Synnøve Hovden

An Optimal Model Predictive Control-Based Energy Management System for Microgrids

Master's thesis in Energy and Environmental Engineering
Supervisor: Olimpo Anaya-Lara
Co-supervisor: Raymundo E. Torres-Olguin
June 2021

Master's thesis

NTNU
Norwegian University of
Science and Technology

Synnøve Hovden

An Optimal Model Predictive Control-Based Energy Management System for Microgrids

NTNU
Norwegian University of
Science and Technology

Preface

This thesis is the final requirement for acquiring a Master of Science degree in Energy and Environmental Engineering with the Department of Electric Power Engineering at the Norwegian University of Science and Technology. The work is a continuation of the specialization project completed in the fall of 2020, titled "Energy Management in Microgrids - Fundamentals, modeling and simulations" [1]. Moreover, this thesis is a part of the activities in WP4 in the FME-CINELDI.

I would like to thank my supervisors, Olimpo Anaya-Lara and Raymundo Torres-Olguin, for guiding me through this thesis and the end chapter of my academic studies. I sincerely acknowledge the support, guidance, advice, and encouragement they have given me. Further, I would like to thank Lede for providing me with PV and load data. Finally, I would like express gratitude towards my friends and family for always believing in me and supporting me. Thank you!

Trondheim, June 2021.
Synnøve Hovden

Abstract

The concept of microgrids is considered a promising building block for realizing the modern and future power system much due to its ability to integrate distributed energy resources, energy storage systems, and controllable loads. However, to utilize the full potential of microgrids, they need to be controlled and managed optimally. Therefore, this thesis aims to develop optimal control methods to perform the energy management of a grid-connected microgrid. Following an extensive literature review on microgrid energy management systems, the first phase of this work developed a microgrid simulation platform in MATLAB/Simulink. This platform utilized a variable-step phasor solving method to simulate a grid-connected microgrid comprising a photovoltaic (PV) system, a variable load, a static load, and a battery, including a degradation model.

In the second phase of this work, two energy management strategies were developed to determine the battery charging and discharging power set-points. As a first step, a simple heuristic method was developed to work as a reference for comparison. Further, an optimization-based scheduling algorithm based on the model predictive control (MPC) approach was proposed. The overall problem was formulated using mixed-integer linear programming (MILP), which can effectively be solved using commercially available solvers resulting in significant improvements in solution quality and computational burden. In this work, the power converter efficiency and battery capacity were considered time-varying by updating the values at each sample time and assuming them constant over the prediction horizon. Consequently, the resulting energy management strategy was cast as a multi-objective MILP problem incorporated in an MPC framework to account for disturbances and to capture some of the nonlinear dynamics of the system.

Finally, the proposed control approaches were investigated through an extensive case study over a two-month simulation period using actual PV, load, and electricity price data. For all cases, the MILP-MPC control algorithm succeeded in reducing the daily cost of the energy drawn from the main grid compared to the heuristic algorithm. Furthermore, depending on the chosen settings, the results showed that the MILP-MPC energy management strategy managed to determine the reference values for the battery power in a way that: (1) minimized the purchased energy during peak times; (2) maximized self-consumption of locally produced PV power; (3) made good use of the battery, keeping it within its limits and reducing its degradation. Thus, the result is a flexible algorithm that can be tuned depending on the overall control objective. Moreover, the two-month simulation was performed within an appropriate execution time using a short sample time of five minutes, which enables real-time operation.

Sammendrag

Mikronett anses som lovende byggesteiner for å realisere fremtidens smarte kraftsystem siden de legger til rette for å integrere distribuerte energikilder, energilagringssystemer, og kontrollerbare laster. For å utnytte det fulle potensialet av mikronett er det avgjørende at de styres og kontrolleres optimalt. Derfor er formålet med denne masteroppgaven å utvikle optimale kontrollmetoder for energistyringen i et nettilknyttet mikronett. Etter en omfattende litteraturstudie om energistyringssystemer (EMS) for mikronett, utviklet den første fasen av dette masterprosjektet en mikronettsimuleringsplattform i MATLAB/Simulink. Denne plattformen benyttet en fasorbasert løsningsmetode med variabel steglengde for å simulere et nettilknyttet mikronett bestående av et solcelleanlegg, en variabel last, en statisk last, og et batteri med en nedbrytningsmodell.

I den neste fasen av dette masterprosjektet ble det utviklet to energistyringssystemer for å kontrollere mikronettets batteri. Først ble det utviklet en enkel heuristisk metode som skulle fungere som en referanse for sammenligning. Videre ble det designet en optimaliseringsbasert algoritme basert på modellprediktiv kontroll (MPC). Det overordnede problemet ble formulert ved hjelp av blandet lineær heltallsoptimering (MILP). Denne typen optimeringsproblemer kan effektivt løses ved hjelp av kommersielt tilgjengelige programvarer for å oppnå mer nøyaktige resultater med lavere løsningshastighet. Metoden inkluderer variabel kraftelektronikk-virkningsgrad og batterikapasitet ved å oppdatere verdiene for hvert nye tidssteg og anse dem som konstante utover predikeringshorisonten. Dermed kan den utviklede kontrollalgoritmen anses som et multi-objektivt MILP-problem innlemmet i et MPC rammeverk for å kompensere for forstyrrelser og for å inkludere noe av den ikke-lineære dynamikken i systemet.

Avslutningsvis ble de foreslåtte kontrollmetodene undersøkt gjennom en omfattende casestudie over en to-måneders simuleringsperiode ved bruk av faktiske data for solkraftproduksjon, laster, og elektrisitetspriser. I alle beregningene lyktes MILP-MPC kontrollalgoritmen i å redusere de daglige energikostnadene sammenliknet med den heuristiske metoden. Videre viste resultatene at MILP-MPC kontrollalgoritmen klarte å fastsette referanseverdier for batteriet på en måte som: (1) minimerte innkjøp av energi fra det overordnede strømnettet ved pristopper; (2) maksimerte forbruk av egenprodusert solkraft; (3) utnyttet batteriet på en god måte innenfor dets grenser slik at nedbrytingen av batteriet ble minimert. Dermed kan det konkluderes med at det utviklede optimale energistyringssystemet er en fleksibel algoritme som kan konfigureres avhengig av det overordende styringsmålet. I tillegg ble en to-måneders simulering gjennomført med et kort tidssteg på fem minutter innenfor en passende kjøretid, noe som muliggjør sanntidsdrift.

Contents

List of figures

List of tables

Acronyms

ANN Artificial neutral networks

BESS Battery energy storage system

CIGRÉ Conseil International des Grandes Réseaux Électriques

DG Distributed generation

DOC Depth of cycle

DP Dynamic programming

EMS Energy management system

ESS Energy storage system

GA Genetic algorithms

HBC Hysteresis band control

ICT Information and communication technology

MILP Mixed-integer linear programming

MINLP Mixed-integer nonlinear programming

MPC Model predictive control

NLP nonlinear programming

PCC Point of common coupling

PSO Particle swarm optimization

PV Photovoltaic

RES Renewable energy sources

SimSES Software for techno-economic simulation of stationary energy storage systems

SOC State of charge

SOH State of health

TS Tabu search

1 Introduction

The purpose of this chapter is to provide a background and a motivation for the thesis and use this to formulate the overall problem. Further, this chapter will describe the thesis' objectives, state the methodology adopted to achieve the objectives, limit the scope of the research, highlight the contributions, and, finally, outline the contents of the report.

This master's thesis builds upon the work conducted by the author during the execution of the specialization project [1]. In order to give proper context to the present work, some elements in the background section have been reproduced here. The motivation remains as originally defined.

1.1 Background and motivation

The power system is currently in the middle of a significant transition. Traditionally, the power system has been based on a centralized structure with large-scale power generation that is transmitted in several stages to the end consumer [2], as illustrated in Figure 1.1. Some advantages associated with this structure are unidirectional power flows and few generating units with a reliable and predictable supply. Moreover, the traditional power system has an inherent ability to regulate frequency due to the high inertia provided by the large synchronous generators. These systems are therefore easy to adjust to changing demand cycles and have historically managed to deliver power to consumers in a reliable, secure, and safe way.
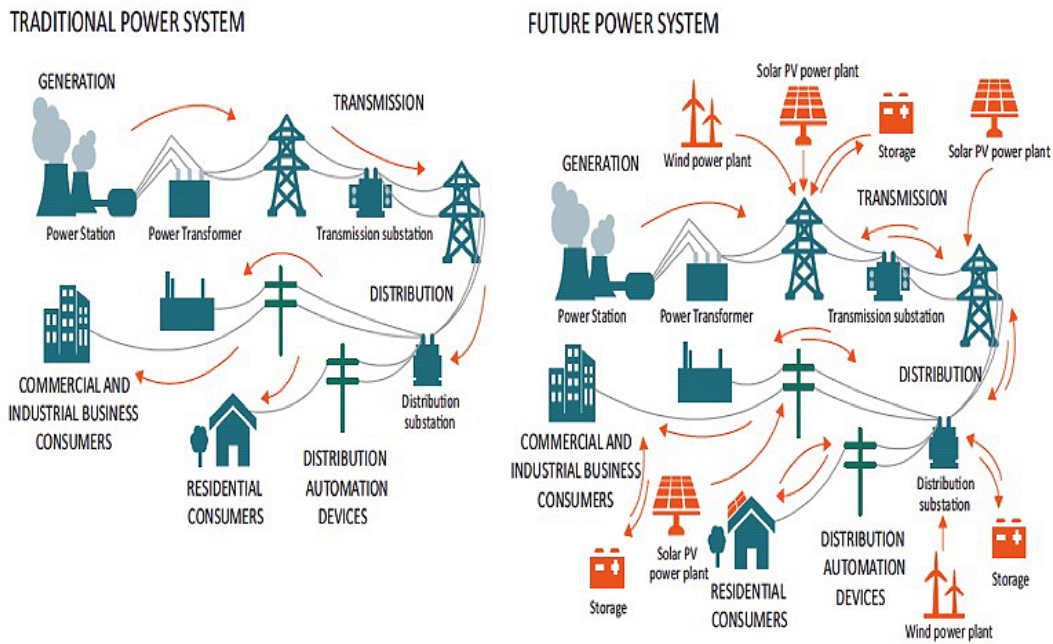
Figure 1.1: Overview of the traditional power system and the future power system [3].

However, due to technological advancements and the demand for a more sustainable future, the power system is currently undergoing massive changes that challenge the conventional grid structure. One of the main changes is the large-scale deployment of variable and unpredictable renewable energy sources (RES). Introducing RES to the power system will result in a more decentralized structure, where generation also happens at the distribution level closer to the end consumer, which is made evident in Figure 1.1. This leads to lower transmission losses but also bidirectional power flows [4]. Moreover, as opposed to the controllable generation in the conventional power system, RESs do not allow the adjustment of production to demand. Hence, energy storage systems (ESS) will have a central role in the future power system. Other changes to the power system include increased use of information and communication technology (ICT) and more active consumers [5]. The traditional power system has a limited ability to face these changes, and it is thus necessary to rethink the way the power system is organized.

The concept of microgrids is considered a promising building block for realizing the modern and future power system. Microgrids are small-scale power systems that can operate either connected to the main grid, or while islanded. They typically comprise local control systems, distributed generation units (DGs), distributed energy storage systems (ESS), and controllable loads [6]. The result is a more active low-voltage distribution side that

participates in the operation of the power system. This can be advantageous for integrating RESs, optimizing operation, providing a secure power supply, and meeting challenges related to variable generation, bidirectional power flows, and more active consumers.

However, to fully utilize the benefits microgrids bring, they need to be controlled and managed optimally. This can be a challenging task and a prerequisite for optimal operation of a microgrid is a properly designed energy management system (EMS). An EMS is an inter-disciplinary system that utilizes ICT to ensure optimal coordination between the microgrid units to supply reliable, sustainable, and high-quality energy in a cost-efficient way. This system must be capable of supervising, planning, and controlling the microgrid operation to manage not only technical but also economic and environmental issues [7, 8].

Based on the above background and motivation, this work intends to contribute to the area of research by developing, implementing, and testing a microgrid EMS. The next subsection will formulate the overall problem of this thesis, including the components of the studied microgrid and the requirements placed on its EMS.

1.2 Problem formulation

Motivated by the microgrid challenges related to optimal control and management, the purpose of this master's thesis is to develop optimal control methods that can be used in the energy management of a microgrid. The microgrid considered in this thesis comprises a PV system, residential loads, and a battery. Along with the coordination of the microgrid generation and demand, the proposed energy management system should be able to:

- Account for the variable and unpredictable nature of the PV system, the residential loads, and the electricity prices in the daily operation.

- Simultaneously present solutions to multiple objectives, including minimization of the daily operating costs, maximization of locally produced PV, and making good use of the battery, keeping it within its limits and reducing its degradation.

- Incorporate battery models that include nonlinearities associated with power conversion losses and battery capacity degradation.

- Control and operate the microgrid in real-time, and not just present a day-ahead schedule. In this context, the EMS should reach an appropriate trade-off between accuracy and computational efforts.

- Apply methods that are long-term in nature over a specified prediction horizon instead of considering the present time only.

Generally, optimization-based methods improve the overall performance of the microgrid operation when compared to heuristic methods. However, solving optimization-based techniques offline for day-ahead scheduling of the microgrid units often results in constraint violations and poor performance in real-time scenarios. One way to address this challenge, is to solve the optimization problem within a model predictive control (MPC) framework, i.e., dynamically adjust the schedule according to real-time conditions. MPC includes several features suitable for addressing the requirements listed above. Some of these features are [9, 10, 11]:

- *Optimization-based.* It enables a microgrid optimization problem to be formulated and solved by the controllers using several solving algorithms. In this way, an appropriate solving algorithm can be implemented to effectively handle the constraints associated with microgrids, at the same time as the optimization of multiple objectives is made possible.

- *Feedback mechanism.* It enables the system to compensate for errors in generation, demand, and electricity price forecasts. In addition, the feedback mechanism allows the system model to be updated every sample time. In this way, the current battery capacity and power electronic efficiencies can be used in the control.

- *Receding prediction horizon.* It considers the future behaviour of the system in the control, and it is thus long-term in nature.

- *Dynamic operation.* It is suitable for making real-time control decisions.

MPC can incorporate any optimization method, and selecting a suitable method can be a challenging task. Based on the extensive review of energy management methods performed in Chapter 2 of this thesis, mixed-integer linear programming (MILP) is selected due to less computational effort, compatibility with available solvers, and a guaranteed optimal solution without noticeable loss of accuracy compared to nonlinear methods. MILP has been successfully incorporated with MPC in several previous studies [12, 13, 14], and some deficiencies observed in these studies are:

- Concerning the battery modeling, the battery capacity is often assumed constant, while in reality, it decreases as the battery is used. This is a valid assumption for a

short period of time, but it will affect the results in the long run. Considering the variable battery capacity in the problem formulation involves high nonlinearities.

- Although much work aims at limiting battery degradation, few evaluate the performance of the proposed algorithm in terms of battery aging. The algorithms are often investigated and verified in a microgrid simulation platform where the incorporated battery model neglects degradation, making it difficult to properly evaluate how the schedule provided by the EMS will affect the battery.

- Simulations are typically performed for one day, which does not show the long-term effects.

- Sampling times ranging from 15 minutes to 2 hours are commonly used. A shorter sampling time is preferable because it allows the EMS to observe and respond to small changes in the load, generation, and electricity price throughout the day.

- The power converter efficiency is often neglected or assumed constant while in reality, it depends on the battery power. Considering the variable efficiency of power converters in the problem formulation involves high nonlinearities.

1.3   Goal and objectives

Based on the above problem formulation, the overall goal of this thesis is to

Develop a microgrid energy management control approach that combines MPC with MILP to effectively account for uncertainties and to capture some of the nonlinear dynamics of the system by updating the system model every time step.

Four secondary objectives are derivated for the main goal:

- Perform a literature review to investigate the state of the art in EMS to select a method applicable for controlling the microgrid considered in this thesis.

- Create a microgrid simulation platform in MATLAB/Simulink suitable for implementing and testing an EMS. The simulation platform should include a battery degradation model to properly evaluate how the EMS affects the battery.

- Propose and implement two energy management strategies, namely a high-level optimization algorithm for a grid-connected microgrid using MPC in combination

with MILP and a heuristic management strategy for comparison.

- Investigate the performance of the proposed control approaches through an extensive case study over a two-month simulation period using actual PV and load data from Skagerak Energilab and electricity price profiles from Nordpool.

1.4   Methodology

A step-by-step methodology was applied to achieve the objectives of this thesis. The methodology included the following steps:

1. Perform a thorough literature review of microgrid energy management systems to obtain a solid theoretical foundation and to understand the main gaps and challenges.

2. Choose the MILP-MPC control approach for controlling the microgrid based on the literature review.

3. Implement a phasor microgrid simulation platform in MATLAB/Simulink by modifying and combining elements from two existing models, namely the microgrid component library developed by an application engineer at the MathWorks, Jonathan LeSage[15], and the SimSES battery model developed by Maik Naumann and Nam Truong at the Technical University of Munich [16].

4. Design and implement a MILP-MPC control approach by formulating and solving a MILP energy management optimization problem over a receding horizon using the Optimization Toolbox in MATLAB.

5. Design and implement a heuristic control method as a reference for comparing the performance of the proposed MILP-MPC algorithm.

6. Propose seven case studies to assess the dynamic performance of the developed control algorithms for a two-month simulation time.

7. Collect real-life data from Skagerak Energilab and Nordpool

8. Analyze the results of the case study and conclude the work.

To aid in the investigation of the developed control methods, actual PV and load data were provided by Skagerak Energilab [17]. Skagerak Energilab is a testing facility for local production, storage, and distribution of electrical energy located in Skien in Norway. In

particular, this testing facility comprises a PV system, residential loads, and a battery that are combined to make up a virtual microgrid. The project is run by Lede in collaboration with ABB, Kontorbygg AS, and Odds Ballklubb.

1.5   Scope and main assumptions

To achieve the objectives of the master's thesis in time, the scope is limited by the following assumptions and limitations:

- The microgrid model is built for performing high-level optimization, meaning that the control of frequency, power quality, and voltage stability is assumed ideal and performed at a lower control level. The higher control level also works with long-term behavior and transients are therefore neglected.

- The developed microgrid model is grid-connected. Hence, island mode and related issues are not considered in the model development.

- The well-known SimSES battery model has been used to simulate the complete behaviour of the battery including degradation. It is outside the scope to build new models. Moreover, no verification of the SimSES battery model is conducted because it was extensively verified by Maik Naumann in his doctoral dissertation [18]. SimSES has also successfully been used in several publications [19, 20, 21, 22].

- Developing a forecasting algorithm to predict the PV production, load demand, and electricity prices is outside the scope. Instead, an error function is developed to simulate uncertainty in forecasts by adding errors to the actual values. These errors are modelled with a gradient uncertainty level in which the forecast error increases when the prediction horizon becomes larger.

- A comprehensive cost analysis including battery investment and net present value is outside the scope. However, energy costs are calculated and discussed.

1.6   Relation to Specialization Project

During the fall of 2020, the author wrote a specialization project titled "Energy Management of Microgrids - fundamentals, modeling, and simulations"[1]. The specialization project provided a solid foundation for the master's thesis by gaining insight into the theory,

modeling, and simulation of a microgrid and its energy management system. The master's thesis takes this work further by developing a more sophisticated EMS and incorporating a more realistic battery model into the simulation platform.

Some sections in this thesis contain material reused from the specialization project, where most of the material is modified. The sections including some reused material are listed below and will not be further referenced in the running text:

- The background and motivation in Section 1.1.

- Parts of Sections 2.1.1 and 2.2 about control of microgrids and the microgrid energy management system.

- The model description of the PV system, the loads, and the utility point-of-connection in Sections 3.2.2, 3.2.3, and 3.2.4.

1.7   Key contributions

The key contributions of this master's thesis are:

- The generation of guidelines and suggestions of energy management strategies.

- The development of a microgrid simulation platform in MATLAB/Simulink suitable for implementing and testing energy management strategies. This platform utilizes a variable-step phasor solving method to simulate a grid-connected microgrid comprising a PV system, a variable load, a static load, and a complete battery model including degradation and power-dependent converter efficiencies

- The design of a flexible, multi-objective MILP-MPC energy management strategy that effectively accounts for uncertainties in load, PV, and electricity price forecasts in addition to dealing with nonlinearities associated with power conversion losses and battery capacity degradation.

- The presentation of simulation results showing the effectiveness of the proposed MILP-MPC algorithm where it successfully used real data to perform a two-month simulation using a five minutes sample time within a short execution time.

1.8   Structure of the report

The master's thesis is organized into different chapters, and the main purpose of each chapter is described below.

- Chapter 2 presents information on microgrid control and management and performs an extensive review of existing energy management methodologies.

- Chapter 3 develops a microgrid simulation platform in MATLAB/Simulink compatible with the implementation and testing of an EMS.

- Chapter 4 proposes two microgrid energy management strategies: one simple heuristic algorithm and a more sophisticated MILP-MPC algorithm.

- Chapter 5 investigates the performance of the proposed control methods through an extensive case study in MATLAB/Simulink.

- Chapter 6 gives the main conclusions and provides suggestions for further work.

2 Literature review on microgrid Energy Management Systems

This chapter will first give a definition of microgrids and provide an overview of typical control approaches for microgrid operation. Next, the microgrid energy management system is introduced, and its role in the microgrid control structure is defined. Finally, an extensive review of energy management strategies is conducted to select an appropriate method for controlling the microgrid considered in this thesis.

2.1 Microgrids and the modern energy system

Several steps must be taken before the traditional grid structure can be transformed into a smarter, greener, and more efficient modern grid. These steps include the deployment of distributed generation units (DGs), with emphasis on renewable energy sources (RESs). Further, the issues associated with a high penetration of DGs can be mitigated by connecting energy storage systems (ESSs), more active consumers, increased use of information and communication technology (ICT), and the development of new control systems. A microgrid comprising the above-mentioned components can result in a more efficient, reliable, and greener way of organizing the power system.

Relevant literature gives several definitions of what a microgrid is, and even though the different definitions vary to some extent, they all include the same key characteristics. For the purpose of this thesis, the microgrid definition established by the Conseil International des Grandes Réseaux Électriques (CIGRÉ) is used:

> "Microgrids are electricity distribution systems containing loads and distributed energy resources, (such as distributed generators, storage devices, or controllable loads) that can be operated in a controlled, coordinated way either while connected to the main power network or while islanded [23]."

In other words, a microgrid is a small-scale power system that can operate either in a non-autonomous way while connected to the main grid, or in an autonomous way while islanded. It is connected to the main grid at the point of common coupling (PCC), and consists of local control systems, DGs, distributed ESSs, and controllable and non-controllable loads [6]. Moreover, a microgrid has clear electrical boundaries and acts as one single

controllable entity with respect to the main grid. Figure 2.1 depicts a typical microgrid.
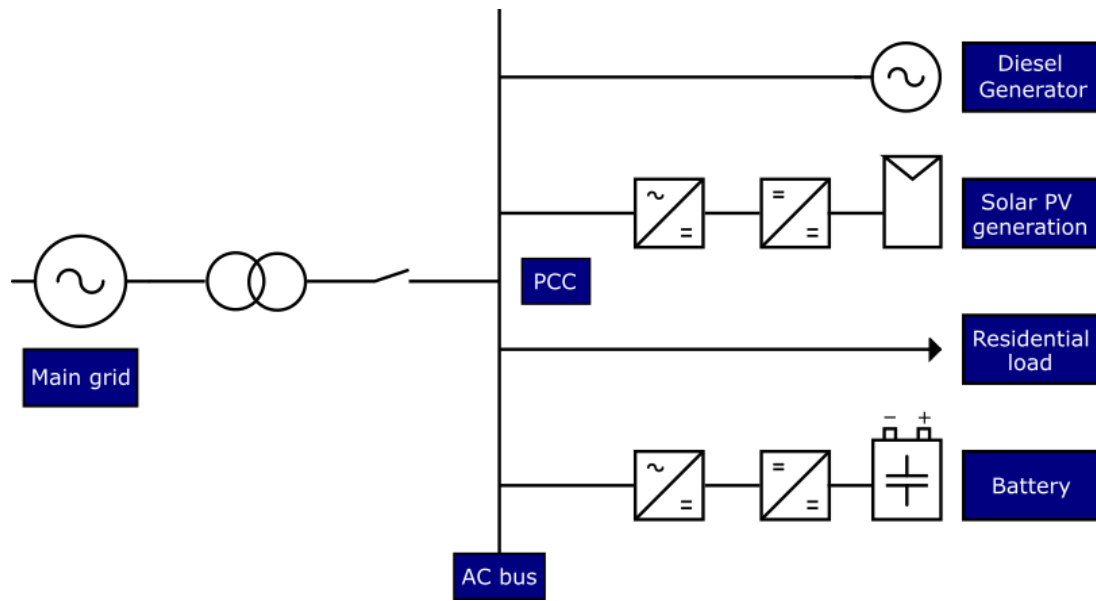
Figure 2.1: A basic AC microgrid comprising generation, storage, power electronic converters, and loads. Inspired by Figure 1.3 in [24].

There are several advantages to configuring a power system as a microgrid. The configuration will be essential in the shift towards a smarter grid and will work as a beneficial solution for pilot projects, enabling the testing of modern smart grid technologies [25]. Microgrids also facilitate the implementation of renewable DGs such as wind turbines, photovoltaic (PV) systems, and small-scale hydropower [6, 26]. Moreover, the ability to switch to island mode of operation when faults and contingencies occur in the main grid improves reliability from the end-user perspective. With autonomous control structures and local generation and storage of energy, microgrids alleviate the dependency and, consequently, the stress on the existing grid. In addition, if managed and coordinated optimally, local generation and storage of energy mitigate distribution losses and costs.

However, to utilize the full potential of microgrids, several challenges must be addressed. Central technical challenges found in the current literature consist of:

- *Bidirectional power flow.* Unlike conventional grids, generation in microgrids also happens at the distribution level closer to the end consumer. This causes bidirectional power flows, giving rise to complications in protection systems and undesired flow behavior [6, 25].

- *Low inertia.* The absence of synchronous generators in microgrids with a high number of power electronic converters results in low inertia. Microgrids thus have no inherent

stability mechanism, and adequate control mechanisms must be implemented to avoid large frequency and voltage deviations in island mode [6, 27].

- *Uncertainty.* Microgrids experience a significant uncertainty associated with demand and generation since the use of RES ties generation to environmental conditions. Therefore, to obtain a reliable and economical operation, the control system should include forecasts of generation, demand, and electricity prices [9].

- *Coordination between entities.* Coordinating the microgrid units becomes additionally challenging when considering factors such as power balance, component failure rates, weather forecast, and uncertain and variable load and generation profiles. Additionally, it must be confirmed that all components are compatible with each other [9, 26].

- *Stability.* The control systems in the microgrid may lead to local oscillations. In addition, the transition between grid-connected and island mode of operation causes issues in terms of stability [25].

To effectively cope with these challenges, the control system must guarantee a reliable operation of the microgrid. The next subsection gives an overview of the microgrid control system and selects a control area for further investigation.

2.1.1   Control of microgrids

Controls are crucial in the operation of microgrids. The microgrid control system must be able to handle the aforementioned characteristics of microgrids while simultaneously ensuring a reliable and economical operation. Desirable features of the control system include frequency and voltage regulation in both operational modes; control of currents and voltages in the DG units by tracking references and appropriately damping oscillations; seamless transition between operational modes; optimizing the microgrid operating costs; sharing power among the microgrid units; and more [26, 28].

The microgrid operation can be managed through several control architectures, as illustrated by Figure 2.2. A control system is **centralized** if there is a central controller that sends control signals to each controllable agent based on data from the microgrid components and the external grid. The control is **decentralized** when local control of each microgrid unit is performed without exchanging information with other units,

except for some leader agents that transmit and receive information through the center. When the local controllers utilize a communication network to exchange information and find a cooperative solution to the overall control problem, the control system is **distributed**. Finally, a compromise between a fully centralized and a fully decentralized control architecture can be obtained by implementing a **hierarchical** control scheme, where centralized and non-centralized methods can be utilized within each hierarchy level [7, 28, 29, 30, 31].
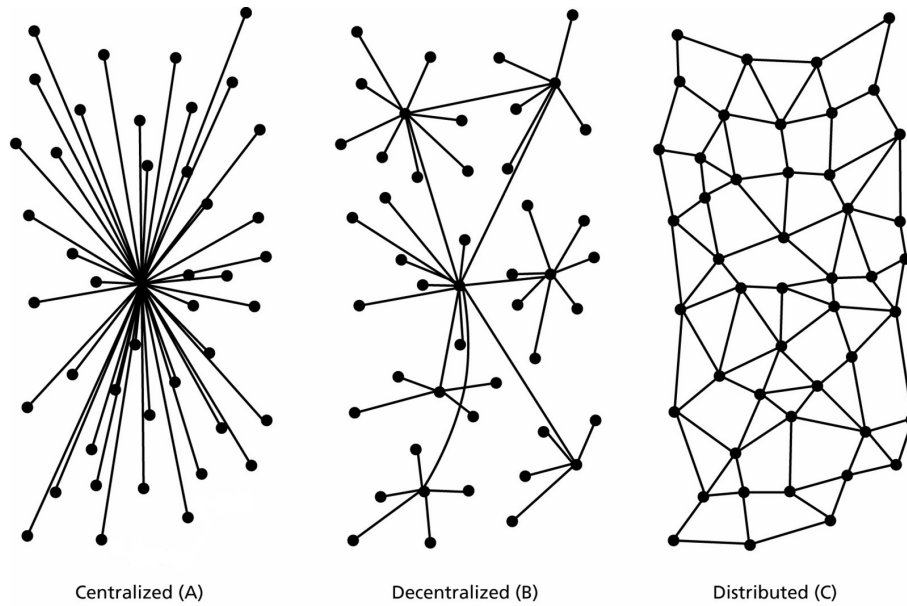
Figure 2.2:   Overview of the centralized, decentralized, and distributed control architectures [29].

The hierarchical control strategy has been widely accepted as the standardized control approach for microgrids [26, 31, 32, 33], much due to the different time scales present in the microgrid. Although this hierarchy is commonly used, the definition of the layers given by relevant literature differs slightly. For this thesis, the representation in Figure 2.3 is used. This figure shows that the hierarchical control strategy consists of the following three layers with their designated response times and roles in controlling the microgrid:

1. **Primary control** operates in a decentralized manner and consists of local controllers implemented in the power converter interfaces of the microgrid components. This control layer performs control functions that require a fast response time, such as converter output control, power-sharing, and island detection.

2. **Secondary control** operates at a slower speed than the primary control layer and aims at correcting steady-state deviations by adjusting the voltage and frequency

reference points of the primary control. In addition, this control layer is responsible for synchronization and power exchange with the external grid.

3. **Tertiary control** is the highest control level and controls the long-term behavior of the microgrid. This layer introduces intelligence to the system by optimizing the microgrid operation.
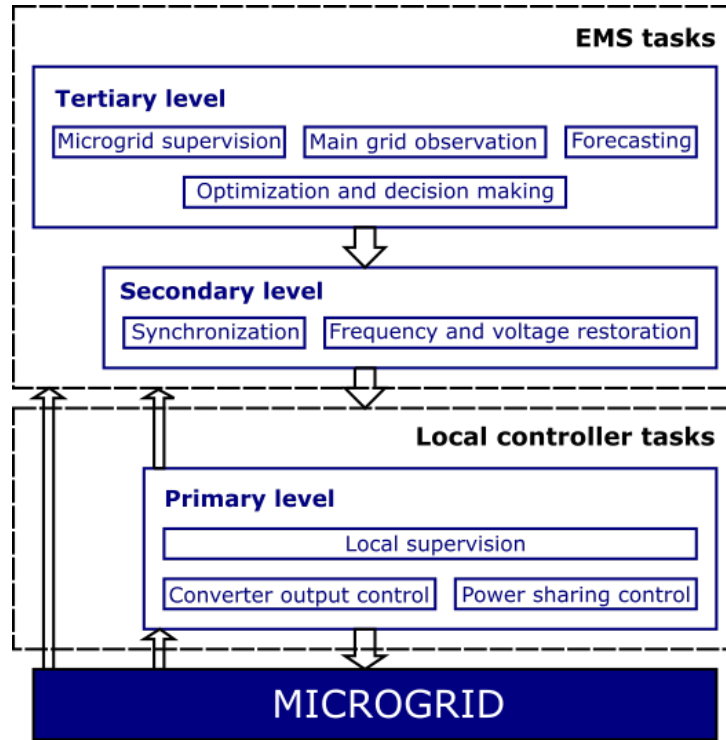
Figure 2.3: Hierarchical control structure for a microgrid. Inspired by Figure 2 in [7].

Based on the given description of microgrid control, the selected focus of this thesis is the energy management system (EMS). The microgrid EMS has been subject to extensive research in recent years, and its definition and functions within the hierarchical control structure vary slightly from work to work. In the control structure presented in Figure 2.3, both the secondary and the tertiary control levels are implemented in the EMS, following the work conducted in [7], [32], and [33]. Other work, such as [26], places the energy management functions in the secondary control level and defines tertiary control only for the grid-connected mode of operation.

2.2   Microgrid Energy Management Systems

A prerequisite for optimal operation of a microgrid is a properly designed energy management system (EMS). An EMS is an inter-disciplinary system that utilizes ICT to

ensure optimal coordination between the microgrid units to supply reliable, sustainable, and high-quality energy in a cost-efficient way [7, 8]. To achieve this, the EMS performs several functions such as data monitoring, data analytics, forecasting, optimization, and real-time control, as illustrated by Figure 2.4 [34].
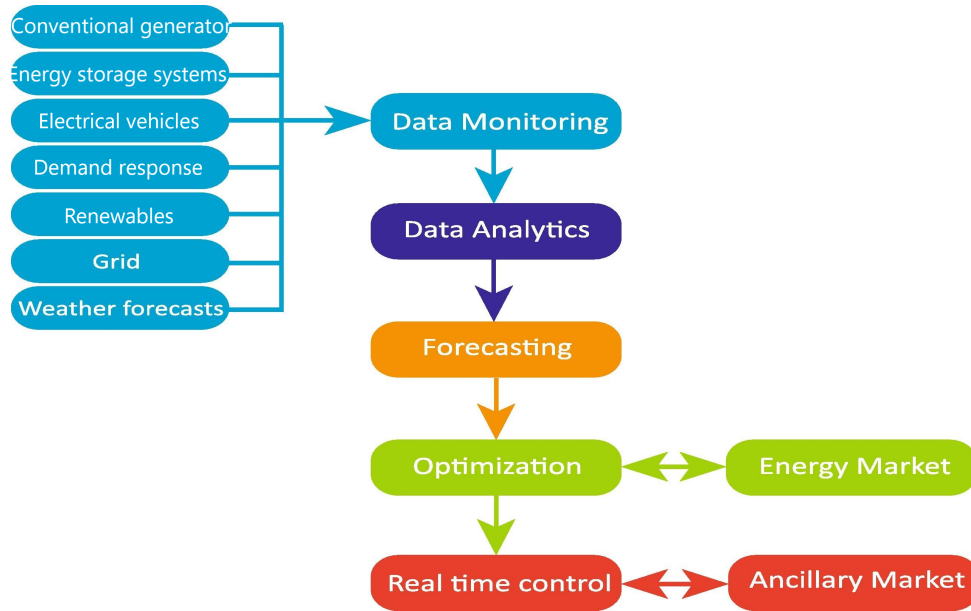
Figure 2.4: Microgrid energy management functions [34].

The microgrid energy management functions help the EMS to optimize operation while satisfying the system constraints. Historical and forecasted data is constantly monitored and analyzed to obtain better insight into the microgrid operation. This insight can be employed to adjust forecasts and optimization models to improve performance. In addition, data analysis can be useful when designing new control policies and developing better forecasting algorithms to predict demand, generation, and electricity prices. Furthermore, the monitored data sets, the insight obtained from the data analytics, and the forecasted data are used to solve decision-making strategies to achieve optimized microgrid operation. Finally, the output of the optimization is used to perform real-time control of the microgrid.

Since the optimization is the brain of the EMS, methods to achieve an optimal microgrid operation will be the focus of the remaining literature review.

2.3   Review of energy management methods

The methods used in microgrid energy management systems range from simple rule-based algorithms to complex multi-parametric optimization techniques. In the following review,

two main groups of methods are considered, namely methods based on heuristics and methods based on the optimization of some criteria. In addition, the set of control approaches known as model predictive control (MPC) will be discussed.

The objective of the review is to find a suitable method for scheduling the battery in the microgrid considered in this thesis, i.e., a grid-connected microgrid comprising a PV system, residential loads, and a battery.

2.3.1   Heuristic methods

Heuristic methods are based on a set of algorithms that use rules to perform the energy management in the microgrid [9]. These methods are characterized for being simple, reliable, and computationally efficient, which has made them popular and widely used in small microgrids.

Hysteresis band control (HBC) is one of the most common heuristic methods [9]. HBC is used in [35] and [36] to control the operation of ESSs to follow a hysteresis band whose limits are defined according to the state of charge (SOC). The operation of HBC is straightforward when only one ESS is considered. The ESS handles the unbalance between generation and demand in the microgrid if the SOC is between its upper and lower limit. If the SOC reaches one of its limits, the ESS is disconnected, and other units or the external grid must be used in its place. This method is fast, simple, and suitable for real-time control. However, considering several ESSs and adding more rules to the problem quickly increase the complexity of the algorithm. Moreover, the solution is sub-optimal in terms of cost minimization.

The first step towards optimization is to apply a fuzzy approach where the rules are determined from a fuzzy logic controller, such as in [37] and [38]. Fuzzy logic controllers can simplify the microgrid management and control when the addition of heuristic rules makes the energy management problem challenging to solve.

Heuristic methods are also commonly used as a reference for comparing the performance of a developed algorithm. An example of this can be found in [39], where the advantages of the proposed dynamic programming algorithm are highlighted by comparing its performance to a simple ruled-based management strategy. Moreover, in [12] a heuristic algorithm is used as one of four energy management strategies to control a grid-connected microgrid. The heuristic algorithm led to cost savings, but not as much as the other strategies.

To conclude, the primary features of heuristic methods are simplicity and computational speed. However, the solution provided by these methods is not optimal. In addition, heuristic methods consider the present time only and are not long-term in nature. Another important issue is that the inclusion of many details and additional features quickly makes the problem too complicated to be solved using traditional heuristic methods. To overcome these issues, optimization-based methods can be considered.

2.3.2   Optimization-based methods

Optimization-based methods are characterized by the technique of solving an optimization problem to design a control input. The microgrid optimization problem is commonly formulated as an objective function to be minimized over a set of inputs restricted by constraints [40]. The solution to this problem provides optimal operating points for the microgrid units along with different time frames.

In the optimization problem formulation, the objective function is the output to be minimized or maximized. One or more objective functions can be included in the microgrid optimization problem, resulting in either a mono-objective or a multi-objective problem [41]. In a mono-objective microgrid optimization problem, the objective function typically corresponds to the operating costs of the microgrid, while in a multi-objective problem, a solution to the technical, economic, and environmental problems is simultaneously presented [8]. Figure 2.5 lists some of the costs commonly included in the microgrid objective function, and it is based on the microgrid EMS literature reviews performed in [34], [41], and [42].
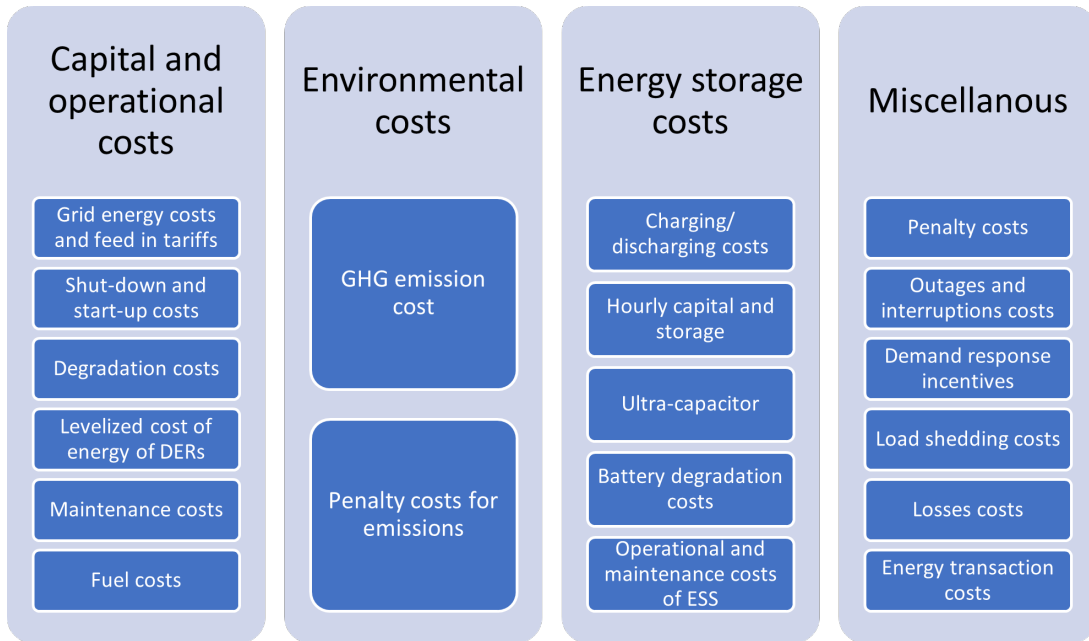
Figure 2.5: Common objective functions in the energy management optimization problem.

Additionally, the microgrid energy management optimization problem is subject to a number of constraints that can be formulated as equality expressions or inequality expressions [40]. These constraints define the microgrid operational framework by reflecting the limits of the microgrid units required for a safe and economic operation. Examples include maximum and minimum limits for charge and discharge of storage devices, power balance constraints, and generation power output limits [34, 42].

The microgrid optimization problem can take different forms depending on the complexity of the system, the objective function, the constraints, and the types of variables involved. Therefore, an appropriate optimization technique must be chosen according to the complexity of the problem. Various techniques have been proposed in the literature to solve the microgrid optimization problem, and the following subsections briefly describe some of these techniques.

2.3.2.1   Linear and nonlinear programming methods

The model of the microgrid must be included in the optimization problem as a constraint. If the model or other constraints are nonlinear, nonlinear programming (NLP) can be used to solve the problem. An NLP formulation is used in [43] to optimize the scheduling of an energy system where nonlinear functions represent the combined cooling and power generation system. An issue with NLP is its inability to handle discrete variables such

as binary variables. This is problematic in a microgrid system where both continuous and discrete-valued dynamics interact. Physical quantities, such as power flows, can be represented using continuous variables, while the discrete features of microgrid units, such as the ON/OFF state of generating units and the charge/discharge state of storage units, can be captured using binary decision variables [12].

Adding binary variables to the NLP problem transforms it into a mixed-integer nonlinear programming (MINLP) problem. An example of this can be found in [44], where binary variables were included to handle the ON/OFF states of the photovoltaic, wind, biomass, gas-turbine, and fuel cell generators in the microgrid. Although MINLP is able to capture the microgrid dynamics well, the resulting nonlinear problem is generally non-convex. Hence, the existence of a global optimal solution cannot be guaranteed, which can lead to less ideal economic effects. Moreover, no exact solution technique exists and the introduction of binary variables seriously increases run time [9, 12]. Solving an MINLP problem can therefore be quite complex and computationally demanding, and the authors in [45] argue that obtaining a global optimal solution to the large-scale non-convex MINLP problem in a sufficient computational time is still an unsolved problem.

An efficient way to mitigate the problems related to MINLP is to use a linear approximation of the objective function and constraints to form a mixed-integer linear programming (MILP) problem. This was done in [46], where the quadratic curve of the fuel costs was approximated using a piece-wise linear function. In [47], the thermal constraints were linearized to enable the formulation of the energy management problem as a MILP problem. Consequently, the problem can be solved using powerful commercially available solvers like GAMS and CPLEX [9], which can provide solutions even for short execution times. Moreover, the linear objective function and constraints of a MILP problem result in a convex feasible region, which guarantees a global optimal solution [9].

A possible issue with the MILP formulation is that it requires the objective function and constraints to be linear. This generally implies simplifications of the problem where the variables can be forced to change their nature to meet the requirements. However, the results of the MILP algorithm proposed in [48] show that the linearizations and approximations produce accurate solutions when compared with a nonlinear three-phase OPF formulation, with an error in the objective function close to 2% and a maximum error in the voltage close to 1%. Moreover, comparisons between MILP and MINLP

optimization approaches were performed in [45] and [49]. These papers showed that the solution time of MILP is faster than that of MINLP due to the relatively simple equations involved in MILP. Hence, MILP is generally more suitable for real-time control where short execution times are essential. In addition, it was shown that in the case where MINLP gave an optimal solution, this was similar to that given by MILP, which confirms that MILP leads to little loss of accuracy compared to MINLP.

2.3.2.2   Dynamic programming

Dynamic programming (DP) can also be used to solve the microgrid optimization problem. DP is a methodology that makes decisions in stages by breaking the optimization problem into simpler sub-problems in a recursive manner [9]. These sub-problems are optimally solved and superimposed to form the optimal solution to the overall problem.

In contrast to heuristic methods that make decisions based on simple rules, DP methods make decisions based on costs. Hence, they can optimize operation based on a cost function. This contrast is highlighted in [39], where both a DP algorithm and a simple rule-based algorithm are used to perform energy management in the microgrid. The rule-based algorithm guarantees operation of the system within its constraints, but it does not optimize the use of solar power. In contrast, the DP algorithm utilizes a cost function to optimize the microgrid operation, which provides around 13% higher economic gain than the rule-based management.

An important advantage of DP is that the performance index and the constraints can be both linear and nonlinear, convex and concave, differential and not. DP is used in [50] to solve a nonlinear optimization problem to maximize the daily economic benefit of a microgrid. The objective function is composed of the energy cost and the battery degradation cost, and it is expressed as a nonlinear function of the battery SOC. Moreover, the power injected into the grid is curtailed through a nonlinear constraint to help mitigate over-voltage problems caused by reverse power flows.

The biggest limitation of using DP is the "curse of dimensionality". Dividing the problem into sub-problems and storing intermediate results consume memory. This is highlighted in [51], where DP is used to co-optimize the operation of a battery for arbitrage and frequency regulation. The results were reasonable. However, a disk space of 60 Terabytes was required to store the values for all possible states over a one-day simulation period.

The high number of recursive functions also makes the implementation complex, and no general formulation of DP is available except for in simple cases [9]. These drawbacks make DP challenging to implement in a real-time controller.

2.3.2.3 Meta-heuristic approaches

Meta-heuristics are a family of methodologies where heuristic techniques are combined to approximate the best solution to an optimization problem using biological evolution, genetic algorithms, and statistical mechanisms [8]. These methods can quite quickly obtain highly accurate approximated optimal solutions to the microgrid optimization problem for various cost functions and constraints. They can especially be an option in the case of non-convex problems or high numbers of variables and constraints [7]. In these cases, the complexity of the associated control problem increases, making it difficult to find a solution using classical methods. An example can be found in [52], where meta-heuristic methods were applied to solve a MINLP problem to plan natural gas and electricity distribution networks optimally.

A wide range of meta-heuristic approaches has been utilized in microgrid control. A review of some of these approaches can be found in [53], where the most representative ones are: tabu search (TS), genetic algorithms (GA), particle swarm optimization (PSO), and artificial neutral networks (ANN). PSO was applied in [54] to find real-time optimal energy management solutions for an islanded hybrid microgrid. The results demonstrate that PSO can be used to solve an extensive solution space while incorporating multiple objectives such as minimizing the cost of generated electricity, maximizing micro-turbine operational efficiency, and reducing environmental emissions.

Although complex microgrid energy management problems can be solved using meta-heuristics instead of classical optimization methods, these methods also have some disadvantages. These disadvantages include not guaranteeing the global optimality of a solution, long calculation times, complex formulation, and obtaining different results at each run due to the inherently uncertain behavior [7, 8, 53].

2.3.3 Model Predictive Control

The aforementioned methods are often used offline to perform day-ahead scheduling of the microgrid units. This implies that the optimal power dispatch for each sample time

of the next day is completed in one calculation based on forecasts of generation and demand. The schedule produced by this static and open-loop method is highly dependent on the accuracy of the forecasts and may not remain optimal in real-time scenarios. One way to address this problem is to dynamically adjust the schedule according to real-time conditions, namely to solve the optimization problem within a model predictive framework.

The term model predictive control (MPC) does not refer to a specific control strategy but rather to a wide range of control methods that explicitly use a system model to obtain the optimal control signals by minimizing an objective function. The model predictive control family is essentially defined by the three following characteristics [9, 10, 11]:

1. Explicit use of a model to predict future system outputs.

2. Calculation of a control sequence through the minimization of an objective function over a finite time horizon.

3. Use of the receding horizon principle, stating that only the first element in the calculated control sequence is applied to the system at each time step. The horizon is displaced towards the future at the next time step, and a new optimal control sequence is calculated using new predicted values. Hence, a feedback mechanism is obtained, resulting in closed-loop control.

The control decision imposed by the MPC may not result in the optimal operation for the microgrid at the current time instant, but it results in the optimal operation in the forecasted time horizon based on the forecasted system behavior.

The methodology of all the controllers belonging to the MPC family is characterized by the following strategy [10], as illustrated in Figure 2.6.
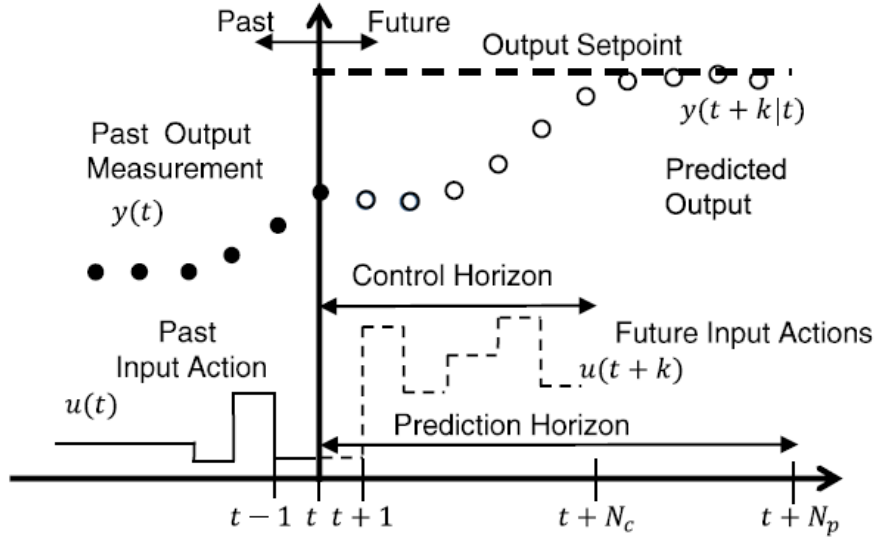
Figure 2.6: MPC strategy [9].

1. The dynamic system model predicts the future outputs for the prediction horizon, N_p, at each sample time t. These predicted outputs $y(t + k|t)$ for $k = 1...N_p$ depend on the known values of past inputs and outputs, on the current state, and on the future control signals $u(t + k|t)$, $k = 0, ..., N_p - 1$.

2. The sequence of future control signals, $u(t + k|t)$, is computed by optimizing a determined criterion.

3. Although a complete sequence of N_p future control signals is computed, only the first element, $u(t|t)$, is sent to the process. The rest of the elements are discarded because at the next sample time, a new output $y(t + 1)$ is already known. Step 1 is repeated with this new value and all the sequences are brought up to date. Further, the control sequence $u(t+1|t+1)$ is calculated (which may be different from $u(t+1|t)$ due to the new information available) using a receding horizon.

2.3.3.1   Model Predictive Control in microgrids

The above description of MPC includes several features that make it a good candidate for microgrid control. These features can be summarized as follows:

- It provides a receding prediction horizon with a feedback mechanism, which helps the system to react more robustly to uncertainty in generation and demand.

- It enables an objective function to be formulated and optimized by the controllers.

- It can handle the constraints associated with power systems, such as storage capacity, ramp rates, and minimum up and down times of generators.

- It considers the future behavior of the system in the optimization, which is useful when planning and allocating resources of microgrids that integrate forecasts of renewable generation and demand.

- It handles multi-variable systems well, which is useful for managing the operation of multiple microgrid units in a coordinated way.

- It is suitable for making real-time control decisions.

Figure 2.7 depicts a general diagram of a grid-level MPC. The different MPC algorithms differ by the model used to represent the system, the objective function to be minimized, and the solving algorithm utilized in the minimization [55].
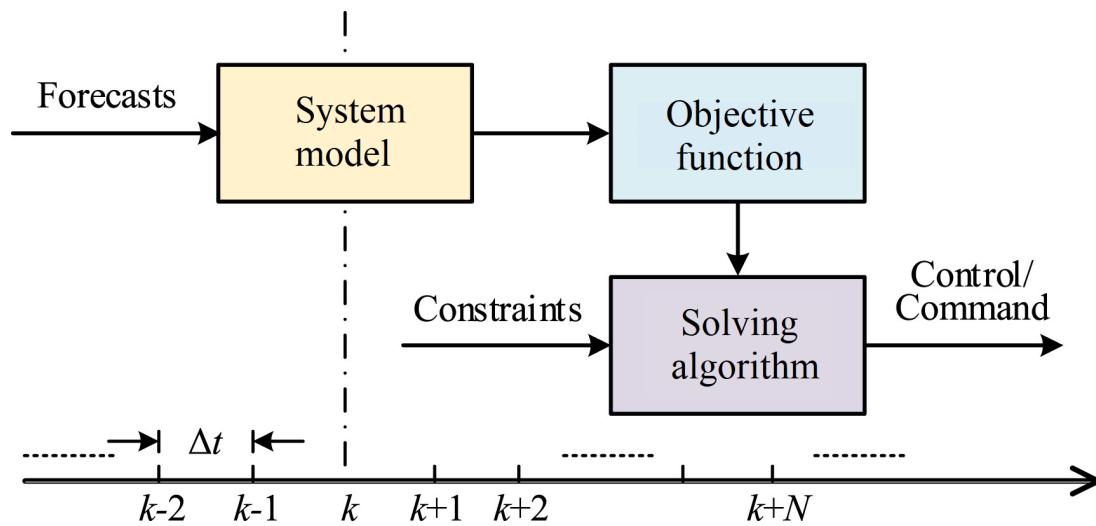
Figure 2.7: General diagram of a grid-level MPC. Adapted from [55].

The system model is built upon the system states with possible forecasts, where an expression for future predictions is formulated based on current and past states. *The objective function* should reflect the concerns of the control objectives, and can result in either a mono-objective or a multi-objective problem including several costs, as depicted in Figure 2.5. For solving the optimization problem, an MPC algorithm can incorporate any *solving algorithm* (LP, MINLP, MILP, meta-heuristics, etc.) depending on the type of model used and the cost function employed [9, 55].

Although MPC has several features suitable for microgrid operation and control, it also comes with some drawbacks. The use of a receding horizon requires high computational

efforts since an optimization problem is solved at each sample time [10]. This especially holds for the constrained case or when using long prediction horizons. The calculation time depends on the model formulation and the optimization method used. Moreover, an MPC controller also has a more complex derivation than classical controllers. Another important issue is the need for an appropriate system model [11].

2.3.3.2 Previous studies addressing MPC in microgrids

The surveys in [11] and [55] show that MPC is emerging as a hot topic in the world of microgrid control. This is also reflected in the frequent publications of research on the topic. A short review of previous studies applying MPC in microgrid economic operation optimization is performed in the following paragraphs to gain insight into the main gaps and challenges.

A generic MPC is proposed in [56] to manage thermal storage tanks in buildings. The algorithm considers short-term load forecasts and a dynamic model of the storage tanks. For computational efficiency, the optimization problem is approximated by a meta-heuristic algorithm that cannot guarantee an optimal solution but converges close to it. The computation time is also kept low by linearizing the system model to avoid computationally demanding nonlinearities.

The work conducted in [57] applies MPC to coordinate the operation of a wind/solar subsystem and a battery to provide enough energy to a water desalination system to meet the desired water production demand. The resulting optimization problem is nonlinear and non-convex, yielding sub-optimal solutions. However, the algorithm still performs better than the reference control. Moreover, a large sample step of one hour is used, not allowing the algorithm to quickly adjust the schedule according to the variable real-time water demand and PV and wind conditions.

Another approach to microgrid MPC is taken in [58], where MPC is employed in the lower layer of a two-layer structure to control the battery to ensure that the microgrid accurately follows the power references given by the upper layer. This allows the lower layer to use a very short sample time of one minute and, consequently, effectively observe and respond to small changes in demand and generation. This is an improvement compared to much of the research published in the context of microgrid EMS, which tends to use sampling times ranging from 15 minutes to 2 hours. The results obtained from experimental studies show

that the algorithm is capable of real-time control and that it succeeded in simultaneously reducing the daily costs of the microgrid and increasing the self-consumption of renewable energy sources.

The work in [12] solves the optimization problem for a grid-connected microgrid using a MILP technique. It incorporates a feedback control law by implementing the MILP algorithm in a model predictive framework. This MILP-MPC approach is tested both through simulations and experiments, with a 1 hour and 15 minute sample time, respectively. The results show that the operational costs are improved compared to when the MILP problem is solved open-loop due to uncertainties in generation and demand. However, the proposed algorithm does not consider battery degradation and assumes a constant power electronic efficiency. Therefore, the results are less realistic and may lead to battery references that validate the constraints in the long run.

A MILP-MPC energy management methodology was also developed in [13]. The developed algorithm considers multiple objectives, including minimizing energy costs, microgrid power profile shaping for utilities, and battery usage costs. The battery usage costs avoid unnecessary charging-discharging actions that would reduce battery life. In addition, battery management is enhanced through the concept of incremental red-zone power rates. However, the simulation process considers a relatively long sample time of 30 minutes over the three-day simulation period. Furthermore, the algorithm does not update the battery model by calculating the actual battery degradation, i.e., the battery capacity is assumed constant in the optimization.

Battery degradation is considered in the MILP-MPC approach proposed in [14]. This algorithm updates the battery model daily based on real measurements, enabled by the closed-loop MPC approach. Once the actual usable battery capacity is known, the battery is utilized to minimize the microgrid operation costs while avoiding redundant charging-discharging actions since those actions significantly reduce battery lifetime. However, the case study considers a short simulation period of three days where the results are not evaluated in terms of battery degradation. In addition, the algorithm completely neglects power electronic losses.

2.3.4 Discussion of methods

This section will shortly summarise and discuss the energy management methodologies considered in the above literature review before selecting a method for the work conducted in this thesis. Figure 2.8 presents an overview of the considered methods.
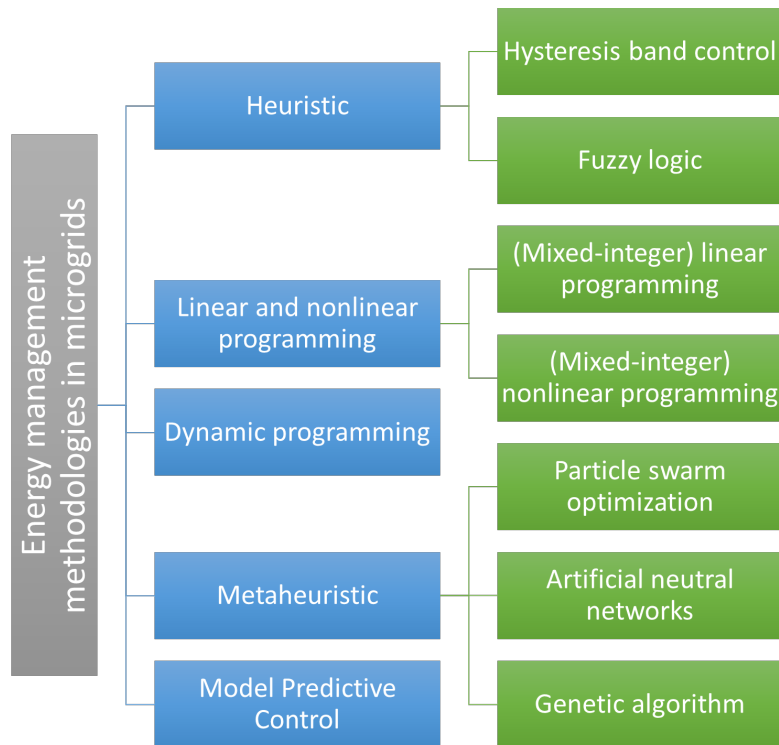
Figure 2.8: Energy management methodologies in microgrids.

The literature review argued that optimization-based methods improve the overall performance when compared to heuristic methods. This claim will be further investigated in the thesis by developing both a simple rule-based method and a more advanced optimization-based method to thoroughly investigate their differences.

Although optimization-based methods generally result in reduced costs and improved microgrid operation, solving these methods offline often result in constraint violations and poor performance in real-time scenarios due to the inability to handle forecast errors. Therefore, the optimization problem should be solved within a model predictive framework. MPC can incorporate any of the discussed optimization algorithms, i.e., linear and nonlinear programming, dynamic programming, and meta-heuristics. Selecting a suitable method can be challenging. To aid in this choice, Table 2.1 briefly presents the main advantages and disadvantages of each of the reviewed optimization-based energy management methodologies.

Table 2.1: Comparative analysis of optimization-based energy management methodologies in microgrids.

Method	Advantages	Disadvantages
MILP	• Convex feasible region, i.e., can guarantee a global optimal solution. • Powerful solvers exist, resulting in reduced computational effort.	• Limited capabilities for applications with not differentiable and/or continuous objective functions. • Linearizations might lead to loss of accuracy.
MINLP	• Captures the nonlinear microgrid dynamics well.	• Complex and computationally demanding to solve. • Non-convex feasible region, i.e., cannot guarantee a global optimal solution.
DP	• Can solve more complex problems that can be discretized and sequenced. • The performance index and the constraints can hold all the natures (linear or nonlinear, convex or concave, differential or not).	• The curse of dimensionality. • Complex implementation. • Long calculation times.
Meta-heuristics	• Can obtain approximated optimal solutions to complex, nonlinear problems with many variables and constraints.	• Non-convergence to global optimum. • Complex formulation. • Different results at each run. • Long calculation times.

DP has successfully been combined with MPC and used to control microgrids with any type of constraints (linear/nonlinear, convex/concave, differential) in previous studies. However, the implementation is complex, and implementing it within a model predictive framework is increasingly difficult. In addition, DP suffers from the "curse of dimensionality," which will be even worse when implemented with MPC, and it is thus less suitable for real-time control. Meta-heuristic methods are also very powerful, but their disadvantages make them less ideal for MPC. These disadvantages include non-convergence to global optimum, high computational effort, intractable adjustment of parameters, and complex formulation.

Linear and nonlinear programming methods are considered in multiple successful MPC examples in current literature. Section 2.3.2.1 argued that there are several advantages to keeping the problem linear. Therefore, MILP is a good option due to less computational effort, compatibility with available solvers, and a guaranteed optimal solution without noticeable loss of accuracy compared to MINLP.

Incorporating MILP in an MPC framework can potentially reduce some of the disadvantages related to the linearization and simplifications of the microgrid optimization problem required to utilize a MILP technique. Despite this, much of the MILP-MPC work considered in the above literature review comprised of linearities and were solved with implicit MPC to obtain the control objectives without updating the system model. The following bullet points sum up some of the deficiencies observed in current literature:

- Concerning the battery modeling, the battery capacity is assumed constant in most work, while in reality it decreases as the battery is used. This is a valid assumption for a short period of time, but it will affect the results in the long run. Considering the variable battery capacity in the problem formulation involves high nonlinearities.

- Although much work aims at limiting battery degradation, few evaluate the performance of the proposed algorithm in terms of battery aging. The algorithms are often investigated and verified in a microgrid simulation platform where the incorporated battery model neglects degradation, making it difficult to properly evaluate how the schedule provided by the EMS will affect the battery.

- Simulations are typically performed for one day, which does not show the long-term effects.

- Sampling times ranging from 15 minutes to 2 hours are commonly used. A shorter sampling time is preferable for real-time control because it allows the EMS to observe and respond to small changes in the load, generation, and electricity price throughout the day.

- The power converter efficiency is often neglected or assumed constant while in reality it depends on the battery power. The result is less realistic results in terms of how much power the battery can deliver or absorb.

Based on the above discussion, this thesis aims at developing a control approach that combines MPC with MILP to effectively account for uncertainties and to capture some of the nonlinear dynamics of the system by updating the system model every sample time.

2.4   Summary

This chapter has provided a microgrid definition formulated by CIGRÉ and showed that microgrids are essential in realizing the smart, green, and efficient power system of the

future. However, although the introduction of microgrids results in multiple benefits, there are several challenges to address before their full potential can be utilized. Motivated by the microgrid challenges related to optimal control and management, energy management systems were selected as the focus of this thesis. Further, this chapter conducted an extensive review of energy management methods to choose an appropriate method for solving the energy management optimization problem. In this review, two main groups of methods were considered, namely heuristic and optimization-based, in addition to the set of control approaches known as model predictive control. Based on this review, the MILP-MPC approach was selected to control the microgrid considered in this thesis.

3   The microgrid model

This chapter presents a microgrid simulation platform in MATLAB/Simulink suitable for implementing and testing an energy management system. It also gives the resources used to develop this model. Section 3.1 defines the goal of the simulation platform and selects a suitable solving system method. Section 3.2 describes the model and its components, comprising a complete battery model including degradation and power-dependent converter efficiencies, a PV model, load models, and the modeling of the utility point-of-connection.

3.1   Solving system method

The microgrid model is developed using MATLAB/Simulink. Simulink has libraries with electrical components and can build and run simulations of electrical systems, control systems, and more. As a first step in the model development, an appropriate electrical circuit solving method must be chosen. The Simulink powergui block allows the choice between the three following solving system approaches [59, 60]:

- Continuous, which uses a variable-step solver from Simulink.

- Discretization of the electrical system for a solution at fixed time steps.

- Continuous or discrete phasor solution.

The aim of developing a microgrid simulation platform in this thesis is to enable the implementation and testing of an energy management system belonging to the higher control level of the microgrid operation. The higher control level works with the long-term behavior of the system, and it is thus little dependent on the fast dynamics' transient behavior. This means that it is no need to calculate the system state in detail every split second. Moreover, since all AC devices in the microgrid work with a frequency of 50 Hz and the simulation speed is essential, the phasor solution is a suitable solving system approach for the microgrid model developed in this thesis.

3.2   Model description

This section will describe the microgrid simulation model and its components. The developed microgrid model is depicted in Figure 3.1 and is of a three-phase AC grid-connected microgrid, which implies that the main grid performs the voltage and frequency

regulation. A variable time-step is selected to simulate the model, which means that the step size varies during the simulation. The step size is reduced to increase accuracy when model states are changing rapidly, and the step size is increased to avoid taking unnecessary steps when model states are changing slowly. Although the model is simple, it contains key elements of a microgrid and is still representative of a physical application. The simplicity of the model is also suitable for the purpose of testing an EMS.

The microgrid simulation model consists of a photovoltaic (PV) panel, a static load, a variable load, and a battery. The PV panel, the static load, and the variable load are modeled using the microgrid library developed by application engineer at the MathWorks, Johnathan LeSage [15]. The blocks in this microgrid library are built using components from Simscape Electrical, which is a MATLAB/Simulink library developed to provide components for modeling and simulating electronic, mechatronic, and electrical power systems [61]. The battery is built using the SimSES software tool developed by Maik Naumann and Nam Truong at the Technical University of Munich [16]. Further details on the microgrid components are provided in the following subsections.
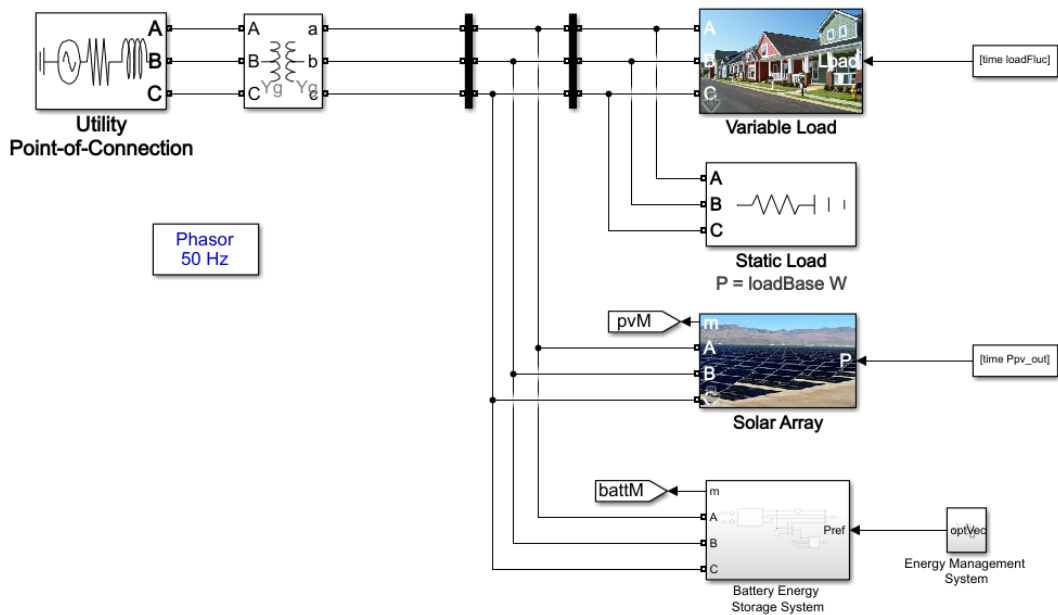
Figure 3.1: Simulink model of a three-phase AC grid-connected microgrid.

3.2.1   Battery model

Optimal usage of the battery has an important role in cost optimization, and harsh usage will decrease its lifetime significantly. Aging effects directly affect the usable battery capacity. Therefore, the complete modeling of a battery should include an aging model

to properly evaluate how the schedule provided by the EMS will affect the battery. The battery model included in the aforementioned microgrid library neglects degradation and is thus not applicable for meeting the model requirements of this thesis.

Therefore, the open-source software tool SimSES (software for techno-economic simulation of stationary energy storage systems) is used to simulate the battery in this thesis [16]. SimSES applies a modular, flexible, and abstract approach to modeling energy storage systems, allowing the user to select the system structure and technology. Furthermore, it incorporates several aging models, which enables the estimation of the energy storage degradation. The tool has been used to analyze various battery fields of application in several publications [19, 20, 21, 22], where it has proven to achieve results with sufficient accuracy for techno-economic considerations within an appropriate execution time.

A comprehensive documentation of all scripts and functions is included in the source code [62]. Moreover, an extensive verification of the SimSES model was conducted by Maik Naumann in his doctoral dissertation [18]. Hence, only the basics about SimSES that are relevant for the work conducted in this thesis are presented in the following, and no verification is performed. All SimSES scripts implemented in this work are listed in Appendix B together with the corresponding modifications and settings used.

3.2.1.1   SimSES implementation in Simulink

The SimSES software uses object-oriented programming to create an encapsulated model of an energy storage system with the relevant technical parameters and functions. To enable compatibility to Simulink, SimSES uses MATLAB's System Object structure, which is designed specifically for implementing and simulating dynamic systems with inputs that change over time. The SimSES System Object is included in the Simulink microgrid model by using a MATLAB System block, as depicted in Figure 3.2. More details on how System Objects can be included in the Simulink environment can be found in [63].
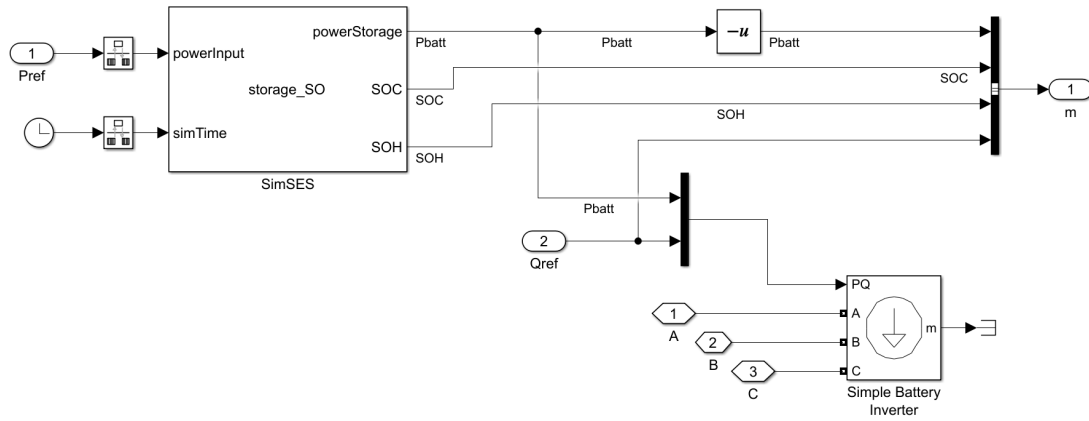
Figure 3.2: Integration of the SimSES battery model in Simulink.

The SimSES System Object block is initialized by the initialization script attached in Appendix A. This script uses the SimSES functions and scripts listed in Appendix B to create a system object representing a battery with the relevant technical parameters and functions. Furthermore, the system object requires two inputs, namely the sample time and a reference power signal provided by the EMS and its control algorithms. The inputs are used to compute updated battery states and outputs at each sample time. Sections 3.2.1.2 and 3.2.1.3 describe the outputs and the equations used to compute them.

A challenge with the implementation is that SimSES simulates batteries using a fixed sample time, while the rest of the microgrid model uses a variable time step. Therefore, a rate transition block is required at the inputs of the SimSES block to handle the data transfer between the two dynamics. Moreover, a Three-Phase Dynamic Load block from the component library in Simscape Electrical is included to convert the output power of the SimSES block into the three-phase phasor signal required by the microgrid model.

3.2.1.2   SimSES battery energy storage system model

SimSES enables the use of either an equivalent circuit battery model or a power flow battery model. Since the aim of the model is to work with the high-level optimization and long-term behavior of the microgrid, the power flow battery model will be used in this thesis. Several battery types can be selected for calculation in the model, and the parameters of the power flow model vary depending on the type selected. This thesis considers a lithium-ion battery because this is the battery type used in Skagerak Energilab.
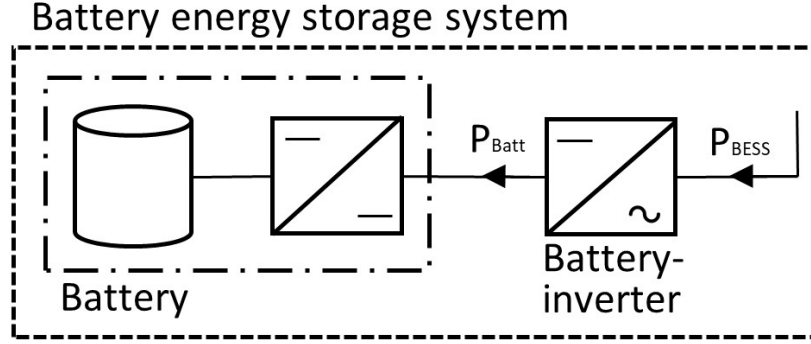
Figure 3.3: Illustration of the coupling-topology of the AC connected battery energy storage system. Inspired by Figure 1 in [22].

Figure 3.3 shows the coupling-topology of the AC connected battery energy storage system (BESS) considered in SimSES. The BESS is controlled by a reference power signal provided by the EMS and its control algorithms. This reference signal is compared to the power limits of the inverter, $P_{rated,inv}$, and the battery, $P_{rated,batt}$, as follows:

$$|P_{batt}| \leqslant P_{rated,batt} \tag{3.1}$$

$$|P_{BESS}| \leqslant P_{rated,inv} \tag{3.2}$$

Consequently, the power output from the BESS is limited to the nominal power of the inverter. After checking the power limits, the battery model computes maximum power based on the operation voltage range of the cell, the state of charge (SOC), and the state of health (SOH). Finally, the system response to the power reference is determined.

The modeled system equations for the SimSES AC-coupled BESS are given in the following paragraphs. More details can be found in [16] and its MATLAB source code [62].

State of charge

The state of charge (SOC) is the amount of charge left in the battery and it is directly related to the available energy. It is not possible to measure the SOC and it thus has to be estimated. In the SimSES model the SOC is estimated by the following equation:

$$SOC(k) = \frac{E_{batt}(k)}{E_{cap}(k)} \tag{3.3}$$

where $SOC(k)$ is the state of charge at sample step "k", $E_{batt}(k)$ is the energy stored in the battery at sample step "k", and $E_{cap}(k)$ is the total capacity of the battery at sample step "k". E_{cap} does not remain constant, but continuously decreases over time due to aging effects. The degradation of E_{cap} is estimated by the degradation model described in Section 3.2.1.3.

The energy stored in the battery is calculated by (3.4) depending on if the battery is charging or discharging.

$$E_{batt}(k) = \begin{cases} E_{batt}(k-1) + \sqrt{\eta_{batt}} \cdot P_{batt}(k) \cdot \Delta t - E_{SD}, & P_{batt}(k) \geqslant 0 \\ E_{batt}(k-1) + \frac{P_{batt}(k) \cdot \Delta t}{\sqrt{\eta_{batt}}} - E_{SD} & P_{batt}(k) < 0 \end{cases} \tag{3.4}$$

where $E_{batt}(k)$ and $E_{batt}(k-1)$ are the energies stored in the battery at sample steps "k" and "k-1" respectively, η_{batt} is the battery round-trip efficiency which is assumed equal for charging and discharging, Δt is the sample time, $P_{batt}(k)$ is the power flow in and out of the battery at sample step "k" (positive for charging, negative for discharging), and E_{SD} is the self-discharge of the battery.

The SOC is kept within its boundaries by the charge limits:

$$SOC_{min} \leqslant SOC \leqslant SOC_{max} \tag{3.5}$$

Power electronics

The relationship between the input power of the battery, P_{batt}, and the input power of the inverter, P_{BESS}, is given by:

$$P_{batt}(k) = \begin{cases} \eta_{inv} \cdot P_{BESS}(k), & P_{BESS}(k) \geqslant 0 \\ \frac{1}{\eta_{inv}} \cdot P_{BESS}(k), & P_{BESS}(k) < 0 \end{cases} \tag{3.6}$$

where η_{inv} is the inverter efficiency.

The battery round-trip efficiency given by the manufacturers does not include the battery inverter efficiency. Therefore, SimSES includes power electronics efficiency models to

represent the power loss in the inverter. These models include both a fixed efficiency model and functions based on experimental results. In addition, SimSES offers the possibility to implement new efficiency functions based on own investigations.

In this thesis, the inverter efficiency is modeled with the power dependent efficiency curve shown in Figure 3.4, which is described by (3.7) [64].

$$\eta_{inv} = \frac{p}{p + p_0 + k \cdot p^2} \text{ where } p = \frac{|P_{out}|}{P_{rated,inv}} \tag{3.7}$$

with k = 0.0345 and p_0 = 0.0072 for a high-efficiency inverter.

According to the efficiency curve, the inverter efficiency remains above 90% for an output power of about 10% to 100% of the rated inverter power. The maximum efficiency equals $\eta_{inv} = 96.9\%$ for an output power of $0.45 \cdot P_{rated,inv}$.
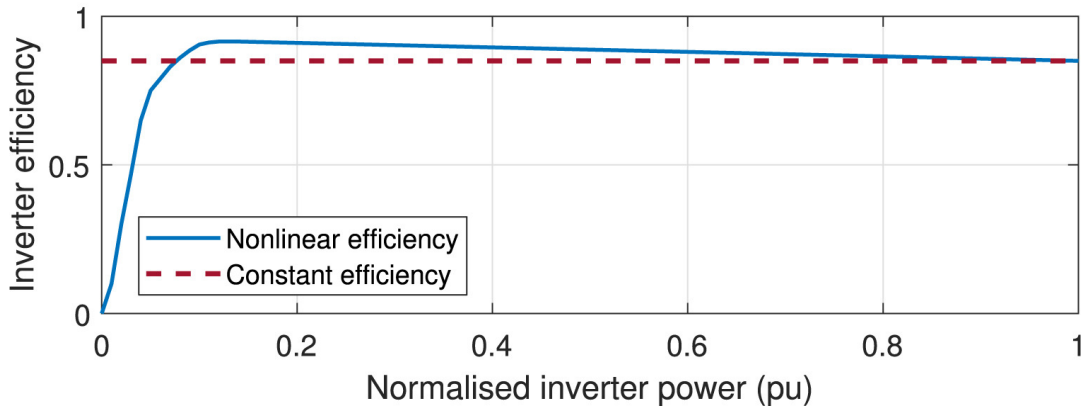
Figure 3.4: Generic inverter efficiency curve [64].

3.2.1.3 SimSES battery degradation model

The structure of the SimSES battery degradation model is depicted in Figure 3.5. The model differentiates between cyclical and calendrical aging processes for battery degradation. Calendrical aging occurs when the battery is idle and leads to constant capacity fade over time, while cyclical aging occurs during load periods where each cycle contributes to a gradual reduction of the storage capacity.
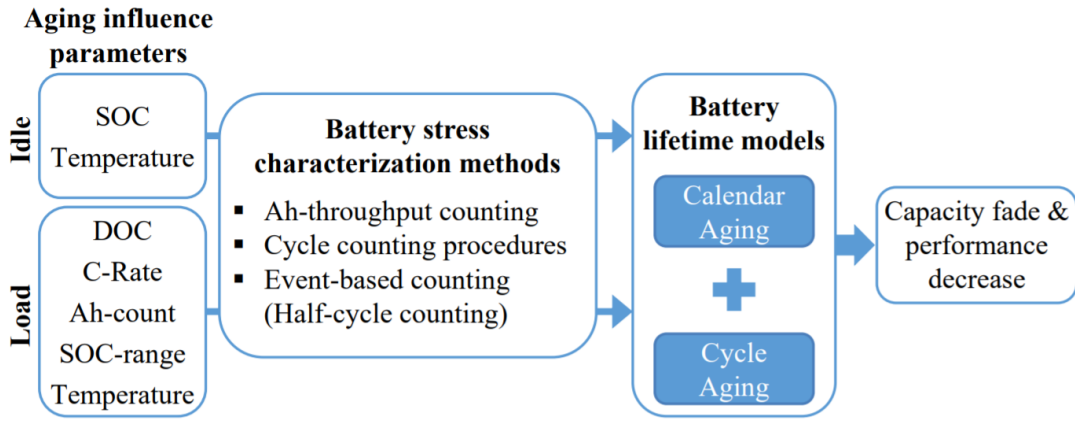
Figure 3.5: SimSES battery degradation model [16].

The inputs to the battery degradation model are the idle and load aging influence parameters. The idle parameters include the SOC and the temperature and are used to estimate the calendrical aging. The load parameters include the particular properties of a cycle, such as temperature, cycle throughput (Ah-count), relative power (C-rate), cycle SOC-range, and depth of cycle (DOC), and are used to estimate cyclical aging.

The aging influence parameters are inputs to the half-cycle counting battery stress characterization method. This cycle-counting algorithm detects half-cycles by counting zero-crossings of the battery power. Every time a zero-crossing occurs, the end of a half-cycle is declared. Further, the algorithm obtains the depth of cycle (DOC) by computing the difference in SOC from the beginning to the end of the detected cycle. The outputs of the half-cycle counting method are used to estimate the battery calendrical and cyclical degradation, given by (3.8) and (3.9), respectively.

$$\Delta C_{cal} = \frac{0.2 \cdot E_{rated}}{t_{cal}} \tag{3.8}$$

where ΔC_{cal} is capacity degradation due to calendrical aging, and t_{cal} is the time period until 20% of the battery capacity is diminished just by calendrical aging.

$$\Delta C_{cyc} = \frac{0.2 \cdot E_{rated}}{k_{cyc}(DOC) \cdot DOC} \tag{3.9}$$

where ΔC_{cyc} is capacity degradation due to cyclical aging, DOC is the depth of cycle, and k_{cyc} is the amount of equivalent full cycles until the battery degrades by 20% of its rated capacity.

The effects of the calendrical and cyclical aging are superimposed in the battery lifetime model to estimate the overall battery capacity degradation and performance decrease. Within this framework, the state of health (SOH) is an important indicator of battery degradation. SOH is defined as the capacity fade over time related to the nominal battery capacity, where SOH = 100 % represents a new battery and SOH = 0 % represents a battery with no capacity left. SOH = 80 % is commonly assumed as the end-of-life criterion [65]. A common expression used to estimate the SOH is (3.10).

$$SOH = \frac{E_{cap}(k)}{E_{rated}} \tag{3.10}$$

where E_{cap} is the total remaining battery capacity (which declines with time), and E_{rated} is the rated battery capacity.

3.2.2   PV model

The PV system is modeled using the Three-Phase Dynamic Load Block from the component library in Simscape Electrical, as shown in Figure 3.6. The input to this block is PV power profiles and not irradiance profiles. To ensure that the load block outputs source currents and not load currents, the PV power is inverted when injected into the block.
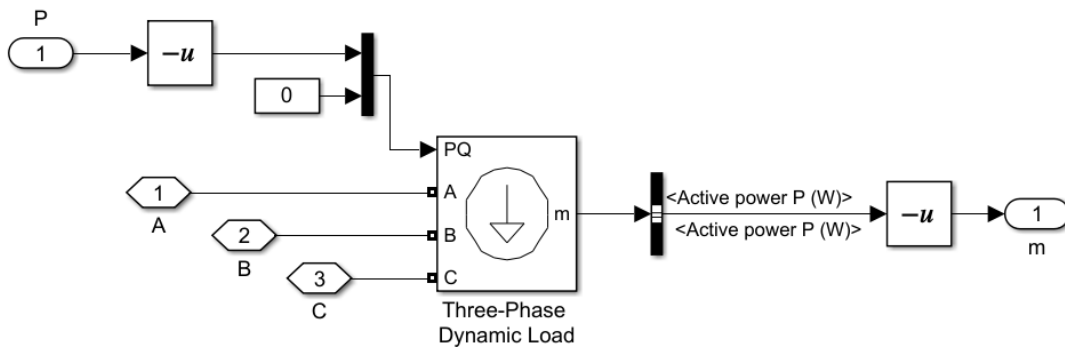
Figure 3.6: Simulink model of the simplified PV system.

3.2.3   Load models

The microgrid developed in this work consists of two loads: one variable load and one static load, where both are non-controllable. Non-controllable loads cannot reduce their energy consumption, and their power demand should always be met. The static load does not change with time and represents the base load of the system, while the variable load

represents the residential homes. The total load demand, P_{load}, at each time interval "t" can be expressed as the sum of the variable load, P_{lv}, and the constant load, P_{lc}:

$$P_{load}(t) = P_{lv}(t) + P_{lc} \tag{3.11}$$

The Three-Phase Series RLC Load block from the component library in Simscape Electrical is used to model the static load. The variable load is modeled using the Three-Phase Dynamic Load block. The active and reactive power of this block can be controlled via external control signals, which is useful when implementing demand-side management strategies. Since demand-side management is outside the scope of this project, the variable load is controlled by a dynamic load control to account for the case of a non-unity power factor, as shown in Figure 3.7. Then the dynamic load control ensures that the active power input is scaled to generate the desired load.
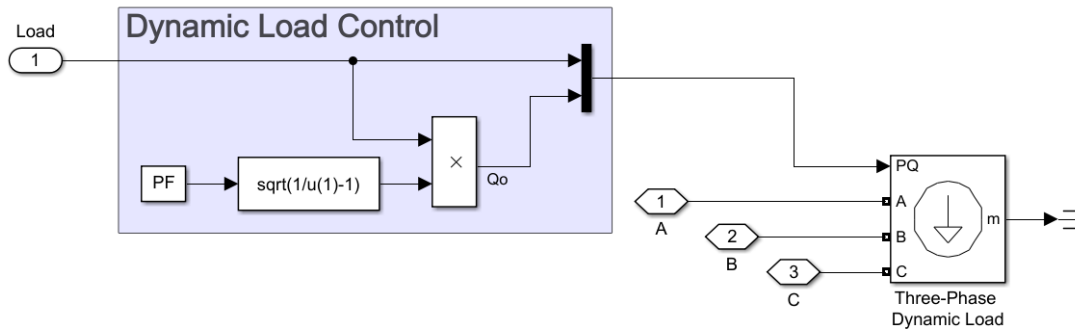
Figure 3.7: Simulink model of a variable load with dynamic load control.

3.2.4 Utility point-of-connection

The microgrid connection to the main grid is modeled as a three-phase ideal voltage source with a root-mean-squared voltage of 20 kV and a frequency of 50 Hz. A 20/0.4 kV step-down transformer is set after this block to reduce the voltage to a value suitable for microgrid operation. The microgrid in this work is designed to operate grid-connected where the main grid helps maintain the voltage, frequency, and power balance. Future work can adapt the model to also work islanded by adding a switch between the utility point-of-connection and the microgrid and adapt the battery model to help the microgrid maintain voltage and frequency when islanded.

3.3   Summary

This chapter has presented a microgrid simulation platform comprising a PV panel, a variable load, a static load, and a battery suitable for implementing and testing an energy management system. The simulation platform is developed in MATLAB/Simulink and has the following key characteristics:

- The model is simulated using a variable-step phasor solving method to enhance simulation speed and to include a sufficient level of detail.

- The model is of a three-phase grid-connected microgrid which implies that the voltage and frequency regulation is performed by the main grid. Future work can adapt the model to also work islanded.

- The PV panel, the variable load, and the static load are modeled using the microgrid library developed by application engineer at the MathWorks, Johnathan LeSage [15]. Although the loads are non-controllable, it is possible to control the variable load via external control signals which enables the implementation of a demand-side management strategy in future work.

- The battery model is built using the SimSES software tool developed by Maik Naumann and Nam Truong at the Technical University of Munich [16]. The model includes a lithium-ion battery model, a variable power converter efficiency model, and a degradation model.

4 The Energy Management System

The purpose of this chapter is to develop energy management strategies to determine the charging and discharging power set-points of the battery in the microgrid model developed in Chapter 3. Section 4.1 develops a heuristic algorithm based on simple rules. This algorithm will work as a reference for comparison. Section 4.2 develops a control approach that combines model predictive control with mixed-integer linear programming. The overall optimization problem is formulated by defining constraints and a multi-objective cost function. Finally, this chapter describes the implementation of the proposed algorithm in the microgrid simulation platform developed in Chapter 3.

4.1 Heuristic method

As a first step, a heuristic method is developed to perform the energy management in the microgrid and to work as a reference for evaluating the performance of the optimization algorithm proposed in Section 4.2. The heuristic algorithm determines the battery power reference based on a series of rules designed to ensure that the system always operates within the defined constraints. This simple heuristic method only considers the present time, and it is not long-term in nature over a specified prediction horizon. Moreover, no cost function is considered in the algorithm.

Figure 4.1 shows a flowchart of the heuristic algorithm, where the sign convention of the power flows are given by Figure 4.2. The heuristic algorithm receives energy and power information from the EMS. This information is used to compute the net power. First, the sign of the net power is checked. If the net power is negative, there is not enough PV power to supply the loads, and the battery should discharge to meet the demand. However, the battery cannot discharge if it violates either the minimum SOC constraint or the maximum discharge power rate. Then, the load demand must be met by power from the main grid. In the opposite case, if the net power is positive, there is an excess of PV power available that can be used to charge the battery. However, this is only possible if the SOC has not already reached its maximum limit. In addition, the battery cannot charge at a higher rate than the maximum charge power rate. Appendix C includes the script of the heuristic algorithm implemented in this thesis.
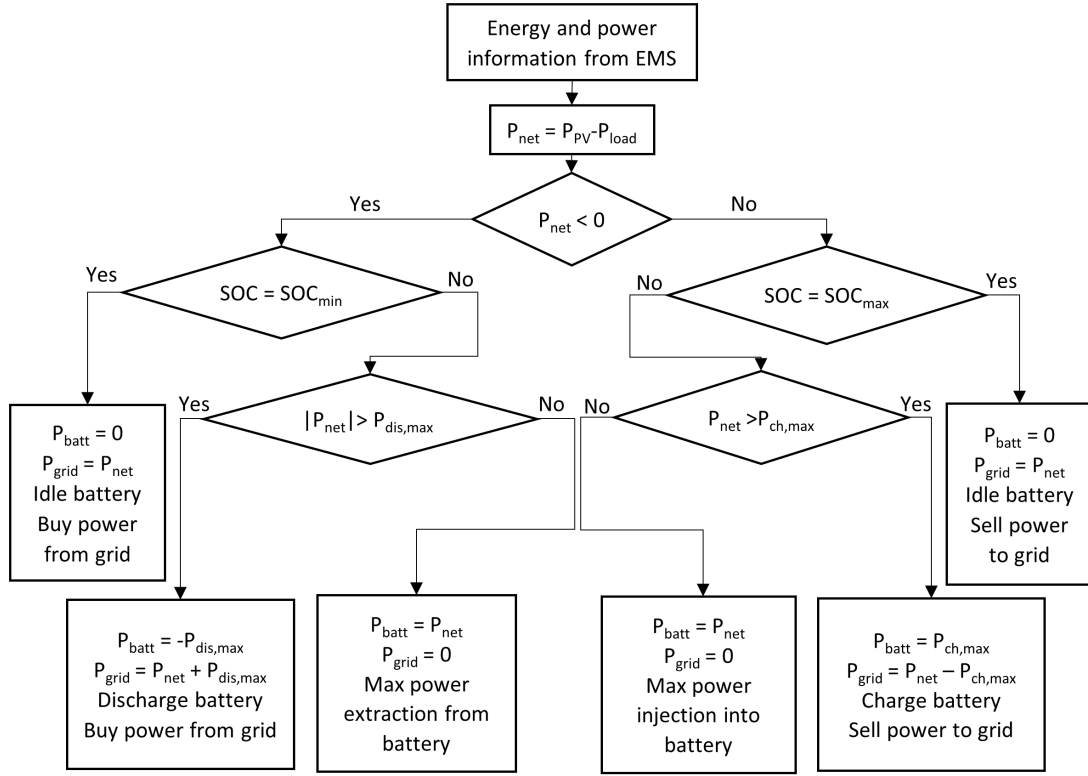
Figure 4.1: Flowchart for the heuristic algorithm.

4.2   Optimization-based method

In accordance with the literature review performed in Section 2.3, the aim of this section is to develop a scheduling algorithm that combines model predictive control (MPC) with mixed-integer linear programming (MILP) to determine the optimal charging and discharging power set-points of the battery. In the following sections, the overall MILP optimization problem will be formulated by defining constraints and a multi-objective cost function, before the MPC approach is applied to solve the problem. Finally, the implementation of the developed algorithm into the Simulink microgrid model is described.

4.2.1   Constraints

This section introduces the model constraints of the optimization problem. These constraints define the microgrid operational framework and reflect the limits of the microgrid components. Positive convention for power flows in the constraints are defined by Figure 4.2.
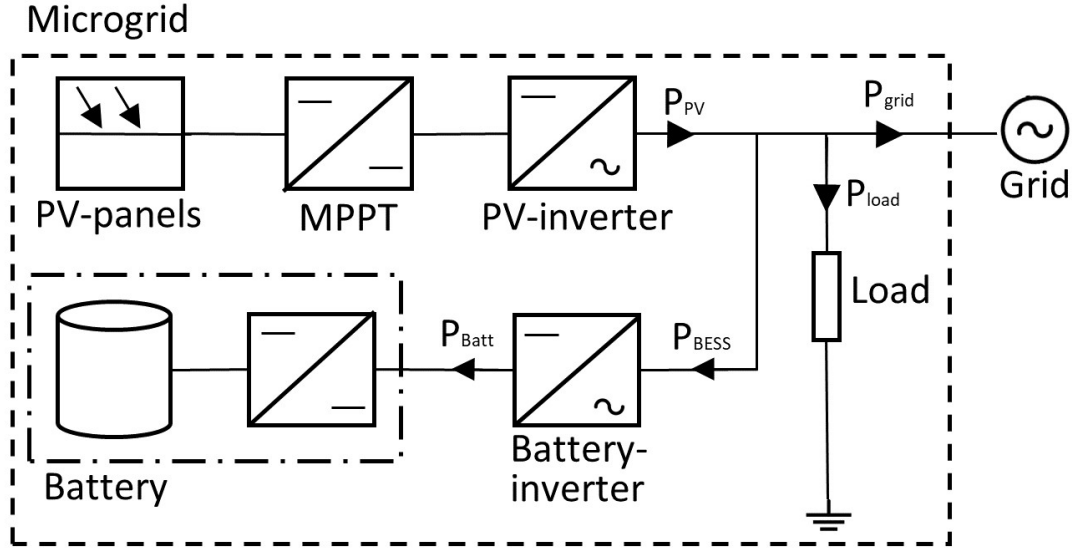
Figure 4.2: Illustration showing positive sign convention for power flows in the microgrid. Inspired by Figure 1 in [22].

4.2.1.1   Battery

The battery model in the optimization problem needs to indicate the available energy in the battery, which is generally estimated as the state of charge (SOC). The discrete model expressed by (4.1) estimates the SOC based on the data of the previous time instant.

$$
SOC(k) = \begin{cases} SOC(k-1) + \frac{\eta_{ch} \cdot P_{batt}(k)}{E_{cap}(k)} \cdot \Delta t, & P_{batt}(k) \geqslant 0 \\ SOC(k-1) + \frac{P_{batt}(k)}{\eta_{dis} \cdot E_{cap}(k)} \cdot \Delta t, & P_{batt}(k) < 0 \end{cases} \tag{4.1}
$$

where $SOC(k)$ and $SOC(k-1)$ denote the state of charge at time instants "k" and "k-1" respectively, $\frac{P_{batt}(k)\Delta t}{E_{cap}(k)}$ is the change in battery energy during time interval Δt, η_{ch}/η_{dis} is the battery charging/discharging efficiency, and $E_{cap}(k)$ is the total capacity of the battery at time instant "k". E_{cap} does not remain constant, but continuously decreases over time due to aging effects.

The battery is charging if P_{batt} is positive and discharging if P_{batt} is negative. A binary variable defined by (4.2) is introduced to the model to represent this logical condition and avoid simultaneous charging and discharging of the battery.

$$
\delta_b(k) = \begin{cases} 0, & \text{for charging} \\ 1, & \text{for discharging} \end{cases} \tag{4.2}
$$

Then, following the power flow directions given by Figure 4.2, the battery power can be represented as:

$$P_{batt}(k) = P_{ch}(k) \cdot (1 - \delta_b(k)) - P_{dis}(k) \cdot \delta_b(k) \tag{4.3}$$

where $P_{ch}(k)$ and $P_{dis}(k)$ represent the power transferred to/from the battery during charging and discharging at time instant "k", respectively.

Finally, combining both the charging and discharging properties of the battery into one equation, the SOC of the battery is:

$$SOC(k) = SOC(k-1) + \frac{\eta_{ch} \cdot P_{ch}(k)}{E_{cap}(k)} \cdot \Delta t - \frac{P_{dis}(k)}{\eta_{dis} \cdot E_{cap}(k)} \cdot \Delta t \tag{4.4}$$

The SOC range should be limited to maximize the battery lifetime by avoiding deep discharge or overcharge. The following constraint reflects the minimum and maximum SOC limits:

$$SOC_{min} \leq SOC(k) \leq SOC_{max} \tag{4.5}$$

Finally, (4.4) and (4.5) can be combined to form one overall SOC constraint:

$$SOC_{min} - SOC(k-1) \leq \frac{\eta_{ch} \cdot P_{ch}(k)}{E_{cap}(k)} \cdot \Delta t - \frac{P_{dis}(k)}{\eta_{dis} \cdot E_{cap}(k)} \cdot \Delta t \leq SOC_{max} - SOC(k-1) \tag{4.6}$$

In addition, (4.7) is often added as a constraint in the optimization problem to prevent the battery schedule from draining all the stored energy in the battery. (4.7) addresses this by ensuring that the SOC at the end of the horizon is the same as the SOC at the beginning of the horizon. In this way, the continuity of the battery operation is maintained. However, the negative effects of omitting (4.7) may be small when a receding horizon approach is employed [66]. Including (4.7) may also lead to unnecessary cycling of the battery. This is further investigated in the case study performed in Chapter 5.

$$SOC(N) = SOC(0) \tag{4.7}$$

where $SOC(N)$ is the battery energy level at the end of the horizon N and $SOC(0)$ is the battery energy level at the beginning of the horizon N.

Lastly, constraints on the battery power are added to reflect the maximum power that can be charged/discharged by the battery over a fixed time interval. Considering the binary variable defined by (4.2), the battery power constraints for discharging and charging can be expressed as:

$$\delta_b(k)P_{dis}^{min} \leq P_{dis}(k) \leq \delta_b(k)P_{dis}^{max} \tag{4.8}$$

$$(1 - \delta_b(k))P_{ch}^{min} \leq P_{ch}(k) \leq (1 - \delta_b(k))P_{ch}^{max} \tag{4.9}$$

4.2.1.2 Interaction with the main grid

In the grid-connected mode of operation, the microgrid can either sell or buy energy to/from the main grid. To model this possibility and avoid simultaneous selling and buying of energy, a binary variable defined by (4.10) is added to the optimization problem.

$$\delta_g(k) = \begin{cases} 0, & \text{for selling energy to the grid} \\ 1, & \text{for buying energy from the grid} \end{cases} \tag{4.10}$$

Then, following the power flow directions given by Figure 4.2, the microgrid's interaction with the main grid can be represented as:

$$P_{grid}(k) = P_{sell}(k) \cdot (1 - \delta_b(k)) - P_{buy}(k) \cdot \delta_b(k) \tag{4.11}$$

where $P_{sell}(k)$ and $P_{buy}(k)$ represent the power transferred to/from the grid at time instant "k", respectively.

Constraints (4.12) and (4.13) are included in the problem formulation to limit the maximum power that can be drawn from the grid and the maximum power that can be fed into the grid, respectively.

$$0 \leq P_{buy}(k) \leq \delta_g(k)P_{buy}^{max} \tag{4.12}$$

$$0 \leq P_{sell}(k) \leq (1 - \delta_g(k))P_{sell}^{max} \tag{4.13}$$

4.2.1.3 Power balance

The balance between power production and consumption in the microgrid must be met at every time instant. This is ensured by always satisfying (4.14), where power conversion efficiencies and positive power directions are included as defined in Figure 4.2.

$$\hat{P_{load}}(k) = \hat{P_{pv}}(k) + \eta_{inv}(k) \cdot P_{dis}(k) - \frac{P_{ch}(k)}{\eta_{inv}(k)} + P_{buy}(k) - P_{sell}(k) \tag{4.14}$$

where $\hat{P_{load}}(k)$ and $\hat{P_{pv}}(k)$ are the forecasted load and PV output curves, respectively. The battery inverter efficiency, $\eta_{inv}(k)$, varies depending on the battery power as described by (3.7) in Chapter 3.

4.2.2 Objective function

An objective function must be formulated in the optimization problem to achieve optimized microgrid operation. This thesis considers a multi-objective optimization problem, where the goal is to achieve an optimal solution for several competing objectives. The following subsections will first describe each of the objectives before the overall objective function is presented.

4.2.2.1 Grid cost function

The first objective is to minimize the total cost of variable-priced electricity. This objective can be expressed as the difference between the economic cost of buying energy from the grid and the monetary income of selling energy to the grid:

$$J_{grid} = \sum_{k=0}^{N-1} \left(c_{buy}(k) \cdot P_{buy}(k) - c_{sell}(k) \cdot P_{sell}(k) \right) \Delta t \tag{4.15}$$

where N is the optimization step horizon, $c_{buy}(k)$ is the price of buying energy from the main grid at time instant "k", $c_{sell}(k)$ is the price of selling energy to the main grid at time instant "k", $P_{sell}(k)/P_{buy}(k)$ are the power flows to/from the grid, and Δt is the sample time.

4.2.2.2   Battery cost function

Minimizing the grid cost function will lead to savings in the energy bill. However, frequent cycling of the battery will shorten its lifetime. Therefore, every charge or discharge cycle accumulates an additional cost, and it is desirable to reduce the number of cycles during the day. Consequently, in addition to the grid cost function, a cost function representing the opportunity costs of battery cycles should be included in the optimization problem.

The opportunity costs of battery cycles can be calculated using (4.16), where the change in battery energy, ΔE_{batt}, is multiplied by a battery cost weight, c_b, to penalize battery cycles. The size of c_b can be defined in several ways, which will be discussed further in Case 1 of the case study performed in Chapter 5.

$$J_{battery} = \sum_{k=0}^{N-1} \Delta E_{batt}(k) \cdot c_b \tag{4.16}$$

where $\Delta E_{batt}(k)$ is the absolute change in battery energy from one time step, k-1, to the next time step, k. $\Delta E_{batt}(k)$ is calculated by the following expression:

$$\Delta E_{batt}(k) = |E_{batt}(k) - E_{batt}(k-1)| = \left(\eta_{ch} \cdot P_{ch}(k) + \frac{P_{dis}(k)}{\eta_{dis}} \right) \cdot \Delta t \tag{4.17}$$

Combining (4.16) and (4.17), the final expression for the battery cost function is:

$$J_{battery} = \sum_{k=0}^{N-1} \left(\eta_{ch} \cdot P_{ch}(k) + \frac{P_{dis}(k)}{\eta_{dis}} \right) \cdot \Delta t \cdot c_b \tag{4.18}$$

The battery cost function reduces battery degradation by ensuring that only activities that generate higher profitability than the cycle opportunity costs are executed. In this way, battery degradation awareness is introduced into the optimization model.

4.2.2.3   Global cost function

Each of the two aforementioned objectives is formulated as a term in the following global cost function:

$$J = J_{grid} + J_{battery} \tag{4.19}$$

The two objectives of the global cost function are conflicting, and a single solution that simultaneously optimizes each objective may not exist. Consequently, a trade-off between saving energy costs and reducing battery degradation must be made. Energy costs are saved by utilizing the battery constantly to charge and discharge at all peaks in the energy price. However, frequent cycling of the battery will increase its degradation and lead to an earlier reinvestment of the battery. A trade-off between these conflicting objectives can be made by selecting a suitable value for the battery cost, c_b. As this value increases, the proposed algorithm will perform a lower number of charging-discharging actions. The effect of varying this value is investigated through the case study in Chapter 5.

4.2.3   Model Predictive Control using mixed-integer linear programming

Combining the constraints and the objective function defined in the previous sections, the overall optimization problem can be formulated as

$$
\begin{aligned}
\text{minimize} \quad & J \tag{4.19} \\
\text{subject to} \quad & \text{storage model (4.4),} \\
& \text{storage constraints (4.6), (4.7), (4.8), (4.9),} \\
& \text{grid constraints (4.12), (4.13),} \\
& \text{power balance (4.14),} \\
& \text{binary variables (4.2), (4.10)}
\end{aligned}
\tag{4.20}
$$

This problem is characterized as a mixed-integer nonlinear programming (MINLP) problem with a non-convex cost function due to the nonlinear dynamics of the battery capacity and power electronic efficiencies. Solving this MINLP optimization problem generates an optimal battery schedule. However, this schedule will be subject to uncertainties; the system model will be imperfect, forecast errors will be present, and the system state will not evolve as predicted. The single optimization problem yields an open-loop solution, which does not account for these uncertainties.

Alternatively, the optimization problem can be solved within a model predictive control (MPC) framework. In this way, a feedback control mechanism is implemented, which potentially compensates for the uncertainties. In addition, the MPC approach enables

the power converter efficiency and battery capacity to be computed and updated prior to the optimization at each sample time and then considered constant over the prediction horizon. In this way, the optimization problem is transformed into a MILP problem, which is preferred over the corresponding MINLP formulation due to, e.g., less computational effort, access to available solvers, and a guaranteed optimal solution without noticeable loss of accuracy compared to MINLP. A more detailed discussion of the MILP and MINLP approaches can be found in the literature review conducted in Section 2.3.

This thesis thus proposes a control approach that combines MPC with MILP to effectively account for uncertainties and to capture some of the nonlinear dynamics of the system by updating the system model every time step. With the general concept of MPC being described in Section 2.3.3, this section will describe the MPC controller developed for the microgrid considered in this thesis. Figure 4.3 depicts a block diagram of the MPC controller, and the following paragraphs describe the four steps that characterize the MPC strategy utilized in the controller.
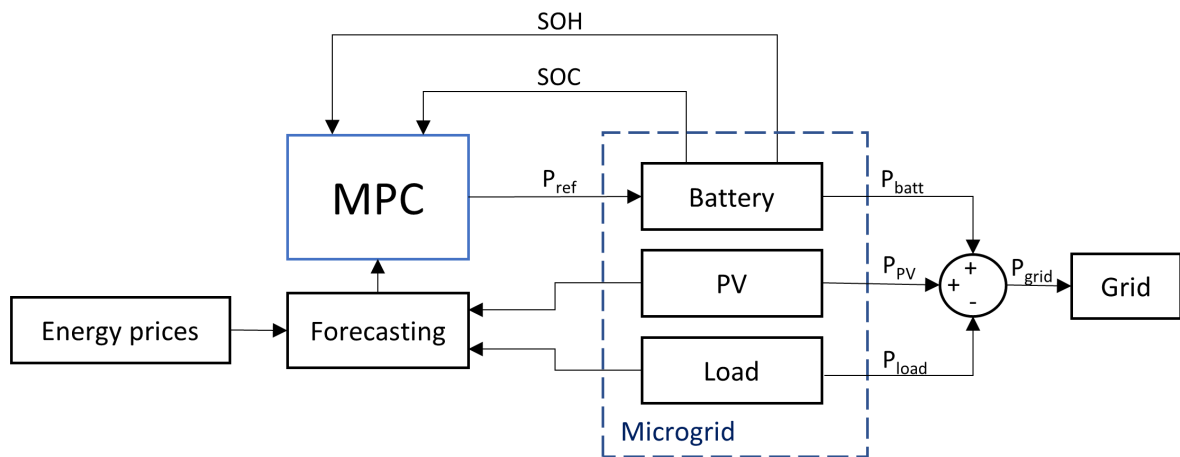
Figure 4.3: Overview of the MPC controller.

Step 1: Formulate the optimization problem

At the current point in time, an optimization problem is formulated for a selected prediction horizon, N_p, based on updated values of the system states and power electronic efficiencies as well as forecasts of the future PV production, load demand, and electricity prices. The current system states, $SOH(0)$ and $SOC(0)$, are updated using the SimSES battery model. Further, the power electronic efficiency is updated using the formula expressed by (3.7). It is important to notice that the efficiency and the SOH are considered constant over

the prediction horizon to keep the problem linear and enable the use of a MILP solving technique.

As illustrated in Figure 4.3, the proposed MPC-based EMS allows the implementation of a forecasting module to predict the load demand, the PV production, and the electricity prices at each time step $k = 1, ..., N_p$ of the chosen prediction horizon. However, proposing such a module is outside the scope of this thesis. Instead, Section 5.1.3 gives an alternative way of simulating forecast errors.

In order to solve the overall optimization problem expressed by (4.20), it must be reformulated into a form that can be solved using an available MILP solver. In this thesis, it is contemplated to use the *intlinprog* solver in MATLAB, and the optimization problem must thus be reformulated into the matrix form given by (4.21). The rewriting of the constraints into vectors and matrices is performed in Appendix D.1.

$$
\begin{aligned}
\underset{u}{\text{minimize}} \quad & J^T u \\
\text{subject to} \quad & u(intcon) \text{ are integers,} \\
& A_{ineq} \cdot u \leq b_{ineq}, \\
& A_{eq} \cdot u = b_{eq}, \\
& lb \leq u \leq ub
\end{aligned}
\tag{4.21}
$$

where

- The decision variables are collected in the vector

$$
u(k) = \begin{bmatrix} P_{dis}(k) & P_{ch}(k) & P_{buy}(k) & P_{sell}(k) & \delta_b(k) & \delta_g(k) \end{bmatrix} \text{ for k} = 0, ..., N_p - 1
$$

- The objective function J (4.19) is written as a vector with linear coefficients.

- The inequality constraints (4.6), (4.8), (4.9), (4.12), and (4.13) are represented using the matrix A_{ineq}, the vector b_{ineq}, and the decision vector u.

- The equality constraint (4.14) is represented using the matrix A_{eq}, the vector b_{eq}, and the decision vector u.

- The upper and lower limits of the decision variables, (4.2), (4.8), (4.9), (4.10), (4.12), and (4.13), are collected in lb and ub.

- *intcon* is a vector containing the integer constraints. The values in *intcon* indicate the components of the decision vector u that are integer-valued.

Step 2: Solve the optimization problem

By solving the MILP problem in (4.21), the optimal input sequence for the prediction horizon N_p is obtained as

$$u_{opt}(k) = \left[((u_{opt}(0))^T \quad ((u_{opt}(1))^T \quad \cdots \quad (u_{opt}(N_p-1))^T\right]$$

where each vector $u_{opt}(0)$, $u_{opt}(1)$, ..., $u_{opt}(N_p-1)$ is the future optimal input sequence for the sample times $k = 0, 1, ..., N_p - 1$, respectively.

Step 3: Execute the control set-points

Although a complete sequence of N_p future control signals is computed, only the first element, $u_{opt}(0)$, is applied to the system, and the remaining optimal values in $u_{opt}(k)$ are discarded.

Step 4: Shift the prediction horizon

At the next sample time, the prediction horizon is shifted, and a new optimal sequence, $u_{opt}(k)$, is obtained by repeating steps 1-3. This includes estimating the new state of the system, recalculating the power electronic efficiencies, obtaining new forecasts, and finally using this updated information to solve a new optimization problem. By this receding horizon approach, a feedback mechanism is created where the new optimal plan can potentially compensate for any disturbances that have meanwhile acted on the system.

4.2.4   Simulink implementation

This section will shortly describe how the MILP-MPC algorithm is implemented in Simulink. First, the MILP optimization algorithm is formulated as a MATLAB function using the MILP solver *intlinprog*. Further, a MATLAB Function block is used to implement the optimization function in Simulink, as shown in Figure 4.4. A discrete update method is selected for the MATLAB Function block, which means that the block is sampled at a rate specified in the block's Sample Time property. The optimization is thus rerun every sampling period, and a feedback mechanism (MPC) is created.
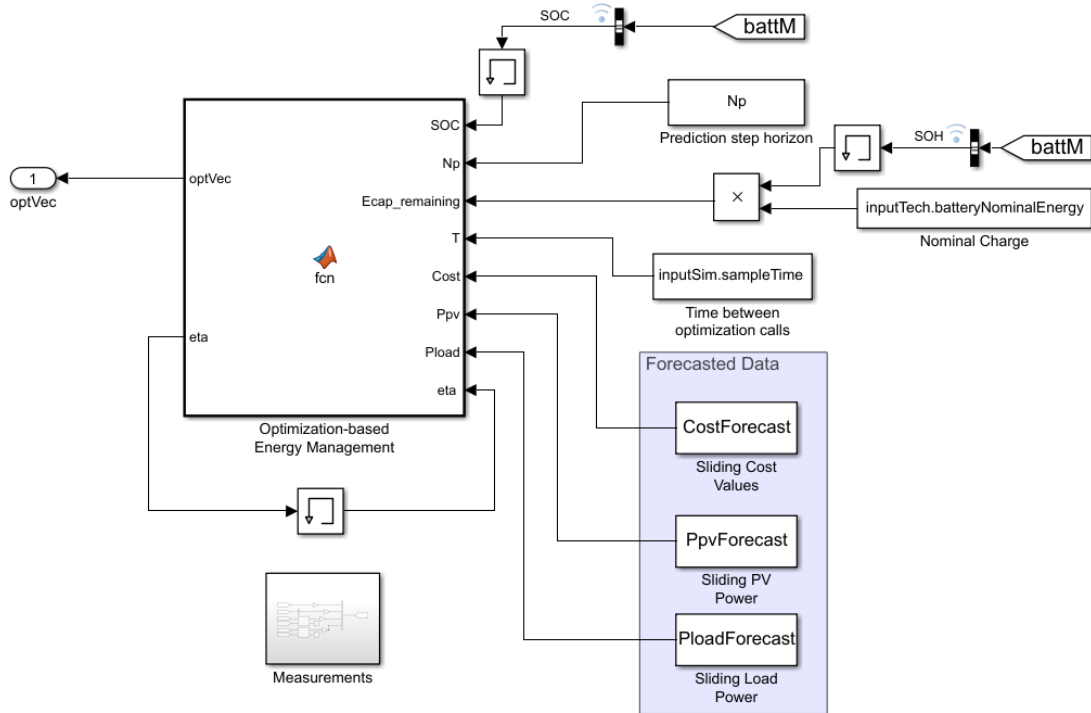
Figure 4.4: Exterior view of MATLAB Function block for optimization based energy management with all input and output flows.

It must be noticed that the code generator in the MATLAB Function block does not support the *intlinprog* function. To overcome this problem, the optimization function must be declared as extrinsic in the code generator script. The MILP optimization function and the code generator script are given in Appendices D.2 and D.3, respectively.

All inputs to the MATLAB Function block are initialized by the initialization script given in Appendix A. The inputs to the block are:

- Estimated values of SOH and SOC from the SimSES battery block.

- The prediction step horizon.

- The sample time, i.e. how often the MILP optimization function is called in Simulink.

- Forecasted data of energy costs, PV production, and load demand, generated by the forecast error function attached in Appendix E.

- The nominal battery capacity.

- Power electronic efficiency, which is updated at each sample time step according to the formula given by (3.7).

4.3   Summary

In this chapter, two energy management strategies were developed to determine the charging and discharging power set-points of a battery energy storage system in a grid-connected microgrid. As a first step, a simple rule-based method was developed to work as a reference for comparison. Further, an optimization-based scheduling algorithm based on the model predictive control approach was proposed. In the model formulation, the defined nonlinear problem was rewritten as a mixed-integer linear problem by considering the power converter efficiencies and the battery capacity constant over the prediction horizon. Then, the receding horizon principle was applied to determine the optimal input sequence, where the MPC controller updates the system model at each sample time. Consequently, the resulting energy management strategy was cast as a multi-objective MILP problem incorporated in a model predictive framework to account for disturbances and to capture some of the nonlinear dynamics of the system. Finally, the proposed MILP-MPC algorithm was implemented in the microgrid simulation platform developed in Chapter 3 by utilizing a MATLAB Function block.

5 Simulation results and discussion

This chapter investigates the performance of the proposed control approaches through an extensive case study over a two-month simulation period using actual PV and load data from Skagerak Energilab and electricity price profiles from Nordpool. Section 5.1 presents the parameters, data, and forecasting method used in the simulation process. Section 5.2 presents, comments, compares, and discusses the results of seven simulation cases to analyze the performance of the developed MILP-MPC based control method. Section 5.3 studies the algorithms' effects on the microgrid in terms of voltage, frequency, and operational limits. In addition, the computational approach is evaluated.

5.1 Parameters, data and forecasting

The microgrid parameters and data used in the simulation process were collected from Skagerak Energilab during September and October 2020 [17]. Skagerak Energilab is a testing facility for local production, storage, and distribution of electrical energy located in Skien in Norway. In particular, this testing facility comprises a PV system, loads, and a battery that are combined to make up a virtual microgrid. The project is run by Lede in collaboration with ABB, Kontorbygg AS, and Odds Ballklubb.

Figure 5.1 depicts a schematic of Skagerak Energilab illustrating how it can be considered a virtual microgrid. The blue arrow and the red box show how the PV panel from Segment 1 is virtually connected to the 0.4 kV busbar in Segment 2, which also has a battery connected. In reality, there is a power flow between the two segments, as illustrated by the green arrow. However, it is possible to operate the virtual microgrid isolated from the rest of the 11 kW system and the utility grid. The boxes marked with PoM1, PoM2, PoM3, PoM4, PoM5, PoM6, and PoM7 are the measurement points of the system.
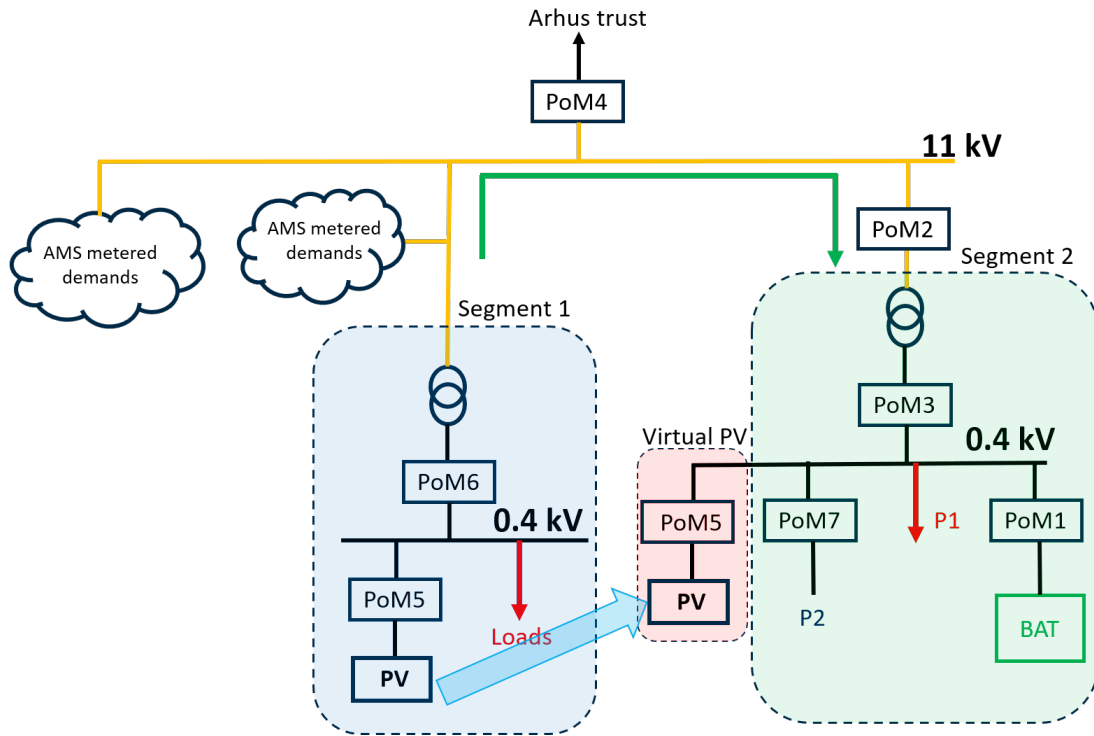
Figure 5.1: Schematic of Skagerak Energilab [67].

The next sections give the parameters and data obtained from the Skagerak Energilab testing facility in addition to the rest of the simulation parameters and forecasting options considered in the simulations.

5.1.1   Parameters

Table 5.1 lists the microgrid parameters used in the simulation process.

Table 5.1: Microgrid parameters used in the simulation process.

Parameter	Value
Microgrid voltage	400 V
Microgrid frequency	50 Hz
$P_{buy,max}$	1000 kW
$P_{sell,max}$	1000 kW
Base load	10 kW
Battery type	Lithium-ion
Battery capacity	1000 kWh
Initial SOC	50 %
SOC_{min}	20 %
SOC_{max}	80 %
$P_{batt,min}$	400 kW
$P_{batt,max}$	400 kW
$P_{rated,inv}$	400 kW
Battery round trip efficiency, η_{batt}	94 %
Inverter efficiency, η_{inv}	$\eta_{inv} = \frac{p}{p+p_0+k \cdot p^2}$ where $p = \frac{|P_{out}|}{P_{rated,inv}}$
PV panel type	REC295TP2 and REC300TP2
PV area	4400 m^2
Maximum PV power	800 kW(p)
Simulation time	2 months
Sample time	5 min

The simulation process considered a two-month simulation time to investigate the capability of the algorithm to control the microgrid operation for an extended period of time. The PV, load, and electricity price profiles varied noticeably over the two months, providing a solid foundation for an interesting and challenging case study. The sample time was set to five minutes as a compromise between computational effort and control performance. A large sample time may lead to deficient control performance, while a small sample time can result in a high computational effort.

The SOC range should be limited to maximize the battery lifetime by avoiding deep discharge or overcharge. Experiments show that charging lithium-ion batteries to 85% provides a longer service life than charging them to 100% [68]. Fully discharging batteries is also not recommended because many cell chemistries cannot tolerate deep discharge, and cells may be permanently damaged if fully discharged. Therefore, the minimum and maximum SOC limits considered in this thesis are 20% and 80%, respectively.

5.1.2   Data

The simulation process considered the PV and load output power at Skagerak Energilab for September and October 2020 [17], as shown in Figure 5.2. These data sets have a one-minute time resolution, and missing data points were filled using linear interpolation.
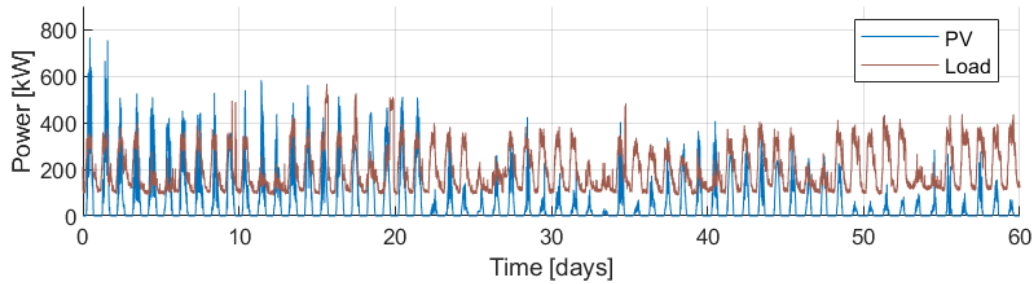
Figure 5.2: PV and load output power at Skagerak Energilab for September and October 2020 [17].

In addition to the data from Skagerak Energilab, the simulation process utilized the real-time electricity market price data reported by Nordpool for September and October 2020 [69]. This data has a one-hour time resolution, and it is depicted in Figure 5.3. In this real-time pricing scheme for buying energy from the main grid, the cost per kWh depends on the time of using that kWh. The price received for selling energy to the main grid was assumed equal to the price of buying energy for all cases except for Case 2. In Case 2, the selling price was set to 50% of the buying price.
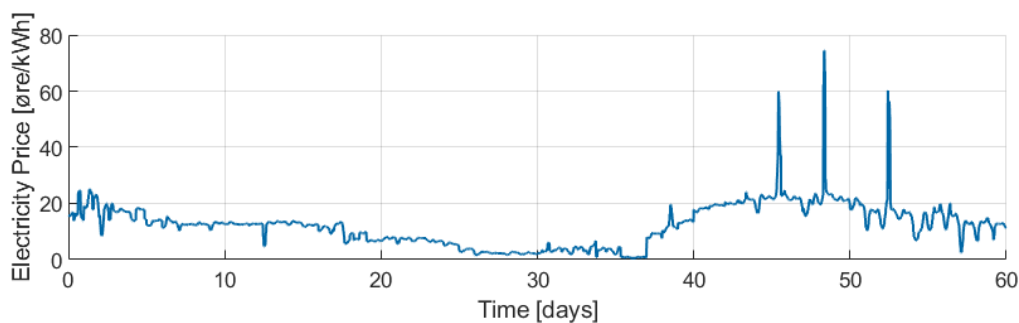
Figure 5.3: Hourly spot price reported by Nordpool for September and October 2020 [69].

5.1.3   Forecasting

The PV, load, and electricity price data presented in the previous sections are not known by the energy management system in advance. Therefore, the MILP-MPC control strategy proposed in Chapter 4 allows the implementation of a forecast module to predict these

outputs. However, proposing such a module is outside the scope of this thesis. Instead, an error function was developed to simulate uncertainty in forecasts by adding errors to the actual values. These errors were modeled with a gradient uncertainty level in which the forecast error increases when the prediction horizon becomes larger. A similar forecast error function has also been used by other researchers [70], and it should be sufficient for testing how the feedback mechanism of the MPC handles uncertainties in forecasts. Most cases studied in the case study in Section 5.2 consideres a forecast error that increased from 10% to 20% over the prediction horizon. In addition, one of the cases investigates how the MPC handled different forecast errors. Appendix E includes the MATLAB script used to model forecast errors.

5.2   Case study

To investigate the capability of the developed energy management system and microgrid model for dealing with different scenarios, the following cases have been proposed and simulated:

- Case 0a: Heuristic reference case

- Case 0b: MILP-MPC reference case

- Case 1: Impact of battery cost

- Case 2: Impact of selling price

- Case 3: Impact of end of the day SOC constraint

- Case 4: Impact of prediction horizon

- Case 5: Impact of forecast accuracy

Each case tests the impact of varying one parameter in the optimization problem. Table 5.2 gives an overview of the parameter values considered in each case.

Table 5.2: Parameters used in the case study.

Case	Selling price	Battery weight	Forecast error	Prediction horizon	SOC(end) = SOC(0)[1]
0	Equal to buying price	0	10-20%	8h	No
1	Equal to buying price	**0-0.2**	10-20%	8h	No
2	**50% of buying price**	0	10-20%	8h	No
3	Equal to buying price	0	10-20%	8h	**Yes**
4	Equal to buying price	0	10-20%	**4-24h**	No
5	Equal to buying price	0	**0-40%**	8h	No

[1] This condition states whether the end of the day SOC constraint expressed by (4.7) is included in the optimization problem or not.

The following subsections give the simulation results of the proposed cases. All cases were simulated for a time period of two months, and the numerical results are plotted with different time resolutions depending on the studied variables and the case objectives. A two-day time resolution is considered when plotting the battery SOC, the electricity price, and the PV, load, battery, and grid powers. A two-month time resolution is used when plotting the battery SOH. Finally, the total grid energy costs for the two months are used to find the daily average energy costs for each scenario.

5.2.1   Case 0a: Heuristic reference case

A heuristic case is included in the case study to provide a broader base for comparison where the results of the optimization-based MILP-MPC algorithm can be compared to the results of a simple rule-based heuristic algorithm that does not optimize a cost function. In this way, the advantages of utilizing an optimization-based energy management technique to schedule the battery are highlighted. The heuristic algorithm is described in Section 4.1, the corresponding MATLAB script is attached in Appendix C, and the values of the battery constraints are selected as given by Table 5.1. Figure 5.4 shows the simulation results of the heuristic case for the first two days of the simulation period.
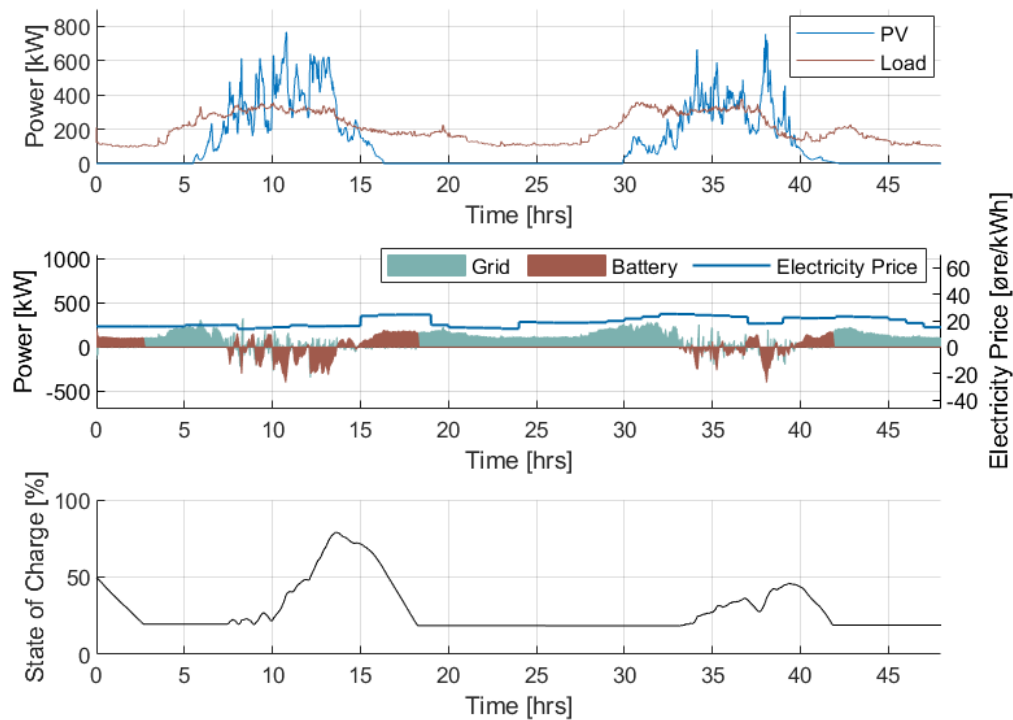
Figure 5.4: Results of the heuristic reference case for the first two days of the two-month simulation period.

From Figure 5.4, it can be observed that the battery charges only when there is an excess of PV power available, which is expected because the defined rules do not allow the battery to charge from the main grid. There is enough excess PV power to charge the battery to its maximum SOC limit (i.e., 80%) during the first day, as opposed to the second day when the battery barely charges. On days where the load demand is higher than the PV power generation, the heuristic algorithm will not charge the battery at all. This is the case at Skagerak Energilab for most days in September and October 2020, as shown in Figure 5.2, which implies that the battery is idle most of the time.

Figure 5.4 also shows that the heuristic strategy never stores energy in the battery for a long period of time. Instead, it prefers to use the stored energy to feed the microgrid loads as soon as the demand exceeds the generation. By coincidence, this occurs when the electricity price is high even if no cost function is minimized. In this way, the microgrid avoids importing expensive energy from the grid during these periods. Moreover, the results in Figure 5.4 show that the battery can serve the loads for approximately three hours before the SOC falls from its maximum value (80%) to its minimum value (20%), as seen on the first day. From this hour on, all power demand must be met by the main grid

until more PV power is generated.

Table 5.3 shows the microgrid interaction with the main grid for the considered simulation period, i.e., September and October 2020. Little excess PV power is available for these months. Therefore, a large amount of energy is imported from the main electrical grid to ensure that the load demand is always met. Moreover, the defined rules of the heuristic algorithm do not allow the battery to discharge to the grid. Thus, all exported energy is excess PV generation that could not be stored in the battery, and the microgrid exports little energy to the main grid.

Table 5.3: Heuristic case: Total exchange with the main grid for the two-month simulation period.

	kWh	NOK
Imported (purchased)	199857	24484
Exported (sold)	-4409	-587
Total	195448	23897

Based on the above discussion, it can be concluded that the heuristic algorithm prefers to utilize the PV generation internally instead of selling it to the utility grid, i.e. self-consumption of locally produced PV power is prioritized. Moreover, the algorithm does not schedule the battery to feed the microgrid loads strategically based on the peaks in electricity price. Instead, the battery serves the loads as soon as the PV array generates too little power. Furthermore, the heuristic algorithm always operates the battery within its limits. The SOC never exceeds its minimum value of 20% or its maximum value of 80%, and the power output of the battery is always within the $\pm$ 400 kW power constraint.

5.2.2   Case 0b: MILP-MPC reference case

This case will serve as a reference for the case study, where each case alters one parameter of the reference case at a time, as indicated in Table 5.2. This enables a thorough investigation of how each parameter of the proposed MILP-MPC algorithm affects the optimal battery schedule.

The MILP-MPC reference case considers the PV and load data shown in Figure 5.2, the electricity price profile in Figure 5.3, and the microgrid parameters listed in Table 5.1. The price received for selling energy to the grid is set equal to the buying price. A prediction horizon of 8 hours is considered, where the forecast error increases from 10% to 20% over

the prediction horizon. Moreover, the battery cost c_b is zero in the reference case, which means that the objective function only includes the grid cost function.

Figure 5.5 shows the results of the optimal dispatch provided by the MILP-MPC reference case for the two first days of the simulation period. As expected, the operation of the battery is mainly scheduled by the electricity price, in which the battery charges when the price is low and discharges when the price is high.
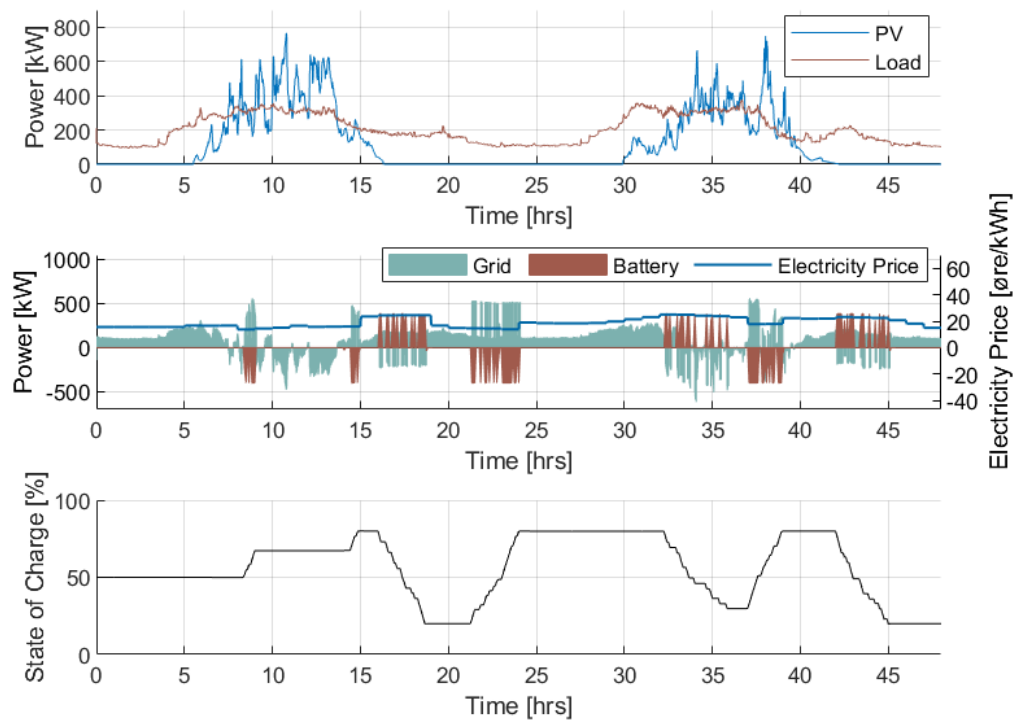
Figure 5.5: Results of the MILP-MPC reference case for the two first days of the two-month simulation period.

The results in Figure 5.5 show that the battery charges at off-peak times (hours 9, 15, 22, and 37) when the price of buying electrical energy from the main grid is low. Later, this stored energy is used to serve the microgrid loads at peak tariff times (hours 16, 33, and 43) to avoid importing expensive energy from the main grid. In addition, since the reference case assumes equal selling and buying prices, the algorithm also wants to sell energy to the main grid at peak tariff. Therefore, the battery outputs as much power as allowed by the power limits (400 kW) during peak periods to both feed the loads and obtain a high selling reward for exporting energy to the main grid.

From Figure 5.5, it can also be observed that the excess PV power is not used to charge

the battery. Instead, it is sold to the main grid to obtain a revenue due to the high selling price considered in this case. Therefore, in contrast to the heuristic case, the MILP-MPC reference case does not prioritize self-consumption of locally produced PV power.

The microgrid interaction with the main grid for the MILP-MPC reference case is given in Table 5.4. By comparing the results presented in this table to the results from the heuristic case in Table 5.3, it can be seen that the MILP-MPC reference case both exported and imported a higher amount of energy than the heuristic case while still obtaining lower total energy costs. Therefore, it can be concluded that the MILP-MPC algorithm managed to use the grid energy more optimally, i.e., when it was cheaper. This highlights the advantage of implementing an optimization-based energy management system in the microgrid.

Table 5.4: MILP-MPC reference case: Total exchange with the main grid for the two-month simulation period.

	kWh	NOK
Imported (purchased)	221263	25473
Exported (sold)	-21988	-3041
Total	199275	22431

5.2.3 Case 1: Impact of battery cost

This case investigates the effects of varying the value of the battery cost, c_b, in the objective function. A higher battery cost should reduce the number of cycles during the day and thus reduce battery degradation.

The battery cost can be selected in several ways. One option is to choose a fixed battery cost per kWh. However, since the electricity price varies a lot over the year, the ratio between the grid cost and the battery cost in the objective function will vary a lot when considering a fixed battery cost. The result is long periods where this ratio is too small for the battery to be cycled, even for high peaks in the electricity price. Also, [70] argues that a fixed battery cost does not effectively reduce the charging and discharging frequency. Therefore, this thesis considers the battery cost as a percentage of the grid cost.

In this case, four scenarios where the battery cost is increased from 0% to 20% of the grid cost is simulated, i.e., $c_b=0$, $c_b=0.05$, $c_b=0.1$ and $c_b=0.2$. If a higher percentage than this is used, cycling the battery will be very expensive, and the battery will thus be idle most

of the time. Figure 5.6 depicts the simulation results of the scenario where the battery cost is 5% of the grid cost.
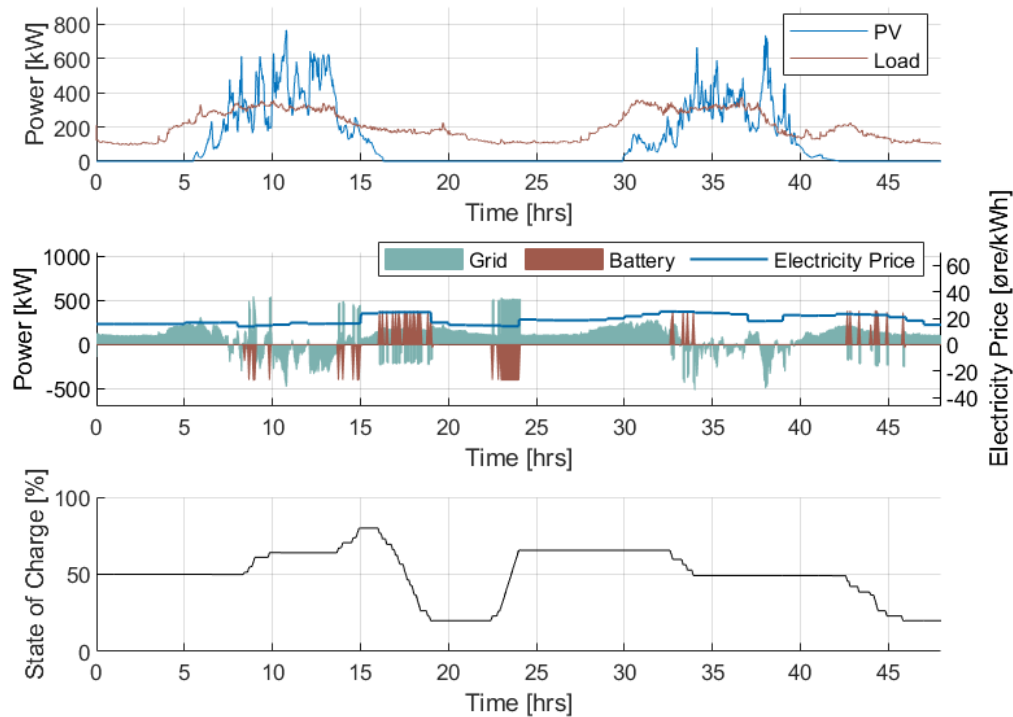
Figure 5.6: Case 1: Results obtained for the scenario when the battery cost is 5% of the grid cost, i.e., $c_b = 0.05$.

In the MILP-MPC reference case, where no battery cost was considered, even a slight difference between the electricity prices at different time slots cycled the battery. However, as expected, the results in Figure 5.6 show that introducing a battery cost function to the optimization problem prevents such redundant behavior. In this case, the battery will only export energy to the grid if the savings made by selling the stored energy are higher than the battery cost of cycling that energy. Moreover, the battery is less likely to charge from the grid if the algorithm does not see a very significant peak in the predicted electricity price. The battery is, for instance, no longer charging during the low price period at hour 37.

Figure 5.7 depicts the SOH development for different battery costs over the two-month simulation period. As expected, the battery degrades less for high battery costs due to the reduced number of cycles. Moreover, the results show that the heuristic case has a similar SOH development as the MILP-MPC reference case ($c_b = 0$) for the first 25 days when the PV generation is high. However, for the last 35 days, little excess PV power

is available. Therefore, the heuristic algorithm barely cycles the battery, and the SOH decreases at a lower rate because all degradation originates from calendrical aging.
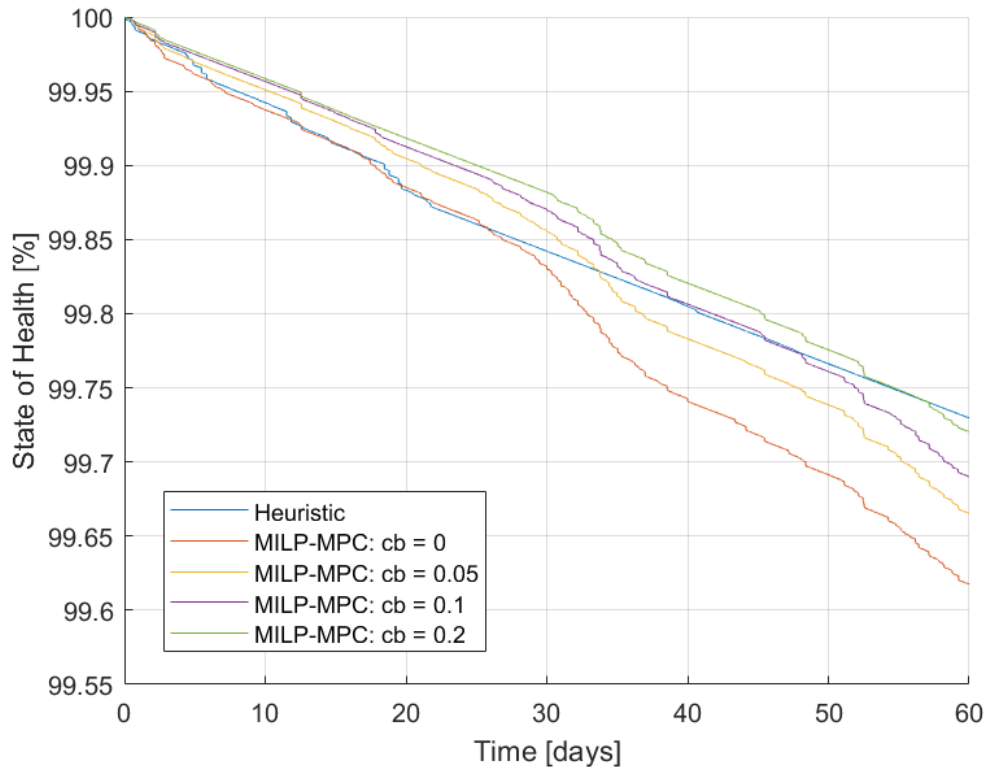
Figure 5.7: Case 1: Battery SOH development for scenarios with different battery costs for a two-month simulation period.

The total energy costs of each scenario have been averaged over the 60 simulated days and are presented in Figure 5.8. Expectedly, the energy costs increase when battery costs are included because the battery is no longer cycled at all peaks in the electricity price. However, since higher battery costs prolong battery lifetime (i.e., postpone battery reinvestment), increased energy costs do not necessarily mean that the total long-term microgrid costs are higher. For example, the scenario with $c_b = 0.2$ has a final SOH value close to the heuristic case but much lower costs. It will therefore result in a more economically favorable battery schedule at the same time as the battery needs reinvestment at a later point. This result highlights the advantage of utilizing an optimization-based energy management method over a heuristic method.
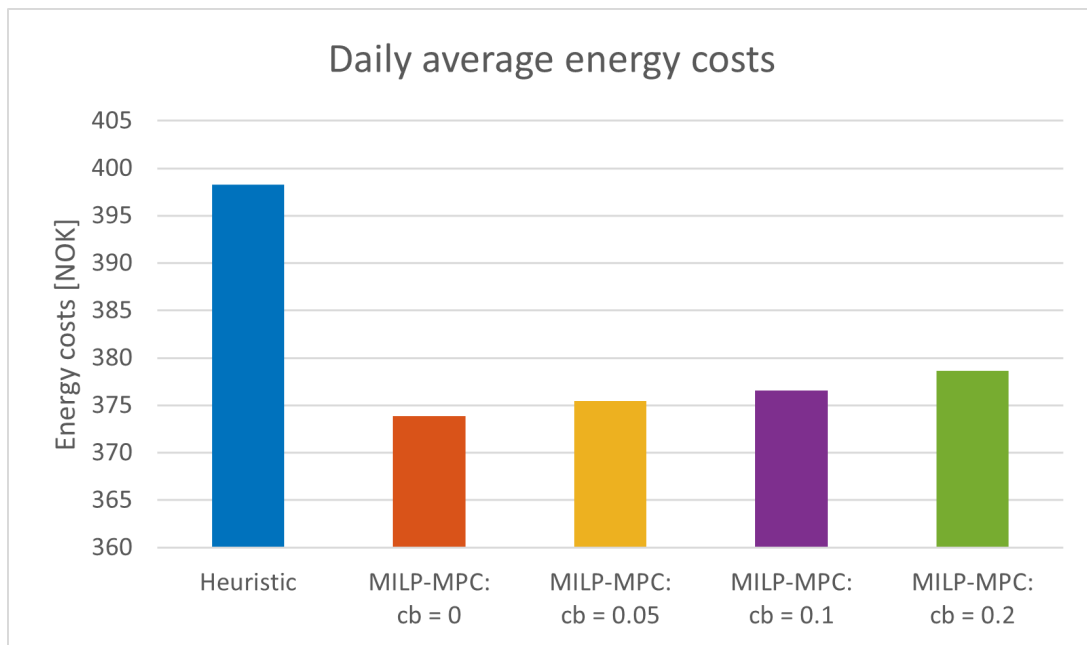
Figure 5.8: Case 1: Daily average energy costs for scenarios with different battery costs.

The above discussion clearly shows the conflicting objectives of the global cost function. The grid cost function wants to cycle the battery as much as possible to reduce costs related to the energy exchange with the main grid, while the battery cost function aims at reducing the number of battery cycles to limit battery degradation. However, it can be concluded that introducing the battery cost function to the optimization problem improves the overall result because battery degradation is limited while the daily energy costs barely increase and are still much lower than for the heuristic case.

5.2.4   Case 2: Impact of selling price

Case 2 investigates the effects of considering a different pricing scheme for selling energy to the main grid. Instead of assuming equal selling and buying prices, this case sets the selling price to 50% of the buying price. This should make the algorithm less likely to sell electricity to the main grid because a smaller reward is obtained. Figure 5.9 shows the results obtained for the two first days of the simulation period.
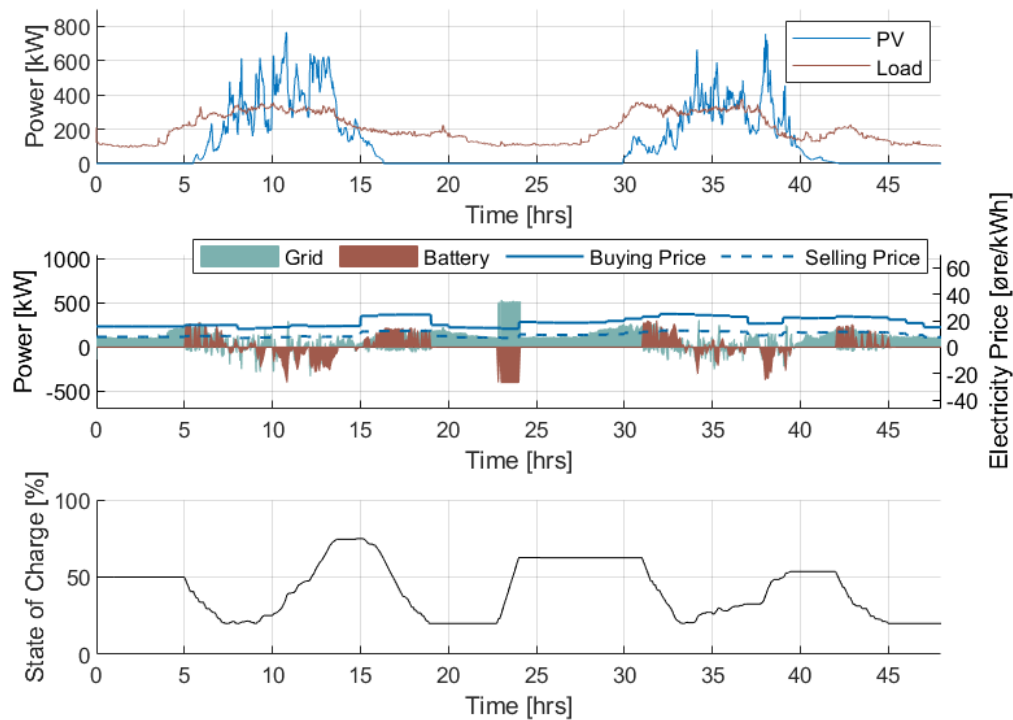
Figure 5.9: Case 2: Results obtained when the selling price is set to 50% of the buying price.

From Figure 5.9, it can be observed that the battery receives minimal charging from the main grid. It only charges from the grid at hour 24 when the electricity price is low. Instead, the battery stores energy during periods of excess PV generation and uses this energy to meet the load demand at a later point when the electricity price is higher. In this way, the PV power is utilized within the microgrid.

By comparing case 2 to the heuristic case, it can be seen that the two cases have a similar behavior where self-consumption of locally produced PV power is prioritized. Both cases prefer to use excess PV power to charge the battery. However, a difference is observed in the way the battery uses its stored energy to feed the microgrid loads. While the battery feeds the loads as soon as the demand exceeds the generation in the heuristic case, the MILP-MPC strategy does not discharge the battery until the electricity price is high to avoid buying expensive energy from the main grid. The result is a lower total energy cost for the two months considered, where the costs are reduced from 24191 NOK in the heuristic case (when $c_{sell} = 0.5 \cdot c_{buy}$) to 23299 NOK in case 2.

Since less reward is obtained from selling energy to the main grid, the algorithm avoids

exporting energy. By comparing Table 5.5 with Table 5.4, it can be seen that the exported energy has been reduced from 21988 kWh in the MILP-MPC reference case to 8455 kWh in case 2, which corresponds to a 61.6 % reduction percentage. Less energy is also imported in case 2 because the battery prefers to use excess PV power for charging instead of charging from the grid. Therefore, it can be concluded that the microgrid interacts less with, and consequently alleviate stress on, the main electrical grid in case 2.

Table 5.5: Case 2: Total exchange with the main grid for the two-month simulation period.

	kWh	NOK
Imported (purchased)	206893	23913
Exported (sold)	-8440	-614
Total	198453	23299

The results in Figure 5.10 show that the SOH decreases less, i.e. the battery lifetime is prolonged, when a lower selling price is considered. This is because the algorithm no longer prioritizes to cycle the battery at all peaks in the electricity price. Instead, the battery mostly charges when excess PV power is available and discharges to feed the loads when the electricity price is high. It can also be observed that for a high battery cost, the evolution of the SOH is very similar regardless of the selling price scheme implemented.
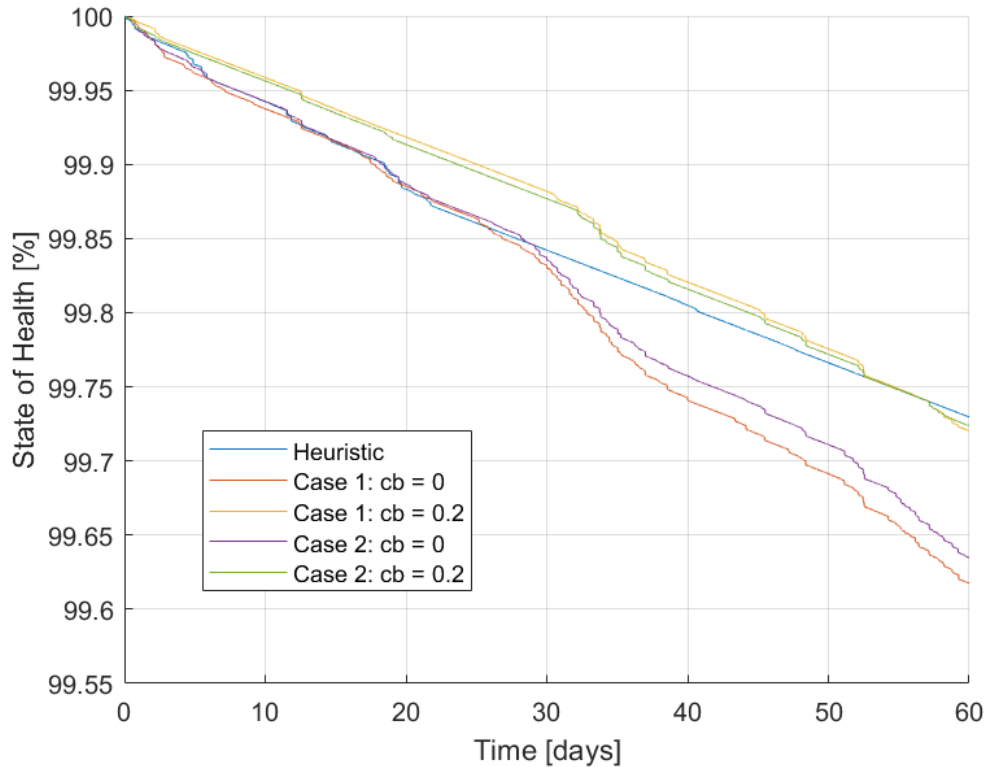
Figure 5.10: Case 2: Battery SOH development for scenarios with different selling price and battery weight for a two-month simulation period.

It makes no sense to compare the daily average energy costs of Case 2 with the costs of the other cases, because a lower selling price is assumed. Therefore, this case will naturally have higher costs, since a lower selling reward is obtained per kWh exported.

5.2.5   Case 3: Impact of end of the day SOC constraint

Section 4.2.1.1 suggested that the negative effects of omitting the end of the day SOC constraint expressed by (4.7) may be small when a receding horizon approach is employed. Therefore, case 3 investigates the impact of this constraint within a model predictive framework.

First, the battery behavior for the MILP-MPC reference case where the final SOC constraint is not included will be studied for a three-day simulation period. Figure 5.11 depicts these simulation results.
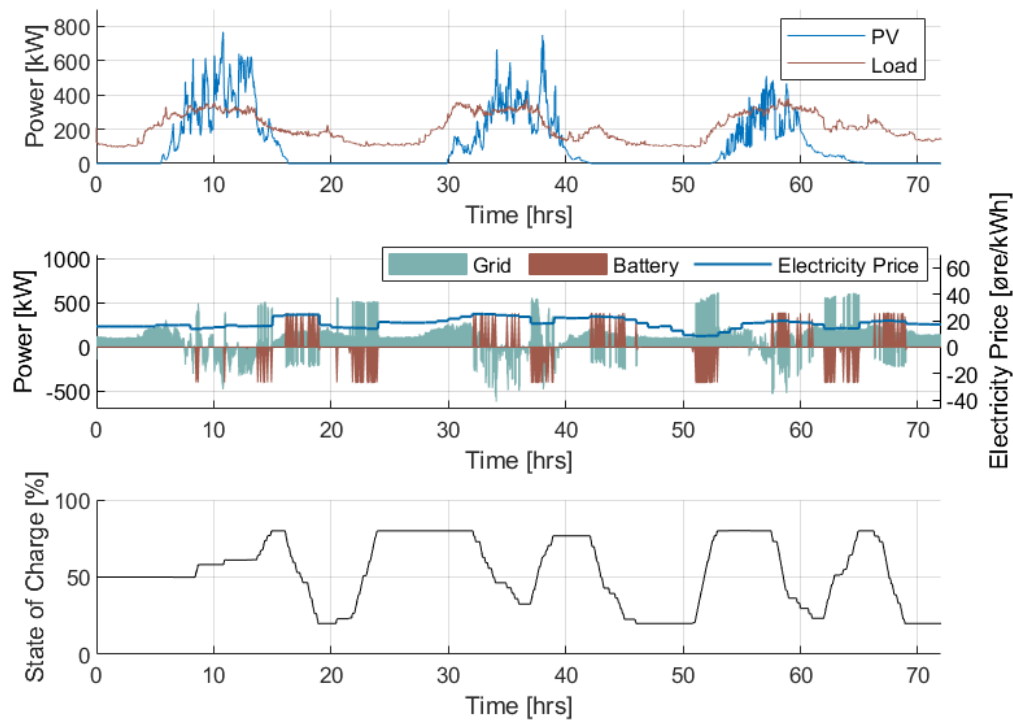
Figure 5.11: Case 3: Results **without** the end of the day SOC constraint for the first three days of the two-month simulation period.

The results in Figure 5.11 show that the SOC level at the end of the first day (hour 24) is equal to the initial SOC level (i.e., 50%) due to a predicted peak in the electricity price. This result occurs even though the optimization problem does not include the end of the day SOC constraint. At the end of the second day (hour 48), the battery is drained to its minimum SOC level (i.e., 20 %). This is because the foresight of the MILP-MPC algorithm expects the price to decrease more, and to obtain higher economic benefits, the algorithm waits for the low price at hour 51 to charge the battery. Therefore, it can also be assumed that the final SOC at day three (hour 72) is not 50% because it is more economically favorable to charge the battery at a later time.

Next, the battery behavior will be studied for a scenario where the end of the day SOC constraint expressed by (4.7) is included in the optimization problem. This constraint forces the algorithm to prepare the battery for the next day. It also generalizes the optimal dispatch of the system and maintains the continuity of the battery operation. Figure 5.12 shows the corresponding simulation results.
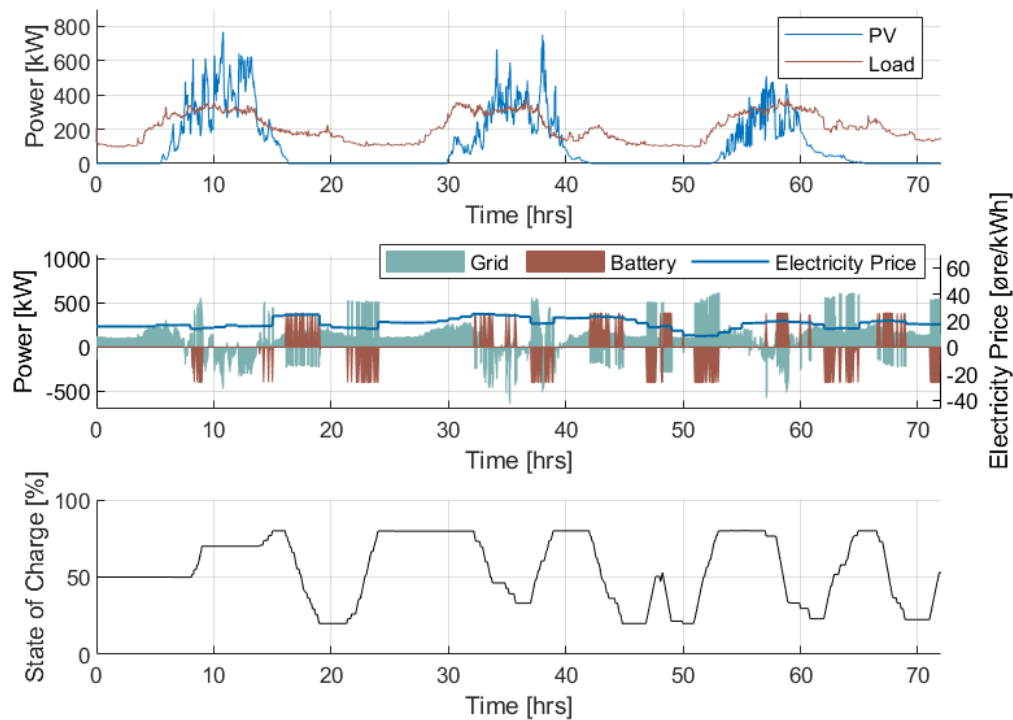
Figure 5.12: Case 3: Results **with** the end of the day SOC constraint for the first three days of the two-month simulation period.

From Figure 5.12, it can be observed that the SOC equals the initial SOC at the end of all three days (hours 24, 48, and 72). The results for the first day are the same as in the reference case because this case met the final SOC constraint in the first place. On the second and the third day, the battery behavior is also similar, except for the charging at the end of the day. To obtain this result, the battery charges without taking the electricity price into account, and the battery is charged at a less ideal time. This also leads to an instant discharge when the next day starts because savings are made from selling the stored energy at the higher electricity price and then recharging the battery with cheaper energy from the grid at hour 51. However, if a lower selling price was considered, the battery would not be instantly emptied at the beginning of day three.

Ultimately, the algorithm does what it is told to do; if it is told to always charge the battery to 50% at the end of the day, it will do so. But then it will also use this energy on the next day to obtain economic benefits. Thus, the number of charging-discharging actions increases, and the battery degradation is escalated. However, it is also possible to implement the SOC constraint differently. For example, it could be expensive to empty the battery at the end of the day. Then, instead of completely draining the battery at

hour 42, it could discharge to 50% and keep this level to the end of the day. This would
also lead to fewer battery cycles than the results in Figure 5.12, which would increase
battery lifetime. However, then the microgrid loses the revenue it could have obtained by
discharging completely at hour 42.

Based on the above discussion, it is expected that the SOH decreases more when the
optimization problem includes the SOC(N) = SOC(0) constraint due to a higher amount of
battery cycles. This expectation is confirmed by the results in Figure 5.13, which depicts
the SOH development for the heuristic case, the MILP-MPC reference case, and case 3 for
the two-month simulation period.
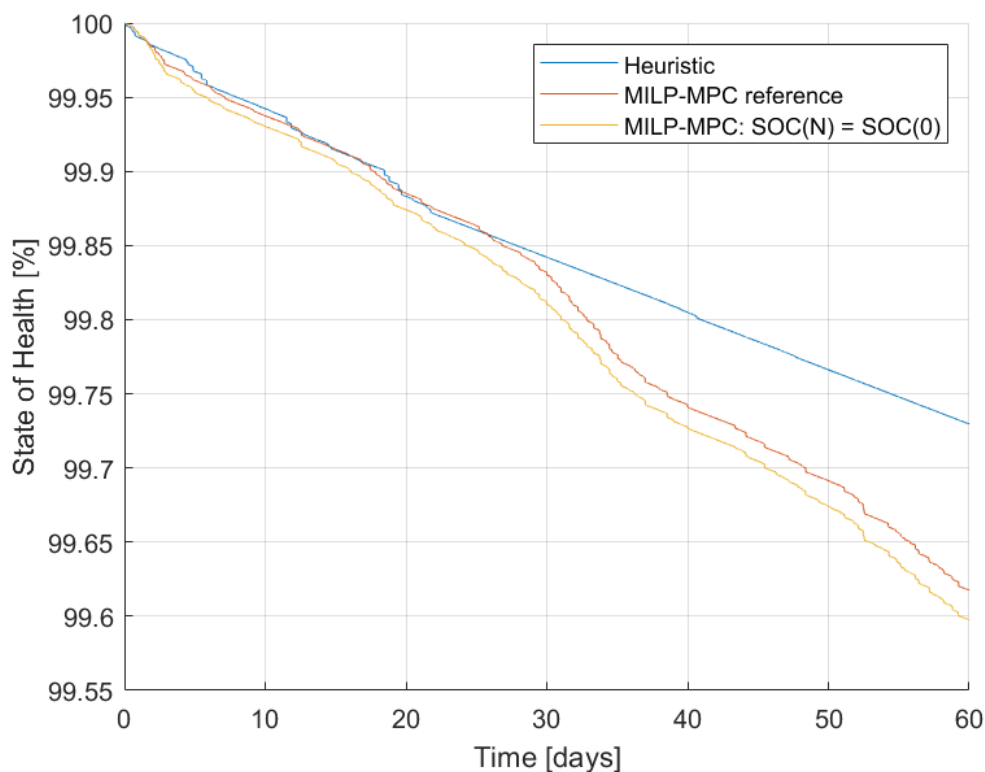
Figure 5.13: Case 3: Battery SOH development for the two-month simulation period.

The total energy costs for each scenario have been averaged over the 60 simulated days
and are presented in Figure 5.14. Expectedly, the energy costs increase when the end
of the day SOC constraint is included because the battery is cycled at less ideal times.
Including this constraint thus results in both higher daily energy costs and reduced battery
lifetime. However, the difference between the two cases is not that big, and both perform
better than the heuristic case.
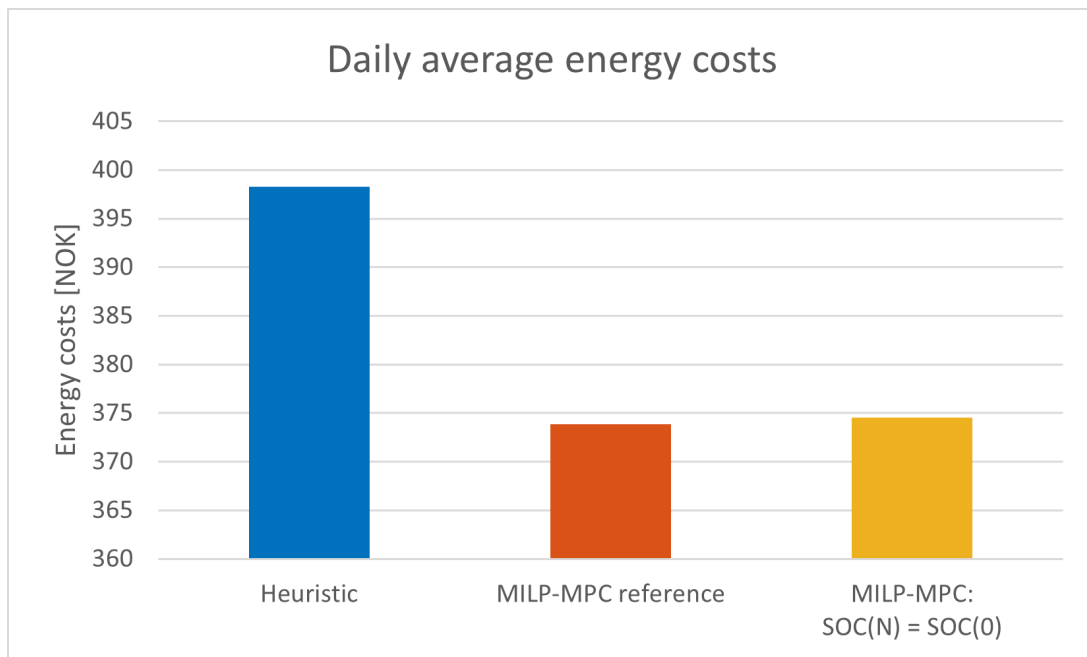
Figure 5.14: Case 3: Daily average energy costs.

Lastly, based on the results obtained in case 3, it can be concluded that the MPC mitigates the negative effects of omitting the end of the day SOC constraint expressed by (4.7). However, it is worth pointing out that this relies on the ability of the forecasting algorithm to predict the PV power production, load demand, and electricity price accurately. If the forecasting is very inaccurate, the unfortunate situation where the battery is empty at a critical time may occur.

5.2.6   Case 4: Impact of prediction horizon

This case tests the influence of the prediction horizon on the MPC, i.e., how far into the future the data is available and will be optimized over at each time step. The prediction horizon is increased from 4 to 24 hours in four different scenarios.

The results presented in Figure 5.15 illustrate how the horizon length affects the controller behavior. By comparing the two subfigures, it can be observed that when a longer prediction horizon is considered, the controller cycles the battery more carefully and tends to store more energy. The scenario with a 24 hours prediction horizon only cycles the battery for significant peaks in the electricity price, as opposed to the scenario with a 4 hours prediction horizon where the battery is cycled even for minor variations. This can be explained by the fact that longer prediction horizons give the controller a larger vision into the future, and it can thus better anticipate future events. Therefore, for long
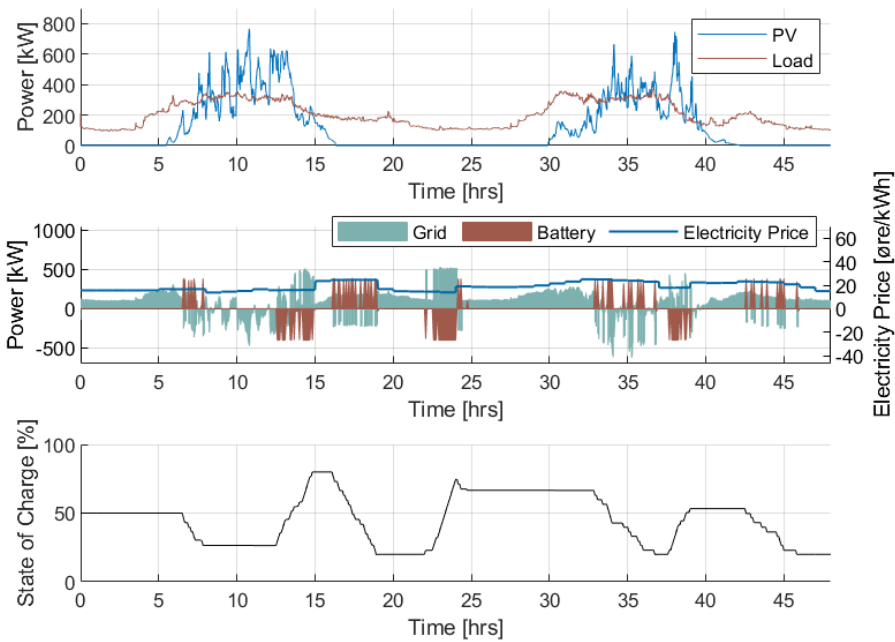
(a) 4 hours prediction horizon.
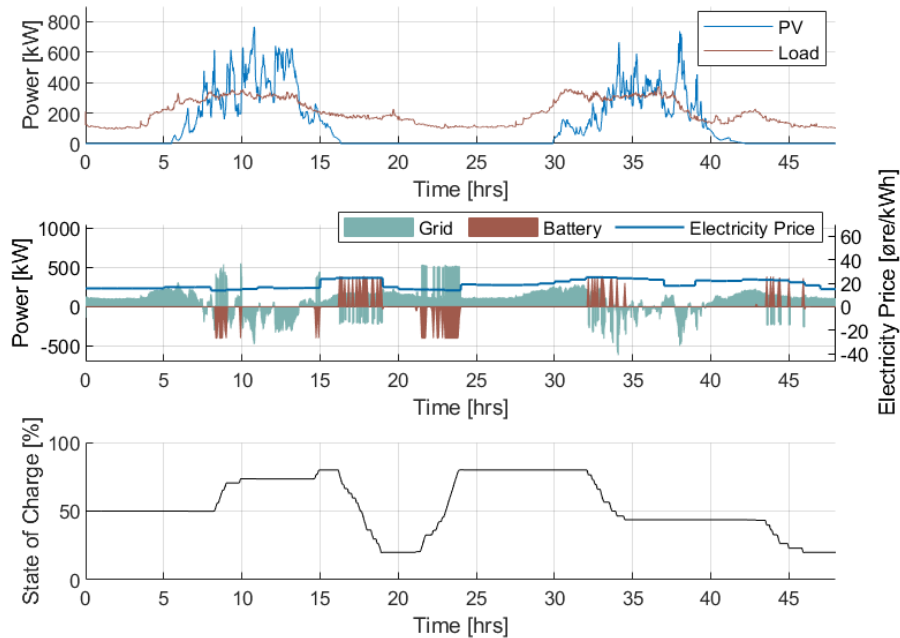
(b) 24 hours prediction horizon.

Figure 5.15: Case 4: Results for different prediction horizons for the first two days of the two-month simulation period.

prediction horizons, the controller will not discharge the battery at hour 6 because it sees a peak in the electricity price at hour 15 and expects to obtain a higher economic benefit by discharging to the grid at this hour instead.

To better compare the scenarios with different prediction horizons, Figure 5.16 depicts the battery SOC for each scenario. These plots substantiate what has already been discussed, namely that the battery is cycled more for shorter prediction horizons. However, for short horizons, the algorithm does not prioritize fully charging and discharging the battery to its minimum and maximum SOC limits of 20% and 80%. Therefore, less energy is exchanged by the battery. In addition, the results in Figure 5.16 show that the SOC profiles are very similar for the scenarios with 8 and 16 hours prediction horizon.
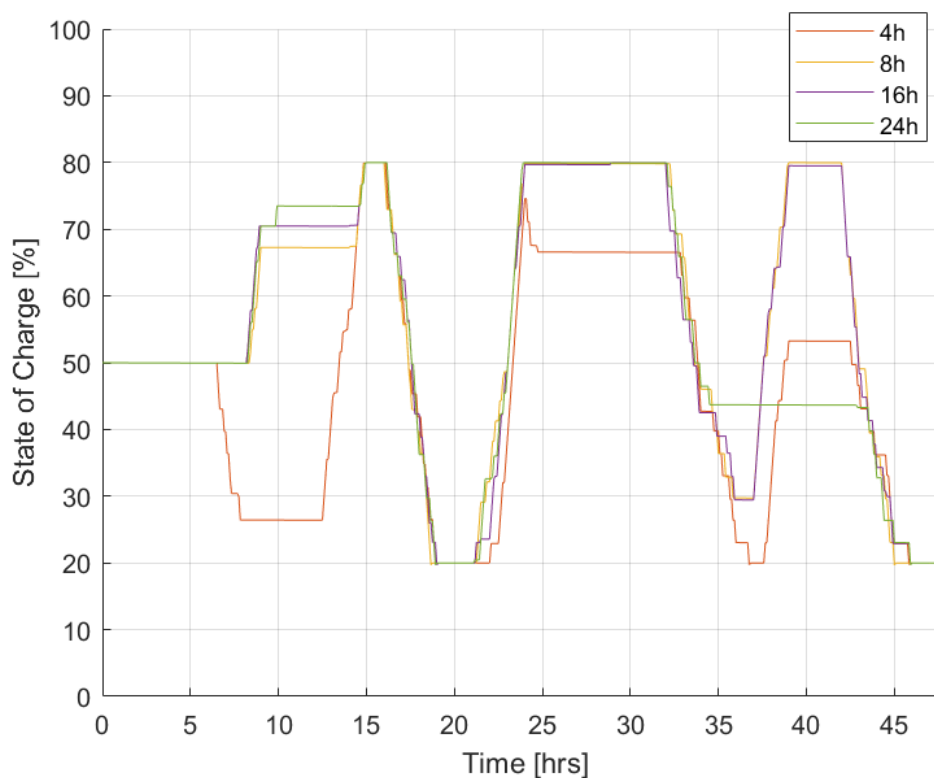
Figure 5.16: Case 4: Battery SOC development for scenarios with different prediction horizon for the first two days of the two-month simulation period.

Since the number of charging-discharging actions is higher when the controller considers a 4 hours horizon length, a natural expectation is that the battery will age quicker for this scenario. However, the results in Figure 5.18 indicate the opposite. A possible explanation is that the battery is rarely charged/discharged to its maximum/minimum value in this scenario. Therefore, the number of *full* cycles is lower, and the battery degrades less.

Furthermore, the results in Figure 5.18 give no clear connection between the horizon length and the SOH development. A shorter prediction horizon does not automatically lead to a longer battery lifetime. The SOH development is also very similar for all scenarios. Therefore, it can be argued that other factors, such as the battery cost, are more important for reducing battery degradation.
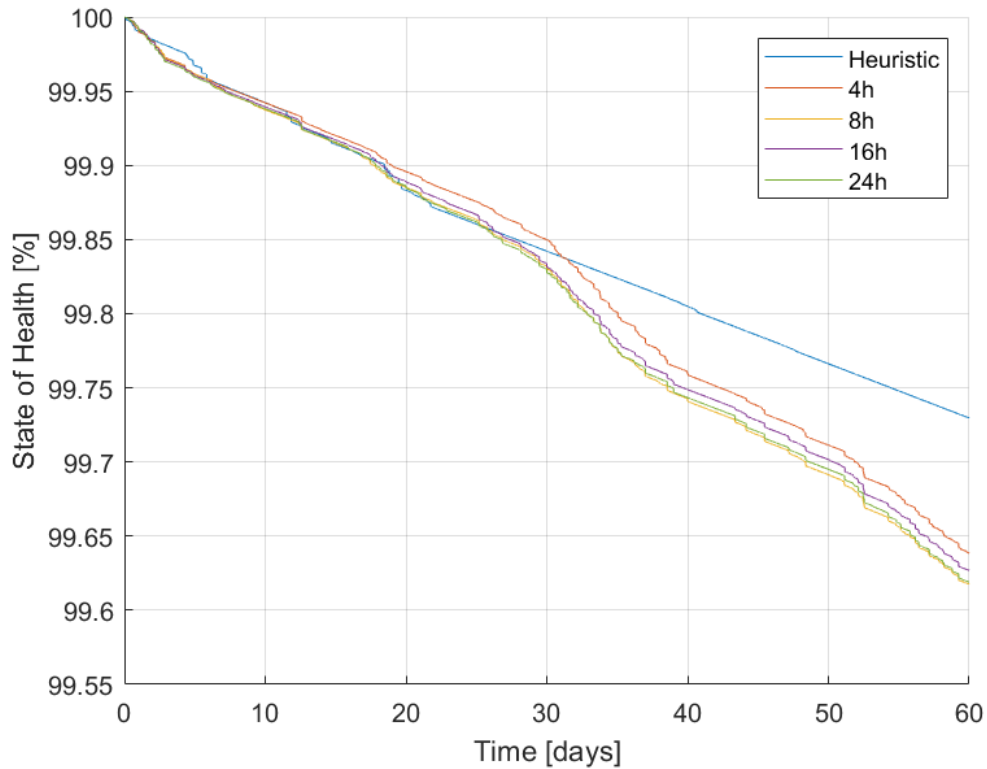
Figure 5.17: Case 4: Battery SOH development for scenarios with different prediction horizon for a two-month simulation period.

The daily average energy costs are presented in Figure 5.18 for all scenarios. As discussed above, longer horizons give the optimization the urge to store electricity for future electricity price peaks instead of directly selling it. The result is lower daily energy costs. However, the cost differences between the scenarios are marginal.
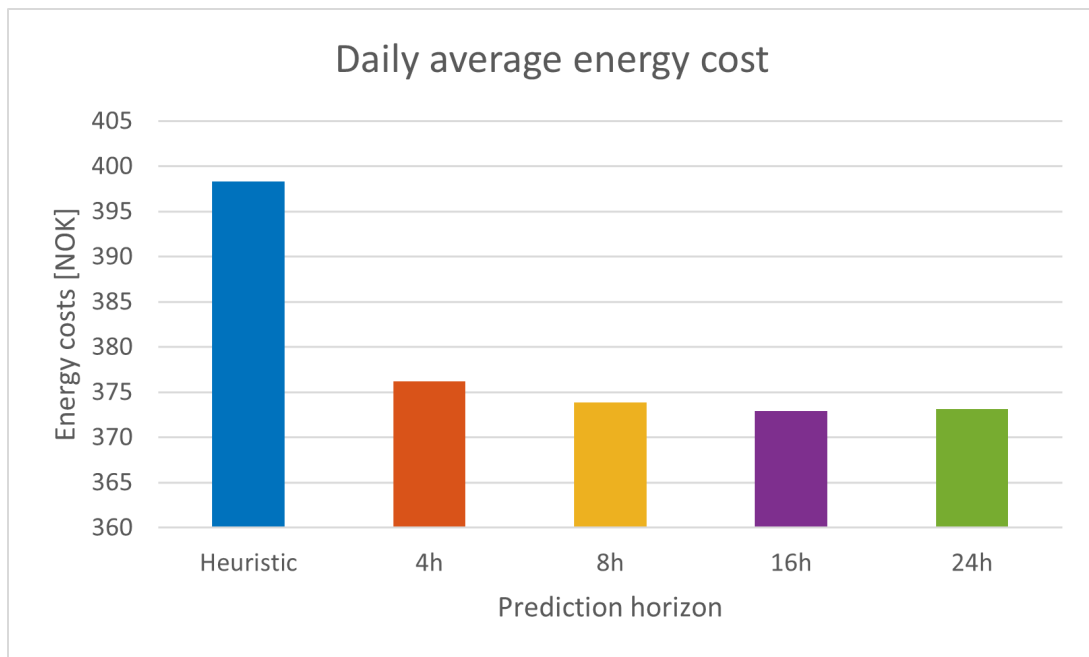
Figure 5.18: Case 4: Daily average energy costs for scenarios with different prediction horizon.

The results of case 4 are not entirely as expected, and it is worth pointing out that the energy management system does not utilize an actual forecasting algorithm for predicting the future PV generation, load demand, and electricity prices. Therefore, the results might not properly capture that the prediction accuracy and, thus, the correctness of the controller operation generally decrease for long prediction horizons. However, the results of case 4 are similar to the results obtained for different prediction horizons in [70], which utilized a similar forecast error function.

5.2.7 Case 5: Impact of forecast accuracy

Case 5 incorporates different forecast accuracies to explore how well the feedback mechanism of the MILP-MPC algorithm handles errors in predicted PV generation, load demand, and electricity price. Four scenarios with increasing forecast error from 0% to 40% were simulated, and the corresponding results for the battery SOC are depicted in Figure 5.19.
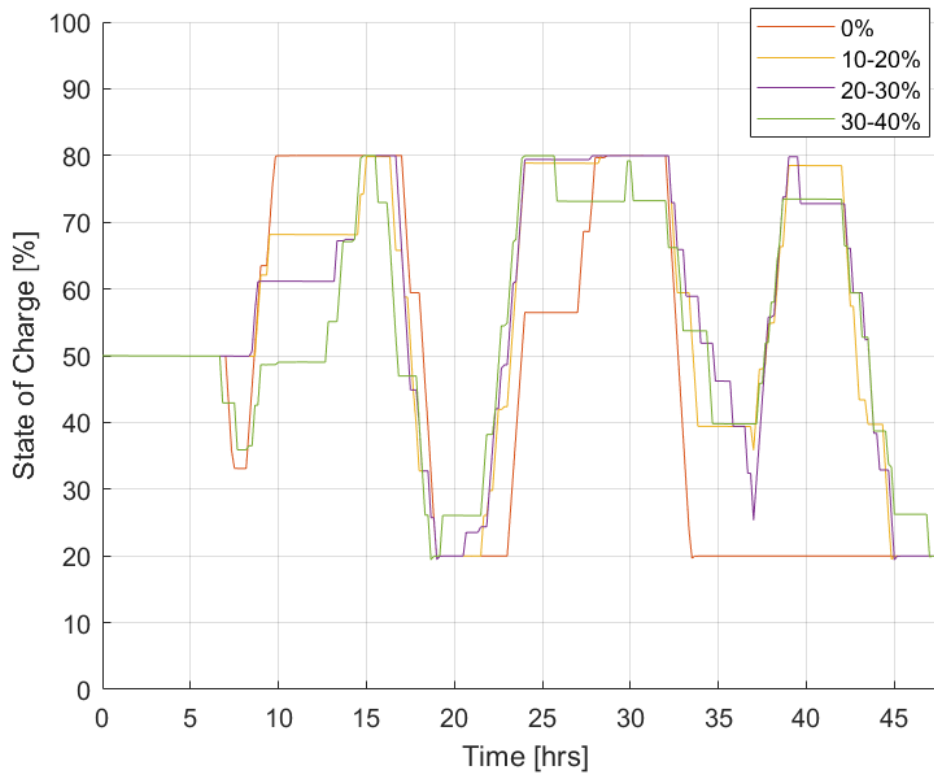
Figure 5.19: Case 5: Battery SOC development for scenarios with different forecast errors for the first two days of the two-month simulation period.

From Figure 5.19 it can be observed that the battery operation changes slightly with increasing forecast errors but that the general form of the SOC curves is maintained. The main difference is the increased ripple in the SOC for higher errors. This occurs because an increased error leads to a more noticeable difference between the predicted and the actual values. The compensating action of the feedback mechanism reduces this difference by quickly charging or discharging the battery, which results in a ripple in the SOC. Moreover, Figure 5.19 shows that the algorithm does not charge the battery at hour 37 for the scenario of perfect forecasting, while it does so for all the other cases. A possible explanation is that the added error makes the electricity price seem lower than it is. Therefore, the algorithm thinks it can save more costs by charging the battery at hour 37 than it actually can.

The development of the battery SOH over two months is depicted in Figure 5.20 for all scenarios. These results show that the SOH develops very similarly for all scenarios, and it can thus be concluded that the MPC can handle forecasting errors in a satisfactory way. It is also worth pointing out that the worst SOH development occurs when the forecasting is perfect. This makes sense because the battery cost is zero, meaning that the algorithm

only considers the grid cost function when optimizing. Consequently, the battery tends to be used more, and since the forecasting is perfect, the battery is fully charged and discharged at all the correct times.
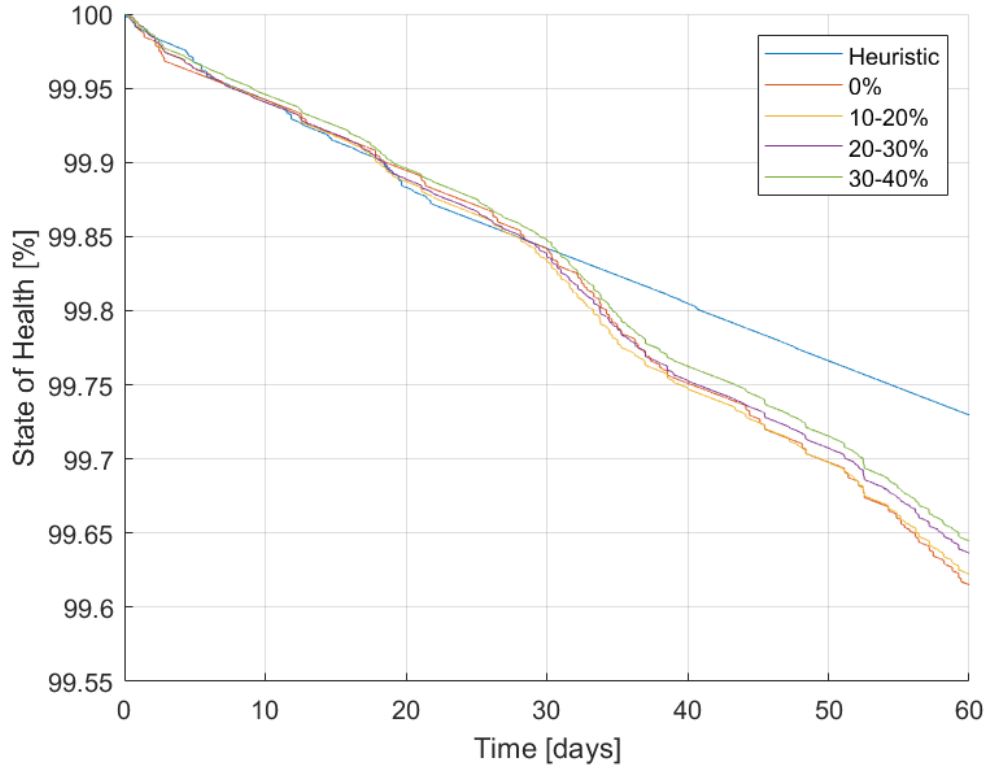
Figure 5.20: Case 5: Battery SOH development for scenarios with different forecast errors for a two-month simulation period.

Figure 5.21 depicts the daily average energy costs for all scenarios. The results show that the costs have no significant change with increasing forecast errors, and that the lowest costs are naturally obtained when the algorithm perfectly sees the development of PV, load and cost profiles. However, perfect forecasting is an unfeasible case that rarely occurs in real cases.
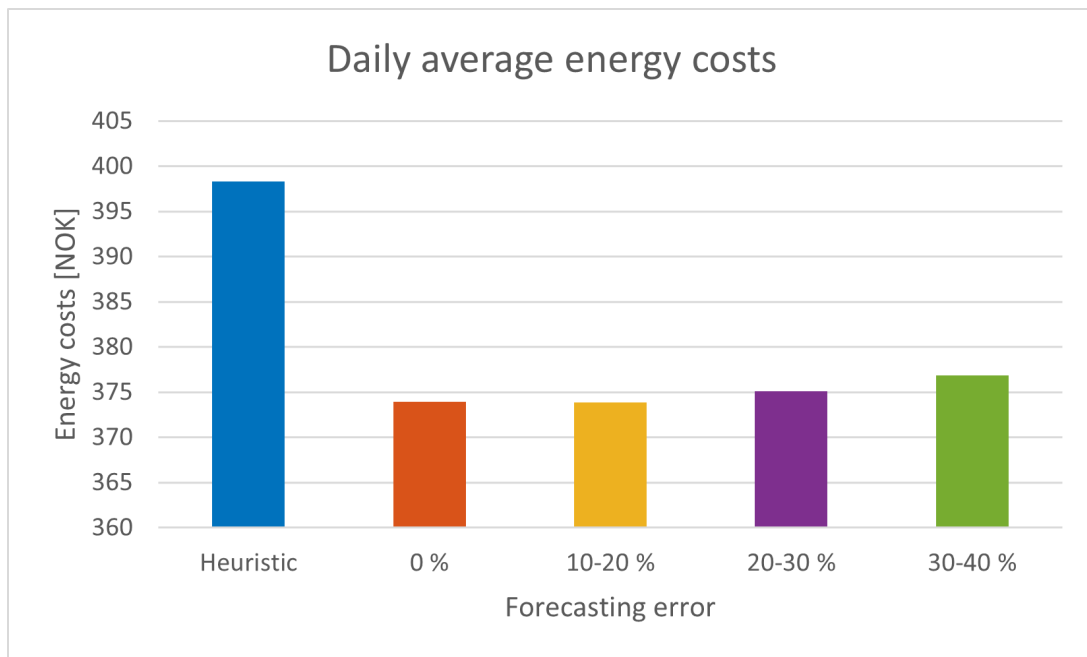
Figure 5.21: Case 5: Energy profit for a month for scenarios with different forecasting errors.

Overall it can be concluded that the model is robust to uncertainty in forecasting. Since the model handles forecast errors as high as 30-40%, it is reasonable to assume that it could also operate satisfactorily when a real forecasting algorithm is implemented. Especially since the short sample time of five minutes allows the algorithm to constantly adjust the battery operation according to the actual values of PV, load, and costs. Implementing a forecasting module could be interesting for future work.

5.3 Microgrid operational limits and computational approach

This chapter has so far investigated the performance of the proposed energy management system for multiple cases. In this section, the effects of this performance on the microgrid will be discussed in terms of voltage, frequency, and operational limits. In addition, the computational approach is evaluated.

5.3.1 Microgrid operational limits

Figure 5.22 depicts the microgrid voltage and frequency for five of the simulated cases with the acceptable $\pm 10\%$ deviation limits drawn in, and Figure 5.23 shows a more zoomed-in

version of the same results to observe how the deviations look like. These figures show that the microgrid voltage and frequency stay within $\pm 10\%$ of their operating values of 400 V and 50 Hz, respectively. Therefore, it can be concluded that the voltage and frequency regulation performed by the main grid is satisfactory and that the developed algorithms affected the frequency and voltage little.
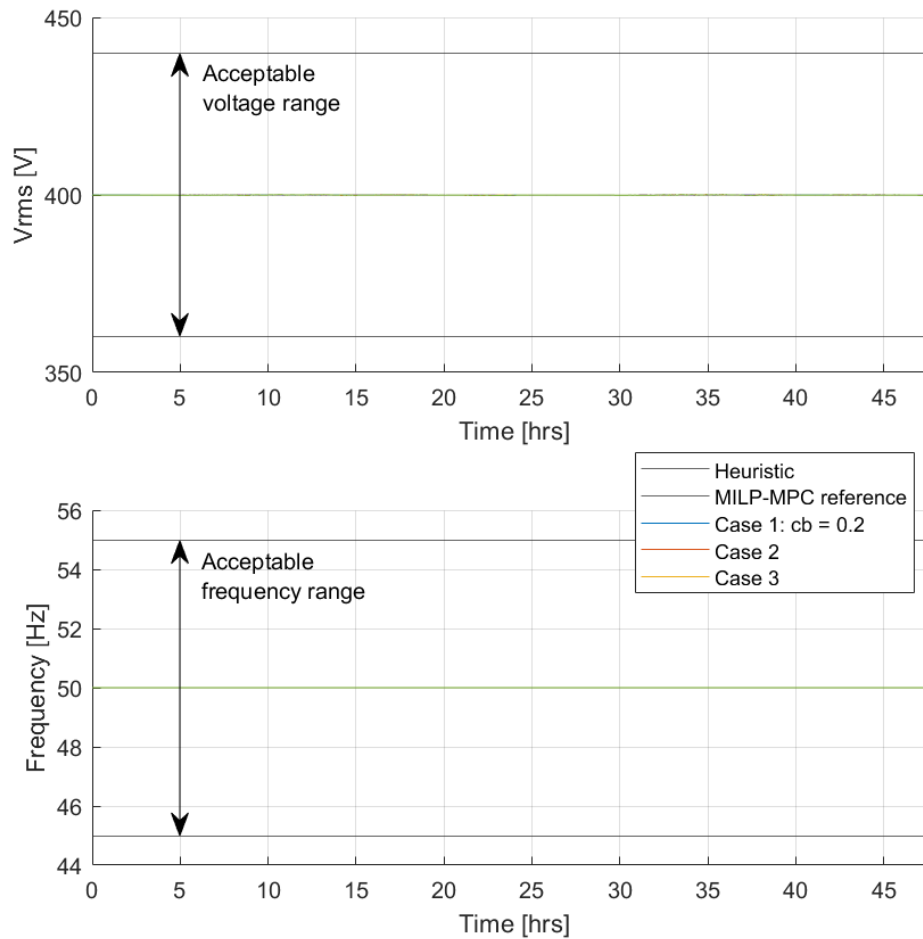
Figure 5.22: Microgrid voltage and frequency for five of the simulation cases with their acceptable ranges of $\pm 10\%$.
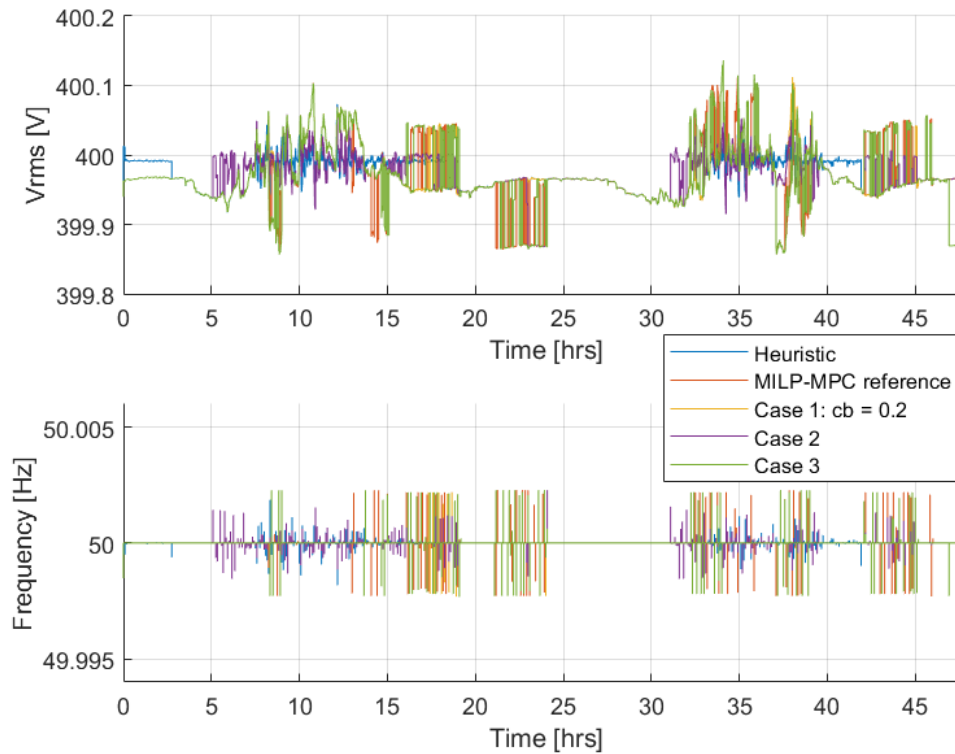
Figure 5.23: Microgrid voltage and frequency for five of the simulation cases zoomed in to capture the minor deviations.

The simulated cases also demonstrated the capability of the developed microgrid model in other areas. For all cases, the battery operated within its limits and behaved as expected. When the battery charged, the SOC increased, and when it discharged, the SOC decreased. In addition, the SOH decreased over the two months at a rate that could be expected from the parameters selected and the resulting battery behavior. Moreover, simultaneous charging and discharging of the battery never occurred. Neither did simultaneous export and import of grid power.

Beyond the above paragraph, the SimSES battery model will not be further verified in this thesis because an extensive verification was conducted by Maik Naumann in his doctoral dissertation [18]. In addition, the SimSES model has been used to analyze various battery fields of application in several publications [19, 20, 21, 22], where it has proven to achieve results with sufficient accuracy for techno-economic considerations within an appropriate simulation time.

5.3.2   Computational approach

The proposed energy management system used mixed-integer linear programming (MILP) to formulate the overall optimization problem. The method can effectively be solved using the *intlinprog* function included in MATLAB's Optimization Toolbox. This function utilizes the branch-and-bound technique for solving the optimization problem by constructing a sequence of subproblems that attempt to converge to a solution of the MILP. This solving technique guarantees that the solution is global once found. However, the method requires both the constraints and the objective function to be linear, which is not compatible with the nonlinear dynamics of the battery capacity and power electronic efficiencies. Therefore, the MILP method was implemented within a model predictive framework to effectively account for uncertainties and to capture some of the nonlinear dynamics of the system by updating the system model every time step. As a result, the developed algorithm proved to work successfully for two months of simulations using real PV, load, and electricity price data that varied a lot over the months. Some days had excess PV power and other days had not, which created an interesting and challenging foundation for the case study.

Moreover, the proposed methodology successfully controlled the microgrid using a short sample time of five minutes. This short sample time enabled the proposed methodology to observe and respond to small changes in the load, generation, and electricity price throughout the day. Even if a short sample time was used, the model had a low execution time. Table 5.6 shows the average execution time for a two-month simulation of the heuristic method and the MILP-MPC method with four different prediction horizons. It can be reasonably observed that a larger prediction horizon increases the computation time. Therefore, the reference case in this thesis utilized a prediction horizon of 8 hours.

However, it must be noted that the execution time will vary slightly from simulation to simulation, even if the same settings are used because it dependents on the overall processor activity. Additionally, the first simulation is always slower than the following ones due to the compilation of the model into runnable code. Nevertheless, the results in Table 5.6 gives an indication of the algorithms' computational effectiveness.

Table 5.6: Execution time of scenarios with different prediction horizon for a two-month simulation period.

Prediction horizon	Computation time
Heuristic	27 sec
4h	3 min, 44 sec
8h	7 min, 12 sec
16h	34 min, 4 sec
24h	1 hour, 43 min, 43 sec

5.4   Summary

This chapter has presented, commented, compared, and discussed the results of seven simulation cases to investigate the performance of the developed MILP-MPC based control method. Depending on the chosen settings, the results showed that the proposed MILP-MPC energy management strategy managed to determine the reference values for the battery power in a way that: (1) minimized the purchased energy during peak times; (2) maximized self-consumption of locally produced PV power; (3) made good use of the battery, keeping it within its limits and reducing its degradation. The result is a flexible algorithm that can be tuned depending on the overall control objective. It is also possible to vary multiple parameters at a time to obtain a trade-off between several objectives. The main effect of tuning each parameter is as follows:

- Battery cost: Introduces battery degradation awareness to the model. As this value increases, the proposed algorithm will perform a lower number of charging-discharging actions.

- Selling price: Affects how the microgrid interacts with the main grid. If a low selling price is selected, the microgrid avoids selling energy to the main grid and self-consumption of PV power is increased.

- End of the day SOC constraint: The inclusion of this constraint ensures that the battery can give the same performance every day. However, it is argued that the constraint is unnecessary when MPC is used.

- Prediction horizon: The horizon length can be chosen depending on the availability and accuracy of forecasts and on the requirements placed on the computational speed.

- Forecasting: The model formulation allows the implementation of a forecasting

module to predict the load demand, PV power, and electricity price. The feedback
mechanism of the algorithm proved to react well to high errors in forecasting accuracy.

Moreover, for all simulation cases, the MILP-MPC algorithm succeeded in reducing the daily
cost of the energy drawn from the main grid when compared to the heuristic algorithm.
This demonstrated the benefit of utilizing an optimization-based energy management
strategy for controlling the microgrid operation.

The control algorithm also affected the microgrid voltage and frequency little for all of
the simulated cases, and it never violated the limits of the microgrid components. In
addition, it successfully used real data to perform a two-month simulation using a five
minutes sample time within a short execution time. Therefore, it can be assumed that the
algorithm would work well in real-time operation.

6 Conclusion and further work

This final chapter presents the main conclusions of the work conducted in this master's thesis, as well as suggestions for further work.

6.1 Conclusion

This thesis concentrated on the implementation of an energy management system (EMS) for a grid-connected microgrid. The overall goal of the thesis was to develop a control approach that combines model predictive control (MPC) with mixed-integer linear programming (MILP) to effectively account for uncertainties and to capture some of the nonlinear dynamics of the system by updating the system model every time step. Four secondary objectives were derivated to obtain the main goal, and the following paragraphs conclude each of these objectives.

First, a literature review on microgrid EMSs was performed to obtain a solid theoretical foundation and to identify the main gaps and challenges. The main part of this literature review included an extensive investigation of methods for solving the energy management optimization problem. Based on this investigation, it was concluded that solving a MILP problem within an MPC framework would be a suitable approach for controlling the microgrid considered in this thesis. In addition to selecting the MILP-MPC approach, the literature review generated guidelines and suggestions for energy management methodologies.

Next, a microgrid simulation platform suitable for testing energy management strategies was implemented in MATLAB/Simulink. This platform was created by modifying and combining two existing models to build a grid-connected microgrid model comprising a PV system, a variable load, a static load, and a battery including a degradation model. Moreover, the model was simulated using a variable-step phasor solving method to enhance simulation speed and to include a sufficient level of detail. A key element of the developed simulation platform was the SimSES battery energy storage system comprising a lithium-ion battery model, a power electronic efficiency model, and a degradation model. This enabled a more detailed and realistic study on how the proposed EMS affected the battery. However, the model only supported the grid-connected mode of operation. Therefore, further work could consider modifying the model to also support island mode.

Further, two energy management strategies were implemented in the developed microgrid simulation platform, namely a high-level optimization algorithm using MPC in combination with MILP and a heuristic management strategy for comparison. The proposed MILP-MPC algorithm utilized predictions of the system's future behavior, including PV, load, and electricity price forecasts, to decide the microgrid operation. Furthermore, a feedback mechanism was introduced (MPC) to compensate for potential forecast errors and uncertainties. In addition, the algorithm took the nonlinearities associated with power electronic efficiencies and battery degradation into account by considering them constant over the prediction horizon. The mathematical formulation of the microgrid optimization problem was straightforward, reproducible, and can be used and enhanced to other microgrids. In conclusion, the resulting energy management strategy was cast as a flexible, multi-objective MILP problem incorporated and solved in an MPC framework to account for uncertainties and to capture some of the nonlinear dynamics of the system.

Finally, the performance of the proposed control approach was investigated through an extensive case study over a two-month simulation period using actual PV and load data from Skagerak Energilab and electricity price profiles from Nordpool. For all cases, the MILP-MPC control algorithm succeeded in reducing the daily cost of the energy drawn from the main grid compared to the heuristic algorithm. Furthermore, depending on the chosen settings, the results showed that the proposed MILP-MPC algorithm managed to determine the reference values for the battery power in a way that: (1) minimized the purchased energy during peak times; (2) maximized self-consumption of locally produced PV power; (3) made good use of the battery, keeping it within its limits and reducing its degradation. Thus, the result is a flexible algorithm that can be tuned depending on the overall control objective. Moreover, the two-month simulation was performed within an appropriate execution time using a short sample time of five minutes, which enables real-time operation. However, no real forecasting algorithm was implemented in the EMS, and the data prediction was performed in a less realistic fashion.

Based on the above paragraphs it can be concluded that this thesis managed to obtain its overall goal, namely to design an efficient energy management system for a grid-connected microgrid by repeatedly solving a mixed-integer linear programming problem within a model predictive framework.

6.2   Further work

There are multiple ways of taking the system proposed in this thesis further. This section will provide some of the proposals for further work.

- *Incorporate a forecasting algorithm in the EMS.* Although the proposed control structure enables the implementation of forecasting algorithms to predict the PV production, load demand, and electricity prices, this thesis did not utilize actual forecasting algorithms. Instead, an error function was developed to simulate uncertainty in forecasts by adding errors to the actual values. Therefore, an essential step in further work would be to implement actual forecasting algorithms in the EMS and investigate how this would affect the results.

- *Experimental verification.* Further work could consider verifying the MILP-MPC control approach experimentally to: (1) ensure that it can be applied in a real system without any difficulties; (2) observe the system response when using a real battery; (3) ensure that the proposed strategy will operate correctly in the presence of a real communication system that can introduce a time-lag in the control signals.

- *Consider grid losses in the optimization.* The proposed MILP-MPC algorithm neglects grid losses, which should be a valid assumption for a small microgrid. However, further work can reformulate the algorithm to include grid constraints and check its effect. An interesting angle to this could be to combine MPC with an optimal power flow approach. It is not clear whether the use of optimal power flow will influence the outcome of an MPC problem drastically.

- *Implement a demand-side management strategy.* Demand-side management is an emerging application in microgrids for exploiting timely interactions between utilities and their customers to improve the reliability and sustainability. Therefore, a way to take the system proposed in this thesis further, could be to implement a demand-side management strategy.

- *Add more controllable units and apply a decentralized approach.* The EMS proposed in this thesis only considers one controllable unit, namely the battery, and it thus utilizes a centralized approach. Further work could include more controllable units, e.g. a controllable load with a demand-side management strategy, and apply a decentralized approach.

- *Further develop the microgrid simulation platform.* The microgrid model could for instance be modified to allow island operation. The adaptation is relatively easy, and can be performed by adding a switch between the utility point-of-connection and the microgrid. In addition, the battery model should be adapted to help the microgrid maintain voltage and frequency when islanded. In this way, an EMS that handles both operational modes can be implemented and tested. However, if island operation is intended for a longer period of time, additional DG units must be added to ensure that the load demand is always met.

References

[1] S. Hovden, "Energy Management in Microgrids - fundamentals, modeling and simulations," Department of Electrical Power Engineering, NTNU – Norwegian University of Science and Technology, Project report in TET4520, Dec. 2020.

[2] G. A. Pagani and M. Aiello, "Towards decentralization: A topological investigation of the medium and low voltage grids," *IEEE Transactions on Smart Grid*, vol. 2, no. 3, pp. 538–547, 2011.

[3] R. Howard, "Power 2.0: The Policy Exchange 'Smart Power' report in seven key charts," [Online, last accessed on 13 Jun 2021]. Available: https://www.businessgreen.com/opinion/3000137/ power-20-the-policy-exchange-smart-power-report-in-seven-key-charts/page/2.

[4] G. Antonova, M. Nardi, A. Scott, and M. Pesin, "Distributed generation and its impact on power grids and microgrids protection," in *2012 65th Annual Conference for Protective Relay Engineers*, 2012, pp. 152–161.

[5] Statnett, Fingrid, Energinet, and Svenska Kraftnät, "The way forward - Solutions for a changing Nordic power system," 2018, [Online, last accessed on 6 Jun 2021]. Available: https://www.statnett.no/globalassets/om-statnett/nyheter-og-pressemeldinger/ the-way-forward---solutions-for-a-changing-nordic-power-system_lowres.pdf.

[6] S. Parhizi, H. Lotfi, A. Khodaei, and S. Bahramirad, "State of the art in research on microgrids: A review," *IEEE Access*, vol. 3, pp. 890–925, 2015.

[7] L. Meng, E. R. Sanseverino, A. Luna, T. Dragicevic, J. C. Vasquez, and J. M. Guerrero, "Microgrid supervisory controllers and energy management systems: A literature review," *Renewable and Sustainable Energy Reviews*, vol. 60, pp. 1263 – 1273, 2016.

[8] Y. Garcia, R. Dufo-López, and J. Bernal-Agustín, "Energy Management in Microgrids with Renewable Energy Sources: A Literature Review," *Applied Sciences*, vol. 9, p. 3854, 09 2019.

[9] C. Bordons, F. Garcia-Torres and M. A. Ridao, *Model Predictive Control of Microgrids*. Springer, 2020.

[10] E. F. Camacho and C. Bordons, *Model Predictive Control.* Springer, 2007.

[11] F. Garcia-Torres, A. Zafra-Cabeza, C. Silva, S. Grieu, T. Darure, and A. Estanqueiro, "Model Predictive Control for Microgrid Functionalities: Review and Future Challenges," *Energies*, vol. 14, p. 1296, 02 2021.

[12] A. Parisio, E. Rikos, and L. Glielmo, "A model predictive control approach to microgrid operation optimization," *IEEE Transactions on Control Systems Technology*, vol. 22, no. 5, pp. 1813–1827, 2014.

[13] P. Malysz, S. Sirouspour, and A. Emadi, "An optimal energy storage control strategy for grid-connected microgrids," *IEEE Transactions on Smart Grid*, vol. 5, no. 4, pp. 1785–1796, 2014.

[14] U. C. Yilmaz, M. E. Sezgin, and M. Gol, "A model predictive control for microgrids considering battery aging," *Journal of Modern Power Systems and Clean Energy*, pp. 1–9, 2019.

[15] J. LeSage, "Microgrid energy management system (ems) using optimization," 2020, [Online, last accessed on 2 Jun 2021]. Available: https://github.com/jonlesage/Microgrid-EMS-Optimization/releases/tag/v19.1.0.

[16] M. Naumann, C. N. Truong, M. Schimpe, D. Kucevic, A. Jossen, and H. C. Hesse, "Simses: Software for techno-economic simulation of stationary energy storage systems," in *International ETG Congress 2017*, 2017, pp. 1–6.

[17] "Skagerak energilab," 2021, [Online, last accessed on 2 Jun 2021]. Available: https://www.skagerakenergilab.no.

[18] M. Naumann, "Techno-economic evaluation of stationary battery energy systems with special consideration of aging," Ph.D. dissertation, The Technical University of Munich (TUM), 2018.

[19] "Lithium-ion battery cost analysis in pv-household application," *Energy Procedia*, vol. 73, pp. 37–47, 2015, 9th International Renewable Energy Storage Conference, IRES 2015.

[20] S. Englberger, A. Jossen, and H. Hesse, "Unlocking the potential of battery storage with the dynamic stacking of multiple applications," *Cell Reports Physical Science*, vol. 1, no. 11, p. 100238, 2020.

[21] D. Kucevic, B. Tepe, S. Englberger, A. Parlikar, M. Mühlbauer, O. Bohlen, A. Jossen, and H. Hesse, "Standard battery energy storage system profiles: Analysis of various applications for stationary energy storage systems using a holistic simulation framework," *Journal of Energy Storage*, vol. 28, p. 101077, 2020.

[22] C. Truong, M. Naumann, R. Karl, M. Müller, A. Jossen, and H. Hesse, "Economics of residential photovoltaic battery systems in germany: The case of tesla's powerwall," *Batteries*, vol. 2, p. 14, 05 2016.

[23] C. Marnay, S. Chatzivasileiadis, C. Abbey, R. Iravani, G. Joos, P. Lombardi, P. Mancarella, and J. von Appen, "Microgrid evolution roadmap," in *2015 International Symposium on Smart Electric Distribution Systems and Technologies (EDST)*, 2015, pp. 139–144.

[24] R. Buchmann, "Harmonic sharing in microgrid applications. modeling, developing and evaluating a microgrid control system with harmonic sharing capability," Master's thesis, Norwegian University of Science and Technology, 2018.

[25] A. Hirsch, Y. Parag, and J. Guerrero, "Microgrids: A review of technologies, key drivers, and outstanding issues," *Renewable and Sustainable Energy Reviews*, vol. 90, pp. 402–411, 2018.

[26] D. E. Olivares, A. Mehrizi-Sani, A. H. Etemadi, C. A. Cañizares, R. Iravani, M. Kazerani, A. H. Hajimiragha, O. Gomis-Bellmunt, M. Saeedifard, R. Palma-Behnke, G. A. Jiménez-Estévez, and N. D. Hatziargyriou, "Trends in microgrid control," *IEEE Transactions on Smart Grid*, vol. 5, no. 4, pp. 1905–1919, 2014.

[27] B. K. W. Kramer, S. Chakraborty and H. Thomas, "Advanced Power Electronic Interfaces for Distributed Energy Systems Part 1: Systems and Topologies," Technical Report NREL/TP-581-42672, Tech. Rep., 2008.

[28] H. Pourbabak, T. Chen, B. Zhang, and W. Su, *Control and Energy Management System in Microgrids*, 05 2017.

[29] N. Truong, U. Jayasinghe, T.-W. Um, and G. M. Lee, "A survey on trust computation in the internet of things," *The Journal of Korean Institute of Communications and Information Sciences (J-KICS)*, vol. 33, pp. 10–27, 01 2016.

[30] M. Yazdanian and A. Mehrizi-Sani, "Distributed control techniques in microgrids," *IEEE Transactions on Smart Grid*, vol. 5, no. 6, pp. 2901–2909, 2014.

[31] F. Katiraei, R. Iravani, N. Hatziargyriou, and A. Dimeas, "Microgrids management," *IEEE Power and Energy Magazine*, vol. 6, no. 3, pp. 54–65, 2008.

[32] A. Bidram and A. Davoudi, "Hierarchical structure of microgrids control system," *IEEE Transactions on Smart Grid*, vol. 3, no. 4, pp. 1963–1976, 2012.

[33] J. M. Guerrero, J. C. Vasquez, J. Matas, L. G. de Vicuna, and M. Castilla, "Hierarchical control of droop-controlled ac and dc microgrids—a general approach toward standardization," *IEEE Transactions on Industrial Electronics*, vol. 58, no. 1, pp. 158–172, 2011.

[34] M. F. Zia, E. Elbouchikhi, and M. Benbouzid, "Microgrids energy management systems: A critical review on methods, solutions, and prospects," *Applied Energy*, vol. 222, pp. 1033 – 1055, 2018.

[35] M. Little, M. Thomson, and D. Infield, "Electrical integration of renewable energy into stand-alone power supplies incorporating hydrogen storage," *International Journal of Hydrogen Energy*, vol. 32, no. 10, pp. 1582–1588, 2007, eHEC2005.

[36] L. Valverde, F. Rosa, A. del Real, A. Arce, and C. Bordons, "Modeling, simulation and experimental set-up of a renewable hydrogen-based domestic microgrid," *International Journal of Hydrogen Energy*, vol. 38, no. 27, pp. 11 672–11 684, 2013.

[37] M. Urbina and Z. Li, "A fuzzy optimization approach to pv/battery scheduling with uncertainty in pv generation," in *2006 38th North American Power Symposium*, 2006, pp. 561–566.

[38] R. S. Sreeleksmi, A. Ashok, and M. G. Nair, "A fuzzy logic controller for energy management in a pv — battery based microgrid system," in *2017 International Conference on Technological Advancements in Power and Energy (TAP Energy)*, 2017, pp. 1–6.

[39] Y. Riffonneau, S. Bacha, F. Barruel, and S. Ploix, "Optimal power flow management for grid connected pv systems with batteries," *IEEE Transactions on Sustainable Energy*, vol. 2, no. 3, pp. 309–320, 2011.

[40] J. Snyman, *Practical mathematical optimization. An introduction to basic optimization theory and classical and new gradient-based algorithms.* Springer, 2005.

[41] I. Serna-Suarez, G. Ordonez, and G. Carrillo-Caicedo, "Microgrid's energy management systems: A survey," 05 2015.

[42] A. Ahmad Khan, M. Naeem, M. Iqbal, S. Qaisar, and A. Anpalagan, "A compendium of optimization objectives, constraints, tools and algorithms for energy management in microgrids," *Renewable and Sustainable Energy Reviews*, vol. 58, pp. 1664 – 1683, 2016.

[43] Y. Zhao, Y. Lu, C. Yan, and S. Wang, "Mpc-based optimal scheduling of grid-connected low energy buildings with thermal energy storages," *Energy and Buildings*, vol. 86, pp. 415–426, 2015.

[44] Y.-H. Chen, S.-Y. Lu, Y.-R. Chang, T.-T. Lee, and M.-C. Hu, "Economic analysis and optimal energy management models for microgrid systems: A case study in taiwan," *Applied Energy*, vol. 103, pp. 145–154, 2013.

[45] J. P Ruiz and I. E Grossmann, "Global optimization of non-convex generalized disjunctive programs: a review on reformulations and relaxation techniques," *Journal of Global Optimization*, vol. 67, pp. 43 – 58, 2017.

[46] Q. Jiang, M. Xue, and G. Geng, "Energy management of microgrid in grid-connected and stand-alone modes," *IEEE Transactions on Power Systems*, vol. 28, no. 3, pp. 3380–3389, 2013.

[47] F. De Angelis, M. Boaro, D. Fuselli, S. Squartini, F. Piazza, and Q. Wei, "Optimal home energy management under dynamic electrical and thermal constraints," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 3, pp. 1518–1527, 2013.

[48] P. P. Vergara, J. C. López, L. C. da Silva, and M. J. Rider, "Security-constrained optimal energy management system for three-phase residential microgrids," *Electric Power Systems Research*, vol. 146, pp. 371–382, 2017.

[49] U. Klanšek, "A comparison between milp and minlp approaches to optimal solution of nonlinear discrete transportation problem," *Transport*, vol. 30, pp. 1–10, 09 2014.

[50] I. Ranaweera and O.-M. Midtgård, "Optimization of operational cost for a grid-supporting pv system with battery storage," *Renewable Energy*, vol. 88, pp. 262–272, 2016.

[51] B. Cheng and W. B. Powell, "Co-optimizing battery storage for the frequency regulation and energy arbitrage using multi-scale dynamic programming," *IEEE Transactions on Smart Grid*, vol. 9, no. 3, pp. 1997–2005, 2018.

[52] C. A. Saldarriaga, R. A. Hincapié, and H. Salazar, "A holistic approach for planning natural gas and electricity distribution networks," *IEEE Transactions on Power Systems*, vol. 28, no. 4, pp. 4052–4063, 2013.

[53] B. Khan and P. Singh, "Selecting a meta-heuristic technique for smart micro-grid optimization problem: A comprehensive analysis," *IEEE Access*, vol. 5, pp. 13 951–13 977, 2017.

[54] S. A. Pourmousavi Kani, M. H. Nehrir, C. Colson, and C. Wang, "Real-time energy management of a stand-alone hybrid wind-microturbine energy system using particle swarm optimization," *Sustainable Energy, IEEE Transactions on*, vol. 1, pp. 193 – 201, 11 2010.

[55] J. Hu, Y. Shan, J. M. Guerrero, A. Ioinovici, K. W. Chan, and J. Rodriguez, "Model predictive control of microgrids – an overview," *Renewable and Sustainable Energy Reviews*, vol. 136, p. 110422, 2021.

[56] F. Berkenkamp and M. Gwerder, "Hybrid model predictive control of stratified thermal storages in buildings," *Energy and Buildings*, vol. 84, pp. 233–240, 2014.

[57] W. Qi, J. Liu, and P. D. Christofides, "Supervisory predictive control for long-term scheduling of an integrated wind/solar energy generation and water desalination system," *IEEE Transactions on Control Systems Technology*, vol. 20, no. 2, pp. 504–512, 2012.

[58] M. Elkazaz, M. Sumner, and D. Thomas, "Energy management system for hybrid pv-wind-battery microgrid using convex programming, model predictive and rolling horizon predictive control with experimental validation," *International Journal of Electrical Power Energy Systems*, vol. 115, p. 105483, 2020.

[59] "powergui - environment block for Simscape Electrical Specialized Power Systems models," [Online, last accessed on 9 Jun 2021]. Available: https://se.mathworks.com/help/physmod/sps/powersys/ref/powergui.html.

[60] "Choose a Solver," [Online, last accessed on 9 Jun 2021]. Available: https://se.mathworks.com/help/simulink/ug/choose-a-solver.html.

[61] "Simscape Electrical. Model and simulate electronic, mechatronic, and electrical power systems." [Online, last accessed on 9 Jun 2021]. Available: https://se.mathworks.com/products/simscape-electrical.html.

[62] "OpenSimSES - MATLAB scripts for the SimSES modeling framework," [Online, last accessed on 9 Jun 2021]. Available: https://bitbucket.org/Team_SES/opensimses/src/master/.

[63] "System design in Simulink using System Objects." [Online, last accessed on 28 Apr 2021]. Available: https://se.mathworks.com/help/simulink/ug/system-design-in-simulink-using-system-objects.html.

[64] D. Azuatalam, K. Paridari, Y. Ma, M. Förstl, A. C. Chapman, and G. Verbič, "Energy management of small-scale pv-battery systems: A systematic review considering practical implementation, computational requirements, quality of input data and battery degradation," *Renewable and Sustainable Energy Reviews*, vol. 112, pp. 555–570, 2019.

[65] H. Hesse, R. Martins, P. Musilek, M. Naumann, C. Truong, and A. Jossen, "Economic optimization of component sizing for residential battery storage systems," *Energies*, vol. 10, p. 835, 06 2017.

[66] P. Malysz, S. Sirouspour, and A. Emadi, "An optimal energy storage control strategy for grid-connected microgrids," *IEEE Transactions on Smart Grid*, vol. 5, no. 4, pp. 1785–1796, 2014.

[67] V. Lakshmanan, "CINELDI-piloter innen fleksibilitet – Skagerak Pilot Case study," 2020.

[68] "How to prolong lithium-based batteries." 2003, [Online, last accessed on 2 Jun 2021]. Available: http://batteryuniversity.com/learn/article/how_to_prolong_lithium_based_batteries.

[69] Nordpool, "Historical Market Data," 2020, [Online, last accessed on 9 Jun 2021]. Available: https://www.nordpoolgroup.com/historical-market-data/.

[70] C. Ju, P. Wang, L. Goel, and Y. Xu, "A two-layer energy management system for microgrids with hybrid energy storage considering degradation costs," *IEEE Transactions on Smart Grid*, vol. 9, no. 6, pp. 6047–6057, 2018.

Appendix

A   Initialization script

This script initializes the microgrid simulation platform and the optimization algorithm. The initialization of the SimSES battery model is adapted from [62] to enable the implementation of the SimSES battery in the Simulink microgrid model.

```matlab
%% Clear workspace
    clc; clear; close all;

%% Import data
days = 60;
data_all = xlsread('pvLoadPriceData_2months');
pvData = data_all(1:1441*days,1);      % 1 = PV from Skagerak
costData = data_all(1:1441*days,3);    % 2 = TOU, 3 = Nordpool elspot
loadData = data_all(1:1441*days,4);    % 4 = Load from Skagerak
time = data_all(1:1441*days,5);        % 5 = Time in seconds



% Global helping variables for conversion
global gvarYEARS2SECONDS gvarDAYS2SECONDS gvarKWH2WS
gvarYEARS2SECONDS  = 3600 * 24 * 365;  % convert between years and seconds
gvarDAYS2SECONDS   = 3600 * 24;        % convert between days and seconds
gvarKWH2WS         = 3600e3;           % convert between kWh and Ws



%% Simulation parameters
% Simulation parameters _inputSim_.
% Sample time, simulated time and logging flags are set here.

inputSim.simStart   = 0 * gvarDAYS2SECONDS; % starting time of simulation [s]
inputSim.simEnd     = days/365 * gvarYEARS2SECONDS; % end time of simulation [s]
inputSim.sampleTime = 5*60;
% sample time of simulation [s]

inputSim.saveResults = false; % TRUE: the variables will be saved in a .mat-file
```

```matlab
inputSim.logAgingResults = false;
% TRUE: all output values of _detectStress_ are logged


%% Technical parameters
% Technical parameters _inputTech_.
% Check fieldnames of variable to set desired values.


inputTech = techParam();  % generates standard struct for required parameters


% Battery data
inputTech.batteryNominalEnergy=1000*gvarKWH2WS; % nominal battery capacity [Ws]
inputTech.powerElectronicsRatedPower = 400e3 ;  % rated battery inverter power [W]
inputTech.startSOC = 0.5;                       % initial SOC at starting step
inputTech.sohCapacityStart = 1;                 % initial SOH of storage capacity
inputTech.agingModelType = 'Lib_Rosenkranz';    % aging model
inputTech.batteryType = 'Lib_Rosenkranz';       % battery type


% Compute load and PV array power output
Ppv_out = pvData;
loadFluc = loadData;
loadBase = 10e3;      % Base load of microgrid [W]


run('createTechParam.m')



%% Optimization parameters
% Optimization parameters _inputOpt_.
% Check fieldnames of variable to set desired values


T_min = inputSim.sampleTime/60; % Optimization time step = simulation time step
stepAdjust = (inputSim.sampleTime)/(time(2)-time(1));
N = (numel(time(1:stepAdjust:end))-1);    % Number of time slots
Np_hour = 8;                              % Prediction horizon in hours
Np = (60*Np_hour)/T_min;                  % Prediction horizon in steps
tvec = (1:N)'*inputSim.sampleTime;


% Adjust data
pvDataOpt = repmat(pvData(2:stepAdjust:end),2,1);
loadDataOpt = repmat(loadData(2:stepAdjust:end),2,1) + loadBase;
costDataOpt = repmat(costData(2:stepAdjust:end),2,1);
```

```matlab
% Forecast error
pv_param = [0.1, 0.2];  % e.g. [0.1, 0.2] gives forecast with 10-20% error
load_param = [0.1, 0.2];
cost_param = [0.1, 0.2];


% Create forecasts
[CostForecast,PpvForecast,PloadForecast] = genForecasts(N,Np,pvDataOpt,
    loadDataOpt,costDataOpt,tvec,pv_param,load_param,cost_param);


% Price signals
inputOpt.price_sell = -1; % If -1 is selected selling price = buying price
inputOpt.battery_weight = 0;


% Battery data for optimization
eta = [0.9 0.9]; % Initial converter charging and discharging efficiency
inputOpt.PERatedPower = inputTech.powerElectronicsRatedPower;
% Rated power for power electronics [W]
inputOpt.SOCinit = inputTech.startSOC;        % Initial SOC
inputOpt.SOCmax = 0.8;                         % Upper limit for SOC
inputOpt.SOCmin = 0.2;                         % Lower limit for SOC
inputOpt.Pb_ch_max = 400e3;                    % Max charging rate [W]
inputOpt.Pb_ch_min = 0;                        % Min charging rate [W]
inputOpt.Pb_dis_max = 400e3;                   % Max discharging rate [W]
inputOpt.Pb_dis_min = 0;                       % Min discharging rate [W]
inputOpt.eta_b = inputTech.etaBatt;            % Battery efficiency
inputOpt.SOHinit = inputTech.sohCapacityStart; % Initial battery SOH


% Grid data for optimization
inputOpt.grid_sell_max = 1000e3;     % Max limit of grid power to sell [W]
inputOpt.grid_sell_min = 0;          % Min limit of grid power to sell [W]
inputOpt.grid_buy_max = 1000e3;      % Max limit of grid power to buy [W]
inputOpt.grid_buy_min = 0;           % Min limit of grid power to buy [W]


%% Clear workspace
clear cost_param load_param pv_param
clear costDataOpt loadDataOpt pvDataOpt
clear data_all i N Np_hour T_min stepAdjust tvec
```

B   SimSES scripts

This section lists all scripts and functions used to simulate the SimSES battery model. All material can be found in [62], and the functions and settings of the battery are initialized in the initialization script.

Folder: @storage_SO (system object of battery model)

- calcAging.m

- calcPowerResidual.m

- characterizeStress.m

- setPowerStorageEquivalentCircuit.m

- setPowerStoragePowerFlow.m

- setReplacement.m

- setupEtaBatt.m

- setupEtaPowerElectronics.m

- storage_SO.m

Folder: @stressCharacterization

- detectHalfCycles.m

- stressCharacterization.m

Folder: 03_TechParameters

- createAgingModel.m

- createBatteryData.m

- createPowerElectronicsData.m

Folder: 05_MiscFunctions

- calcCRateDOC.m

- convtime.m

- returnDataHashCode.m

- subdir.m

- verifyDataHashCode.m

Folder: 06_CoreFunctions

- LiB_Rosenkranz_CalAging.m

- LiB_Rosenkranz_CycAging.m

- callMethodAgingModels_AverageValues.m

- combAgingType_Superposition.m

No folder:

- techParam.m

- addRequiredPaths.m

- createTechParam.m

C Heuristic algorithm

This section includes the heuristic algorithm script.

```matlab
function Pref = fcn(PV, load, SOC)

Pnet = PV-load;
Pdis_max = 400e3;
Pch_max = 400e3;
SOCmin = 0.2;
SOCmax = 0.8;


if Pnet < 0      % if discharging
    if SOC < SOCmin     % lower soc constraint
        Pbatt = 0;
    else
        if abs(Pdis_max) < abs(Pnet)   % battery discharging power constraint
            Pbatt = -Pdis_max;
        else
            Pbatt = Pnet;
```

```matlab
        end
    end
else             % if charging
    if SOC > SOCmax     % upper soc constraiint
        Pbatt = 0;
    else
        if Pnet > Pch_max      % battery charging power constraint
            Pbatt = Pch_max;
        else
            Pbatt = Pnet;
        end
    end
end

Pref = Pbatt;
```

D   Optimization-based algorithm

D.1   Matrix form of the MILP optimization problem

The matrix form of the MILP optimization problem is given by:

$$
\begin{aligned}
\underset{u}{\text{minimize}} \quad & J^T u \\
\text{subject to} \quad & u(intcon) \text{ are integers,} \\
& A_{ineq} \cdot u \leq b_{ineq}, \\
& A_{eq} \cdot u = b_{eq}, \\
& lb \leq u \leq ub
\end{aligned}
\tag{D.1}
$$

where the decision variables are collected in the vector

$$
u(k) = \begin{bmatrix} P_{dis}(k) & P_{ch}(k) & P_{buy}(k) & P_{sell}(k) & \delta_b(k) & \delta_g(k) \end{bmatrix} \text{ for k} = 0, ..., N_p - 1
$$

and the constraint matrices are given below.

Constraint matrices for bounded constraints, i.e., (4.2), (4.8), (4.9), (4.10), (4.12), and (4.13):

$$lb = \begin{bmatrix} P_{dis}^{min}\mathbf{1} \\ P_{ch}^{min}\mathbf{1} \\ P_{buy}^{min}\mathbf{1} \\ P_{sell}^{min}\mathbf{1} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}, \quad ub = \begin{bmatrix} P_{dis}^{max}\mathbf{1} \\ P_{ch}^{max}\mathbf{1} \\ P_{buy}^{max}\mathbf{1} \\ P_{sell}^{max}\mathbf{1} \\ \mathbf{1} \\ \mathbf{1} \end{bmatrix}, \quad \mathbf{1} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

Constraint matrices for the equality constraint, i.e., (4.14):

$$A_{eq} = \begin{bmatrix} \eta_{inv,dis}\mathbf{1} & -\frac{1}{\eta_{inv,ch}} & \mathbf{1} & -\mathbf{1} & \mathbf{0} & \mathbf{0} \end{bmatrix}, \quad b_{eq} = \begin{bmatrix} P_{load} - P_{PV} \end{bmatrix}$$

Constraint matrices for the inequality constraints, i.e., (4.6), (4.8), (4.9), (4.12), and (4.13):

$$A_{ineq} = \begin{bmatrix} 1 & 0 & 0 & 0 & -P_{dis}^{max}\mathbf{1} & 0 \\ -1 & 0 & 0 & 0 & P_{dis}^{min}\mathbf{1} & 0 \\ 0 & 1 & 0 & 0 & P_{ch}^{max}\mathbf{1} & 0 \\ 0 & -1 & 0 & 0 & -P_{ch}^{min}\mathbf{1} & 0 \\ 0 & 0 & 1 & 0 & 0 & -P_{buy}^{max}\mathbf{1} \\ 0 & 0 & -1 & 0 & 0 & P_{buy}^{min}\mathbf{1} \\ 0 & 0 & 0 & 1 & 0 & P_{sell}^{max}\mathbf{1} \\ 0 & 0 & 0 & -1 & 0 & -P_{sell}^{min}\mathbf{1} \\ -\frac{T}{\eta_{dis}}\phi & T\eta_{ch}\phi & 0 & -1 & 0 & 0 \\ \frac{T}{\eta_{dis}}\phi & -T\eta_{ch}\phi & 0 & -1 & 0 & 0 \end{bmatrix}, \quad b_{ineq} = \begin{bmatrix} 0 \\ 0 \\ P_{ch}^{max}\mathbf{1} \\ P_{ch}^{min}\mathbf{1} \\ 0 \\ 0 \\ P_{sell}^{max}\mathbf{1} \\ -P_{sell}^{min}\mathbf{1} \\ (SOC_{max} - SOC)\mathbf{1} \\ (-SOC_{min} + SOC)\mathbf{1} \end{bmatrix}$$

$$\phi = \begin{bmatrix} 1 & 0 & \cdots & 0 & 0 \\ 1 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & & \vdots & \vdots \\ 1 & 1 & \cdots & 1 & 0 \\ 1 & 1 & \cdots & 1 & 1 \end{bmatrix}$$

D.2   MILP optimization script

This subsection includes the MILP optimization script. Reoptimizing this every sample time creates a feedback mechanism (MPC).

```matlab
function [Pgrid,Pbatt,Ebatt,eta] = optimize_microgrid(Np,T,Ppv,Pload,Cost,
    SOC,Ecap_remaining,inputOpt,eta)
% Np = prediction step horizon
% T = time between optimization calls/sample time
% Ppv, Pload, Cost = forecasted data generated by genForecasts.m
% SOC = current state-of-charge of the battery
% Ecap_remaining = current/remaining battery capacity
% inputOpt = struct containing objective data such as constraints,
% objective weights, and costs
% eta = variable power electronic efficiency

% Use current battery capacity to update SOC-limits
Eb_init = SOC*Ecap_remaining;
% SOC at the beginning of the new optimization
inputOpt.Eb_max = inputOpt.SOCmax*Ecap_remaining;  % Upper limit for SOC
inputOpt.Eb_min = inputOpt.SOCmin*Ecap_remaining;  % Lower limit for SOC

% Update power electronic efficiency
inputOpt.efficiency_ch = eta(1); inputOpt.efficiency_dis = eta(2);

% Initialize battery parameters
Ebatt = zeros(Np,1); Ebatt(1) = Eb_init;

% Selling price:
if inputOpt.price_sell == -1
    Cf = Cost;
else
    Cf = inputOpt.price_sell*Cost;
end

% Defining matrices
I = eye(Np,Np);
Z = zeros(Np,Np);
Phi = tril(ones(Np,Np));
```

```matlab
Tau = ones(Np,1);
Zcol = zeros(Np,1);


% Defining matrices for inequality constraints
A_ineq = [I Z Z Z -inputOpt.Pb_dis_max*I Z
          -I Z Z Z inputOpt.Pb_dis_min*I Z
          Z I Z Z inputOpt.Pb_ch_max*I Z
          Z -I Z Z -inputOpt.Pb_ch_min*I Z
          Z Z I Z Z -inputOpt.grid_buy_max*I
          Z Z -I Z Z inputOpt.grid_buy_min*I
          Z Z Z I Z inputOpt.grid_sell_max*I
          Z Z Z -I Z -inputOpt.grid_sell_min*I
          -T*Phi/inputOpt.eta_b T*inputOpt.eta_b*Phi Z Z Z Z
          T*Phi/inputOpt.eta_b -T*inputOpt.eta_b*Phi Z Z Z Z];


b_ineq = [Zcol; Zcol; inputOpt.Pb_ch_max*Tau; -inputOpt.Pb_ch_min*Tau;
          Zcol; Zcol; inputOpt.grid_sell_max*Tau; -inputOpt.grid_sell_min*Tau;
          (inputOpt.Eb_max-Eb_init)*Tau; (-inputOpt.Eb_min+Eb_init)*Tau];


% Defining matrices for equality constraint
A_eq = [inputOpt.efficiency_dis*I -I/inputOpt.efficiency_ch I -I Z Z];
b_eq = [Pload-Ppv];


% Defining matrices for bounded constraints
lb = [inputOpt.Pb_dis_min*Tau;
      inputOpt.Pb_ch_min*Tau;
      inputOpt.grid_buy_min*Tau;
      inputOpt.grid_sell_min*Tau;
      Zcol;
      Zcol];
ub = [inputOpt.Pb_dis_max*Tau;
      inputOpt.Pb_ch_max*Tau;
      inputOpt.grid_buy_max*Tau;
      inputOpt.grid_sell_max*Tau;
      Tau;
      Tau];


% Defining objective function and integers
vec1 = inputOpt.battery_weight*Cost*T;
vec2 = inputOpt.battery_weight*Cost*T;
```

```matlab
vec3 = Cost*T;

vec4 = -Cf*T;

vec5 = Zcol;

vec6 = Zcol;

f = [vec1; vec2; vec3; vec4; vec5; vec6];


% Indicate which decision variables that are integers
intcon = [4*Np+1:5*Np, 5*Np+1:6*Np];


% Solve the MILP optimization problem
options = optimoptions('intlinprog','Display','off');
x = intlinprog(f,intcon,A_ineq,b_ineq,A_eq,b_eq,lb,ub,[],options);


% Execute the control set points by applying the receding horizon principle
Pdis = x(1:Np);             % Battery discharging power
Pch = x(Np+1:2*Np);         % Battery charging power
Pbuy = x(2*Np+1:3*Np);      % Buying power from grid
Psell = x(3*Np+1:4*Np);     % Selling power to grid
delta_b = x(4*Np+1:5*Np);   % Binary variable for battery
delta_g = x(5*Np+1:6*Np);   % Binary variable for grid
Ebatt(2:Np) = Ebatt(1:Np-1)+T*inputOpt.eta_b*Pch(1:Np-1)
    -T*Pdis(1:Np-1)/inputOpt.eta_b;


% Update power electronic efficiency
Rch = Pch(2)/inputOpt.PERatedPower;
eta(1) = Rch/(Rch+0.072+0.0345*Rch^2);
Rdis = Pdis(2)/inputOpt.PERatedPower;
eta(2) = Rdis/(Rdis+0.072+0.0345*Rdis^2);


% Output powers
Pbatt = Pch - Pdis;
Pgrid = Pbuy - Psell;
end
```

D.3 Code generator script

This subsection includes the script necessary for implementing the MILP optimization script
in the Simulink microgrid modeling platform. The code generator script is implemented in
the MATLAB Function block depicted in Figure 4.4.

```matlab
function [optVec,SOCopt,eta]  = fcn(SOC,Np,Ecap_remaining,T,Cost,Ppv,
    Pload,inputOpt,eta)
%#codegen

% Initialize
optVec = 0;
SOCopt = 0;
Pbatt = zeros(Np,1);
Ebatt = zeros(Np,1);

% Declare function (contains intlinprog) as extrinsic
coder.extrinsic('optimize_microgrid');
[~,Pbatt,Ebatt,eta] = optimize_microgrid(Np,T,Ppv,Pload,Cost,SOC,
    Ecap_remaining,inputOpt,eta);

SOCopt = Ebatt(1);
optVec = Pbatt(1);
```

E   Script for generating forecasts

This section includes the error function developed to simulate uncertainty in forecasts by adding errors to the actual values. These errors are modelled with a gradient uncertainty level in which the forecast error increases when the prediction horizon becomes larger.

```matlab
function [CostForecast,PpvForecast,PloadForecast] = genForecasts(N,Np,
    pvData,loadData,costData,tvec,pv_param,load_param,cost_param)
% N = number of steps in optimization
% Np = number of steps in prediction horizon
% pvData, loadData, costData = data prepared with size of N
% tvec = vector containing N time signals
% pv_param, load_param, cost_param = selected forecast error

loadData = loadData';
pvData = pvData';
costData = costData';

% Construct forecast vectors for optimization (N x M) matrix
pv_out = []; load_out = []; cost_out = [];
```

```matlab
p = 1;


for i = 1:N
    % Create a vector with Np random values
    pv_ranV = randn(1,Np);
    load_ranV = randn(1,Np);
    cost_ranV = randn(1,Np);


    % Choose a sequence of Np values
    if (i+(Np-1)) <= N
        pv_seq = pvData(1,i:i+(Np-1));
        load_seq = loadData(1,i:i+(Np-1));
        cost_seq = costData(1,i:i+(Np-1));
    else
        pv_seq = [pvData(1,i:(i+(Np-1)-p)),pvData(1,1:p)];
        load_seq = [loadData(1,i:(i+(Np-1)-p)),loadData(1,1:p)];
        cost_seq = [costData(1,i:(i+(Np-1)-p)),costData(1,1:p)];
        p = p+1;
    end



    pv_sigma=[0 linspace(pv_param(1),pv_param(2),Np-1).*pv_seq(2:Np)];
    load_sigma=[0 linspace(load_param(1),load_param(2),Np-1).*load_seq(2:Np)];
    cost_sigma=[0 linspace(cost_param(1),cost_param(2),Np-1).*cost_seq(2:Np)];


    % Adding the forecasted sequences to the output matrices
    pv_out = [pv_out; pv_seq+pv_ranV.*pv_sigma/2];
    load_out = [load_out; load_seq+load_ranV.*load_sigma/2];


    for j = 1:Np


        if j == 1
            cost_pred(j) = cost_seq(j)+cost_ranV(j)*cost_sigma(j)/2;
        elseif cost_seq(j) == cost_seq(j-1)
            cost_pred(j) = cost_pred(j-1);
        else
            cost_pred(j) = cost_seq(j)+cost_ranV(j)*cost_sigma(j)/2;
        end


    end
```

```matlab
    cost_out = [cost_out; cost_pred];


end


PpvForecast.time = tvec;

PpvForecast.signals.values = pv_out(1:N,:);

PpvForecast.signals.dimensions = Np;


PloadForecast.time = tvec;

PloadForecast.signals.values = load_out(1:N,:);

PloadForecast.signals.dimensions = Np;


CostForecast.time = tvec;

CostForecast.signals.values = cost_out(1:N,:);

CostForecast.signals.dimensions = Np;


end
```

F Script for plotting results

This section includes the script used for plotting the results.

```matlab
%% Adjust data
time_hr = distP.PV.Time./3600;

time_day = time_hr/24;

Cost = gridPrice.Data;

if inputOpt.price_sell == -1

    Cf = 0;

else

    Cf = inputOpt.price_sell*Cost;

end


SOC = ESS_SOC.Data(:);

SOH = ESS_SOH.Data(:);


Ppv = distP.PV.Data*1000;
```

```matlab
Pload = distP.Load.Data*1000;
Pgrid = distP.Grid.Data*1000;
Pbatt = distP.ESS.Data(:)*1000;


%% Plot results
figure(1);
subplot(3,1,1);
hold on
grid
xlim([0,24*days]);
ylim([0,900]);
xlabel('Time [hrs]');
ylabel('Power [kW]');


plot(time_hr,Ppv(:,1)./1e3,'Color','#0072BD');
plot(time_hr,Pload(:,1)./1e3,'Color','#A05A4C');
legend('PV','Load')


% Plot battery and grid power + electricity prices
subplot(3,1,2);
hold on
grid
%ylim([-500,1100]);
xlabel('Time [hrs]');
colororder({'k','k'})


yyaxis left
ylabel('Power [kW]');
xlim([0,24*days]);
ylim([-700,1050]);
agrid = area(time_hr,Pgrid./1e3);
agrid.FaceColor = [0.4863 0.6941 0.6863];
agrid.EdgeColor = [0.4863 0.6941 0.6863];
abatt = area(time_hr,Pbatt./1e3);
abatt.FaceColor = [0.6275 0.3529 0.2980];
abatt.EdgeColor = [0.6275 0.3529 0.2980];


if inputOpt.price_sell == -1
    yyaxis right
    xlim([0,24*days]);
```

```matlab
 ylim([-(700/15),70]);
    ylabel('Electricity Price [ re /kWh]');
    plot(time_hr,Cost(:,1),'Color','#0067a6','LineWidth',1);


    legend ('Grid', 'Battery ','Electricity Price');
else
    yyaxis right
    xlim([0,24*days]);
    ylim([-(700/15),70]);
    ylabel('Electricity Price [ re /kWh]');
    plot(time_hr,Cost(:,1),'Color','#0067a6','LineWidth',1);
    plot(time_hr,Cf,'--','Color','#0067a6','LineWidth',1);
    legend ('Grid', 'Battery ');
    legend ('Grid', 'Battery ','Buying Price','Selling Price');
end


 subplot(3,1,3);
 hold on
 grid
 xlabel('Time [hrs]');
 colororder({'k','k'})
 ylabel('State of Charge [%]');
 xlim([0,24*days]);
 ylim([0,100]);
 plot(time_hr,SOC*100);


 %% Plot SOH
 figure(2);
 hold on
 grid
 xlim([0,days]);
 ylim([99.50,100])
 xlabel('Time [days]');
 ylabel('State of Health [%]');
 plot(time_day,SOH*100);
```