rated Voltage	240 V
	P_{max}	9 kW
	P_{min}	-9 kW
	Capacity	400 Ah
EV	Rated Voltage	240 V
	P_{max}	9 kW
	P_{min}	-9 kW
	Capacity	400 Ah

On the AC side, a load consisting of a base load of 10 kW and a variable load between 0 and 10 kW are connected in paralell to the AC bus. There are two circuit breakers on the AC side of the microgrid. The first breaker, S1, is between the grid and the AC bus, and the other, S2, is between the Inverter and the AC bus. This allows for switching between the two main operating modes of the system where the two switches always will have opposing status.

When S1 is closed and S2 is open, the AC load is provided from the power grid, while the rest of the microgrid is islanded. In this mode, any production from the solar panels are absorbed by the two batteries. Conversely, if S1 is open and S2 closed, the main grid is disconnected and the AC loads are supplied by the power production from the DC side of the microgrid.

3.3 Control Systems

The Simulation model includes several different control systems in order to properly regulate the different parts of the system. These controllers are divided into the main controller and the secondary controllers. The whole Simulink® model can be found in Appendix A.

There are two different main control systems in this model in order to address the objectives of the thesis. The first one is the power flow algorithm based controller which contains an specially tailored algorithm which takes in data from the microgrid, calculates the power flow situation and provides reference values for two secondary control systems and the grid switch. The power flow algorithm is explained in detail in Chapter 4.

The second main controller is the ANN based control system, which is designed to take in the same inputs and provide the same outputs as the power flow algorithm, but with the use of a neural network instead of algorithms and equations. Only one of these will be used to control the microgrid for each simulation.

The secondary control systems are the individual control schemes connected to the converters and inverters of the system. A DC-DC boost converter is connected to the solar PV array, which uses an MPPT algorithm. This setup ensures that the solar panels are providing the maximum power available at all times. In the EV subsystem, the battery is connected to a DC-DC bidirectional converter with current control. This allows for active control of power flow both from and to the EV battery, by adjusting the current reference.

The last secondary control system on the DC side of the microgrid is the BESS with another DC-DC bidirectional converter. This converter is tuned to function as a grid forming voltage controller, ensuring that the voltage level of the DC system is kept at the desired value despite the varying power flow from the other components.

In addition to the above mentioned control systems, a self synchronised universal droop controller is included. This controller adjusts the gate signals to the DC-AC inverter which controls how power is transferred from the DC to the AC side of the microgrid. The SUDC model is based on the design given in [23].

3.4 Simulation Cases

In order to properly test the AI based control system of the microgrid, three cases was created. These cases will illustrate the performance of the system in the different cases, and will allow for assessing the system properly. Case 1 is the base case simulating ”normal” operations, case 2 has variable solar irradiance and case 3 has variable load. From these simulations, the voltage and power at the PCC will be assessed, as well as the EV current and the power flow to and from the other components of the system.

These cases were chosen based on the desire to look at the two main factors that affect the power flow of a microgrid system. The first is the amount of locally produced power, which often comes from solar panels. Since a variety of both natural and unnatural elements can obstruct the sun from the PV panels, the production can vary between full production and no production in a short time period.

The second large factor of the power balance of such a system is the power demand. In addition to a base load that is mostly constant, an added variable demand is dependent on lifestyle, weather, day of week or holidays. Furthermore, coincidences can happen where many loads unwittingly are connected at the same time.

3.4.1 Case 1: Base Case

The base case uses the solar irradiance curve from the 20th of April 2018 in Oslo, with a load varying between 12 and 16 kW throughout the day. This day was chosen based on it having quite a large amount of irradiance, while still not being the middle of summer. This would give an indication of how much the solar power production will be from spring to fall, while staying below the yearly maximum values. The chosen day also includes a slight decrease in the middle of the day, making the example more realistic as most days will have some clouds or other obstructions reducing production at some point.

The load is based on data from Statnett showing the power demand in Norway during a weekday in April. The scale of the data was adjusted to fit with the microgrid system, while keeping the general trend of the demand the same. The focus area when deciding the load curves was to have the general trend somewhat similar to a realistic system while scaling it to fit the rest of the microgrid rather than having a perfect recreation of a real load.

3.4.2 Case 2: Irregular Solar Irradiance

For the second case, the irradiance curve is changed in order to assess how the system is able to handle a sudden loss in local power production. As the solar panels are dependent on sunlight to produce power, a very realistic case would be that the weather changes, resulting in a change in production. In Norway, this can be a light passing cloud cover or more heavy weather with rain or snow. The precipitation may also affect the power production when landing on the solar panels.

A new set of datapoints was created for simulating the variable solar conditions. In the variable case, the irradiance is identical to the base case until 08:00, where it begins to deviate. Figure 3.2 illustrates the base case compared to case 2. The figure shows that the irradiance is decreased to values between 40 and 160 W/m^2. From 13:00 to 14:00 the irradiance is back at the normal level before decreasing down to around 100 W/m^2. After 17:00 the irradiance is the same as the base case.

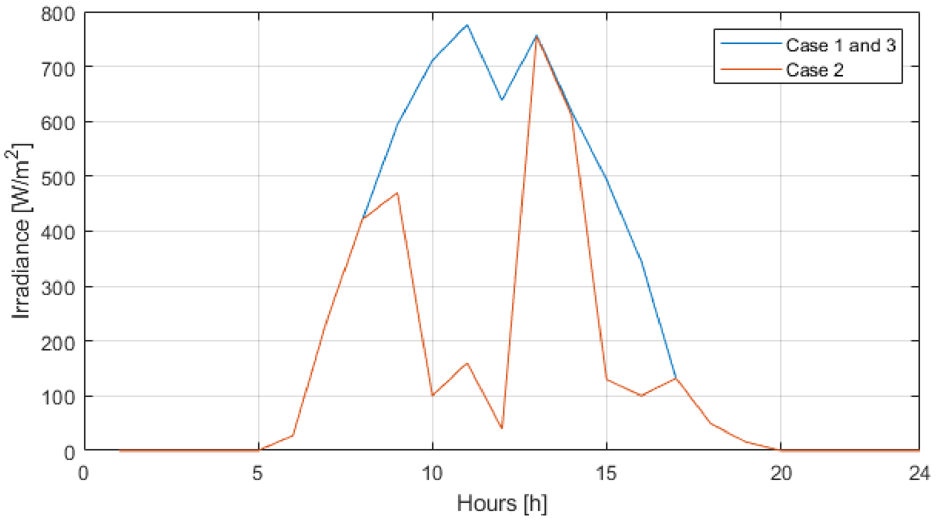


Figure 3.2: Comparison of the irradiance between the cases

3.4.3 Case 3: Irregular Load

Even though it is possible to quite accurately estimate the power demand for most days based on historical data, the exact values may differ substantially because of external factors. With the increased focus on energy prices in Norway, individual consumers have started to adjust their power consumption depending on how expensive it will be. This may lead to less demand for certain hours, while having an increased demand at other times. In addition to this, people do not live their lives in the same way each day, and some variations in power consumption should be expected.

Figure 3.3 shows the normal load curve compared to the variable load. The normal load curve only varies between 12 and 16 kW, and illustrates how a power system with a large share of mostly constant loads are quite predictable. The load in the base case is lowest in the night with the largest peak in the morning and one smaller peak in the afternoon. The variable load is adjusted to have higher peaks in the morning and the afternoon, in addition to a reduction in the load during the middle of the day. The variable load is also never lower than 10 kW, as this is assumed as the constant part of the load which always stays the same.

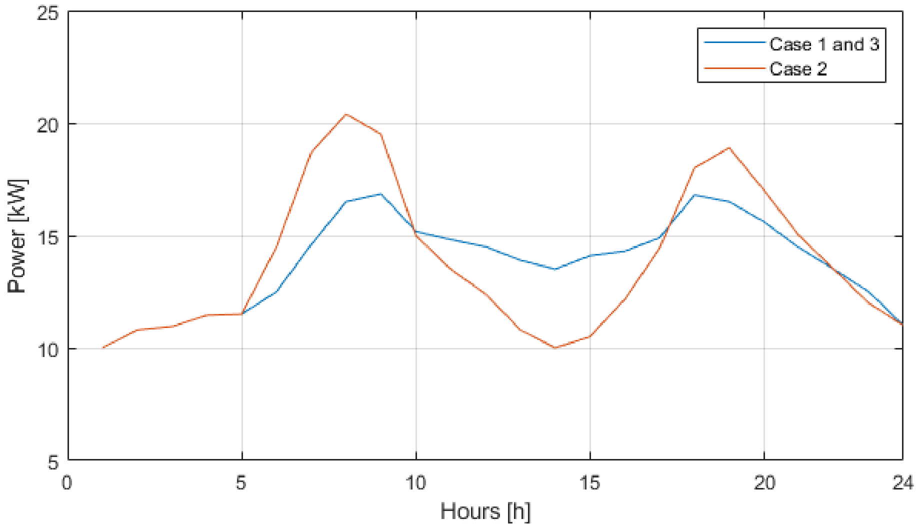


Figure 3.3: Comparison of the load between the cases

4 Power Management based Optimal Power Flow Algorithm

This section contains the content related to the power management based optimal power flow algorithm. Firstly the theory and basis used for the creation of the algorithm is explained, with the assumptions and specifications used. Secondly, the simulation results from simulating the microgrid with the power flow algorithm is presented for the three different cases. Lastly, a discussion of the results and method of creating and simulating the model with the power flow control is included, reflecting on how this part of the thesis was resolved.

There are two reasons for needing such a power flow algorithm in this thesis. Firstly, in order to be able to assess the ANN performance, a more traditional control system is needed as a baseline. In addition to this, the development of an ANN requires a database of training data. In many cases of artificial intelligence training, the required databases of training data is publicly available. One example of this is image recognition tasks. However, when the inputs, outputs and application of the ANN is specifically tailored to a certain task this data is not publicly available and a database needs to be constructed.[6]

In most cases, the amount of training data needed is greater than what is possible to create manually. An algorithm tailored to the specific requirements can be created to solve this problem. This algorithm can be run as many times as necessary with different input values to create a large enough database to train the ANN properly.[6]

4.1 Structure

The power flow algorithm is designed to run at each simulation timestep in Simulink with the inputs of the algorithm being the outputs from the microgrid. These describe the state of the microgrid, and is used as a basis for predicting the state of the microgrid in the next time step. The values describing the state of the microgrid are the loads, solar irradiation, EV-switch and the state of charge for the BESS and the EV. The predicted state of the microgrid is further used to calculate the output values which are the ones used to control the microgrid. These are the grid switch (S_{grid}), the active power reference to the control system of the VSC ($P_{VSC,ref}$) and the current to the EV ($I_{EV,ref}$). A flowchart showing the function of the power flow algorithm is shown in Figure 4.1.[6]

The load and irradiance predictions for the next timestep are based on actual data gathered from Statnett and PVGIS while the switch states, power flow and reference values are set based on optimal power flow and the model specifications given in Table 4.1.

The Simulink simulation was run with the optimum power flow algorithm, and the input and output was recorded for each simulation timestep. The recorded input and output values of the simulation is randomised and the needed training data is extracted and used to train the ANN.

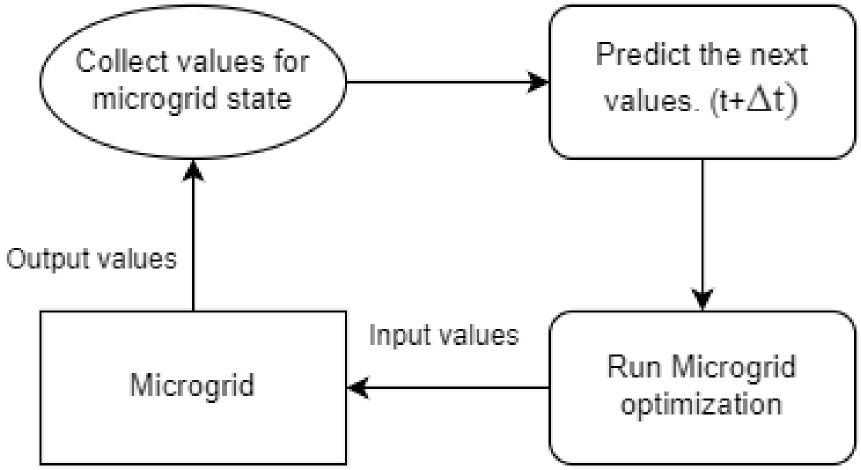

Collect values for microgrid state
Predict the next values. (t+Δt)
Output values
Input values
Microgrid
Run Microgrid optimization

Figure 4.1: Flowchart of predictive power flow model

4.2 Model Specifications

Table 4.1: Assumptions and specifications of the power flow algorithm.[6]

Category	Assumptions
Solar PV	- The temperature is constant at 25 °C - The power produced in grid connected mode is only provided to the DC system. - Maximum power output is 17.5 kW
BESS	- The battery only provides power to the AC load when in island mode - Maximum input/output power is 9 kW - The battery is only charged with the power from the solar PV - Initial state of charge is set to 50%
EV	- The EV-switch is disconnected at certain times of the day. At this time it is not part of the system. - The EV discharges 20 % of maximum battery capacity when disconnected from 08:00-16:00 and 10 % for the second disconnection from 18:00-20:00. - Maximum input/output power is 9 kW - The battery is only charged with the power from the solar PV - Initial state of charge is set to 50%
Grid Switch	- Is only closed when the net power of the system is negative. (Microgrid is unable to be self-sufficient)

4.3 Objective Functions

An OPF model can be used to optimize a certain objective function within the specified criteria. The criteria are constraints that the objective function is subject to. A general OPF problem can be mathematically formulated as shown in Equation 4.1. **x** is the state variable vector, and **u** is the control variable vector. The **f** is the objective function while **g** is the power balance equations and **h** is the system constraints.[28].

$$\begin{aligned} &\text{Min} \\ &\quad f(\mathbf{x},\mathbf{u}) \\ &\text{Subject to:} \\ &\quad \mathbf{g}(\mathbf{x},\mathbf{u}) \\ &\quad \mathbf{h}(\mathbf{x},\mathbf{u}) \end{aligned} \tag{4.1}$$

There are several ways to solve such an optimisation problem as stated in Equation 4.1. The different techniques use a broad range of mathematical approaches. Some examples of these are stochastic programming, decomposition methods, linear programming, mixed-integer programming and artificial intelligence methods. The power flow algorithm in this thesis uses linear programming.[28]

The main goal of the optimum power flow algorithm is to ensure that there is a balance between the power produced and the power consumed by the loads. As a result of this, the objective function can be written as follows when the system is operating in island mode:

$$\text{Min } (\sum_{i=1}^{n} P_i^G - \sum_{i=1}^{n} P_i^L) \tag{4.2}$$

$$\begin{aligned} &\text{Island mode:} \\ &\quad P^G = [P_{PV}, P_{EV}, P_{BESS}] \\ &\quad P^L = [P_{EV}, P_{BESS}, P_{cl}, P_{vl}, P_{loss}] \end{aligned} \tag{4.3}$$

The two batteries are given as both loads and generating units as they can both provide power to the system and receive power from the system.

When the system is grid connected, the grid and the loads are connected to the AC-bus where the power to the loads is provided by the grid. The resulting generated power is only from the PV system and all the produced power is consumed by the two batteries as given in Equation 4.4.

Grid connected mode:

$$\begin{aligned} P^G &= [P_{PV}] \\ P^L &= [P_{EV}, P_{BESS}] \end{aligned} \tag{4.4}$$

4.3.1 OPF System Constraints and Variables

The equations for the system constraints are as given in Table 4.1 and Equation 4.5.

$$\begin{gathered} SOC_{EV,min} \leq SOC_{EV} \leq SOC_{EV,max} \\ SOC_{BESS,min} \leq SOC_{BESS} \leq SOC_{BESS,max} \\ -P_{EV,max} \leq P_{EV} \leq P_{EV,max} \\ -P_{BESS,max} \leq P_{BESS} \leq P_{BESS,max} \end{gathered} \tag{4.5}$$

The control variables of the OPF are the variables used to control the microgrid and includes the grid switch, the current reference for the electric vehicle and the power reference of the VSC. These can be seen in Equation 4.6.

$$u = [S_{grid}, P_{VSC,ref}, I_{EV,ref}] \tag{4.6}$$

Table 4.2: System constraint variables and other specifications.

Specifications	**Value**
$SOC_{BESS\backslash EV,min}$	1 %
$SOC_{BESS\backslash EV,max}$	99 %
$P_{BESS\backslash EV,max}$	9 kW
$S_{VSC,max}$	20 kVA

4.4 Predictions

4.4.1 Solar Power Prediction

The solar irradiance calculation of the power flow algorithm was based on a real solar irradiance curve from 20 April 2018 in Oslo. This was downloaded from PVGIS and contained hourly values for the solar irradiance throughout the day in an array. As the timesteps of the simulation may vary, a new array was created with the difference in irradiance from one hour to the next. This is shown in Equation 4.7.

$$SI_{diff}(i) = SI(i+1) - SI(i) \tag{4.7}$$

As there may be a desire to change the step size of the calculation, a new array is created with length equal to $steps * 24$, where the value of the variable "steps" is the number of times between each hour the irradiance value is updated. A higher value results in a more continuous curve between any two given hours, while a low value results in a more stepped curve. As the irradiance data is given hourly, the resulting array has a linear change between each full hour. The value calculated for the next timestep is based on the current irradiance with an added value based on the integral of the irradiance curve and the step size.

The main power flow model checks the relationship between the timestep of the simulation and the step. As a result the correct irradiance for each timestep is used, even when changing the simulation sample time.

The resulting power production from the PV panels is calculated with the following equation:

$$P_{PV} = \frac{SI}{1000} \cdot P_{panel} \cdot N_s \cdot N_p \tag{4.8}$$

4.4.2 Variable Load Prediction

The variable load is constructed to emulate the use of a household, with peaks in the morning at 08:00 and in the afternoon at 17:00-21:00. The load curve is based on the base case as specified in subsubsection 3.4.1. The load prediction for each timestep is calculated in the same way as the solar irradiance, with the same step size as specified earlier.[6]

4.4.3 EV and BESS State of Charge Prediction

The prediction of the state of charge is calculated based on the grid switch and the net power. Equation 4.9 shows how the SOC($t + \Delta t$) is calculated.

$$SOC(t + \Delta t) = SOC(t) + \frac{\int_t^{t+\Delta t} I dt}{C_{\text{ref}}} * 100\% \tag{4.9}$$

For cases when both the BESS and the EV are connected, the power flow both to and from the two batteries will be shared. If any of the two batteries are below 30%, it will not discharge. The maximum current flow from the each battery is limited by maximum power divided by the voltage of the battery, $I = \frac{9000W}{240V} = 37.5$ A.

When the microgrid operates in island mode, the power flows from the solar PV panels to the two batteries, with 1/3 of the power going to the BESS and 2/3 to the EV. This is controlled by adjusting the EV reference based on 4.10, while the voltage control system of the BESS adjusts the power flow in order to absorb the excess power in the system to uphold the correct voltage level. The power distribution when charging both batteries was based on prioritising the charging of the EV in order for it to have a high enough SOC to drive when needed.

$$I = \frac{2}{3} \cdot \frac{P_{PV} - P_{cl} - P_{vl} - P_{losses}}{V_{EV}} \tag{4.10}$$

If the EV is disconnected, the current to the BESS will be as seen in Equation 4.11. If the grid is connected, the loads and the losses are neglected because the grid will supply power to the loads. Only the losses in the VSC are considered, and are calculated from the current flow and the internal resistance.[6]

$$I = \frac{P_{PV} - P_{cl} - P_{vl} - P_{losses}}{V_{BESS}} \tag{4.11}$$

4.5 Algorithm Output

As previously mentioned, the outputs of the power flow algorithm are the grid switch S_{grid}, $P_{VSC,ref}$ and $I_{EV,ref}$. The current to the EV and the power reference for the VSC depends on whether the microgrid is in grid connected or island mode, which means this must be decided first. This is calculated by taking the max production and subtracting the load and losses.

4.5.1 Grid-Connected Mode

When grid-connected, the DC bus is isolated, and the power production from the solar panels are consumed by the two batteries. This means that there is no power flowing across the inverter, resulting in $P_{VSC,ref} = 0$. The reference current is calculated by Equation 4.12.

$$I_{EV,ref} = \frac{P_{EV}}{V_{EV}} \tag{4.12}$$

P_{EV} will depend on the state of charge, and the EV-switch. When the EV-switch is off, $P_{EV} = 0$. When the EV-switch is on, and the state of charge on the EV is less than 100 % the power to the EV will depend on the state of the BESS. If the SOC of the BESS is 100 %, $P_{EV} = -P_{PV}$, and if the SOC of the BESS is less than 100 % $P_{EV} = -P_{PV} \cdot \frac{2}{3}$ with the rest going to the BESS. It is important that the SOC_{max} is set to a number that allows full charge for one step without crossing the limit for the battery.[6]

4.5.2 Island Mode

When in Island mode, the goal is for the output values to create balance the system. For a balanced system, the power production and demand needs to be equal. $P_{VSC,ref}$ is therefore set equal to the estimated power demand from the algorithm as seen in Equation 4.13. The reference value is sent to the internal control system of the inverter, making sure the correct amount of power is transferred to the AC-bus.

$$P_{VSC,ref} = P_{vl} + P_{cl} + P_{losses} \tag{4.13}$$

When the power reference needed for power balance is calculated, the power demand from the EV and the BESS can be estimated based on the power demand and the local power production. The Solar PV panels will produce the maximum amount based on the irradiance and is calculated from Equation 4.8. As a result, the combined power required from the EV and BESS can be stated as:

$$P_{need} = P_{VSC} - P_{PV} \tag{4.14}$$

If both batteries have a sufficient SOC, and are connected, the current reference of the EV will be set as given in Equation 4.15, while the BESS will discharge an equal amount of power as the EV in order to keep a stable voltage in the DC grid. If any of the constraints stated in subsection 4.3 are violated in order to achieve power balance while in island mode, the grid switch is connected, and the values are recalculated based on the equations given under grid connected mode.

$$I_{EV,ref} = \frac{P_{need}}{2} \tag{4.15}$$

4.6 Simulation Results

Below are the results from simulating the power flow algorithm based control on the microgrid for one whole day. When doing the simulations, it became evident that the power flow algorithm did not correctly control the battery settings with regards to the SOC. Because of time limits, the size of the batteries was increased. This prevented the SOC to fall below the limits during the day, allowing for simulating without the results being affected by this mistake.

4.6.1 Case 1: Base Case

For the base case, the power flow algorithm chooses to connect to the main grid three times during the course of 24 hours, which can be seen in Figure 4.2. The first connection is very brief and occurs when the EV disconnects at 08:00. The two other times the controller connects the microgrid to the main grid are at 16:00 and 19:00. The power reference goes to zero when the grid switch is turned on and the current reference is zero for the period where the EV is disconnected from the grid which is from 08:00 to 17:00

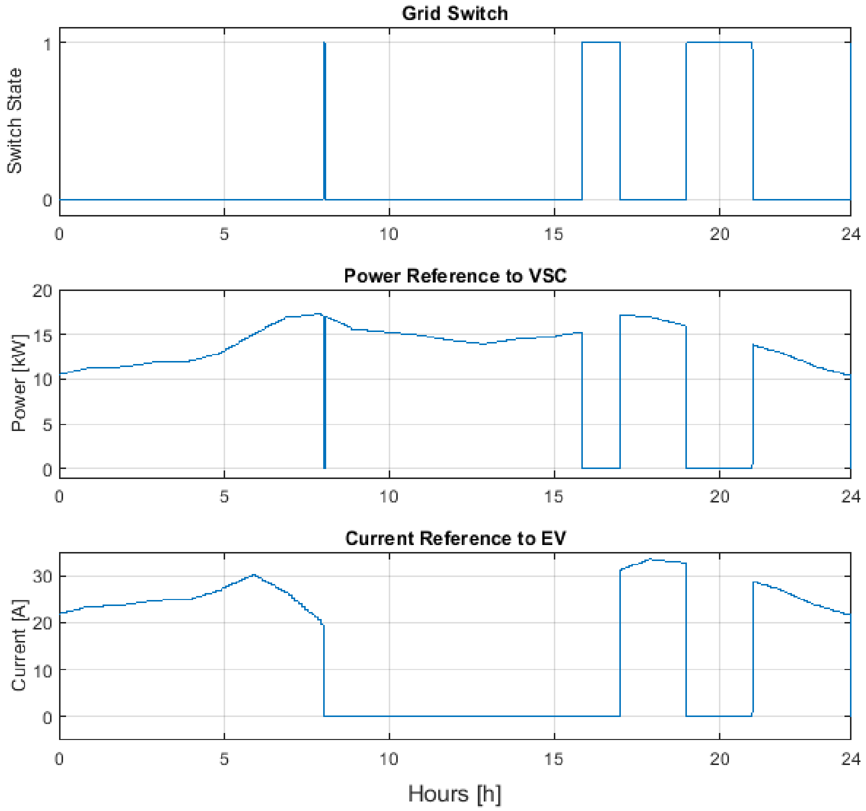


Figure 4.2: The output values from the simulation of case 1 with the power flow algorithm as the centralised controller.

The variable load and solar power follows the specified inputs for the simulation, as shown in Figure 4.3. The EV power follows the load until the solar panels start producing power before disconnecting at 08:00. In the period before the disconnection of the EV, the BESS follows the same power output as the EV which adds up to the total demand from the load. The BESS increases when the EV disconnects to compensate for the power need. When the solar power increases, the demand from the BESS decreases and vice versa. The BESS changes to consume power in the period of grid connection between 16:00 and 17:00. This does not happen when the grid is connected at 19:00 as there is no solar power production at this point.

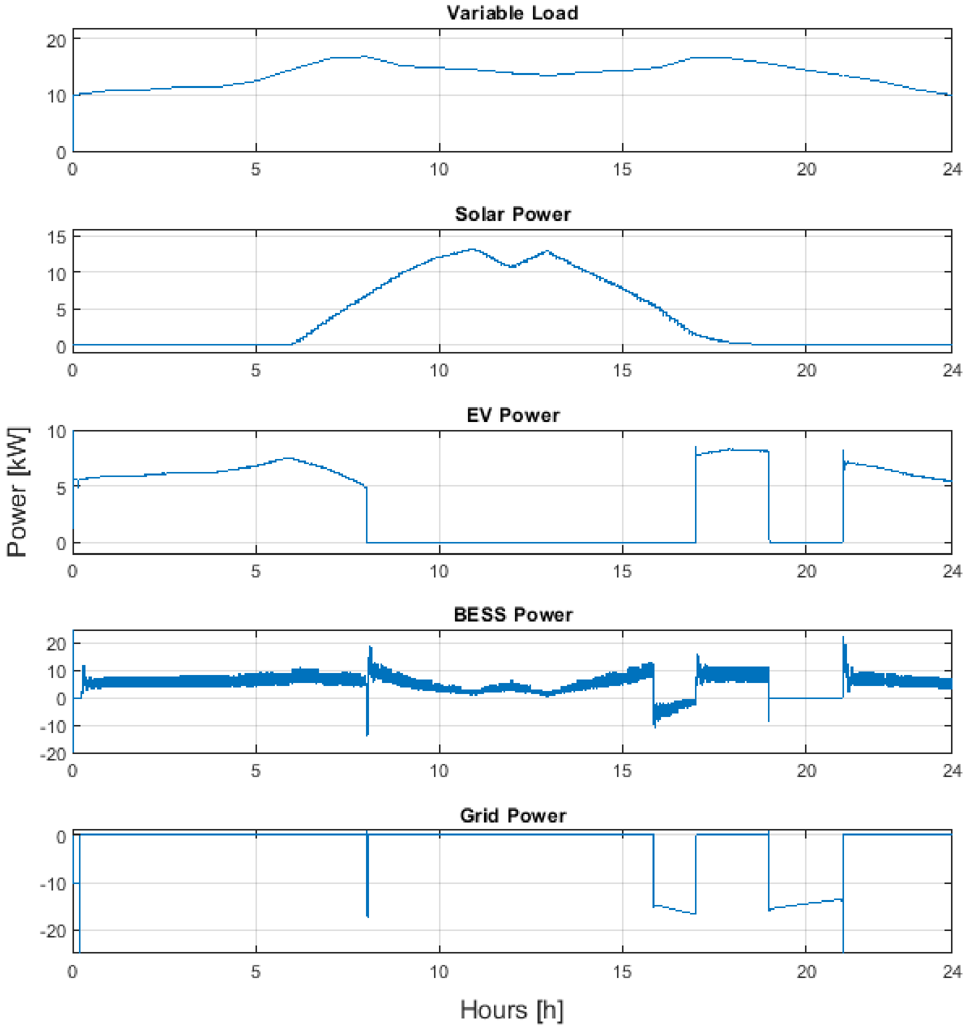


Figure 4.3: Measured values from the microgrid from the simulation of case 1 with the power flow algorithm as the centralised controller.

The voltage at the point of common coupling is mostly nonfluctuating and set to 400 V, which can be observed in Figure 4.4. There are three large peaks, of which the first appears when the SUDC is enabled. After this, both the DC voltage and the voltage at the PCC is mostly steady until the slight moment of grid connection at 08:00, which creates a small amount of transient values and ripples in the three plots. The two other peaks of larger voltage and current can be found at the point of switching from grid connected mode to island mode. Lastly, the DC voltage drops rapidly in the last grid-connected mode, before quickly jumping back to 650 V when islanded.

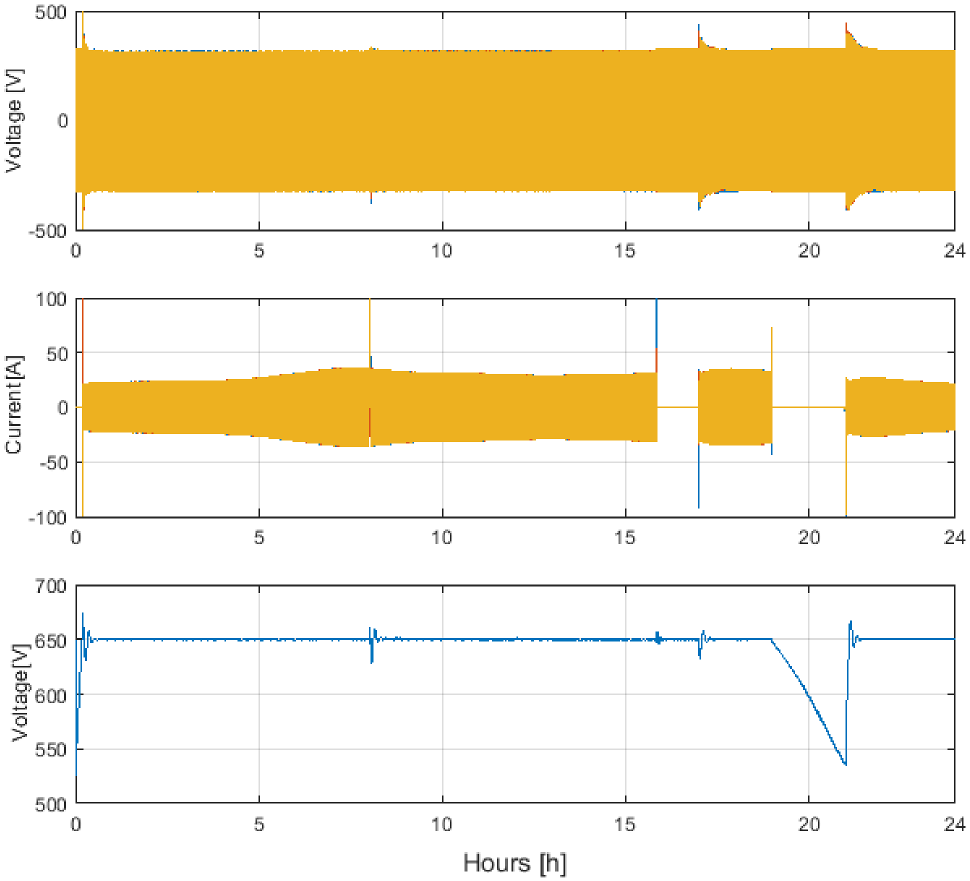


Figure 4.4: Measured values from the microgrid from the simulation of case 1 with the power flow algorithm as the centralised controller.

4.6.2 Case 2: Irregular Solar Irradiance

The output values from the power flow algorithm acting as the centralised controller for case 2 can be seen in Figure 4.5. These output values are identical to the ones from the previous simulation case.

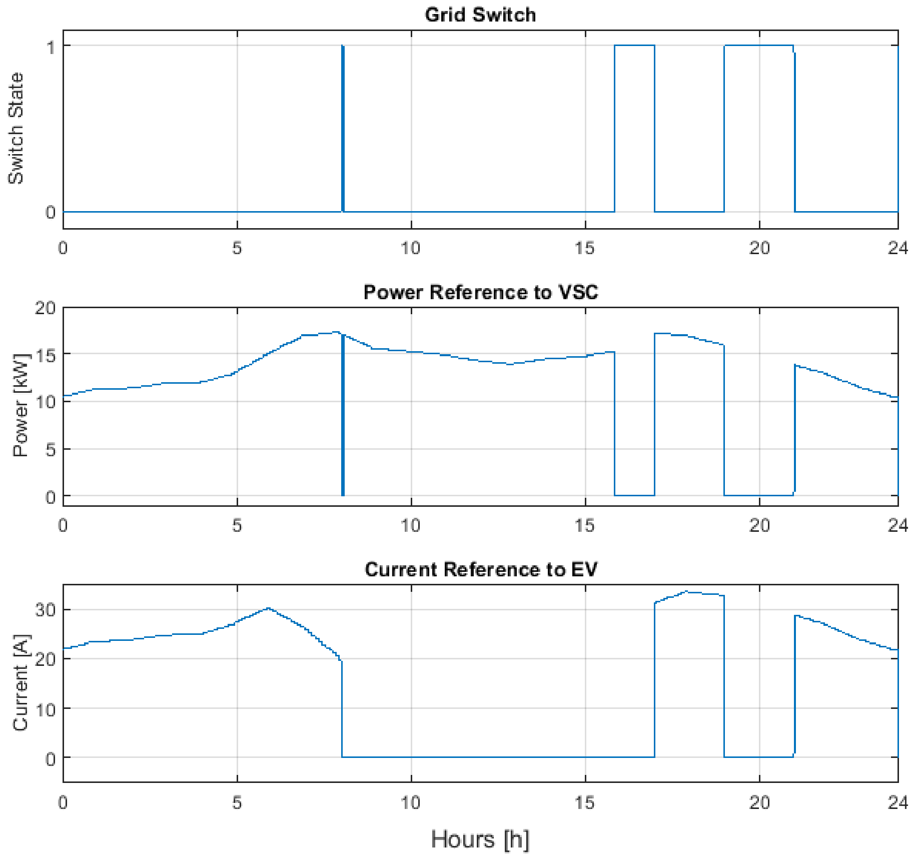


Figure 4.5: The output values from the simulation of case 2 with the power flow algorithm as the centralised controller.

When looking at Figure 4.6, the solar power output follows the expected trend from the irradiance input. The EV power remains similar to the base case, while staying below its maximum output power limit. The EV does however not charge at any part of the day, which is unsustainable. The BESS increases the power output to compensate for the lack of power production from the solar panels, but does however increase past the 9 kW limit of the battery. Even though the BESS has negative power output at small periods of time, these are short and large peaks resulting in little charging. As a result, neither the EV or the BESS receives any power during this 24 hour period.

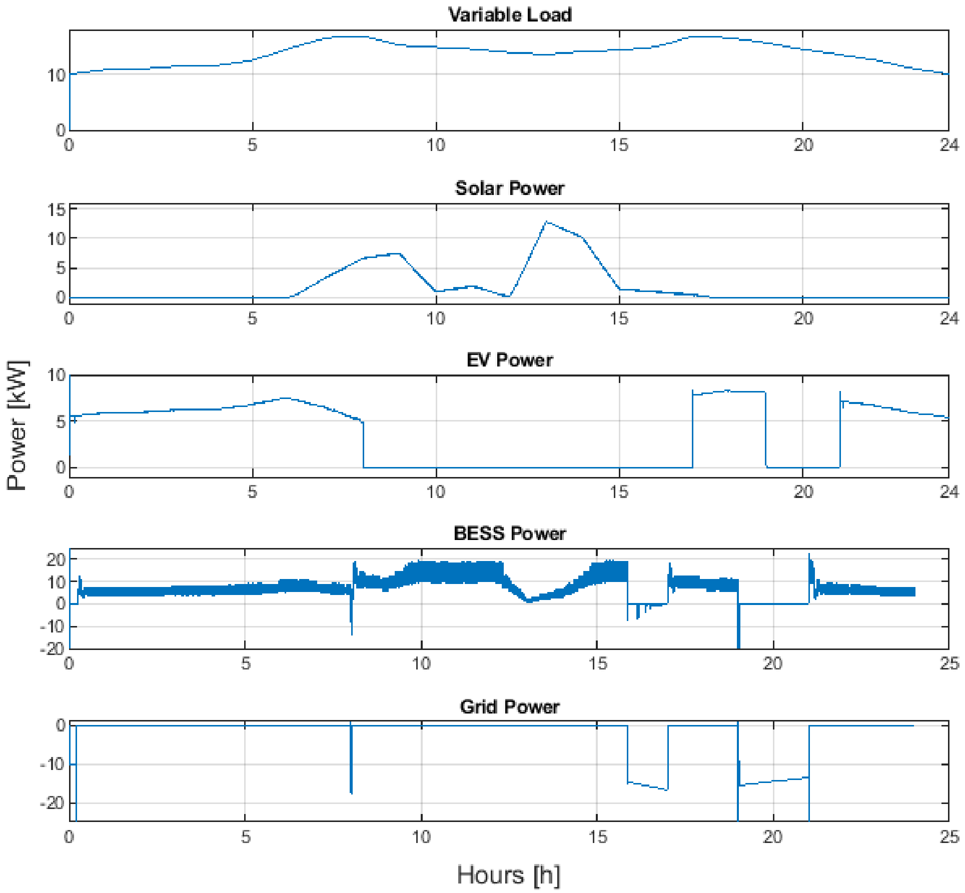


Figure 4.6: Measured values from the microgrid from the simulation of case 2 with the power flow algorithm as the centralised controller

.

The voltage level at the PCC is mostly constant at 400 V, except for the same peaks as seen in the previous case. Figure 4.7 shows how the current varies with the power demand of the load, and goes to zero when the grid is connected as no current travels from the DC to the AC bus. The control system struggles to keep the DC voltage at 650 V with the voltage decreasing both from 10:00 to 12:00 and from 19:00 - 21:00 as could be seen in the previous case as well.

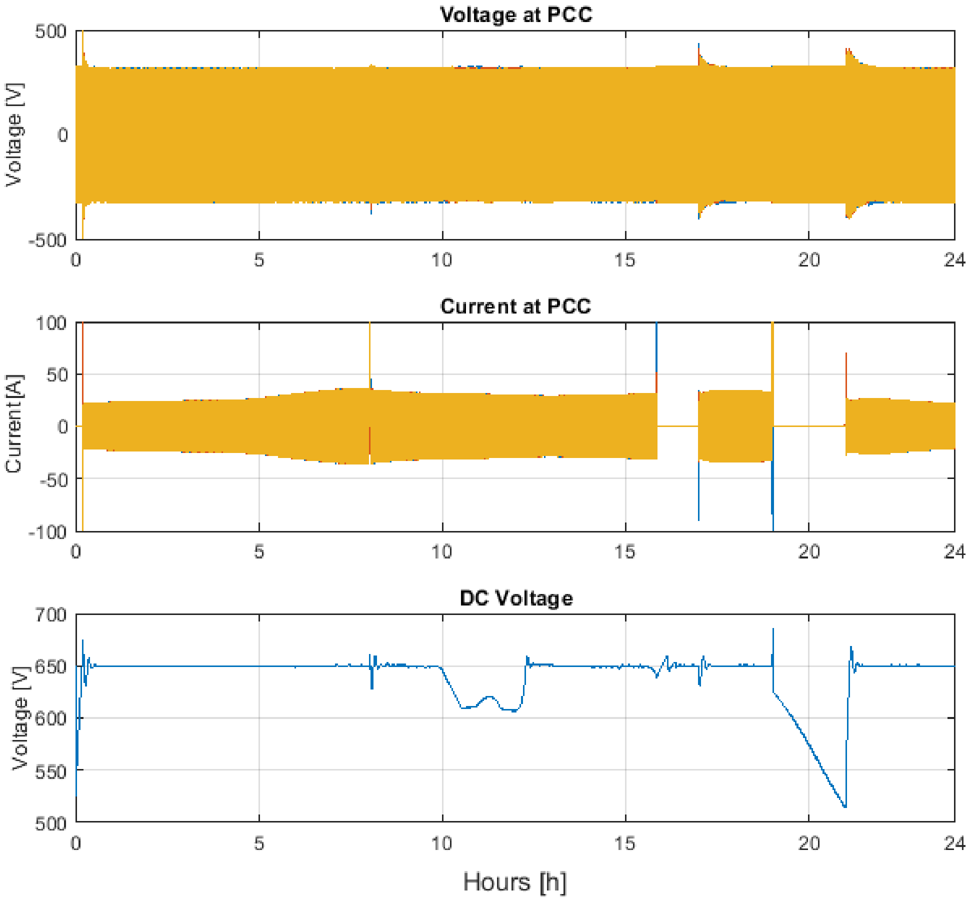


Figure 4.7: Measured values from the microgrid from the simulation of case 2 with the power flow algorithm as the centralised controller

4.6.3 Case 3: Irregular Load

For the third case, Figure 4.8 shows the three output values from the power flow based centralised controller. These are identical to the previous two cases.

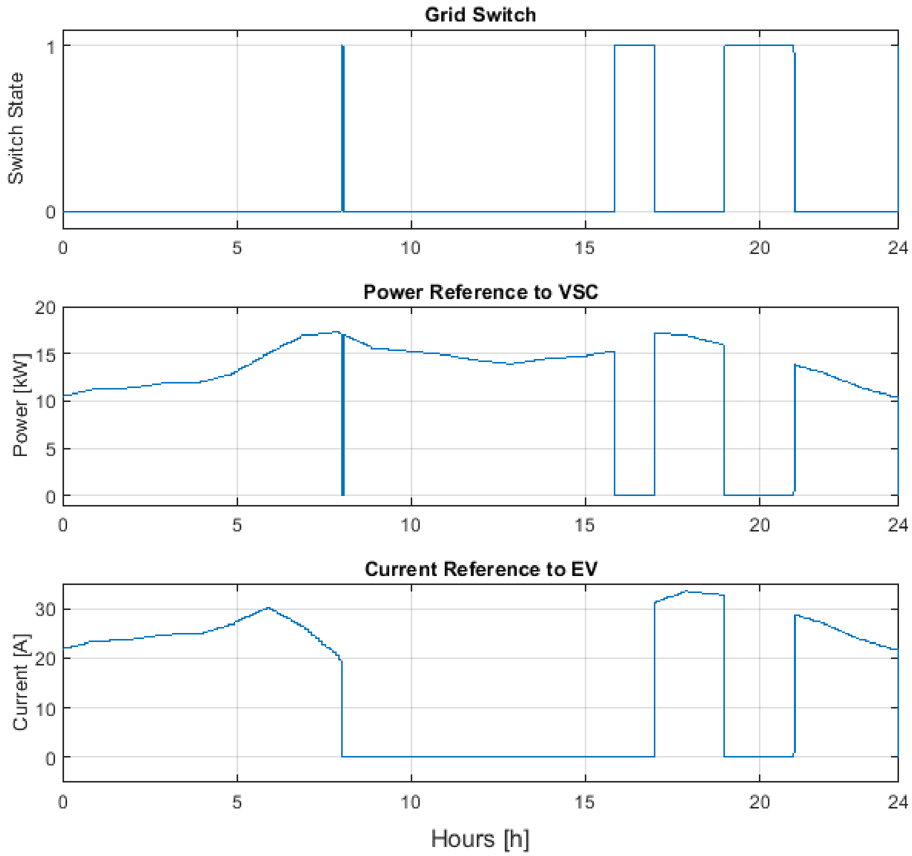


Figure 4.8: The output values from the simulation of case 3 with the power flow algorithm as the centralised controller.

Figure 4.9 contains the measured values from several components in the microgrid. The variable load has peaks of 20 kW in the morning and evening with the lowest values at the base load of 10 kW around midnight and midday. The solar power follows the same trend as the irradiance input, with more power when the sun is higher in the sky, and a decrease because of clouds at 12:00. The EV power follows the EV reference, which is based on the load and local power production.

In this case, the BESS can be seen to increase slightly above its maximum output limit in the period from 06:00 to 09:00 and around 18:00. Both the BESS power and the grid power experiences transients at the time of grid switching.

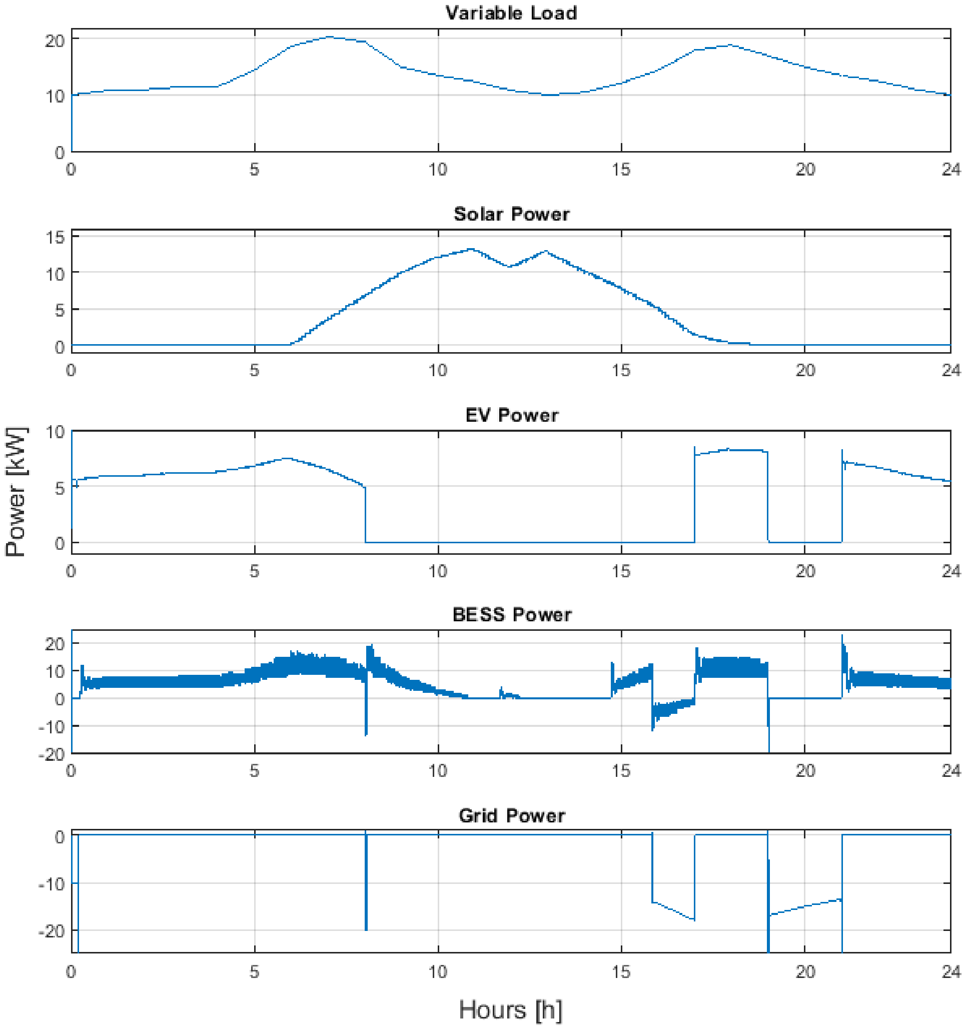


Figure 4.9: Measured values from the microgrid from the simulation of case 3 with the power flow algorithm as the centralised controller

.

The voltage at the PCC is plotted in Figure 4.10 and shows the same three large peaks as seen before, in addition to some noise in the period from 12:00 to 15:00. The same period for the DC voltage plot shows a large increase in voltage, with a sharp decrease in voltage from 19:00 to 21:00. The VSC current shows the same trend of increasing and decreasing with the load curve from Figure 4.9 in addition to transients at the time of switching for the grid switch.

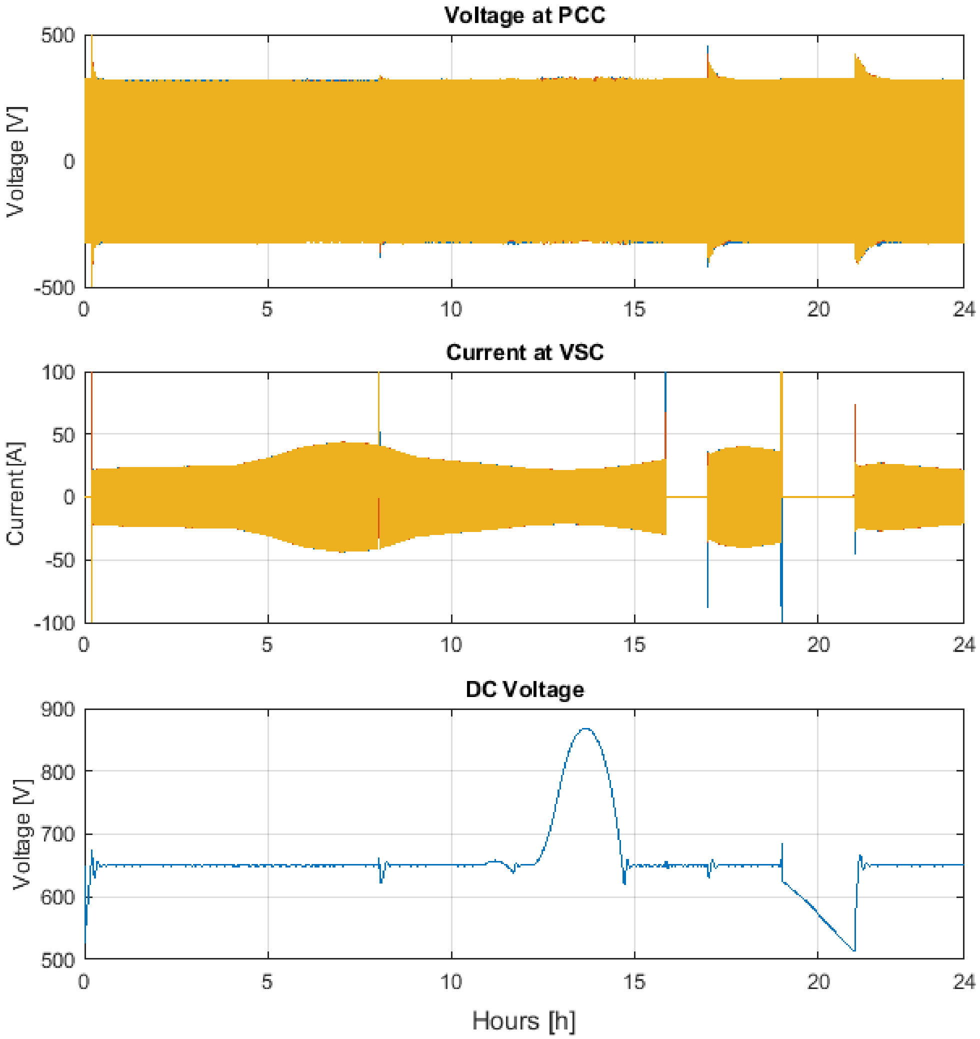


Figure 4.10: Measured values from the microgrid from the simulation of case 3 with the power flow algorithm as the centralised controller

.

4.7 Discussion of Results

When considering the performance of the of the power flow algorithm based controller and its development, certain objectives can be considered. The main goal of the controller was to identify the power balance and decide if the power demand can be supplied internally, or if the main grid is needed. When this is done, the second most important function is to provide reference values for the EV and VSC in order for them to operate optimally to achieve a balanced power flow.

As the batteries was scaled up to prevent the SOC to fall below the specified limits, the general system becomes slightly less realistic. This change allows the two batteries to discharge at all times during the day, which is unsustainable in a practical use case. A proper system should strive to achieve net zero discharge during a day, or in a period of a couple of days in order to perform similarly over time. This could result in the microgrid being less self sufficient over one day of simulation, but in turn becoming more self sufficient over time. Smart battery deployment is an important factor that is needed for a well performing system, and is an area that could be independently researched.

The EV is disconnected at the exact same time each day, which also is the time of day with the most solar power production. For simplicity, the settings of this control system is tuned to only charge the batteries from the local solar production and not from the grid. This results in a system which only discharges from the battery. In a case where the EV is charged at the user's workplace, this could be a realistic scenario.

However, it would be more realistic to tune the system to charge the two batteries from the grid when grid connected, in addition to being able to connect the grid in order to charge. A smart battery management system can also be tuned to charge the batteries at times with lower energy prices, and focusing on discharging at times with high energy prices. Nevertheless, this was not the focus area of this thesis, as the performance of the ANN based centralised control was not dependent on smart battery deployment.

The transient values that appear when connecting and disconnecting the grid are not desired in such a system, but will not affect the objectives for this thesis. This is mainly because the goal of the thesis is not to create an optimal microgrid system, but rather to assess the challenges and possibilities of ANN based centralised control. These transients can be caused by several components in the microgrid, but can most likely be largely mitigated by implementing a re-synchronisation element to the SUDC, as mentioned in section 2.4.2.

The results provided in the previous section shows that the system does achieve the two main objectives for the base case, but struggles when the input values are different to what is expected. This can be observed in the three output values of the centralised controller which are identical in the three cases.

However, the measured power flow, current and voltage levels of the microgrid system are not the same in the three cases. The reason for this is the function of the secondary control systems such as the voltage controlled bidirectional controller for the BESS. The case with low local production and the case with irregular loads have different values for net power (production - demand) in the system at various times of the day.

The period between 10:00 and 15:00 behaves quite differently in the three cases, even with similar central control output and is therefore an interesting area to highlight when analysing the function of the power flow algorithm. In this period the EV is disconnected which leaves the solar power production and the BESS to provide enough power to the load. With normal solar production as in case one and three, the solar power production is above 10 kW for the whole period allowing the BESS and solar panels to produce enough power for the loads. For case one, this results in a stable voltage level for that period.

When the power flow algorithm provides mistaken outputs, the secondary control systems that use the output values are affected. The function that decides if the BESS is in charge or discharge mode, uses $P_{VSC,ref}$ as an input. As a result, case three shows that even though the solar power production is higher than the load, the BESS is not charging. This creates an excess of power, resulting in an increase of DC voltage.

Lastly, case two has a different challenge. As the load is the same as case one, $P_{VSC,ref}$ is correct, but with the reduced solar power production the net power produced is not enough to supply the load. The internal control of the BESS increases in an attempt produce enough, surpassing the specified limit, but is still not able to supply enough power. As a result, the voltage level decreases in this period.

Even though these results are sub-optimal with regards to the performance of the OPF, they are still highly valuable. The main goal for this thesis is to create an artificial neural network that can be used as a centralised controller. As the base case provides the desired output, this allows for the comparison of performance based on the measurements from the microgrid.

In addition to this, the above results show how the power flow algorithm was unable to handle the variety of inputs to provide feasible results. As long as the ANN is trained on data from a simulation with an acceptable centralised controller, the base case can be used as a baseline for the two control systems while case two and three can be compared to see how the two systems handle a variety of input values. As one of the motivational factors for using ANN is the possibility of achieving acceptable results with a varying or missing inputs, the poor results from case two and three allow for assessing this exact theory.

5 Power Management based Artificial Neural Network

The following chapter contains the introduction, background and appropriate theory on artificial neural networks relevant for this thesis. A description of how the ANN was developed and trained is also covered along with the results from simulating the microgrid with the ANN as the centralised controller. Lastly, a discussion of the performance of the ANN, and how it compares to the performance of the power flow algorithm is included. As the research on ANNs was started in the previous project, some paragraphs containing background and theory on ANNs are provided as shown in the previous project while some have been modified or amended with added research.

With increasing research, more complex problems and the fast development of technological practises, the world of artificial intelligence and machine learning is becoming an integral part of our society. There are several types of AI, but the one researched in this paper is the artificial neural network. The following chapter gives an overview of the research done on existing papers about ANNs based on the overall goal of the project. The state of the art of artificial neural networks were reviewed in the project preceding this thesis, which serves as a basis for subsections 5.1-5.4, along with varying degree of amendments and additions based on new insights from the recent months.

5.1 What is an ANN?

The human brain consists of a collection of more than 10 billion interconnected neurons. Each neuron has a threshold of electric potential, which when reached sends or *fires* a signal of activation. These signals are used by the brain to complete computationally complex tasks with the use of massive parallelism, a highly parallel computing structure, and the capability of imprecise information-processing. This non-linear, dynamic interaction process among the neurons are what is often referred to as intuitive thinking in humans. The artificial neural networks builds on this biological structure, and differs from the conventional computing machines which often employ a complex set of equations, requires a certain input and follows a given path.[29, 30]

The ANN allows using very simple computational operations to solve complex problems that conventional computers may struggle to do. The structure of each individual neuron is quite simple, and has a limited function. The power of the systems lies in the network of large amount of interconnected neurons. In contrast to traditional computational methods where the rules and equations are given in the neural network is *trained*, which makes the system more adaptive and flexible in addition to being able to develop and learn and reach a higher accuracy. However, as the system is trained by the use of input and output values, the inner functions of the ANN remain unknown. This is called a black box model, where the exact process that happens in between the input and output is concealed.[29, 31]

5.2 Architecture

To understand how to build a neural network, the general architecture of the network must be explained. The architecture describes the different layers of the network, with the amount of layers, neurons and their connections. For the most basic problems a single layer structure, which only has one layer of input neurons and one layer of output neurons, can be used. This type of ANN may be limited in performance because of the lack of complexity leading to the wide use of multilayered ANNs. The basic architecture of a multilayered ANN consists of three types of neuron layers. These are the input layer, the hidden layer/layers and the output layer. Each neuron has a value between 0 and 1, and all neurons in one layer are connected to each neuron in the next layer. An example of a multilayered artificial neural network with one hidden layer is shown in Figure 5.1.[6, 30]

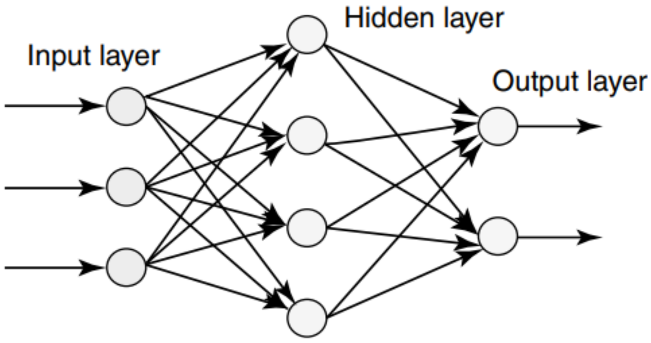


Figure 5.1: An example of a multilayered artificial neural network.[30]

The number of neurons in the input and output layer can vary depending on the problem. For image recognition for example, each neuron in the input layer can represent one pixel each. For a power system, the input values can be anything from a switch state to a current measurement as long as the value is between 0 and 1. Which inputs that are used and how they should be represented is one of the first challenges of making an ANN.[32]

The output layer depends on what the desired result of the calculations by the ANN is. For a number recognition software, this could be 10 neurons from 0 to 9, while for a simpler calculation the output layer may only have one neuron with binary ”yes” or ”no” result. For power electronic systems this would often be switch states, expected power production, reference voltage or current, or other signals. The number of output neurons will therefore vary largely from one ANN to another.[32]

The hidden layers are more complex and variable and can consist of one or more sets of neurons and one or several layers. Designing the structure of the hidden layers is more challenging than deciding the input and output layers. It is not possible to sum up the design process of the hidden layers in a few simple rules. Instead, researchers in the field of neural networks have developed many design heuristics for the hidden layers. This can guide people in the design process to get a desired behaviour from their network. One example of this is how the amount of layers affect the training time of the neural network. However, there may be several different ANN architectures that give a similar result which often triggers the need for trial and failure by iterating through different setups and assessing the performance.[6, 32]

5.3 Computations

The ANN method to be used in this thesis is a feed-forward configuration where the signals travel from the input layer through the network to the output layer, with no feedback loops. Each neuron in a given layer receives a signal with the value from all the neurons in the previous layer. Each received value in combination with a weight and a bias is used to decide the value of the selected neuron. How the weights and biases are set and adjusted is explained later in this thesis. Figure 5.2 shows the fundamental structure of a neuron, with Equation 5.1 and 5.2 showing the mathematical expressions.[6, 30, 32, 33]

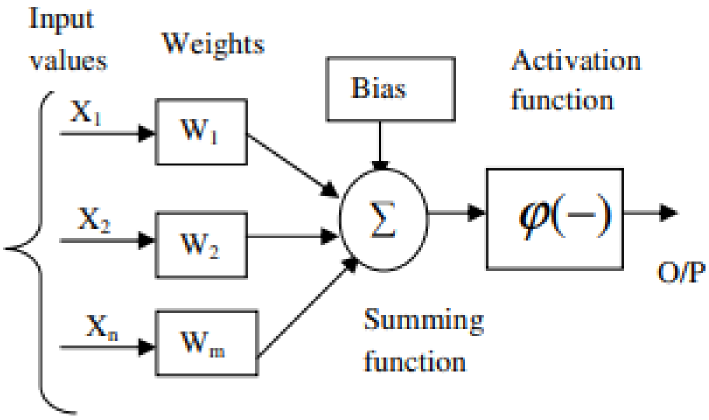


Figure 5.2: The fundamental structure of a neuron.[33]

$$U = \sum_{j=1}^{m} W_j X_j \tag{5.1}$$

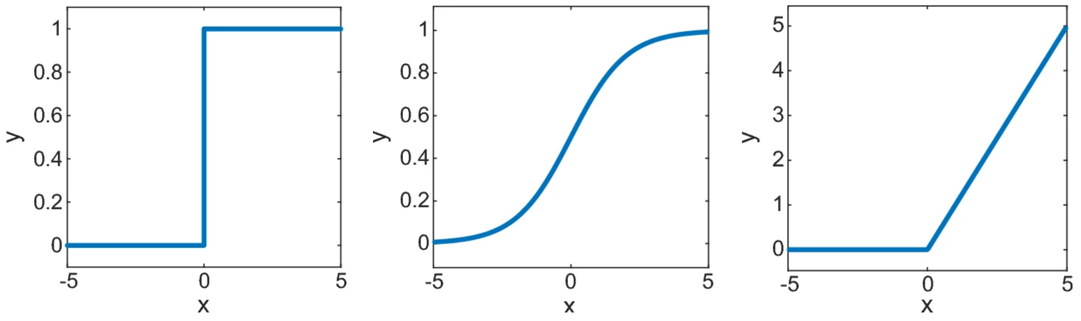
$$y = \varphi(U + b) \tag{5.2}$$

As can be seen in the above equations, the output value from each neuron is dependent on four elements. The input values to the neuron, the weights for each input value, the bias of the neuron and the activation function. The input values are the output values from the previous layer, which either is the input values of the ANN or the previous hidden layer for a feed-forward neural network. In other types of ANNs such as a recurrent neural network, the input values may also be from all the other layers in the ANN.[32, 33]

Each input value, X, is the signal from one single other neuron in the system. Each of these signals are multiplied with a weight which works as an indication of the importance of that signal. The neuron signals with a larger value for W will have a larger impact on the value of the neuron, compared to the signals with a lower weight.[32]

After all the input values and weights are summarised, a bias is added. This bias allows for adjusting the resulting value from all the input signals and weights up or down. In biological terms where the neurons either fire or not, a large bias would lower the threshold for the neuron to fire. The bias in combination with all the weights allow for a large variation of possible adjustments affecting the final value of each individual neuron. In this way, a network with several layers and neurons becomes quite complex, while still using simple multiplication and addition at each step. Modern Graphic Processing Units (GPUs) have the ability to process multiple matrix computations in parallel, which allows for training and running large neural networks without it being too complex and time consuming.[6, 34]

The resulting value after the sum of the input and weights with the added bias can be very large or small. As a result, an activation function is needed to decide when the artificial stimulation value of input signals are large enough to fire the neuron. In ANNs, the neurons are not required to be binary, but can be a wide variety of values. As a result of this, several different activation functions have been developed, each with different characteristics. Three of the most used activation functions are the Sigmoid function, the Rectified Linear Unit (ReLu) function in addition to the binary Step function function.[35]



a) Step function b) Sigmoid function c) ReLu function

Figure 5.3: Commonly used activation functions

Equation 5.3 shows the equations for each of the three functions in Figure 5.3:

$$\varphi_a(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases} \qquad \varphi_b(x) = \frac{1}{1+e^{-x}} \qquad \varphi_c(x) = \max(0, x) \tag{5.3}$$

The step function is mostly used when there is a need for a binary output which is either 0 or 1, and is the activation function that is the most similar to how the biological neurons operate. The Sigmoid function also provides a value between 0 and 1, though not binary. This function allows for a variable output between 0 and 1, and is therefore useful in the last layer before the output layers in situations where a decimal value between 0 and 1 is desired. However, the function suffers from a situation called the vanishing gradient: a situation where the derivatives of the function, which contributes to the updating of weights and biases, reduces to zero in the saturation area. This reduces the contribution of the first layers and lowers the accuracy of the neural network. Because of this, the Sigmoid function is mostly used in the output layers of a system.[35]

The ReLu function is also commonly used as its derivative is either 1 or 0, thus mitigating the problem of the vanishing gradient. In addition to this, the linear nature of the function allows for faster convergence and training because of the ease of computing compared to the exponential functions used in other activation functions.[35]

When looking at a network with a large number of neurons in the previous layer, the value of neuron 0, (with a zero-based numbering) in layer 1 can be calculated by Equation 5.4. The superscript corresponds to the layer number while the subscript corresponds to the neuron number in each layer.[6, 32, 36]

$$x_1^{(1)} = \varphi(w_{0,0} \cdot x_0^{(0)} + w_{0,1} \cdot x_1^{(0)} + ... + w_{0,n} \cdot x_n^{(0)} + b_1) \tag{5.4}$$

This equation is repeated for each neuron in the layer. The values for all the neurons can be added as a matrix, as shown in Equation 5.5, which can be compressed into Equation 5.6 where X is the matrix of all the x^n values of the neurons in layer n. W is the matrix of weights, and B is the matrix of biases.[6]

$$X^{(1)} = \varphi\left(\begin{bmatrix} w_{0,0} & w_{0,1} & \cdots & w_{0,n} \\ w_{1,0} & w_{1,1} & \cdots & w_{1,n} \\ \vdots & \vdots & \ddots & \vdots \\ w_{k,0} & w_{k,1} & \cdots & w_{k,n} \end{bmatrix} \begin{bmatrix} a_0^{(0)} \\ a_1^{(0)} \\ \vdots \\ a_n^{(0)} \end{bmatrix} + \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_n \end{bmatrix}\right) \tag{5.5}$$

$$X^{(1)} = \varphi(WX^{(0)} + b) \tag{5.6}$$

As a result, the calculation of the values for all the neurons in the next layer, based on all the neurons in the previous layer with the respective weights and biases can be done with a quite simple matrix multiplication. This ends up making the code for the neural network quite simple, in addition to utilising the potential of matrix optimisation libraries in various programming languages.[6, 32]

5.4 Learning

One of the most important parts of the artificial neural network is the ability to learn. This allows for training the system, and is what is needed to achieve a network that gives the expected results. The learning consists of giving the system training data, where the network is provided an input, followed by the ANN providing an output. This output value is compared to the correct value, and the neural network then adjusts its internal values to reduce the chance of this mistake happening again.[6, 30, 32]

With the forward-propagating system as proposed in this thesis, the adjustments that can be made by the artificial neural network are to the weights and the biases. This can be done by a back-propagation adjustment, where depending on the result and how wrong the ANN was, the system backtracks from the output and adjusts the weight and biases for the neurons back towards the start up or down depending on how they contributed to the result.[29, 31]

To achieve this, a cost function is implemented to quantify how well the goal was achieved. The cost function is shown in Equation 5.7, where n is the total number of training inputs, X is the vector of all the outputs from the network when d is the input and y(d) is the desired output of the network. y(d) and X will be the same dimensions. w and b are the weights and biases respectively.[6, 32, 36]

$$C(w, b) \equiv \frac{1}{2n} \sum_{d} \|y(d) - X\|^2 \tag{5.7}$$

This cost function is often called the quadratic cost function or the mean square error. The cost function will always be a non-negative number and it will be a large number if the difference between the desired output and the calculated output is large while it approaches zero for a better performing network. From this cost function, the gradient ∇C can be decided by Equation 5.8.[6]

$$\nabla C \equiv \left(\frac{\partial C}{\partial v_1}, \ldots, \frac{\partial C}{\partial v_m} \right)^T \tag{5.8}$$

In this equation, ∇C is a vector with the gradient value for all the weights and biases v. This gives the neural network a number quantifying how each weight and bias should adjust to get a lower value in the cost function. This will also give a higher value for adjustment to the neurons that contribute more to a wrong answer, and a smaller adjustment to the neurons that are contributing to the correct answer.[6, 37]

Since the goal of the cost function is to reach a low value, the adjustment of the weights and biases must be the negative of the gradient value. The resulting expression for the change in values for the vector v containing all the weights and biases is shown in Equation 5.9, where η is the *learning rate* which determines the step size. Larger steps may reach a low value faster, but may overshoot the target and oscillate more.[6, 37]

$$\Delta v = -\eta \nabla C \tag{5.9}$$

When running such a learning structure the simulation calculates an average cost function based on all the training data, which is then used to calculate the gradient vector. This method ensures that all the training data is taken into consideration when deciding how to adjust the weights and biases. If this is not done, one adjustment of the weights and biases may only be good for reaching one type of result, and not for making the whole neural network better. By doing this many times with thousands or millions of training values, the system will gradually learn and improve the results.

5.5 ANN Model

There are many ways to develop an ANN model. The method explained in this chapter points the neural network in the direction of the closest local minimum, which is not necessarily the global minimum. This can result in the cost function reaching a point where all further adjustments decreases the accuracy, while the system is still not functioning as intended. This means that what initial values are placed on the weights and biases can have an effect on the result.[6]

5.5.1 Development and Specifications

The artificial neural network was developed and trained within MATLAB® using a combination of original code and existing functions. The first step of the process was simulating the microgrid in Simulink with the power flow algorithm as the centralised controller. Data from the simulation of the power system was registered at a rate of 1000 datapoints per hour and saved to a matrix containing the input and output values. This number was chosen based on several tests with varying number of datapoints until the performance increase of additional datapoints stagnated.

Following this, the data was separated into the three different datasets needed: one for training, one for validation and one for testing. The training data consisted of 70% of the values while the validation data and test data each consisted of 15% each. This was based on the middle ground of the most common distributions, which are 80/10/10, 70/15/15 and 60/20/20, where the largest percentage is the training set with equal sized validation and testing sets. The data was randomised once when saved from the simulation and once before training the ANN in order to make sure the ANN was trained by samples in a truly random order. By not doing this, there is a risk of the ANN results being sub optimal.

At this stage, the data has been properly processed, separated and ready for the actual training of the artificial neural network. The process of training the network is done with the Deep Learning Toolbox™ in MATLAB®, where several aspects of the ANN can be specified. Figure 5.4 shows the process of development. The loop inside the dotted box is done by the built in deep learning functions, while the two boxes on the outside is done manually.

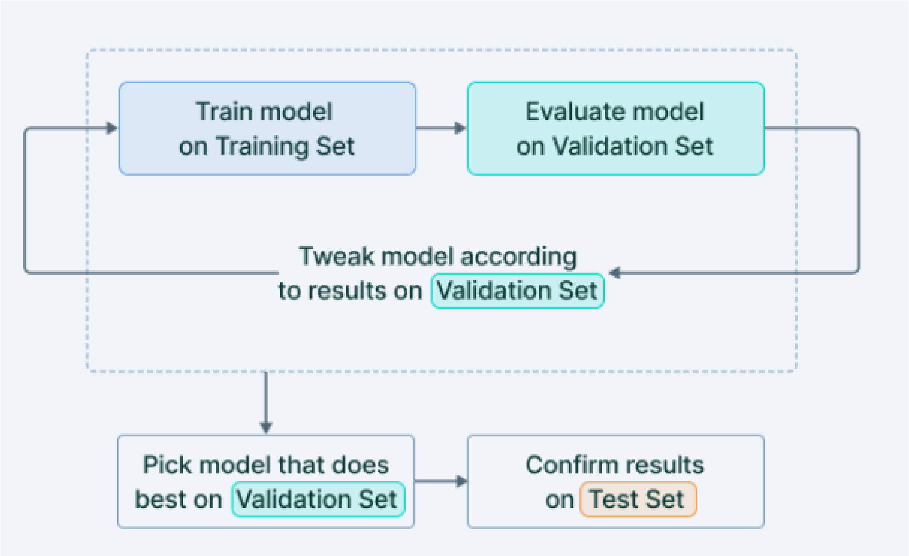

Train model on Training Set
Evaluate model on Validation Set
Tweak model according to results on Validation Set
Pick model that does best on Validation Set
Confirm results on Test Set

Figure 5.4: Training, validation and testing process of the ANN.[38]

As artificial neural networks are black-box models, the inner workings of the neural network is unknown. This means that one can get both similar and vastly different results from simple changes when training an ANN. In addition, as there is a wide range of applications for ANN, the optimal setup will vary greatly depending on the scope of the task. Because of this, the approach chosen for the development of the ANN was to begin with a set of basic variables and test many alternative structures to see which ones improve the results. The performance can be seen by comparing the output of the power flow algorithm and the ANN with the same input values as well as looking at the root mean square error (RMSE) to make sure it is low without overfitting.

In this project, more than 100 neural nets with different configuration properties were trained and tested. The chosen iteration of the neural net was one with three hidden layers with 16 neurons each, with a ReLu layer after each of the hidden layers. The net was trained with 24000 samples, with a learning rate of 0.1 resulting in an RMSE of 0.063. The training function used the stochastic gradient descent method (SGDM), and took 2 minutes to train. Figure 5.5 shows the training progress in MATLAB®.

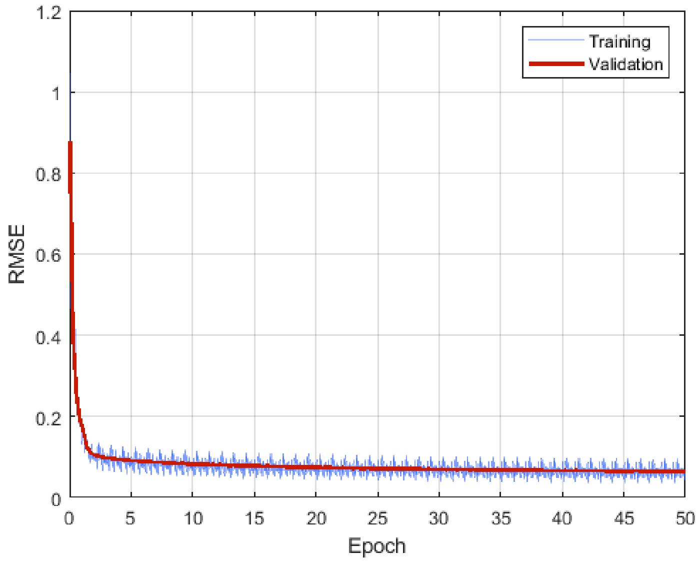


Figure 5.5: RMSE per epoch of the ANN training process

After training the net, the input values from the testing dataset were sent through the neural net. The resulting output values was then compared to the target values and the plotted in order to identify how large the general error of the neural net was from the test values. Figure 5.6 shows the chart, where each point is the difference between the target value and output value. As all output values are per unit, the difference will be a number between 1 and 0, where lower values implies an accurate prediction.

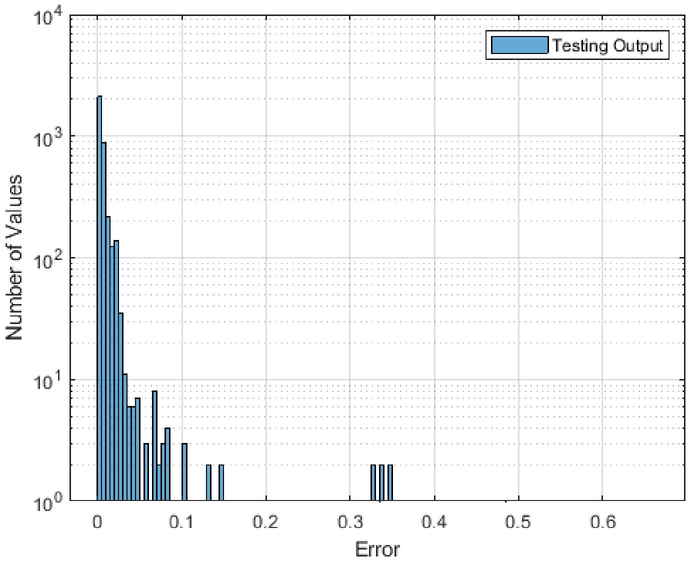


Figure 5.6: Logarithmic histogram of the error between neural net outputs and target values

5.6 Simulation Results

5.6.1 Case 1: Base Case

The following are the results from simulating the microgrid with the ANN as the centralised controller with normal irradiance and load. Figure 5.7 shows that the microgrid is able to provide enough power from the local production for most of the day. The grid switch is turned on (switch state at 1) at three times during the day, once briefly at around 8:00, once from 16-17 and from 19-21. The power reference to the VSC starts at around 10 kw with a trend of peaks in the morning at 07:30 and 17:00. At 8:00 the power reference does not reach zero, even though the grid is connected, and at 16:00 the power reference has a more gradual decrease down to zero. The power reference is mostly inverse proportional to the grid switch with the $P_{VSC,ref}$ going towards as the grid switch goes to 1.

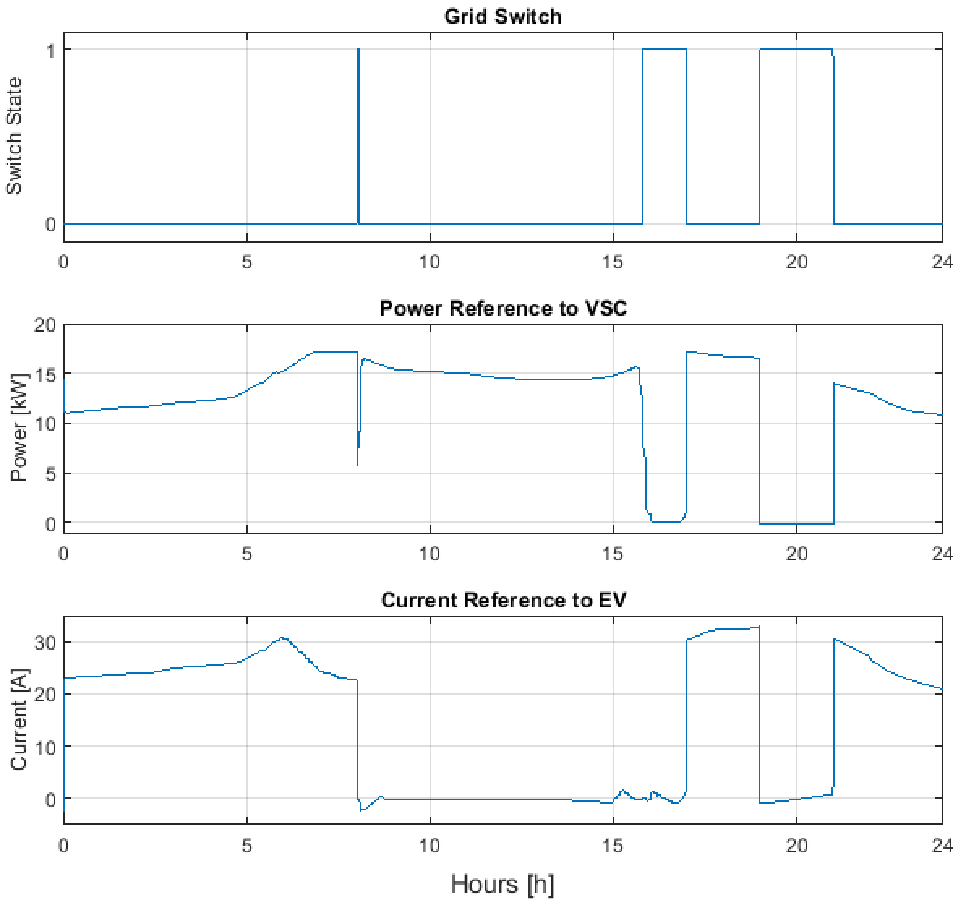


Figure 5.7: The output values from the simulation of case 1 with the ANN as the centralised controller.

Figure 5.8 shows measurements of the power flow to and from several of the components in the system. The Variable load receives between 10 and 16 kW over a day, with 10 kW being the base load as well as peaks in the morning end evening. The solar power output follows the trend of the irradiance. The EV and BESS power increases gradually with the load increase until approximately 06:00, where the solar power reduces the demand from the two batteries. The EV power is reduced to close to zero in the period that the EV is disconnected, but with some small fluctuations above and below zero. This also happens in the second period when the EV disconnects at around 19:00.

The BESS largely follows the EV power until the EV disconnects, and increases the power output to compensate for the missing EV. The power output is negative from the BESS at 16:00 and slightly at 08:00 the BESS charges, which aligns with the times where the grid is connected as can be seen in Figure 5.7. At the times of disconnection and connection to the grid, the BESS has some sharp peaks and bottoms before settling at the desired value.

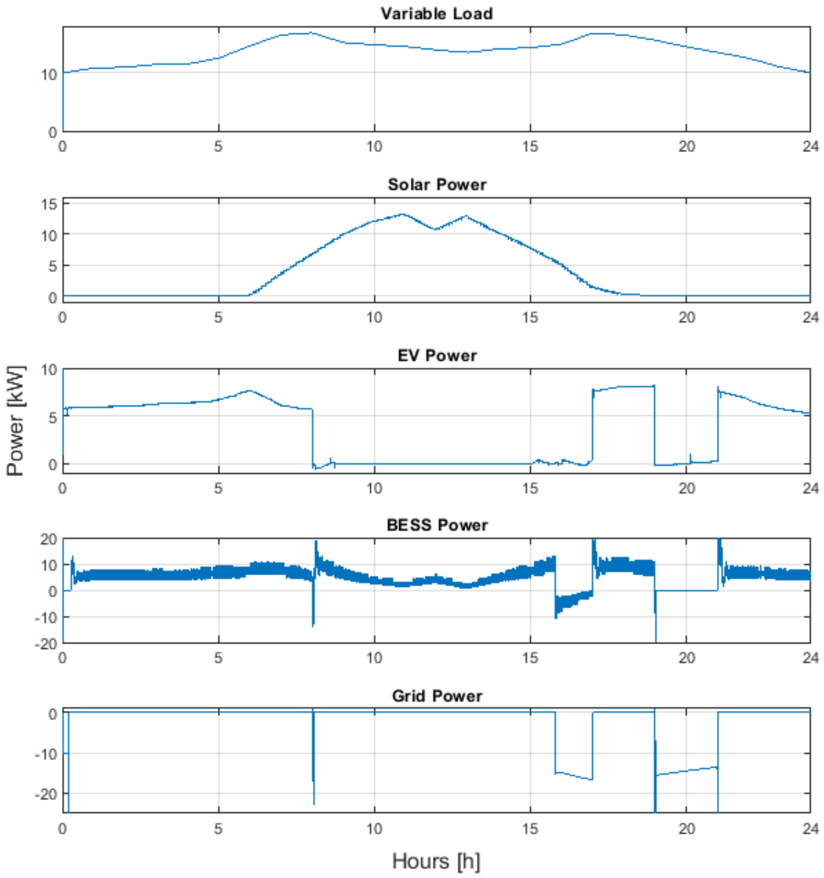


Figure 5.8: Measured values from the microgrid from the simulation of case 1 with the ANN as the centralised controller

.

When looking at the point of common coupling in Figure 5.9, the three phase voltage is consistently at approximately 400 volts, except for three peaks over the 24 hours. The first peak can be seen at the start, and the second and third can be seen to occur when the microgrid disconnects from the grid. Both the voltage at the PCC an the DC voltage are mostly stable, with ripples at the time of connection and disconnection to the grid. The current goes to zero when connected to the grid, as there is no power flow from the DC side to the AC side when grid connected. The DC voltage decreases rapidly to close to 500 V between 19:00 and 21:00.

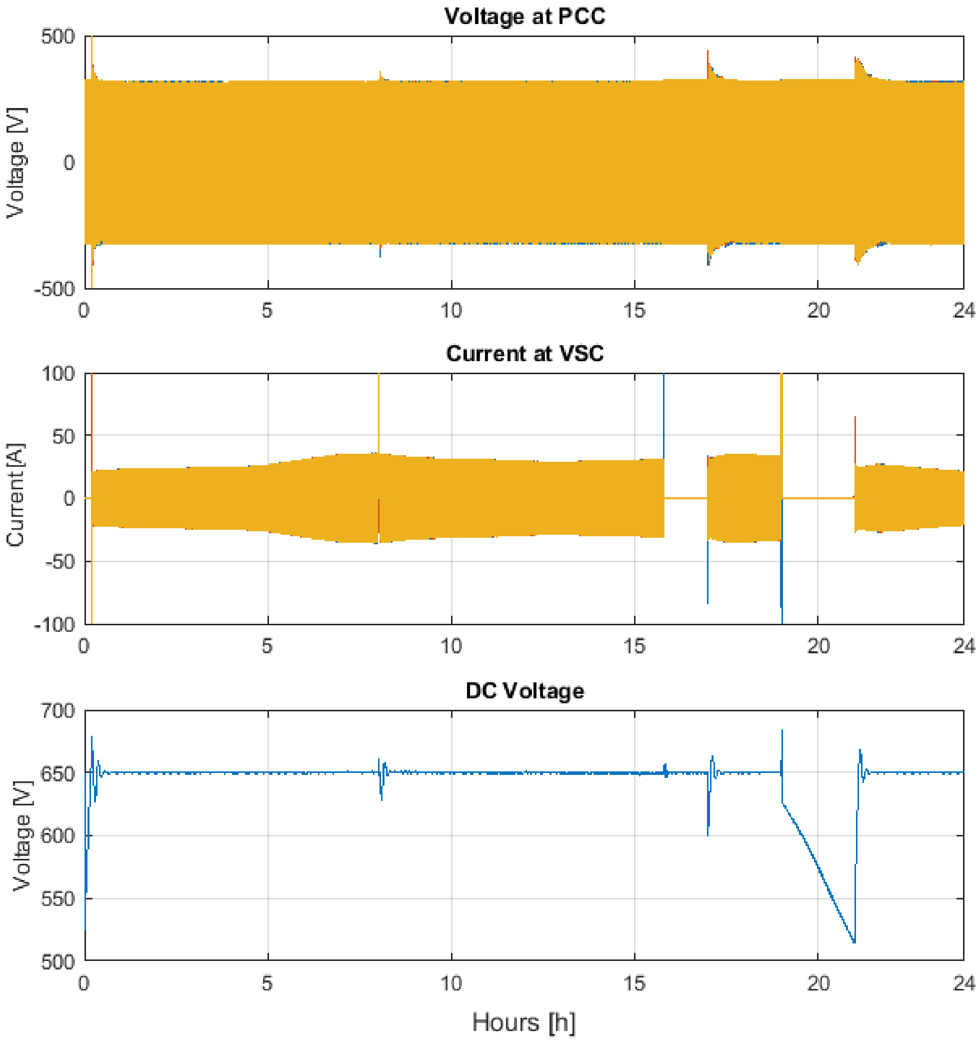


Figure 5.9: Measured values from the microgrid from the simulation of case 1 with the ANN as the centralised controller

.

5.6.2 Case 2: Irregular Solar Irradiance

When the irregular solar irradiance is applied, neural network decides to connect the grid four times. In addition to the three periods as in the base case, an added period in the middle of the day is included in this case as there is reduced local power production. The neural net manages to follow the trend of the power reference, by setting the value towards zero when grid connected. In this case, however, the actual power reference only reaches zero in the last period of connecting to the grid. The current reference follows the power reference trend, but as Figure 5.10 illustrates, the current reference changes between -8 A and 3 A instead of staying at zero as the EV is disconnected at this point.

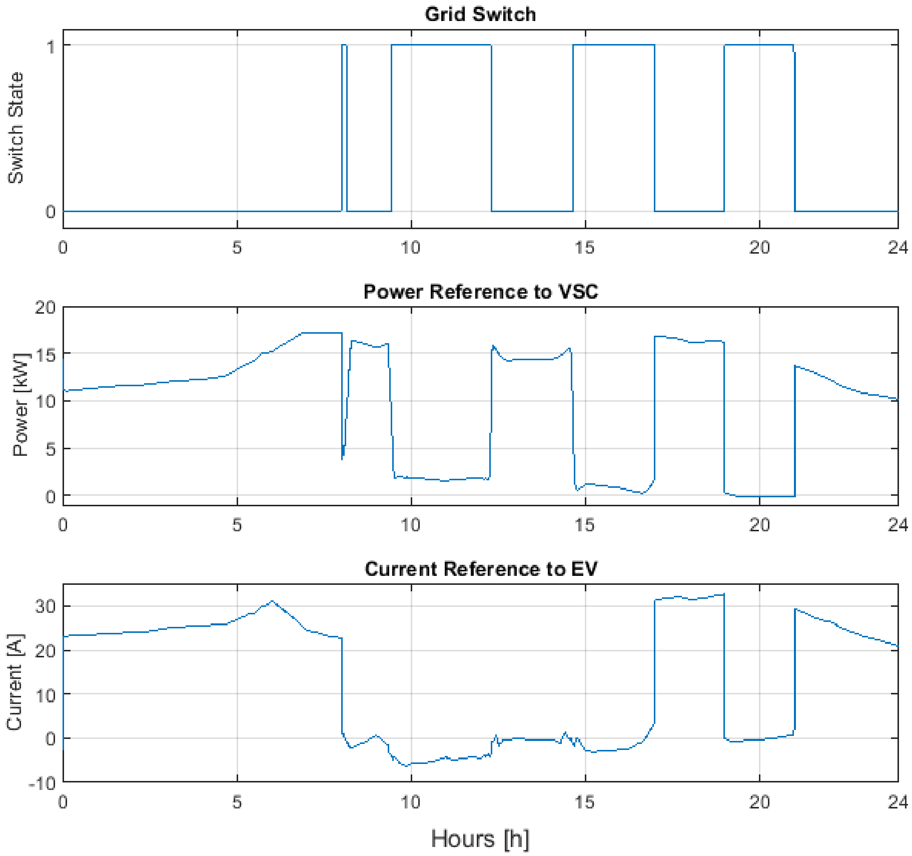


Figure 5.10: The output values from the simulation of case 2 with the ANN as the centralised controller.

Figure 5.11 shows the power flow of the elements of the microgrid. The solar power is greatly reduced compared to the base case between 09:00 and 13:00. The load is the same as the base case, and the EV power follows the EV reference from Figure 5.10. When looking at the power from the BESS and grid, they both follow the desired trend when switching between island mode and grid connected mode. The same two plots also show the BESS go slightly above its maximum discharge at 9 kW when connecting to the grid. A similar trend can be seen with the grid power, where there are high peaks around the connection and disconnection points of the grid switch.

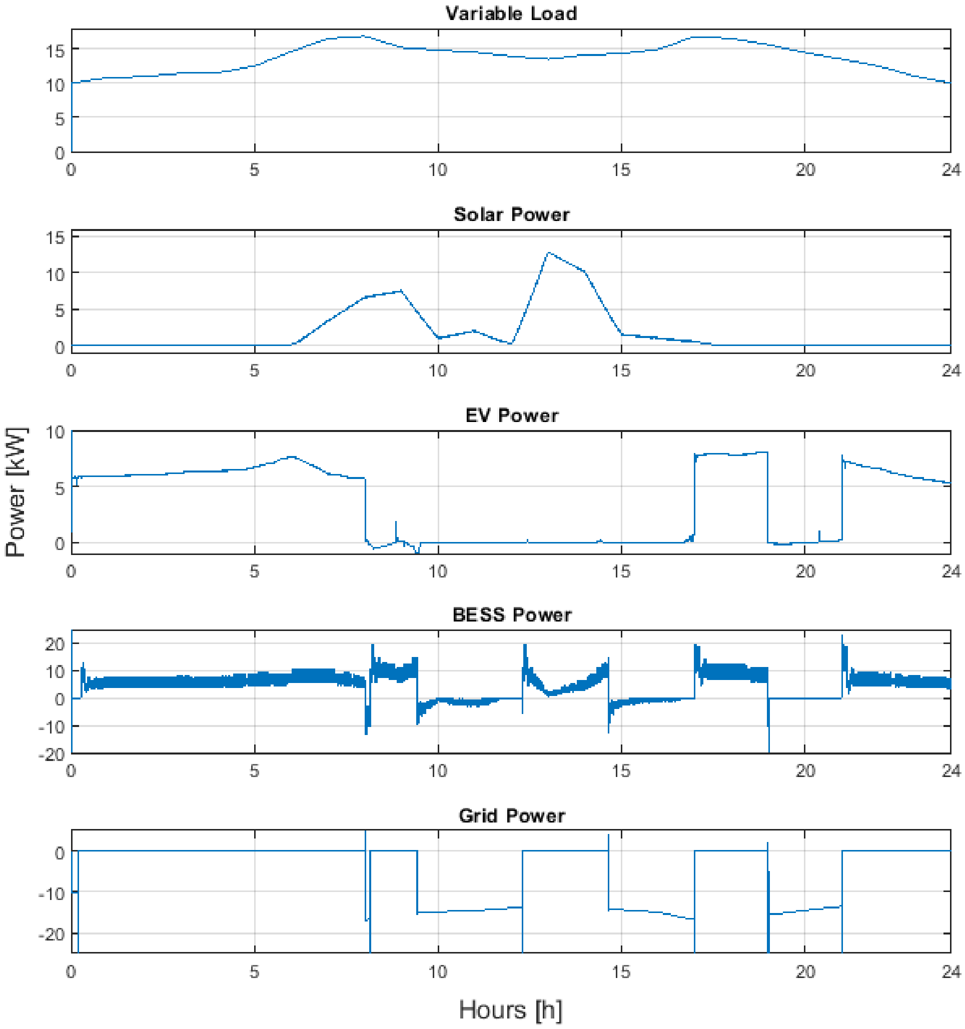


Figure 5.11: Measured values from the microgrid from the simulation of case 2 with the ANN as the centralised controller

.

The same trend as previously stated can be seen on the voltage at the PCC, and the current at VSC in Figure 5.12. The transient overvoltage occurs when disconnecting from the grid as well as overcurrents both at connection and disconnection. The DC voltage does also have a similar trend with a mostly steady level in both island mode and grid connected mode, but with some oscillations and sharp peaks at the time of switching before stabilising.

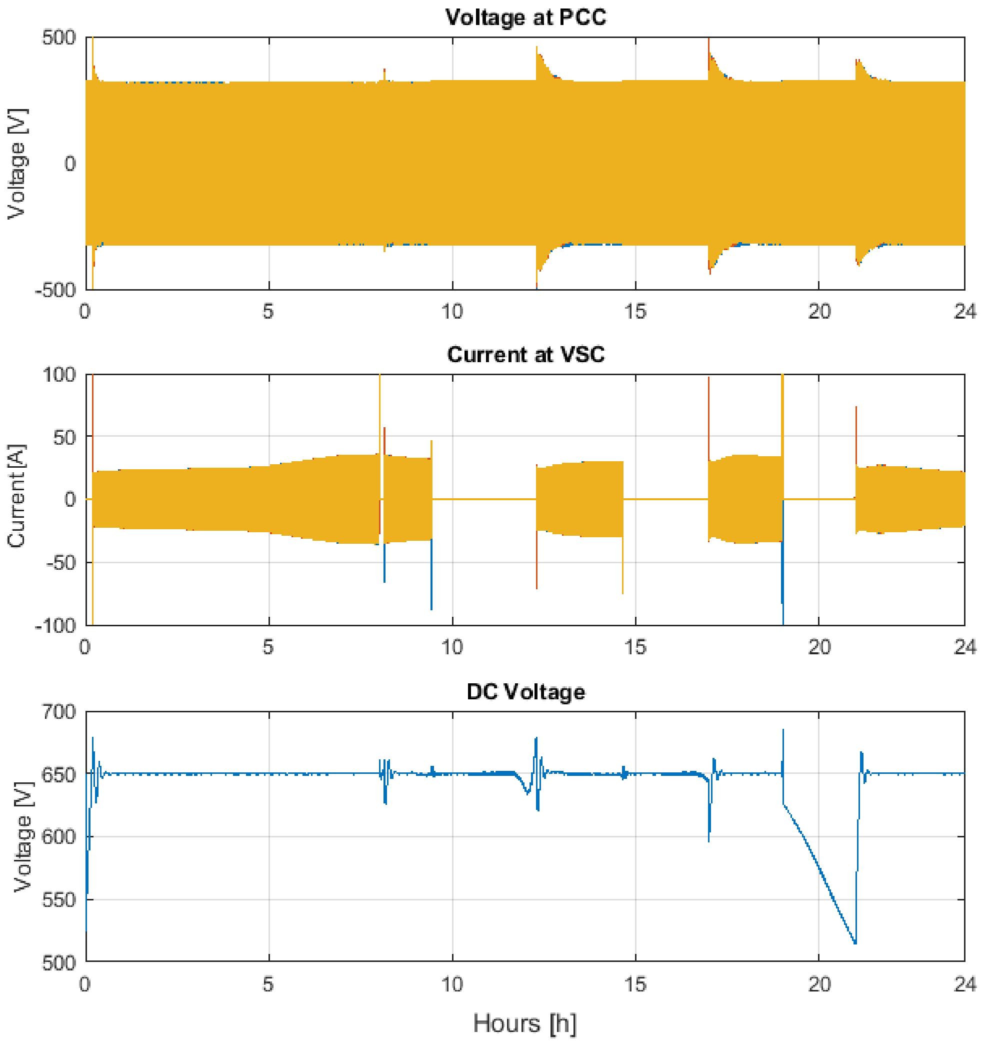


Figure 5.12: Measured values from the microgrid from the simulation of case 2 with the ANN as the centralised controller

.

5.6.3 Case 3: Irregular Load

As can be seen in Figure 5.13, the grid is connected for three periods, the first of which is slightly longer than previously seen in the other cases, as the load in the morning hours is higher. The power reference stays at around 15 kW in the middle of the day, even though the load is significantly lower in this case. In addition to this, the reference struggles to reach zero when the grid switch is turned on.

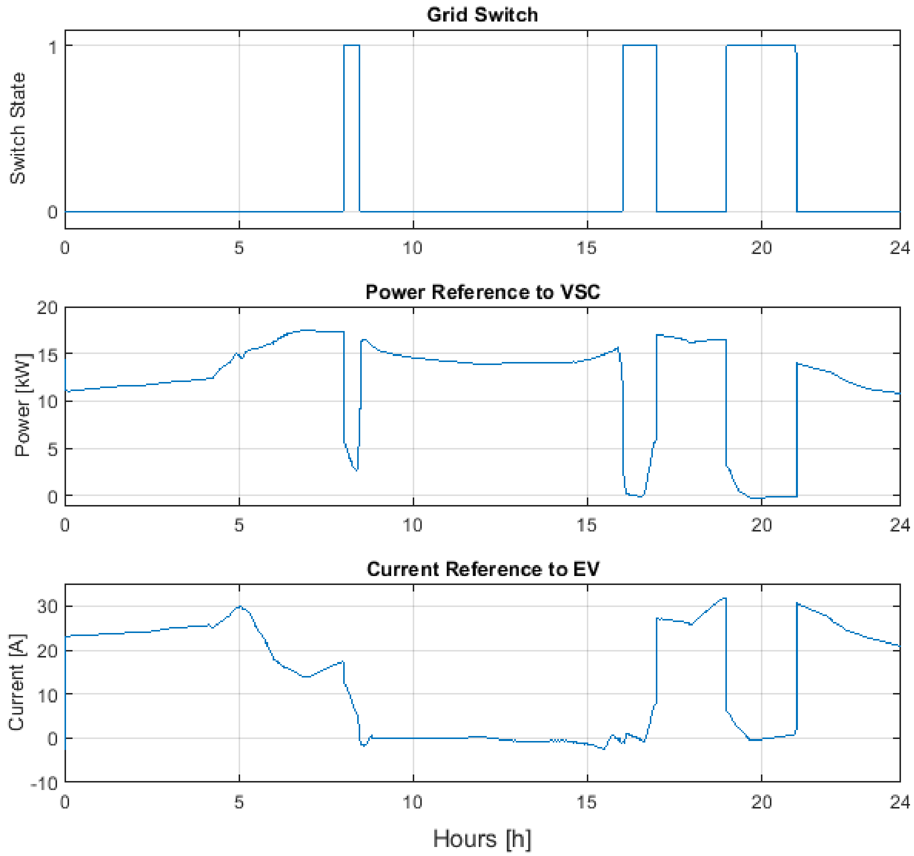


Figure 5.13: The output values from the simulation of case 3 with the ANN as the centralised controller.

Figure 5.14 illustrates how the power to the variable load and solar power is in this case. The EV power follows the reference value given, with some added transients. The BESS power increases above the output power limit at the same time of the load peaks, and the the grid power to the load experiences transients when switching the grid switch.

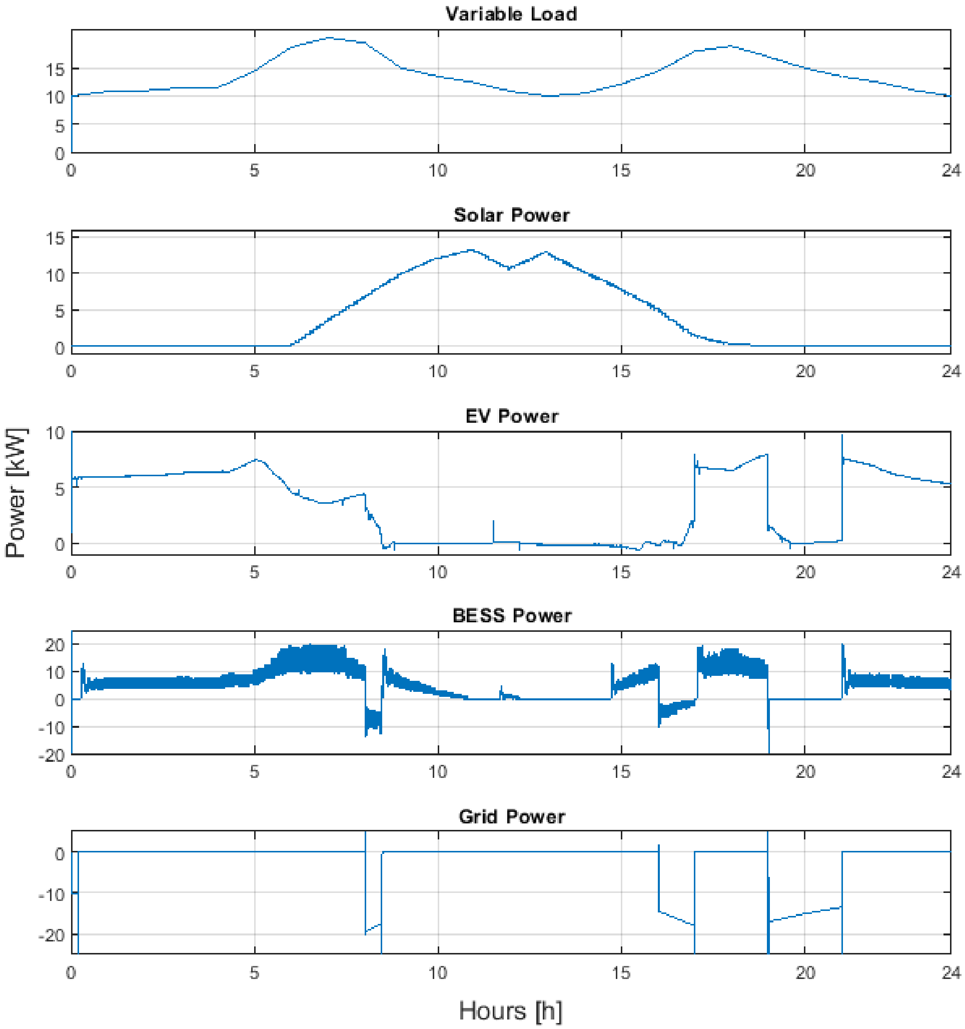


Figure 5.14: Measured values from the microgrid from the simulation of case 3 with the ANN as the centralised controller

.

The voltage at the PCC is quite steady at 400 V throughout the whole day, with small peaks when disconnecting the grid. Figure 5.15 shows that the DC system struggles to keep the DC voltage at a 650 volts, with it dipping below 600 V at 07:00 and 20:00 in addition to increasing to over 650 volts at 14:00.

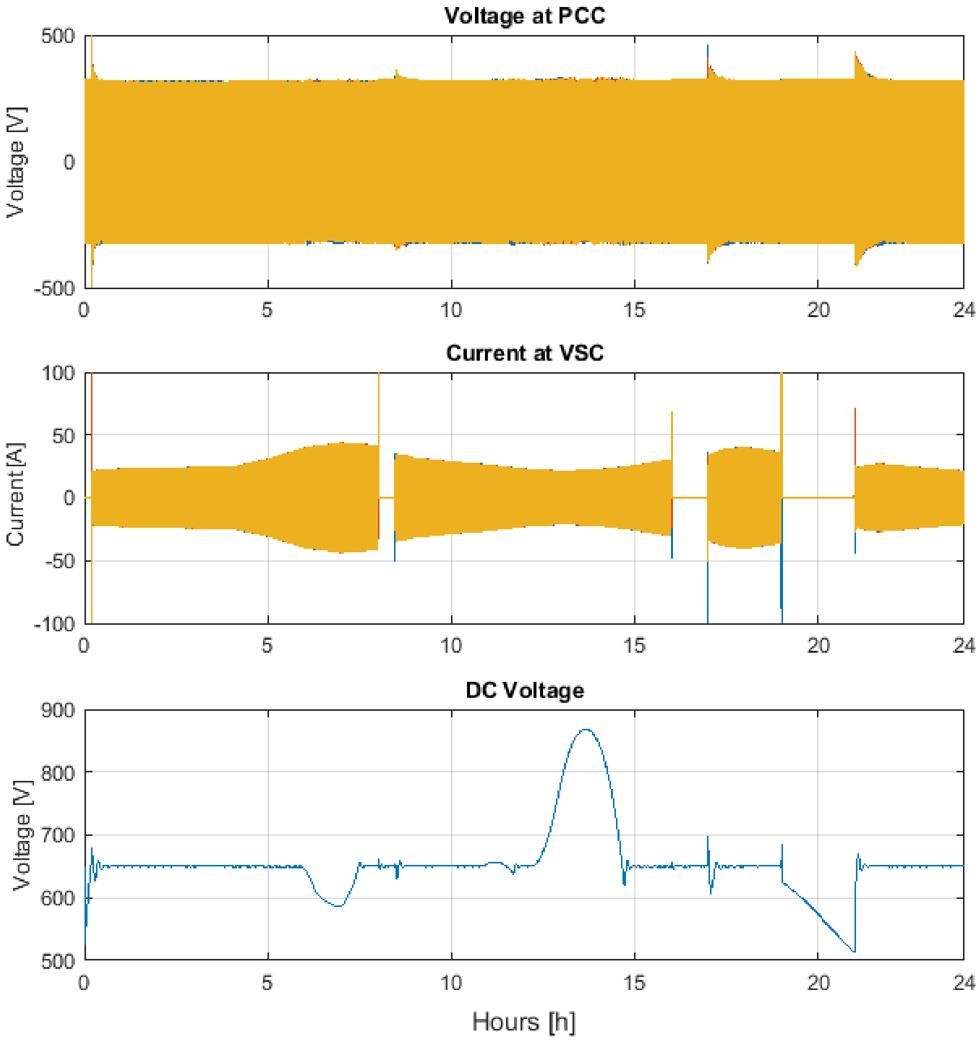


Figure 5.15: Measured values from the microgrid from the simulation of case 3 with the ANN as the centralised controller

.

5.7 Discussion of Results

There are many points of interest when looking at the development of the ANN. Firstly, the process of choosing the architecture and training must be considered. With the large variation in areas of use, determining the starting values and structure for the ANN can be demanding. As a consequence, the result and the performance of the ANN may be affected by what values were chosen as a start, and how many times the neural net was tested and changed before reaching the final iteration.

Finding the optimal configuration is difficult and time consuming, as a neural net with a low RMSE does not necessarily result in a better system across several cases. With each simulation of the microgrid taking from a couple of ours to one whole day to complete depending on the complexity of the neural net, time was a limiting factor in the iteration process. However, it is worth noting that the neural net does not need to be optimal or *perfect* in order to provide valuable data for the project.

The training process of the ANN was done in MATLAB®, but there may be other software or libraries for other programming languages that may allow for a better training process. PyTorch and TensorFlow are two such deep learning frameworks that may have provided a better result, but the use of Python in combination with Simulink® may also have resulted in a more time-consuming and problematic process.

5.7.1 ANN vs Power Flow Algorithm as a Centralised Controller

As the main objective of this thesis is to assess how an AI based centralised control system for a microgrid performs, a comparison between the methods simulated in this thesis is needed. First, the output values from the controller is compared between the two methods in the two cases. In addition to this, a comparison of the measured values from the microgrid is shown to illustrate how the performance of the centralised controller affects the rest of the system.

Comparison of Output Values

Figure 5.16 shows a comparison of the switch state and $P_{VSC,ref}$ for the three cases. The blue plots are the output from the ANN based control, and the orange plots are from the power flow algorithm based control.

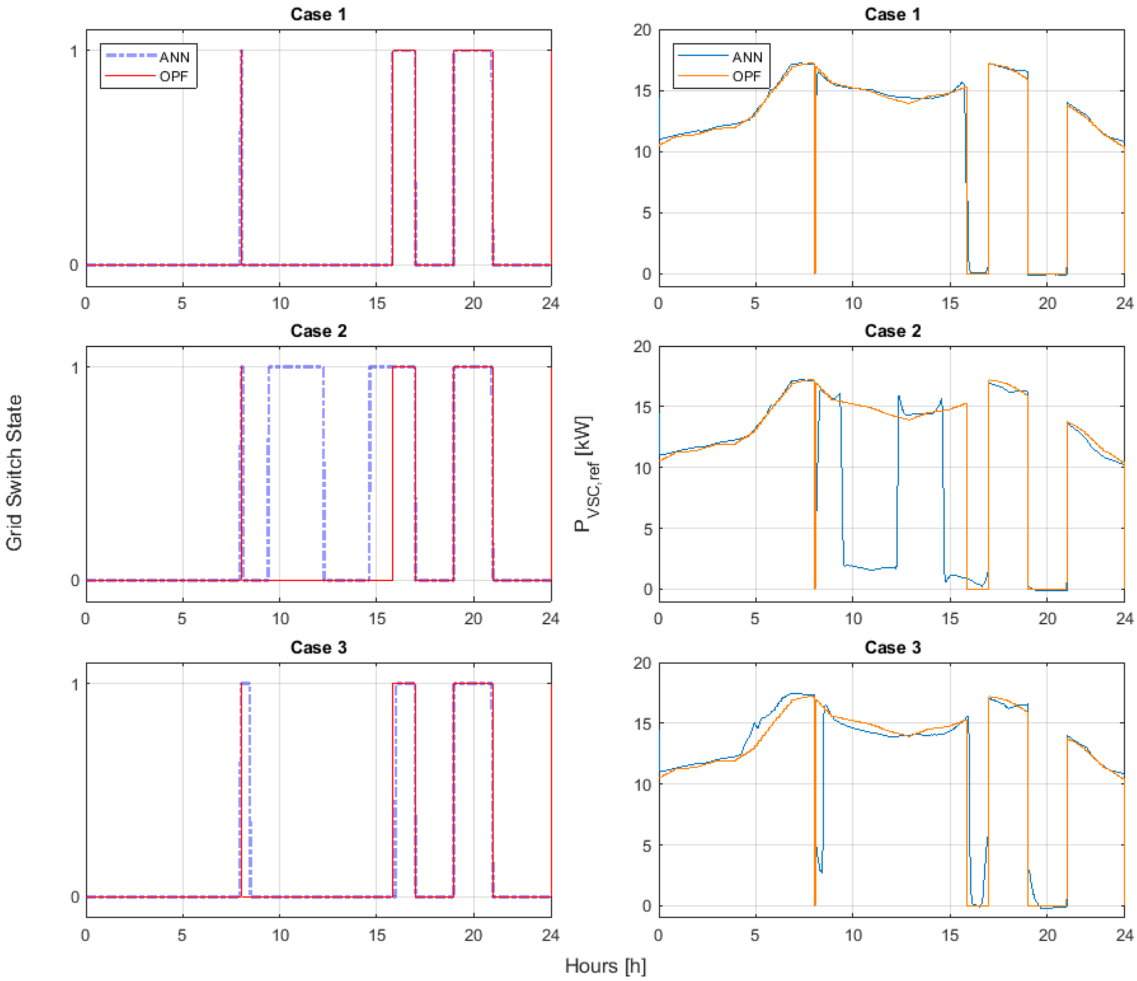


Figure 5.16: The output values for Grid Switch and $P_{VSC,ref}$ for all the simulations.

One general point to take away from the power flow algorithm was the inability of providing a different result based on the input. This was considered and implemented in the programming of the algorithm, but did not work as intended. As a result, the switch state and power reference to the VSC from the power flow algorithm is equal in all three cases. The ANN was trained solely on values from simulating the base case in order to be able to test the two other cases on two systems that was set up on the same basis. Subsequently, the ANN and OPF switches the grid switch at the exact same time, while $P_{VSC,ref}$ is very close to similar.

However, when moving to the second case where the solar irradiance is reduced in the middle of the day, the difference between the two methods are quite pronounced. As the power flow algorithm does not register the reduced local production, it fails to identify the need for the main grid in order to achieve power balance. Conversely, the ANN based control does identify the need for added power in the two periods of reduced solar power, resulting in the grid being connected for four additional hours.

With the ANN correctly recognising the need of help from the grid to supply additional power, and in turn adjusting the reference value towards zero, the use case of ANN based control is illustrated. In all the training, testing and validation data, the microgrid operates in island mode in this period of the day. In spite of this being a completely new situation that the network of neurons never has seen before, a pattern is recognised and the ANN gives an acceptable output.

The last case also proves that the ANN manages to identify that the load is higher than in the base case. It must be noted that even though the trend aligns with the load profile of case three, the scale is not correct. The peak of the load is around 20 kW and the lowest point in the middle of the day is 10 kW. In comparison, the peak of the $P_{VSC,ref}$ is around 17.5 kW and the bottom at 14 kW, exhibiting how the ANN manages to get the trends correct, but struggles to accurately get values that are far away from the base case.

The outputs are, however, not perfect as the $P_{VSC,ref}$ never reaches zero in case two for the two periods of grid connection in the middle of the day. This may be an effect of overfitting, where the system is trained to be too similar to the base case, resulting in reduced accuracy in other cases. This can likely be reduced by further tuning of the system as well as using training data from several cases instead of just one.

At this point, the system is generally well versed to handle cases that are similar to the base case, with the accuracy decreasing for the cases with a larger variety. This will be an important element to take into consideration in further development of such systems, as a live AI based control system is expected to perform in a similar manner in all simulations.

Comparison of microgrid values

An important factor that needs to be assessed in such a system as in this thesis is the voltage level at the point of common coupling. Figure 5.17 shows V_{pcc} for all the cases with the two different control systems.

One element that can be seen in all the plots is the sharp peak in voltage that occurs at the time of disconnecting the microgrid from the main grid. As this is not an element that is affected directly from the centralised controller, but rather from the SUDC it does not affect the results of this thesis. The existing research available on resynchronisation of SfUDCs can help mitigate such signals.

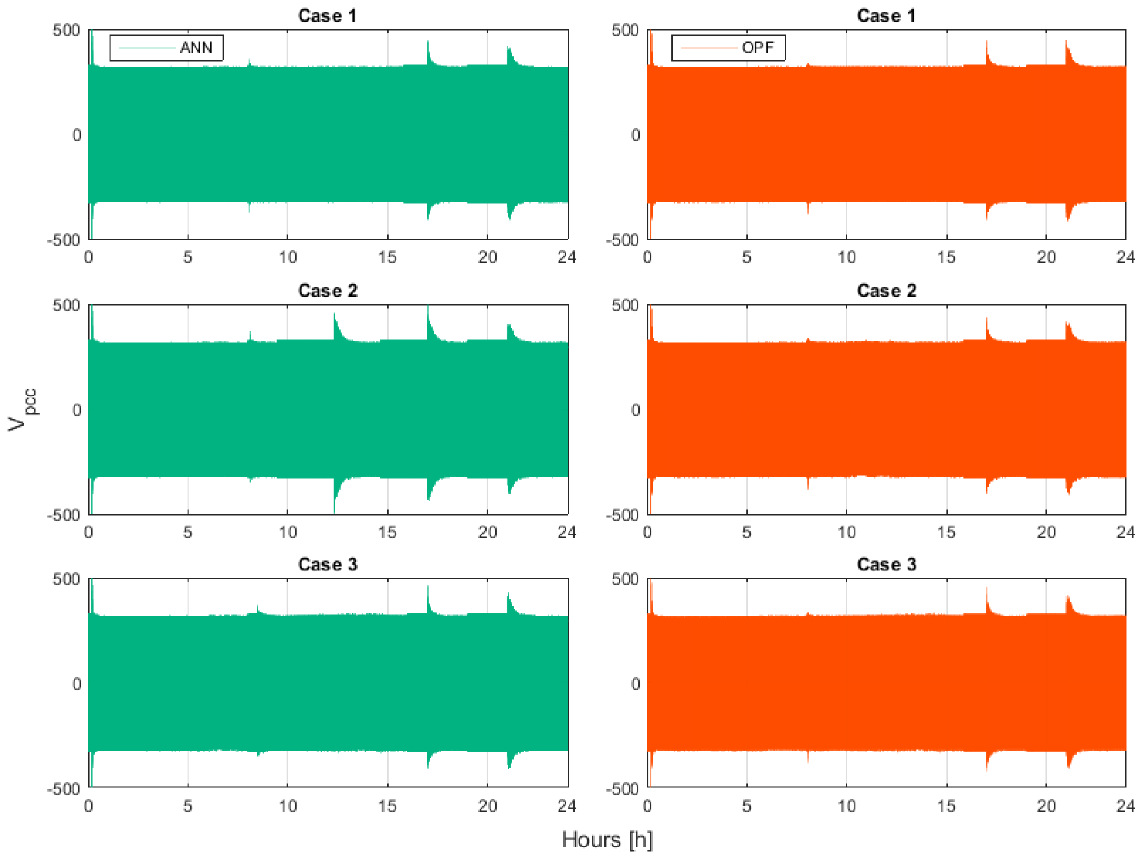


Figure 5.17: The measured V_{pcc} from all the simulation cases.

In general, the V_{pcc} is quite similar in all the cases, except for the peaks generated from the additional switching of the grid switch in case two. In addition to this, there is slightly less noise on the ANN controlled cases. This does not necessarily mean that the centralised controller is working perfectly in any of the cases, as other secondary controllers can have an effect on the general voltage level of the system. If the SUDC is operating in droop mode, the controller will compare V_{pcc} to a reference value, thereby adjusting the voltage level towards the desired value even if the $P_{VSC,ref}$ is slightly wrong. Additionally, the BESS functions as a voltage controller at the DC bus which also can affect the voltage level at the PCC.

6 Conclusion

With the rapidly changing power system structure, the need for reconsidering how these systems are controlled increases. In this project, a microgrid model of a hybrid microgrid was constructed, containing common components such as local production, loads and storage in addition to the required converters. A power management based power flow algorithm was created to serve as a baseline for a centralised controller, in addition to provide training data for an ANN based centralised controller. The two controllers were simulated for three cases each: one base case, a case with variable irradiance and a case with variable load.

In general, the results from the work completed in this thesis confirm that there are large possibilities within the field of AI based power system control, and especially with the use of artificial neural networks. These neural nets show the ability to identify patterns and situations that are unknown or new to the system, and provide a realistic and expected output. The performance of the ANN based control proved to be superior to the power flow algorithm in the two cases with irregular load and solar irradiance, confirming the theory that the implementation of ANNs can improve performance. + Even though the general concept and motivation was confirmed for this thesis, the test was only conducted for a limited simulation period. Subsequently, there are numerous elements needing consideration prior to the ANN being implemented in a real life scenario in order to ensure a predictable and stable performance over a variety of cases.

Firstly, the training procedure conducted in this project used data from the simulation of one day. This resulted in an ANN with a satisfactory performance on the base case, but with a decreasing accuracy with cases that are less similar to the base case. Secondly, the power flow a algorithm did not manage to account for the SOC in the batteries, resulting in a substantial net discharge of the two storage elements in the microgrid.

All things considered, the project achieved its main goal which was to propose, test and assess the performance an ANN based control development of a microgrid system. The results have highlighted the promising prospect of ANN based control systems, while also identifying multiple challenges and considerations of this research field, allowing for further development and research.

6.1 Future Work

With regards to future research, multiple elements of this thesis that can be continued or improved in order to further develop the knowledge and understanding on AI based control systems for microgrids. As for the ANN, several features have large potential for further development. The first of which is the basis for the training data used for training the ANN. In this thesis, an algorithm was created and used on the simulation of the microgrid with the output and input values of that simulation being used as the training data. Unfortunately, the algorithm did not work as expected, and should therefore be a focus area for further improvement. Even though a comparison between the methods are possible even with sub-optimal results, it does not allow for an assessment of the absolute performance of the methods.

In addition to improving the algorithm for the accumulation of training data, a consideration of the contents of the training data should be completed. This thesis confirms some of the functionality, but with a reduced overall quality. By making a dataset containing the resulting values of simulations of several cases, the performance of the ANN controller with unknown situations may be increased. This may also reduce overfitting, as the dataset contains values from a variety of cases.

With regards to the two storage elements in the microgrid in this thesis, several aspects can be reconsidered and improved. This thesis did include a simple, and partly unrealistic battery deployment scheme. Further work should look at how artificial neural networks can be used to implement a smarter and more optimal battery scheme. This can include battery deployment based on energy costs, net zero discharge and charge during a certain period or to be used as a peak shaving mechanism in order to reduce the maximum load in the system.

This thesis uses a feed forward neural network with three hidden layers. There are, however, several other types of neural networks that could be considered and tested with such a microgrid such as the recurrent neural network. In addition to looking at other types of neural network as a centralised controller, neural networks can also be considered used on the secondary controllers. This thesis has illustrated that the performance of the microgrid system is dependent on several elements, and as the same perks of the ANN structure applies to the secondary controllers, this is an area that could be looked more into.

A The Simulink Model

A.1 Overview

Figure A.1 shows a screenshot of the general Simulink model used for the simulations. The microgrid is on the lower half and two control sub-systems on the top. The left side of the microgrid is the DC side, with the three DC components highlighted in red. The orange block is the inverter, and the green block is the LCL filter. The two switches are shown in grey, in addition to the main grid. Lastly the AC load is shown in the red block on the right side. The main controller and the SUDC are highlighted in orange.

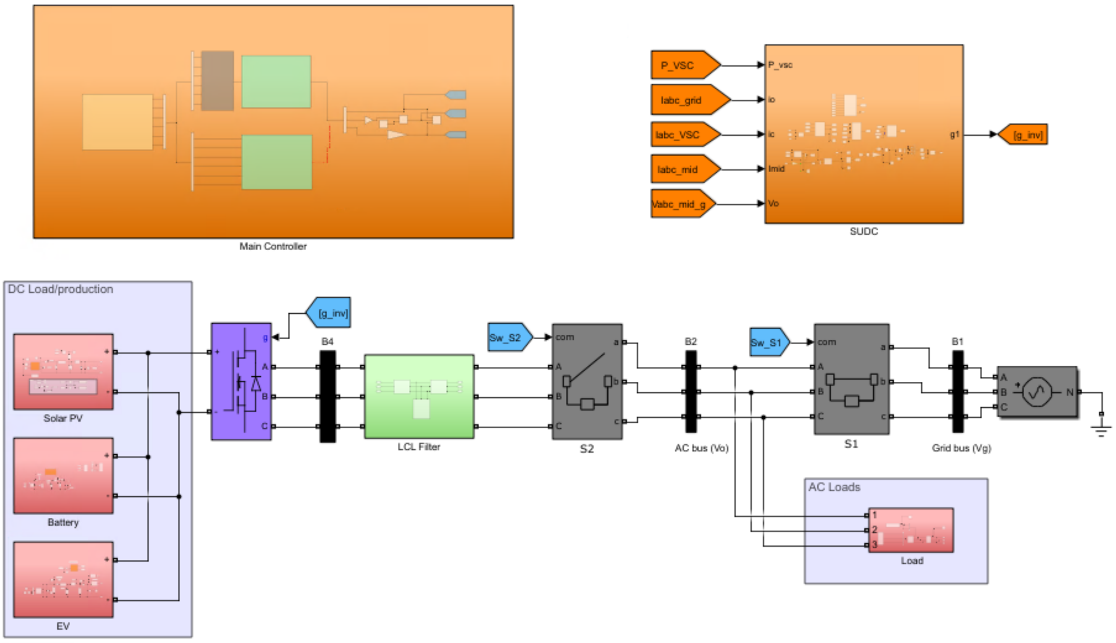


Figure A.1: The model of the simulated microgrid.

A.2 Main Controller

Figure A.2 shows the main controller sub-system. The input values are in the yellow sub system and the two green blocks are the controllers. The top one is the ANN based central controller and the bottom one is the power flow algorithm based centralised controller.

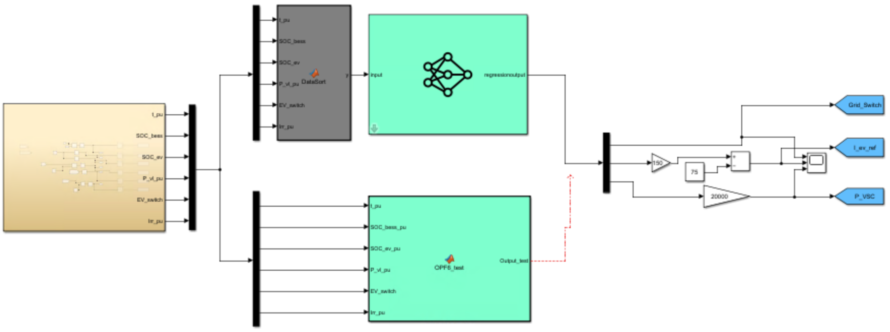


Figure A.2: The two main controllers used in the simulation model.

A.3 DC subsystems

Figure A.3, A.4 and A.5 show the the three DC subsystems.

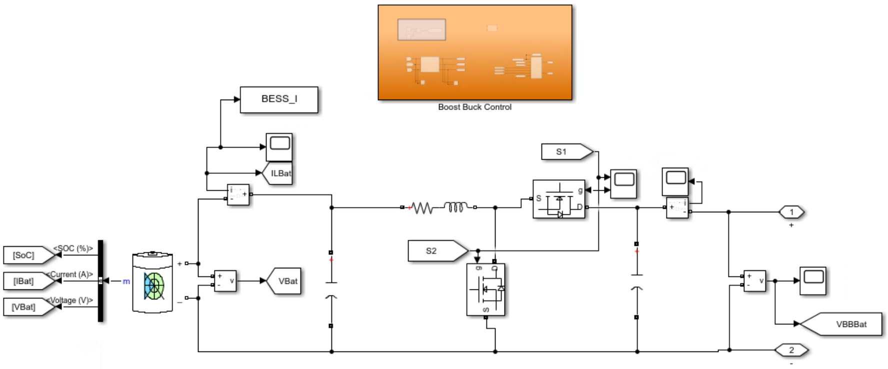


Figure A.3: The simulation model of the BESS subsystem.

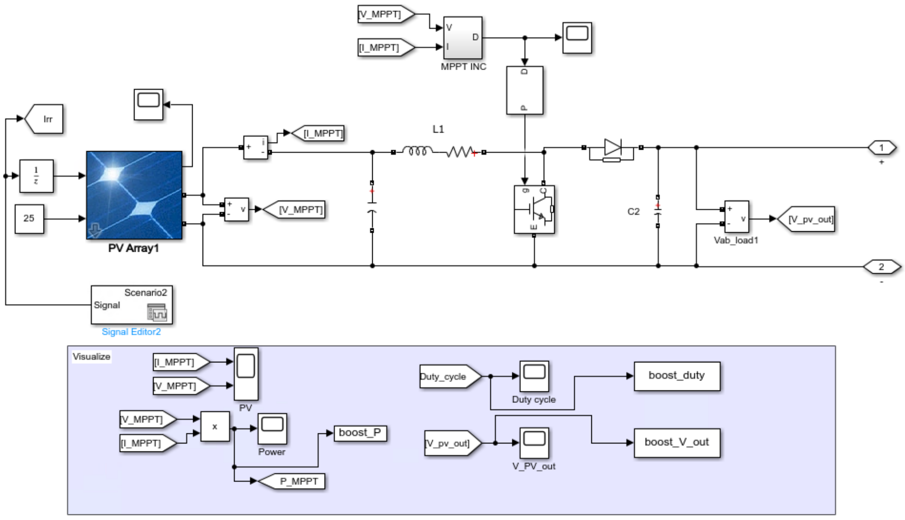


Figure A.4: The simulation model of the PV subsystem.

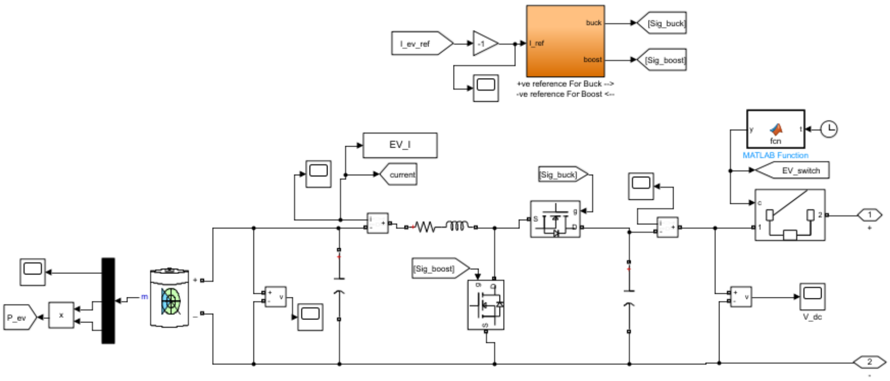


Figure A.5: The simulation model of the EV subsystem.

B Load and Irradiance Values

Table B.1: The values for the load [kW] and the irradiance [W/m^2] for the three cases

	Base Case		Case 2		Case 3	
Hour	**Load**	**Irradiance**	**Load**	**Irradiance**	**Load**	**Irradiance**
1	10	0.0	10	0	10	0
2	10.8	0.0	10.8	0	10.8	0
3	10.95	0.0	10.95	0	10.95	0
4	11.45	0.0	11.45	0	11.45	0
5	11.5	0.0	11.5	0	11.5	0
6	12.5	27.8	12.5	27.83	14.5	27.83
7	14.6	243.3	14.6	243.26	18.7	243.26
8	16.5	421.9	16.5	421.94	20.4	421.94
9	16.85	593.7	16.85	470	19.5	593.66
10	15.18	710.8	15.18	100	15	710.82
11	14.82	775.6	14.82	160	13.5	775.56
12	14.5	638.9	14.5	40	12.4	638.94
13	13.9	757.0	13.9	755	10.8	756.98
14	13.5	618.4	13.5	610	10	618.44
15	14.1	493.6	14.1	130	10.5	493.59
16	14.3	345.4	14.3	100	12.16	345.43
17	14.9	131.9	14.9	131.92	14.43	131.92
18	16.8	48.7	16.8	48.7	18	48.7
19	16.5	15.8	16.5	15.81	18.9	15.81
20	15.6	0.0	15.6	0	17	0
21	14.46	0.0	14.46	0	15	0
22	13.5	0.0	13.5	0	13.5	0
23	12.5	0.0	12.5	0	12.5	0
24	11	0.0	11	0	11	0

C Power Flow Algorithm

```
function Output_test   = OPF6_test...
(t_pu, SOC_bess_pu, SOC_ev_pu, P_vl_pu, EV_switch, Irr_pu)
%Input - This timesteps values from the microgirid.
%Output - Iev_ref, Pvsc,ref and Grid switch based on predictions.

%Initialize values
Ts = 5e-6 ;
delta_t = Ts ;
delta_t_hour = delta_t;
%Base to real values
t = t_pu*24   ;  %To simulate 24 hours. 1 second simulation per hour
SOC_ev_0 = SOC_ev_pu *100 ;  %State of Charge EV-Battery
SOC_bess_0  = SOC_bess_pu * 100 ; %State of Charge Battery
P_vl = P_vl_pu * 20000 ; % Power from variable load
Irr_0 = Irr_pu *1000 ; %Solar irradiance
step = 10 ;


%Constraints
SOC_max = 99 ;
SOC_min = 1;
S_VSC_max = 20000 ;
P_batt_max = 9000; %Same limits for both batteries
P_batt_min = -9000 ; %Same limits for bothe batteries
Batt_cap = 250 ; %

%Base values
Vn = 650 ; %V
Sb = 20000 ; %VA
fb = 50 ; %Hz
Vb = (1/sqrt(3))*Vn ; %V
Ib = (2/3) * (Sb/Vb) ; %A
wb = 2*pi*fb ;
Zb = Vb/Ib ;
Cb = 1/(wb*Zb);

```

```
Negative_trigger = 0 ;
Positive_trigger = 0 ;


%Predict PV
P_pv_max = 220 ;
Ns = 8 ;
Np = 10 ;
P_pv  =  (Irr_0/1000) *P_pv_max * Ns * Np ;

%Loads
P_cl = 0 ; % implemented the constant load into the variable load
P_losses = Ib ^2 * 0.4 ;  %Zb

%Create load curve for solar irradiance based on step time
%if t < Ts
    Load_c = Load_calc(Ts, step);
    %figure(1)
    %plot(1:size(Load_c,2),Load_c);
    %f = size(Load_c,2);

    Irr_c = Irr_calc(Ts,step) ;
    %figure(2)
    %plot(1:size(Irr_c,2),Irr_c);
    %f = size(Irr_c,2);
%end



function [Load_curve] = Load_calc(Ts, steps)

load1 = [0.8, 0.15, 0.5, .05, 1,  2.1, 1.9, .35, -1.67, -.36,...
-.32, -.6, -.4, 0.6, .2, .6, 1.9, -.3, -.9, -1.14, -.96, -1, -1.5, -1] ;
init = 10 ;
Time = 0 ;
Load_curve = zeros(1,24*steps) ;
p = 2 ;
for a = 1:24
```

```
    for b = 1:steps
        if a ==1 && b==1
            Time = Time + Ts ;
            Load_curve(p-1) = init ;
        else
            Time = Time + Ts ;
            Load_curve(p) = Load_curve(p-1) +  load1(a)/steps;
            p = p + 1 ;
        end
    end
end

end

%Creates an array of estimated irradiance with as many steps as specified
function [Irr_curve] = Irr_calc(Ts, steps)

irr2 = [0, 0, 0, 0, 0, 27.83, 215.43, 178.68, 171.72, 117.22, 64.74,...
-136.62, 118.04, -138.04, -124.85, -148.16, -213.51, -83.22, -32.89, -15.81, 0, 0, 0, 0]

Time = 0 ;
Irr_curve = zeros(1,24*steps) ;
p = 2 ;
for a = 1:24
    for b = 1:steps
        if a ==1 && b==1
            Time = Time + Ts ;
            Irr_curve(p-1) = 0 ;
        else
            Time = Time + Ts ;
            Irr_curve(p) = Irr_curve(p-1) +  irr2(a)/steps;
            p = p + 1 ;
        end
    end
end

end
```

```
%-------------------simulation functions (predictions for t+dt)
%% What to do when p_net is positive
function [positive_out] = positive()
%Divide unused power equally between battery and EV
%Give to only one if other is full or disconnected

if SOC_bess < SOC_max && SOC_ev< SOC_max && EV_switch == 1
    P_bess = (-P_PvMinusLoad)/2 ;
    P_ev = (-P_PvMinusLoad)/(2) ;
elseif SOC_bess<SOC_max && (SOC_ev==SOC_max || EV_switch == 0)
    P_bess = -P_PvMinusLoad ;
    P_ev = 0 ;
elseif SOC_bess==SOC_max && SOC_ev<SOC_max && EV_switch == 1
    P_bess = 0;
    P_ev = -P_PvMinusLoad;
else
    P_bess = 0 ;
    P_ev = 0 ;
end


%Check if the distributed power violates any limits.
%When P_net is positive, there will only be charging, so negative values
if P_ev < P_batt_min
    if P_bess < P_batt_min
        %If both batteries take too much power, set it at
        P_excess = - (P_ev + P_bess - (2*P_batt_min)); %This will be the power...
        %not able to use
        P_ev = P_batt_min ;
        P_bess = P_batt_min ;
        %Note = P_excess + " is unused in DC system" ;
    else
        P_excess = - (P_ev - P_batt_min) ; %This is excess power in system
        P_ev = P_batt_min ;
        %Note = P_excess + " is unused in DC system" ;
    end
```

```
else
    if P_bess < P_batt_min
        P_excess = - (P_bess - (P_batt_min)); %This will be the power...
        %not able to use
        P_bess = P_batt_min ;
        %Note = P_excess + " is unused in DC system" ;
    else
        P_excess = 0 ;
        %Note = "The system is operating within its limits" ;
    end
end


positive_out =  {P_ev, P_bess, P_excess}  ;

%positive_out_string = {P_ev, P_bess, P_excess, Note} ;
%positive_out = cellstr(positive_out_string);
end

%% what to do when p_net is negative
function [negative_out] = negative()

%Divide power need between battery and EV (edit V6) only if the EV
%and bess has more than 30% SOC
%Take from only one if other is empty or disconnected

if SOC_bess > SOC_min && SOC_ev > SOC_min && EV_switch == 1
    if SOC_ev > 30
        if SOC_bess > 30
            P_bess = -P_PvMinusLoad*(1/3) ; %Will be positive values
            P_ev = -P_PvMinusLoad*(2/3) ;
        else
            P_ev = 0 ;
            P_bess = 0 ;
        end
    else
        if SOC_bess > 30
            P_bess = -P_PvMinusLoad ; %Will be positive values
            P_ev = 0;
        else
            P_ev = 0 ;
            P_bess = 0 ;
        end
    end

elseif SOC_bess > SOC_min && (SOC_ev <= SOC_min || EV_switch == 0)
    P_bess = -P_PvMinusLoad ;    %Will be positive
    P_ev = 0 ;
elseif SOC_bess <= SOC_min && SOC_ev > 30 && EV_switch == 1
    P_bess = 0;
    P_ev = -P_PvMinusLoad;
else
    P_bess = 0 ;
    P_ev = 0 ;
end

%Check if the distributed power violates any limits.
%Since P_net is negative, there will only be discharging
%If power need from batteries are larger than max, we need to connect
%to grid to provide enough power for the loads.
%This would also trigger charging from solar power
if P_ev > P_batt_max
    if P_bess > P_batt_max
        %If both batteries provide too much power, set it at max
        P_need = (P_ev + P_bess - (2*P_batt_max)); %This will...
        %be the power not able to provide
        P_ev = P_batt_max ;
        P_bess = P_batt_max ;
        %Note = P_need + " #1 is needed in the system" ;
    else
        P_need = (P_ev - P_batt_max) ; %This is needed extra power in system
        P_ev = P_batt_max ;
        %Note = P_need + " #2 is needed in the system" ;
    end
else
    if P_bess > P_batt_max
```

```
        P_need = - (P_bess - (P_batt_max)); %This will be te needed extra power
        P_bess = P_batt_max;
        %Note = P_need + " #3 is needed in the system" ;
        %Note = "TEST" ;
    else
        P_need = 0;
        %Note = "The system is operating within its limits" ;

    end
end


if P_need ~= 0
    if EV_switch == 1
        P_ev = -P_pv*(2/3) ;
        P_bess = -P_pv*(1/3) ;
    else
        P_bess = -P_pv ;

    end
end




negative_out = {P_ev, P_bess, P_need} ;

end

%% Calculations for this hour. Used to predict next hour.
%Find net power
%Give the battery power 1 or 0 depending on state of charge
if SOC_bess_0 > SOC_min
    SOC_bess_t = 1;
else
    SOC_bess_t = 0;
end
if SOC_ev_0 > SOC_min
```

```
    SOC_ev_t = 1;
else
    SOC_ev_t = 0;
end


% Change the level of discharge from the EV based on SOC.
    % - No discharge if battery level is less than 30%
    % - Max power if above 40%
if SOC_ev_0 <= 30
    P_ev = 0;
else
    P_ev = EV_switch * SOC_ev_t * 9000 ; %Either 9kW or 0kW output

end

if SOC_bess_0 <= 30
    P_bess = 0;
else
    P_bess = SOC_bess_t * 9000 ; %Either 9kW or 0kW output
end

%If the max available power production can compensate for the demand, grid
%swich has been off prefious hour.
P_net_max = P_pv - P_vl - P_cl - P_losses + P_bess + P_ev ;
    %P_net_max = 0 --> Just enough power vs demand
    %P_net_max > 0 --> More than enough power prod, consider charging battery
    %P_net_mac < 0 --> Not enough power prod. Need grid connection.

%Start predicting the input values for t + delta_t
t= t + delta_t ;

%Set voltage level from bess and EV
V_battery = 240 ;

%Find current and then state of charge for t+dT in the two batteries

if P_net_max >= 0
```

```
    if EV_switch == 1
        I_ev_t = (P_ev)/(V_battery);
        I_bess_t = (P_bess)/(V_battery);
    else
        I_ev_t = 0 ;
        I_bess_t = (P_bess)/(V_battery) ;
    end
else
    if EV_switch == 1
        I_ev_t = -2/3*(P_pv/(V_battery)) ;
        I_bess_t = -1/3 *(P_pv/(V_battery)) ;
    else
        I_ev_t = 0 ;
        I_bess_t = -P_pv/(V_battery) ;
    end
end

SOC_ev = SOC_ev_0 - delta_t_hour*(I_ev_t/Batt_cap) ; % SOC +  h * A/(Ah)
SOC_bess = SOC_bess_0 - delta_t_hour*(I_bess_t/Batt_cap) ;

%Predicted solar irradiance and load

g = round(t*step)+1 ;   %Making the index match the time.
if g > 24*step
    g= round(t*step);
end

%test = Irr_c ;
Irr_1 = Irr_c(g) ;
P_vl  = Load_c(g)*1000 ;

%update P_pv
P_pv  =  (Irr_1/1000) *P_pv_max * Ns * Np ;



%Total load
P_load_1 = P_cl + P_losses + P_vl ;

```

```
%Predicion of EV switch status
if (t >= 0) && (t <8 )||(t >= 17) && (t < 19) ||(t >= 21) && (t <= 24)
    EV_switch = 1;
else
    EV_switch = 0 ;
end

%When solar pv and load prediction is done, we can find if batteries need
%discharge or charging next hour. But first we also need to see if both
%batteries can discharge fully.
    %- We check state of charge in t+dt to see the discharge rate of the
    %batteries
    % - We use the same calculation as previous timestep but with the
    % updated SOC:

% Change the level of discharge from the EV based on SOC.
    % - No discharge if battery level is less than 30%
    % - 50 % of max output if battery level is between 30 and 40%
    % - Max power if above 40%
if SOC_ev <= 30
    P_ev = 0;
else
    P_ev = EV_switch * SOC_ev_t * 9000 ; %Either 9kW or 0kW output
end

if SOC_bess <= 30
    P_bess = 0;
else
    P_bess = SOC_bess_t * 9000 ; %Either 9kW or 0kW output
end

P_PvMinusLoad = P_pv - P_load_1 ;
P_net_prev = P_pv -P_load_1 + P_ev + P_bess ;

%trigger functions to see if batteries are needed to supply the load
if P_PvMinusLoad >= 0 || P_net_prev >= 0
    Out = positive() ;
else
```

```
    Out = negative() ;
end


P_ev = Out{1} ;
P_bess = Out{2} ;


I_ev = P_ev/V_battery ;

%Calculating the net power of the system if the batteries were needed to
%make sure the power output is enough based on their
P_net = P_pv + P_bess + P_ev - P_load_1 ;

%Deciding what to do with the grid switch based on p_net

if P_net >= -1
    Grid_switch = 0;
    P_vsc = P_load_1 ; % /S_VSC_max
else
    Grid_switch = 1; %Connected to the grid.
    P_vsc = 0 ;
end

P_vsc = double(P_vsc);

t = t/24 ;
SOC_ev = SOC_ev/100 ;
SOC_bess = SOC_bess/100 ;

%Back to PU
I_ev = (I_ev + 75)/150
P_vsc = P_vsc/20000
Output = [Grid_switch, I_ev, P_vsc]  ;

end
```

Bibliography

[1] BP. ‘Statistical Review of World Energy, 2021’. In: (2021).

[2] IEA. *Share of low-carbon sources and coal in world electricity generation, 1971-2021,* URL: `https://www.iea.org/data-and-statistics/charts/share-of-low-carbon-sources-and-coal-in-world-electricity-generation-1971-2021` (visited on 20th Nov. 2021).

[3] Sol Up. *Microgrids - The next evolution of the grid.* URL: `https://solup.com/microgrids-next-evolution-grid/` (visited on 5th May 2022).

[4] Johan Driesen and Farid Katiraei. ‘Design for Distributed Energy Resources’. In: (2008).

[5] Adam Hirsch, Yael Parag and Josep Guerrero. ‘Microgrids: A review of technologies, key drivers, and outstanding issues’. In: *Renewable and Sustainable Energy Reviews* 90 (2018).

[6] Martin Wirak Onsrud. *AI-based Control Development for Microgrid.* Project report in TET4510. Department of Electric Power Engineering Technology, NTNU – Norwegian University of Science and Technology, Dec. 2021.

[7] Michael Haenlin and Andreas Kaplan. ‘A brief history of Artificial Intelligence’. In: (2019).

[8] Arden Dertat. *Applied Deep Learning - Part 1: Artificial Neural Networks.* URL: `https://towardsdatascience.com/applied-deep-learning-part-1-artificial-neural-networks-d7834f67a4f6` (visited on 10th May 2022).

[9] Bimal K. Bose. ‘Artificial Intelligence Techniques in Smart Grid and Renewable Energy Systems — Some Example Applications’. In: (2017).

[10] Katrine Hansen Reisænen. ‘An Artificial Neural Network-Based Power Management in Hybrid Microgrid’. In: (2021).

[11] Weizhen Dong, Shuhui Li and Xingang Fu. ‘Artificial Neural Network Control of A Standalone DC Microgrid’. In: (2018).

[12] Maria B. et al. ‘An autonomous hybrid DC microgrid with ANN-fuzzy and adaptive terminal sliding mode multi-level control structure’. In: (2022).

[13] Divya R, Gopika N P and Manjula G Nair. ‘ANN Based Solar Power Forecasting in a Smart Microgrid System for Power Flow Management’. In: (2019).

[14] Shravan Kumar Akula and Hossein Salehfar. ‘Frequency Control in Microgrid Communities Using Neural Networks’. In: (2019).

[15] Ritu Raj Shrivastwa et al. ‘Understanding Microgrids and Their Future Trends’. In: (2019).

[16] Vijay Kumar Garg and Sudhir Sharma. ‘Overview on Microgrid System’. In: (2018).

[17] A. A. Eajal et al. ‘Power Flow Analysis of AC/DC Hybrid Microgrids’. In: (2016).

[18] Salvatore D’Arco and Jon Are Suul. ‘Equivalence of Virtual Synchronous Machines and Frequency-Droops for Converter-Based MicroGrids’. In: *IEEE Transactions on Smart Grid* 5.1 (2014), pp. 394–395. DOI: `10.1109/TSG.2013.2288000`.

[19] NREL. *Documenting a Decade of Cost Declines for PV Systems*. URL: `https://www.nrel.gov/news/program/2021/documenting-a-decade-of-cost-declines-for-pv-systems.html` (visited on 15th May 2022).

[20] Quing-Chang Zong and Tomas Hornik. *Control of Power Inverters in Renewable Energy and Smart Grid Integration*. Wiley, 2013.

[21] Subhash Chander, Pramod Agarwal and Indra Gupta. ‘Design, Modeling and Simulation of DC-DC Converter’. In: (2010).

[22] Shalil Shah. ‘Step-by-step Design of an LCL Filter for Three-phase Grid Interactive Converter’. In: (2015).

[23] Quing-Chang Zhong, Wen-Long Ming and Yu Zeng. ‘Self-Synchronized Universal Droop Controller’. In: (2016).

[24] Farzam Nejabatkhah and Yun Wei Li. ‘Overview of Power Management Strategies of Hybrid AC/DC Microgrid’. In: *IEEE Transactions on Power Electronics* 30 (2015).

[25] Amandeep Kaur, Jitender Kaushal and Prasenjit Basak. ‘A review on microgrid central controller’. In: (2016).

[26] Dong Dong et al. ‘Analysis of Phase-Locked Loop Low-Frequency Stability in Three-Phase Grid-Connected Power Converters Considering Impedance Interactions’. In: (2015).

[27] Mohammad Amin and Qing-Chang Zhong. ‘Resynchronization of Distributed Generation Based on the Universal Droop Controller for Seamless Transfer Between Operation Modes’. In: *IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS* 67 (2020).

[28] Jeremy Lin, Fernando Magnago and Juan Manuel Alemany. ‘Optimization Methods Appliedto Power Systems: CurrentPractices and Challenges’. In: (2018).

[29] Daniel Graupe. *Principles Of Artificial Neural Networks*. World Scientific, 2013.

[30] Ajith Abraham. ‘Artificial Neural Network’. In: *Handbook of Measuring System Design* (2005).

[31] Zhang Hong. ‘A Preliminary Study on Artificial Neural Network’. In: (2011).

[32] Michael A. Nielsen. ‘Neural Networks and Deep Learning’. In: (2015).

[33] S. Jeyanthi and M. Subadra. ‘Implementation of single neuron using various activation functions with FPGA’. In: (2014), pp. 1126–1131. DOI: 10.1109/ICACCCT.2014.7019273.

[34] D. Steinkraus, I. Buck and P.Y. Simard. ‘Using GPUs for machine learning algorithms’. In: (2005), 1115–1120 Vol. 2. DOI: 10.1109/ICDAR.2005.251.

[35] Bin Ding, Huimin Qian and Jun Zhou. ‘Activation functions and their characteristics in deep neural networks’. In: (2018), pp. 1836–1841.

[36] *Neural Networks*. URL: https://www.3blue1brown.com/lessons/neural-networks (visited on 14th Dec. 2021).

[37] *Gradient Descent*. URL: https://www.3blue1brown.com/lessons/gradient-descent (visited on 14th Dec. 2021).

[38] Pragati Baheti. *Train, Validation, and Test Sets: How to Split Your Machine Learning Data*. URL: https://www.v7labs.com/blog/train-validation-test-set (visited on 15th May 2022).

NTNU

Norwegian University of
Science and Technology