Intelligent energy management scheme for a hybrid microgrid using machine learning

Prepared by:
Julian Banks (BNKJUL001)

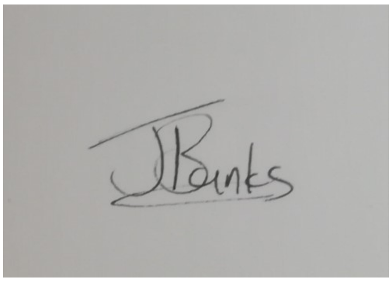
Prepared for:
Professor Sunetra Chowdhury
Department of Electrical Engineering
University of Cape Town

Submitted to the Department of Electrical Engineering at the University of Cape Town in partial fulfilment of the academic requirements for a Bachelor of Science degree in Mechatronics Engineering

September 26, 2023

Declaration

1. I know that plagiarism is wrong. Plagiarism is to use another's work and pretend that it is one's own.
2. I have used the IEEE convention for citation and referencing. Each contribution to, and quotation in, this report from the work(s) of other people has been attributed, and has been cited and referenced.
3. This report is my own work.
4. I have not allowed, and will not allow, anyone to copy my work with the intention of passing it off as their own work or part thereof.



September 26, 2023

Julian Banks

Date

Acknowledgements

No man is an island entire of itself; every man is a piece of the continent, a part of the main.

—John Donne

Abstract

- Open the **Project Report Template.tex** file and carefully follow the comments (starting with %).
- Process the file with **pdflatex**, using other processors may need you to change some features such as graphics types.
- Note the files included in the **Project Report Template.tex** (with the .tex extension excluded). You can open these files separately and modify their contents or create new ones.
- Contact the latex namual for more features in your document such as equations, subfigures, footnotes, subscripts & superscripts, special characters etc.
- I recommend using the **kile** latex IDE, as it is simple to use.

Contents

Abbreviations

ACF Auto Correlation Function

ANN Artificial Neural Network

AR Auto Regressive

ARIMA Auto Regressive Integrated Moving Average

AutoML Automated Machine Learning

CEEMDAN Complete Ensemble Mode Decomposition with Adaptive Noise

CNN Convolutional Neural Network

ESS energy storage system

GRU Gated Recurrent Unit

IMF Intrinsic Mode Functions

KPSS Kwiatkowski–Phillips–Schmidt–Shin

LSTM Long Short Term Memory

MAPE Mean Absolute Percentage Error

MDP Markov Decision Process

ML machine learning

MPPT maximum power point tracking

PACF Partial Auto Correlation Function

PER Prioritized Experience Replay

PID Proportional, Integral and Derivative

PV photovoltaic

RNN Recurrent Neural Network

SARIMA Seasonal Auto Regressive Integrated Moving Average

SARIMAX Seasonal Auto Regressive Integrated Moving Average with Exogenous Variables

UCT University of Cape Town

WNN Wavelet Neural Network

Chapter 1

Introduction

1.1 Background to the study

A very brief background to your area of research. Start off with a general introduction to the area and then narrow it down to your focus area. Used to set the scene .

1.2 Objectives of this study

1.2.1 Problems to be investigated

Description of the main questions to be investigated in this study.

1.2.2 Purpose of the study

Give the significance of investigating these problems. It must be obvious why you are doing this study and why it is relevant.

1.3 Scope and Limitations

Scope indicates to the reader what has and has not been included in the study. Limitations tell the reader what factors influenced the study such as sample size, time etc. It is not a section for excuses as to why your project may or may not have worked.

1.4 Plan of development

Here you tell the reader how your report has been organised and what is included in each chapter.

I recommend that you write this section last. You can then tailor it to your report.

Chapter 2

Literature Review

There has been extensive research into hybrid microgrids, their configuration, optimisation and interconnectivity with a utilities grid.

To remain within the scope of this report the literature review will explore three key topics related to hybrid microgrids. Namely, configurations of hybrid microgrids with both AC and DC subgrids. A look into contemporary energy management techniques for systems with less than 1MW generation capacity. Lastly, it will explore how machine learning has been applied to the energy management of hybrid microgrids.

~~2.0.1~~ Hybrid microgrids

Microgrids have gained traction as new forms of renewable energy generation have been developed and political pressure to decarbonise energy has increased. Microgrids are typically low-voltage networks employing distributed generation to meet local demand. Typical forms of generation include solar photovoltaic (PV), wind and diesel. Microgrids are usually equipped to operate in islanded or grid-tied modes. If they are required to reliably provide power to a load during islanded modes it is also necessary to include energy storage systems in the design.

Micrograds often use diesel generators to ensure energy security. This is due to the intermittent nature of popular renewables such as solar and wind. Additionally, it is typically costly to install an energy management system that is large enough to ensure energy security in the event of the microgrid operating in islanded mode. Diesel is a potent fossil fuel and an unattractive option in an energy system for the future. Alternatives such as bioenergy that implement a waste-to-energy cycle using biomass or bio-wast have been proposed. These have been excluded from this study as the composition of the hybrid microgrid is not the main focus, however, it would be interesting to explore their integration into microgrids in the future. [1]

Since microgrids function in a localised area they require integrated control systems to perform energy management, voltage regulation and frequency regulation. [2] When considering common microgrid control algorithms, the majority have been designed to operate AC microgrids. This is largely due to all major transmission lines being AC. The rise in DC load from consumer electronics and the growth of electric vehicles coupled with the DC nature of renewables such as PV creates a good argument for the installation of DC subgrids in a hybrid microgrid. Other advantages of DC subgrids include being free from frequency, power factor and phase sequence issues.[3]

A hybrid microgrid composed of both AC and DC subgrids will allow for loads to be met with fewer conversion stages if both AC and DC generation is employed. This will reduce power conversion losses and lower the number ac-dc and dc-ac converters needed. [2] Bidirectional converters can be used to control the flow of active power between the AC and DC subgrid. This allows one subgrid to become a load or a source while the other subgrid becomes the opposite. Meaning only the imbalance in load between the subgrids needs to be converted. However, the balancing of loads needs to be done in proportion to the generation capacity of each subgrid. This can be achieved by using a normalisation process that is explained in more detail in the energy management section of the literature review. [4]

Various configurations and topologies of hybrid grids have been explored in the literature. The two main types are coupled AC hybrid microgrids where the AC subgrid is attached to the utility grid via transformers. The second type is decoupled AC hybrid microgrids where the DC subgrid is connected to the grid and the AC subgrid connects to the DC subgrid. [5] The main advantage of a couple AC hybrid microgrid is that there are fewer power conversion stages as only a transformer is required to connect the subgrid to the utility grid. Additionally, the AC network is ~~fixed~~ to the utility grid when it is operating in grid-connected mode. This removes the need for vigorous frequency control etc. The main advantage of the decoupled AC configuration comes from the inherent control provided by the power converters between the utility grid and the microgrid. They allow for accurate tracking of power flow and makes fault detection easier. [5]

Coupled ac configurations can be further broken down into partially isolated and completely isolated systems.

Partially isolated systems consist of a low-voltage AC subgrid connected to the utility grid via a transformer and a low-voltage or medium-voltage DC subgrid connected to the utility grid via a power converter. This gives the advantage of requiring a power converter with a lower rating since it does not need to handle the power flow of the AC grid. However, it means that there is no galvanic isolation of the DC subgrid. Figure 2.1 outlines the configuration.

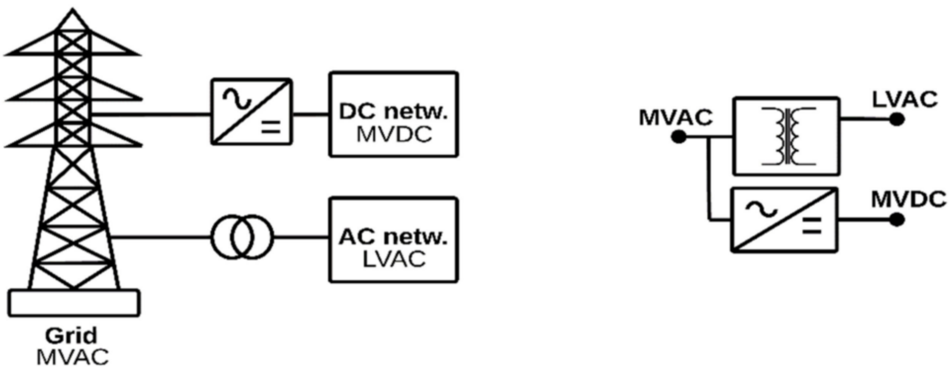


Figure 2.1: Coupled AC microgrid in partially isolated configuration. (source: [6])

Completely isolated systems consist of an AC subgrid that connects to the utility grid via a transformer and a DC subgrid that connects to the AC subgrid via a power converter. This means that the entire microgrid experiences galvanic isolation. This is useful for connections that have a high risk of experiencing faults. However, the transformer will require a higher power rating as it needs to be able to handle the power flow from both the AC and the DC subgrid. Figure 2.2 outlines ~~the~~ configuration.

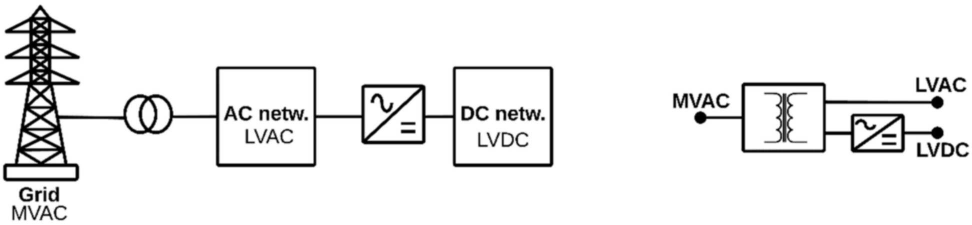


Figure 2.2: Coupled AC microgrid in completely isolated configuration. (source: [6])

Decoupled hybrid microgrids can be broken up into three main groups. Two-stage completely isolated, Two-stage partially isolated and Three-stage partially isolated. This review will not look into the Three-stage partially isolated system as it is typically used on larger scale solar or wind farms.

The two-stage completely isolated configuration can be seen in Figure 2.3. This configuration benefits from complete galvanic isolation due to the transformer placed between the utility grid and the AC/DC converter. This makes it more suitable to scenarios expecting a higher level of grid faults.

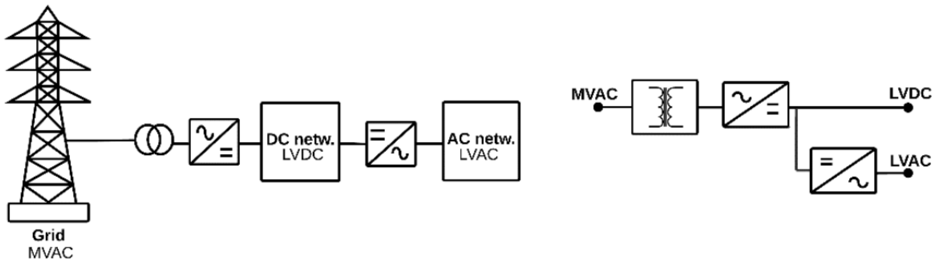


Figure 2.3: Decoupled AC microgrid in completely isolated configuration. (source: [6])

The two-stage partially isolated configuration can be seen in figure 2.4. This major difference in this configuration is that the transformer is moved and placed inbetween the DC subgrid and the AC subgrid. The advantage of this is that the transformer's power rating can be lower than in the completely isolated decoupled scenario.

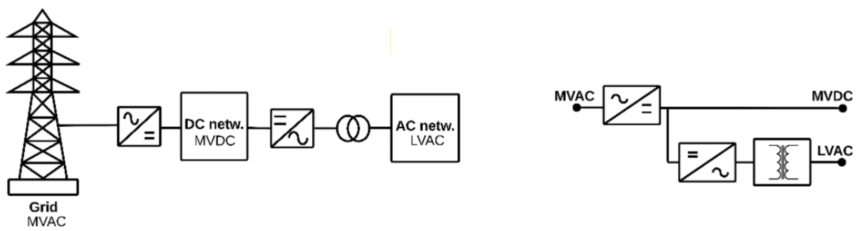


Figure 2.4: Decoupled AC microgrid in a partially isolated two-stage configuration. (source: [6])

What none of these configurations have explicitly included is an energy storage system (ESS). Due to the intermittent nature of renewables and the likelihood of load shedding, it will be essential to integrate an ESS into the system. An ESS will be responsible for storing energy during periods of excess generation or in off-peak grid times. It can then supplement the supply at times when the load is higher than the generation. This typically requires a bi-directional DC-Dc converter to help with stabilising the voltage in the DC subgrid.[7]

MAYBE ADD SOME STUFF ABOUT DIFFERENT ESS's OVER HERE! Batteries Hydrogen storage for fuel cells!

It is worth noting that the composition and design of microgrids are based on too many factors for a universal approach to be viable. Each design needs to be tuned for its environment and ~~parameters.~~ In [8], this is handled by employing a ~~2 layered~~ Multi-criteria decision-making model. ~~This~~ first calculates the weights of each criterion and then uses ranking methods to determine the optimum compromise. HOMER software was employed to test simulations of various configurations.

Components of Hybrid microgrids? Solar PV,

Wind,

Fuel Cells,

Diesel,

((((Make a final conclusion regarding the type of Hybrid topology that I am going to use based on the literature as well as a conclusion on the type of generation it is going to be made up of.)))

~~2.0.2~~ Energy management techniques

A crucial element of any microgrid is its control and energy management systems. The power flow between the distributed generation and the loads, the flow of power between the AC and the DC subgrids, the management of energy storage systems and the switching between islanded and grid-tied operations are just some of the necessary control systems. Additionally, the implemented control system needs to be flexible and able to handle any number of operating points due to the unreliable nature of renewables. [9]

In [7] the control system is split into three parts, the energy management system, the supervisory control and the local control. The energy management system schedules the power generation according to a day ahead schedule. This comes from a prediction of weather patterns and load profiles. The supervisory control manages the discrepancy between the scheduled power and the actual power used and generated. The local control system manages the voltage, current, and frequency of the microgrid. The results showed that the combined control system was effective at reducing costs and increasing the reliability of the hybrid microgrid. The proposed system in [7] includes solar PV and hydrogen fuel cells, the local control of each of these systems is performed by a Proportional, Integral and Derivative (PID) controller. The fuel cell setup uses a reference voltage and increases the generation whenever the voltage at the terminals falls below the terminals and vice versa. The solar PV system uses maximum power point tracking (MPPT) to optimise generation from the solar PV panels and the output of the MPPT is used as the input to the PID loop.

Matching controllable loads to peaks in power generation was explored in [10]. This was done by using predictions of the upcoming wind patterns and moving a controllable load to be attached during periods of peak production. An example of this could be scheduling the ESS to be charged at a point of peak power production based on a weather prediction. The study found that the method showed an increase in the state of charge of the energy storage system as well as a decrease in the usage of the supplementary diesel generator. However, this method relies on accurate forecasting of weather.

Droop control is a well-established form of control in AC microgrids. In [4] its use was extended to control a hybrid microgrid with both AC and DC subgrids. Droop control is achieved by determining the droop co-efficient for each energy source. In an AC microgrid, each source has two droop co-efficient, one for voltage and one for frequency. These coefficients are given by the $f_x^* = f_x' + m_x P_{a,x}$ and $V_{a,x}^* = V_{a,x}' + n_x Q_{a,x}$. ~~Where~~ the x represents the source in the AC subgrid. To then share the load in proportion to the rated capacity of the source the droop coefficients are used together with the ~~sources~~ rating, given as $S_{a,x}$. These values are used to satisfy the equations $m_1 S_{a,1} = m_2 S_{a,2} = ... = m_x S_{a,x}$ and $n_1 S_{a,1} = n_2 S_{a,2} = ... = n_x S_{a,x}$ balancing the load proportionally between the sources. This was done for all of the sources in the AC subgrid. For the sources in the DC subgrid, the process was slightly different as there is no frequency component. Instead the equation $V_{d,y}^* = V_{d,y}' + v_y P_{d,y}$ was used to calculate the droop coefficient, where $V_{d,y}'$ is the maximum output of a source under no load. And v_y is the droop coefficient. The proportional load sharing is then managed by the equation $v_1 S_{d,1} = v_2 S_{d,2} = ... = v_x S_{d,x}$

In [11], the energy management system is implemented logically by using a set of conditions. The conditions include production from the microgrid, the price of grid electricity, the load and the state of charge of the energy storage system. It is assumed that all of the loads and generation values are known at all times. This is a high-level form of control where the main priority is to keep the flow of power balanced between the loads and the sources. The rule-based control scheme can be visualised using a flow chart. The scheme was successful in balancing the loads and improving the performance of the system theoretically. However, it does not address how the actual control will be implemented, the high level of information that all of the conditions require implies that the system will require many sensors and a centralised control unit. These are factors that greatly affect the cost and reliability of the system.

This issue of reducing sensors is tackled in [12]. The approach focuses on the voltage of the DC-link. Control is performed by measuring the voltage of the DC link and using PI controllers to keep the voltage at a constant reference voltage. This is done by injecting power or drawing more power (selling it to the grid or storing it). A rule-based control scheme is still implemented at this stage. However, now the only inputs that are required for the controller are current through the DC-link (derived from the voltage), a boolean for whether the microgrid is islanded or not, a boolean for whether it is a peak pricing time or not and the state of charge of the batteries. ~~The paper~~ also includes supercapacitors to help smooth the transients and reduce the stress on the batteries. An important consideration that [12] brings up is the operation of the hybrid in off-MPPT mode. Typically MPPT is used to draw the maximum power from a PV system. However, if generation exceeds the load, the energy storage system is fully charged and the microgrid is in islanded mode then there is nowhere for the excess power to go and the PV system should not track the maximum power point.

Similar to a rule-based scheme, [13] uses states to implement supervisory control. In this case, every combination of loads and generation is assigned to a state. The controller then decides what state to put the system into. The states have entry conditions that are based on the available power of the sources and the required powers of the loads. The states have been chosen to maximise the renewable penetration and a diesel generator is only used to prevent load shedding if the wind and PV generation cannot meet the demand and the batteries are fully discharged. The supervisory control is shown to operate as expected and successfully change states to match loads. However, the performance is not compared to a benchmark so it is difficult to judge how effective the energy management system truly was. An important practical note the [13] brings up is the need to ensure impedance and voltage matching of converters. For example, if the DC subgrid is connected to the PV panels via a DC/DC converter and to the AC subgrid via an AC/DC converter then any mismatch between these converters will negatively affect power quality.

The energy management system that [14] uses is for a simple microgrid. The researchers approach the problem from an angle of optimisation and use integer linear programming to determine the optimal method of control. The microgrid in question contains a PV system, a wind turbine and an ESS of lead-acid batteries. The energy management system uses weather and load profile predictions along with grid pricing of electricity to schedule the state of charge of the batteries and to minimise the objective function. The objective function is designed to minimise operating costs. For cases when the load was low enough for the microgrid to never need to buy from the grid, it was found that using MPPT controllers was slightly more effective than the optimised energy management system. However, in cases when the microgrid was required to buy electricity from the grid the optimised energy management system outperformed the MPPT controllers as it was able to schedule the purchase of electricity during off-peak periods at a lower price.

((((Make a final conclusion, probably going to recommend that this paper focuses on a high level supervisory energy management system as the low level control of the sources is more hardware focused, and the high level supervisory control of prediction and scheduling will lend itself better to Machine Learning))))

~~2.0.3~~ Application of machine learning to energy management.

Machine learning as a tool for prediction.

For renewables that are heavily dependent on the weather, such as wind or PV, an accurate forecast of generation capacity is vital. Using machine learning techniques to make these predictions has been a growing field of study.

In [15], researchers use a Convolutional Neural Network (CNN) combined with a Gated Recurrent Unit (GRU). Weather data in the form of temperature, humidity, air pressure, and air density were collected at the site of renewable energy sources. The collected data was used by the CNN and GRU algorithms to predict weather patterns. After the method was refined and an attention mechanism was added to the network the Mean Absolute Percentage Error (MAPE) was decreased to 17.99% for a day ahead prediction. In [16] a similar combination of CNN and GRU was used to forecast price, wind speed and solar irradiation. However, much more accurate results were reported with MAPE values as low as 3.6% for wind speed and 5.47% for solar irradiance.

A feature that machine learning has shown promise in is the prediction of load profiles. If it is possible to accurately perform next-day load prediction it will be possible to implement an informed energy management strategy. In [17], this idea is explored and showed promising results. Various combinations of Artificial Neural Network (ANN), Wavelet Neural Network (WNN) and Kalman Filters are used to test the accuracy of load forecasting using neural networks. It was found that using a WNN and then feeding the results into an ANN in a hybrid neural network produced the best results. Additionally using statistical techniques such as clustering was found to improve the results. Given the necessity for an intelligent grid to be able to accurately schedule production, accurate predictions of loads will be incredibly valuable. The work done in [15] also used a combination of algorithms, a Complete Ensemble Mode Decomposition with Adaptive Noise (CEEMDAN) and a GRU ~~was~~ trained to predict the day ahead electrical load. An interesting property of the CEEMDAN network is that it can break the data into Intrinsic Mode Functions (IMF)s. This allows the model to identify the slower-moving and dominant IMFs. These are easier to predict and help to reduce the prediction error. The MAPE error was lowered to only 2.88% for a day-ahead prediction of the electric load.

An important argument is raised by [18], supervised learning can provide a high level of accuracy and can be tempting when research is carried out with a defined model and set of data. However, this will create models that are case-specific and not easily adaptable to the real world, unless of course a huge amount of data can be accessed. A preferable method of training could instead include reinforcement learning. In [18] the energy management system is split into two main tasks, prediction of generation and load and generation scheduling. The day ahead prediction of the energy generation and load is performed by Automated Machine Learning (AutoML) combined with Prioritized Experience Replay (PER). The results ~~boast~~ a high level of accuracy with a MAPE of 8% for the PV prediction, 7% for the wind prediction and 1.8% for the load prediction. ~~The paper~~ also goes on to use linear programming to find the optimal scheduling scheme for the system once the predictions have been made.

((WRITE ABOUT [19] Recurrent networks))

Machine learning as a tool for scheduling

In [20] the energy management problem is simplified to finding the best combination of energy sources that should meet the current load. The scheduling predictions were simplified further to be based on ~~4~~ inputs, namely, the hourly load demand, the temperature and two binary values for the availability of sun and wind. This accounts for varying loads but does not account for varying levels of wind or solar irradance as both factors have been simplified to a boolean. The output of the neural network was divided into classes. Where each class represented a different configuration of active energy sources. Four neural net algorithms were tested. All of the algorithms were found to perform well however the best one in terms of accuracy and performance was the Decision Tree algorithm. The Decision Tree algorithm is used to solve classification and regression problems. A major advantage of the algorithm is that it does not require scale normalisation and that it can operate with a range of attributes.

The scheduling challenge is optimised in [21] by transforming the problem into a Markov Decision Process (MDP). A Q-learning algorithm is then chosen due to its suitability for handling MDP problems to tackle the issue of energy storage management. ~~The paper~~ found varying results with the six variations of the algorithm that it tested. Unfortunately, none of them could match the theoretical limit that was calculated by the mixed integer linear programming method. However, they were able to converge on their result more quickly and were thus computationally cheaper.

((I would stil like to read about more examples of using Machine Learning to perform scheduling as it may well be more flexible that the strict Liner porgramming optimiser.)))

((I am going to learn more about using machine learning for data analysis and prediction and then come back to make final recommendations about which techniques to use for prediction of load and generation.))

Chapter 3

Methodology

3.1 Machine Learning Theory Development & Software Selction

This section aims to outline and develop the machine learning aproach taken to the challenge at hand. There are three tasks that will require a machine learning aproach, each one will need to utilise an algorithm that suits its challenges.

The first task is the day ahead prediction of load. The aim of this task is to predict the hourly load profile of the next day. This type of challenge typically uses regression learners. An interesting feature of load profiles is that they typically contain many harmonic components due to the daily, weekly and seasonal trends in energy usage.

The next task is the day-ahead prediction of electrical generation. This is done by predicting weather conditions and calculating the expected generation from the distributed generation of the microgrid. For example, predicting the wind speed and direction will allow for the generation from a wind turbine to be calculated. Predicting the solar irradiance will allow for the generation capacity of the PV array to be calculated. This type of prediction is also typically performed by regression learners. An alternative approach would be to use current weather patterns as an input and using a model to predict the future generation~~, this would essentially bake the weather to generation calculation into the model~~.

The results of the first two tasks will be used to perform energy management and scheduling of the hybrid microgrid. The scheduling will be designed to minimize OPERATING COSTS, DOWN TIME, GRID USAGE, ADAPTABILITY? This will be done by designing states for the microgrid. A classification algorithm will be used to predict what the optimal configuration of the hybrid microgrid is for the given predictions.

There are many options for software that can train and validate machine learning (ML) algorithms. Due to its widespread use in the engineering field, this report will be making use of MATLAB by MathWorks. This section will also explore the possibility of using open-source software such as TensorFlow & PyTorch in conjunction with MATLAB.

3.1.1 Analysis of Potential Machine Learning Techniques

Machine learning can be broadly ~~broken~~ into two categories, supervised and unsupervised learning methods. The key difference between these two categories is that supervised learning makes use of labelled target data while unsupervised learning does not.

Unsupervised learning methods

Unsupervised learning seeks to find patterns, structures or relationships in a set of data that does not have labelled data. Typically unsupervised learning is used for tasks such as data clustering, dimensionality reduction and anomaly detection.

Data clustering is used to collect similar groups of data into a cluster. This can then be used to understand how the data is distributed and can help to show patterns in data with no pre-existing labels. Importantly the analysis of the clusters will typically not be the job of the same algorithm performing the clustering. The algorithm performing the clustering has two main aims. To create clusters with a very high intra-cluster similarity, meaning that all of the data within a cluster is similar. Additionally, the algorithm will aim to make the inter-cluster similarity as low as possible, meaning that each cluster contains data that is different to the other clusters. Due to its popularity and the growing number of algorithms available for data clustering, it has become a staple of unsupervised learning and has many real-life use cases. These include biological sequence analysis, social network or search result grouping for recommendations and network traffic analysis.

Dimensionality Reduction is used to reduce the number of features or dimensions of a dataset without losing essential information. The main advantage of this is that it can reduce the size of massive datasets, aid in data compression and decrease computation times. Additionally reducing the dimensionality of data means the model could be less complex and easier to interpret, however interpreting transformed features could also be challenging leading the independent variables to become harder to comprehend. While model accuracy can be improved due to the removal of misleading data it is also possible that due to losing some information there will be an impact on how well some algorithms perform. MATLAB includes methods to perform dimensionality reduction such as Sequential Feature Selection and Principal Component Analysis.

Anomaly detection is used to detect data that vary significantly from the norm. Outliers can be indicative of errors in data collection, data preprocessing, noise or intentional interference(fraud, attacks). Outliers are a threat to an algorithm's potential robustness and security as they will affect the performance of the model. Anomaly detection could use labelled data and be performed in a supervised method. However, in the real world, there are very few recorded anomaly data points compared to normal data points and therefore with very few labelled anomaly samples it is almost always carried out as an unsupervised task.

Supervised learning methods

Supervised methods used past data that has been labelled to train models. The primary objective of most supervised learning methods is to learn a mapping from a set of input features to a target response variable. The model aims to learn relationships that can be applied to unseen data to make predictions. These predictions can be in the form of classification or regression.

Classification aims to take input data and determine what it belongs to from a pre-determined set of categories or classes. WRITE MORE HERE WHEN I USE THEM

Regression aims to predict a continuous numerical value. This is done by plotting a line of best fit or a curve to the data. Typically a trained regression algorithm is evaluated using the variance, bias and error. A high variance can lead to overfitting while a high bias can lead to underfitting. Regression can be linear (one input to one output), multiple (many inputs to one output), or multivariate (many outputs). Linear regression is limited to plotting linear equations and works best when predictors are independent and uncorrelated. For non-linear data, it is necessary to use polynomial regression. Polynomial regression is able to plot curves between data. In order to avoid overfitting the degree of the polynomial needs to be tuned. Regularization is also used to reduce the potential for overfitting by adding a penalty term to the cost function. The ridge and lasso regression techniques each make use of their own formulas to perform regularization as can be seen in Figure 3.1.

Ridge Regression

$$\min_{c_0, c_j} \left(\frac{1}{2n} \left(\sum_{i=1}^{n} \left(y_i - c_0 - \sum_{j=1}^{p} x_{ij} c_j \right)^2 + \lambda \underbrace{\boxed{\sum_{j=1}^{p} c_j^2}}_{\text{Penalty term}} \right) \right)$$

Lasso

$$\min_{c_0, c_j} \left(\frac{1}{2n} \sum_{i=1}^{n} \left(y_i - c_0 - \sum_{j=1}^{p} x_{ij} c_j \right)^2 + \lambda \underbrace{\boxed{\sum_{j=1}^{p} |c_j|}}_{\text{Penalty term}} \right)$$

Figure 3.1: Ridge vs. Lasso regularization techniques. (source: [22])

3.1.2 Auto-regressive Models

Auto Regressive (AR) models are widely used time series forecasting models that work by capturing and fitting different aspects of temporal data patterns. The key feature of a pure AR model is that the future values of a time series depend only on its past values(known as lags).

$$Y_t = \alpha + \beta_1 Y_{t-1} + \beta_2 Y_{t-2} + ... + \beta_p Y_{t-p} + e$$

Where Y_{t-1} is the first lag, β_1 is the coefficient of the first lag term, α is the model's intercept term and e is the error term. The index 'p' is what defines the order of an AR model. It determines how many lags of the function will be used as predictors (how many past values contribute to the future value).

AR models are a fundamental component of models such as Auto Regressive Integrated Moving Average (ARIMA),Seasonal Auto Regressive Integrated Moving Average (SARIMA) amd Seasonal Auto Regressive Integrated Moving Average with Exogenous Variables (SARIMAX).

ARIMA

ARIMA adds two important features to AR models. Firstly it uses differencing to make the time series stationary. Differencing entails subtracting the previous value from the current value and can be performed multiple times. This helps it to model real-world time series data. Secondly, it uses a moving average (MA) component to make the model dependent on its past forecasting errors. ARIMA models can be used to model any 'non-seasonal' time series that shows patterns and is not a random white noise function. Therefore an ARIMA model can be defined as a time series that was differenced at least once to make it stationary and can be described by a combination of the equation for AR and MA.

$$Y_t = \alpha + +\beta_1 Y_{t-1} + \beta_2 Y_{t-2} + ... + \beta_p Y_{t-p} + e_t + \phi_1 e_{t-1} + \phi_2 e_{t-2} + ... + \phi_q e_{t-q}$$

The first half of the equation is identical to the equation for AR models while the terms with $\phi_i e_{t-i}$ contribute the moving average. Each error term comes from the error of the respective lag term, ϕ is the coefficient for that error term and the index 'q' defines how many error terms are included in the equation.

As mentioned earlier, it may be necessary to difference a time series to make it stationary. It is optimal to perform the minimum differencing required to achieve a time series that is near-stationary. This means it stays close to a defined mean and the Auto Correlation Function (ACF) plot tends to zero within a reasonable frame.
If a high number of the lag terms ~~display~~ a positive autocorrelation then the series should be differenced further. Conversely, if the first lag experiences a negative autocorrelation then the series may be over-differenced.
Determining the number of 'p' or AR terms can be determined by examining the Partial Auto Correlation Function (PACF). The PACF helps to convey the correlation between a specific lag and the series, which helps to determine whether the lag is needed in the AR term or not. A good starting estimate is to set the order of the 'p' term to equal the number of lags that are above the significance limit in the PACF plot.

The moving average or the 'q' term can be determined by looking at the ACF. Similarly to the 'p' term, the 'q' term can initially selected by setting it equal to the number of lags that are above the significance limit of the ACF.

Once these initial terms have been chosen it will likely be necessary to tune them after defining and training the model. This is done by inspecting the coefficients and the significance of each term. The residuals of the model can also help to show a good fit, residuals with no patterns and a constant mean and variance are ideal.

In summary, an ARIMA model can be used to forecast values of a non-seasonal time series and is characterised by three terms.
The 'p' term is the order of the AR component.
The 'q' term is the order of the Moving Average (MA) component.
The 'd' term is the number of times the series needs to be differenced to make the series stationary.

SARIMA

A possible issue with using an ARIMA model is that it does account for seasonality. If a time series displays seasonal trends then the standard method of differencing will not make the series stationary. Seasonal differencing works by subtracting the value from the previous season instead of from the previous term. IMAGE TO SHOW THIS????? SARIMA models have an additional set of parameters that describe the seasonality. 'P' is the order of the Seasonal Auto Regression (SAR). 'D' is the order of the seasonal differencing. 'Q' is the order of the Seasonal Moving Average (SMA). Additionally, the frequency of the seasonal pattern needs to be defined.

SARIMAX

SARIMAX adds in an external predictor called an exogenous variable. This can help to improve the forecast accuracy by including external factors that will influence the time series. The only condition for the exogenous variable is that it must be known during the forecast period.

3.1.3 Neural Nets

3.1.4 Long Short Term Memory Networks

Long Short Term Memory (LSTM) networks are a type of Recurrent Neural Network (RNN) architecture that is particularly well suited for sequential data and time series analysis. The special feature of an LSTM is that they are able to capture long-range dependencies. This is done by using a memory block that is in charge of remembering the neural nets' temporal state. A network of gates is used to control the flow of information through the memory block, namely input gates, output gates and forget gates. Input gates determine whether new information enters the cell. While f[23]

3.1.5 Feature Scaling

Typically raw data has many different variables with different scales and units of measurement. This can lead to a model showing bias towards features with more extreme distribution values. While some machine learning algorithms are immune to these issues, they heavily affect the performance of others.

Feature scaling helps to solve this issue by transforming all of the data to a specific scale, distribution or range. Some of the most common methods include re-scaling, standardisation and normalisation.

Rescaling

Rescaling a vector is analogous to changing the units of measurement for the data. This is done by first adding or subtracting a constant and then multiplying or dividing by a constant. An example would be data entries that hold units of energy that could be rescaled to be in the kilo-units (kW) from units (W).

Standardisation

The core aim of standardisation is to scale a set of data so that it fits a standard normal distribution. Meaning that after standardisation the data should have a mean of 0 and a standard deviation of 1. This process will change the values of the data set but keep the shape of the distribution the same. The formula is simple to apply, $\hat{X} = \frac{X-\mu}{\sigma}$. However, it is important to note that the mean and standard deviation should be calculated on only the training data with the test data excluded. Once they have been calculated on the training data they can be applied to the test data set.

For example:

Training Data: [1, 10, 3, 7, 5]
Test Data: [3, 6]
$\mu = \frac{1+10+3+7+5}{5} = 5.2$
$\sigma = \sqrt{\frac{(1-5.2)^2+(10-5.2)^2+(3-5.2)^2+(7-5.2)^2+(5-5.2)^2}{5}} = 3.124$

These values are then applied to both the training and test data according to the formula. $\hat{X} = \frac{X-5.2}{3.124}$

Normalisation

The main aim of normalisation is to fit data into a specific range, typically between 0 and 1. In a similar manner to standardisation, the process does not alter the shape of the distribution, in order to preserve information. The formula for normalisation is $\hat{X} = \frac{X-X_{min}}{X_{max}-X_{min}}$. In a similar manner to standardisation, the values for X_{min} and X_{max} are calculated according to the training data and then applied to both the training and testing data.

3.1.6 Machine Learning with Matlab

MATLAB offers an array of ML capabilities. Included in this are the Deep Network Designer, Classification Learner and the Experiment Manager apps. MATLAB has extensive documentation and a strong user base that will make the software easy to use and apply.

3.1.7 Co-execution with MATLAB, TensorFlow

Requires MATLAB and TensorFlow. This option is powerful and flexible as you are not required to convert between platforms, you can make use of the advantages each platform provides. This option requires Python and TensorFlow loaded and initialised into the MATLAB workspace. It is then possible to execute ML algorithms that have been trained in TensorFlow in the MATLAB environment, making use of both MATLAB features and Python libraries at the same time.

3.2 Load Data Collection and Preparation

Data was provided by University of Cape Town (UCT) for the Menzies Engineering building. This data came in the form of an hourly load profile. Table 3.1 shows the headings of the data that was received. Columns that held identical values for every row were removed from the table immediately as they held no information.

The data from the ~~Menzies~~ building was used to model the AC load for the hybrid microgrid. Unfortunately, the ~~Menzies~~ building does not have a DC load. In order to model the DC component of the hybrid microgrid a data set with an hourly load profile was found in Mendeley data under a Creative Commons license [24]. A middle school's load profile was chosen as it showed restively similar trends and distributions to the ~~Menzies~~ data. The data was shifted to start on the same day of the year. Since it came from the northern hemisphere the data was shifted so that the winter season lined up with the winter in the southern hemisphere.

The AC load data from Menzies and the DC load data from Mendeley Data were joined into one table.

Heading	Description
impwh	Imported True Energy, stored as a running total in kilo-Watt-hour (kWh)
diff_imp_kwh	The Imported True Power during the last hour.
ptot	System True Power, in kilo-Watts (kW)
stot	Apparent System Power, in kilo-Volt-Ampere (kVA)
qtot	System Reactive Power, in kilo-Volt-Ampere Reactive (kVAr)
pftot	System Power Factor
md	Maximum Demand Achieved
epoch	Time Stamp the Modem Measured the Data (UNIX TIMESTAMP)
timestamp	Time and Date When the Data Point Was Measured
time	Time in 24 hour format
date	Date in yyyy/mm/dd format
day	Day of the week, eg. Monday, Tuesday ect.
tou	Time of Use eg. Off-peak, Standard, Peak
tou_id	Time of Use ID, a numerical representation of tou
week_number	Week number in the year.
DC_load	The DC load profile was appended to the Menzies dataset

Table 3.1: Table headings for Menzies Data

3.2.1 Cleaning data

The data contained 6 rows with null entries. In order to preserve the time series characteristics as far as possible the values were not removed as this would create inconsistencies in the time steps. Instead, the previous value was used to fill the missing values.

While looking for outliers in the AC load it was clear to see that there were numerous instances of near-zero power being recorded as can be seen in figure 3.2. When the days that load shedding was implemented during 2022 were plotted over the Menzies electrical load plot, it was clear to see that these anomalies were due to load shedding. It was therefore necessary to replace all of these values since they would impede a model's ability to predict the load. The anomalies were cleaned by removing the

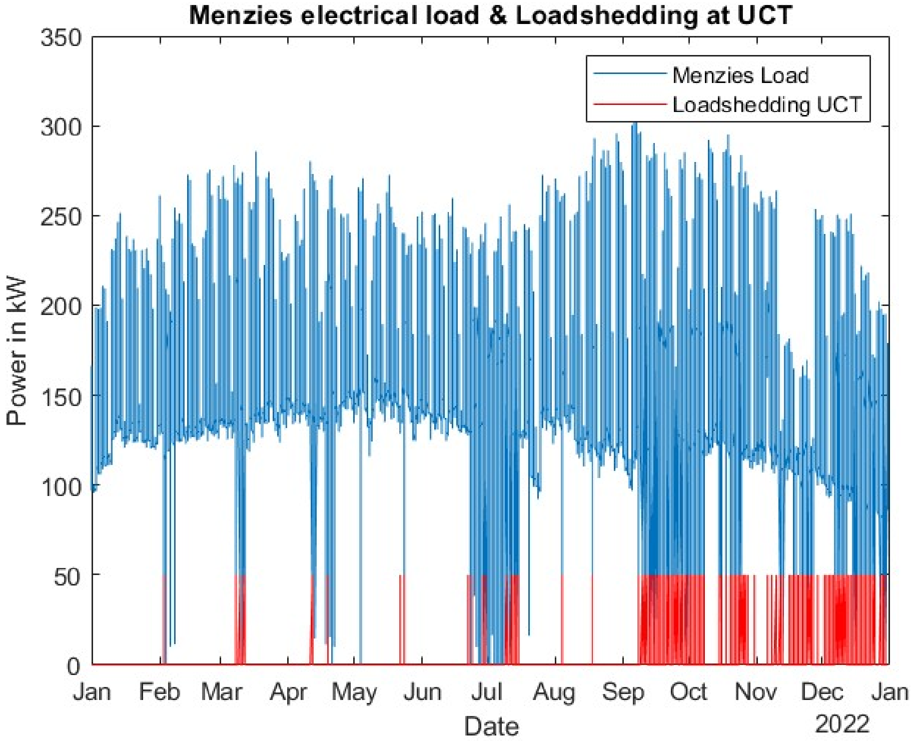


Figure 3.2: ~~Menzies~~ electrical load, overlayed with loadshedding at UCT.

lower 4.7 percentile of data and replacing the points using linear interpolation. The cleaning process is visualised in figure 3.3.

Much of the data is duplicated in columns such as timestamp, time and date. Rather than remove the duplicates upfront, it was decided to leave all of the columns in the cleaned database. This is due to different models requiring data to be prepared in different formats. Thus the cleaned database would be modified to suit the needs of each specific model.

3.2.2 Exploratory Data Analysis

Code for this section can be found here. (Very rough and still a working version)

In order to further understand the data it is necessary to perform an exploratory data analysis. The aim is to unearth valuable insights, patterns and trends that may have initially been hidden. The focus will be on the load profile and consumption patterns in an attempt to inform our approach when using the data for predictions.

When observing a histogram of the distribution for the AC load data is distinctly bi-modal as can be seen in figure 3.4. This was expected and can largely be attributed to consumption during the day versus consumption during night hours as can be seen in the sub-plots below.

However, what is interesting is that the AC load during the day also has two distinct peaks. This was further explored and it was determined that it arose from different usage patterns during working days and during weekend days. The breakdown goes further to show that weekends also have two distinct peaks, one for Saturday and another for Sunday as can be seen in figure 3.5.

The average hourly load profiles follow a very similar trend that is inline with what was expected. This

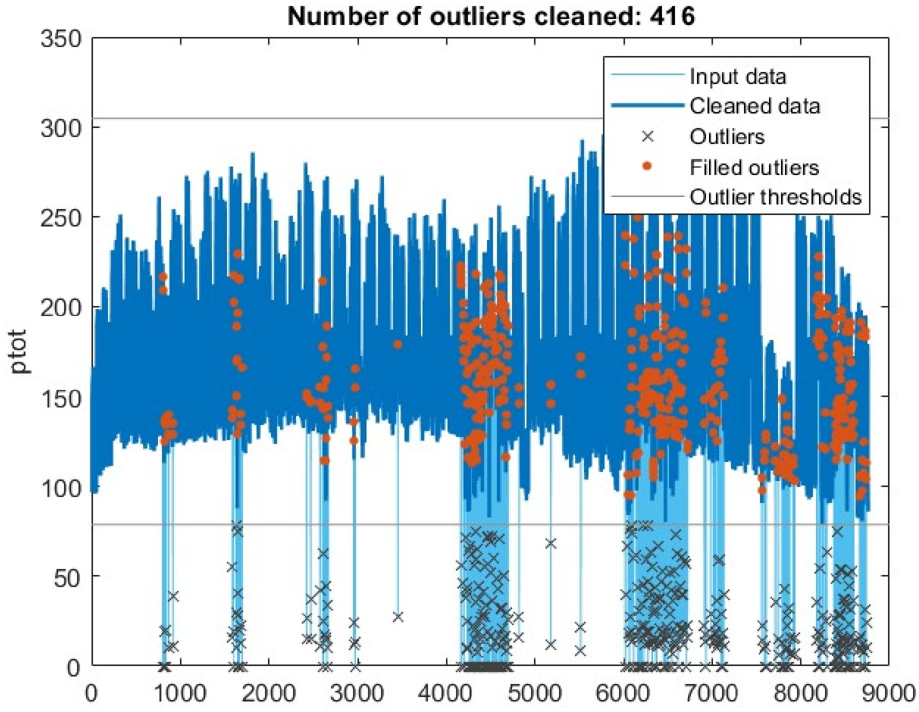


Figure 3.3: AC load and outlier replacement

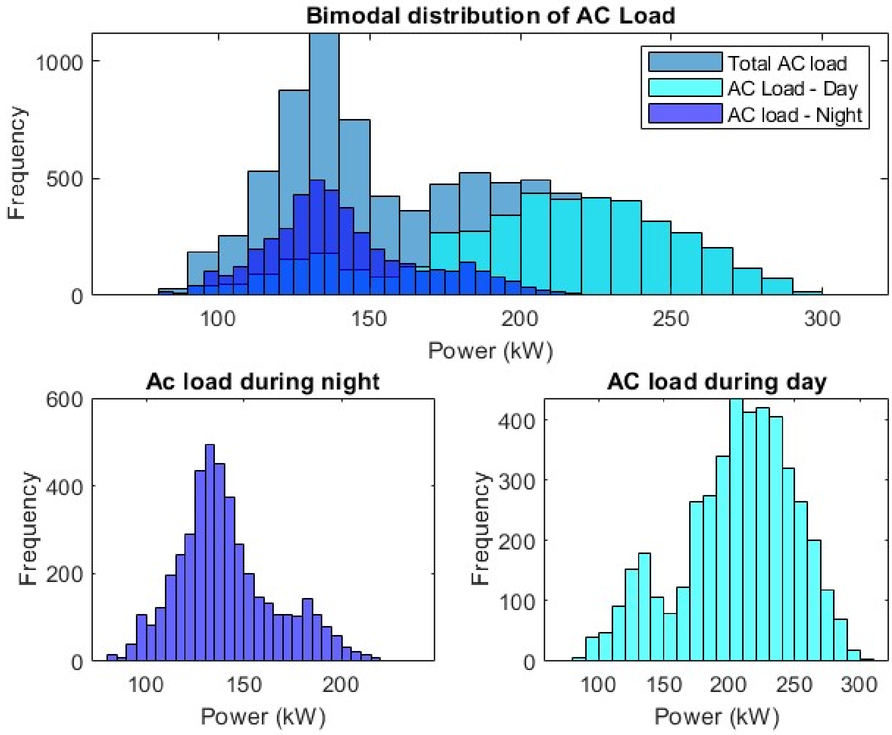


Figure 3.4: The bimodal distribution of the AC load.

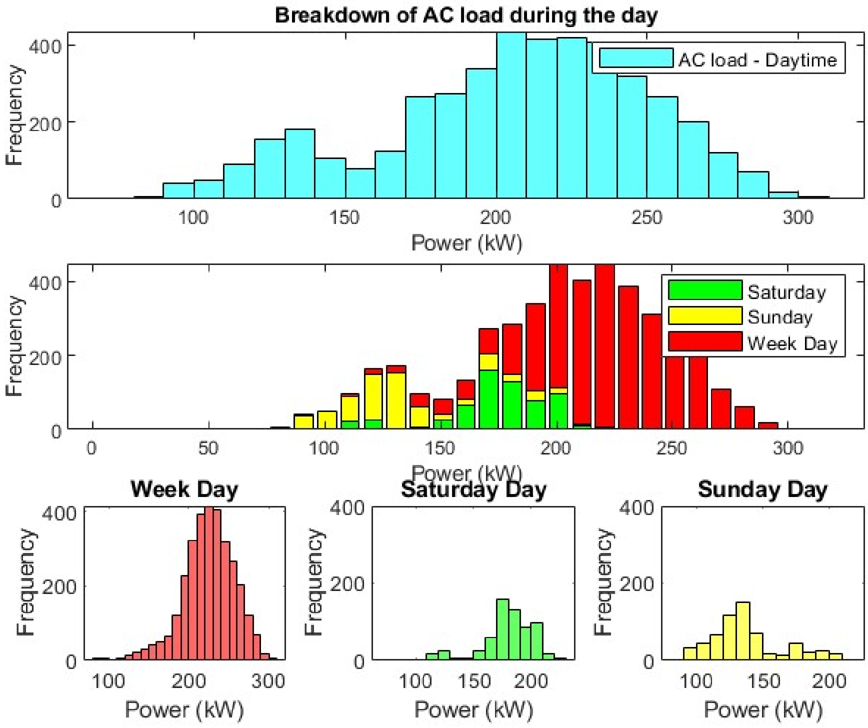


Figure 3.5: The bimodal distribution of the AC load.

can be seen in figure 3.6

The seasonality of the data can be seen through a boxplot of the average power used per month. There is a slight uptick in power usage during the winter months although it is not as prominent as was expected. The large number of outliers shown in the boxplots of the DC load was also unexpected. These outliers arise from the heavy skewness of the data with many more readings being taken outside of working hours and on weekends compared to at peak usage~~. Leading~~ to the median of the data being a comparatively low value.

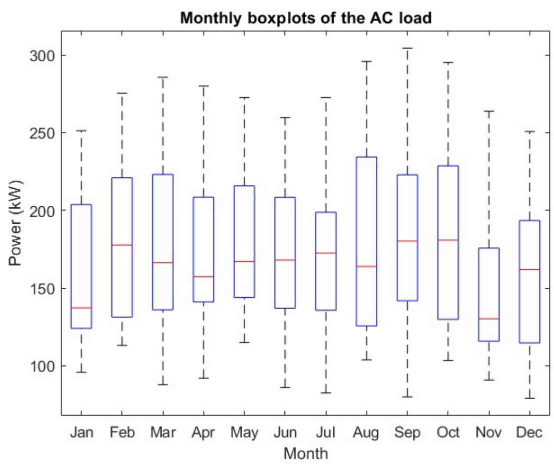


Figure 3.7: The average hourly AC & DC load profiles.

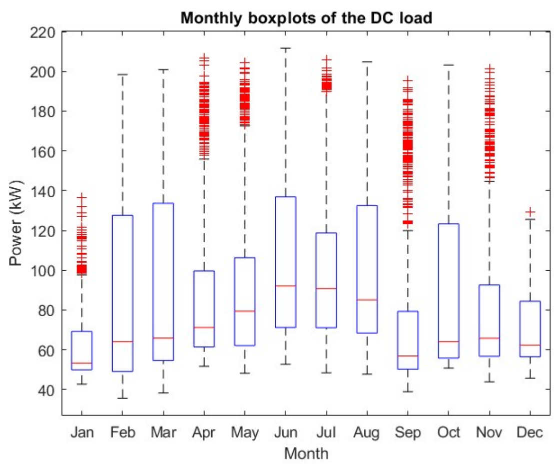


Figure 3.8: The average hourly AC & DC load profiles.

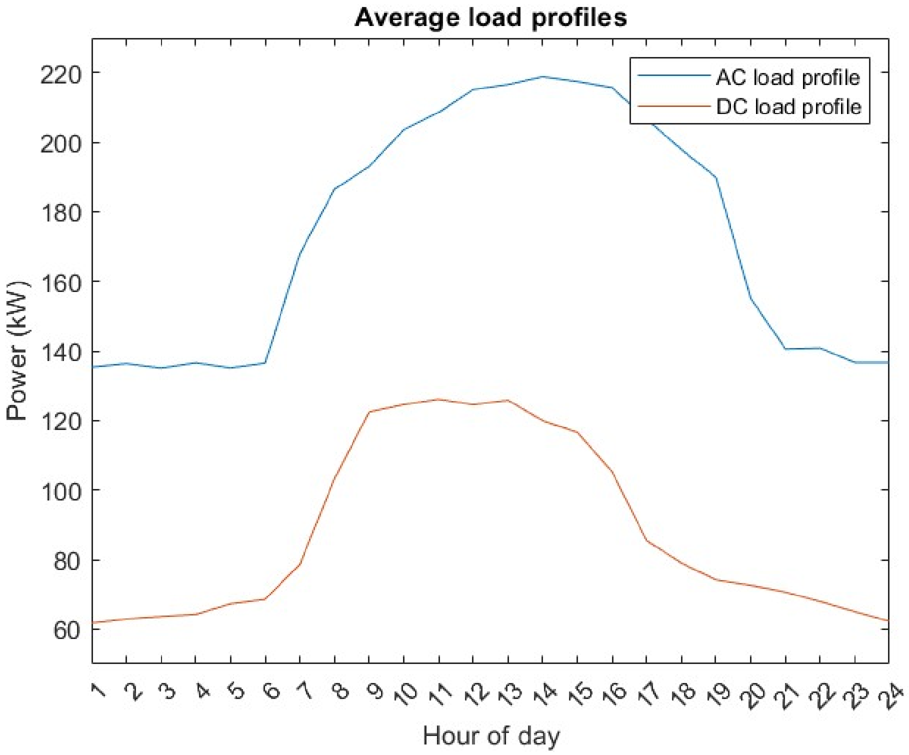


Figure 3.6: The average hourly AC & DC load profiles.

3.2.3 Feature Selection

After the data was examined it was decided that indicator variables would be added to the table and tested. These were in the form of boolean values. An indicator was added for whether a reading was taken during working hours, on a weekday or on a weekend.

Additionally, the time of use variable was ~~not~~ one indexed in an attempt to prevent biases being shown the the categories with numerically higher values.

3.3 Design of Hybrid Microgrid

The hybrid microgrid is going to use the Menzies Engineering building as a loose base model. This will be used to approximate available solar PV capacity and wind capacity. The current AC load will be used to model the AC subgrid and a hypothetical DC subgrid will be modeled to complete the hybrid setup. The microgrid will have the capability to act in islanded or grid-tied mode. This is essential to avoid grid failures such as load shedding. This means that the microgrid will have its own energy management system in order to provide energy security during load-shedding.

Homer Pro is a powerful software tool that is widely used for optimizing and designing microgrids and renewable energy projects. It assists with modelling, simulating and analysing energy systems with the aim of achieving optimal performance, cost-effectiveness and sustainability.

The user input required is data related to loads, energy resources, equipment specifications and economic parameters. HOMER Pro then runs an optimization algorithm to find the best configuration and sizing of the system.

3.3.1 Homer PRO Input

HOMER Pro can make use of real-world data to formulate a load profile for the system in question. The data provided for the Menzies building was used for the AC load input. Data for the DC load was taken from an open-source database [24].

The cleaned data from the previous section was used as the input.

Pricing

Homer PRO requires pricing input for each component of the grid in order to find the most economically optimal solution.

In order to find rough approximations of the component prices, purchase prices were found online and tabulated in Appendix A, table A.1. A summary of the prices and final inputs is shown below in table 3.2

Component	**Size**	**Price (R)**
Solar PV	1 kW	4 396.00
Wind Turbine	10kW	48 598.00
Diesel Generator	1kW	2 919.00
Battery Li-ion	100kWh	646 963.00

Table 3.2: Input prices to Homer PRO

Grid Reliability

The loadshedding schdule for UCT in 2022 was uploaded to Homer PRO. This was used to get an idea of how loadshedding and an unstable grid would affect the sizing of grid components.

The cost of unmet demand was set to R5/kWh. This conservative estimate was used to incentivise the optimiser to reduce unmet demand and find a system that was capable of meeting demand.

Economic Factors

Inflation was set to a conservative 6% which is the estimated headline inflation for 2023 taken from a statement made by the South African Reserve bank [1]. It is worth noting that inflation of electricity prices has been much higher, however for simplicity the headline inflation was used.

Sell in Tariff Structure

-Will there be a sell-in tariff? Is currently set to zero becuase I couldn't find any information on large power users selling more than they buy (which we would)

Current Tariff Structure

According to the City of Cape Town's Annual Tariff Policies (NEED TO CITE, Document is downloaded in 4022-Data) An electricity tariff is defined based on the category of service provided. The categorisation of a user is based on the type of service, level of service consumption, type of connection, time of use and any other important factors.

The type of service could include wheeling, renewable supply, residential or commercial.

The level of service consumption is based on the amount of electricity that the user consumes and is typically a tiered structure.

The type of connection refers to low voltage, medium voltage or high voltage connections. Further, it includes the capacity that has been authorised at that level. For example, a commercial building might have a low voltage connection that is authorised for 400kVA capacity.

The time of use is a factor that changes the tariff structure based on whether the consumption occurs at a peak, standard or off-peak time.

Other indicators which could affect the applied tariff structure include the type of meter installed, municipal property valuation or geographic location.

The Menzies building will fall into commercial/industrial power user since it is not a lifeline or residential user. (CITE: Tariff Policies - 3 Categories of Users. 3.3)

The max recorded power used was 300kW. Therefore the Menzies building qualifies as a Large Power User with LV tariffs. (CITE: Chapter 5.2 Commercial/industrial)

The tariff structure that outlines the time of use tariff prices for the Large Power User with a Low voltage connection can be found in the (Cite Electricity Consumptive Tariffs, in 4022-Data). It is copied into table 3.4
Peak time of use: Weekdays 07:00 to 10:00; 18:00 to 20:00
Standard: Weekdays 06:00 to 07:00; 10:00 to 18:00; 20:00 to 22:00
Off Peak: All other times

[1] https://www.resbank.co.za/en/home/publications/publication-detail-pages/statements/monetary-policy-statements/2023/Statement-of-the-Monetary-Policy-Committee-July-2023

Table 3.3: Large User Low Voltage Time of Use Tariff

Services	**Unit**	**Remarks**	**R** (reg)	**R** (unreg)	**R** (total)	**R** incl. VAT
High Demand Jun-Aug		Winter Months				
Peak	c/kWh		483.99	31.42	515.41	592.72
Standard	c/kWh		150.58	31.42	182.00	209.30
Off Peak	c/kWh		84.39	31.42	115.81	133.18
Low Demand Sept-May		Summer Months				
Peak	c/kWh		161.71	31.42	193.13	222.10
Standard	c/kWh		113.10	31.42	144.52	166.20
Off Peak	c/kWh		73.82	31.42	105.24	121.03
Service Charge	R/day		133.69	0.00	133.69	153.74
Demand Charge	R/kWh		219.93	0.00	219.93	252.92

Table 3.4: Time of use rates for the a large power user in Cape Town

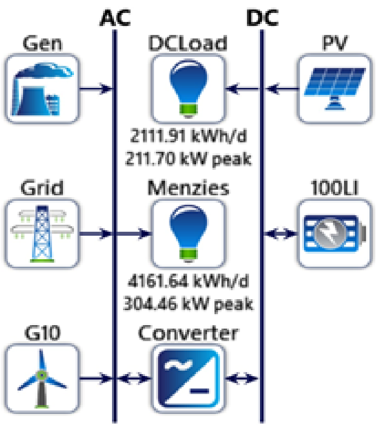


Figure 3.9: Input grid layout to Homer Pro.

Final grid input

The final input configuration to Homer PRO can be seen in figure 3.9. It includes an AC bus with the AC load, the grid, the wind turbines and a diesel generator. It also includes a DC grid with a the DC load, the PV panels, the Li-ion batteries and a 2 way converter.

3.3.2 Homer PRO output

THIS IS JUST TO UPDATE ON PROGRESS, NEED TO PICK ONE OF THE THREE OPTIONS.

1st OPTION: Limited capacity to max 1MW. Electrical fig: 3.10
500* 1kw units PV
50 * 10kw units Wind
6 * 100kWh Li-Ion Battery
Requires buying 716 012kWh from grid.

Excess and Unmet

Quantity	Value	Units
Excess Electricity	974,938	kWh/yr
Unmet Electric Load	16,794	kWh/yr
Capacity Shortage	31,794	kWh/yr

Production Summary

Component	Production (kWh/yr)	Percent
Generic flat plate PV	874,253	26.6
Generic 10 kW	1,693,591	51.6
Grid Purchases	716,062	21.8
Total	3,283,905	100

Consumption Summary

Component	Consumption (kWh/yr)	Percent
AC Primary Load	1,505,115	66.2
DC Primary Load	767,934	33.8
Deferrable Load	0	0
Total	2,273,049	100

Figure 3.10: Electrial Summary Option 1.

2nd option: Limited capacity to max 2MW. Electrical fig: 3.11
1000* 1kW units PV
100* 10kw units Wind
5* * 100kWh Li-Ion Battery
requires buying 513 495kWh from grid.

3rd option: NO limit on capacity let Homer Pro find the most optimal. Electrical fig: 3.12
1334 *1kW PV
223 * 10kw units Wind
4 * 100kWh Li-ion Battery
requires buying 379 914kWh from grid.

It is interesting to note is that none of the configurtions included a diesel generator as they all found it to be too expensive.

Electrical Summary

Excess and Unmet

Quantity	Value	Units
Excess Electricity	3,341,718	kWh/yr
Unmet Electric Load	13,314	kWh/yr
Capacity Shortage	22,782	kWh/yr

Production Summary

Component	Production (kWh/yr)	Percent
Generic flat plate PV	1,748,505	31.0
Generic 10 kW	3,387,181	60.0
Grid Purchases	513,495	9.09
Total	5,649,182	100

Consumption Summary

Component	Consumption (kWh/yr)	Percent
AC Primary Load	1,507,782	66.2
DC Primary Load	768,747	33.8
Deferrable Load	0	0
Total	2,276,528	100

Figure 3.11: Electrial Summary Option 2.

Electrical Summary

Excess and Unmet

Quantity	Value	Units
Excess Electricity	7,962,252	kWh/yr
Unmet Electric Load	11,875	kWh/yr
Capacity Shortage	19,279	kWh/yr

Production Summary

Component	Production (kWh/yr)	Percent
Generic flat plate PV	2,333,318	22.7
Generic 10 kW	7,553,414	73.6
Grid Purchases	379,914	3.70
Total	10,266,647	100

Consumption Summary

Component	Consumption (kWh/yr)	Percent
AC Primary Load	1,509,194	66.3
DC Primary Load	768,773	33.7
Deferrable Load	0	0
Total	2,277,967	100

Figure 3.12: Electrial Summary Option 3.

3.4 Machine Learning Process for Load and Generation Prediction

3.4.1 Fitting LSTM model

In this step, we adapt the hourly electrical load data for LSTM, which excels at handling multivariate time-series forecasting.

Code for this section can be found here. (Very rough and still a working version)

Data Preparation

The electrical load data that was prepared in the previous section was imported into the workspace. The data was re-organised from the imported table into a cell array. Since LSTM models are capable of multivariate predictions, some feature engineering was done. Several indicators were created such as weekday, time of use, and working hours to improve the model's performance. NOTE TO SELF: MUST TRY TO TRAIN WITHOUT THESE INDICATORS AND COMPARE

Data Partitionig

The array was grouped into batches to improve training efficiency. The number of time steps in each batch should be enough to provide information about the time series but not so much as to slow the training down extensively. Different size batches were tested and it was found that a batch size between 14 and 35 days provided the best results.

The dataset is split into a 70-30 ratio for training and testing purposes, ensuring that the temporal sequence is preserved to maintain the time-series integrity. A target matrix is also created. To forecast the values of future time steps, the targets are specified as the training sequences with values shifted by one time step. The training values are stored in the matrix XTrain and the training targets are stored in TTrain. The same idea is applied to the testing data with the test inputs being stored in XTest and the test targets being stored in TTest.

3.4.2 Normalisation

The mean and standard deviation for the training dataset were calculated. This was used to normalise, giving it a zero mean and unit standard deviation using the formula: $X_{normalised} = (X - mean)/sigma$ where $sigma$ denotes the standard deviation.
It is important to note that the testing partition was normalised using the mean and standard deviation calculated for the training data set. This code can be seen in figure 3.4.2.

```
    %Mean of training
meanX = mean(cat(2,XTrain{:}),2);
sigmaX = std(cat(2,XTrain{:}),0,2);
%Mean of training Targets
meanT = mean(cat(2,TTrain{:}),2);
sigmaT = std(cat(2,TTrain{:}),0,2);

%apply to the training data
```

```
for n = 1:numel(XTrain)
    XTrain{n} = (XTrain{n}-meanX)./sigmaX;
    TTrain{n} = (TTrain{n}-meanT)./sigmaT;
end
%apply to the test data
for n = 1:numel(XTest)
    XTest{n} = (XTest{n}-meanX)./sigmaX;
    TTest{n} = (TTest{n}-meanT)./sigmaT;
end
```

Listing 3.1: Code used to normalise data for LSTM training

The values for the mean and the standard deviation were stored so that the predictions could be transformed back into the original scale.

LSTM configuration and training

A basic LSTM configuration was first tested. With an input layer that corresponds to the size of the input matrix. An LSTM layer is used to learn the long-term pattern, with the fully connected layer feeding into the regression layer to estimate the next value. This can be see in listing 3.1

```
numChannels = size(dataCell{1},1);
layers = [
    sequenceInputLayer(numChannels)
    lstmLayer(128)
    fullyConnectedLayer(numChannels)
    regressionLayer];
```

Listing 3.2: LSTM configuration.

The ADAM optimiser was used to train the network due to its widespread popularity. The training seemed to converge around 2000 epochs with more training resulting in poor generalisation to the testing data. The left padding of sequences was done to prevent the network attempting to predict the empty values at the ends of inputs with different numbers of observations. These details can be seen in listing 3.2

```
    options = trainingOptions("adam", ...
    MaxEpochs=2000, ...
    SequencePaddingDirection="left", ...
    Plots="training-progress", ...
    Verbose=0);
```

Listing 3.3: LSTM training parameters.

Forecasting method

Closed loop forecasting is used to predict values **t** through to **t+k** when observations have been recorded for steps **1** through to **t-1**. When using an LSTM network in MATLAB it entails initialising the net to a set point. A prediction is then made for one step into the future. The net is then updated with the value of the prediction set to the current value. Another prediction is then made for the next step into the future (using the previous prediction value as the current value). This process is repeated for the number of time steps that are needed for the forecast. This is shown in lising **??**

```
        net = resetState(net);
        offset =  j*timeStep+1;
        [net,Z] = predictAndUpdateState(net,X(:,1:offset-1));

        numPredictionTimeSteps = 24;
        Xt = Z(:,offset-1);
        Y = zeros(numChannels, numPredictionTimeSteps); % the 2 is for the 2 output channels

        for t = 1:numPredictionTimeSteps
            [net,Y(:,t)] = predictAndUpdateState(net,Xt);
            Xt = Y(:,t);
        end
```

After all the predictions for a full 24 hours ahead have been made, values for all 24 timesteps are stored in an output array. The start point is then incremented by one timestep and another closed prediction for 24 hours is made. Using this method it is possible to make a 24-hour closed loop prediction starting from every timestep in the testing data. It is necessary to make lots of future predictions to be able to get an accurate estimate of the MAPE of the predictions.
An example of the output of what a day ahead forecast looks like can be seen in figure 3.13

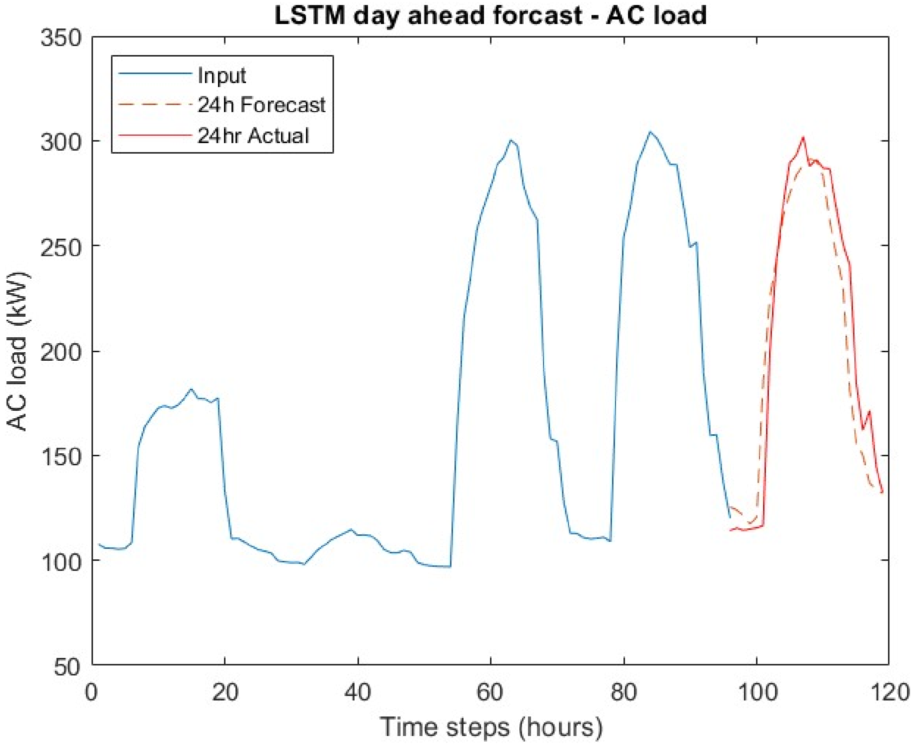


Figure 3.13: Day ahead forecast of the AC load using LSTM.

3.4.3 Fitting ARIMA model

Code for this section can be found here. (Very rough and still a working version)

Data Preparation

For ARIMA the data was streamlined to the core time-series. That is a timestamp, the AC power, and the DC power.

Determining Model Parameters

Determining the value of d: The ACF of the data was examined and a Kwiatkowski–Phillips–Schmidt–Shin (KPSS) test was performed to determine if the data was stationary or not. The KPSS test returned that it was not stationary and the ACF showed that the series had many significant lags.

```
% KPSS test on AC and DC loads
[h_AC, ~, ~, ~] = kpsstest(data.AC, 'Alpha', 0.05);
[h_DC, ~, ~, ~] = kpsstest(data.DC, 'Alpha', 0.05);
```

Listing 3.4: KPSS test being used to test the data for whether it is stationary.

The data was then differenced and the same tests were repeated. The KPSS test returned that the data was stationary and the ACF looked much better. Although there were still some significant correlations as can be seen in figure 3.14. Additionally, there is a clear spike in the ACF around the 24th lag. This indicates that there is a solid seasonal component at the 24th lag. This is a good reason to attempt a

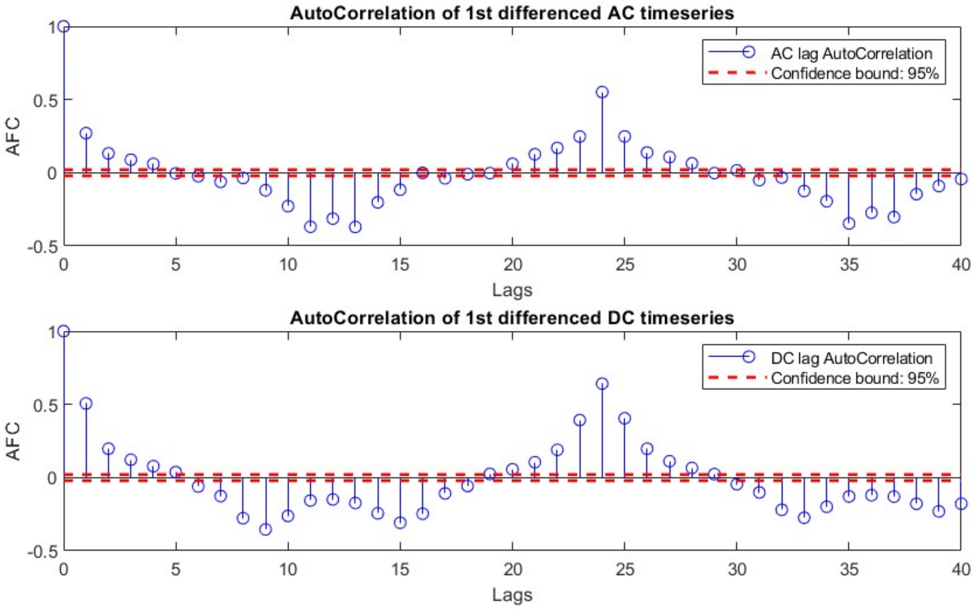


Figure 3.14: AFC of the 1st order differenced timeseries.

SARIMA or SARIMAX model. A second differencing was attempted but the lag components quickly became highly negative. Therefore the simpler model with first-order differencing was selected for the initial tests.

Determining the value of p: In order to determine the value of 'p' the PACF of the first differenced sequence was taken as seen in figure 3.15. This showed that only 1 lag was significantly over the confidence bound in both the AC and the DC timeseries. The AC timeseries did have a 2nd and 3rd lag that were marginally over the confidence bound.

It was decided that a value of 2 would be used for p in initial tests.

Determining the value of q: The value of 'q' can be determined from the ACF of the first-order differenced series, shown in figure 3.14. Both the AC and DC series have 1 lag significantly over the confidence boundary. They also have 2-3 and 2-4 lags respectively that are marginally over the confidence boundary.

The value of q was first set to 2 to keep the model simple.

Training the model

The econometrics toolbox in MATLAB was used to try variations of the expected model inputs determined above. It was found that a model with p = 3, d = 1, and q = 2, produced the most accurate forecasts.

The code used to create and train the model is shown below in listing 3.4.

```
ARIMA_AC = arima('Constant',NaN,'ARLags',1:2,'D',1,'MALags',1:2,'Distribution','Gaussian
    ↪ ');
ARIMA_AC = estimate(ARIMA_AC,X.AC,'Display','off');
```

Listing 3.5: Training of ARIMA model

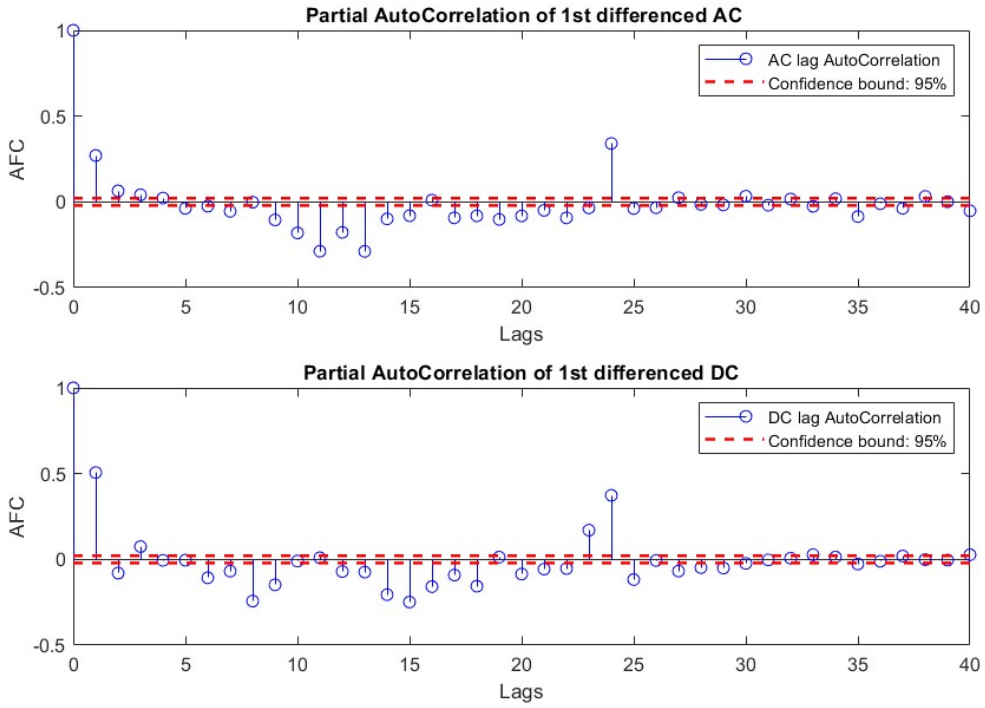


Figure 3.15: PAFC of the 1st order differenced timeseries.

Forecasting method

ARIMA models in MATLAB have a built in closed loop forecast function. This was used to forecast 24 steps and produce day-ahead predictions of the load as can be seen in figure 3.16.
This did not prove to be a very accurate model and it was clear that the seasonal component of the series needed to be taken into account. It was therefore decided to fit a SARIMA network.

3.4.4 Fitting SARIMA model

Have done but haven't written up yet.

Training Model

Forecasting method

3.4.5 Fitting ANN model

Have done but haven't written up yet.
Code for this section can be found here. (Very rough and still a working version)

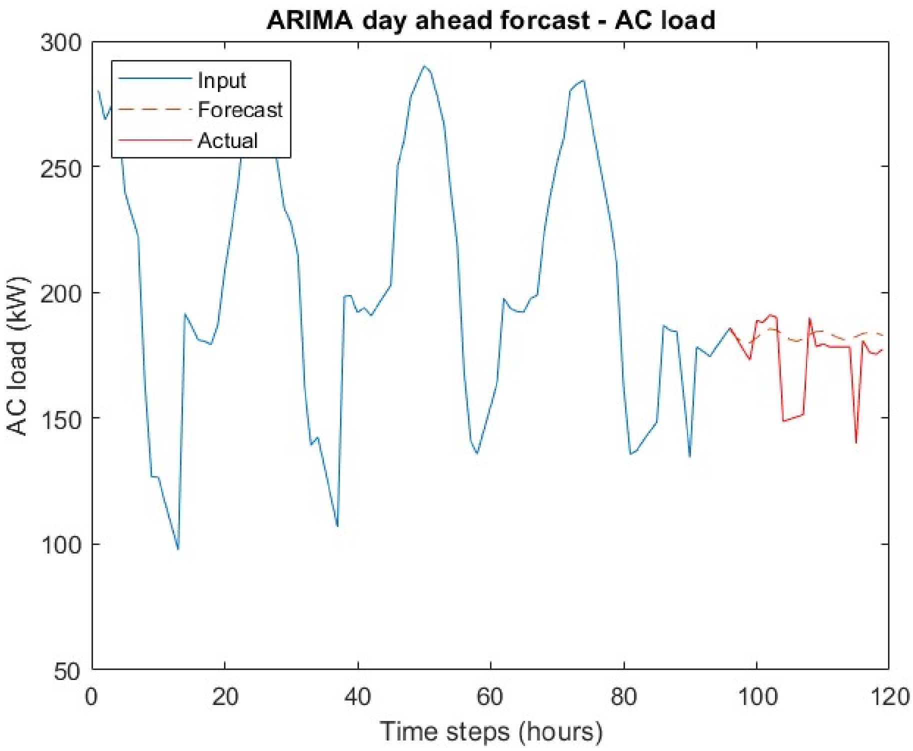


Figure 3.16: Day ahead forecast of the AC load using LSTM.

Data Preparation

Training Model

Forecasting method

3.5 Validation of Load Prediction

-Correlation between the Actual and the Forecast (corr) -Min-Max Error (minmax)

3.5.1 comparison of MSE, RMSE, MAPE

Testing and comparing the results of the Load and Generation Predictions. Maybe this should be the final step of the above section instead of its own section.

3.6 Weather data

3.6.1 Historical weather data

3.6.2 Weather forecasts

3.7 Generation Calculations

3.8 Modeling of Cases for Scheduling

The scheduling will most likely be done in a decision-making tree manner such as the one below. The decisions will be designed to optimise the cost function, for example, the total cost of meeting the load

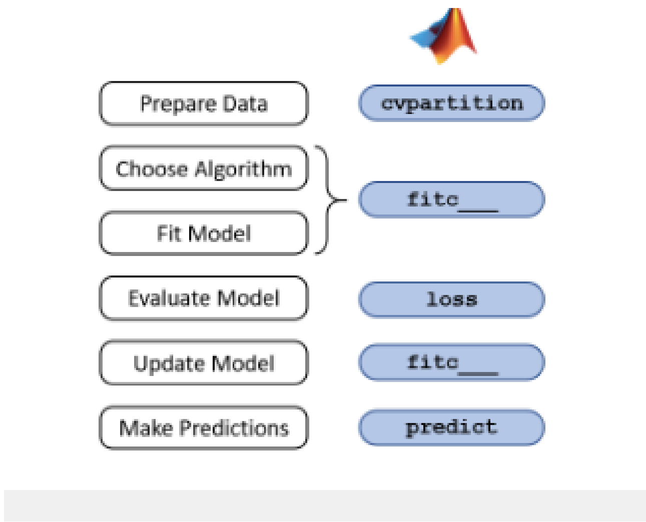


Figure 3.17: Machine Learning steps

demand.

3.9 Machine Learning/Linear Optimiser for schdeuling

If I use a machine learning approach to this I will separate all of the options into classes and use a classification model to make predictions of what the optimal approach is. This would be interesting and perhaps a more flexible approach, making it more applicable to other use cases.. Alternatively, the use of linear optimisers has show extremely good results as they are able to find global minimums for given problems. However, they require rigorous definitions and parameters which may make the solution less flexible

3.10 Validation of Scheduling Method

Again perhaps this should be included in the above section. This will be used to compare and validate the chosen and fitted scheduling algorithm

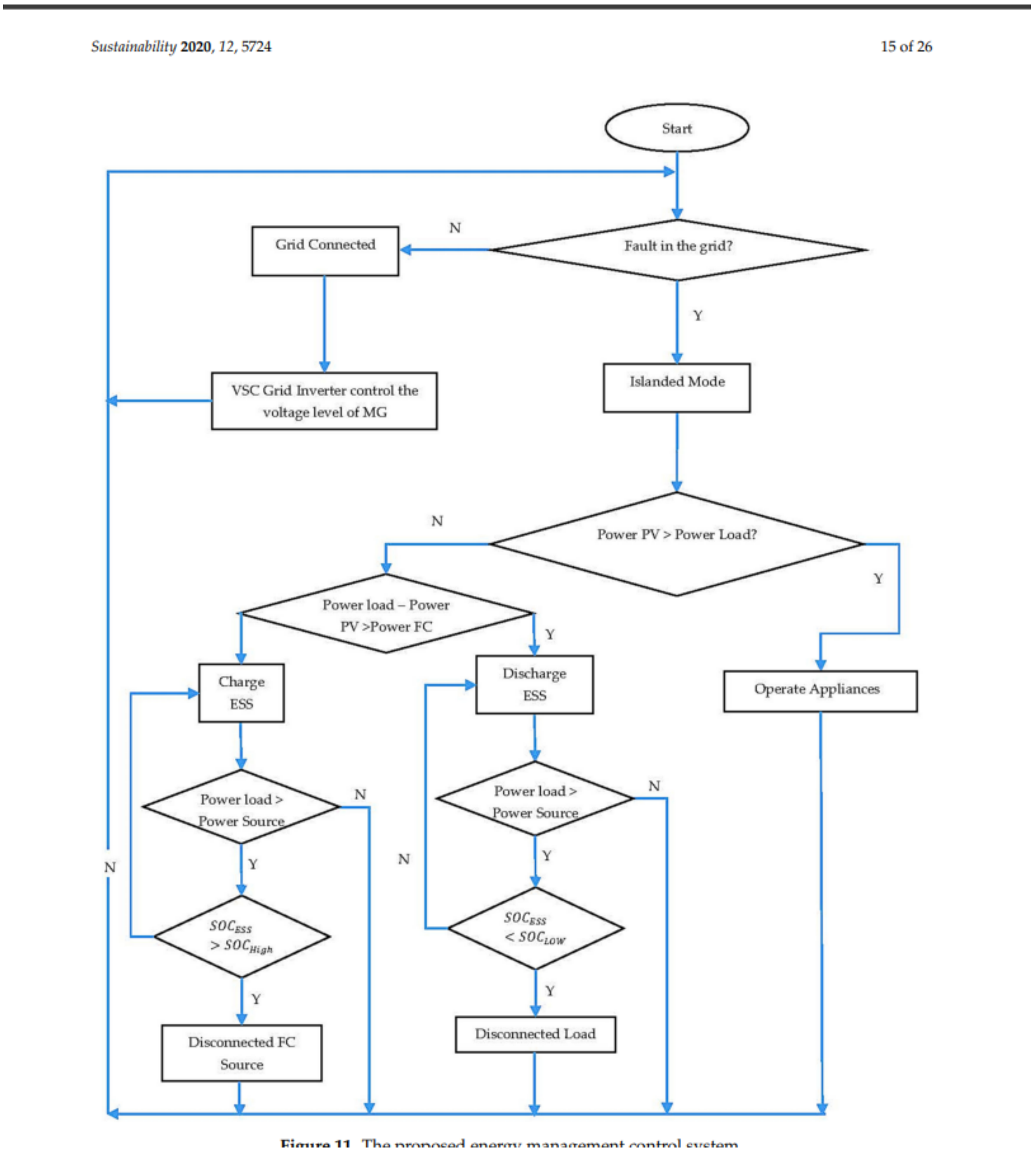


Figure 3.18: Example of Scheduling cases and flow

Chapter 4

Results

These are the results I found from my investigation.
Present your results in a suitable format using tables and graphs where necessary. Remember to refer to them in text and caption them properly.

4.1 Simulation Results

4.2 Experimental Results

Chapter 5

Discussion

Here is what the results mean and how they tie to existing literature...
Discuss the relevance of your results and how they fit into the theoretical work you described in your literature review.

Chapter 6

Conclusions

These are the conclusions from the investivation and how the investigation changes things in this field or contributes to current knowledge...
Draw suitable and intelligent conclusions from your results and subsequent discussion.

Chapter 7

Recommendations

Make sensible recommendations for further work.

Bibliography

[1] A. K. Barik, S. Jaiswal, and D. C. Das, "Recent trends and development in hybrid microgrid: a review on energy resource planning and control," *International Journal of Sustainable Energy*, vol. 41, no. 4, pp. 308–322, 2022. [Online]. Available: https://doi.org/10.1080/14786451.2021.1910698

[2] S. Moradi, G. Zizzo, S. Favuzza, and F. Massaro, "A stochastic approach for self-healing capability evaluation in active islanded ac/dc hybrid microgrids," *Sustainable Energy, Grids and Networks*, vol. 33, p. 100982, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2352467722002272

[3] R. A. Kaushik and N. M. Pindoriya, "A hybrid ac-dc microgrid: Opportunities & key issues in implementation," in *2014 International Conference on Green Computing Communication and Electrical Engineering (ICGCCEE)*, 2014, pp. 1–6.

[4] P. C. Loh, D. Li, Y. K. Chai, and F. Blaabjerg, "Autonomous operation of hybrid microgrid with ac and dc subgrids," *IEEE Transactions on Power Electronics*, vol. 28, no. 5, pp. 2214–2223, 2013.

[5] E. Unamuno and J. A. Barrena, "Hybrid ac/dc microgrids—part i: Review and classification of topologies," *Renewable and Sustainable Energy Reviews*, vol. 52, pp. 1251–1259, 2015. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1364032115008412

[6] K. Cabana-Jiménez, J. E. Candelo-Becerra, and V. Sousa Santos, "Comprehensive analysis of microgrids configurations and topologies," *Sustainability*, vol. 14, no. 3, 2022. [Online]. Available: https://www.mdpi.com/2071-1050/14/3/1056

[7] B. Naji Alhasnawi, B. H. Jasim, and M. D. Esteban, "A new robust energy management and control strategy for a hybrid microgrid system based on green energy," *Sustainability*, vol. 12, no. 14, 2020. [Online]. Available: https://www.mdpi.com/2071-1050/12/14/5724

[8] Z. Ullah, M. Elkadeem, K. M. Kotb, I. B. Taha, and S. Wang, "Multi-criteria decision-making model for optimal planning of on/off grid hybrid solar, wind, hydro, biomass clean electricity supply," *Renewable Energy*, vol. 179, pp. 885–910, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0960148121010673

[9] Z. Ullah, S. Wang, J. Lai, M. Azam, F. Badshah, G. Wu, and M. R. Elkadeem, "Implementation of various control methods for the efficient energy management in hybrid microgrid system," *Ain Shams Engineering Journal*, vol. 14, no. 5, p. 101961, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2090447922002726

[10] J. M. Lujano-Rojas, C. Monteiro, R. Dufo-López, and J. L. Bernal-Agustín, "Optimum load management strategy for wind/diesel/battery hybrid power systems," *Renewable Energy*, vol. 44, pp. 288–295, 2012. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0960148112001243

[11] N. Attou, S.-A. Zidi, M. Khatir, and S. Hadjeri, "Energy management system for hybrid microgrids." *Electrotehnica, Electronica, Automatica*, vol. 69, no. 2, 2021.

[12] S. Kotra and M. K. Mishra, "A supervisory power management system for a hybrid microgrid with hess," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 5, pp. 3640–3649, 2017.

[13] M. Hosseinzadeh and F. R. Salmasi, “Power management of an isolated hybrid ac/dc micro-grid with fuzzy control of battery banks,” *IET Renewable Power Generation*, vol. 9, no. 5, pp. 484–493, 2015. [Online]. Available: https://ietresearch.onlinelibrary.wiley.com/doi/abs/10.1049/iet-rpg.2014.0271

[14] A. C. Luna, N. L. Diaz, M. Graells, J. C. Vasquez, and J. M. Guerrero, “Mixed-integer-linear-programming-based energy management system for hybrid pv-wind-battery microgrids: Modeling, design, and experimental verification,” *IEEE Transactions on Power Electronics*, vol. 32, no. 4, pp. 2769–2783, 2017.

[15] L. Lv, Z. Wu, L. Zhang, B. B. Gupta, and Z. Tian, “An edge-ai based forecasting approach for improving smart microgrid efficiency,” *IEEE Transactions on Industrial Informatics*, vol. 18, no. 11, pp. 7946–7954, 2022.

[16] M. Afrasiabi, M. Mohammadi, M. Rastegar, and A. Kargarian, “Multi-agent microgrid energy management based on deep learning forecaster,” *Energy*, vol. 186, p. 115873, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0360544219315452

[17] H. H. Aly, “A proposed intelligent short-term load forecasting hybrid models of ann, wnn and kf based on clustering techniques for smart grid,” *Electric Power Systems Research*, vol. 182, p. 106191, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0378779619305103

[18] Y. Li, R. Wang, and Z. Yang, “Optimal scheduling of isolated microgrids using automated reinforcement learning-based multi-period forecasting,” *IEEE Transactions on Sustainable Energy*, vol. 13, no. 1, pp. 159–169, 2022.

[19] H. Zimmermann, R. Grothmann, C. Tietz, and H. von Jouanne-Diedrich, “Market modeling, forecasting and risk analysis with historical consistent neural networks,” *Studies in Computational Intelligence*, vol. 410, pp. 531–536, 06 2011.

[20] H. Musbah, H. H. Aly, and T. A. Little, “Energy management of hybrid energy system sources based on machine learning classification algorithms,” *Electric Power Systems Research*, vol. 199, p. 107436, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S037877962100417X

[21] K. Zhou, K. Zhou, and S. Yang, “Reinforcement learning-based scheduling strategy for energy storage in microgrid,” *Journal of Energy Storage*, vol. 51, p. 104379, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2352152X22004030

[22] A. Bayas, “Machine learning with matlab.” [Online]. Available: https://matlabacademy.mathworks.com/R2023a/portal.html?course=mlml#chapter=5&lesson=4&section=2

[23] Y. Hua, Z. Zhao, R. Li, X. Chen, Z. Liu, and H. Zhang, “Deep learning with long short-term memory for time series prediction,” *IEEE Communications Magazine*, vol. 57, no. 6, pp. 114–119, 2019.

[24] F. Angizeh, A. Ghofrani, and M. A. Jafari, “Dataset on hourly load profiles for a set of 24 facilities from industrial, commercial, and residential end-use sectors,” 2020.

Appendix A

Additional Files for Design of Hybrid Microgrid

Name of panel	Num units	Capacity (W)	Price (R)	Price Per Kw	Link
Jinko Solar Panel Tiger MonoFacial	1	470	1895	4031.91	Link
Jinko Solar Panel Tiger Mono-Facial Pallet	31	555	67275	3910.20	link
JA Solar 144 cell Module	1	465	2192	4713.98	Link
Canadian Solar 555W Super High Power Mono - Pallet	31	555	79143	4600.00	Link
Canadian Solar 545W panel	1	545	2575	4724.77	Link

Table A.1: Prices of Solar PV units.

Appendix B

Addenda

B.1 Ethics Forms