Analysing Dutch Microgrid Performance

A study on the performance of Dutch microgrids under different testing conditions

M.R. van der Werff

Delft University of Technology

Analysing Dutch Microgrid Performance

A study on the performance of Dutch microgrids under different testing conditions

by

M.R. van der Werff

In partial fulfilment of the requirements for the degree of:

Master of Science in Complex Systems Engineering and Management

at the Delft University of Technology,
to be defended publicly on Thursday July 31, 2025 at 01:00 PM.

Graduation committee:	Dr. S.J. Pfenninger-Lee,	TU Delft, Chair
	Dr.ir. I. Nikolic,	TU Delft, Supervisor
	L. Ho PhD,	TU Delft, Advisor
	Dr. A. Meyer,	TU Delft, Advisor

Cover Image: The Link Between Solar Energy and Extreme Weather. From "EcoWatch" by Irma Omerhodzic, 2023 (`https://www.ecowatch.com/solar/solar-extreme-weather`).

An electronic version of this thesis is available at http://repository.tudelft.nl/.

TU Delft

Acknowledgements

In front of you is my master's thesis, the final part in obtaining the degree Master of Science in Complex Systems Engineering and Management (CoSEM) at the TU Delft. This thesis is the result of five months of hard work and dedication. I hope that this thesis can provide a valuable contribution to the future of the Dutch energy system, how small this contribution may be.

This thesis also marks the end of my time as a student. Over the past six years, I have developed myself both academically and personally and I look back on a fantastic time. In my time as a student, I have lived and studied abroad, taken part in one of the most innovative student teams, and enjoyed every moment of it. During this time, my interest in energy systems and the changes needed to make it future-proof has been sparked, ultimately resulting in the topic of this master's thesis.

Without having too much knowledge and experience with coding, this thesis has been challenging to me, but I am proud to present you with this end result. Obviously, this would not have been possible without my daily supervisor, Linh. Her eye for detail, insightful collaboration and inspiring topic suggestions have been absolutely vital to my thesis. Furthermore, I would like to thank my other supervisors Stefan, Igor, and Angela, for their flexibility, critical questions, and thought-provoking discussions, which have been highly useful throughout the process.

Lastly, I would like to thank my friends, family, and girlfriend, as their support and motivation has helped me through the challenges I have faced during this time.

I hope you enjoy reading this thesis and that it gives you a glimpse into the exciting challenges ahead.

M.R. van der Werff
Delft, July 2025

Executive Summary

Net congestion is becoming a more pressing issue in the Netherlands, partly due to the increased penetration of renewable energy generation sources in the Dutch energy mix. The implementation of residential microgrids into the hundreds of planned Energy Hubs in the Netherlands might alleviate the pressure on the main utility grid through local production, consumption and storage. However, when PV systems have period of low production and the BESSs are depleted, the microgrids would all rely on the main utility grid for electricity supply, potentially only extending the limits needed to be reached by the main grid. Conversely, if all microgrids would have a high PV production and storage would be full, electricity might be exported at such a high rate that the main utility grid is challenged heavily. These situations form the basis of the research objective of this thesis.

A literature review has indicated the need for further research on Dutch residential microgrids, while using historical weather condition data for prolonged periods of time. In addition a clear quantitative definition for the performance of the microgrid. Combining the research objective with the knowledge gaps, the following main research question has been formulated:

How does the performance of residential microgrids in the Netherlands vary under different testing conditions?

By varying load patterns, weather condition data longevity, BESS types, and microgrid sizes throughout the various sub-questions, different scenarios have been developed to assess the performance of the microgrid on different performance metrics. The metrics that different scenarios are scored on are: cumulative deficit, import period duration, import and export power, and import and export ramp rates. Taking on the modelling approach has allowed for answering the sub-questions, filling the knowledge gaps, and achieving the research objective. Using the python library GSEE, PV system production has been estimated and compared against different load patterns (household load patterns with gas heating versus household load patterns with heat pump). Weather data for different time periods have been obtained from the European Commission's PVGIS, SARAH-3, and ERA5 datasets.

Simulating the different scenarios has provided insights into the effects of different testing conditions on the performance of the microgrid. The factor with the highest impact on the performance was the load pattern, with the addition of a high-impact load in the form of a heat pump to be the scenarios requiring the highest capacity from the main utility grid. BESS type generally also impact the results, with community batteries proving to be successful in reducing the peak import power and ramp rate when compared to home batteries that are used for individual households. Microgrid size only impacts the performance results in a minimal matter.

Analysing the system over a 42-year period has proven to be highly useful in redetermining the upper limits required to be handled by the microgrid and main utility grid, when compared to the singular year (TMY) scenarios. In all scenarios and performance metrics, analysing the system over this prolonged period of time has given new insights into system boundaries. It would, therefore, be highly recommended for future studies on the performance of (Dutch residential) microgrids to take this multi-decade perspective and prevent underestimation of the limits the microgrid system is subjected to.

These results have largely been validated by existing academic literature, but are still subject to numerous limitations. This limitations include, but are not limited to, missing values in datasets, low temporal resolution, and the exclusion of the role of monetary costs. Further recommendations would be to focus government policies and subsidies mainly on the demand of Dutch households, as electrification in Dutch households is increasing rapidly. High-impact loads, such as a heat pump, drastically increases the maximum burden the main utility grid has to carry and with the slow development of grid expansion, the electricity grid can not keep up with the additional load. In addition, the Dutch home battery market

is still in its infancy stage, but is developing rapidly. To bear the fruits of community batteries, the Dutch government is advised to act quickly and start with the implementation of community batteries in the planned Energy Hubs.

Understanding the interplay between different microgrid components and methods for analysis is vital for successful implementation of the Dutch Energy Hubs and alleviation of the main utility grid. This forms one of the largest challenges of the upcoming decade in the Dutch energy sector. For a full understanding, the results need to be placed in the context of the socio-technical environment. Dutch government instances will need to adjust regulations to incentivise dynamic pricing structures, rethink the cost allocations to allow a fair distribution of costs among households, and setting up a regulatory framework for the emerging Energy Hubs and communities, while also account for behavioural and cultural barriers to smooth implementation of microgrids into the Dutch Energy Hubs. As these Energy Hubs are still in the infancy stage, the Dutch government still has the opportunity to guide the standards, policies, and market mechanisms that will underpin scalable, community-driven Energy Hubs - ensuring they enhance grid stability, foster public trust, and accelerate the transition to a low-carbon energy system.

Contents

List of Figures

List of Tables

Nomenclature

Abbreviations

Abbreviation	Definition
AC	Alternating Current
BESS	Battery Energy Storage System
CDF	Cumulative Distribution Function
COP	Coefficient of Performance
CoSEM	Complex Systems Engineering and Management
DC	Direct Current
DSO	Distribution System Operator
ECMWF	European Centre for Medium-Range Weather Forecasts
ERA5	ECMWF Reanalysis v5
EU	European Union
EV	Electric Vehicle
GHG	Greenhouse Gas
GSEE	Global Solar Energy Estimator
MG	Microgrid
PCP	Parallel Coordinates Plot
PV	Photovoltaic
PVGIS	Photovoltaic Geographical Information System
SARAH-3	Surface Solar Radiation Data Set - Heliosat
TMY	Typical Meteorological Year
TSO	Transmission System Operator

1 Introduction

Following European and National Climate Law, the Netherlands aims to reduce its greenhouse gas (GHG) emissions with 55% by 2030 and achieve carbon neutrality by 2050 [1] [2] [3]. One of the ways to achieve these targets is to increase renewable energy generation. In 2023, 17% of all energy generated in the Netherlands came from renewable sources, a 2% increase from 2022 [4]. Despite this growth, the Netherlands still belongs to the bottom five countries of the EU-27 when it comes to the renewable energy share [5].

Focusing solely on electricity generation, almost half of the total generation in 2023 came from renewable sources, such as solar, wind, and biomass [6]. Total renewable electricity production rose by 21% in 2023, while electricity produced by fossil fuels dropped by 12%. However, due to various factors, the growth of renewable electricity generation in the Netherlands is challenged.

Firstly, due to the fluctuating nature of solar and wind energy, energy generation is not dispatchable [7]. This means that power grid operators can not match demand with electricity supply in a controllable manner. However, these are dependent on weather variability, storage technologies, and dispatchable forms of electricity generation such as biomass, natural gas, or nuclear power plants.

Secondly, the issue of grid congestion is an overarching problem in the Netherlands. In the upcoming years, companies all over the Netherlands cannot be connected to the electricity grid, as it is at full capacity [8]. The Dutch government has implemented several measures to combat grid congestion, such as faster expansion of the grid, better coordination of supply and demand, and stimulating the installation of hybrid heat pumps, instead of all-electric heat pumps [9].

Another way of battling grid congestion is by implementing microgrids. A microgrid is defined as ”a group of interconnected loads and distributed energy resources that acts as a single controllable entity with respect to the grid. It can connect and disconnect from the grid to operate in grid-connected or island mode”[10]. Microgrids help battle grid congestion, as the generated electricity is consumed locally, meaning that the electricity does not have to be transported through transmission and distribution power lines. However, microgrids face the same challenge as any future-proof electricity grid: intermittency of renewable power sources.

This intermittency of renewable energy generation results from variability in weather conditions [11]. If the aim of the microgrid is to be used in island mode, there should be enough storage capacity for when there is insufficient energy generation. One advantage of microgrids is that during periods of severe grid congestion, typically when there is an oversupply of renewable energy, they are likely to generate sufficient electricity locally. However, when grid congestion is less severe, the microgrid may not produce enough to meet demand, requiring electricity from the national grid and potentially worsening congestion.

As of May 2025, Dutch microgrids are still in its infancy stage, but by 2030, there should be hundreds of them, all integrated into *Energy Hubs* [12]. These Energy Hubs could potentially alleviate approximately 3.2GW of the grid's peak power demands [13]. The Dutch government has provided a roadmap for the implementation of such Energy Hubs [14]. However, the roadmap only briefly mentions that future energy generation is dependent on the weather, but does not go into detail on how to deal with it or what the exact effects are, while prolonged periods of adverse weather conditions are highly critical to future microgrid performance.

1.1. Research Objective

So far, the Energy Hubs in the Netherlands are still a work in progress and mostly conceptual. However, with plans to implement hundreds of them by 2030, it is crucial to gain clear insights into the exact impact of weather, among other significant factors, on electricity generation in Dutch microgrids, so that informed decisions can be made regarding their design, efficiency, and integration into the broader energy system. This thesis aims to contribute to the existing knowledge on microgrids and Energy Hubs by answering the following main research question:

"How does the performance of residential microgrids in the Netherlands vary under different testing conditions?"

The research objective is to get a clear insight into the effects of prolonged periods of adverse weather condition on the microgrids' performance, as well as its effects on the overall stability of the Dutch electric power distribution systems. By varying numerous components of the microgrids' system, such as its household size, battery storage system and PV system configurations, this thesis aims to provide knowledge on which components are vital to the performance of a microgrid and how it impacts full systems' grid stability.

1.2. Connection to CoSEM Programme

This thesis is the final part of the MSc. programme Complex Systems Engineering and Management (CoSEM) of the faculty Technology, Policy and Management at the Delft University of Technology. The programme emphasises on designing technological innovations in complex socio-technical environments [15]. This links perfectly well to the contents discussed in this thesis. Firstly, there is the complex part, which indicates that a system or environment can perform differently while the overall inputs are the same.

This is often associated with the social part of the socio-technical system, since human behaviour is complex and cannot be predicted with full certainty. Specifically, this thesis incorporates the aspect of complex human behaviour in the varying load patterns between different households. Another complex element in this thesis is the randomness in weather conditions, which directly influences the production of electricity. The technical aspect of the thesis is evident, as the microgrid with all its components make up for the technical side of the thesis. Altering the design of the microgrid not only influences technical characteristics, but also has its effect on the social and ethical facets. This makes that the research objective of this thesis aligns with the fundamentals of the CoSEM programme.

1.3. Document Outline

An extensive literature review has been performed in chapter 2 to obtain insights into state-of-the-art academic literature, resulting in the formulation of the main research question and sub-questions. The research methods are explained in chapter 3, elaborating on the research approach, datasets, performance metrics, scenario analysis and modelling assumptions made in this thesis. Results and their analysis are shown in chapter 4. Lastly, the results are discussed in chapter 5, while the report ends with the main conclusions in chapter 6.

2
Literature Review

In this chapter, a thorough literature review is conducted, providing definitions of core concepts, knowledge of existing literature, and ultimately the knowledge gap that has not been fulfilled in the current scientific literature, leading to the main research question.

2.1. Core Concepts

Microgrid: A group of interconnected loads and distributed energy resources that acts as a single controllable entity with respect to the grid. It can connect and disconnect from the grid to operate in grid-connected or island mode [10].

Dunkelflaute: A period of multiple consecutive days in which low or minimal energy can be generated by renewable energy sources, such as solar or wind [16].

Typical Metereological Year (TMY): A collation of historical weather data derived from a multi-year time series selected to present the unique weather phenomena with annual averages that are consistent with long term averages [17].

2.2. Literature Review

State-of-the-art academic literature provides a comprehensive overview on the accumulated theoretical knowledge on the relationship between weather conditions and microgrids. The articles can be divided into two categories: resilience and performance. Whereas performance in the context of microgrids relates to everyday operation, ensuring the power quality and grid stability, resilience refers to the ability to bounce back after a large disruption to the microgrid and to maintain power during emergencies or grid failures. This is an important distinction and it has wide-ranging implications for the direction of research in this thesis.

2.2.1. Resilience

The full analysis of the literature review related to the relationship between weather conditions and microgrid resilience can be found in Table B.1. Several key components of the articles are listed in Table B.1, including the study's location, the assumed generation source of the microgrid, and its main findings.

One of the key factors that stands out from this part of the literature review is that many studies are purely hypothetical, not relying on any specific location and its weather data. From the four articles that do study a specific location, three of them look into microgrid resilience of a location in the United States [18] [19] [20]. Moreover, there is not a single study that looks into the weather effect on microgrid resilience in Europe.

Many of the articles focus on severe weather events, such as wildfires and typhoons, rather than periods of prolonged intervals without considerable renewable energy generation, also known as a dunkelflaute [21] [22] [23]. From these articles, the difference between the terms *resilience* and *performance* becomes apparent. The articles that did not focus on severe weather events mostly used solar data from a specific year, with some exceptions ranging from 4-20 years to identify any outliers [24] [20] [19] [18]. Additionally, the articles use different definitions of resilience, as there does not seem to be consensus on how to quantify the concept. The article by Sepúlveda-Mora & Hegedus (2022) even proposes its own definition of resilience in relation to power systems [20]. Often, the focus was put on commercial microgrids rather than residential ones, because of their higher critical load requirements.

Thus, this part of the literature review provides some highly relevant points to this thesis. Firstly, the term resilience often is used in the context of extreme weather events, which is not relevant to this thesis, as such events do not occur in the Netherlands. Another aspect related to the location is that there are no articles that focus specifically on the Netherlands (or even Europe). Moreover, the term resilience is used without a clear definition or consensus on how to quantify it in relation to microgrids. Lastly, the articles mainly explore the resilience of commercial microgrids instead of residential microgrids, leaving yet another theoretical knowledge gap from this part of the literature review.

2.2.2. Performance

As the findings of the literature review about the resilience of microgrids would deviate from the research objective, a second part of the literature review has been focussing on the performance of microgrids. This part relates to everyday fluctuations and grid stability, rather than extreme events. This part of the literature review also focusses solely on residential microgrids. The main findings of this part of the literature review can be found in Table B.2.

One of the first things which stand out in the second part of this literature reivew is how there are more studies which focus on microgrids and its performance in Europe [25] [26] [27]. Even though there still are no Dutch studies on the performance of residential microgrids, these studies already have greater resemblance with the Netherlands than the studies conducted in the previous part of the literature review. Another notable aspect of this part of the review is how many of these studies focus solely on the demand of the microgrid, without taking any form of supply into consideration [28] [25] [26]. Unfortunately, two of these are also European based, thus providing no initial ideas or insights on the supply side for this thesis. The majority of the studies, if specified, examine the microgrid's performance with PV power as (one of) the main electricity supply technology [29] [28][30] [31] [32] [27] [33] [34] [35]. This underlines that PV power is the most common technology for residential microgrid electricity supply.

However, what none of the studies have performed is combining real-life irradiance and load data from the same location to make accurate predictions for the full system's performance for that precise location. Often, only the location's load data or irradiance data is used, but never both [25] [32] [26] [27] [33]. As in the first part of the review, weather data is not used over a long time period and is almost always limited to a single year. Additionally, just as for the term 'resilience' in the first part of the literature review, the term 'performance' is used in multiple contexts and is refined as seems fit for the corresponding study.

2.2.3. Knowledge Gap

The two parts of the literature review leaves several knowledge gaps to be filled with the results of this thesis. Firstly, there is a geographical gap, as none of the studies have been performed with the Netherlands as the (experimental) location. Another aspect that was prevalent during the review is that barely any of the studies make use of weather data with ranges longer than a singular year. From this, the question is raised if looking at prolonged periods of time will impact results and ask greater achievements of the microgrids' performance, as the study will potentially encounter more severe adverse weather conditions. In addition to this, barely any study combined irradiance and load profile of a certain location to create a full location-specific analysis. Lastly, the terms *resilience* and *performance* are used loosely throughout the different articles. Therefore, it is important for this thesis to accurately determine relevant criteria or metrics to rate the system's performance on. These knowledge gaps provide excellent input for the main research question.

2.3. Main Research Question

From the research objective, the weather variability, microgrid performance, and need for better understanding of weather effects in the Netherlands are taken. From the first part of the literature review, the aspects of lack of EU literature, studies on residential microgrids, and some methodological gaps are taken. The second part of the literature review left knowledge gaps on the areas of combining irradiance and load pattern data on a specific location, mostly focusing on load forecasting. Combining the research objective with the theoretical knowledge gaps, a convincing research question can be formulated:

"How does the performance of residential microgrids in the Netherlands vary under different testing conditions?"

Note that in the main research question, the performance of microgrids will be assessed through its effects on grid stability via multiple metrics, which are elaborated on in section 3.5. The different testing conditions that are varied in this research are the weather conditions, type of BESS, microgrid size, and load patterns. These testing conditions are explored in detail the sub-questions in section 2.4.

As the focus of the thesis is on residential microgrids, only solar power generation will be assumed. This is because wind power, in combination with other energy sources, is often only used for commercial or other types of microgrids, due to higher upfront costs and space requirements [36]. The only study conducted on Dutch residential microgrids at the time of writing is the effect of different pricing policies on PV-battery systems [37]. However, the study by Norouzi et al. also incorporates only one year of weather data, thus likely not accounting for any large outliers in weather conditions in this relatively short period.

2.4. Sub-Questions

To be able to answer the main research question, the following sub-questions have been formulated to structure the research process:

1. How do different load profiles influence the performance of Dutch residential microgrids?
2. How does taking into account a prolonged time period of weather conditions influence the performance of Dutch residential microgrids?
3. How do different battery energy storage systems (BESSs) influence the performance of Dutch residential microgrids?
4. How does microgrid size, in number of households, influence the performance of Dutch residential microgrids?

Each of the sub-questions analyses a *testing condition*, that has been presented in the main research question. Answering these sub-questions therefore will lead to a complete and valid answer to the main research question. Further details on the testing conditions can be found in section 3.4.

Sub-question 1 explores the difference that varying load patterns can make on the microgrid's performance. For example, adding a heat pump to each household will create higher levels of electricity consumption. Answering sub-question 2 will provide insights into the large outliers in PV system electricity production when analysing a prolonged period of several decades of weather data, instead of a singular year. This way, periods when the system's stability will be heavily challenged that would not have appeared when working with shorter periods of weather data may potentially be found.

Sub-question 3 dives into the effects of using large community BESSs instead of home batteries, when dealing with large microgrid sizes. Lastly, sub-question 4 provides insights into how the microgrids will be performing for different sizes. This sub-question incorporates the effect of the smoothing of load patterns when dealing with a large number of households in the microgrid, thereby potentially reducing the need of a relatively high storage capacity and (dis)charge power of the BESS.

3
Research Methods

3.1. Research Approach

Revisiting the sub-questions found in section 2.4, a thorough analysis of many different components is required. Therefore, the Modelling Approach is most suited to be applied to this thesis. More specifically, a scenario-based computational model approach is used, based on the given research objective and sub-questions. This section will go into detail as to why this is the most suitable research approach for this thesis.

Firstly, this approach leads to the filling up of the knowledge gaps, as presented in subsection 2.2.3. A model allows for plugging in weather conditions data, such as irradiance and temperature, and linking it to load data to create location-specific performance output. Additionally, a modelling approach also challenges the temporal knowledge gap of periods running for multiple decades, also linking to sub-question 2. A model running scenarios over multiple decades can point out extreme events in which the microgrid is maximally challenged and quantify the impact of weather conditions on microgrid stability.

Another factor as to why the modelling approach is a fitting approach for this thesis is because it allows the combination of weather condition data and load data, which barely has been done in existing academic literature. The two types of data can be combined while modelling a BESS, meaning that an answer to sub-question 3 can be formulated. Because of the modular set up of a computational model, it will be easy to switch between home batteries and community batteries. The same is true for switching between households with and without a heat pump, allowing the formulation of an answer to sub-question 1.

Lastly, the research approach allows for a formulation to answering sub-question 4, by simply increasing the number of households by simulating multiple households with their unique load pattern (and PV system). This will potentially result in smoothing of the total load curve. Additionally, the microgrid's performance need to be quantified by performance metrics when using a computational model. This makes the results replicable, clear, comparable, and insightful.

3.2. Data

To achieve the research objective (section 1.1) using the research approach that has been elaborated on in the previous section, several research methods have been used. Various key elements are needed in order to find answers to the sub-questions from section 2.4:

1. How do different load profiles influence the performance of Dutch residential microgrids?
2. How does taking into account a prolonged time period of weather conditions influence the performance of Dutch residential microgrids?
3. How do different battery energy storage systems (BESSs) influence the performance of Dutch residential microgrids?

4. How does microgrid size, in number of households, influence the performance of Dutch residential microgrids?

Answering sub-question 1 requires distinguishing load patterns across different household types. At least one year of household electricity consumption data is required for finding reliable results using the model. This study distinguishes between households that heat with natural gas and those that generate heat using a heat pump. This means that, next to a real-world household load pattern, the electricity consumption of a heat pump also has been modelled.

To answer sub-question 2, data on weather conditions is needed for two time periods to allow for comparison. More precisely, irradiance data is needed to estimate PV system output, while temperature also has an effect. Additionally, temperature data is needed for estimating the consumption of the heat pump. A common method to estimate weather conditions for a singular average year is the Typical Meteorological Year (TMY). This method uses real-world weather data, but takes the averages for a certain time period (e.g. one month). To check whether running the model over a prolonged time period, a longer period of historical weather condition data is needed.

Sub-question 3 explores the difference between a home battery and community battery. For both, specifications are required to be used as input for the model. The specifications that are used in this study are: energy capacity, (dis)charge power, and round-trip efficiency.

Sub-question 4 can be answered by altering the number of households in the microgrid, to examine whether the *smoothing* of the load pattern has a considerable effect on the overall performance of the microgrid. The exact datasets used in this thesis are described in the following subsections.

3.2.1. Demand

Household Load Pattern

A private household dataset has been used in addition to the public average national load pattern as described below. It entails the data of a household located in Delft over the period of one year (2023). The temporal resolution of the dataset is 15 minutes, and the total yearly consumption is around 4,300 kWh. Approximately the first two weeks of the year have missing values (01-01-2023 00:00 to 13-01-2023 13:45). The missing values have been accounted for in the model.

Due to legal restrictions, this dataset cannot be made public.

National Load Pattern

The yearly national load patterns are taken from the Energy-Charts database [38]. This database does not only contain Dutch households, but the electricity demand of the Netherlands as a whole. Therefore, this dataset is not entirely representative of the load patterns of a Dutch household, but it does provide a valid starting point. It represents the total load of the Netherlands, meaning that any renewable electricity production in the form of PV or wind power has not been subtracted from the initial load. However, it should be said that this is only true for utility-scale PV power plants, as rooftop PV production for households can be consumed directly without it being reported to the Dutch DSO's. For this reason, the load patterns will likely deviate from reality.

An average yearly load pattern is created by finding the average value per time step over all ten years. To convert the total demand of the Netherlands to the load of an average Dutch household, the load values of all time intervals are summed and then multiplied by a multiplication factor so that the total load over the whole year is equal to 4,300 kWh, which is equal to the total yearly load of the household data, and also close to the average annual electricity demand of a detached house in the Dutch province South Holland in 2023 [39].

3.2.2. Weather Conditions

TMY

The Typical Meteorological Year (TMY) data is extracted from the European Commission's PVGIS (Photovoltaic Geographical Information System) [40]. The system uses the SARAH-3 database from the years 2005 to 2023 to the year for each month which is most representative of that specific month. For example, for the coordinates of Delft, the year 2019 is the most representative for January, while the year 2006 is the most representative for February. Combining all the months gives a full TMY. An example of the global irradiance of the TMY in Delft is shown in Figure 3.1.

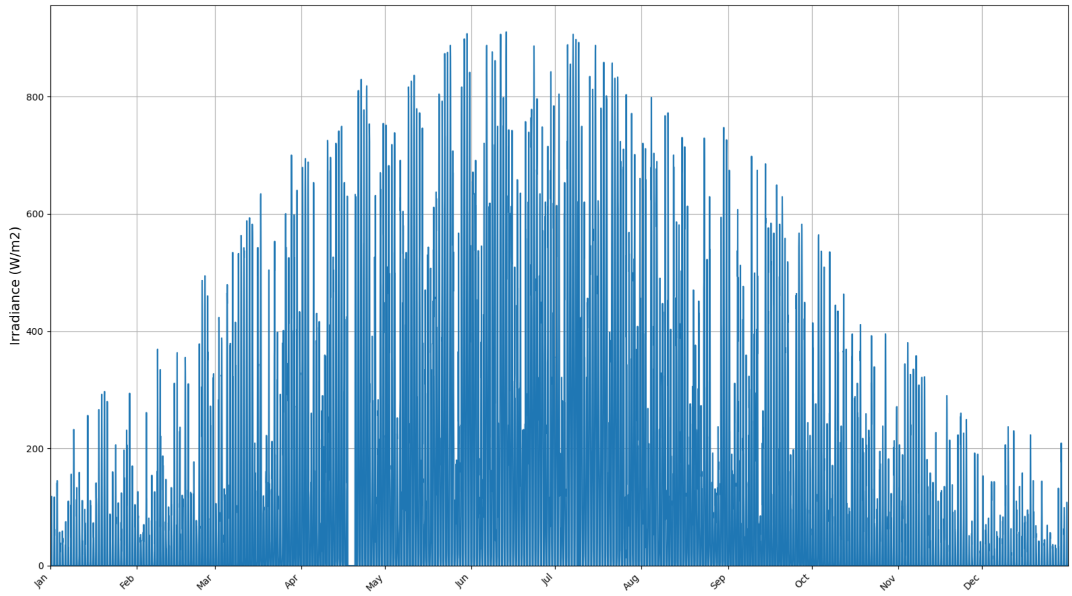

Figure 3.1: Global irradiance during a TMY

The TMY data uses different time periods compared to the demand data, which may lead to some inconsistencies when synthesizing the two datasets. Typically, irradiance is higher on warmer days, while electricity demand tends to rise on colder days, especially when heating is electric (e.g., via heat pumps). On days with lower irradiance, electricity demand would be expected to be higher. However, if the TMY dataset corresponds to a particularly warm day, it will show higher irradiance levels, which could reduce the mismatch (if any) between the datasets, as compared to assuming identical days, times, and years for both.

SARAH-3

The Surface Solar Radiation Data Set - Heliosat (SARAH-3) dataset contains many different products, measured over the period of 01-01-1983 to 31-12-2024 [41]. The products include solar surface irradiance, the surface direct irradiance (direct horizontal and direct normalized), the sunshine duration, the photosynthetically active radiation, daylight, and the effective cloud albedo. These products are measured via satellite-observations of instruments onboard the geostationary Meteosat satellites.

The products are available as monthly and daily means, as well as 30-min instantaneous data. This thesis makes uses of the dataset with a 30-min temporal resolution. The products are available on a latitude/longitude grid with a spatial resolution of 0.05° x 0.05° degrees, which roughly corresponds to 5km x 5km in mid-latitude regions. The three locations (Delft, Groningen, Amsterdam) were taken from Google, and the data points were selected using the nearest method.

From the dataset, the product solar surface irradiance is used for this thesis as an input for the GSEE calculations. The dataset is obtained from the Satellite Application Facility on Climate Monitoring (CM-SAF) [41].

ERA5

ERA5 is the fifth generation of the ECMWF (European Centre for Medium-Range Weather Forecasts) reanalysis for global climate and weather [42]. ERA5 combines model data with observations across the world for a complete dataset. The data is available from 1940 onwards and is updated daily (with a 5-day latency).

The dataset includes more than 200 main variables through different categories, such as atmospheric, ocean-wave and land-surface. The data is available on an hourly basis and has a spatial resolution is 0.25° x 0.25° on a regular latitude/longitude grid, roughly corresponding to 25km x 25km.

The variables that are used for the thesis is the 2m temperature (expressed in degrees of Celsius), as this is used for estimating the output of the PV system. The ERA5 dataset can be obtained from the Climate Data Store, an initiative of four organisations, including the ECMWF [42].

3.3. Model Design

3.3.1. Model Architecture

The metrics described in the previous subsection are calculated using the various datasets as input. Figure 3.2 shows the high-level overview of the model architecture, distinguishing between the data, external python libraries, inner model workings, and the output in the form of performance metrics.

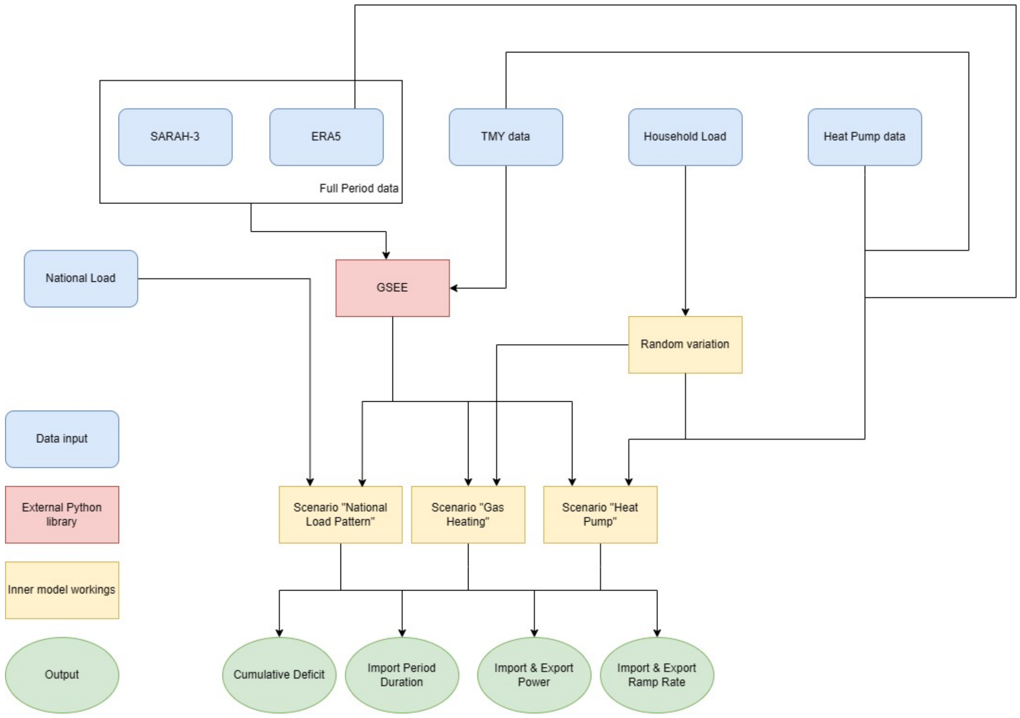

Figure 3.2: Model Architecture and Data Integration

In section 3.2, the exact datasets that have been used in this thesis are explained in detail. An extensive elaboration on the GSEE python library for estimating PV system output, as well as the inner model workings, assumptions, and calculations can be found in section 3.6.

Figure 3.2 also includes the "National Load Pattern" scenario, which is a scenario that has been worked out, but is not part of the main results section in chapter 4. This scenario assumes the aggregated total national load pattern is implemented on a household level, as is described in subsection 3.2.1. Its results can be found in the Appendix, and might serve as a point of reference for implementation on a national scale.

3.3.2. Microgrid Design

The design of the residential microgrid represents a relatively simple system, which is depicted in Figure 3.3. The rooftop PV systems make up for the supply of the microgrid, while the household appliances that consume electricity account for the microgrid's demand. The energy storage (in the form of a BESS) allows for a more balanced and controlled way of importing and exporting electricity from the main grid, which is labelled as *utility grid* in Figure 3.3.

The PV system provides direct current (DC) power, while the BESS also stores and supplies electricity as DC power. The household appliances run on alternating current (AC) power and the main grid provides AC power, too. Note that in the figure, all households own electric vehicles (EVs), but that

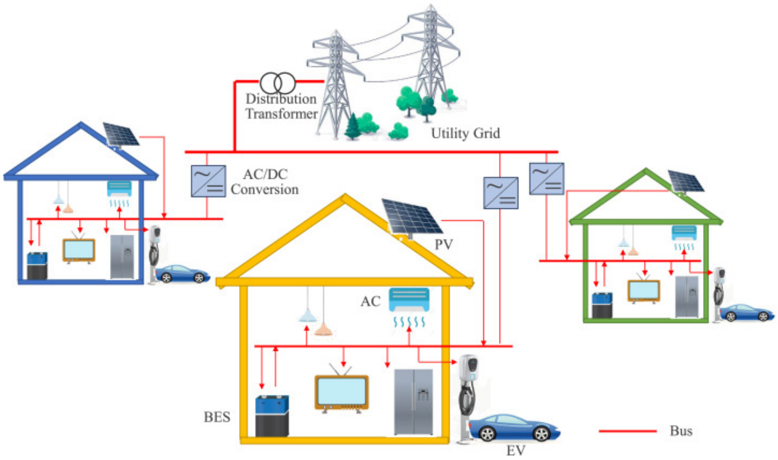

Figure 3.3: Residential Microgrid Design [43]

is not the case in the scenarios in this thesis. Also, the individual home batteries are swapped with community batteries in some of the scenarios.

3.4. Testing Conditions & Scenarios

Using the sub-questions, several testing conditions have been determined to test different microgrid configurations and components to reassess the microgrids' performance and find any potential differences across the scenarios, which are discussed in the upcoming section. Each of the sub-questions is represented by one of the testing conditions, listed below.

3.4.1. Load Patterns

The two different load patterns that are used are the national and household load patterns. The national has a way smoother distribution, with very low peak demand values, but also a higher baseload. The household load pattern, on the contrary, follows a pattern of a relatively low base load, but with much higher spikes in peak demand, resulting in different ways the microgrid needs to satisfy the load.

3.4.2. Period Duration

The TMY, as the name suggests, provides data for one year, from which the irradiance and temperature values can be obtained as input for the model [40]. The data is used to estimate PV system output and heat pump electricity consumption.

The TMY will be compared to a period of 42 years (1983-2024) of weather data, from here on out referred to as ”full period”. The comparison will point out if using a prolonged period of weather data does influence the results of the model, and thus is worth using. Additionally, these tests will point out which weather conditions can exacerbate grid congestion.

3.4.3. Battery Type

An alternative to home batteries for each individual household could be a community battery (also known as neighbourhood battery). This battery (or stack of batteries) will be shared with the whole neighbourhood, giving it the potential to be smaller in capacity than the individual home batteries combined, benefitting from economies of scale. Community batteries provide many advantages, such as reduced costs, higher resource efficiency, and improved collaboration [44].

However, the power output of such a battery (pack) could be a bottleneck in the systems' resilience, as a lot of households would require electricity simultaneously during periods of adverse weather conditions. The specifications for the home battery and community battery can be found in Table 3.1 and Table 3.2, respectively.

3.4.4. Microgrid Size

Varying the number of households in the microgrid might result in different conclusions in terms of performance. Therefore, household number will be altered to create different scenarios. The number of households in the benchmark model will be vary between 30 and 300 households to create a ‘small microgrid’ and ‘large microgrid’.

The Netherlands has a total of 14,574 neighbourhoods [45]. With a total of 8,374,404 households in the Netherlands, it means that there are almost 575 households per neighbourhood in the Netherlands, on average. However, this does give a distorted view, as there are 50 neighbourhoods with more than 5,000 households, 5 of which even have more than 10,000 households. These neighbourhoods drastically increase the average. The median of the dataset is 315 households per neighbourhood, which provides a more accurate metric.

30 households in a microgrid is another realistic number, as 2152 neighbourhoods (14.8%) have 30 or less households in the Netherlands [45]. Therefore, varying the microgrid sizes to 30 and 300 households provides valuable and realistic insights into the Dutch microgrid implementation potential.

3.4.5. Scenarios

Using the testing conditions described above, multiple scenarios have been developed. Firstly, a distinction is made between the type of heating used for the households, which has an effect on its load patterns. The first scenario assumes households to be heated by natural gas, meaning that there is no additional electricity consumed. The second scenario assumes households to be heated by a heat pump, which does add to the electricity consumption of households. For each of these types of load patterns, eight distinct scenarios have been developed based on the testing conditions:

- `tmy_homebattery_small`: TMY weather conditions with home battery with small microgrid size
- `tmy_communitybattery_small`: TMY weather conditions with community battery with small microgrid size
- `tmy_homebattery_large`: TMY weather conditions with home battery with large microgrid size
- `tmy_communitybattery_large`: TMY weather conditions with community battery with large microgrid size
- `fullperiod_homebattery_small`: Full period weather conditions with home battery with small microgrid size
- `fullperiod_communitybattery_small`: Full period weather conditions with community battery with small microgrid size
- `fullperiod_homebattery_large`: Full period weather conditions with home battery with large microgrid size
- `fullperiod_communitybattery_large`: Full period weather conditions with community battery with large microgrid size

Each of these scenarios will be simulated for microgrid households with two different load patterns (gas heating and heat pump) in order to formulate answers to the sub-questions.

3.5. Performance Metrics

The system will be evaluated through several performance metrics, each vital for understanding the effect on overall performance of the system. The following metrics are used in assessing the performance of the system: peak power, ramp rate, import duration period, and cumulative deficit. Below, they are explained in detail.

The first resilience metric is the peak power for one time interval (t) that is required from the main grid [46]. The choice for this metric relates heavily to the grid congestion issue in the Netherlands, where peak power withdrawn from (or fed into) the main grid causes increased chances of outages and voltage instabilities. This metric is relevant for both importing from and exporting to the main utility grid, hence both will be used for further analysis.

The peak power is closely related to the ramp rate, which is the change in power output at a given time step [47]. An appropriate balance between electricity supply and demand prevents harmful effects on the transmission system of both main grid and microgrid that can result from sudden, excessive fluctuations [48]. Both the peak power and ramp rate are effective metrics through their simplicity and ease of interpretation and provide a strong initial insight into the systems' performance.

Furthermore, the systems' resilience will be analysed based on the longest period of continuous import needed from the main utility grid. This is the time where the supply of the PV system and battery storage is insufficient to meet demand, resulting in a positive residual load. The final performance metric is the total electricity needed during one continuous period, also known as the cumulative deficit. Thus, the following resilience metrics are used in this thesis:

- Peak power [kW]
- Ramp rate [kW / time step]
- Continuous period of main grid import [hours]
- Cumulative deficit [kWh]

Each of the metrics will be expressed per microgrid household to ensure fair comparison, as there is variation in the number of households between the different scenarios. The maximum value of the metrics holds the most important information, as this is the indication of the boundaries needed for the whole system not to disintegrate and black out. In addition, a value at risk (VaR) is given for each of the metrics, which is set at the 99th percentile of the full distribution.

3.6. Model Assumptions & Calculations

To assess the microgrid's performance, a model has been developed, following the research approach. By running different scenarios, the differences between load patterns and other testing conditions can be found. All the calculations and modelling have been done using the programming language Python [49]. The exact location of the microgrid system is in Delft (latitude: 52.025, longitude: 4.375). The model simulates with 15 minute time intervals, which is the highest temporal resolution of any dataset, described in section 3.2. A higher temporal resolution would mean that computational power and run time of the scenarios would increase considerably. Below the different components of the model are described. The full python code can be found in Appendix A.

3.6.1. Demand

Appliances

To create random variation in the private household dataset for every household, a randomized variant of the original household load pattern is generated, where the demand value at each time step can change between -10% and +10% scaling factor. Additionally, to account for load diversity when dealing with a high number of households, each load value is shifted randomly anywhere between -2 hours and +2 hours. This results in modest changes in peak demand times for different households, and thus a more realistic model.

Figure 3.4 shows an example for 10 households displaying the effect of time shifting and the scaling factor across four days.

Heat Pump

Power consumption of the heat pump is calculated using a few basic equations. One of the inputs is the temperature data of either the TMY or ERA5 datasets, which are elaborated on in section 3.2. Firstly, the temperature difference is calculated using Equation 3.1, being the difference between the set indoor temperature and the ambient temperature (obtained from either the TMY or ERA5 dataset). The set temperature of the household is 18°C, which is the average indoor temperature in Dutch households in 2022-2023 [50].

$$\Delta T = \max(T_{\text{set}} - T_{\text{amb}},\ 0) \tag{3.1}$$

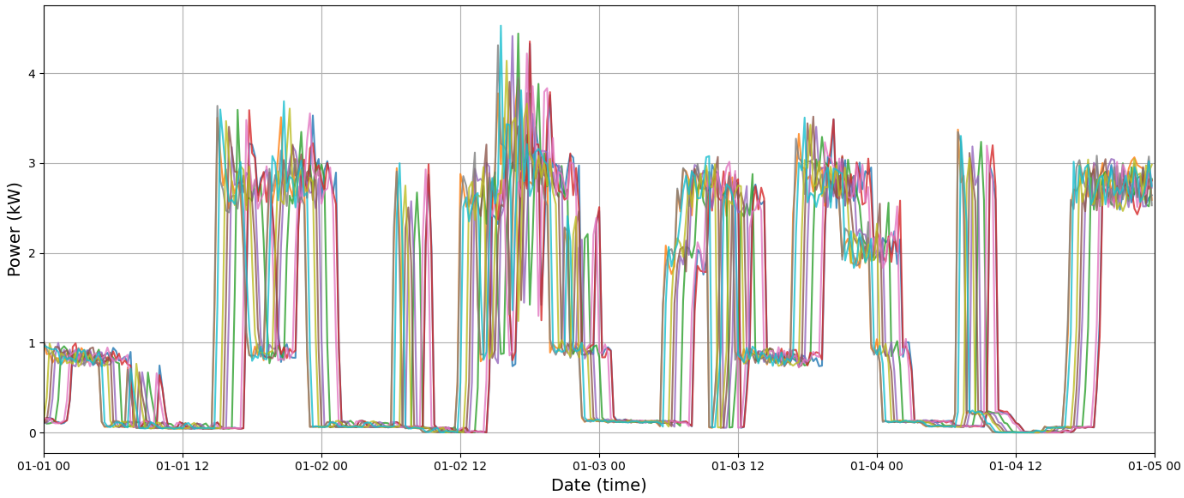

Figure 3.4: Randomized Load Pattern for 10 Households (January 1st - January 4th)

Subsequently, the heating demand (in kW) is calculated using Equation 3.2, using the ΔT and the heat loss coefficient K_{building}, which is the inverse of the R_c value. In current practice, an R_c value of 3.5-5 is advised, with a value of 3.5 already providing a good level of insulation [51] [52]. From 1992 all buildings were required to have a minimal R_c value of 2.5 [53]. For this thesis, an R_c value of 3.5 is taken, resulting in a K_{building} value of 0.285 kW / °C.

$$Q_{\text{demand}} = K_{\text{building}} \cdot \Delta T \tag{3.2}$$

The Coefficient of Performance (COP) of the heat pump is calculated using Equation 3.3. The constant is 3.5, with the performance dropping by 2% of the constant (= - 0.07 / °C) with an increasing temperature [54] [55].

$$\text{COP} = \max(a_{\text{cop}} - b_{\text{cop}} \cdot T_{\text{amb}},\ 1.5) \tag{3.3}$$

Finally, the heat pump power consumption can be calculated as the minimum of the maximum power output of the heat pump and the Q_{demand} divided by the COP, as is shown in Equation 3.4. Average heat pumps in the Netherlands have a maximum power output of 5 - 10kW [56]. For this thesis a $P_{\text{HP,max}}$ of 7.5kW is assumed.

$$P_{\text{HP}} = \min\left(P_{\text{HP, max}}, \frac{Q_{\text{demand}}}{\text{COP}}\right) \tag{3.4}$$

As an example, the electricity consumption of the heat pump in 2024 is shown in Figure 3.5. The peak load is 1.68kW, while the total consumption in 2024 equals 5,520 kWh.

3.6.2. Supply

GSEE

The GSEE library builds on pvlib python, which is a “community developed toolbox that provides a set of functions and classes for simulating the performance of photovoltaic energy systems and accomplishing related tasks” [57]. GSEE is designed for quicker calculations and generally higher ease of use [58].

GSEE uses irradiance data, as well as location data - latitude and longitude - to simulate the output of the PV system. Additionally, the user can provide the specifications of the PV system itself to tweak results. These specifications include the tilt, tracking, and capacity of the PV system.

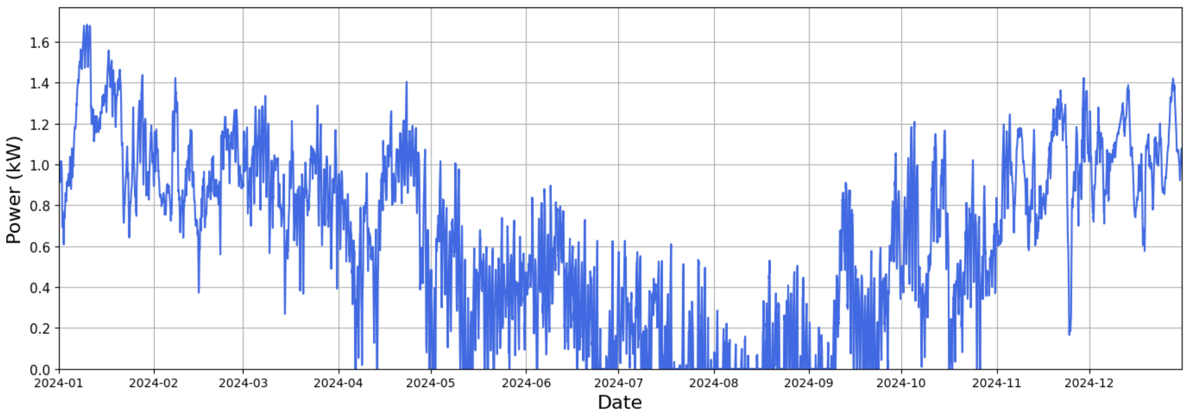

Figure 3.5: Electricity Consumption of Heat Pump (2024)

The irradiance values of the SARAH-3 dataset provide an input for the GSEE library. In the model, an installed capacity of the PV system of 4,000W is assumed, which roughly equals an amount of ten to sixteen solar panels [59]. Implementing all the parameters discussed gives a supply pattern shown in Figure 3.6.

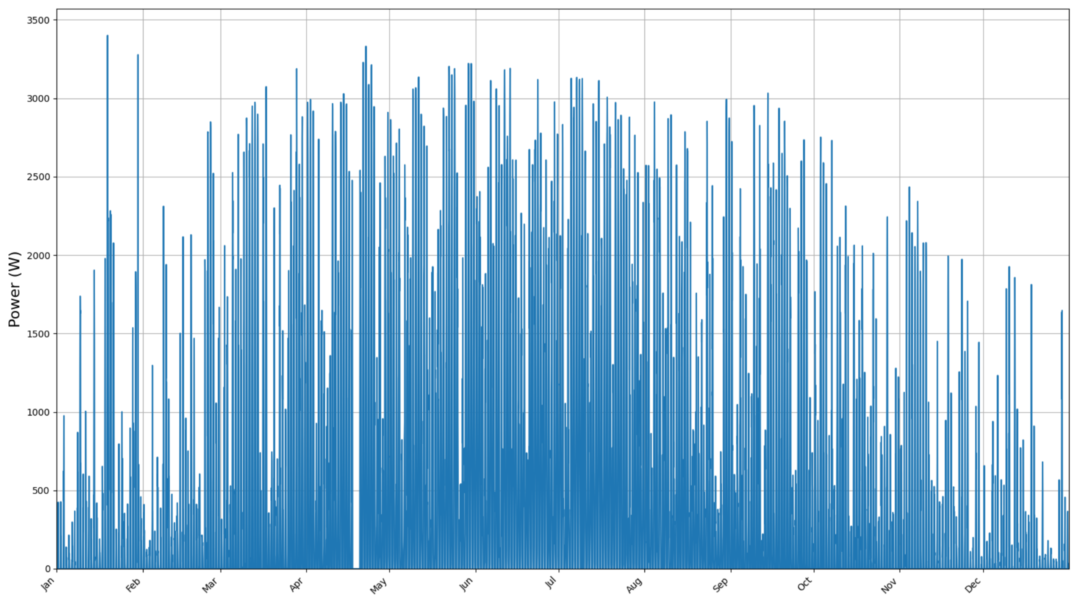

Figure 3.6: PV System Production during a TMY using GSEE

Data Integration & Modification

TMY

Similar to the national load, the TMY dataset is also in hourly temporal resolution. Therefore, it has been reworked to 15 min time intervals with linear interpolation for all temperature and irradiance variables. GSEE requires two input variables, which are $global\_horizontal$ and $diffuse\_fraction$. The global horizontal irradiance is already in the original dataset under a different name ($G(h)$), so the variable can simply be renamed. The $diffuse\_fraction$ can be calculated via Equation 3.5.

$$diffuse\_fraction = \frac{diffuse\_horizontal}{global\_horizontal} \tag{3.5}$$

SARAH-3
The SARAH-3 datasets temporal resolution is 30 minutes, so linear interpolation is used for the irradiance variables to alter this resolution to 15 minutes. The $global\_horizontal$ can be calculated via Equation 3.6. Then the $diffuse\_fraction$ is calculated according to Equation 3.5.

$$global\_horizontal = direct\_horizontal + diffuse\_horizontal \tag{3.6}$$

The original dataset contains 3.92% missing values for both direct and diffuse horizontal radiation, mainly in the early years of the dataset. To improve data quality, all missing data points that lie between two time steps that do contain values for irradiance, the average of its neighbouring time steps is taken. Furthermore, all missing values for the time steps between 20:30 and 03:00 (UTC time) have been set to a value of 0. These two modifications decrease the share of missing values to 1.26% over the full time period.

ERA5
The temperature data from the ERA5 dataset is extracted using the "nearest" method, which selects the closest available data point based on the provided coordinates (latitude: 52.025, longitude: 4.375). Then, the 2m temperature is converted from Kelvin to Celsius and changed to 15 minute time resolution using the linear interpolation method. The SARAH-3 and ERA5 datasets are combined to create a full period dataset which includes both the irradiance and temperature data for the full period 1983 - 2024.

3.6.3. Storage

Battery System Parameters

Home Battery

The home battery is assumed to be the LG RESU10, a common home battery in the Netherlands, whose specifications are outlined in Table 3.1 [60]. These values are average for lithium-ion batteries. For the round-trip efficiency, 95% efficiency is assumed.

The ratio between the total energy capacity and the average daily household electricity load (6.85 kWh) is around 1.43. The ratio between usable capacity and household load is 1.28.

Table 3.1: Specifications of the LG RESU10 home battery

Parameter	Symbol	Value	Unit
Total Energy Capacity	$E_{\text{cap\_tot}}$	9.8	kWh
Usable Energy Capacity	E_{cap}	8.8	kWh
Maximum Charge/Discharge Power	P_{max}	5.0	kW
Battery Pack Round-Trip Efficiency	η	0.95	

Community Battery

An existing Dutch community battery is a 128kWh/55kW battery in Rijssenhout, placed in 2017 [61]. The community battery is used by 35 households to store excess electricity produced by PV systems to be used during times of undersupply. If the households have the same average demand as is assumed for this thesis, the ratio between the energy capacity of the battery and the average daily household electricity load is approximately 0.53. This ratio is almost three times smaller than the ratio assumed for the home battery, resulting in an unfair comparison.

However, it should be accounted for that community batteries are generally more efficient in terms of total installed capacity relative to the number of households than individual home batteries. Therefore, a ratio of installed capacity to household load of 1.1 is assumed. The maximum charge/discharge power follows a capacity factor (kWh / kW) of 4, following the same parameter as an earlier home battery and community battery study performed by CE Delft and Witteveen+Bos [62]. The same round-trip efficiency as for home batteries is assumed. An overview of the community battery specifications can be found in Table 3.2.

Table 3.2: Specifications of the community battery

Parameter	**Symbol**	**Value**	**Unit**
Energy Capacity	E_{cap}	$7.53 \times$ # households	kWh
Maximum Charge/Discharge Power	P_{max}	$1.88 \times$ # households	kW
Battery Pack Round-Trip Efficiency	η	0.95	

Integration

The storage model uses the input of the supply and demand data. It is calculated for every time step (t). Essentially, it can be split into two scenarios, one where the residual load is positive and one where it is negative. The residual load is calculated by Equation 3.7.

$$\text{Residual Load} = \text{Electricity Demand} - \text{PV Generation} \tag{3.7}$$

Starting off with the residual load being negative, thus PV power supply exceeding the demand at this time step, with excess power being denoted as P_{excess}:

The excess power is used to charge the battery, which is limited by the charge rate (P_{max}) and remaining capacity of the battery (E_{space}). If there is any remaining capacity, the power (kW) is converted into energy (kWh) by multiplying it with the timestep_hours constant (Δt), which equals 0.25 as the data is in 15 minute intervals. The full battery charging logic is described in Equation 3.8.

$$P_{\text{excess}} = -\text{Residual Load} \tag{3.8a}$$
$$P_{\text{in}} = \min\left(P_{\text{excess}}, P_{\text{max}}\right) \tag{3.8b}$$
$$E_{\text{in}} = P_{\text{in}} \cdot \Delta t \tag{3.8c}$$
$$E_{\text{storable}} = E_{\text{in}} \cdot \eta \tag{3.8d}$$
$$E_{\text{space}} = E_{\text{cap}} - \text{SOC}_i \tag{3.8e}$$
$$E_{\text{stored}} = \min\left(E_{\text{storable}}, E_{\text{space}}\right) \tag{3.8f}$$
$$\text{SOC}_i = \text{SOC}_i + E_{\text{stored}} \tag{3.8g}$$

All of the excess power which is not used to charge the battery is exported to the main utility grid (P_{export}). These calculations are done using Equation 3.9.

$$E_{\text{export}} = \left(P_{\text{excess}} \cdot \Delta t\right) - \left(\frac{E_{\text{stored}}}{\eta}\right) \tag{3.9a}$$
$$P_{\text{export}} = \frac{E_{\text{export}}}{\Delta t} \tag{3.9b}$$

Alternatively, in case of a positive residual load, the microgrid requires additional electricity (P_{shortage}). The shortage can be made up either by battery discharge or main utility grid import. The battery discharge logic is shown in Equation 3.10.

$$P_{\text{shortage}} = \text{Residual Load} \tag{3.10a}$$
$$E_{\text{shortage}} = P_{\text{shortage}} \cdot \Delta t \tag{3.10b}$$
$$P_{\text{discharge,max}} = \min\left(P_{\text{max}}, \frac{\text{SOC}_i}{\Delta t}\right) \tag{3.10c}$$
$$E_{\text{outgoing}} = P_{\text{discharge,max}} \cdot \Delta t \tag{3.10d}$$
$$E_{\text{usable}} = E_{\text{outgoing}} \cdot \eta \tag{3.10e}$$
$$E_{\text{discharged}} = \min\left(E_{\text{usable}}, E_{\text{needed}}\right) \tag{3.10f}$$
$$E_{\text{actual}} = \frac{E_{\text{discharged}}}{\eta} \tag{3.10g}$$
$$\text{SOC}_i = \text{SOC}_i - E_{\text{actual}} \tag{3.10h}$$

In case of insufficient battery capacity to satisfy the demand, the electricity is imported from the main utility grid. This process is explained in Equation 3.11.

$$E_{\text{import}} = E_{\text{shortage}} - E_{\text{discharged}} \tag{3.11a}$$
$$P_{\text{import}} = \frac{E_{\text{import}}}{\Delta t} \tag{3.11b}$$

3.6.4. Metrics

The model's calculations and assumptions for the cumulative deficit and import period duration metrics are greatly similar. The model checks if there is a positive value for main utility grid import. If this is the case, an import streak is started which adds the total energy imported and the time step for the cumulative deficit and import period duration, respectively. The streak is ended when there is a streak of more than four time steps (1 hour) of continuous non-import. The streak is ended and this counts as one period, both for the cumulative deficit and the import period duration. All of these streaks are added in one large array to be analysed. In case of missing values in the SARAH-3 dataset, the time step is skipped. This means the streak is not ended, but it also does not count as an extra time step for the import duration period.

Import power and export power, as well as the import and export ramp rates (which is the difference between import and export power between two consecutive time steps (Δt)) are documented in separate arrays, too. As either import or export power, or both, are zero at a give time step, only the positive values for both power and ramp rate statistics are taken for analysis to filter out all the non-positive values which do not make for an insightful analysis into these metrics.

4

Results

This chapter compares and analyses the various testing conditions introduced by the sub-questions. More specifically, this chapter analyses the maximum values of the six metrics introduced in section 3.5, as these values represent the highest performance demands placed on the microgrid and the main utility grid. Given the research objective, it is evident that grid congestion is a significant issue in the Netherlands. Evaluating the maximum values of the performance metrics helps determine whether the testing conditions mitigate or exacerbate this problem. The distribution of all the individual scenarios barely differs between scenarios and are, therefore, not relevant to consider. For reference, the cumulative distribution functions (CDFs) of all scenarios can be found in Appendix D.

Each of the scenarios are run a total of 10 times to create an estimate of the performance metrics of the scenarios. A higher number of runs would result in significant increases in the required computational capacity and total time for running each of the scenarios, hence the choice of running it 10 times has been made. The modular character of the model makes it so that switching between the different scenarios (subsection 3.4.5) is straightforward. Multiple parallel coordinate plots (PCPs) have been created to highlight the differences between the testing conditions. In the main text, some illustrative examples are given, but all of the PCPs can be found in Appendix C.

One of the main results found in this chapter is the immense effect different load patterns have on the performance metric output. The addition of a heat pump mainly increase the maximum cumulative deficit and import period duration, but also maximum import power and ramp rate are increased. Although have a lower effect than the addition of a heat pump, analysing the full period results in increases for all of the performance metrics across all scenarios. The effect of the different types of batteries are a less direct than for the testing conditions named above. The ”Gas Heating” scenario shows either reduced or similar values for the community battery, indicating that it is the preferred option. For the ”Heat Pump” scenario, however, there seems to be a trade-off between the self-sufficiency performance metrics (cumulative deficit and import period duration) and the main utility grid requirements (import and export power and ramp rate) for different battery types. Lastly, microgrid size does show to have an impact on the results, but this is lowest compared to the other testing conditions.

4.1. Gas Heating vs Heat Pump

Figure 4.1 shows the maximum values across the two scenarios for each of the metrics. This gives insights into the limits the microgrid system should be able to withstand. An average is provided for the scenarios, which acts as an indicator for the use of differing load patterns on the system’s performance requirements.

In terms of the cumulative deficit, an evident difference between the scenarios with a heat pump and the ones without is shown, as the average cumulative deficit is 5.23 times higher throughout all the scenarios. Scenario ”Gas Heating” also has longer maximum import durations than the ”Heat Pump” scenario, being 2.56 times higher. An explanation for this is that during the day when there are higher levels of irradiance, residents are not at home, resulting in low load values (possibly even base load).

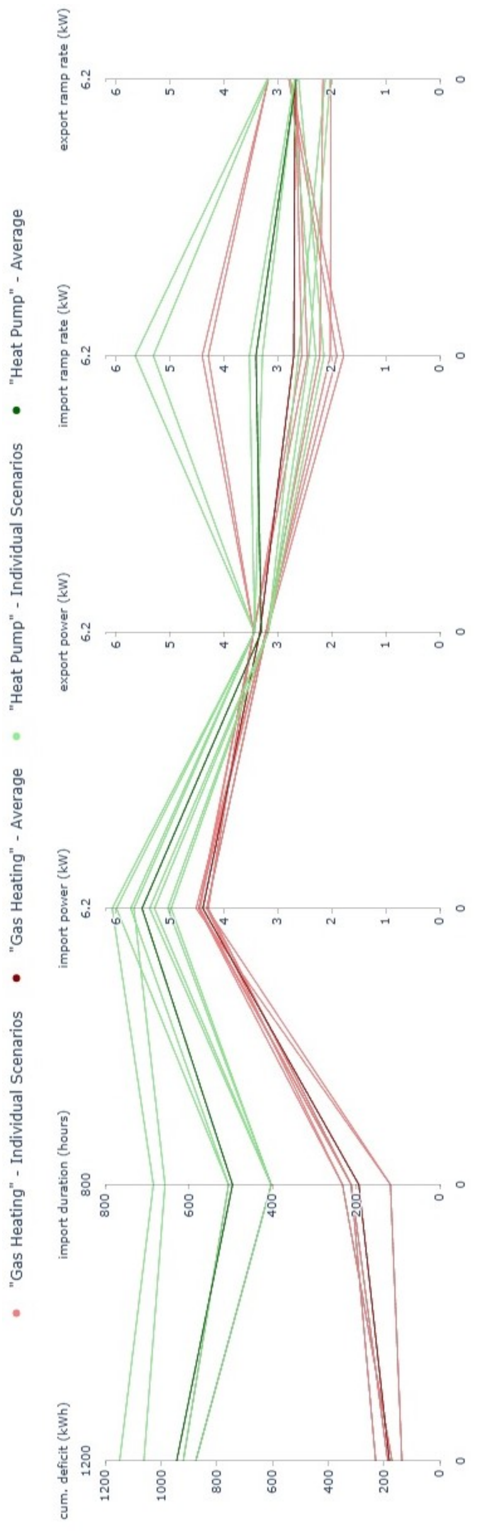

Figure 4.1: PCP comparing different load patterns (Gas Heating vs Heat Pump)

This makes so that the load can be met by the relatively low PV power production, thus ending the import duration streak. However, the model assumes the temperature of the house to always be at least at 18°C, resulting in higher load requirements, especially during colder months.

Two other metrics where the ”Heat Pump” scenario makes a large difference are the import peak power and peak ramp rate, with both metrics being 26% higher for the ”Heat Pump” scenario, on average. For the export peak power and peak ramp rate requirements, there is barely any difference between the averages of the two scenarios. However, there is a relatively large amount of variation between the scenarios for both import and export ramp rates, which will be explained by other testing conditions below.

All in all, it can be concluded that the addition of a heat pump has negative effects on the performance of a microgrid, mostly on the import side. Contrarily, the distinct load patterns have barely any effect on the export metrics.

4.2. TMY vs Full Period

The effect of using a time period of 42 years as compared to a singular year (TMY) is most clearly shown in Figure 4.2, which shows the comparison between the TMY and Full Period scenarios for the ”Heat Pump” scenario. From this figure, it becomes apparent what the effect of using the prolonged time period is. For essentially all of the performance metrics, the Full Period scenarios show higher results, indicating the increased capacity required by the main utility grid in order to supply the microgrid.

The maximum cumulative deficit increases by a factor of 15.7%, while the maximum import period duration even increases by 45.9%. This shows the reduced self-sufficiency of the microgrid under the most pressuring of times. Peak import power and ramp rate are also affected, increasing by a factor of 13.3% and 87.3% when the full period is taken into account, respectively. Maximum export power and ramp rate are affected less than the import, but increase nevertheless, by factors of 7.9% and 23.1%, respectively.

The effect of analysing a full period for the ”Gas Heating” scenario gathers similar results, the maximum cumulative deficit (35.9%) reporting an even higher difference than for the ”Heat Pump” scenario. All of the other performance metrics also show increased average maximum values when accounting for the full period, albeit in relatively lower terms than for the "Heat Pump" scenario.

From these results, the importance of considering prolonged time periods of weather data when conducting similar types of research is highlighted. Using singular year datasets (like the TMY dataset) will potentially lead to grave underestimation of the required capacity and limits needed by the main utility grid (and microgrid) to ensure a fully functioning system.

4.3. Home Battery vs Community Battery

Whereas the previous two sets of testing conditions has shown a clear difference between the two options, the relationship between the BESS type and performance metrics is more complicated. For the "Gas Heating" scenarios, the community battery achieves more favourable (scores lower on the performance metrics) results than the home batteries for all individual households. Maximum cumulative deficit decreases by 19.7%, while maximum import duration is reduced by 25.9%. Both maximum import ramp rate (25.2%) and export ramp rate (20.1%) are also reduced by using a community battery. The peak import and export power is not affected and stays equal for both battery types.

Comparing battery types for the "Heat Pump" scenario (Figure 4.3) proves to be less obvious, due to both battery types performing better on some parts of the performance metrics than the other. For instance, the community battery scenarios show higher maximum cumulative deficit (10.3%) and import period duration (18.3%) values than the home battery scenarios. Contrarily, the average maximum import ramp rate and export ramp rate for community batteries are 23.1% and 19.0% lower, respectively. Similar to the results from the "Gas Heating" scenario, the average peak power for both import and export is equal across all scenarios for this testing condition.

This makes for an interesting choice to be made by future grid operators and government institutions. On the one hand, self-sufficiency of the microgrid(s) can be favoured by opting for home batteries, but this does increase the ramp rate limits needed by the main utility grid, which might impact grid congestion. On the other hand, community batteries show reduced required ramp rate limits, but this is at the expense of self-sufficiency, since higher maximum values of cu-

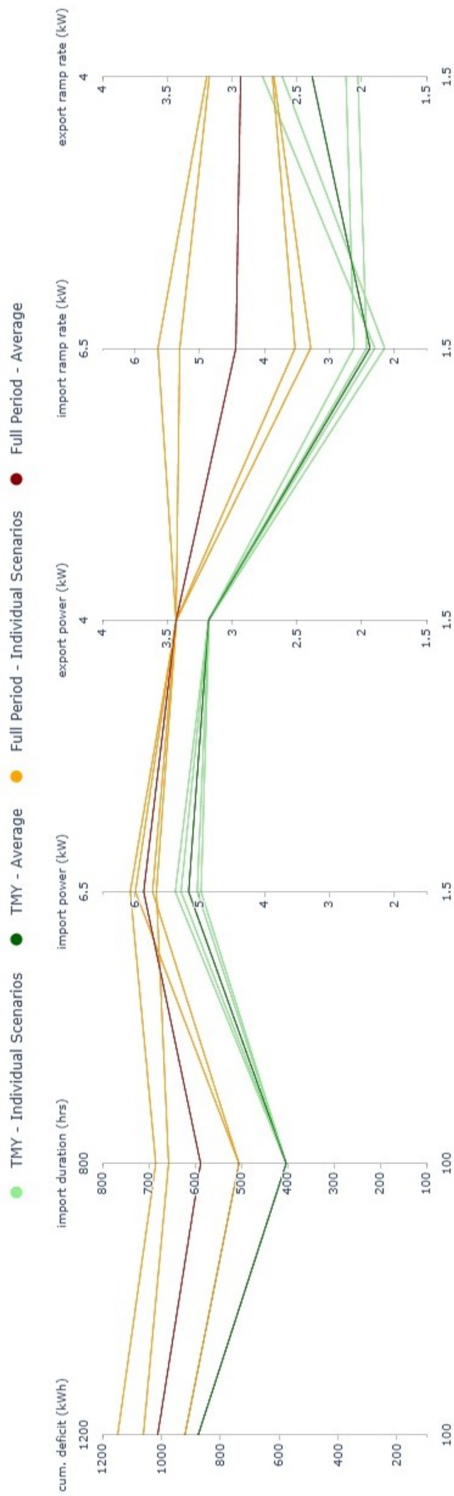

Figure 4.2: PCP comparing different time periods (TMY vs Full Period) - "Heat Pump" scenario

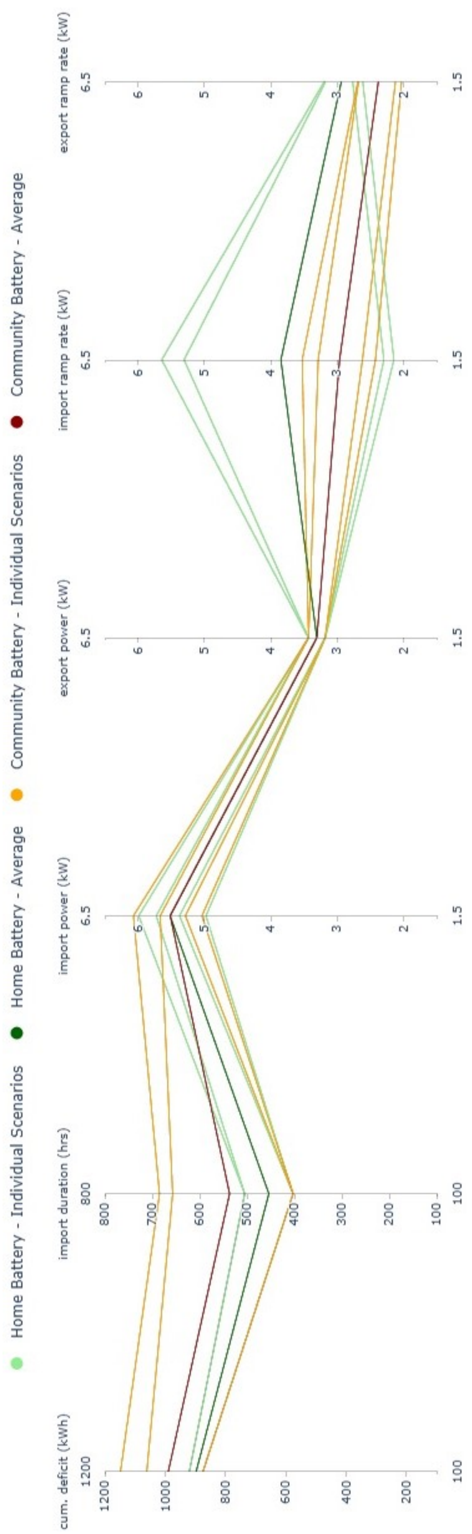

Figure 4.3: PCP comparing different BESS types (Home Battery vs Community Battery) - ”Heat Pump” scenario

mulative deficit and import period duration are found.

4.4. Small Microgrid vs Large Microgrid

Microgrid size has an effect on the performance metrics, but it is very small compared to the other testing conditions. The ”Gas Heating” scenario finds a maximum of 5.5% decrease across the maximum values across all performance metrics for the large microgrid size. The same conclusion holds for the ”Heat Pump” scenario, where a larger microgrid size results in a reduction of the maximum performance metrics by 5.8% or less, which is an effect far lower than found between the other testing conditions. Thus, as expected, a larger microgrid decreases the required capacity and limits from the main utility grid and increases self-sufficiency of the microgrid, but its effects are far lower than for the other testing conditions.

The relatively small effect of microgrid size can have multiple reasons. Firstly, there is a chance that the diversified load, resulting in peak shaving, genuinely has less effect than the other testing conditions. Another reason could be the setup of the model, which is further elaborated on in the limitations section in section 5.2.

4.5. Results Analysis

The cumulative deficit in the ”Gas Heating” scenario shows significantly lower values for community batteries, which aligns with the findings of other studies, which found community batteries to be more efficient in terms of required storage capacity than home batteries [63]. Also, significantly higher values are found for the full period scenarios in comparison to the TMY scenarios, highlighting the value of using datasets with a longer time span. The longest import period happened in December 2006 from a period for almost 9 days, 7 of which did not have a single sun hour [64].

A notable aspect is that import power for scenario ”Gas Heating” has superseded export power, due to the implementation of the household load pattern which is oscillating between higher and lower power values to a large extend. The community battery leads to lower ramp rate requirements than the home battery, which is a similar effect that is found in academic literature [65]. However, this is only for the ramp rate, as for the maximum required power, there is no difference in using home batteries or a community battery. This indicates that there are times that there is either too little or too much PV power that the difference in BESS does not matter and either all load needs to be imported when the BESS is empty or all excess electricity needs to be exported when the BESS is already fully charged.

The maximum cumulative deficit and longest import period duration of the full period scenarios in the ”Heat Pump” exceed the maximum values of the TMY scenarios. The maximum cumulative deficit and longest import duration period occur in November to December 2004, when in a period of 21 days, the number of days with 1.5 sun hours or more only occurred twice [66] [67]. The deviation in this time period compared to the ”Gas Heating” scenario can be explained by the much colder December month, especially in the first weeks. For the first 16 days in December 2004, there only was one day with a higher average temperature than the long-term average temperature. For the full month, the average temperature was 3.2°C, compared to an average long-term average of 4.0°C.

Comparing these weather conditions against the temperature measurements in December 2006, it is clear why this time period did not show up as the ’worst’ period in the ”Heat Pump” scenario. Unlike December 2004, December 2006 was a ”very mild” month, with an average temperature of 6.5°C, compared to the same average long-term average of 4.0°C. This would have resulted in lower power consumption by the heat pump, leading to lower peak power requirements and total consumed electricity. The combination of both low solar irradiance and temperature has led the November-December 2004 period to be the ’worst’ period in the ”Heat Pump” scenario. This is also illustrated in Figure 4.4, showing unusually low values for both irradiance and temperature.

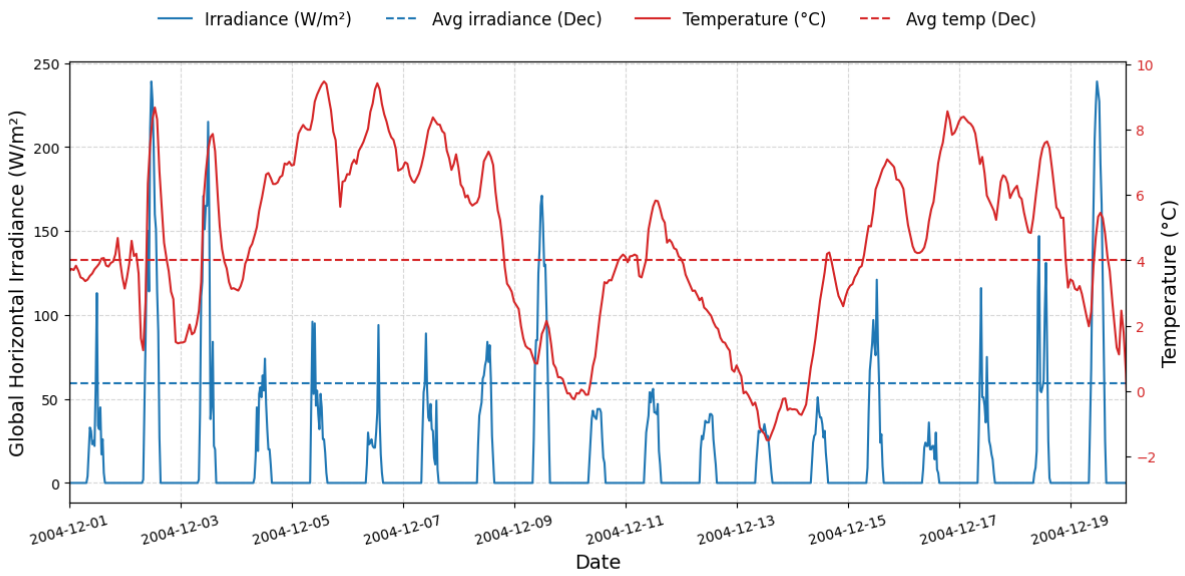

Figure 4.4: Irradiance and Temperature against the monthly average December 1 - December 20, 2004

Remarkably enough, December 2004 (54 hours) and 2006 (47 hours) both recorded higher total sun hours over the whole month compared to the long-term average of 44 hours. This indicates that the microgrid system is always challenged during periods of low irradiance, even if this period succeeds another period with higher than average solar irradiance. For instance, the ’worst’ run for the ”Gas Heating” scenario starts just after a period of three days with an average of 5 sun hours per day, while the long-term average is only 1.42 sun hours per day for December. Still, this period recorded as the worst run over the full 42-year period. The irradiance values over this period can be found in Figure 4.5, indicating an initial few days of high irradiance followed by days of extremely low irradiance.

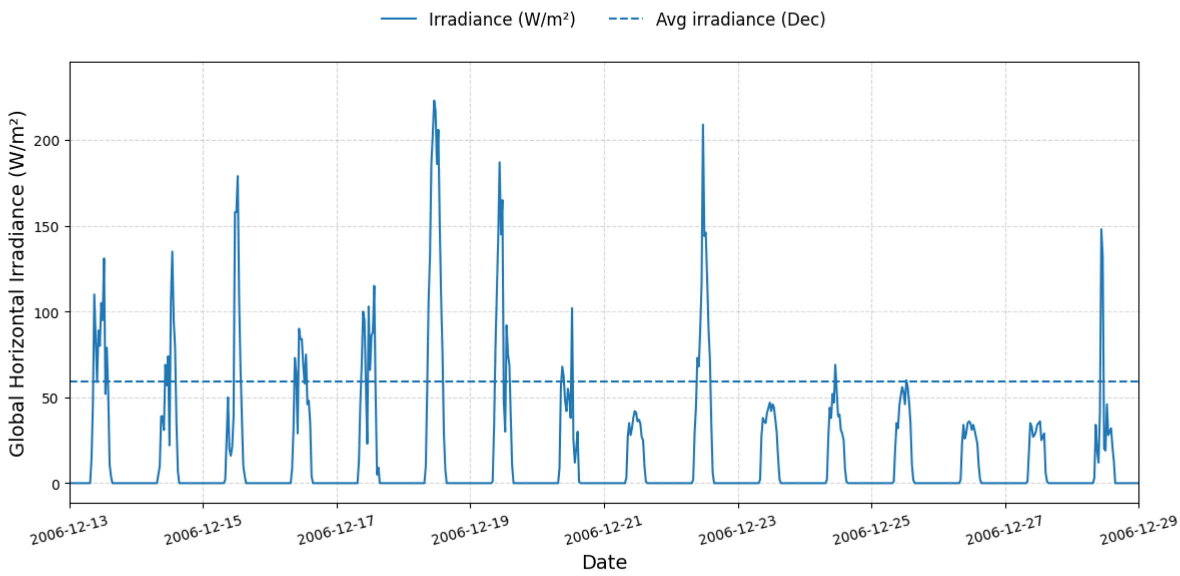

Figure 4.5: Irradiance against the monthly average December 13 - December 29, 2006

The main takeaways in this chapter are:

1. Load Pattern: Load pattern has a very large effect on the performance metric results, with "Heat Pump" scenarios reporting higher maximum values for almost all performance metrics compared to the "Gas Heating" scenarios, with the exception of the metrics related to export;
2. Time Period: Full period analyses show that greater main utility grid capacity and limits are needed when compared to the TMY scenarios. This is true for all of the performance metrics;
3. Battery Type: Community batteries seem to report lower values for all of the performance metrics (except peak import and export power) than the home batteries. However, for the "Heat Pump" scenario, higher maximum cumulative deficit and import period duration are reported, indicating reduced self-sufficiency of the microgrid;
4. Microgrid Size: Microgrid size does impact the results, but on a relatively small scale when compared to the other testing conditions.
5. Worst Run: No matter how beneficial the weather conditions preceding a period of low irradiance and temperature are to the microgrid system, it will still get challenged heavily over a prolonged period of several days or even weeks.

5 Discussion

5.1. Literature Validation

The results of the thesis are placed in the context of current academic literature in order to determine the validity and novelty of these results. The results show an overall clear difference between the maximum performance required from the system when simulating for a full period in comparison to a single year (TMY). This indicates studies which performed with a single year might underestimate the upper boundaries needed to be met by the microgrid, thus resulting in unforeseen situations and grid instability caused by voltage or frequency drops. Studies performed in Italy and Kenya found similar results when comparing traditional single-year approaches to multi-year tools [68] [69]. Even though these studies express the performance in terms of monetary costs instead of actual system boundaries, it is still clear that there is an advantage to be gained by using multi-year tools and approaches instead of singular year ones. Results in this thesis underline the same principle.

Another result that is found in this thesis is the large effect a heat pump has on the overall required capacity of the microgrid system, which is also found by a sut [70]. Peak demand even went up with a factor of 36%, which is slightly higher than the averages that were found using the model in this thesis.

In many of the scenarios, a community battery has proven its advantages over the home batteries used for single households. This has been especially the case in the full period scenarios, where maximum microgrid requirements have been lowered by implementing community batteries. Although the results in this thesis have not been as large as in some other studies, the community battery still performs better, especially at larger microgrid sizes.

For instance, a German study found community batteries to increase self-sufficiency by 11.6%, alongside an 8% increase in self-consumption [71]. Contrarily, the results in this thesis show a decrease in self-sufficiency when using community batteries compared to individual home batteries, when assuming that all households utilise a heat pump. Capacity-wise, the community battery enables savings of up to 68% compared to home batteries while achieving similar results.

Pilz et al. (2019) found similar benefits of implementing a community battery, allowing for a lower installed capacity due to less pronounced asynchrony of the demand to the PV system production [72]. These results show a clearer advantage of the community battery than the results in this thesis. This is potentially the case because of the standard household load pattern used in this thesis. Even though random variation is added, there is not as much variation in the (peak) load as there would be with distinctly different households and individual load patterns.

This aspect also links to the highly limited effect microgrid sizing has on the performance metrics. The anticipated effect before generating the results of this thesis was that microgrid size was (one of) the testing conditions with the largest effect on the performance metrics, but it has proven to be the testing condition with the smallest overall effect on each of the metrics throughout the scenarios. The effect of a larger microgrid size has been much more apparent in other academic literature, such as in studies performed in Switzerland and the United States [73] [74].

5.2. Limitations

5.2.1. Main Utility Grid Boundaries

This thesis aims to explore the ultimate boundaries the main grid has to endure to facilitate a smoothly operating microgrid. However, there is no knowledge on the actual upper limits of the main utility grid at this point in time. It is well-known that the main utility grid is under heavy pressure and that net congestion is an everyday issue only to get worse in the future. Increasing peak power and ramp rates will only exacerbate this problem. However, due to the unknown upper limits of the TSO (TenneT) and the various DSOs, it is not possible to place the results of this thesis into this context.

5.2.2. Missing Values SARAH-3

The amount of missing values of the SARAH-3 dataset, providing irradiance values from 1983 - 2024, make it difficult to estimate PV system production for certain time periods. Especially in the early years, 1983-1990, with at least 10% of values missing every year. The two modifications of the dataset in this thesis reduces the amount of missing values to around an average of 5% for the early years, but this amount is still significant to alter the metrics, be it in a positive or negative way.

5.2.3. Temporal Resolution Datasets

Of all the datasets, the household load pattern has the highest temporal resolution with a 15 minute interval between time steps. Many of the other datasets are hourly, while SARAH-3 is provided in 30 minute time intervals. This limitation leads to two problems. The first is that higher resolution time sets can be created via linear interpolation, but this method averages out the variables over a certain time period. This method is likely fine for variables like temperature or irradiance, but in case of load patterns, there can be large differences and fluctuations between seconds. Secondly, the voltage control on the main utility grid is also controlled over several seconds, meaning that fluctuations on this temporal resolution can already impact the grid's stability [75]. Therefore, analysing the system on an even higher temporal resolution would benefit the validity of the results. A caveat that needs to be made is that this would significantly impact the run time and computational power required by the model.

5.2.4. Household Load Data

The household load pattern and the added random variation in peak and time shifting used in this thesis provides a solid starting point for estimating the load pattern of an increased number of households. However, it would be better to use the load data of many different households, as this would increase the validity of the results, by simulating more realistic scenarios. DSO's have the data to do this, so these organisations would likely provide more accurate results using the model of this thesis when laid against an increased number of real-world household load patterns. Having a more diversified overall load pattern would mostly benefit the community battery scenarios when compared to the home battery scenarios, resulting in a more prominent distinction between the two. Additionally, increased microgrid sizes would likely perform substantially better on the performance metrics than microgrids with a low number of households. This might be one of the main reasons that this effect is relatively small in the results of this thesis.

5.2.5. Monetary Costs

All results in this thesis are based on the technical boundaries and limitations of the PV system and BESS. However, monetary costs have not been considered in this thesis. This aspect is important to consider before implementing the solutions proposed in this thesis. For instance, a community battery might be lower in costs per installed capacity when compared to home batteries for each household, but this would require cooperation between the households and/or municipality in the microgrid. Additionally, heat pumps or a PV system can be a high up-front investment for households, so before assuming that each household can install these, an overview should be made on the ability and willingness-to-pay of the households in the microgrids. Additionally, the grid stability and other technically related aspects can also be expressed in monetary terms, such as the value of lost load or the costs of grid failure. Expressing all assets and effects in terms of monetary costs will also allow for a fair and clear comparison between the scenarios and testing conditions in this thesis.

5.2.6. Heat Pump Consumption

The model in this thesis assumes the heat pump to always heat up the household to 18°C, when ambient temperature is lower. This also happens during the night and middle of the day, when residents are either asleep or their everyday activities (work, school, study). At these times, it is not realistic to be heating each household as if someone were to be at home. Therefore, heat pump electricity consumption is likely to be overestimated, given the other constant variables used in this thesis. This effect would therefore lead to a higher pronounced difference between the ”Heat Pump” and ”Gas Heating” scenarios than would have been the case when using real household data.

5.2.7. Constant Variables

Some of the variables in this thesis have been kept constant, but could be varied to achieve more testing conditions and a more complete analysis of the system. Examples of this are the installed capacity of the PV system, the home batteries, the community battery, as well as the tilt, rotation, tracking and placement of the PV system. Additionally, other variables, such as the partial shading effect, could have been added to allow for a more realistic and complete overview of all the effects of testing conditions in this thesis.

5.3. Theoretical Implications & Future Research

From the literature review, several theoretical knowledge gaps have been identified, which this thesis aims to fill. First off, there is the geographical knowledge gap, with no studies being performed on the relation between weather effects and Dutch microgrid performance. This interplays with the fact that current academic literature barely ever combines load data and weather data from the same location, which allows for a more accurate model, and thus results. Also, the time of analysis of the studies often has been limited to a singular year, without considering possible (negative) deviations in terms of weather conditions which would critically impact the required performance of the microgrid. By combining the load and weather data of the Netherlands, while also analysing the system's performance over a period of more than 40 years, this study has bridged these knowledge gaps, while also adhering to the research objective.

Coming back to the limitations of the previous subsection, there is still potential to enrich this study in future research. Varying other components of the microgrid, such as BESS capacity, PV system capacity and configuration, and adding the partial shading effect would result in a more complete analysis of the full system. Furthermore, placing the results in context of grid boundaries or monetary costs might lead to even more insightful results and conclusions by taking another perspective on the same issue.

Additionally, working with real-world household load patterns will lead to increased diversification of the total load of the microgrid, thus resembling a more realistic model. This effect can also be achieved by simulating a large number of household loads and adding more complexity to generate distinctly different household load patterns, increasing the overall validity of the model. Lastly, increasing the temporal resolution of the datasets, especially the load data, would lead to even more useful and accurate results. These research directions could be taken by future studies to enhance the model, results, and overall knowledge of microgrid systems.

5.4. Societal Implications & Recommendations

This thesis has mainly looked into the technological aspect of the net congestion issue, but in this section the issue will be placed in a larger socio-technical view. Combining technical, social, institutional, and ethical dimensions, the full socio-technical environment around net congestion in the Netherlands can be understood, as well as how the results of this thesis impact this environment.

5.4.1. Technical Inisghts & Direct Recommendations

Extreme Period Analysis

This thesis has proven the benefit of using multi-decade datasets and planning tools, as opposed to a singular year or TMY. When using the weather data of one year, it might lead to underestimations of the preparations required to overcome longer periods of low microgrid performance, and thus reliance on the main utility grid. Due to the small size of the Netherlands, weather conditions vary only slightly over the whole country, which means that when there is a period with a low number of sun hours (like in December 2024), this applies to the whole country. This signifies that all the microgrids, or Energy Hubs, would need electricity from the main utility grid simultaneously, potentially leading to enhanced levels of grid instability and net congestion. On the other side of the spectrum, during periods of extremely high irradiance, all microgrids will be exporting electricity simultaneously, resulting in increased chances of grid instability and net congestion, too.

To overcome the unforeseen limits the microgrid system and national electricity system should endure, planning studies should incorporate the weather data of several decades to capture worst-case extremes. A more accurate estimation of the worst-case extremes would result in microgrid models and planning tools that would appreciate these real-life limits a microgrid system has to face up to. This, on its turn, will lead to policies and subsidy measurements that reflect the true performance needs of the microgrid systems.

An important addition to this feat is that extreme weather is becoming more and more extreme by the year due to the effects of global warming. For instance, from the 31 heat waves in the Netherlands measured from 1901, 20 have occurred since 1990 [76]. Precipitation has also increased drastically, showing an increase of 26% over the period 1910-2022 [77]. Precipitation has increased most during winters (46%). For these reason, a historical analysis of weather data might result in underestimation of the performance required by the microgrid and the system as a whole to ensure smooth operation.

Focus on High-Impact Loads

Of all the components and testing conditions in this study, the load pattern has proven to be the component with the highest impact on the performance of a microgrid. Especially the scenarios with households utilising a heat pump, highly increased levels of cumulative deficit, import period duration, and import power and ramp rates are found. With increasing levels of electrification, such as induction stoves, heat pumps, and EVs, the limits of the electricity grid are going to be met. Without serious expansion of the utility grid, it is only a matter of time before serious issues start to emerge. This is a well-known problem in the Netherlands and grid expansion is high on the political agenda, but the expansions are costly and require long times before fully implemented.

The Dutch government has already targeted the Dutch society in a campaign called *Also turn the switch* [78]. With this campaign, the Dutch government hopes to move a portion of the peak load between 16:00 and 21:00 to another part of the day. The campaign indicates that the Dutch government is well aware of the (upcoming) issues the electricity grid is facing and has already started taking action. However, shifting the peak demand through appliances like dish washers or washing machines cannot compete with heat pumps or EVs in terms of power requirements. With an increasing number of both heat pumps and EVs in the Netherlands, the Dutch government would do well to shift their perspective to campaigns and policies on reducing the effect that these high-impact loads have on the electricity grid.

The results in this thesis have pointed out that load diversification and peak shifting will not make a difference for the low-impact loads. Even though increasing microgrid size did result in a few percentage points reduction of the metrics, its effect has been substantially lower than the differences other testing conditions have made. Therefore, the Dutch government would do well to target their policies and subsidies specifically on the high-impact loads, since households without heat pumps (and EVs) only result in minor improvements. While the campaign *Also turn the switch* is a good first step, it underscores a mismatch between political messaging and the physics: shifting a dishwasher load can never match the peak power of EV charging. This gap points to the need for tariff reforms and targeted subsidies.

Battery Storage

This study has proven the benefits and disadvantages of community batteries when compared to home battery for all of the individual households. In terms of maximum cumulative deficit and import period duration, community battery scenarios score worse (in the "Heat Pump" scenario), but they do have a large beneficial effect on the peak power and ramp rate required from the main utility grid. As these are the main culprits in causing the net congestion effects, national and local governments would do well by advising or stimulating residential neighbourhoods to opt for a community battery instead of home batteries. Home battery storage systems are still better than no storage at all, but as the choice is still largely to be made, community batteries would be highly recommended.

The number of home batteries in the Netherlands is still relatively low, but a steep increase over the past year has seen the number of home batteries more than double to 40.000, with total capacity even triple [79]. The choice between community batteries and home batteries is still largely to be made, but owners of PV systems are taking matters into their own hands by purchasing home batteries. If the Dutch government wants to maximally alleviate net congestion and ensure grid stability, community batteries are recommended. The Dutch government is therefore advised to act quickly, before the growth of home batteries reaches a level where there is no point of return. Another advantage of community batteries over home batteries is the level of control the DSO's and other controlling organisations have on the import and export of electricity to the main utility grid. Without this control, home battery owners would only look to maximise benefits and profits for individual households, which could have catastrophic effects when this happens on a large scale. Choosing between home and community batteries impacts more than just peak power—it also highlights two different governance approaches: individual, profit-driven ownership versus centrally coordinated management by the DSO.

5.4.2. Governance, Economic & Equity Dimensions

Regulatory Misalignments

The results in this thesis have shown that heat pumps can significantly increase electricity consumption, thereby decreasing microgrid self-sufficiency and overall performance. In the Netherlands, most residential network tariffs are still flat; a tariff per kWh is paid, regardless of the time or location. As of April 2025, only 6% of Dutch households have a dynamic energy contract [80]. Incentivising households with heat pumps and EVs to withdraw electricity at off-peak moments is key in relieving the grid.

Cost Allocation & Justice

Infrastructure costs (such as grid reinforcements) are funded through network tariffs paid for by all users. However, the lower-income households (without a PV system or BESS) derive much lower direct benefits from these reinforcements than the prosumers, which already captured several subsidies for these purchases. This will only further increase the gap between lower-income and higher-income households. Therefore, as equity should be one of the goals of the Dutch government, an altered tariff system should be instigated, so that costs and benefits are distributed fairly among households. A first step in the right direction would be the wide-spread implementation of community batteries, which, as the results have shown, reduces the upper limits needed by the main utility grid (and thus reduces the total costs for grid reinforcements).

Emerging Actors & Regulatory Frameworks

Since Energy Hubs and other similar communities in the Netherlands have not been implemented on a large scale, a direct regulatory framework is yet to be implemented. Aggregators and energy cooperatives are stepping in to organise these communities, but are operating in a regulatory gray zone [81]. Establishing clear market rules and integration pathways will be key in successful integration of Energy Hubs in the Netherlands.

5.4.3. Behavioural & Cultural Perspectives

Demand Response Barriers

Even though the *Also turn the switch* campaign has the right aim: shifting loads away from peak hours, many households simply can not do so. These households lack smart timers or home-automation systems to schedule dishwashers and washing machines. Even with the technology available, daily routines, such as turning on the dish washer at night, need to be broken for these campaigns to work.

Trust & Participation

Roll-out of smart meters in the Netherlands faces resistance in some neighbourhoods, due to the lack of trust of grid operators and data security. Local residents often question who really benefits from these projects and whether they will have control. One example of this happening is in Hoogkerk, a neighbourhood in Groningen [82]. To counter this issue, robust engagement through town hall meetings, participatory design workshops, and transparent reporting on performance and revenues will be essential to build the social license needed for new grid infrastructure and Energy Hub implementation.

6 Conclusion

This thesis has explored the interplay of different components of a Dutch residential microgrid to find out how these affect its performance and broaden the knowledge on the implementation of microgrids into the Dutch Energy Hubs. The formulated main research question and its sub-questions is the result of the combination of the research objective with the knowledge gap, which was found in the extensive literature review. More specifically, this thesis aimed to find the effect of different load patterns, multi-decade weather data, BESS types, and microgrid size on the overall performance of a microgrid and its role in the net congestion issue. To answer these questions, a modelling approach was taken on. The results of this thesis allow for answering the sub-questions.

Firstly, the addition of a heat pump on the load pattern has the largest impact on the performance metrics. With the further electrification of Dutch households, this seems to be the main challenge in smooth implementation of microgrids into the Dutch Energy Hubs. DSO's, local and national government would do well to plan the expansion of grids, implementation of microgrids, and future policies aimed at the load patterns of Dutch society according to the assumed high penetration of heat pumps and EVs into the Dutch households.

Furthermore, the model incorporates the different time periods and compares results of a TMY with the ones of a 42-year time period (1983-2024). Its findings indicate that when using a multi-decade dataset, the limits of the microgrid system are challenged to a larger extend than when compared with analysing the system in a single TMY. Using single year datasets for irradiance, temperature and other weather conditions might lead to grave underestimations of the required main utility grid capacity for microgrid electricity importation.

BESS types also affect the performance of the microgrid, with each having its advantages and disadvantages. While community batteries perform better throughout the "Gas Heating" scenario home batteries perform better for the performance metrics of cumulative deficit and import period duration in the "Heat Pump" scenario. Community batteries have proven to do better in terms of peak power and ramp rate requirements throughout both scenarios. As the latter are the main indicators in alleviating or exacerbating net congestion, which already is a pressing problem in the Netherlands, community batteries seem to be the preferred option of the two. Even though community batteries might lead to decreased self-sufficiency in terms of energy and duration, the peak requirements of the grid are alleviated, which leads to enhanced grid stability.

Lastly, microgrid size has an unanticipated small effect on the overall performance of the microgrid. For most of the six metrics, a larger microgrid does perform slightly better than a small microgrid, but this effect is only small compared to the other testing conditions. This might have to do with the household load pattern used in this thesis, and it would be beneficiary for the validity of results if the model was used in the reference of real-world household load patterns of a greater number of households.

To ensure successful implementation of Energy Hubs in the Dutch energy system, these results have been placed in the greater context of the socio-technical environment. Policy instruments, such as campaigns, subsidies, and regulations should account for cultural and behavioural barriers, while gaining the trust of residents for fair and safe implementation of smart grids and energy communities. By adjusting regulations to incentivise dynamic pricing structures, rethinking the cost allocations to allow a fair distribution of costs among households, and setting up a regulatory framework for the emerging Energy Hubs and communities, the Dutch government has the ability to shape the future energy system into a resilient, inclusive, and sustainable energy future - one that empowers consumers, fosters local communities, and accelerates the Netherlands' transition to a low-carbon power system.

References

[1] Open Overheid. *Regeerprogramma: Uitwerking van het hoofdlijnenakkoord door het kabinet*. 2024. URL: https://open.overheid.nl/documenten/ronl-f525d4046079b0beabc6f897f79045ccf2246e08/pdf.

[2] European Commission. *European Climate Law*. 2024. URL: https://climate.ec.europa.eu/eu-action/european-climate-law_en.

[3] Overheid.nl. *Klimaatwet*. 2023. URL: https://wetten.overheid.nl/BWBR0042394/2023-07-22.

[4] Centraal Bureau voor de Statistiek (CBS). *Energieverbruik uit hernieuwbare bronnen gestegen naar 17 procent*. 2024. URL: https://www.cbs.nl/nl-nl/nieuws/2024/23/energieverbruik-uit-hernieuwbare-bronnen-gestegen-naar-17-procent.

[5] European Environment Agency. *Share of energy from renewable sources, by country*. Accessed: 2025-03-03. 2024. URL: https://www.eea.europa.eu/en/analysis/maps-and-charts/countries-breakdown-actual-res-progress-15.

[6] Centraal Bureau voor de Statistiek (CBS). *Nearly half the electricity produced in the Netherlands is now renewable*. 2024. URL: https://www.cbs.nl/en-gb/news/2024/10/nearly-half-the-electricity-produced-in-the-netherlands-is-now-renewable.

[7] Energy Education. *Dispatchable source of electricity*. 2024. URL: https://energyeducation.ca/encyclopedia/Dispatchable_source_of_electricity.

[8] Open Overheid. *Kamerbrief: Netcapaciteit, de versnelling van de energietransitie en de noodzaak van flexibiliteit*. 2023. URL: https://open.overheid.nl/documenten/90621968-e148-482c-80f7-b832a1ba0978/file.

[9] Rijksoverheid. *Kabinet neemt maatregelen tegen vol elektriciteitsnet (netcongestie)*. 2024. URL: https://www.rijksoverheid.nl/onderwerpen/duurzame-energie/kabinet-neemt-maatregelen-tegen-vol-elektriciteitsnet-netcongestie.

[10] National Renewable Energy Laboratory (NREL). *Microgrids*. n.d. URL: https://www.nrel.gov/grid/microgrids.html#:~:text=A%20microgrid%20is%20a%20group,grid%2Dconnected%20or%20island%20mode..

[11] Aude Pommeret and Katheline Schubert. "Optimal energy transition with variable and intermittent renewable electricity generation". In: *Journal of Economic Dynamics and Control* 134 (2022), p. 104273. ISSN: 0165-1889. DOI: https://doi.org/10.1016/j.jedc.2021.104273. URL: https://www.sciencedirect.com/science/article/pii/S0165188921002086.

[12] Netbeheer Nederland. *Energyhubs*. Accessed: 2025-01-31. 2025. URL: https://www.netbeheernederland.nl/netcapaciteit-en-flexibiliteit/energyhubs.

[13] Topsector Energie. *Energy Hubs: een sleutelrol in ons toekomstige energiesysteem*. Accessed: 2025-03-03. 2024. URL: https://topsectorenergie.nl/nl/nieuws/energy-hubs-een-sleutelrol-in-ons-toekomstige-energiesysteem/.

[14] TNO. *Samenwerken in energiehubs: energie lokaal slim organiseren*. Accessed: 2025-03-03. 2024. URL: https://www.tno.nl/nl/newsroom/insights/2024/04/samenwerken-energiehubs-energie-lokaal/.

[15] Delft University of Technology. *MSc Complex Systems Engineering and Management (CoSEM)*. Accessed: 2025-05-28. 2025. URL: https://www.tudelft.nl/onderwijs/opleidingen/masters/cosem/msc-complex-systems-engineering-and-management.

[16] GridX. *What is Dunkelflaute?* Accessed: 2024-12-17. 2024. URL: https://www.gridx.ai/knowledge/what-is-dunkelflaute.

[17] Solcast. *Typical Meteorological Year (TMY) Weather Data*. Accessed: 11-06-2025. URL: https://solcast.com/tmy.

[18] Sarah Newman et al. "A comparison of PV resource modeling for sizing microgrid components". In: *Renewable Energy* 162 (2020), pp. 831–843. ISSN: 0960-1481. DOI: https://doi.org/10.1016/j.renene.2020.08.074. URL: https://www.sciencedirect.com/science/article/pii/S0960148120313197.

[19] Nicholas D. Laws et al. "Impacts of valuing resilience on cost-optimal PV and storage systems for commercial buildings". In: *Renewable Energy* 127 (2018), pp. 896–909. ISSN: 0960-1481. DOI: https://doi.org/10.1016/j.renene.2018.05.011. URL: https://www.sciencedirect.com/science/article/pii/S0960148118305305.

[20] Sergio B. Sepúlveda-Mora and Steven Hegedus. "Resilience analysis of renewable microgrids for commercial buildings with different usage patterns and weather conditions". In: *Renewable Energy* 192 (2022), pp. 731–744. ISSN: 0960-1481. DOI: https://doi.org/10.1016/j.renene.2022.04.090. URL: https://www.sciencedirect.com/science/article/pii/S0960148122005584.

[21] Jeremy Jie Ming Kwok et al. "Microgrid scheduling for reliable, cost-effective, and environmentally friendly energy management". In: *Industrial & Engineering Chemistry Research* 52.1 (2013), pp. 142–151. DOI: https://doi.org/10.1021/ie3006897.

[22] Sifat Chowdhury and Yu Zhang. "Two-Stage Stochastic Optimal Power Flow for Microgrids With Uncertain Wildfire Effects". In: *IEEE Access* 12 (2024), pp. 68857–68869. DOI: 10.1109/ACCESS.2024.3397920.

[23] Mohamed A. Mohamed et al. "Proactive Resilience of Power Systems Against Natural Disasters: A Literature Review". In: *IEEE Access* 7 (2019), pp. 163778–163795. DOI: 10.1109/ACCESS.2019.2952362.

[24] Xuan Wang et al. "Modeling smart electrical microgrid with demand response and storage systems for optimal operation in critical conditions". In: *Science and Technology for Energy Transition* 79 (2024), p. 55.

[25] Ahmed Khayat et al. "Hybrid model for microgrid short term load forecasting based on machine learning". In: *IFAC-PapersOnLine* 58.13 (2024). 12th IFAC Symposium on Control of Power and Energy Systems - CPES 2024, pp. 527–532. ISSN: 2405-8963. DOI: https://doi.org/10.1016/j.ifacol.2024.07.536. URL: https://www.sciencedirect.com/science/article/pii/S2405896324006359.

[26] Andrei Marinescu et al. "Residential electrical demand forecasting in very small scale: An evaluation of forecasting methods". In: *2013 2nd International Workshop on Software Engineering Challenges for the Smart Grid (SE4SG)*. 2013, pp. 25–32. DOI: 10.1109/SE4SG.2013.6596108.

[27] Andrzej Ożadowicz and Gabriela Walczyk. "Energy Performance and Control Strategy for Dynamic Façade with Perovskite PV Panels—Technical Analysis and Case Study". In: *Energies* 16.9 (2023). ISSN: 1996-1073. DOI: 10.3390/en16093793. URL: https://www.mdpi.com/1996-1073/16/9/3793.

[28] Yu-Jen Liu et al. "Development of a Modelling and Simulation Method for Residential Electricity Consumption Analysis in a Community Microgrid System". In: *Applied Sciences* 7.7 (2017). ISSN: 2076-3417. DOI: 10.3390/app7070733. URL: https://www.mdpi.com/2076-3417/7/7/733.

[29] Usman Bashir Tayab et al. "Microgrid Energy Management System for Residential Microgrid Using an Ensemble Forecasting Strategy and Grey Wolf Optimization". In: *Energies* 14.24 (2021). ISSN: 1996-1073. DOI: 10.3390/en14248489. URL: https://www.mdpi.com/1996-1073/14/24/8489.

[30] D. Naware and A. Mitra. "Weather classification-based load and solar insolation forecasting for residential applications with LSTM neural networks". In: *Electrical Engineering* 104 (2022), pp. 347–361. DOI: 10.1007/s00202-021-01395-2. URL: https://doi.org/10.1007/s00202-021-01395-2.

[31] G. Bruni et al. “Energy management in a domestic microgrid by means of model predictive controllers”. In: *Energy* 108 (2016). Sustainable Energy and Environmental Protection 2014, pp. 119–131. ISSN: 0360-5442. DOI: https://doi.org/10.1016/j.energy.2015.08.004. URL: https://www.sciencedirect.com/science/article/pii/S0360544215010488.

[32] Oussama Hafsi et al. “Integration of hydrogen technology and energy management comparison for DC-Microgrid including renewable energies and energy storage system”. In: *Sustainable Energy Technologies and Assessments* 52 (2022), p. 102121. ISSN: 2213-1388. DOI: https://doi.org/10.1016/j.seta.2022.102121. URL: https://www.sciencedirect.com/science/article/pii/S2213138822001734.

[33] Khalid Hanbashi et al. “Modelling and Validation of Typical PV Mini-Grids in Kenya: Experience from RESILIENT Project”. In: *Energies* 16.7 (2023). ISSN: 1996-1073. DOI: 10.3390/en16073203. URL: https://www.mdpi.com/1996-1073/16/7/3203.

[34] Sheroze Liaquat et al. “Day-ahead continuous double auction-based peer-to-peer energy trading platform incorporating trading losses and network utilisation fee”. In: *IET Smart Grid* 6.3 (2023), pp. 312–329. DOI: https://doi.org/10.1049/stg2.12103. eprint: https://ietresearch.onlinelibrary.wiley.com/doi/pdf/10.1049/stg2.12103. URL: https://ietresearch.onlinelibrary.wiley.com/doi/abs/10.1049/stg2.12103.

[35] Youness Hakam et al. “Hybrid ANN–GWO MPPT with MPC-based inverter control for efficient EV charging under partial shading conditions”. In: *Science Progress* 108.2 (2025). PMID: 40170511, p. 00368504251331835. DOI: 10.1177/00368504251331835.

[36] ICL Group. *Microgrids as a Solution for Industrial Sites*. Blog post on ICL Group website. Accessed: 11-06-2025. Aug. 2023. URL: https://www.icl-group.com/blog/icl-industrial-scale-microgrid-system-pioneer/.

[37] F. Norouzi et al. “Analysing the impact of the different pricing policies on PV-battery systems: A Dutch case study of a residential microgrid”. In: *Energy Policy* 204 (2025), p. 114620. ISSN: 0301-4215. DOI: https://doi.org/10.1016/j.enpol.2025.114620. URL: https://www.sciencedirect.com/science/article/pii/S0301421525001272.

[38] Energy-Charts. *Power consumption and production in the Netherlands*. https://energy-charts.info/charts/power/chart.htm?l=en&c=NL. Accessed: 12-03-2025. Energy-Charts, 2025.

[39] Centraal Bureau voor de Statistiek (CBS). *Energieverbruik particuliere woningen; woningtype en regio's*. https://www.cbs.nl/nl-nl/cijfers/detail/81528NED. Accessed: 13-03-2025. CBS, Mar. 2025.

[40] European Commission. *Photovoltaic Geographical Information System (PVGIS) tools*. Joint Research Centre. Dec. 2024. URL: https://re.jrc.ec.europa.eu/pvg_tools/en/.

[41] U. Pfeifroth et al. *Surface Radiation Data Set - Heliosat (SARAH) - Edition 3*. Satellite Application Facility on Climate Monitoring. 2023. DOI: 10.5676/EUM_SAF_CM/SARAH/V003. URL: https://doi.org/10.5676/EUM_SAF_CM/SARAH/V003.

[42] H. Hersbach et al. *ERA5 hourly data on single levels from 1940 to present*. Copernicus Climate Change Service (C3S) Climate Data Store (CDS). 2023. DOI: 10.24381/cds.adbb2d47. URL: https://doi.org/10.24381/cds.adbb2d47.

[43] Mao Tan et al. “Federated Reinforcement Learning for smart and privacy-preserving energy management of residential microgrids clusters”. In: *Engineering Applications of Artificial Intelligence* 139 (2025), p. 109579. ISSN: 0952-1976. DOI: https://doi.org/10.1016/j.engappai.2024.109579. URL: https://www.sciencedirect.com/science/article/pii/S0952197624017378.

[44] Amit Mohanty et al. “Neighborhood and community battery projects: A systematic analysis of their current state and future prospects”. In: *Journal of Energy Storage* 95 (2024), p. 112525. DOI: 10.1016/j.est.2024.112525. URL: https://doi.org/10.1016/j.est.2024.112525.

[45] Centraal Bureau voor de Statistiek (CBS). *Kerncijfers wijken en buurten 2024*. Mar. 2025. URL: https://www.cbs.nl/nl-nl/maatwerk/2025/13/kerncijfers-wijken-en-buurten-2024.

[46] Yichen Yao et al. “Quantitative metrics for grid resilience evaluation and optimization”. In: *IEEE Transactions on Sustainable Energy* 14.2 (2023), pp. 1244–1258. DOI: 10.1109/TSTE.2022.3230019. URL: https://doi.org/10.1109/TSTE.2022.3230019.

[47] Jakub Jurasz et al. “A review on the complementarity of renewable energy sources: Concept, metrics, application, and future research directions”. In: *Solar Energy* 195 (2020), pp. 703–724. DOI: 10.1016/j.solener.2019.11.087. URL: https://doi.org/10.1016/j.solener.2019.11.087.

[48] Francisco A. Canales and Gonzalo J. Acuña. “Chapter 2 - Metrics and indices used for the evaluation of energetic complementarity—a review”. In: *Complementarity of variable renewable energy sources*. Ed. by J. Jurasz and A. Beluco. Academic Press, 2022, pp. 35–55. DOI: 10.1016/B978-0-323-85527-3.00020-0. URL: https://doi.org/10.1016/B978-0-323-85527-3.00020-0.

[49] Python Software Foundation. *The Python language: A brief overview*. https://www.python.org/doc/essays/blurb/. Python.org, n.d.

[50] Tado°. *Verwarming in Europese Huizen Lager Ingesteld vanwege Energiekosten en Milieu*. 2023. URL: https://w3.tado.com/nl-nl/pers/verwarming-verlaagd-vanwege-energiekosten (visited on 06/05/2025).

[51] Kim. *Rc-waarde of Rd-waarde voor isolatie – welke waarde is goed?* Feb. 6, 2022. URL: https://fabercomfortvloer.nl/blog/rc-waarde-of-rd-waarde/ (visited on 06/05/2025).

[52] Milieu Centraal. *Vloerisolatie: voor warmere voeten*. URL: https://www.milieucentraal.nl/energie-besparen/isoleren-en-besparen/vloerisolatie/ (visited on 06/05/2025).

[53] Boris van Beijnum et al. *Referentieverbruik warmte woningen: Achtergrondrapport*. Tech. rep. 5168. Den Haag: Planbureau voor de Leefomgeving, 2023. URL: https://www.pbl.nl/sites/default/files/downloads/pbl-2023-referentieverbruik-warmte-woningen-achtergrondrapport_5168.pdf (visited on 06/05/2025).

[54] Energiewacht. *SCOP en COP warmtepomp: we leggen uit hoe het zit*. n.d. URL: https://www.energiewacht.nl/kennisbank/warmtepompen/cop-scop-warmtepomp (visited on 06/05/2025).

[55] Klimaatproductenkiezen.eu. *Wat houdt de 2% vuistregel in bij warmtepompen?* n.d. URL: https://klimaatproductenkiezen.eu/verwarming/warmtepomp/twee-procent-regel (visited on 06/05/2025).

[56] Energieplein. *Hoeveel vermogen heeft een warmtepomp nodig?* 2024. URL: https://www.energieplein.nl/hoeveel-vermogen-heeft-een-warmtepomp-nodig/ (visited on 06/05/2025).

[57] Kyle Anderson et al. “pvlib python: 2023 project update”. In: *Journal of Open Source Software* 8.92 (2023), p. 5994. DOI: 10.21105/joss.05994. URL: https://doi.org/10.21105/joss.05994.

[58] Stefan Pfenninger and Iain Staffell. “Long-term patterns of European PV output using 30 years of validated hourly reanalysis and satellite data”. In: *Energy* 114 (2016), pp. 1251–1265. DOI: 10.1016/j.energy.2016.08.060. URL: https://doi.org/10.1016/j.energy.2016.08.060.

[59] Essent. *Hoeveel kWh leveren zonnepanelen?* n.d. URL: https://www.essent.nl/kennisbank/zonnepanelen/wat-zijn-zonnepanelen/hoeveel-kwh-zonnepanelen.

[60] LG Home Battery. *RESU10*. LG Electronics, https://www.lghomebattery.com.au/resu10. n.d.

[61] Energy Storage NL. *Buurtbatterij*. Energy Storage NL. n.d. URL: https://www.energystoragenl.nl/projects/listing/buurtbatterij.

[62] L. Van Cappellen et al. *Thuis- en buurtbatterijen: Kansen, knelpunten en beleidsaanbevelingen*. Tech. rep. 23.230315.183. CE Delft & Witteveen+Bos, 2023. URL: https://ce.nl/wp-content/uploads/2023/12/CE_Delft_WitteveenBos_230315_Thuis-_en_buurtbatterijen_Def.pdf.

[63] Dilum Hettiarachchi et al. “Development of control strategy for community battery energy storage system in grid-connected microgrid of high photovoltaic penetration level”. In: *International Journal of Electrical Power & Energy Systems* 155 (2024), p. 109527. ISSN: 0142-0615. DOI: https://doi.org/10.1016/j.ijepes.2023.109527. URL: https://www.sciencedirect.com/science/article/pii/S0142061523005847.

[64] KNMI. *Maandoverzicht van het weer in Nederland, december 2006*. Accessed: 11-06-2025. URL: https://www.knmi.nl/nederland-nu/klimatologie/maand-en-seizoensoverzichten/2006/december.

[65] Chathurika P. Mediwaththe and Lachlan Blackhall. “Network-Aware Demand-Side Management Framework With A Community Energy Storage System Considering Voltage Constraints”. In: *IEEE Transactions on Power Systems* 36.2 (2021), pp. 1229–1238. DOI: 10.1109/TPWRS.2020.3015218.

[66] KNMI. *Maandoverzicht van het weer in Nederland, november 2004*. Accessed: 12-06-2025. URL: https://www.knmi.nl/nederland-nu/klimatologie/maand-en-seizoensoverzichten/2004/november.

[67] KNMI. *Maandoverzicht van het weer in Nederland, december 2004*. Accessed: 12-06-2025. URL: https://www.knmi.nl/nederland-nu/klimatologie/maand-en-seizoensoverzichten/2004/december.

[68] Mehdi Jafari, Magnus Korpås, and Audun Botterud. “Power system decarbonization: Impacts of energy storage duration and interannual renewables variability”. In: *Renewable Energy* 156 (2020), pp. 1171–1185. ISSN: 0960-1481. DOI: https://doi.org/10.1016/j.renene.2020.04.144. URL: https://www.sciencedirect.com/science/article/pii/S0960148120306820.

[69] Davide Fioriti et al. “Multi-year stochastic planning of off-grid microgrids subject to significant load growth uncertainty: overcoming single-year methodologies”. In: *Electric Power Systems Research* 194 (2021), p. 107053. ISSN: 0378-7796. DOI: https://doi.org/10.1016/j.epsr.2021.107053. URL: https://www.sciencedirect.com/science/article/pii/S0378779621000341.

[70] Claire E. Halloran, Filiberto Fele, and Malcolm D. McCulloch. “Impact of spatiotemporal heterogeneity in heat pump loads on generation and storage requirements”. In: *2022 IEEE Power & Energy Society General Meeting (PESGM)*. 2022, pp. 1–5. DOI: 10.1109/PESGM48719.2022.9916794.

[71] Jonas Quernheim and Eberhard Waffenschmidt. “A Comparative Evaluation of Community-Used District and Individual Battery Storage Systems for Photovoltaic Energy Systems”. In: *Energies* 17.17 (2024). ISSN: 1996-1073. DOI: 10.3390/en17174306. URL: https://www.mdpi.com/1996-1073/17/17/4306.

[72] Matthias Pilz, Omar Ellabban, and Luluwah Al-Fagih. “On Optimal Battery Sizing for Households Participating in Demand-Side Management Schemes”. In: *Energies* 12.18 (2019). ISSN: 1996-1073. DOI: 10.3390/en12183419. URL: https://www.mdpi.com/1996-1073/12/18/3419.

[73] Danielle Griego et al. “Aggregation effects for microgrid communities at varying sizes and prosumer-consumer ratios”. In: *Energy Procedia* 159 (2019). Renewable Energy Integration with Mini/Microgrid, pp. 346–351. ISSN: 1876-6102. DOI: https://doi.org/10.1016/j.egypro.2019.01.004. URL: https://www.sciencedirect.com/science/article/pii/S1876610219300049.

[74] Philip Odonkor and Samuel Ashmore. “Regional performance analysis of residential microgrids: A multi-factor assessment of cost, resilience, and environmental impact”. In: *Energy and Buildings* 332 (2025), p. 115433. ISSN: 0378-7788. DOI: https://doi.org/10.1016/j.enbuild.2025.115433. URL: https://www.sciencedirect.com/science/article/pii/S037877882500163X.

[75] Pietro Tumino. “Frequency Control in a Power System”. In: *EEPowers* (Oct. 2020). Accessed: 2025-06-14. URL: https://eepower.com/technical-articles/frequency-control-in-a-power-system/.

[76] KNMI. *Lijsten van hittegolven*. https://www.knmi.nl/nederland-nu/klimatologie/lijsten/hittegolven. Accessed: 2025-07-16. 2025.

[77] Compendium voor de Leefomgeving (CLO). *Jaarlijkse hoeveelheid neerslag in Nederland, 1910–2022*. https://www.clo.nl/indicatoren/nl050809-jaarlijkse-hoeveelheid-neerslag-in-nederland-1910-2022. Accessed: 2025-07-16. 2023.

[78] Ministerie van Klimaat en Groene Groei. *Zet ook de knop om: anders omgaan met energie*. https://www.rijksoverheid.nl/actueel/nieuws/2024/11/19/zet-ook-de-knop-om-anders-omgaan-met-energie. [Online; accessed 2025-06-14]. Nov. 2024.

[79] André Oerlemans. *Batterijopslag in Nederland verdriedubbeld: eigenaren zonnepanelen kopen massaal thuisbatterijen*. https://www.change.inc/energie/batterijopslag-in-nederland-verdriedubbeld-eigenaren-zonnepanelen-kopen-massaal-thuisbatterijen-41259. [Online; accessed 2025-06-16]. Oct. 2024.

[80] Overstappen.nl. *Dit zijn de grootste leveranciers met een dynamisch energiecontract*. https://www.overstappen.nl/nieuws/dit-zijn-de-grootste-leveranciers-met-een-dynamisch-energiecontract/. Accessed: 2025-07-16. 2025.

[81] Katarzyna Ewa Rollert. "Demand response aggregators as institutional entrepreneurs in the European electricity market". In: *Journal of Cleaner Production* 353 (2022), p. 131501. ISSN: 0959-6526. DOI: https://doi.org/10.1016/j.jclepro.2022.131501. URL: https://www.sciencedirect.com/science/article/pii/S0959652622011222.

[82] Ifigenia Psarra et al. "Exploring residents' perspectives on local energy transition in Northern Netherlands". In: *International Journal of Urban Sustainable Development* 16.1 (2024), pp. 282–298. DOI: 10.1080/19463138.2024.2379334.

[83] Puspendu Ghosh and Mala De. "Resilience-oriented planning for active distribution systems: A probabilistic approach considering regional weather profiles". In: *International Journal of Electrical Power & Energy Systems* 158 (2024), p. 109976. ISSN: 0142-0615. DOI: https://doi.org/10.1016/j.ijepes.2024.109976. URL: https://www.sciencedirect.com/science/article/pii/S0142061524001972.

[84] Santhan Kumar Ch et al. "Improvement of the Resilience of a Microgrid Using Fragility Modeling and Simulation". In: *Journal of Electrical and Computer Engineering* 2022.1 (), p. 3074298. DOI: https://doi.org/10.1155/2022/3074298. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1155/2022/3074298. URL: https://onlinelibrary.wiley.com/doi/abs/10.1155/2022/3074298.

[85] Mathaios Panteli and Pierluigi Mancarella. "Influence of extreme weather and climate change on the resilience of power systems: Impacts and possible mitigation strategies". In: *Electric Power Systems Research* 127 (2015), pp. 259–270. ISSN: 0378-7796. DOI: https://doi.org/10.1016/j.epsr.2015.06.012. URL: https://www.sciencedirect.com/science/article/pii/S037877961500187X.

[86] Tetsuya Wakui et al. "Operation management of residential energy-supplying networks based on optimization approaches". In: *Applied Energy* 183 (2016), pp. 340–357. ISSN: 0306-2619. DOI: https://doi.org/10.1016/j.apenergy.2016.08.171. URL: https://www.sciencedirect.com/science/article/pii/S0306261916312818.

[87] M. Mroueh et al. "Residential Electrical Load Forecasting Based on a Real-Time Evidential Time Series Prediction Method". In: *IEEE Access* 13 (2025), pp. 7448–7461. DOI: 10.1109/ACCESS.2025.3526578.

[88] Ana Lagos et al. "State-of-the-Art Using Bibliometric Analysis of Wind-Speed and -Power Forecasting Methods Applied in Power Systems". In: *Energies* 15.18 (2022). ISSN: 1996-1073. DOI: 10.3390/en15186545. URL: https://www.mdpi.com/1996-1073/15/18/6545.

A
Code

A.1. Reworking Datasets

A.1.1. TMY

```python
# REWORKING TMY DATASET

import pandas as pd
import matplotlib.pyplot as plt

tmy_delft = pd.read_csv(# Path to TMY csv file,
     skiprows=17, nrows=8760,
                          usecols=['time(UTC)','T2m','G(h)','Gb(n)','Gd(h)','WS10m'], index_col
                               =0)

tmy_delft.index = pd.date_range(start="2023-01-01␣00:00", end="2023-12-31␣23:00", freq="h")

tmy_delft.columns = ['temperature', 'global_horizontal', 'dni', 'dhi', 'wind_speed']

tmy_delft['diffuse_fraction'] = tmy_delft['dhi'] / tmy_delft['global_horizontal']


new_index = pd.date_range(start="2023-01-01␣00:00", end="2023-12-31␣23:45", freq="15min")

def resample_to_15min(df):
    # Reindex with new 15-min frequency, initially with NaNs
    df_15min = df.reindex(new_index)

    # Interpolate linearly to fill NaNs in irradiance and weather columns
    df_15min = df_15min.interpolate(method='time')

    # Recalculate diffuse_fraction (DHI / G(h)) for the new data points
    df_15min['diffuse_fraction'] = df_15min['dhi'] / df_15min['global_horizontal']

    return df_15min

tmy_delft_15min = resample_to_15min(tmy_delft)

tmy_delft_15min.to_csv('Delft_15min_TMY.csv')
```

A.1.2. ERA5

```python
# REWORKING ERA5 DATASET

import xarray as xr
import pandas as pd
from datetime import datetime, timedelta
import os
from glob import glob

# Folder containing all the ERA5 .nc files
```

```python
folder_path =  # Adjust path to folder containing all ERA5 .nc files
target_lat = 52.01
target_lon = 4.36

# List all .nc files
nc_files = sorted(glob(os.path.join(folder_path, "*.nc")))

# Use the first file to find the nearest coordinate
first_ds = xr.open_dataset(nc_files[0], decode_times=True)
nearest_lat = float(first_ds['latitude'].sel(latitude=target_lat, method='nearest').values)
nearest_lon = float(first_ds['longitude'].sel(longitude=target_lon, method='nearest').values)

# Collect DataFrames
df_list = []

for file in nc_files:
    ds = xr.open_dataset(file, decode_times=True)

    # Fully reduce to 1D over valid_time by selecting a single point
    t2m_scalar = ds['t2m'].sel(latitude=nearest_lat, longitude=nearest_lon, method="nearest")

    # Ensure it's 1D over time
    if t2m_scalar.ndim > 1:
        t2m_scalar = t2m_scalar.squeeze()

    # Convert to Celsius
    datetimes = ds['valid_time'].values
    t2m_celsius = t2m_scalar.values - 273.15

    # Build DataFrame
    df = pd.DataFrame({
        'datetime': datetimes,
        't2m_C': t2m_celsius
    })

    df_list.append(df)

# Combine and sort
combined_df = pd.concat(df_list).sort_values(by='datetime').reset_index(drop=True)

# Add lat/lon columns
combined_df['latitude'] = nearest_lat
combined_df['longitude'] = nearest_lon

# Save
combined_df.to_csv("t2m_delft.csv", index=False)
```

A.1.3. SARAH-3

```python
# REWORKING SARAH-3 DATASET

import os
from collections import defaultdict
import pandas as pd


folder_path = # Path to folder containing all SARAH-3 .csv files
location_data = defaultdict(list)

for fname in os.listdir(folder_path):
    if not fname.lower().endswith('.csv'):
        continue

    path = os.path.join(folder_path, fname)
    df = pd.read_csv(path, parse_dates=['time'])

    # collect each 'files data under its (lon, lat)
    for (lon, lat), group in df.groupby(['lon', 'lat']):
        location_data[(lon, lat)].append(group)

# concat, interpolate single NaNs, -zeroout nights, compute GHI
```

```
for (lon, lat), chunks in location_data.items():
    df_loc = pd.concat(chunks, ignore_index=True)
    df_loc = df_loc.sort_values("time")

    # compute GHI and diffuse fraction
    df_loc['global_horizontal'] = (
        df_loc['direct_horizontal'] + df_loc['diffuse_horizontal']
    )
    df_loc['diffuse_fraction'] = (
        df_loc['diffuse_horizontal'].where(df_loc['global_horizontal'] > 0, 0)
        / df_loc['global_horizontal'].where(df_loc['global_horizontal'] > 0, 1)
    )

    # save
    out_name = f"location_lon{lon}_lat{lat}.csv"
    df_loc.to_csv(out_name, index=False)
```

A.2. Integration of Datasets & GSEE supply

A.2.1. TMY

```
# TMY GSEE PV Model Run

import numpy as np
import pandas as pd
import xarray as xr
import matplotlib.pyplot as plt
import matplotlib.dates as mdates
from datetime import datetime

import gsee

df = pd.read_csv(# Path to reworked TMY CSV,
     index_col=0)

coords = (52.025, 4.375)
df.index = pd.to_datetime(df.index)
df.index = df.index.tz_localize('UTC')

result_w = gsee.pv.run_model(
    df,
    coords=coords,
    tilt=30,
    azim=180,
    tracking=0,
    capacity=4000,
)

# Find indices where values are above 1750
high_indices = np.where(result_w > 3520)[0]

for idx in high_indices:
    # Check boundaries: skip first and last because they have only one neighbor
    if idx == 0 or idx == len(result_w) - 1:
        continue

    # Average of previous and next time step
    avg = (result_w.iloc[idx - 1] + result_w.iloc[idx + 1]) / 2

    # Replace the high value with average
    result_w.iloc[idx] = avg

result_w.to_csv('Delft_15min_TMY_supply.csv')
```

A.2.2. ERA5 & SARAH-3

```
# INTEGRATING ERA5 and SARAH-3 datasets for GSEE

import numpy as np
import pandas as pd
import xarray as xr
```

```
import matplotlib.pyplot as plt
import matplotlib.dates as mdates
from datetime import datetime
import gsee

# === Load data ===
era5_df = pd.read_csv(
    # Path to the ERA5 CSV file
)
sarah3_df = pd.read_csv(
    # Path to the SARAH-3 CSV file
)

coords = (52.025, 4.375)

# === 1) Make sure all timestamps are -UTCaware ===
era5_df["datetime"] = pd.to_datetime(era5_df["datetime"], errors="coerce", utc=True)

if not pd.api.types.is_datetime64_any_dtype(sarah3_df["time"]):
    sarah3_df["time"] = pd.to_datetime(sarah3_df["time"], errors="coerce", utc=True)

# In many CSVs SARAH-3 is already local time without tz, so tag as UTC:
sarah3_df["time"] = sarah3_df["time"].dt.tz_localize("UTC")

# Prepare ERA5 → 15 min
era5_df = era5_df[["datetime", "t2m_C"]].rename(columns={
    "datetime": "time",
    "t2m_C": "temperature"
})
era5_df = era5_df.set_index("time").sort_index()

era5_15min = (
    era5_df
    .resample("15T")
    .interpolate("time")
    .reset_index()
)

# Prepare SARAH-3 at its original 30 min resolution
sarah3_df = sarah3_df[[
    "time",
    "direct_horizontal",
    "diffuse_horizontal",
    "global_horizontal"
]]
sarah3_df = sarah3_df.set_index("time").sort_index()

# Fill NaNs that are between two valid values

def fill_only_single_na(s: pd.Series) -> pd.Series:
    """
    Given a Series `s` indexed by datetime (30 min apart),
    fill *only* those NaNs that form a block of exactly length=1,
    i.e. s[i] is NaN but both s[i - 30min] and s[i + 30min] are -nonNaN.
    All other NaN blocks (length   2) remain NaN.
    """
    # 1Compute a ""raw -timebased interpolation for comparison
    s_interp = s.interpolate(method="time")

    # Identify which positions in the original are NaN
    is_na = s.isna()

    # Label ""runs of consecutive is_na values so that
    grp = (~is_na).cumsum()

    # For each group, count how many NaNs are in it
    block_size = is_na.groupby(grp).transform("sum")

    # Only fill those NaNs whose block_size == 1
    fillable = is_na & (block_size == 1)

```

```python
    # 6) Build a new Series: keep the original where not fillable; where fillable, take the interpolated value;
    # if part of a larger block_size > 1, leave NaN.
    out = s.copy()
    out[fillable] = s_interp[fillable]
    return out

# Apply to each irradiance column at 30 min resolution
sarah3_f30 = sarah3_df.copy()
for col in ["direct_horizontal", "diffuse_horizontal", "global_horizontal"]:
    sarah3_f30[col] = fill_only_single_na(sarah3_f30[col])

orig_na = sarah3_df["global_horizontal"].isna()
grp = (~orig_na).cumsum()
orig_block_size = orig_na.groupby(grp).transform("sum")
multi_block_mask_30 = orig_na & (orig_block_size >= 2)

# Resample the "30 min with single gaps "filled to 15 min ===
sarah3_15 = (
    sarah3_f30
    .resample("15T")
    .interpolate("time")
)

mask_30_idx = multi_block_mask_30.index
mask_30_vals = multi_block_mask_30.values
# Create a 15 -minfrequency index covering the same span:
full_15_idx = pd.date_range(
    start=mask_30_idx.min(),
    end=mask_30_idx.max(),
    freq="15T",
    tz="UTC"
)
# Build a Series at 15 min by reindex + ffill:
multi_block_mask_15 = (
    pd.Series(mask_30_vals, index=mask_30_idx)
      .reindex(full_15_idx)
      .ffill()       # a 30 -minTrue propagates to its two 15 min slots
      .fillna(False) # anything before the first -30min timestamp is False
)

# Force *those* 15 min slots back to NaN in all irradiance columns ===
for col in ["direct_horizontal", "diffuse_horizontal", "global_horizontal"]:
    sarah3_15.loc[multi_block_mask_15, col] = np.nan

# Force ""night -(20:0003:00 UTC) to zero on the 15 min grid ===
night_mask = (
    (sarah3_15.index.hour >= 20)
    | (sarah3_15.index.hour < 3)
)
sarah3_15.loc[night_mask, ["direct_horizontal", "diffuse_horizontal", "global_horizontal"]] = 0.0

# Recompute diffuse_fraction, forcing 0 at night as well ===
sarah3_15["diffuse_fraction"] = (
    sarah3_15["diffuse_horizontal"]
    / sarah3_15["global_horizontal"]
)
sarah3_15.loc[night_mask, "diffuse_fraction"] = 0.0

# Reset index so ""time becomes a column, then merge with ERA5 ===
sarah3_15 = sarah3_15.reset_index().rename(columns={"index": "time"})

merged_df = pd.merge(
    sarah3_15,
    era5_15min,
    on="time",
    how="inner"
)

# Save to CSV ===
```

```
merged_df.to_csv("full_period_Delft_15min.csv", index=False)
print("Merged␣15-minute␣CSV␣for␣GSEE␣created␣successfully.")

# Ensure 'time' column is datetime with UTC timezone
merged_df['time'] = pd.to_datetime(merged_df['time'], utc=True)

# Set 'time' as index
merged_df = merged_df.set_index('time')

# Double-check: index must be timezone-aware
assert merged_df.index.tz is not None, "Index␣must␣be␣timezone-aware␣(UTC)"

# build a boolean mask for missing direct or diffuse irradiance
missing_irr_mask = merged_df[
    ['direct_horizontal', 'diffuse_horizontal']
].isna().any(axis=1)

# Simulate the GSEE PV model run
result_w = gsee.pv.run_model(
    merged_df,
    coords=coords,
    tilt=30,
    azim=180,
    tracking=0,
    capacity=4000
)
# put NaNs back wherever the inputs were NaN
if isinstance(result_w, xr.DataArray):

    da_mask = xr.DataArray(
        missing_irr_mask,
        coords={'time': merged_df.index},
        dims=['time']
    )
    result_w = result_w.where(~da_mask)
else:

    result_w[missing_irr_mask] = np.nan

result_w.name = 'supply'

result_w.to_csv('full_period_Delft_supply.csv')
```

A.3. Running Scenarios

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import time
from numba import njit, prange
import os
import pickle


# LOAD & PREPARE ONE-YEAR DEMAND & WEATHER & FULL 42-YEAR SUPPLY

# -Oneyear demand pattern
df_base = (
    pd.read_csv(
        # path to the -oneyear demand CSV file,
        parse_dates=["time"]
    )
    .set_index("time")
    .tz_localize(None)
)

df_base["time_key"] = df_base.index.strftime("%m-%d␣%H:%M")

# Full period weather data
```

```
df_weather = (
    pd.read_csv(
        # path to the full period weather CSV file,
        usecols=["time", "temperature"],
        parse_dates=["time"]
    )
    .rename(columns={"time": "time"})
    .set_index("time")
    .tz_localize(None)  # remove +00:00 timezone info
)

df_weather["time_key"] = df_weather.index.strftime("%m-%d␣%H:%M")

# -Fullperiod -(19832024) -PVsupply at 15-minute resolution,
df_supply = pd.read_csv(
    # path to the full period supply CSV file,
    index_col=0,
    parse_dates=True
).rename(columns={"supply": "supply_w"})

# Remove any timezone info
df_supply.index = df_supply.index.tz_convert(None)

# Build helper columns on df_supply:
df_supply["year"] = df_supply.index.year
df_supply["time_key"] = df_supply.index.strftime("%m-%d␣%H:%M")

# LOAD & PREPARE TMY DATA
df_base_tmy = ((
    # path to the TMY -oneyear demand CSV file,
             parse_dates=["time"])
    .set_index("time")
    .tz_localize(None)
)


df_weather_tmy = (
    pd.read_csv(# path to the TMY weather CSV file,
                usecols=["Unnamed:␣0", "temperature"],
                parse_dates=["Unnamed:␣0"])
    .rename(columns={"Unnamed:␣0": "time"})
    .set_index("time")
    .tz_localize(None)
)

df_supply_tmy = pd.read_csv(# Path to the TMY supply file
    , index_col=0, parse_dates=True)

df_supply_tmy.index = df_supply_tmy.index.tz_convert(None)

# Full Period Home Battery Simulation Function
def fp_hb_run_simulation(
    N_households,
    max_shift_steps,
    hp_chance,
    T_set,
    K_building,
    a_cop,
    b_cop,
    HP_capacity,
    battery_capacity_wh,
    max_power_w,
    efficiency,
    timestep_hours
):
    """
    Runs the -residentialmicrogrid simulation over 42 years of supply data,
    using a single ""typical year of demand that is looped/matched onto each supply year.

    Returns:
      - all_deficits       : 1D numpy array of deficit kWh/household (one entry per deficit
```

```
          streak, across all years)
      - all_durations      : 1D numpy array of streak durations in hours (one entry per streak, across all years)
      - import_power_full  : pandas Series indexed by real datetime -(19832024), of import power (W)
      - export_power_full  : pandas Series indexed by real datetime -(19832024), of export power (W)
    """

    # BUILD HOUSEHOLD PROFILES FOR ONE YEAR OF DEMAND ===

    T = len(df_base)
    base_demand = df_base["power_kW"].values  # shape (T,)
    temperature = df_weather.loc[df_base.index, "temperature"].values  # shape matches base_demand

    variation_matrix = np.random.uniform(1 - 0.10, 1 + 0.10, size=(T, N_households))
    shift_steps = np.random.randint(-max_shift_steps, max_shift_steps + 1, size=N_households)

    household_demand = np.empty_like(variation_matrix)
    for i in range(N_households):
        shifted = np.roll(base_demand * variation_matrix[:, i], shift_steps[i])
        household_demand[:, i] = shifted

    has_hp = np.random.rand(N_households) < hp_chance

    # Calculate heat pump power based on temperature and demand
    deltaT = np.clip(T_set - temperature, 0, None)
    Q_demand = K_building * deltaT
    COP = np.clip(a_cop - b_cop * temperature, 1.5, None)
    P_HP_base = np.minimum(HP_capacity, Q_demand / COP)

    P_HP = np.zeros_like(household_demand)
    for i in range(N_households):
        if has_hp[i]:
            P_HP[:, i] = P_HP_base
        else:
            P_HP[:, i] = 0.0

    power_total = household_demand + P_HP

    base_agg_kW = np.sum(household_demand, axis=1)
    hp_agg_kW = np.sum(P_HP, axis=1)
    total_agg_kW = np.sum(power_total, axis=1)

    df_micro = pd.DataFrame({
        "base_agg_kW": base_agg_kW,
        "hp_agg_kW": hp_agg_kW,
        "total_agg_kW": total_agg_kW
    }, index=df_base.index)

    df_micro["time_key"] = df_micro.index.strftime("%m-%d␣%H:%M")

    # build a -perhousehold DataFrame of total load (kW)
    hh_columns = [f"HH_{i+1}" for i in range(N_households)]
    df_households = pd.DataFrame(
        data=power_total,
        index=df_base.index,
        columns=hh_columns
    )
    df_households["time_key"] = df_households.index.strftime("%m-%d␣%H:%M")

    # PREPARE TO LOOP OVER EACH SUPPLY YEAR
    all_deficits = []
    all_durations = []
    import_power_list = []
    export_power_list = []

    # Battery code
    @njit
    def battery_storage_sim_multi_numba(
```

```
        residual_matrix,
        battery_capacity_wh,
        max_power_w,
        efficiency,
        timestep_hours
    ):
        T, N = residual_matrix.shape
        soc = np.zeros(N)
        import_mat = np.zeros((T, N))
        export_mat = np.zeros((T, N))

        for t in range(T):
            for i in range(N):
                resid = residual_matrix[t, i]

                if resid < 0:
                    # Charging case
                    excess_power = -resid
                    incoming_power = min(excess_power, max_power_w)
                    incoming_energy_wh = incoming_power * timestep_hours
                    storable_wh = incoming_energy_wh * efficiency
                    space_left = battery_capacity_wh - soc[i]
                    stored_wh = min(storable_wh, space_left)
                    soc[i] += stored_wh

                    exported_wh = (excess_power * timestep_hours) - (stored_wh / efficiency)
                    export_mat[t, i] = exported_wh / timestep_hours
                    import_mat[t, i] = 0.0

                elif resid > 0:
                    # Discharging case
                    needed_power = resid
                    needed_wh = needed_power * timestep_hours
                    max_discharge_power = min(max_power_w, soc[i] / timestep_hours)
                    outgoing_wh = max_discharge_power * timestep_hours
                    usable_wh = outgoing_wh * efficiency
                    discharge_wh = min(usable_wh, needed_wh)
                    actual_soc_drop = discharge_wh / efficiency
                    soc[i] -= actual_soc_drop

                    import_wh = needed_wh - discharge_wh
                    import_mat[t, i] = import_wh / timestep_hours
                    export_mat[t, i] = 0.0

                # Clamp SoC to [0, capacity]
                soc[i] = min(max(soc[i], 0.0), battery_capacity_wh)

        return import_mat, export_mat

    # Function to compute cumulative deficit streaks
    @njit
    def compute_import_deficit_streaks_numba(grid_import_wh, tolerance):
        deficits = []
        starts = []
        curr_deficit = 0.0
        curr_tol = 0
        temp_start = -1

        for i in range(len(grid_import_wh)):
            val_wh = grid_import_wh[i]
            if np.isnan(val_wh):
                continue
            if val_wh > 0:
                if curr_deficit == 0.0:
                    temp_start = i
                curr_deficit += val_wh / 1000.0
                curr_tol = 0
            else:
                if curr_deficit > 0 and curr_tol < tolerance:
                    curr_tol += 1
                else:
```

```
                    if curr_deficit > 0.0:
                        deficits.append(curr_deficit)
                        starts.append(temp_start)
                    curr_deficit = 0.0
                    curr_tol = 0
                    temp_start = -1

        if curr_deficit > 0.0:
            deficits.append(curr_deficit)
            starts.append(temp_start)

        return deficits, starts

    # Function to compute import streaks
    def compute_import_streaks(grid_import_wh, tolerance=4, timestep_hours=timestep_hours):
        durations = []
        starts = []
        curr_len = 0
        curr_tol = 0
        temp_start = None

        for i, val_wh in enumerate(grid_import_wh):
            if np.isnan(val_wh):
                continue
            if val_wh > 0:
                if curr_len == 0:
                    temp_start = i
                curr_len += 1
                curr_tol = 0
            else:
                if curr_len > 0 and curr_tol < tolerance:
                    curr_len += 1
                    curr_tol += 1
                else:
                    if curr_len > 0:
                        durations.append(curr_len * timestep_hours)
                        starts.append(temp_start)
                    curr_len = 0
                    curr_tol = 0
                    temp_start = None

        if curr_len > 0:
            durations.append(curr_len * timestep_hours)
            starts.append(temp_start)

        return durations, starts

    # MAIN LOOP: FOR EACH SUPPLY YEAR, MERGE & RUN HOME BATTERIES
    unique_years = sorted(df_supply["year"].unique())

    for yr in unique_years:
        supply_yr = df_supply[df_supply["year"] == yr][["time_key", "supply_w"]].copy()

        # Merge supply onto each 'households "-"typicalyear profile
        merged_hh = pd.merge(
            df_households.reset_index(),
            supply_yr,
            on="time_key",
            how="inner",
            sort=False
        )

        merged_hh["year"] = yr
        merged_hh["full_datetime"] = pd.to_datetime(
            merged_hh["year"].astype(str) + "␣" + merged_hh["time_key"],
            format="%Y␣%m-%d␣%H:%M"
        )
        merged_hh = merged_hh.set_index("full_datetime").sort_index()

        # Supply is per household in W → convert to kW for subtraction
        merged_hh["supply_kw_per_hh"] = merged_hh["supply_w"] / 1000.0
```

```
        # === Replace above -perhousehold loop with this: ===

        # Build a 2D array of residuals (shape T × N_households, in W)
        demand_matrix_kW = merged_hh[hh_columns].values
        supply_vec_kW   = merged_hh["supply_kw_per_hh"].values
        residual_matrix = (demand_matrix_kW - supply_vec_kW[:, None]) * 1000.0

        # One call to the vectorized battery for all homes
        import_mat, export_mat = battery_storage_sim_multi_numba(
            residual_matrix,
            battery_capacity_wh=battery_capacity_wh,
            max_power_w=max_power_w,
            efficiency=efficiency,
            timestep_hours=timestep_hours
        )


        # Convert back to DataFrames (indexed by merged_hh.index)
        idx = merged_hh.index
        df_imports_hh = pd.DataFrame(import_mat, index=idx, columns=hh_columns)
        df_exports_hh = pd.DataFrame(export_mat, index=idx, columns=hh_columns)

        # Sum across households to get microgrid import/export (W)
        micro_import_W = df_imports_hh.sum(axis=1)
        micro_export_W = df_exports_hh.sum(axis=1)

        # Compute deficits & durations exactly as before
        micro_import_Wh = micro_import_W * timestep_hours

        deficits_kwh, _ = compute_import_deficit_streaks_numba(
            micro_import_Wh.values,
            tolerance=4,
        )

        deficits_per_household_year = np.array(deficits_kwh) / N_households

        durations_hours, _ = compute_import_streaks(
            micro_import_Wh,
            tolerance=4,
            timestep_hours=timestep_hours
        )

        all_deficits.append(deficits_per_household_year)
        all_durations.append(np.array(durations_hours))

        import_power_list.append(micro_import_W.rename(f"import_{yr}"))
        export_power_list.append(micro_export_W.rename(f"export_{yr}"))


    #  CONCATENATE & RETURN AGGREGATED RESULTS ===
    if len(all_deficits) > 0:
        all_deficits = np.concatenate(all_deficits)
    else:
        all_deficits = np.array([])

    if len(all_durations) > 0:
        all_durations = np.concatenate(all_durations)
    else:
        all_durations = np.array([])

    import_power_full = pd.concat(import_power_list, axis=0).sort_index()
    export_power_full = pd.concat(export_power_list, axis=0).sort_index()


    micro_import_Wh_full = import_power_full * timestep_hours

    # Reuse your functions
    total_deficits, deficit_starts = compute_import_deficit_streaks_numba(
        micro_import_Wh_full.values,
        tolerance=4,
```

```
    )
    total_durations, duration_starts = compute_import_streaks(
        micro_import_Wh_full,
        tolerance=4,
        timestep_hours=timestep_hours
    )

    return all_deficits, all_durations, import_power_full, export_power_full


# FULL PERIOD COMMUNITY BATTERY FUNCTION
def fp_cb_run_simulation(
    N_households,
    max_shift_steps,
    hp_chance,
    T_set,
    K_building,
    a_cop,
    b_cop,
    HP_capacity,
    battery_capacity_wh,
    max_power_w,
    efficiency,
    timestep_hours
):
    """
    Runs the -residentialmicrogrid simulation over 42 years of supply data,
    using a single ""typical year of demand that is looped/matched onto each supply year.

    Returns:
      - all_deficits       : 1D numpy array of deficit kWh/household (one entry per deficit streak, across all years)
      - all_durations      : 1D numpy array of streak durations in hours (one entry per streak, across all years)
      - import_power_full  : pandas Series indexed by real datetime -(19832024), of import power (W)
      - export_power_full  : pandas Series indexed by real datetime -(19832024), of export power (W)
    """

    # BUILD HOUSEHOLD PROFILES FOR THAT  YEAR OF DEMAND

    T = len(df_base)
    base_demand = df_base["power_kW"].values
    temperature = df_weather.loc[df_base.index, "temperature"].values

    variation_matrix = np.random.uniform(1 - 0.10, 1 + 0.10, size=(T, N_households))
    shift_steps = np.random.randint(-max_shift_steps, max_shift_steps + 1, size=N_households)

    household_demand = np.empty_like(variation_matrix)
    for i in range(N_households):
        shifted = np.roll(base_demand * variation_matrix[:, i], shift_steps[i])
        household_demand[:, i] = shifted

    has_hp = np.random.rand(N_households) < hp_chance

    # Heat pump power based on temperature and demand
    deltaT = np.clip(T_set - temperature, 0, None)
    Q_demand = K_building * deltaT
    COP = np.clip(a_cop - b_cop * temperature, 1.5, None)
    P_HP_base = np.minimum(HP_capacity, Q_demand / COP)

    P_HP = np.zeros_like(household_demand)
    for i in range(N_households):
        if has_hp[i]:
            P_HP[:, i] = P_HP_base
        else:
            P_HP[:, i] = 0.0

    power_total = household_demand + P_HP

```

```
    base_agg_kW = np.sum(household_demand, axis=1)
    hp_agg_kW = np.sum(P_HP, axis=1)
    total_agg_kW = np.sum(power_total, axis=1)

    df_micro = pd.DataFrame({
        "base_agg_kW": base_agg_kW,
        "hp_agg_kW": hp_agg_kW,
        "total_agg_kW": total_agg_kW
    }, index=df_base.index)

    df_micro["time_key"] = df_micro.index.strftime("%m-%d %H:%M")

    # PREPARE TO LOOP OVER EACH SUPPLY YEAR
    all_deficits = []
    all_durations = []
    import_power_list = []
    export_power_list = []

    # Function to run battery storage simulation
    @njit
    def battery_storage_sim_numba(
        residual_w,
        battery_capacity_wh,
        max_power_w,
        efficiency,
        timestep_hours
    ):
        T = residual_w.shape[0]

        battery_storage = np.zeros(T)
        battery_discharge = np.zeros(T)
        grid_export = np.zeros(T)
        grid_import = np.zeros(T)

        battery_state = 0.0

        for t in range(T):
            power_w = residual_w[t]

            if np.isnan(power_w):
                battery_storage[t] = battery_state
                battery_discharge[t] = 0.0
                grid_export[t] = np.nan
                grid_import[t] = np.nan
                continue

            if power_w < 0:
                # Charging case
                total_excess_power_w = -power_w
                incoming_power_w = min(total_excess_power_w, max_power_w)
                incoming_energy_wh = incoming_power_w * timestep_hours
                storable_energy_wh = incoming_energy_wh * efficiency
                space_left_wh = battery_capacity_wh - battery_state
                stored_wh = min(storable_energy_wh, space_left_wh)
                battery_state += stored_wh

                total_excess_energy_wh = total_excess_power_w * timestep_hours
                used_energy_wh = stored_wh / efficiency
                main_grid_export_wh = total_excess_energy_wh - used_energy_wh

                main_grid_import_wh = 0.0
                discharge_wh = 0.0

            elif power_w > 0:
                # Discharging case
                needed_power_w = power_w
                needed_energy_wh = needed_power_w * timestep_hours
                max_discharge_power_w = min(max_power_w, battery_state / timestep_hours)
                outgoing_power_w = max_discharge_power_w
                outgoing_energy_wh = outgoing_power_w * timestep_hours
                usable_energy_wh = outgoing_energy_wh * efficiency
```

```
                discharge_wh = min(usable_energy_wh, needed_energy_wh)
                actual_discharge_wh = discharge_wh / efficiency
                battery_state -= actual_discharge_wh
                main_grid_import_wh = needed_energy_wh - discharge_wh
                main_grid_export_wh = 0.0
            else:
                # Zero demand
                discharge_wh = 0.0
                main_grid_export_wh = 0.0
                main_grid_import_wh = 0.0

            # Clamp battery state
            battery_state = max(0.0, min(battery_state, battery_capacity_wh))

            battery_storage[t] = battery_state
            battery_discharge[t] = discharge_wh / timestep_hours
            grid_export[t] = main_grid_export_wh / timestep_hours
            grid_import[t] = main_grid_import_wh / timestep_hours

        return battery_storage, battery_discharge, grid_export, grid_import

    # Function to compute cumulative deficit streaks
    @njit
    def compute_import_deficit_streaks_numba(grid_import_w, tolerance=4, timestep_hours=0.25)
        :
        deficits = []
        starts = []
        curr_deficit = 0.0
        curr_tol = 0
        temp_start = -1

        for i in range(len(grid_import_w)):
            val_wh = grid_import_w[i]
            if np.isnan(val_wh):
                continue
            if val_wh > 0:
                if curr_deficit == 0.0:
                    temp_start = i
                curr_deficit += val_wh / 1000.0 * timestep_hours  # Convert to kWh
                curr_tol = 0
            else:
                if curr_deficit > 0 and curr_tol < tolerance:
                    curr_tol += 1
                else:
                    if curr_deficit > 0.0:
                        deficits.append(curr_deficit)
                        starts.append(temp_start)
                    curr_deficit = 0.0
                    curr_tol = 0
                    temp_start = -1

        if curr_deficit > 0.0:
            deficits.append(curr_deficit)
            starts.append(temp_start)

        return deficits, starts

    # Function to compute import duration streaks
    @njit
    def compute_import_streaks_numba(grid_import_w, tolerance=4):
        durations = []
        starts = []
        curr_len = 0
        curr_tol = 0
        temp_start = -1

        for i in range(len(grid_import_w)):
            val_w = grid_import_w[i]
            # Workaround for NaN-check (Numba-safe)
            if val_w != val_w:
                continue
```

```python

            if val_w > 0:
                if curr_len == 0:
                    temp_start = i
                curr_len += 1
                curr_tol = 0
            else:
                if curr_len > 0 and curr_tol < tolerance:
                    curr_len += 1
                    curr_tol += 1
                else:
                    if curr_len > 0:
                        durations.append(curr_len)  # NO timestep_hours multiplier here
                        starts.append(temp_start)
                    curr_len = 0
                    curr_tol = 0
                    temp_start = -1

        if curr_len > 0:
            durations.append(curr_len)
            starts.append(temp_start)

        return durations, starts

    # MAIN LOOP: FOR EACH SUPPLY YEAR, MERGE & RUN BATTERY ===
    unique_years = sorted(df_supply["year"].unique())

    for yr in unique_years:
        supply_yr = df_supply[df_supply["year"] == yr][["time_key", "supply_w"]].copy()

        demand_one_year = df_micro.reset_index()[["time", "time_key", "total_agg_kW"]].copy()
        merged = pd.merge(
            demand_one_year,
            supply_yr,
            on="time_key",
            how="inner"
        )

        merged["year"] = yr
        merged["full_datetime"] = pd.to_datetime(
            merged["year"].astype(str) + " " + merged["time_key"],
            format="%Y %m-%d %H:%M"
        )
        merged = merged.set_index("full_datetime").sort_index()

        merged["supply_kw"] = merged["supply_w"] / 1000.0 * N_households

        merged["residual_kW"] = merged["total_agg_kW"] - merged["supply_kw"]
        residual_w = merged["residual_kW"] * 1000.0

        battery_storage, battery_discharge, grid_export, grid_import = battery_storage_sim_numba(
            residual_w.values,
            battery_capacity_wh,
            max_power_w,
            efficiency,
            timestep_hours
        )

        battery_df_yr = pd.DataFrame({
            "Battery SoC (Wh)": battery_storage,
            "Battery Discharge (W)": battery_discharge,
            "Export to Grid (W)": grid_export,
            "Import from Grid (W)": grid_import,
        }, index=merged.index)


        merged = merged.join(battery_df_yr)

        deficits_kwh, _ = compute_import_deficit_streaks_numba(
            merged["Import from Grid (W)"].values,
```

```
            4,
            timestep_hours
        )
        deficits_per_household_year = np.array(deficits_kwh) / N_households

        durations, _ = compute_import_streaks_numba(
            merged["Import␣from␣Grid␣(W)"].values,
            4,
        )
        durations_hours = [d * 0.25 for d in durations]

        all_deficits.append(deficits_per_household_year)
        all_durations.append(np.array(durations_hours))

        import_power_list.append(merged["Import␣from␣Grid␣(W)"].rename(f"import_{yr}"))
        export_power_list.append(merged["Export␣to␣Grid␣(W)"].rename(f"export_{yr}"))

    # CONCATENATE & RETURN AGGREGATED RESULTS
    if len(all_deficits) > 0:
        all_deficits = np.concatenate(all_deficits)
    else:
        all_deficits = np.array([])

    if len(all_durations) > 0:
        all_durations = np.concatenate(all_durations)
    else:
        all_durations = np.array([])

    import_power_full = pd.concat(import_power_list, axis=0).sort_index()
    export_power_full = pd.concat(export_power_list, axis=0).sort_index()

    return all_deficits, all_durations, import_power_full, export_power_full

# TMY COMMUNITY BATTERY FUNCTION
def tmy_cb_run_simulation(
    N_households,
    max_shift_steps,
    hp_chance=hp_chance,
    T_set=T_set,
    K_building=K_building,
    a_cop=a_cop,
    b_cop=-b_cop,
    HP_capacity=HP_capacity,
    battery_capacity_wh=battery_capacity_wh,
    max_power_w=max_power_w,
    efficiency=efficiency,
    timestep_hours=timestep_hours
):

    # Function to generate random variation
    def generate_random_variation_profile(df_base_tmy, variation_pct=0.10):
        """
        Generate a new demand profile by randomly varying df_base['power_kW']
        by +/- variation_pct (default 10%).

        Parameters:
        - df_base: pandas DataFrame with a 'power_kW' column
        - variation_pct: maximum percentage variation (e.g. 0.10 for ±10%)

        Returns:
        - df_out: pandas DataFrame with a new column 'power_stochastic'
        """
        n = len(df_base_tmy)
        # Random variation factors between (1 - variation_pct) and (1 + variation_pct)
        random_factors = np.random.uniform(1 - variation_pct, 1 + variation_pct, size=n)

        df_out = df_base_tmy.copy()
        df_out['power_stochastic'] = df_out['power_kW'] * random_factors

        # Clip values below 0 just in case
        df_out['power_stochastic'] = df_out['power_stochastic'].clip(lower=0)
```

```python
        return df_out

    # Function to shift load times
    def shift_profile(df, max_shift_steps=max_shift_steps):
        df = df.copy()
        df = df.sort_values("time")

        shift = np.random.randint(-max_shift_steps, max_shift_steps + 1)

        df["power_stochastic_shifted"] = np.roll(df["power_stochastic"].values, shift)

        return df

    # MAIN LOOP: BUILD HOUSEHOLD PROFILES
    house_profiles = []

    for i in range(N_households):
        # Generate random variation profile
        df_i = generate_random_variation_profile(df_base_tmy)

        # Shift to de-synchronize peaks
        df_i = shift_profile(df_i)

        # Merge in outdoor temperature
        df_i = df_i.merge(df_weather_tmy["temperature"].reset_index(),
                          on="time", how="left")

        # Randomly assign HP
        has_hp = (np.random.rand() < hp_chance)

        df_i["has_heat_pump"] = has_hp

        # Heat pump calculations
        if has_hp:
            deltaT   = np.clip(T_set - df_i["temperature"], 0, None)
            Q_demand = K_building * deltaT
            COP      = np.clip(a_cop - b_cop * df_i["temperature"], 1.5, None)
            P_HP     = np.minimum(HP_capacity, Q_demand / COP)
            df_i["P_HP_kW"] = P_HP
        else:
            df_i["P_HP_kW"] = 0.0

        df_i["power_total_kW"] = df_i["power_stochastic_shifted"] + df_i["P_HP_kW"]
        df_i["household"] = f"HH_{i+1}"
        house_profiles.append(
            df_i[["time","household","has_heat_pump",
                "power_stochastic_shifted","P_HP_kW","power_total_kW"]]
        )

    df_all = pd.concat(house_profiles, ignore_index=True)

    # AGGREGATE TO MICROGRID LEVEL
    df_micro = (
        df_all
        .groupby("time")[["power_stochastic_shifted","P_HP_kW","power_total_kW"]]
        .sum()
        .rename(columns={
            "power_stochastic_shifted": "base_agg_kW",
            "P_HP_kW":                 "hp_agg_kW",
            "power_total_kW":          "total_agg_kW"
        })
        .reset_index()
    )

    df_micro = df_micro.set_index("time")
    df_micro = df_micro.copy()
    df_combined = df_micro.join(df_supply_tmy.rename(columns={"0": "supply_w"}), how="inner")

    # Convert units if needed, e.g., if df_micro is in kW and df_pv in W
    df_combined['supply_kw'] = df_combined['supply_w'] / 1000.0 * N_households
```

```
    df_combined['residual_kW'] = df_combined['total_agg_kW'] - df_combined['supply_kw']

    # Convert to Watts for battery logic
    residual_w = df_combined['residual_kW'] * 1000

    # Function for battery storage simulation
    def battery_storage_sim(residual_w, battery_capacity_wh=battery_capacity_wh, max_power_w=
        max_power_w, efficiency=efficiency, timestep_hours=timestep_hours):
        battery_state = 0
        battery_storage = []
        grid_export = []
        battery_discharge = []
        grid_import = []

        for power_w in residual_w:
            if power_w < 0:
                # Excess supply -> charge battery
                total_excess_power_w = -power_w
                incoming_power_w = min(total_excess_power_w, max_power_w)
                incoming_energy_wh = incoming_power_w * timestep_hours
                storable_energy_wh = incoming_energy_wh * efficiency
                space_left_wh = battery_capacity_wh - battery_state
                stored_wh = min(storable_energy_wh, space_left_wh)
                battery_state += stored_wh

                # Total energy available from excess supply
                total_excess_energy_wh = total_excess_power_w * timestep_hours

                # Calculate what wasn't stored at all - lost to power limit or full battery
                used_energy_wh = stored_wh / efficiency
                main_grid_export_wh = total_excess_energy_wh - used_energy_wh

                main_grid_import_wh = 0
                discharge_wh = 0

            elif power_w > 0:
                # Deficit -> discharge battery
                needed_power_w = power_w
                needed_energy_wh = needed_power_w * timestep_hours
                max_discharge_power_w = min(max_power_w, battery_state / timestep_hours)
                outgoing_power_w = max_discharge_power_w
                outgoing_energy_wh = outgoing_power_w * timestep_hours
                usable_energy_wh = outgoing_energy_wh * efficiency
                discharge_wh = min(usable_energy_wh, needed_energy_wh)
                actual_discharge_wh = discharge_wh / efficiency
                battery_state -= actual_discharge_wh
                main_grid_import_wh = needed_energy_wh - discharge_wh
                main_grid_export_wh = 0

            else:
                discharge_wh = 0
                main_grid_export_wh = 0
                main_grid_import_wh = 0

            battery_state = max(0, min(battery_state, battery_capacity_wh))

            battery_storage.append(battery_state)
            battery_discharge.append(discharge_wh / timestep_hours)
            grid_export.append(main_grid_export_wh / timestep_hours)
            grid_import.append(main_grid_import_wh / timestep_hours)

        return pd.DataFrame({
            'Battery SoC (Wh)': battery_storage,
            'Battery Discharge (W)': battery_discharge,
            'Export to Grid (W)': grid_export,
            'Import from Grid (W)': grid_import,
        }, index=residual_w.index)

    battery_df = battery_storage_sim(residual_w)

    # Merge into combined dataframe
```

```
    df_combined = df_combined.join(battery_df)

    df_combined['Import from Grid (Wh)'] = df_combined['Import from Grid (W)'] * timestep_hours
    import_energy_series = df_combined['Import from Grid (Wh)']  # this is now Wh at each timestep

    # Compute cumulative deficit streaks (kWh per streak):
    def compute_import_deficit_streaks(grid_import_wh, tolerance=4, timestep_hours=0.25):
        """
        Input:
          grid_import_wh : pd.Series or 1D array of [Wh] at each timestep
        Output:
          deficits : list of (kWh) imported during each -continuousimport streak
          starts   : list of the index where each streak began
        """
        deficits = []
        starts = []
        curr_deficit = 0.0  # in kWh
        curr_tol = 0
        temp_start = None

        for i, val_wh in enumerate(grid_import_wh):
            if np.isnan(val_wh):
                continue

            if val_wh > 0:
                # If this is the first -positiveimport step in a new streak:
                if curr_deficit == 0.0:
                    temp_start = i
                # Add val_wh (Wh) ÷ 1000 → kWh
                curr_deficit += val_wh / 1000.0
                curr_tol = 0
            else:
                # val_wh == 0 → no import this step
                if curr_deficit > 0 and curr_tol < tolerance:
                    # still ""in the streak (we allow up to `tolerance` zeros embedded)
                    curr_tol += 1
                    # but do NOT add any energy
                else:
                    # close out the previous streak (if it exists)
                    if curr_deficit > 0.0:
                        deficits.append(curr_deficit)
                        starts.append(temp_start)
                    curr_deficit = 0.0
                    curr_tol = 0
                    temp_start = None

        # If we ended while still in a -positiveimport streak:
        if curr_deficit > 0.0:
            deficits.append(curr_deficit)
            starts.append(temp_start)

        return deficits, starts

    deficits_kwh, deficit_starts = compute_import_deficit_streaks(
        import_energy_series,
        tolerance=4,
        timestep_hours=timestep_hours
    )
    # deficits_kwh is a list of [kWh] per streak. To get "per "household:
    deficits_per_household = np.array(deficits_kwh) / N_households

    # Compute import period duration streaks (in hours) for the same import series:
    def compute_import_streaks(grid_import_w, tolerance=4, timestep_hours=0.25):
        """
        Input:
          grid_import_wh : pd.Series or 1D array of [Wh] per step
        Output:
          durations : list of (hours) for each -continuousimport streak
          starts    : list of the index where each streak began
```

```
        """
        durations = []
        starts = []
        curr_len = 0       # number of steps in this streak
        curr_tol = 0       # number of zeros we've tolerated so far
        temp_start = None

        for i, val_w in enumerate(grid_import_w):
            val_w = grid_import_w[i]
            # Workaround for NaN-check (Numba-safe)
            if np.isnan(val_w):
                continue

            if val_w > 0:
                if curr_len == 0:
                    temp_start = i
                curr_len += 1
                curr_tol = 0
            else:
                if curr_len > 0 and curr_tol < tolerance:
                    curr_len += 1
                    curr_tol += 1
                else:
                    if curr_len > 0:
                        durations.append(curr_len)  # NO timestep_hours multiplier here
                        starts.append(temp_start)
                    curr_len = 0
                    curr_tol = 0
                    temp_start = -1

        if curr_len > 0:
            durations.append(curr_len)
            starts.append(temp_start)

        return durations, starts

    # Existing durations and starts from compute_import_streaks:
    durations, starts = compute_import_streaks(
        import_energy_series,
        tolerance=4,
        timestep_hours=timestep_hours
    )
    durations_hours = [d * timestep_hours for d in durations]

    import_power_w = df_combined['Import from Grid (W)']
    export_power_w = df_combined['Export to Grid (W)']

    return deficits_per_household, np.array(durations_hours), import_power_w, export_power_w

# Function to run the TMY home battery simulation
def tmy_hb_run_simulation(
    N_households,
    max_shift_steps,
    hp_chance=hp_chance,
    T_set=T_set,
    K_building=K_building,
    a_cop=a_cop,
    b_cop=-b_cop,
    HP_capacity=HP_capacity,
    battery_capacity_wh=battery_capacity_wh,
    max_power_w=max_power_w,
    efficiency=efficiency,
    timestep_hours=timestep_hours
):
    variation_pct = 0.10  # 10% random variation

    # Function to compute cumulative deficit streaks
    @njit
    def compute_import_deficit_streaks_numba(grid_import_wh, tolerance, timestep_hours):
        deficits = []
        starts = []
```

```
        curr_deficit = 0.0
        curr_tol = 0
        temp_start = -1

        for i in range(len(grid_import_wh)):
            val_wh = grid_import_wh[i]
            if np.isnan(val_wh):
                continue
            if val_wh > 0:
                if curr_deficit == 0.0:
                    temp_start = i
                curr_deficit += val_wh / 1000.0  # Wh to kWh
                curr_tol = 0
            else:
                if curr_deficit > 0 and curr_tol < tolerance:
                    curr_tol += 1
                else:
                    if curr_deficit > 0.0:
                        deficits.append(curr_deficit)
                        starts.append(temp_start)
                    curr_deficit = 0.0
                    curr_tol = 0
                    temp_start = -1

        if curr_deficit > 0.0:
            deficits.append(curr_deficit)
            starts.append(temp_start)

        return deficits, starts

    # Function to compute import period duration streaks
    @njit
    def compute_import_streaks_numba(grid_import_wh, tolerance, timestep_hours):
        durations = []
        starts = []
        curr_len = 0
        curr_tol = 0
        temp_start = -1

        for i in range(len(grid_import_wh)):
            val_wh = grid_import_wh[i]
            if np.isnan(val_wh):
                continue
            if val_wh > 0:
                if curr_len == 0:
                    temp_start = i
                curr_len += 1
                curr_tol = 0
            else:
                if curr_len > 0 and curr_tol < tolerance:
                    curr_len += 1
                    curr_tol += 1
                else:
                    if curr_len > 0:
                        durations.append(curr_len * timestep_hours)
                        starts.append(temp_start)
                    curr_len = 0
                    curr_tol = 0
                    temp_start = -1

        if curr_len > 0:
            durations.append(curr_len * timestep_hours)
            starts.append(temp_start)

        return durations, starts

    # Function to run the battery storage simulation
    @njit(parallel=True)
    def battery_sim_all(
        residuals,
        battery_capacity_wh,
```

```
        max_power_w,
        efficiency,
        timestep_hours
    ):
        N, T = residuals.shape
        storage_out   = np.zeros((N, T))  # Wh
        discharge_out = np.zeros((N, T))  # W
        export_out    = np.zeros((N, T))  # W
        import_out    = np.zeros((N, T))  # W

        for i in prange(N):
            battery_state = 0.0
            for t in range(T):
                power_w = residuals[i, t]

                if power_w < 0.0:
                    # Excess → charge
                    total_excess_power_w = -power_w
                    incoming_power_w = min(total_excess_power_w, max_power_w)
                    incoming_energy_wh = incoming_power_w * timestep_hours
                    storable_wh = incoming_energy_wh * efficiency
                    space_left = battery_capacity_wh - battery_state
                    stored_wh = storable_wh if storable_wh < space_left else space_left
                    battery_state += stored_wh

                    total_excess_wh = total_excess_power_w * timestep_hours
                    used_wh = stored_wh / efficiency
                    main_grid_export_wh = total_excess_wh - used_wh

                    main_grid_import_wh = 0.0
                    discharge_wh = 0.0
                elif power_w > 0.0:
                    # Deficit → discharge
                    needed_energy_wh = power_w * timestep_hours
                    max_discharge_power_w = battery_state / timestep_hours
                    if max_discharge_power_w > max_power_w:
                        max_discharge_power_w = max_power_w
                    outgoing_power_w = max_discharge_power_w
                    outgoing_wh = outgoing_power_w * timestep_hours
                    usable_wh = outgoing_wh * efficiency
                    if usable_wh < needed_energy_wh:
                        discharge_wh = usable_wh
                    else:
                        discharge_wh = needed_energy_wh
                    actual_discharge_wh = discharge_wh / efficiency
                    battery_state -= actual_discharge_wh
                    main_grid_import_wh = needed_energy_wh - discharge_wh
                    main_grid_export_wh = 0.0
                else:
                    discharge_wh = 0.0
                    main_grid_export_wh = 0.0
                    main_grid_import_wh = 0.0

                # Clamp battery_state
                if battery_state < 0.0:
                    battery_state = 0.0
                elif battery_state > battery_capacity_wh:
                    battery_state = battery_capacity_wh

                storage_out[i, t]   = battery_state
                discharge_out[i, t] = discharge_wh / timestep_hours
                export_out[i, t]    = main_grid_export_wh / timestep_hours
                import_out[i, t]    = main_grid_import_wh / timestep_hours

        return storage_out, discharge_out, export_out, import_out

    # MAIN BLOCK: BUILD ALL HOUSEHOLDS AT ONCE

    # Extract base power series and time index
    base_power = df_base_tmy["power_kW"].values
    times      = df_base_tmy.index
```

```
    T           = base_power.shape[0]

    # Generate all random variation factors
    rng = np.random.default_rng()
    random_factors = rng.uniform(
        1 - variation_pct,
        1 + variation_pct,
        size=(N_households, T)
    )

    # Compute power_stochastic
    power_stochastic = base_power[None, :] * random_factors
    power_stochastic = np.clip(power_stochastic, 0.0, None)

    # Apply random shift
    shifts = rng.integers(-max_shift_steps, max_shift_steps + 1, size=N_households)
    power_stochastic_shifted = np.empty_like(power_stochastic)
    for i in range(N_households):
        power_stochastic_shifted[i, :] = np.roll(power_stochastic[i, :], shifts[i])

    # Weather temperature array (length T)
    temp_arr = df_weather_tmy["temperature"].reindex(times).values

    # Random mask
    has_hp_mask = rng.random(N_households) < hp_chance

    # Compute P_HP_kW for all households & times
    deltaT = np.clip(T_set - temp_arr, 0.0, None)
    Q_base = K_building * deltaT
    COP_vec = np.clip(a_cop - b_cop * temp_arr, 1.5, None)

    Q_demand_2d = np.repeat(Q_base[None, :], N_households, axis=0)
    COP_2d     = np.repeat(COP_vec[None, :], N_households, axis=0)
    HP_cap_arr = np.full((N_households, T), HP_capacity)

    P_HP_kW = np.where(
        has_hp_mask[:, None],
        np.minimum(HP_cap_arr, Q_demand_2d / COP_2d),
        0.0
    )

    # Total power demand per household
    power_total_kW = power_stochastic_shifted + P_HP_kW

    # Supply per household
    supply_1d   = df_supply_tmy["0"].reindex(times).values / 1000.0
    supply_arr  = np.repeat(supply_1d[None, :], N_households, axis=0)

    # Keep list of household names for later
    hh_names = [f"HH_{i+1}" for i in range(N_households)]

    # Compute residuals (N × T) in Watts
    residuals_2d = (power_total_kW - supply_arr) * 1000.0

    # RUN BATTERY SIMULATION FOR ALL HOUSEHOLDS AT ONCE
    storage_2d, discharge_2d, export_2d, import_2d = battery_sim_all(
        residuals_2d,
        battery_capacity_wh,
        max_power_w,
        efficiency,
        timestep_hours
    )


    hh_repeat  = np.repeat(np.arange(N_households), T)    # length = N*T
    time_tile  = np.tile(times.values, N_households)      # length = N*T

    df_out = pd.DataFrame({
        "household_idx":        hh_repeat,
        "time":                 time_tile,
        "Battery SoC (Wh)":     storage_2d.ravel(),
```

```python
        "Battery Discharge (W)": discharge_2d.ravel(),
        "Export to Grid (W)":    export_2d.ravel(),
        "Import from Grid (W)":  import_2d.ravel(),
        "power_total_kW":        power_total_kW.ravel(),
        "supply_kW":             supply_arr.ravel()
    })


    df_out["household"] = pd.Categorical(
        [hh_names[i] for i in df_out["household_idx"]],
        categories=hh_names
    )


    df_all_batt = (
        df_out
        .set_index(["time", "household"])
        .sort_index()
    )

    # AGGREGATE AT MICROGRID LEVEL
    df_import_export_agg = df_all_batt.groupby(level="time")[['Import from Grid (W)', 'Export to Grid (W)']].sum()
    import_energy_series = df_import_export_agg['Import from Grid (W)'] * timestep_hours  # Wh

    deficits_kwh, deficit_starts = compute_import_deficit_streaks_numba(
        import_energy_series.values,
        tolerance=4,
        timestep_hours=timestep_hours
    )

    durations_hours, duration_starts = compute_import_streaks_numba(
        import_energy_series.values,
        tolerance=4,
        timestep_hours=timestep_hours
    )

    deficits_per_household = np.array(deficits_kwh) / N_households

    import_power_w = df_import_export_agg['Import from Grid (W)']
    export_power_w = df_import_export_agg['Export to Grid (W)']

    return deficits_per_household, np.array(durations_hours), import_power_w, export_power_w


# VARIABLES FOR SIMULATION
N_households=300

max_shift_steps=8
timestep_hours=0.25

# Home battery specifications
battery_capacity_wh=8800
max_power_w=5000
efficiency=0.95

# Community battery specifications
battery_capacity_wh=7530*N_households
max_power_w=1880*N_households
efficiency=0.95

# Heat pump specifications
T_set       = 18.0
K_building  = 0.285
HP_capacity = 7.5
a_cop, b_cop = 3.5, 0.07
hp_chance   = 0

# RUNNING THE SCENARIOS
n_runs = 10
```

```
all_deficits = []
all_durations = []
all_import_powers = []
all_export_powers = []

for _ in range(n_runs):
    start = time.time()

    deficits, durations, import_power_w, export_power_w = fp_cb_run_simulation(
        N_households=N_households,
        max_shift_steps=max_shift_steps,
        hp_chance=hp_chance,
        T_set=T_set,
        K_building=K_building,
        a_cop=a_cop,
        b_cop=b_cop,
        HP_capacity=HP_capacity,
        battery_capacity_wh=battery_capacity_wh,
        max_power_w=max_power_w,
        efficiency=efficiency,
        timestep_hours=timestep_hours
    )

    end = time.time()
    print(f"Simulation {_ + 1} took {end - start:.2f} seconds")


    all_deficits.append(deficits)
    all_durations.append(durations)
    all_import_powers.append(import_power_w)
    all_export_powers.append(export_power_w)

# Now combine across runs for "-importenergy deficits (kWh/household)"
all_deficits_combined = np.concatenate(all_deficits)  # 1D array of length = total streaks
    across all runs

all_durations_combined = np.concatenate(all_durations)

import_power_all = pd.concat(all_import_powers, axis=0)
import_power_per_household = import_power_all / N_households / 1000  # Normalize to per
    household in kW

export_power_all = pd.concat(all_export_powers, axis=0)
export_power_per_household = export_power_all / N_households  / 1000 # Normalize to per
    household in kW

# Convert to arrays just once
imp_power = import_power_per_household.values
exp_power = export_power_per_household.values

# Filter power > 0
imp_power_pos = imp_power[imp_power > 0]
exp_power_pos = exp_power[exp_power > 0]

# Compute ramp rates (differences between consecutive time steps)
imp_ramp = np.abs(np.diff(imp_power))
exp_ramp = np.abs(np.diff(exp_power))

# Filter ramp rates > 0
imp_ramp_pos = imp_ramp[imp_ramp > 0]
exp_ramp_pos = exp_ramp[exp_ramp > 0]
```

B

Literature Review Overview

Table B.1: Overview of Literature Review Findings: Microgrid Resilience

Author(s)	Year	Location	MG generation source	Main findings
Ming Kwok et al. [21]	2012	Singapore	Renewable + fossil fuels	Rainfall negatively impacts solar energy collected (7.61% – 14.86%). Light rainfall has a positive effect (3%) on wind energy, but heavy rainfall has a positive effect (24%).
Ghosh & De [83]	2024	-	Unspecified	Adopting a microgrid formation reduces weighted load cost with decrease of 52.62%, 69.1%, and 75.04% compared to the base case for normal, high, and extreme weather event scenarios, respectively.
Chowdhury & Zhang [22]	2024	-	Renewable + microturbine	Leaving out the smoke effect during wildfires results in an overestimation of solar energy output. The addition of quick start units drastically decreases load shedding costs.
Newman et al. [18]	2020	United States	PV + back-up generator	The differences in fuel consumption predicted using the Alternative Solar Profiles (ASP) and TMY solar profiles algorithms were significant, with, on average, a need for 9.4% more fuel predicted using the ASP profiles for the most severe outage period across all load profiles and locations, and 30% more fuel required for the most severe period.
Ch et al. [84]	2022	-	Unspecified	Load shedding increases microgrid resilience. Smaller microgrids are more resilient than large microgrids.
Wang et al. [24]	2024	-	Renewable + microturbine	By optimizing energy storage capacity, implementing load response mechanisms, and integrating renewable energy sources, microgrids can become more efficient, reliable, and resilient in the face of various challenges.

Laws et al. [19]	2018	United States	PV	Generally, accounting for the value of resilience results in larger PV and BESS when minimizing life cycle costs. If the cost to island a system does exceed the islandable premium, then the minimum cost solution is to incur the outage cost. Under the current assumptions for storage prices and utility rate tariffs, battery systems tend to only be economical in locations with relatively high demand charges.
Sepúlveda-Mora & Hegedus [20]	2022	United States	Renewables + Generator	A metric for resilience is provided. Resilience in a hospital is 40% higher than in hotels. When the value of lost load is included in the economic analysis, battery systems with autonomy as large as 12 h in combination with PV, wind and generator are more cost-effective than the baseline configuration.
Panteli & Mancarella [85]	2015	-	Unspecified	A framework has been developed which can be used as a basis for developing weather-related resilience studies.
Mohamed et al. [23]	2019	-	Unspecified	From a literature review, a framework is presented for proactive resilience of power systems with a spotlight on the extreme weather events and their effect. From the framework, several strategies are reviewed.

Table B.2: Overview of Literature Review Findings: Microgrid Performance

Author(s)	**Year**	**Location**	**MG generation source**	**Main findings**
Tayab et al. [29]	2021	-	PV	The grey wolf optimisation (GWO) scheduling strategy has proven to outperform other strategies in forecasting PV power and load demand.
Liu et al. [28]	2017	-	PV	A simulation model to predict microgrid electricity consumption has been developed for differing weather conditions. It found that a community microgrid can reduce electricity costs for its users.
Naware & Mitra [30]	2021	-	PV	The authors propose a classical long short-term memory neural network model to predict day-ahead load and solar insolation.
Wakui et al. [86]	2016	-	Unspecified	A mixed integer linear programming (MILP) model is used to predict and optimize energy use in a microgrid using fuel cells.
Bruni et al. [31]	2015	-	PV	The article compares deterministic and stochastic Model Predictive Control (MPC), with stochastic showing better ability to forecast energy savings and system efficiency.

Khayat et al. [25]	2024	United Kingdom	Not relevant	Short term load forecasting can be predicted more accurately with the Adaptive Neuro-Fuzzy Inference System (ANFIS), instead of a standard neural network model.
Hafsi et al. [32]	2022	Algeria	PV & Wind	With the consumption minimization strategy ECMS and external energy maximization strategy EEMS, the article shows that system performance is enhanced, while system costs are minimized by reducing hydrogen consumption.
Marinescu et al. [26]	2013	Ireland	Unspecified	A comparison of different methods to perform short term load forecasting is conducted.
Mroueh et al. [87]	2025	-	Not relevant	Belief Functions Theory (BFT) decreases forecasting errors by 12% and enhances computational efficiency over existing methods, enabling microgrids to manage loads more reliably using a publicly available real dataset.
Ożadowicz & Walczyk [27]	2023	Poland	PV	With a façade dynamics control system, PV-tracking energy consumption has been reduced from 5% to 1% of energy consumption in autumn and from almost 3.2% to 0.6% in spring. This is significant, as PV system efficiency is below 10%.
Hanbashi et al. [33]	2023	Kenya	PV	A case study in Kenya to optimize PV-based mini-grids design has been conducted, mainly emphasising the need for information on the control algorithms and availability of on-site measurements.
Lagos et al. [88]	2022	-	Wind	The paper reviews the latest wind-speed and wind-power forecasting models used across various power system scales—from large wind farms to residential micro-wind turbines. The results show that there is a large focus on hybrid forecasting methods, especially for short-term predictions.
Liaquat et al. [34]	2023	-	PV	The authors have designed a peer-to-peer (P2P) energy trading market for residential solar PV users using a day-ahead continuous double auction that accounts for network losses and fees. The market improves social welfare by 17.75% on average and also analyzes how forecasting errors affect trading between day-ahead and real-time markets.
Hakam et al. [35]	2025	-	PV	A hybrid maximum power point tracking (MPPT) algorithm is introduced. combining artificial neural networks and grey wolf optimization (ANN-GWO). Coupled with model predictive control (MPC), the approach improves tracking efficiency by more than 9%, while also reducing total harmonic distortion (THD).

C

Scenario Results: PCPs

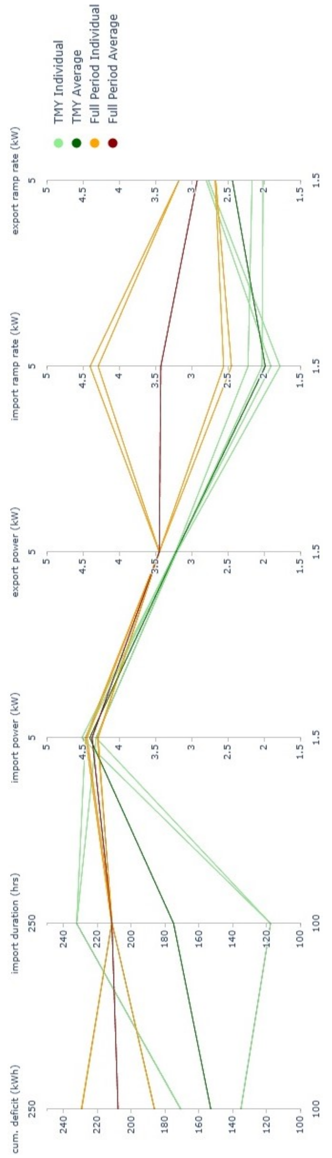

Figure C.1: PCP comparing different time periods (TMY vs Full Period) - ”Gas Heating” Scenario

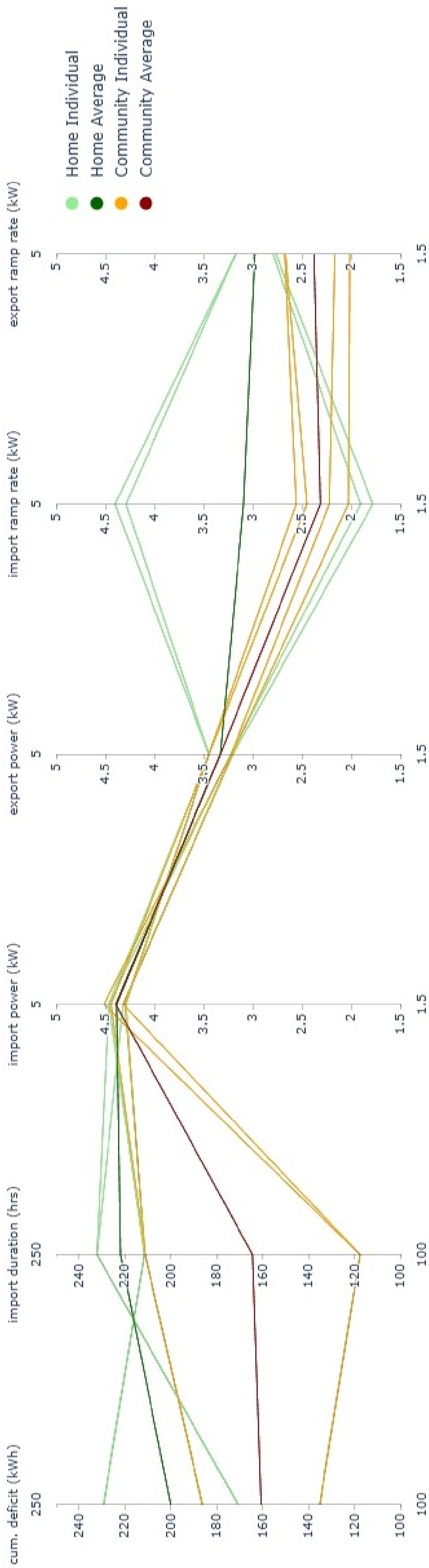

Figure C.2: PCP comparing different BESS types (Home Battery vs Community Battery) - ”Gas Heating” Scenario

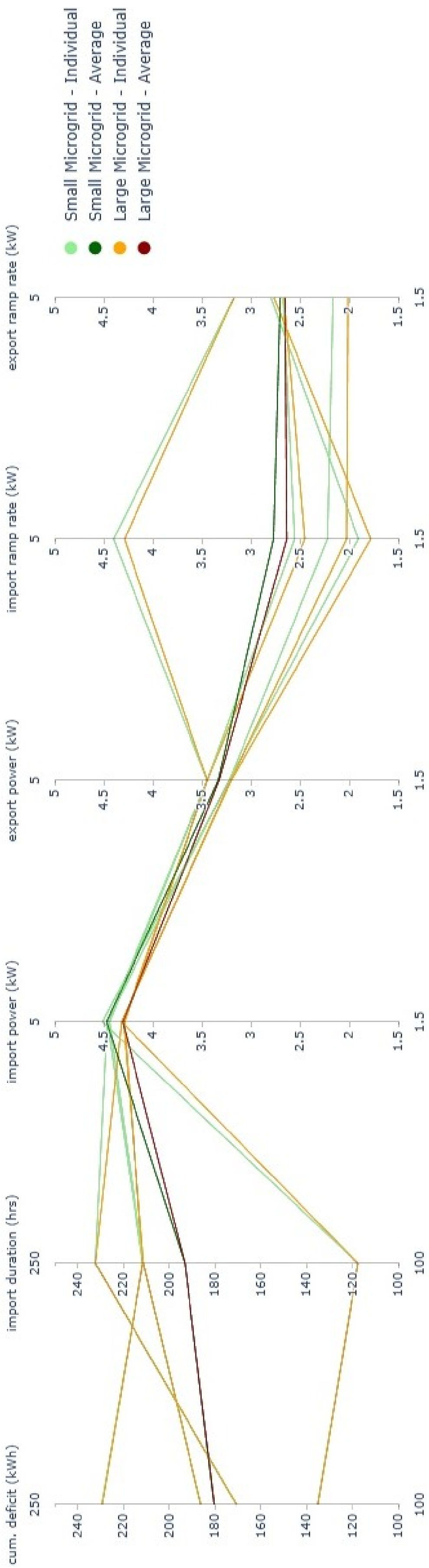

Figure C.3: PCP comparing different Microgrid Sizes (Small Microgrid vs Large Microgrid) - "Gas Heating" Scenario

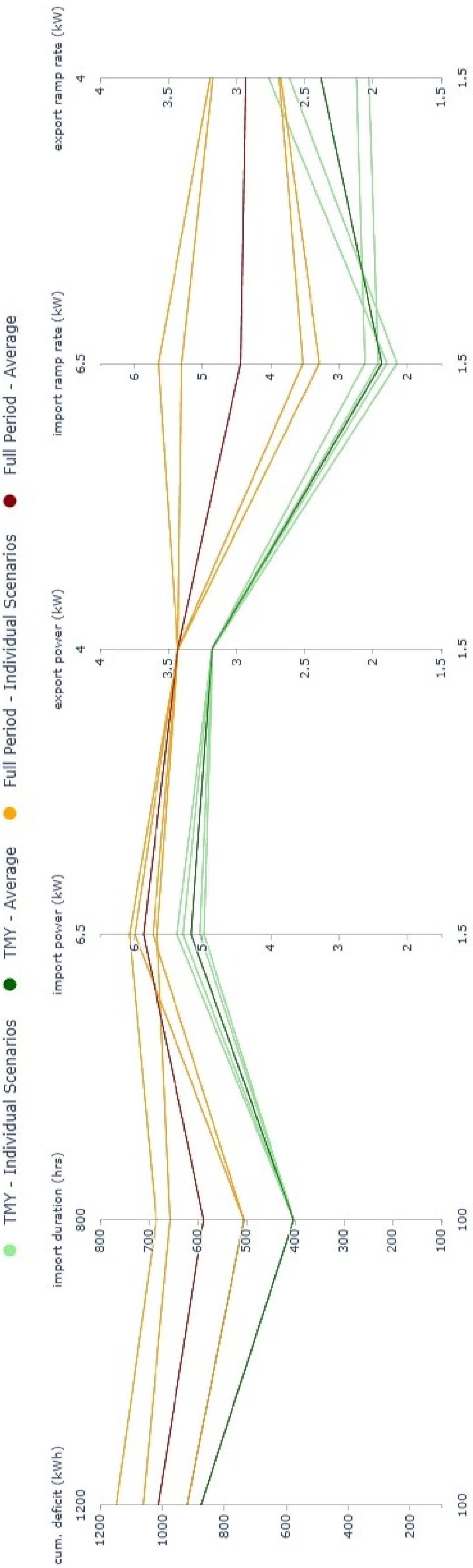

Figure C.4: PCP comparing different time periods (TMY vs Full Period) - ”Heat Pump” Scenario

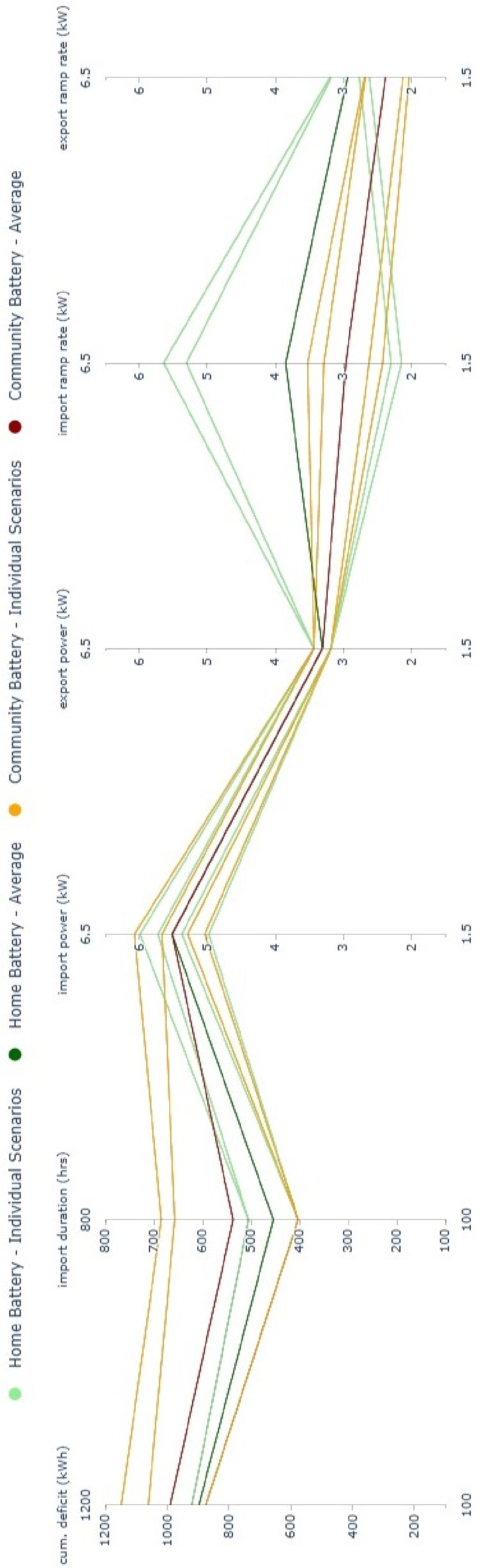

Figure C.5: PCP comparing different BESS types (Home Battery vs Community Battery) - ”Heat Pump” Scenario

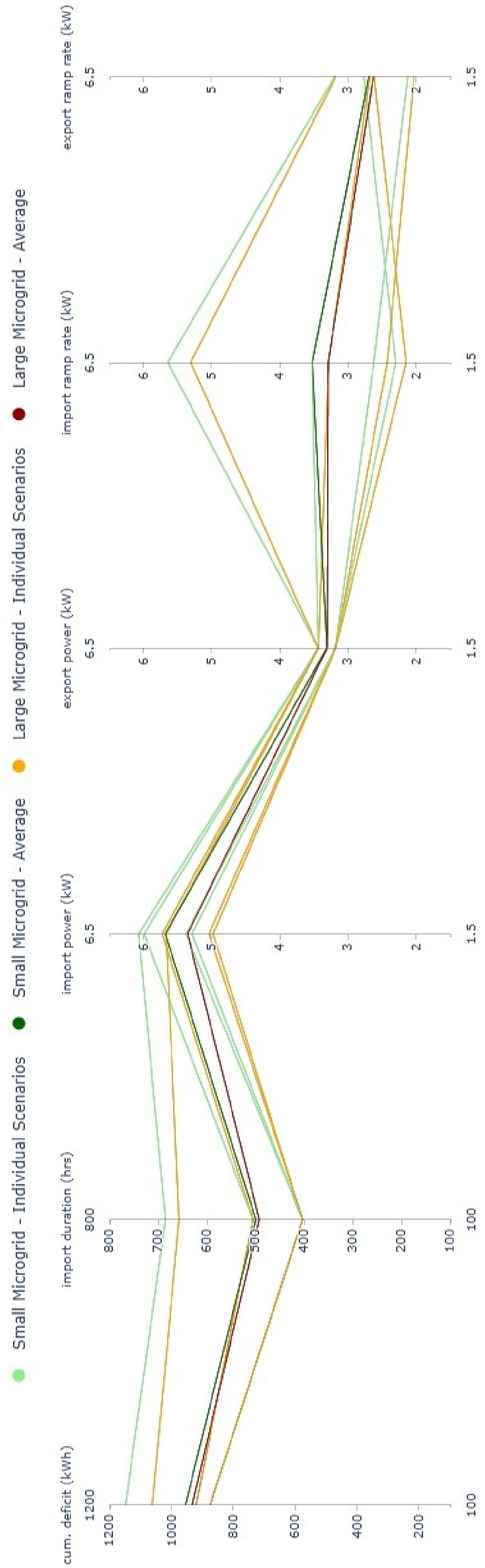

Figure C.6: PCP comparing different Microgrid Sizes (Small Microgrid vs Large Microgrid) - ”Heat Pump” Scenario

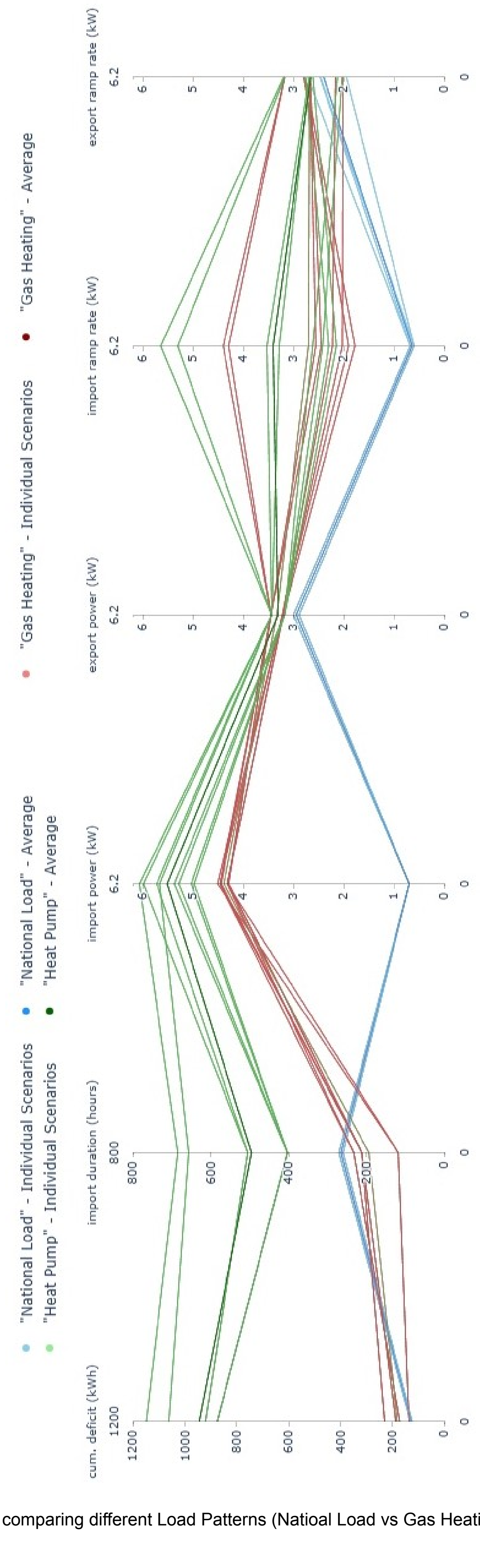

Figure C.7: PCP comparing different Load Patterns (Natioal Load vs Gas Heating vs Heat Pump)

D

Scenario Results: CDFs

D.1. Scenario "National Load Pattern"

D.1.1. Cumulative Deficit

There is a clear distinction in the CDF's between the full period and TMY scenarios, shown in Figure D.1. Even though the full period models have a higher maximum value, the cumulative deficit is higher for the TMY scenarios for most of the cumulative probabilities. For instance at a cumulative probability of 0.9, the full period scenarios are around 20kWh while the cumulative deficit for the TMY scenarios is twice at high, with a cumulative deficit value around 40kWh.

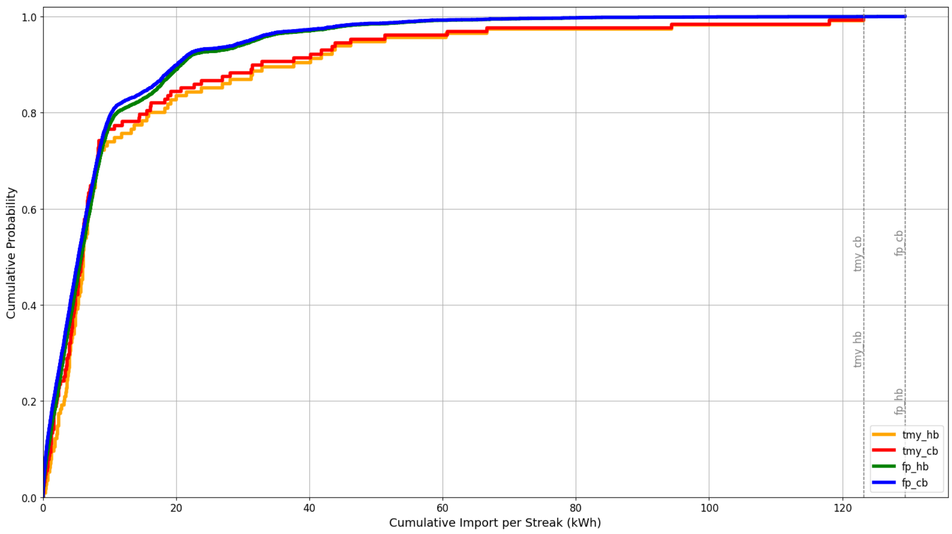

Figure D.1: CDF of the Cumulative Deficit

D.1.2. Import Period Duration

For the import period duration CDF in Figure D.2, there is a clear distinction between the TMY and full period scenarios again. Much like for the cumulative deficit, the maximum import duration is slightly higher for the full period scenarios. TMY scenarios have higher import period durations for all of other cumulative probabilities.

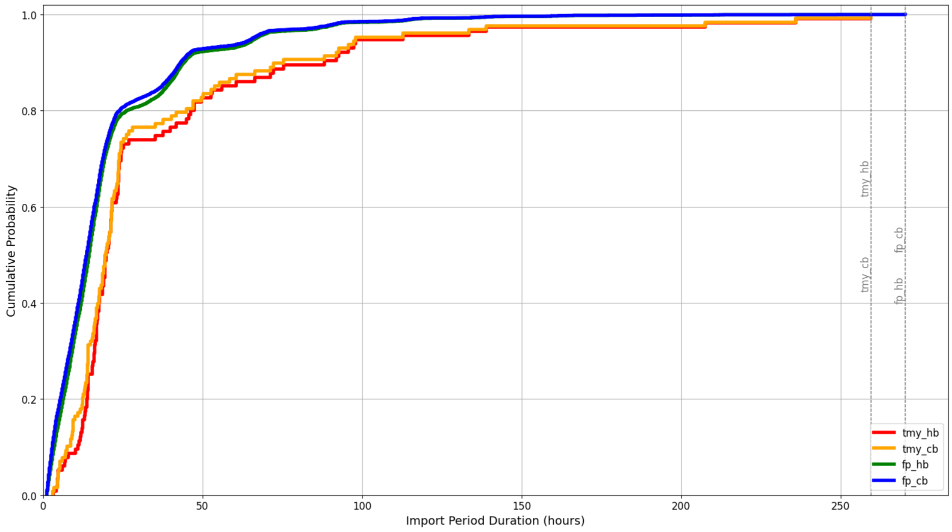

Figure D.2: CDF of the Import Duration

D.1.3. Import Power
Even though the maximum values for import power are all equal for each of the scenarios, as shown in Figure D.3, there is a distinct difference in the CDF between TMY and full period scenarios. The TMY scenarios around a third of import power values lie below 400W, while only 15% of import power values are below 400W. From there, there is a steep increase in probability for increasing power values. This indicates that in the TMY scenarios, a higher share of the demand can partially be covered by electricity supplied from the BESS.

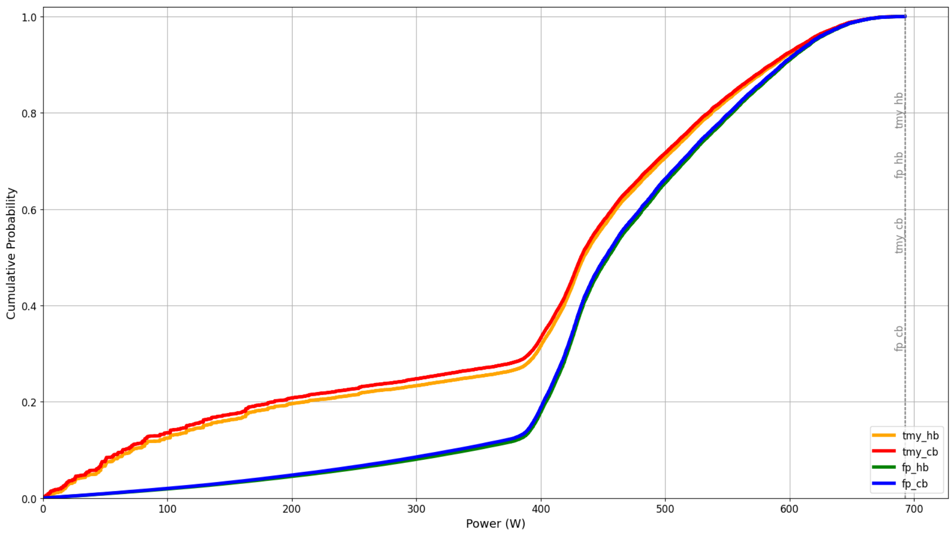

Figure D.3: CDF of the Import Power

D.1.4. Export Power
The CDFs in Figure D.4 mainly show the differences between the BESSs. Home batteries show larger export power for the same cumulative probabilities than the community batteries. This likely is the effect of community batteries have a smaller capacity than home batteries, leading to an increased amount of export power events with low export power. This does not affect the maximum export power values, where there is a different between the TMY and full period scenarios.

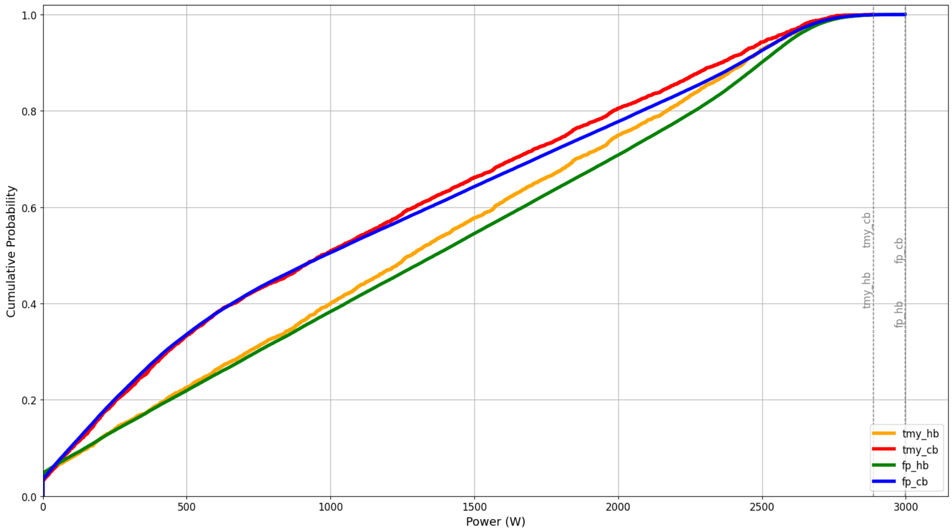

Figure D.4: CDF of the Export Power

D.1.5. Import Ramp Rate

The CDFs in Figure D.5 show a greatly similar distribution for all scenarios, with a slight difference between the TMY and full period scenarios from a cumulative probability of 0.8. From this point onwards, the full period scenarios have slightly higher import ramp rate values for similar cumulative probabilities, while also having a higher maximum import ramp rate.

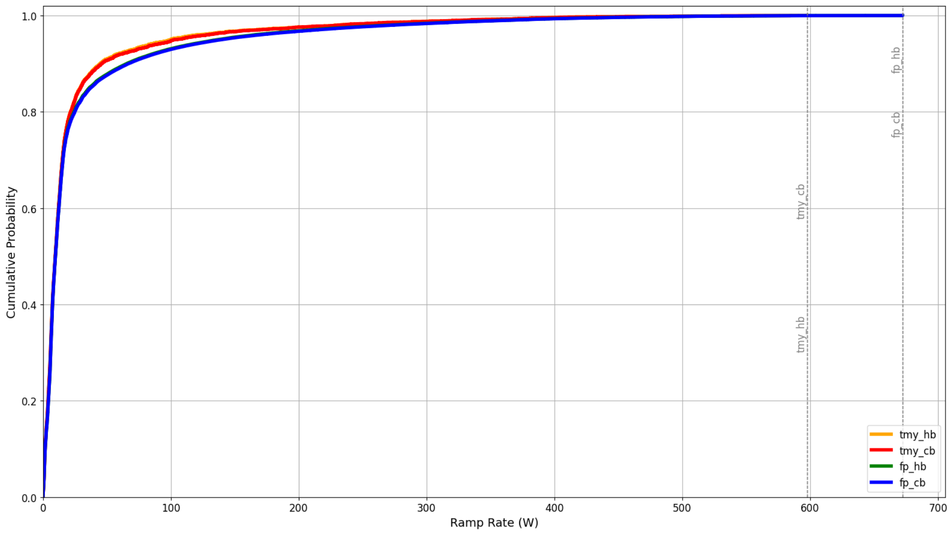

Figure D.5: CDF of the Import Ramp Rate

D.1.6. Export Ramp Rate

The export ramp rate CDFs show similar distributions for the different BESS types, as can be seen in Figure D.6. The home batteries have larger export ramp rate values for equal cumulative probabilities, compared to the community batteries. In terms of maximum export ramp rate values, it can be seen that the full period scenarios have higher maxima, while home batteries also have higher export ramp rate values compared to community batteries. This can be explained by the greater capacity of the home batteries, meaning that once they are full, the microgrid system is required to export large amounts of electricity straight away.

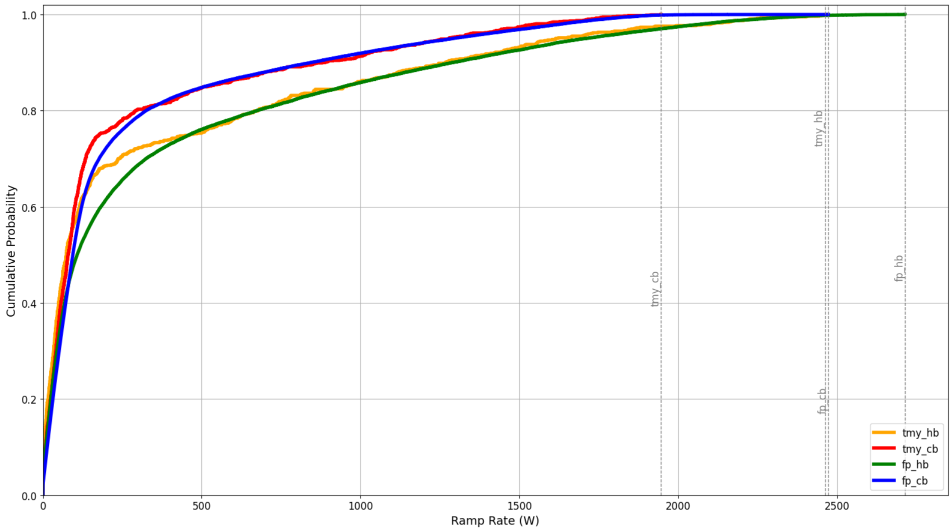

Figure D.6: CDF of the Export Ramp Rate

D.2. Scenario "Gas Heating"

D.2.1. Cumulative Deficit

The distributions presented in Figure D.7 all have a greatly similar pattern, with small deviations between TMY and full period scenarios. In terms of maximum cumulative deficit values, there are two clear distinctions. One is between the TMY and full period scenarios and the other between the BESS types. Microgrid size does not have an impact in relation to the maximum cumulative deficit. Home battery scenarios having a larger maximum cumulative deficit can be explained by the fact that if only one household is importing electricity, it still counts as the same streak. As home battery capacity and electricity can not be shared between households, there might be periods when only a small fraction of households is importing electricity, but this still adds to the cumulative deficit, while community battery scenarios are able to share the capacity between all households and fully cover the demand without importing electricity.

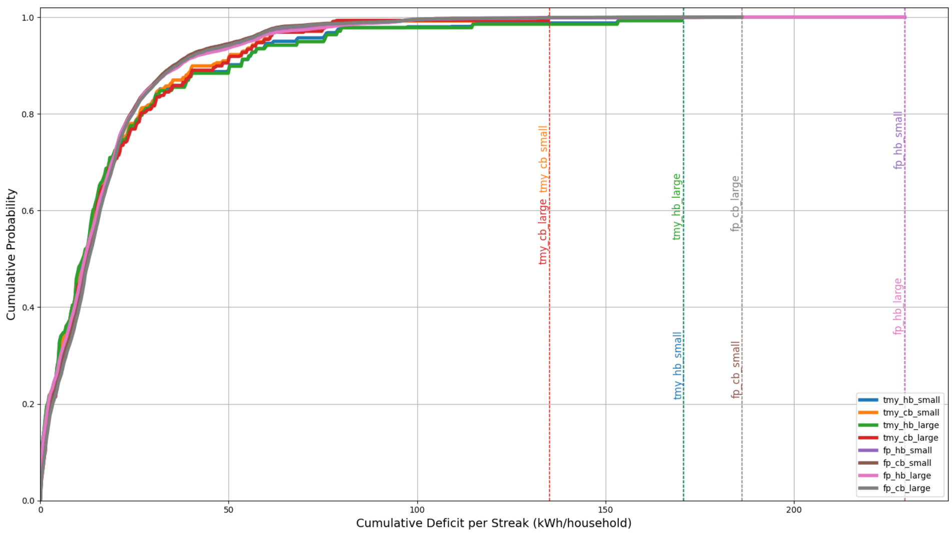

Figure D.7: CDF of the Cumulative Deficit

D.2.2. Import Period Duration

Import Duration CDFs have similar distribution patterns for all scenarios, which are presented in Figure D.8. There is a large difference between the TMY and full period scenarios in terms of the maximum import duration, while the home battery TMY scenarios even exceed the full period scenarios for maximum import duration.

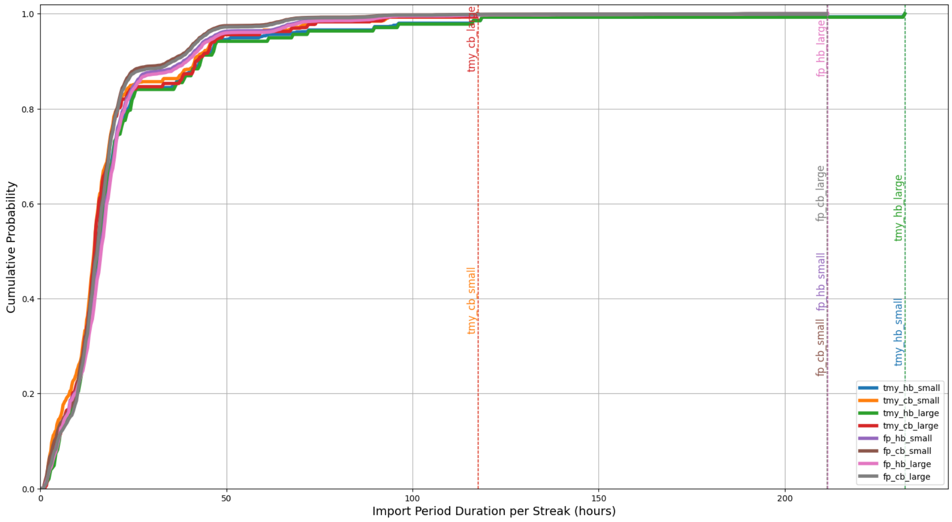

Figure D.8: CDF of the Import Duration

D.2.3. Import Power

The import power CDFs, shown in Figure D.9, only slow slight differences between the TMY and full period scenarios. In terms of maximum import power requirements, the TMY scenarios show slightly higher maxima. This seems counter-intuitive, but can be explained by the battery state of charge, which is set to 0 at the start of the period. Likely, the full period scenarios already have some charged capacity when coming across the same time step, resulting in a lower peak import power value.

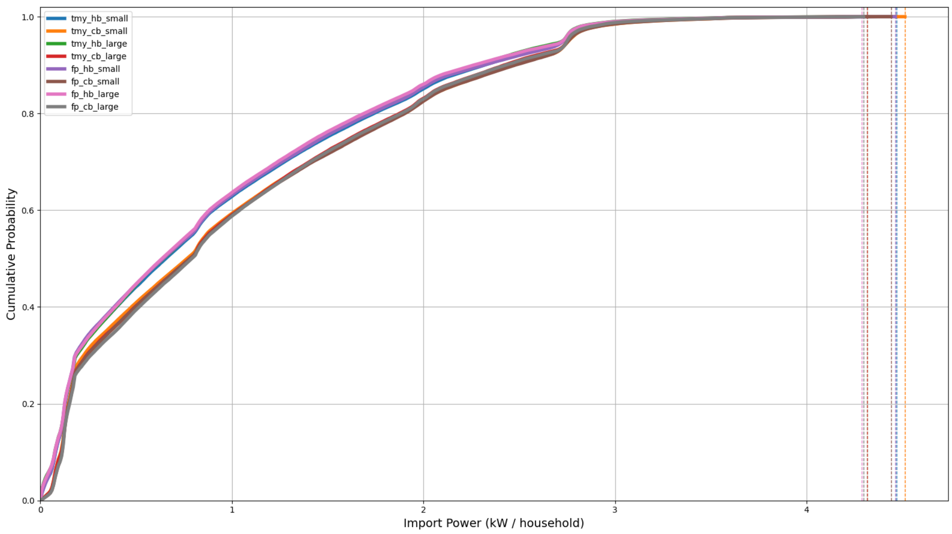

Figure D.9: CDF of the Import Power

D.2.4. Export Power

Export power CDFs show a large difference between BESSs at low export power values, shown in Figure D.10. This is due to the fact that for home battery scenarios lead to a higher frequency of low export power events, as only a small fraction of households are exporting electricity once their individual home batteries are full, while others are still charging the home battery. For the community battery, either the whole microgrid is exporting electricity or all households use the excess electricity to charge the community battery. This results in a relatively low cumulative probability for small export power values. For the home battery scenarios, a difference in TMY and full period scenario can be seen, too. All of this does not impact the maximum export power, which is slightly lower for all TMY scenarios compared to all full period scenarios.

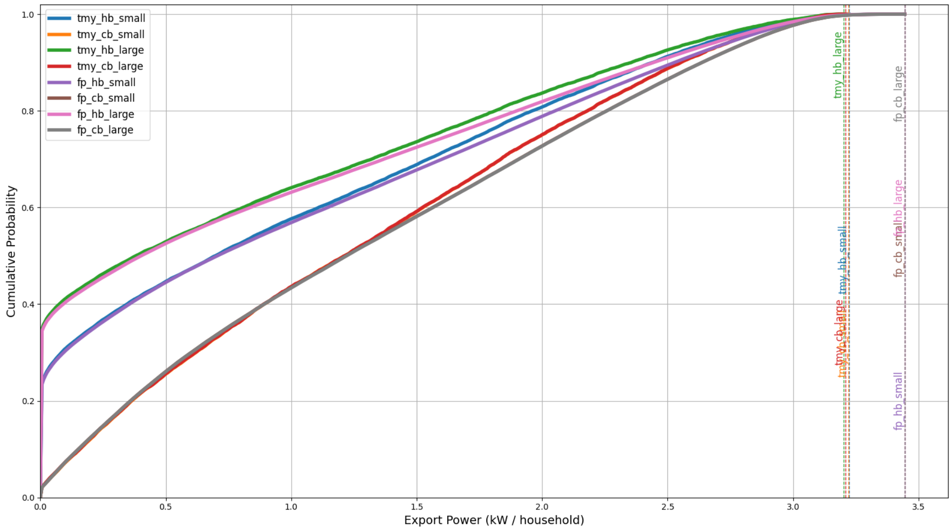

Figure D.10: CDF of the Export Power

D.2.5. Import Ramp Rate

Even though the CDFs of all scenarios are vastly similar, there are three main differences in terms of peak import ramp rate values, illustrated in Figure D.11. Firstly, full period scenarios have higher maxima and the same is true for small microgrid sizes. Furthermore, home batteries require higher import ramp rates than community batteries, especially for the full period scenarios, with values being almost twice as high. This effect is expressed in a reduced manner for the TMY scenarios.

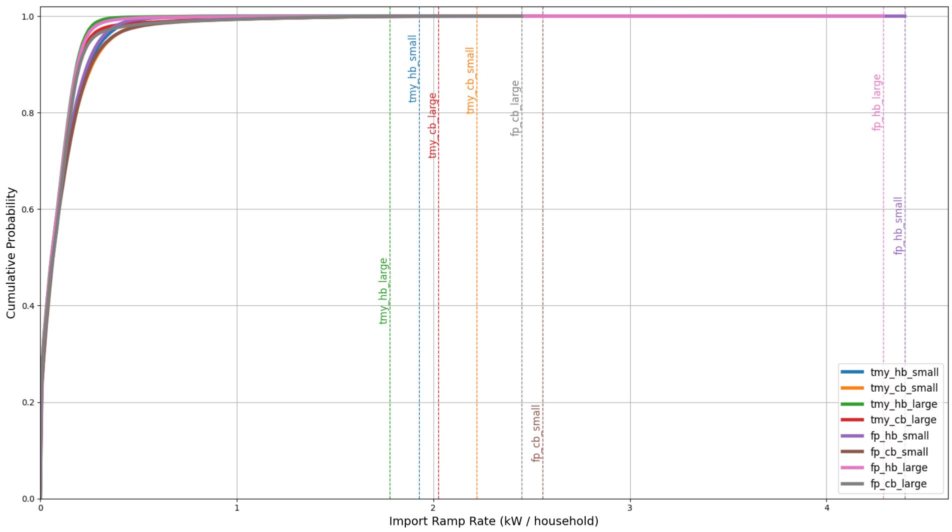

Figure D.11: CDF of the Import Ramp Rate

D.2.6. Export Ramp Rate

A similar difference for small export ramp rate values is detected between home battery and community battery scenarios as for the export power CDF. From Figure D.12, a difference in maximum values between the BESSs can be seen, with community batteries greatly reducing the maximum export ramp rate. For the TMY scenarios, larger microgrids also reduce the maximum required export ramp rate per household. Lastly, the full period scenarios show the need for greater system boundaries in terms of maximum export ramp rate when compared to the TMY scenarios.

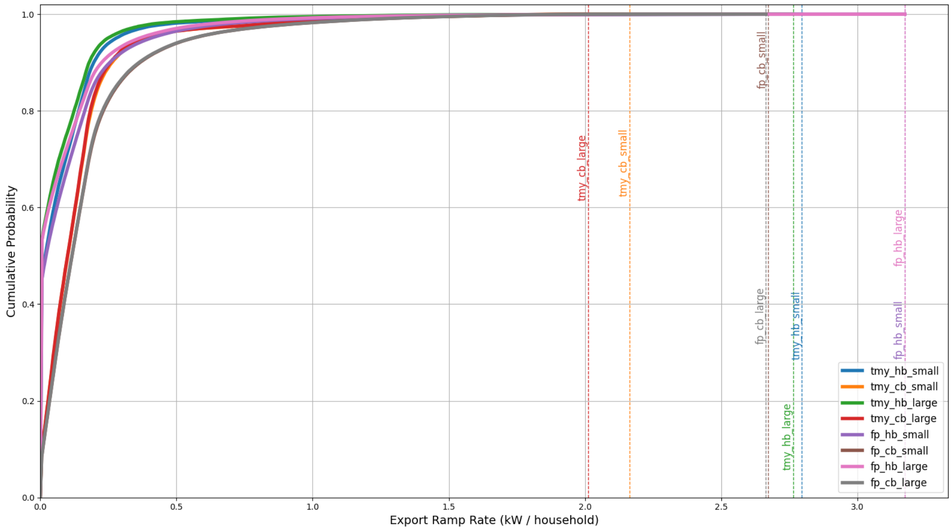

Figure D.12: CDF of the Export Ramp Rate

D.3. Scenario "Heat Pump"

D.3.1. Cumulative Deficit

Cumulative deficit CDFs of Figure D.13 across all scenarios follow a similar distribution, with only slight deviations between TMY and full period scenarios. The TMY scenarios are result in the same maximum values, but for the full period scenarios, the community battery scores worse than the home battery scenarios. For the full period community battery scenarios, there is also a difference between a small and large microgrid size, with large microgrid sizes having a lower maximum cumulative deficit per household.

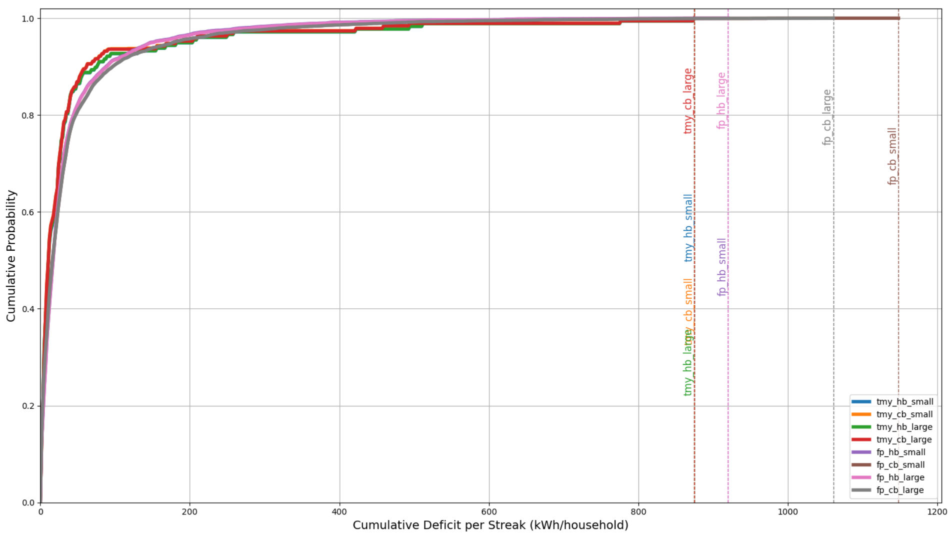

Figure D.13: CDF of the Cumulative Deficit

D.3.2. Import Period Duration

The conclusions from the cumulative deficit CDFs shown in Figure D.14 are directly applicable to the import durations CDFs shown in Figure D.14. There is practically no difference between the two, relatively.

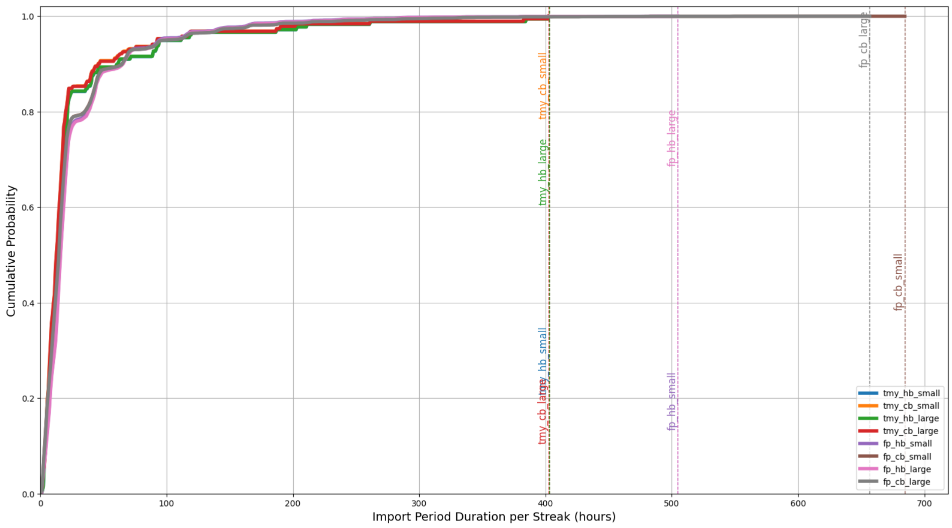

Figure D.14: CDF of the Import Duration

D.3.3. Import Power

The distributions shown in Figure D.15 are similar for all scenarios, with the full period home battery scenarios having a slight deviation at the lower import power values. This is related to the electricity import of only a fraction of the households, as capacity sharing between home batteries is not possible. At the maximum values, there is a clear distinction between the TMY and full period scenarios, as well as a relatively large difference between small and large microgrid sizes. The BESS type has a relatively small effect on the maximum import power value, but an effect nonetheless.

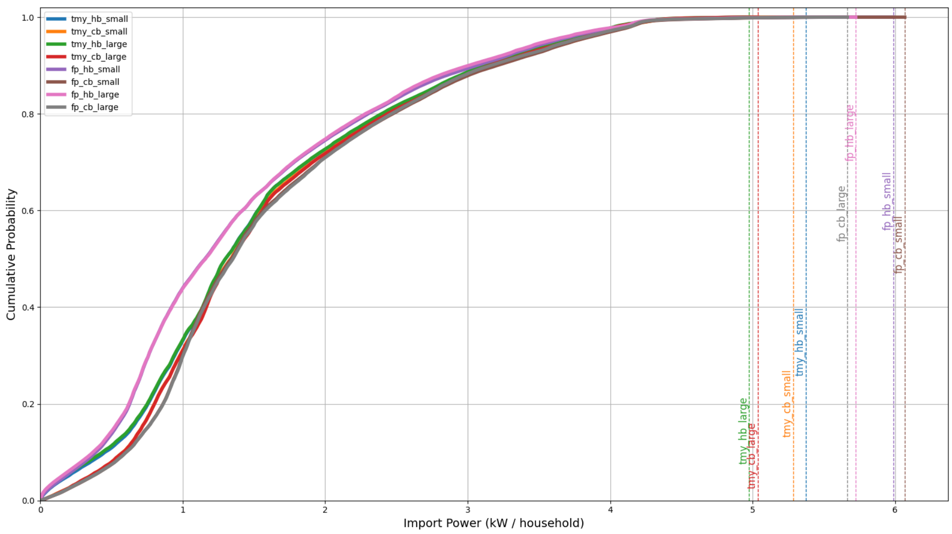

Figure D.15: CDF of the Import Power

D.3.4. Export Power

The export power CDFs of this scenario, shown in Figure D.16, follow the same patterns as in scenario B (Figure D.10). For low export power values, cumulative probabilities of the home battery scenarios is much higher, compared to the community battery scenarios. For the maximum values, there is a clear difference between the TMY and full period scenarios, but not for any of the other testing conditions.

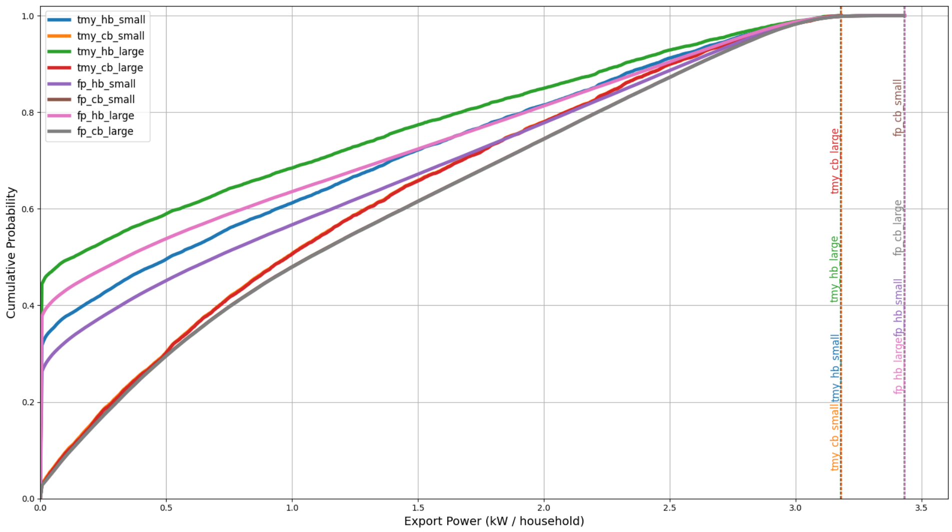

Figure D.16: CDF of the Export Power

D.3.5. Import Ramp Rate

The CDFS in Figure D.17 all follow the same distribution, but there are clear differences in the maximum import ramp rate values. Home batteries score better for the TMY scenarios, while community batteries score better by a bigger margin for the full period scenarios. For each of the scenarios, a benefit can be seen by the larger microgrid size. Lastly, the full period scenario show larger maximum import ramp rates requirements compared to the TMY scenarios.

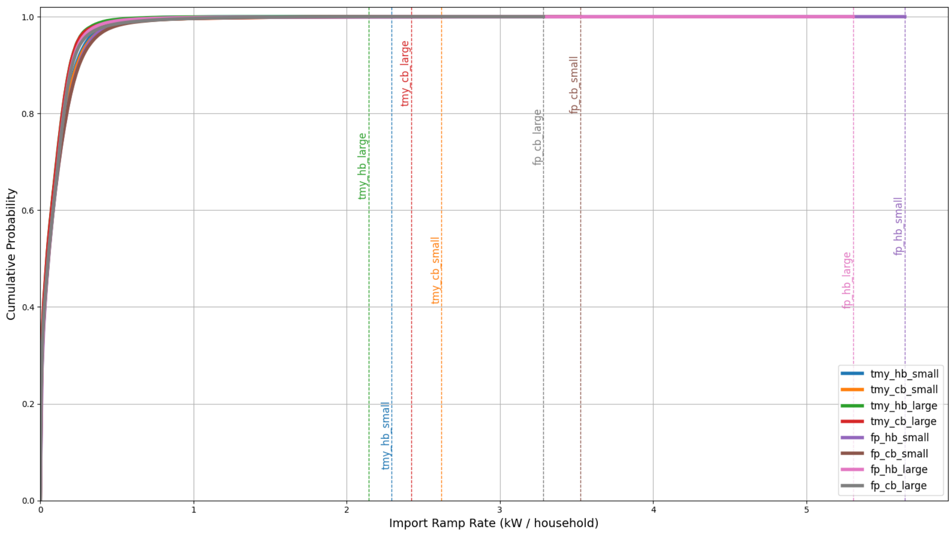

Figure D.17: CDF of the Import Ramp Rate

D.3.6. Export Ramp Rate

Similar to the export power, the cumulative probability at low export ramp rate values is much higher for the home battery models, displayed in Figure D.18. Around the 0.95 cumulative probability mark, all of the distributions converge, but different maximum values are reached. Community battery scenarios show a large benefit compared to the home battery scenarios. Also, there is an effect of a somewhat similar scale between the TMY and full period scenarios. Lastly, the microgrid size also makes a difference, especially for the TMY scenarios, with a larger microgrid size having a lower maximum export ramp rate value.

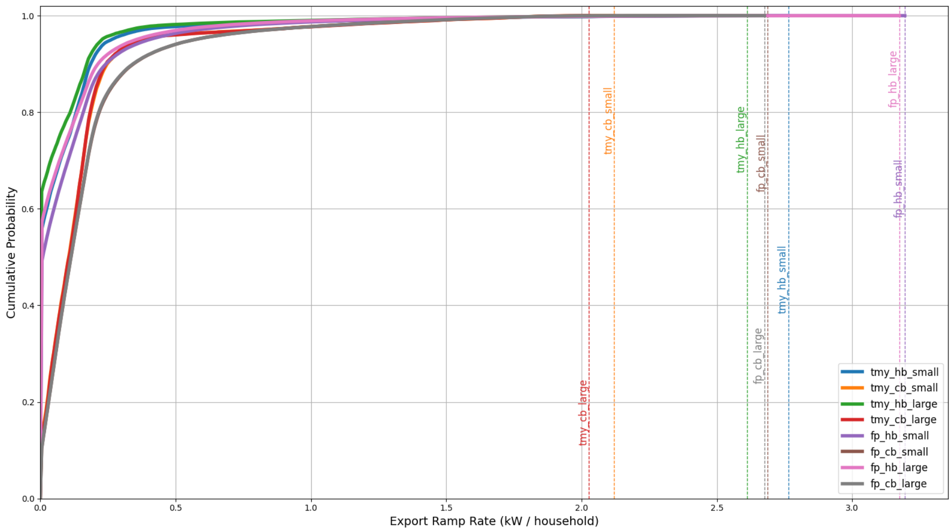

Figure D.18: CDF of the Export Ramp Rate