Data-Driven Methods for Demand-Side Flexibility in Energy Systems

Zur Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften

von der KIT-Fakultät für Informatik

des Karlsruher Instituts für Technologie (KIT)

genehmigte

Dissertation

von

Nicole Nadine Ludwig

aus Ludwigsburg

Tag der mündlichen Prüfung: 25.06.2020
Gutachter: Prof. Dr. Veit Hagenmeyer
Prof. Dr. James W Taylor
apl. Prof. Dr-Ing. Ralf Mikut

Acknowledgements

Writing this dissertation has been quite the journey, and I am incredibly thankful for everyone who has accompanied me on this path.

First and foremost, I want to thank my supervisor Veit Hagenmeyer. Thank you for your guidance, patience, and inspiration. Thank you for giving me the freedom to pursue my research topics at the institute. I would also like to thank Ralf Mikut for the discussions and feedback throughout my time spend within your group.

Thank you, James Taylor and Siddharth Arora for hosting me in Oxford. I have learnt a lot from both of you through the long discussions and valuable feedback and had an extraordinary time. Also thank you, James, for being a reviewer of my dissertation and taking part in my viva via video conference. Thank you Franziska Mathis-Ullrich and Pascal Friederich for being on my committee and providing feedback.

I would not have been able to pursue this PhD without the financial support through the DFG Research Training Group 2153 "Energy Status Data". Thank you to the DFG and all PIs and colleagues within the research training group.

I want to thank all my co-authors, collaborators, and students with whom I had the pleasure to work. Thank you for our joint projects; I thoroughly enjoyed working on them with you. A special thank you, Lukas, for patiently answering all my, far too often organisational, questions and being such a great collaborator, and thank you Shahab for our regular meetings on Campus North discussing research and so much more.

Thank you to the best office mates I could have wished for; Marian, Katharina and Kaleb. I would not be here without you and am incredibly grateful for all the discussions, laughter, guidance and support you have given me. Thanks also to my colleagues at IAI for the great past four years, especially Andreas, Angel, Benjamin, Claudia, Fabian, Friedrich, Ines, Jan, Johannes, Lisa, Markus, Martha, Moritz, Richard, Simon, Tim, Tom, Vojtech and Yanke.

Finally, thank you to my parents, grandparents, Patrick, and my friends, for always believing in me, celebrating my successes and cheering me up when things got tough.

Karlsruhe, June 2020 *Nicole Nadine Ludwig*

Abstract

Many societies want to achieve an energy system which does rely mainly (or even purely) on renewable energy sources. However, the current energy system was not built with the volatility of renewable energy sources in mind. Hence, it is challenging to incorporate all the potential of renewable energy sources in today's energy grid. There exist various ways to ease the integration, one very prominent is using demand-side flexibility. Demand-side flexibility highlights the idea that the demand side is not static in its behaviour and can change and adapt. With enough demand-side flexibility at the right points in time, we can exploit all renewable energy potential so that there is no need to mobilise conventional energy sources in order to keep the network stable. It is still an open topic of research how much flexibility a given system has, if that flexibility is enough for specific aims and how to increase it.

Hence, in this thesis, we establish data-driven methods to analyse the demand-side flexibility in a given energy system. More specifically, we investigate how and to what extent data alone is sufficient to determine flexibility information. We introduce methods for three different aspects of flexibility. We start with determining the potential flexibility from historical consumption time series. In order to properly assess this flexibility from data, we present a respective mathematical modelling framework. Based on this framework, we establish a method to detect how the system is operated and derive flexibility measures from this information. The second aspect of flexibility which we investigate is the required flexibility, thus answering if we need any flexibility at all and how much of it we do need. For this task, we show that it is essential to include properly post-processed weather forecasts into demand forecasting models and especially to account for the dependency structures among weather variables. Additionally, we analyse what amount of change is needed in a given system to achieve goals such as peak shaving with a new method using motif discovery and scheduling. We find that small changes in the operation of processes already reduce peak demand and energy overshooting. Additionally, as having flexibility in the system is not equal to using this flexibility, we derive new incentive strategies for the grid operator to achieve more flexibility usage in the energy grid she supplies. We show through an agent-based simulation that setting the right incentives can reduce the amount of conventional energy needed in a given system. For the case in which these incentives are not sufficient, we develop a new method to size centralised energy storage systems on a low-voltage distribution grid.

Zusammenfassung

Mit dem Pariser Abkommen haben sich immer mehr Staaten dazu entschlossen, aktiv gegen eine weitere Erhöhung der Erdtemperatur vorzugehen. Um die Ziele des Abkommens zu erreichen, spielen Energieerzeugung und -verbrauch eine entscheidende Rolle. Bei der konventionellen Energieerzeugung werden fossile Brennstoffe verwendet, welche die Umwelt und Atmosphäre der Erde besonders stark belasten und zum Klimawandel beitragen. In vielen Staaten wird daher eine Energiewende angestrebt, welche weg von fossilen Energieerzeugern und hin zu erneuerbarer Energie führen soll. Die Erzeugung durch erneuerbare Energieträger, zum Beispiel Windkraftanlagen, kann nicht gesteuert werden und ist zudem wetterabhängig. Besonders die Wetterabhängigkeit führt zu einer unbeständigen, fluktuierenden Versorgung. Das aktuelle Energiesystem ist nicht für diese fluktuierende Erzeugung ausgelegt und wird daher vor große Herausforderungen gestellt.

Um die Integration von erneuerbaren Energieträgern zu erleichtern, gibt es verschiedene Ansätze, sowohl auf der Erzeugungs- als auch auf der Nachfrageseite. Neben dem Netzausbau und Speichertechnologien gibt es auch Ansätze, welche sich damit beschäftigen, das Verbrauchsverhalten so zu ändern, dass es besser zu der erneuerbaren Erzeugung passt. Eine solche Verhaltensanpassung ist jedoch nur möglich, wenn Flexibilität der Nachfrageseite vorhanden ist, das heißt wenn Verbraucher die Möglichkeit haben ihre Nachfrage zu verändern, also zum Beispiel zu unterbrechen oder zu verschieben.

Wie genau die Flexibilität in einem gegebenen Energiesystem aussieht, also ob Flexibilität vorhanden oder benötigt wird, ist meist unklar. In der aktuellen Literatur werden oft Annahmen getroffen, oder nur solche Systeme betrachtet in denen eine Expertin Flexibilitätsinformationen bereitstellen kann. Im Rahmen der vorliegenden Dissertation werden daher Methoden zur Analyse der Flexibilität der Nachfrageseite in einem gegebenen Energiesystem entwickelt. Um Entscheidungsprozesse zu erleichtern und ohne Expertenwissen erste Informationen über die Flexibilität der Nachfrageseite in einem gegebenen System zu erhalten, werden ausschließlich datengetriebene Methoden vorgestellt. Die Arbeit ist in drei Teile gegliedert, welche jeweils von einer grundlegenden Forschungsfrage ausgehen und diese Frage aus einer Datenperspektive beantworten.

Potenzielle Flexibilität der Nachfrageseite. Der erste Hauptteil der Dissertation beantwortet mit datengetriebenen Methoden die Frage, wie viel Flexibilitätspotenzial in einem System vorhanden ist. Hierbei wollen wir dieses Potenzial in historischen Energiezeitreihen bestimmen, um ohne grundlegendes Expertenwissen Entscheidungsprozesse zu unterstützen. Dazu muss zuerst die nachfrageseitige Flexibilität anhand verschiedener Merkmale mathematisch beschrieben werden können. In einem zweiten Schritt sollen in den entsprechenden Zeitreihen relevante Prozesse detektiert werden und hinsichtlich ihrer Flexibilität analysiert. Die Literatur zu Modellierungsansätzen für Flexibilität ist zahlreich, jedoch werden die meisten Modelle vor dem Hintergrund einer spezifischen Anwendung erstellt und können nur schwer auf andere Anwendungen übertragen werden. Wir stellen in dieser Dissertation daher ein umfassendes Modellierungsframework für nachfrageseitige Flexibilität vor mithilfe dessen die Flexibilität von Prozessen in Haushalten und der Industrie beschrieben werden kann und zwar unabhängig von der zu betrachtenden Anwendung. Zudem präsentieren wir ein neues Verfahren basierend auf Motif Discovery, welches uns erlaubt aus historischen Energiezeitreihen das Flexibilitätspotenzial eines Systems zu bestimmen.

Erforderliche Flexibilität der Nachfrageseite. Der zweite Hauptteil der Dissertation evaluiert, wie viel nachfrageseitige Flexibilität erforderlich ist, um das Energiesystem zu stabilisieren. Hierbei liegt der Fokus auf zwei Aspekten: Ist überhaupt Flexibilität erforderlich und falls ja, wie viel Flexibilität ist dann erforderlich? Das heißt, in welchem Maße muss ein bestehendes System verändert werden, um wieder ein Gleichgewicht zwischen Verbrauch und Erzeugung herzustellen? Wir beantworten die erste Teilfrage mithilfe von Prognosen. Wenn Verbrauch und Nachfrage ausreichend gut vorhergesagt werden können für jeden Zeitpunkt, dann kann auch das Ungleichgewicht bestimmt werden. Bisher wird bei der probabilistischen Vorhersage des Verbrauches die Unsicherheit des Wetters nicht ausreichend berücksichtigt. In dieser Dissertation evaluieren wir daher, wie die Unsicherheit des Wetters korrekt durch die Vorhersagemodelle propagiert werden kann. Ein besonderer Fokus liegt dabei auf dem statistischen Postprocessing der Wettervorhersagen. Den zweiten Teil der Frage beantworten wir mithilfe eines neuen Frameworks basierend auf Motif Discovery und Scheduling und analysieren dabei welche Effekte eine Veränderung des Ablaufplans auf zum Beispiel Lastspitzen hat. Die Ergebnisse zeigen dabei, dass kleine Veränderungen im Prozessablauf bereits zu signifikante Verbesserungen in der Möglichkeit der Einbindung von erneuerbaren Energieträgern führen.

Zusammenbringen von potentieller und erforderlicher Flexibilität der Nachfrageseite. Der dritte Hauptteil der Dissertation verknüpft die beiden vorherigen Teile, indem die potenzielle Flexibilität, als auch die erforderliche Flexibilität zusammengeführt werden. Wir fokussieren uns auf ein lokales Energienetz: dabei steht die Frage im Mittelpunkt, welche Anreize gesetzt werden können, damit die potenzielle Flexibilität der Nachfrageseite, zu den Zeitpunkten zu denen sie benötigt wird, auch eingesetzt werden kann. Wir etablieren verschiedene neue, theoretische Anreizmechanismen, bevor wir mithilfe eines Agentenmodells untersuchen, wie sich die verschiedenen Anreize

auf die benötigte Energiemenge aus konventionellen Energieträgern auswirkt. Dabei zeigt sich, dass der Bedarf an konventionellen Energieträgern mit einfachen Anreizsystemen stark zurück geht. Für den Fall, dass die Anreizmechanismen nicht ausreichen, entwickeln wir eine datengetriebene Methode zur Auswahl und Dimensionierung von zentralen Energiespeichern im Niederspannungsnetz.

Zusammengefasst werden in der vorliegenden Dissertation neue Methoden der Datenanalyse für die Flexibilität der Nachfrageseite in Energiesystemen eingeführt. Dabei werden die Aspekte der potenziellen und erforderlichen Flexibilität untersucht sowie Mechanismen diese zusammen zu bringen. Die Dissertation zeigt, dass mithilfe datengetriebener Methoden alle drei Kernforschungsfragen zur Analyse der Nachfrageflexibilität in einem gegebenen Energiesystem beantwortet werden können. Die datengetriebenen Methoden ermöglichen somit bessere Entscheidungsprozesse hinsichtlich der Flexibilität von Energiesystemen.

Contents

List of Figures

List of Tables

Abbreviations

CPP	conventional power plant
CRPS	Continuous Ranked Probability Score
DFT	Discrete Fourier Transform
DSM	demand-side management
ECC	Ensemble Copula Coupling
ECDF	Empirical Cumulative Distribution Function
ECMWF	European Centre for Medium-Range Weather Forecasts
EMOS	Ensemble Model Output Statistics
eSAX	Energy Times Series Motif Discovery using Symbolic Aggregated Approximation
ESS	Energy Storage System
FESS	Flywheel Energy Storage Systems
FPSP	Flexibilisation Project Scheduling Problem
LASSO	least absolute shrinkage and selection operator
LPF	low-pass filter
LV	low-voltage
MAPE	Mean Absolute Percentage Error
ME	Mean Error
MILP	Mixed-Integer Linear Program
NWP	Numerical Weather Prediction
PV	photovoltaic
RES	renewable energy sources
RMSE	Root Mean Squared Error
SAX	Symbolic Aggregated Approximation

Part I

Introduction

1 Motivation

Climate change is a global threat to life on earth as we currently know it. Most states around the globe have acknowledged the impending dangers which come with an increase in temperature. Almost 200 countries signed the Paris Agreement by the United Nations (Paris Agreement 2015) to take measures against a temperature rise greater than 2.5 degree Celsius compared to pre-industrial times (see Figure 1.1 for the temperature rise in the past centuries) and to promote environment friendly industries. A major contributor to anthropogenic climate change are so-called greenhouse gases. These gases most prominently include methane (CH_4), water vapour (H_2O) and carbon dioxide (CO_2). In order to achieve the goals of the Paris agreement, measures have to be taken to reduce the emission of all these gases. Additionally, earth's resources are finite, thus there are initiatives to avoid dependencies on resources such as oil or coal.

Energy usage and especially energy production play an important role when it comes to greenhouse gases and the use of finite resources. Traditional energy supply heavily relies on finite resources (e. g. coal) and emits tons of greenhouse gases using them for converting their inner energy to usable energy (e. g. electricity, heat, ...). It thus seems straight forward that a change in how energy is supplied and consumed can help prevent a more severe climate change. Indeed, many societies aim at generating energy without harming the environment or find production solutions for when the natural resources

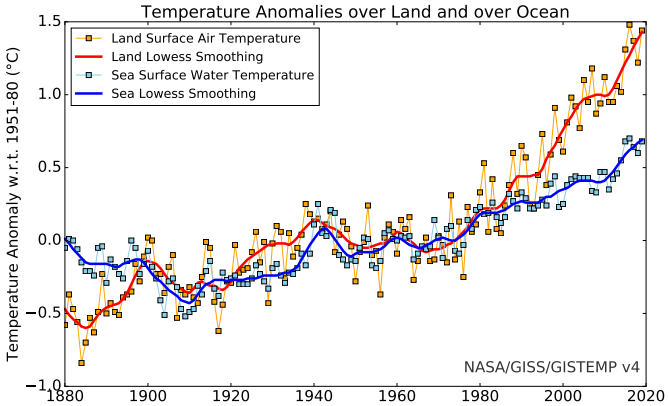


Figure 1.1 Global rise in temperatures since the 18th century. Source: https://data.giss.nasa.gov/gistemp/graphs_v4/

are exhausted. Solar power, as well as wind and water based energy generation, are prominent examples for such renewable energy sources (RES).

A transition from traditional energy sources to renewable ones is not trivial. Various components in today's energy system are build for traditional suppliers and cannot cope with the new requirements imposed by the use of renewable energy sources. The challenges our current energy system encounters with more RES, can broadly be summarised into *where* and *when* energy is needed. The first aspect, *where* energy is needed, is a challenge as different RES are more or less efficient depending on their geographical location. Let us explore this aspect on the example of wind energy. On average, winds are stronger in coastal regions and thus wind turbines located close to the coast or offshore are more effective. However, in some countries such as Germany, the energy intensive industry is mostly located far away from any coast. Thus, while there could be a high energy supply from wind turbines off-shore, there might not be enough demand close to those turbines. Transmission to the parts of an energy system where the energy is needed is possible, but with increasing distance it is also increasingly expensive and decreasingly efficient.

The second challenge, *when* energy is needed, focuses on the fact that renewable energy sources are volatile, intermittent power generators. For example, solar power plants only produce energy when the sun is shining and wind power is only generated when there are moderate winds. Our energy consumption, however, is not synchronized with the sun and wind. Thus, to change the existing energy supply structure one might have to rethink the consumption of energy as well.

Traditionally, the energy supply follows the energy demand, with steerable power plants providing energy whenever necessary. In power grids with a high penetration of renewable energy sources, the supply is not steerable any more, as we have, unfortunately, no way to tell e. g. the wind to increase in speed. Hence, there is often a mismatch between energy available from intermittent RES and current demand. Increasing storage and transmission capacities is one option to ease the integration of renewables, another option is to change the behaviour on the demand-side. The latter option, also called demand-side management, relies on the assumption that consumers can be flexible in their energy usage and can adjust their behaviour to the supply from RES. For example, shifting the energy consumption from early in the morning to noon, when there is PV power available, would result in less transmission and storage capacities needed. However, getting the end consumers to change their behaviour to accommodate more RES supply is difficult, especially when the end consumers are industrial facilities for whom the energy costs are often low compared to other costs such as labour and materials. In general, the ability to adapt the demand to the volatile RES is also called *demand-side flexibility* and will be the main focus of this thesis. The need for more flexibility in energy usage has been well established (e. g. in Taneja (2014)), for an analysis of the benefits see e. g. Strbac (2008) or Feuerriegel and Neumann (2016).

Flexibility in energy systems is discussed frequently as a solution to the volatility imposed through renewable energy systems. It is seen as a key factor for higher adoption rates of RES technology, especially when increasing research and development investments are

taken to "relax technical constraints of flexibility" (Bolwig et al. 2019). Flexibility on the demand side is, however, only one aspect of flexibility in energy systems. The diagram by Heider and Reibsch (2020) in Figure 1.2 shows all areas in which flexibility plays a role, namely *demand*, *storage*, *supply* and *network*. These areas overlap significantly with *sector coupling* and *system operations* being at the intersection of all of them. Nevertheless, the remainder of this thesis is mainly concerned with the left demand part, touching the other areas only when necessary.

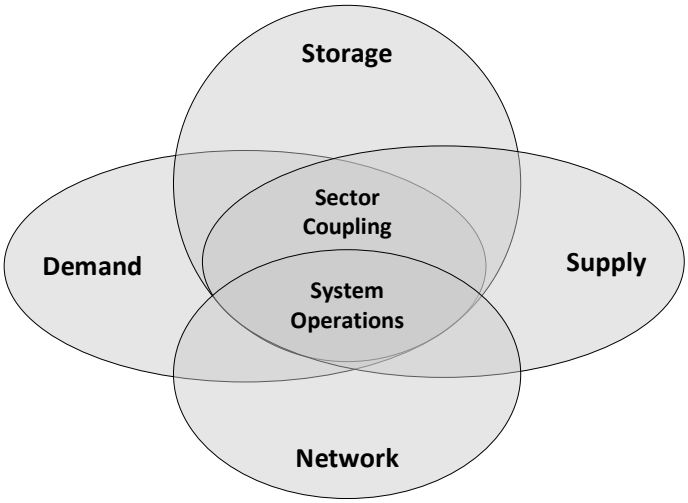


Figure 1.2 Flexibility areas in power systems, according to Heider and Reibsch (2020).

Even with a focus on demand-side flexibility alone, the perspective of the analysis changes a lot in current literature. These perspectives range from smart household appliances (e. g. D'hulst et al. 2015), over buildings (e. g. Chen et al. 2018) to markets (e. g. Eid et al. 2016). While most system approaches usually take several other perspectives into account, market perspectives often connect one or more. An overview of frequently occurring perspectives is given in Figure 1.3.

Regardless of the perspective or question answered, any analysis requires some information on flexibility. Several approaches are used to define or detect flexibility in an energy system. They can broadly be categorised in data-driven and information-driven approaches. The first category is characterised by the main information being found in historical data. In contrast to this data-driven approach, the information-driven approach relies on information being provided based on knowledge or expertise. This knowledge can be given explicitly by the consumer (Alizadeh et al. 2015; D'hulst et al. 2015), or by the manufacturer or process operator (Beier et al. 2017; Chou et al. 2010). The flexibility information can also stem from models, such as process models (Heleno et al. 2015; Kabelitz and Matke 2018) or an EnergyHub model (Dall'Anese et al. 2017). With enough knowledge at hand, one can also determine flexibility functions such as in Junker et al. (2018). More indirect information-driven approaches can also rely on markets to provide flexibility information (Nicolosi 2010), batteries from electrical vehicles (Schuller et al. 2015), or use the willingness to shift as proxy for flexibility (Joe-Wong et al. 2012).

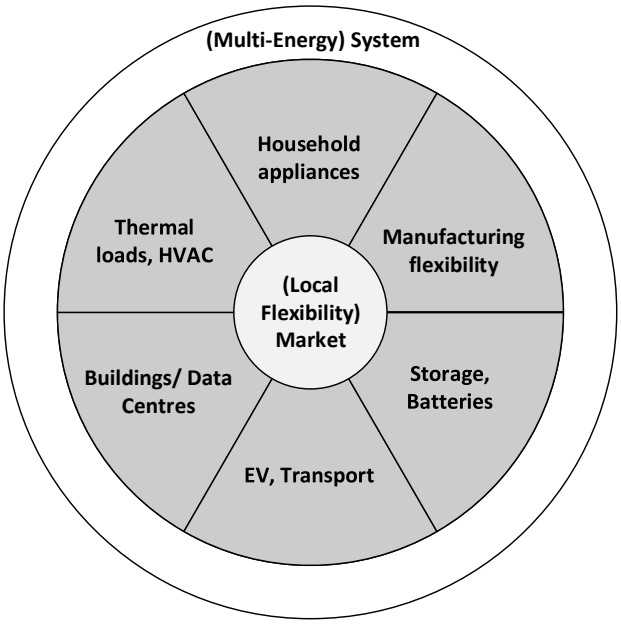


Figure 1.3 Perspectives from which demand-side flexibility is analysed in the current literature.

Lastly, the information can also be predefined (Petersen et al. 2012; Ströhle and Flath 2016) or be based on a reference scenario (Coninck and Helsen 2013).

Only a few papers base their flexibility information in data. For these data-driven approaches simple methods such as clustering daily (load) profiles (Capozzoli et al. 2017; Contreras-Ocaña et al. 2018), or active state detection (Frazzetto et al. 2018; Neupane et al. 2018), but also more advanced methods including neural networks (Förderer et al. 2018) and motif mining (Miller et al. 2015; Shao et al. 2013) are used.

1.1 Research Questions and Contributions

It is the aim of the present thesis to analyse demand-side flexibility in a given grid from a data-driven perspective. The grid under investigation could thereby look as pictured in Figure 1.4, where the grid consists of several actors on the demand side and either conventional energy or renewable energy sources or both supply the grid at any given point in time. The actors on the demand side can be, households, industrial facilities or any other type of consumer. Both demand and supply side are influenced by outside forces such as the weather. Having such a grid in mind, the following three research question guide us through the analysis of demand-side flexibility.

While all the research questions themselves are not new, they have not been answered sufficiently from a data-driven perspective. We thus want to describe each research question in detail and highlight challenges which have to be overcome in order to be able to answer these questions given only information through data.

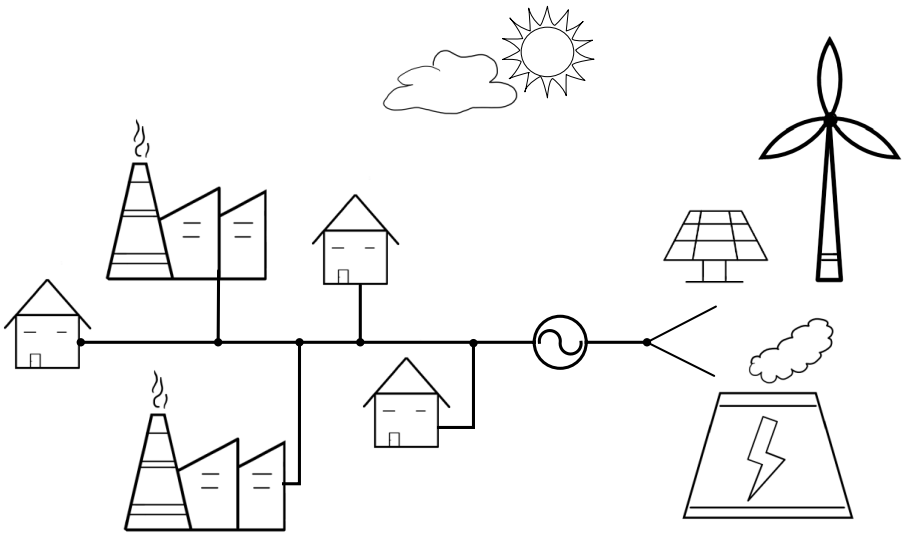


Figure 1.4 A grid to be considered with different demand-side actors, as well as conventional and renewable energy sources.

Research Question 1 [RQ1]: *How much demand-side flexibility is potentially available?*

Before any measures can be taken to increase flexibility, we might want to know how much flexibility is potentially available in a given system. While this question has already been addressed from a system perspective and partly for individual consumers, its analysis so far mostly relies on expert knowledge or broad assumptions. While expert knowledge of a system or machine is clearly the best information obtainable, it is not always easy to get. We thus aim to assess the potential of flexibility from a data-driven perspective, without relying on expert knowledge. This allows decisions makers to estimate the flexibility potential without the need to consult an expert straight away. To answer research question one from a data-driven perspective, we need to establish two items. First, we need to know how to describe flexibility mathematically, i. e. establish features which describe the flexibility of a process or system. Second, we need to find processes and their flexibility features in historical energy consumption data. To answer RQ1, we thus encounter two challenges and aim to contribute to solving these challenges in this thesis.

Modelling Flexibility [C1]. The first contribution of this thesis is a comprehensive modelling framework for demand-side flexibility regardless of the application in which the respective model is used. While there already exists a multitude of flexibility models, they focus on different aspects and applications, in which a new model is needed for any new application. One contribution of this thesis is thus the combination and extension of existing flexibility models such that flexibility features commonly found in literature can be adequately described in a mathematical framework.

Mining Flexibility Potential [C2]. In order to estimate the flexibility potential of processes running on the consumer side, we first need to establish the ground truth from data, i. e. which processes there are. Pattern recognition can help to determine repeating patterns

and processes from historical time series, however detecting flexibility comes with one major challenge. We are looking for patterns in the time series that are assumed to be generated through the same process. However, we are especially interested in changes of these patterns and deviations from any normal shapes, as well as the time when these deviations happen. Given the features which describe flexibility and the patterns which describe processes, some measures need to be established to determine the flexibility potential. One further contribution of this thesis is thus a framework to find processes in energy time series and characterise their flexibility potential.

Research Question 2 [RQ2]: *How much demand-side flexibility is required?*

The second research question is concerned with quantifying the amount of flexibility required in a system. To answer this question, two aspects are of importance. First, any amount of flexibility is only needed, if there is a mismatch between expected demand and expected supply. Thus, proper forecasts to determine the expected demand and supply are important. Second, if flexibility is needed, knowing by how much the current system has to change in order to achieve stability, is important for targeted measures to be most effective. Again, two main challenges arise.

Input Uncertainty in Demand Forecasts [C3]. The first aspect, whether we need any flexibility at all, can be addressed with forecasting. If demand and supply can be adequately forecast for any given point in time, then the need for flexibility results from the predicted mismatch. Unfortunately, forecasts are uncertain in nature, as circumstances such as the weather stem from chaotic systems and forecasting models are never perfect. Accounting for the uncertainty in input features (such as the weather) is challenging as the uncertainty has to be assessed and propagated through the following forecasting model. This thesis contributes to this challenge by introducing statistical post-processing methods to account for input uncertainty in demand forecasting models.

Amount of Individual Flexibility [C4]. To quantify the specific amount of flexibility needed is challenging. While determining the amount of total change that is necessary is trivial (as it is the mismatch between demand and supply), translating this knowledge to what needs to change for an individual consumer is difficult. The present thesis contributes to this challenge by analysing what effects changes to existing schedules have on e. g. the peak demand.

Research Question 3 [RQ3]: *What incentives can increase the usage of the available flexibility, when it is needed?*

The third research question brings together the answers of the two questions posed before. If we know that there is a potential for flexibility and we know how much flexibility is needed, then how can we incentivise the demand side such that this potential is used when needed? In order to answer the question we want to take a look at mechanisms which give sufficient incentives to an actor on the demand side to use her flexibility. Testing these incentives on a real grid is challenging, thus agent-based models and reinforcement

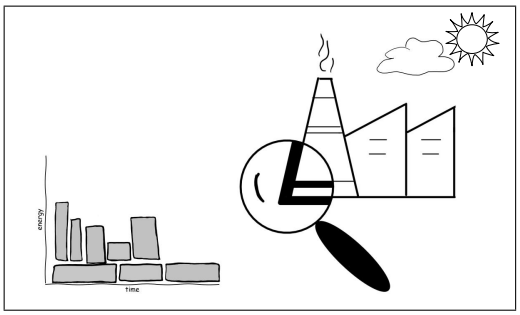
learning could be used to realistically simulate behaviour on a small local grid. However, if the incentives are not enough, the challenge of balancing supply and demand remains.

Designing Mechanisms [C5]. So far flexibility markets mostly focus on households and expect the user to specify start and end times or upper and lower bounds. This practice is both privacy inflicting and time consuming. One challenge this thesis contributes to is thus the design and evaluation of mechanisms that increase the usage of available flexibility without the need for inputs from users.

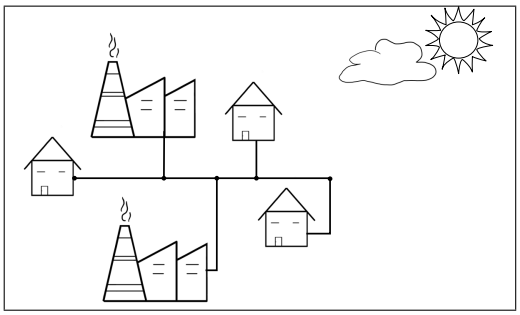
Lack of Incentives [C6]. Even with mechanisms in place, there might still be a mismatch. This thesis thus contributes to the challenge of lacking incentives by introducing properly sized distribution grid storages. These storages could enable more flexibility by aggregating the potential from several individual consumers, while additionally contributing to e. g. grid frequency stability.

1.2 Thesis Outline

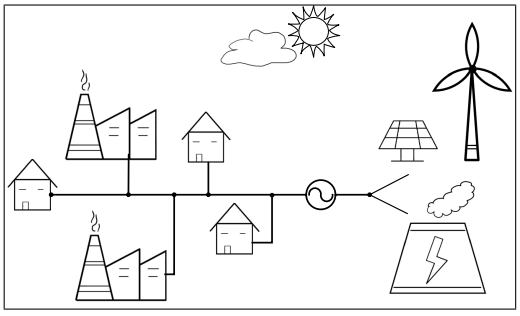
The three research questions [RQ1]-[RQ3] and corresponding challenges [C1]-[C6] guide us through the present thesis, as shown with the graphical outline in Figure 1.5. Each of the three research questions will be addressed in one separate part.



(a) Part II: Potential demand-side flexibility.



(b) Part III: Required demand-side flexibility.



(c) Part IV: Balancing potential and required demand-side flexibility.

Figure 1.5 Graphical outline of the three main parts of this thesis.

We start with the first question [RQ1], thus focusing on individual consumers and their flexibility potential in Part II. We will especially look at challenge [C1] in Chapter 3 and [C2] in Chapter 4. In Part III, we turn to the second research question [RQ2] on required flexibility including challenges [C3] and [C4] in Chapter 6 and Chapter 7 respectively. This part thus focuses on the demand side from a system's perspective, and outside influences such as the weather on the need for flexibility. The third research question [RQ3] brings the two previous parts together in Part IV and focuses again on the whole grid while addressing challenge [C5] in Chapter 8 and [C6] in Chapter 9. Lastly, the contributions of this thesis are discussed and summarised in Part V.

2 Preliminaries

In order to answer the three research questions [RQ1]-[RQ3], we have to introduce some known theoretical concepts on which we will build in this thesis and the notation used. We start with the main object under consideration i. e. a time series on a probability space, before defining what we understand as flexibility.

2.1 Time Series

In the present thesis, the notation is derived from Lerch (2016), which is based on the mathematical framework from Murphy and Winkler (1987), Gneiting and Ranjan (2013), and Strähl and Ziegel (2017). The general setting considers the joint distribution of forecasts and observations on the probability space $(\Omega, \Sigma, \mathcal{P})$. The elements of the sample space Ω can be identified with tuples

$$(F_1, \ldots, F_k, Y), \tag{2.1}$$

whose distribution is specified by the probability measure $\mathcal{P}$. The probabilistic forecasts $F_1, \ldots, F_k$, are probability measures on the outcome space (Ω_Y, Σ_Y) for the random variable observation Y. Realizations of the random variable Y are indexed by t such that a time series can be denoted by $y(t) = y(1), \ldots y(n)$. We restrict the analysis in the present thesis to all real-valued observations where $\Omega_Y = \mathbb{R}$, and denote F the probabilistic forecast with cumulative distribution function (CDF) F and probability distribution function (PDF) f. In the present thesis, we mainly consider the observations Y, apart from Chapter 6, which considers the joint distribution of forecasts and observations.

Furthermore, we often analyse time series subsequences, thus parts of the time series. A m-subsequence of a time series Y can be described as

$$\begin{aligned} y_m := \\ \{\langle y(i), y(i+1), \ldots, y(i+m)\rangle \,|y(i), y(i+1), \ldots, y(i+m) \in Y, i, m \in \mathbb{N}\}\,. \end{aligned} \tag{2.2}$$

The subsequences in the present thesis, often describe processes in a system, where a *process* is defined as follows.

Definition 2.1 (Process). A process is a sequence in time, where a series of actions or steps is taken by an entity, i. e. a machine, in order to achieve a specific result. For example, a coffee machine making a cup of coffee can be described as a process.

The description of all other notations and symbols is provided in the chapter they are used.

2.2 Defining Flexibility

The term *flexibility* plays an essential role in the present thesis. However, there is no general definition what flexibility is in the realm of energy systems. In general, according to the Oxford English Dictionary, the term refers to *being able to bend without breaking* as well as *the ability to be easily modified*, or *the willingness to change*. We will see that all three aspects play an important role, when investigating flexibility in the energy system.

The energy system relies on a balanced demand and supply side at all times. To achieve this balance, the traditional energy system relies on the ability of the supply side actors to adapt according to changes on the demand side. We refer to this ability to adapt as *Supply-Side Flexibility*, which we define as

Definition 2.2 (Supply-Side Flexibility). The ability of supply-side actors to modify their respective energy generation either by generating more or less energy at a given point in time, or by switching between different sources of energy.

In the traditional energy grid, the adaptation takes place on the supply side because traditional energy generation can be controlled. However, the set-up of supply following demand describes the traditional energy system without decentralized energy generation. Nowadays, the energy supply can also come from highly fluctuating decentralised RES, such as photovoltaic power systems (PV). These intermittent energy sources cannot be controlled centrally, as e. g. sunshine is not controllable by any system operator. The traditional concept of adjusting the supply side to the changing demand is hence not possible with only renewable energy sources. Thus, the focus has gone back to the consumer and her ability to change the behaviour to balance the supply from the RES with her demand. We refer to this ability to adapt the behaviour to predictable or unpredictable changes as *Demand-Side Flexibility*, which in the course of this thesis is described similarly to the definition from Neupane et al. (2014) as

Definition 2.3 (Demand-Side Flexibility). Demand-side flexibility is the possibility to adapt the energy demand behaviour, either by changing *when* the energy is used (time flexibility) or *how* the energy is used (operational flexibility).

A consumer, be it a single household device or a complex industrial facility, is said to *have flexibility* if one of the above types of changing the energy behaviour is feasible. Note that in our demand-side flexibility definition we do not consider the option of switching

between sources or types of energy. All energy types are treated equally, and it does not matter for us whether a process needs to be powered by e. g. steam or electricity.

Having established what we consider as flexibility, we have a look on how to achieve more flexibility in energy systems. Any measure to control the behaviour of energy consumers is often called demand-side management (DSM). Additionally, incentives and strategies to use more flexibility are usually also summarised under the same term. The definition of DSM varies considerably in the literature. This thesis uses the following definition (in line with Palensky and Dietrich (2011)).

Definition 2.4 (Demand Side Management). DSM considers all methods that directly or indirectly change the demand side behaviour. A method is a *direct method* if it is implemented or carried out outside the control of the actors on the demand side but influences or limits their behaviour explicitly. A method is an *indirect method* if a mechanism is implemented outside the control of the actors on the demand side but sends signals to these actors which can then respond to those signals. However, the reaction to these signals is not predefined and the actors can often choose not to act.

Part II

Potential Demand-Side Flexibility

Overview Part II

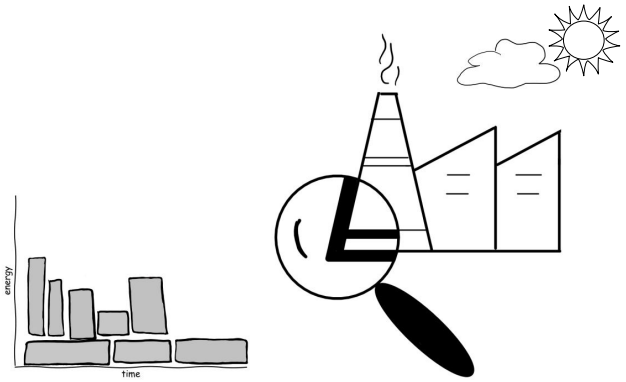


Figure 2.1 Graphical view on the Part II of the present thesis: potential demand-side flexibility.

As established in the introduction, to ease the integration of renewable energy sources (RES), flexibility on the demand side is a promising approach. However, there is hardly any information on how much flexibility is available in a given factory, household or local grid. We have seen that on large scales, flexibility can be assumed from the system settings and on the consumer side we may assume that some processes such as a washing machine are rather flexible. However, from a data perspective, little is known on how flexible processes are currently used. Additionally, while households can certainly shift some of their demand, the potential is larger in industry. We thus focus on industrial use cases.

In this part of the thesis, we investigate flexibility potentials on the demand side from a data-driven perspective, answering [RQ1] and contributing to the challenges [C1] and [C2]. We propose a mathematical modelling framework to describe demand-side flexibility in Chapter 3, before introducing an algorithm to detect processes and their flexibility potential in Chapter 4. This algorithm is then evaluated on real-world industrial data, without further information from an expert or user. Lastly, we introduce a new benchmark data set for flexible process instances in Chapter 5, to help foster more flexibility research on real-world instances in the future.

3 Modelling Flexibility

While many societies aim at shifting their energy mix towards renewable energies, the currently implemented system relies on a centralised dispatch of electricity generation (Schleicher-Tappeser 2012). The integration of the increasing decentralised renewable energy sources into the energy system is therefore one of the two most important research fields in energy informatics (Goebel et al. 2014). High fluctuations in supply, as well as strong intra-day patterns e. g. in the case of solar energy, are challenges for a smooth integration (Denholm et al. 2010). The traditional consumer behaviour is a challenge for the power grid as it results in high peaks and low valleys of the electric load. Currently, this fluctuation is compensated by conventional steerable power plants to ensure a reliable operation of the electricity grid. As more and more intermittent renewable sources generate electricity, this balancing technique is threatened (Weidlich et al. 2012). However, the decrease in supply side flexibility of intermittent generation might be offset by an increase in demand side flexibility. Therefore, one possibility to ease the integration of RES is to control the consumer demand and adapt it to the supply side (Strbac 2008). Thus, the aim of an optimal supply strategy can be reached by providing more flexibility on the demand side. For example, we might use a heat pump whenever the sun is shining instead of when it is most convenient for the consumer. In this chapter, we will focus on demand-side management (DSM). Our objective is to design a holistic modelling framework to schedule demand-side flexibility. There is extensive research discussing DSM applications from a scheduling perspective (e. g. Petersen et al. 2013).

Scheduling energy loads, hence exploiting the flexibilities in the system, to enhance grid stability or reduce energy costs for the consumer is not a new idea. However, in the mathematical set-up to solve these tasks, related works employ application-specific formulations to describe the loads and their characteristics to be scheduled (see Section 3.1 for a more detailed literature review). This practice results in a vast amount of different modelling formulations. Additionally, most authors focus on a single application. Thus, their models are not readily transferable to new data sets or different

Parts of this chapter are reproduced from

L. Barth, N. Ludwig, E. Mengelkamp, and P. Staudt (2018b). "A comprehensive modelling framework for demand side flexibility in smart grids". In: *Computer Science - Research and Development*, Vol. 33, No. 13, pp. 1865–2042. DOI: 10.1007/s00450-017-0343-x

use cases. In this context, it is especially noteworthy that demand-side flexibility of private households and industrial applications exhibit very different characteristics, i. e., household appliances can usually run independently from each other while industrial processes often depend on other production steps. As the considered papers always focus on only one application, no formulation exists that integrates all of these features. This variety of formulations in the literature makes it difficult to compare the modelling approaches, their respective results and adaptability.

We present a novel comprehensive modelling framework in the field of energy informatics to represent flexibility in a household as well as in an industrial context. Based on current literature, we classify the most important characteristics of flexibility represented in various models and incorporate the majority in a single modelling framework. We combine currently existing, wide-ranging research and, to our knowledge, are the first to integrate the different approaches into a single modelling framework.

The remainder of the present chapter is structured as follows. In Section 3.1, we give a short overview of existing literature concerning demand-side flexibility and management. Following this, we describe common features found in the literature describing demand-side flexibility in Section 3.2. Section 3.3 introduces our modelling framework which is evaluated according to its performance in the following Section 3.4. We discuss our work in Section 3.5, before giving an outlook and a conclusion in Section 3.6.

3.1 Related Work

Demand Side Management (DSM) and Demand Response (DR) become increasingly important as more electricity is generated from intermittent sources. This development has been accompanied by a growing interest from researchers. A variety of authors dealt with demand flexibility of private households. Consequently, they ignore most characteristics of industrial loads. For example, He et al. (2013) provide a classification of household flexibility along different dimensions, while Allerding et al. (2012) focus on developing demand response for private households. Gottwalt et al. (2017) also concentrate on private households, however, they incorporate several additional restrictions. Scott et al. (2013) characterise the flexibility of individual household devices. However, the description is tailored to specific appliances and therefore not domain independent. In Fehrenbach et al. (2014) the authors show that thermal appliances and specifically the expansion of heat pump use may have the largest flexibility potential of private households. Du and Lu (2011) provide a scheduling algorithm for those thermal appliances. This work is extended by Alizadeh et al. (2015), who differentiate between curtailable thermal loads and other deferrable loads. Household behaviour with regards to the provision of flexibility and effects on electricity costs is simulated by Gottwalt et al. (2011). They conclude that saving potentials for households are moderate when compared to the investment in smart meter technology. Contrary to this result, Setlhaolo et al. (2014) come to the conclusion that a reduction of up to 25% of the electricity costs of private

households is possible. The investigation by Soares et al. (2014) also considers customer dissatisfaction besides the monetary compensation.

Demand-side flexibility as a means to integrate renewable generation is established by Palensky and Dietrich (2011). Other research has established that fluctuations of a low penetration of renewable generation can be offset by demand-side flexibility, as for example shown by Strbac (2008). However, the authors argue that a monetary compensation is difficult to determine. Halvorsen and Larsen (2001) describe the effects of appliance endowment and additional investment on the ability to provide flexibility. A new approach for a scheduling algorithm was developed by Ströhle et al. (2014) to match uncertain supply with different demand packages to maximise total welfare. The optimal combination of private household flexibility is investigated by Gärttner (2016) and extended in Gärttner (2016) to provide recommendations to flexibility portfolio aggregators.

An extensive description of characteristics of demand-side flexibility beyond residential flexibility is given by Petersen et al. (2013). The authors also develop a first taxonomy for flexibility but chose not to incorporate a variety of characteristics of flexibility (Petersen et al. 2014). Paulus and Borggrefe (2011) establish that demand side management bears considerable monetary potential in energy intensive industries. Qureshi et al. (2014) develop a model to investigate economic potential of demand side management in office buildings. Ashok and Banerjee (2000) pioneer the field of industrial demand side management. Their model is specified in Ashok (2006) but leaves certain restrictions for future research. In Schilling and Pantelides (1996) we find appropriate scheduling algorithms for our problem formulation. However, as they are not specifically developed for electricity loads, individual extensions to the model are necessary. Mitra et al. (2012) and Moon and Park (2014) consider scheduling with regards to electricity costs for industrial production. Oudalov et al. (2007) use batteries to reduce demand peaks.

We present the most relevant models of demand-side flexibility with regards to the restrictions and characteristics they incorporate in Table 3.1 using criteria presented in the next section. The aim of this section is to integrate the features considered in the described models into one holistic modelling framework which allows us to describe flexibilities across all domains, rather than developing another alternative model of demand-side flexibility.

3.2 Modelling Flexibility

We consider a total of 14 features found in the literature (Section 3.1) which help us to describe the flexibility of any process. In the following we describe each of the features and differentiate between features enhancing flexibility and features restricting flexibility. Table 3.1 summarizes the models that can be found in literature and highlights which features are incorporated in each formulation.

Features enhancing flexibility:

- **Interruptible Jobs.** A job is interruptible if it can be stopped at either a given time step or any time step. The job can be, but does not have to be, continued later on. Interrupting a job would provide time flexibility and allows for example short energy consumption reductions in case of sudden energy supply drops.
- **Modes.** For some jobs it might be possible to operate them in different modes. For example the same job could be run for a short amount of time using a lot of resources, or for a longer amount of time using only little of the same resources. Assuming that neither the product quality nor any other product features depend on the operation mode, this enhances the operational flexibility.
- **Storage**. Any possibility to store resources or output enhances a systems flexibility. For example, storing energy can allow a process to be started even when the supply is insufficient.

Features restricting flexibility:

- **Interdependent Jobs.** In any real-world scenario, jobs may depend on other precessing or succeeding jobs. These dependencies usually restrict flexibility as some orders of jobs cannot be changed even if it would be energetically beneficial.
- **Earliest Start Time.** This feature describes the possibility that a job cannot be started earlier than a specific time step. The later this earliest start time is, the more is restricts the flexibility of the job.
- **Deadline**. Closely related to the earliest start time, some jobs might have be finished by a certain time step. The deadline can also be an overall deadline for all jobs. Any form of deadline restricts the time flexibility.
- **Production.** For example, each job or several jobs can be associated with a specific production output. This output can also be fixed if each job has to run exactly once. Production information itself is not restrictive, however, as soon as a specific production output has to be attained by a certain time step, this limits the flexibility options.
- **Multiple Resources**. Jobs might need or be able to use input from multiple resources. If they can use input from several resources this would improve flexibility. However, the most likely scenario is probably that a job needs multiple resources to run which would limit its time flexibility to only a few points where all these resources are available.
- **Base Loads.** Base load or uncontrollable loads, are inflexible loads that must run the whole time or at specific time steps. These loads do not provide any flexibility and might also restrict the flexibility of other jobs in the system.
- **Drain, Losses.** Interrupting a job might come with some drain, loss of energy. Any restarting or continuing might thus lead to more energy needed later as was the case if the job did not get interrupted (e. g. temperature based processes). Drains

and losses can restrict the flexibility and have to be accounted for to properly assess the flexibility of e. g. interruptible jobs.

- **Down-/Uptime.** Some jobs might have to run for a specific time to function properly or need to be off for a specific time after usage (e. g. to cool down). These forced down- or up-times restrict the time flexibility of a job or the system.
- **Ramping.** A job can have a slowly increasing resource intake at the beginning or slowly decreasing intake of energy at the end. Some interdependent jobs might make ramping up or down unnecessary and the runtime usually increases with any ramping of the job. If ramping increases or decreases the flexibility potential highly depends on the specific use case and energy supply available.

The table also mentions a time frame and multiple runs as features. Both these features mainly influence the optimisation model and mathematical model formulation but not the flexibility of the system. Given these features, a comprehensive modelling framework is established and an optimisation model can be derived, which we introduce in the next section. In general one can conclude that the more processes one can describe with enhancing flexibility features and the less restricting features are needed, the more flexibility is in the system.

Table 3.1 Comparison of the integrated flexibility features in work related to our modelling framework.

Reference	Time Frame	Interrupt. Jobs	Modes	Storage	Interde. Jobs	Earliest Start	Deadline	Production	Multi. Resources	Base loads	Drain, Losses	Down-/Uptime	Ramping	Multiple Runs
Allerding et al. (2012)	disc.	✓	✗	✗	✗	✓	✓	✗	✗	(✓)	✗	✗	✗	✗
Ashok and Banerjee (2000)	disc.	✓	✗	✓	✗	✗	✗	✓	✗	✗	✗	✗	✗	✓
Ashok (2006)	disc.	✓	✗	✓	✗	✗	✗	✓	✓	✗	✗	✗	✓	✓
Castro et al. (2002)	cont.	✗	✗	✓	✗	✗	✗	✓	✓	✗	✗	✗	✗	✓
Fink et al. (2014)	disc.	✗	✗	✓	✗	✗	✗	✗	✗	✓	✓	✓	✓	✓
Gottwalt et al. (2017)	disc.	✓	✗	✓	✗	✓	✓	✗	✗	✓	✗	✓	✗	✓
Luo et al. (1998)	disc.	✗	✓	✗	✗	✗	✓	✓	✗	✗	✓	✓	✗	✓
Mitra et al. (2012)	disc.	✗	✓	✓	✗	✗	✓	✓	✓	✓	✗	✓	✓	✓
Moon and Park (2014)	disc.	(✓)	✓	✓	✓	✗	✓	✓	✓	(✓)	✗	✓	✓	✓
Oudalov et al. (2007)	disc.	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✓	✗	✓
Petersen et al. (2013) 2014	disc.	✗	✗	✓[a]	✗	✗	✓	✗	✗	✓	✗	✓	✗	✓
Schilling and Pantelides (1996)	cont.	✗	✗	✓	✗	✗	✓	✗	✗	✓	✗	✗	✗	✓
Sou et al. (2011)	cont.	(✓)	✓	✗	✓	(✓)	✓	✗	✗	✓	✗	✓	✗	✓
This chapter	disc.	(✓)	✓	(✓)	✓	✓	✓	✗	✓	✓	✓	✗	✓	✗

✓refers to features that can be explicitly modelled by the framework, ✗ to features that cannot be modelled, and (✓) to features which can be modelled implicitly using other already defined features or with restrictions.

[a] Only integrated in the first paper by the authors.

3.3 Optimisation Model

In this section, we describe our proposed modelling framework which incorporates the majority of features found in the literature, as summarised in Table 3.1. The optimisation model derived from the modelling framework is a Mixed-Integer Linear Program (MILP) and *jobs* form its basis.

Definition 3.1 (Job). A job represents an uninterruptible process that requires a certain amount of electrical power to run. Each job is associated with a duration. Given an instance with n jobs in which job i can be run in $M_i \geqslant 1$ different modes and the latest job deadline is D_{max}, the decision variables consist of two groups of binary variables. A group $s_i(t)$ of variables indicating whether job i is started in time step t and a group of variables $m_{i,j}$ indicating whether job i is run in mode j. This limits the number of decision variables to $n \cdot D_{\text{max}} + \sum M_i$.

All other variables in the modelling framework can be derived from these decision variables and are summarised together with the model inputs in Table 3.2.

The features described in Section 3.2 are modelled as constraints of the MILP. Overall, the number of binding constraints is quadratic to the number of jobs. In the following, we describe the characteristics of the modelling framework in detail, with an overview of the variables used in Table 3.2. The jobs can get their required power from different resources, where each resource adds to the overall power needed by the job. For simplicity's sake, we focus on the case of only one power resource, such that the overall resources needed at time step t are $\hat{P}(t)$.

Objective Function. We can incorporate multiple objective functions into the framework and introduce two of them in the following. The two objectives could also be combined if necessary. The first objective function regards the problem which we call *overshoot minimisation*, while the second is concerned with *peak shaving*.

In the case of our overshoot minimisation objective, we want to change the way the processes run so that we can use most of the energy from renewable energy sources, e. g. photovoltaics, instead of buying the energy from the grid. The primary goal is then to use the minimum possible energy from the grid by exploiting the inherent flexibility of the processes. We thus minimise the difference between "free" electricity from a renewable non-steerable resource $P_G(t)$ and the power $\hat{P}(t)$ needed to run the desired processes. In our case we do not explicitly include storage but all previously stored energy could be added to the renewable source side of the equation. Using energy from the grid is penalised with a cost function $c(t)$. The objective function is then

$$\min \sum_t c(t) \cdot \Big(\max \Big(\hat{P}(t) - P_G(t), 0 \Big) \Big). \tag{3.1}$$

Table 3.2 Variables used in the modelling framework, with the model input variables in the top, the decision variables in the middle and the derived variables in the bottom part of the table.

	Model Inputs
N	Number of jobs
$\tilde{P}_i$	Base power requirement of job i
$\tilde{T}_i$	Base run time of job i
M_i	Set of mode coefficients of job i
$\phi_{i,m}$	mode coefficient for time adjustment of job i in mode m
$\psi_{i,m}$	mode coefficient for power adjustment of job i in mode m
D_i	Deadline of job i
R_i	Release time of job i
$L_{i,j}$	Minimum time lag between job i and j, measured in time steps from the end of i to the start of j
$\tau_{i,j}$	Runtime extension coefficient for the separation of jobs i and j
Λ_i	Maximum number of ramping steps for job i
$\delta_{i,j,k}$	Number of time steps between the end of job i and the start of job j before job j must execute ramping step k before executing the actual job
$\mu_{i,k}$	Power requirement of job j's k-th ramping step
$P_G(t)$	Power generation available from renewable energy sources at time step t
$c(t)$	Cost function for using energy above capacity limit (i. e. production and storage)
	Decision Variables
$s_i(t)$	Binary variable, becomes 1 if and only if job i starts at timestep t
$m_{i,j}$	Binary variable, indicating if job i is to be run in mode j
	Derived Variables
$\tilde{\phi}_i$	Effective time adjustment coefficient of job i
$\tilde{\psi}_i$	Effective power adjustment coefficient of job i
P_i	Power requirement of job i in its selected mode
T_i	Run time of job i in its selected mode
$\hat{P}(t)$	Total power requirement at timestep t
σ_i	Timestep in which job i starts
η_i	First timestep in which job i is finished
M	Large constant used to switch constraints on an off
$\rho_{i,k}$	Binary variable indicating whether job i must execute its k-th ramping step
$\omega_i(t)$	Ramping power if job i at timestep t

Additionally, we can also use peak shaving as a second objective to our scheduling. Minimising the peaks during our time frame would lower our overall energy costs and might make it easier to rely on renewable generation entirely even when there are only few storage capacities and production available. The objective function for peak shaving is given by

$$\min \left(\max_t \hat{P}(t) \right) \tag{3.2}$$

Everything else is modelled in terms of constraints of the mixed-integer program. We start with describing the constraints that combine the decision variables with the energy resource usage $\forall i,j \in 1, \ldots n$.

$$\sum_t s_i(t) = 1 \tag{3.3}$$

$$\sum_j m_{i,j} = 1 \tag{3.4}$$

$$\sigma_i = \sum_t t \cdot s_i(t) \tag{3.5}$$

$$P_i = \tilde{\psi}_i \cdot \tilde{P}_i \tag{3.6}$$

$$\eta_i = \sigma_i + T_i \tag{3.7}$$

$$\eta_i \leqslant D_i \tag{3.8}$$

$$\sigma_i \geqslant R_i \tag{3.9}$$

$$\tilde{\psi}_i = \sum_j m_{i,j}\psi_{i,j} \tag{3.10}$$

$$\tilde{\phi}_i = \sum_j m_{i,j}\phi_{i,j} \tag{3.11}$$

Equation (3.3) ensures that each job is scheduled once during our optimisation period, with the starting time given by Equation (3.5) as summing over all time instances multiplied by the indicator whether the job starts in this instance results in the time instance σ_i that the job starts in. Similarly, Equation (3.4) ensures that for every job, exactly one mode is selected. Each job needs a power input P_i which depends on the modus m_i the job is running in and its base power $\tilde{P}_i$ (3.6). The power input for some modi can also be negative i. e. we can store/ drain energy. If the energy is later used again it can be added to the overall generated power $P_G(t)$. Before the overall schedule deadline is reached, all jobs have to be finished, with their finishing time η_i depending on the length of the job T_i (Equation (3.7), (3.8)). Also, no job can start before its release time (3.9). Equation (3.10) and (3.11) set the effective time and power coefficients depending on the selected mode.

The model consists of five more constraints that model the specific features from Section 3.2. Since these constraints are more complex, we explain their details in the following.

Interdependent Jobs. Given two jobs i and j, we allow to specify a *minimum time lag* $L_{i,j}$ between i and j, specifying that j may only be started at least $L_{i,j}$ time steps after the start of i. Transformed into an MILP constraint, it looks like

$$\sigma_i + L_{i,j} \leqslant \sigma_j, \quad \forall i \in \{1, \ldots, n\} \tag{3.12}$$

and directly translates to *the start of j must be at least $L_{i,j}$ time steps after the start of i*.

Time Extension for Drain and Modes. Let $\tilde{T}_i$ be the *base time requirement* for job i, and $\phi_{i,j}$ (respectively $\psi_{i,j}$) the power (respectively time) *mode coefficients* for the mode being run in. These coefficients determine how the power requirement (respectively

run time) changes if mode j is selected, i.e., if $m_{i,j} = 1$. Additionally, the actual runtime may depend on one or several *drain factors* $\tau_{a,i}$. The drain factors indicate a runtime extension of i if job i is not started immediately after job a, as the energy that drained between the execution of a and i has to be replenished. However, at some point all the energy is drained resulting in a limit imposed by $\hat{\tau}_{a,i}$. In total, the resulting constraint on the runtime T_i of i is then

$$T_i = \tilde{\phi}_i \cdot \tilde{T}_i + \sum_k \min(\tau_{k,i} \left(\sigma_i - \eta_k\right), \hat{\tau}_{k,i}) \quad \forall i \in \{1, \ldots, n\}. \tag{3.13}$$

Here, the sum of the right hand side of Equation (3.13) sums over all jobs k that might precede i. For jobs that do not precede i, or for which no drain is desired, $\tau_{k,i}$ should be set to zero, thereby making those terms irrelevant. Thus, the last part computes the time lag between the end of job k and the start of job i. Note that this part can never become negative, because k being a predecessor of i forces i to start only after k has finished, i. e., $\sigma_i \geqslant \eta_k$. While the minimum function is not linear, it can be linearised using standard techniques.

Ramping. The ramping of job j is a series of dummy jobs describing the steps in the ramping job. Whether the λ-th ramping job must be executed is denoted by $\rho_{j,\lambda}$, where $\lambda \in \{1, \ldots, \Lambda_j\}$. Here, Λ_j is the maximum number of steps necessary to reach the power input needed for job j to start. At which ramping step we start depends on the time distance between the end of the last dependent job η_i and the start time of the job that needs ramping σ_j. We check if we execute ramping step λ by introducing one of the following constraints for every predecessor i of j

$$\rho_{j,\lambda} \cdot M \geqslant (\sigma_j - \eta_i - \delta_{i,j,\lambda}), \quad \forall i,j \in \{1, \ldots, n\}, \forall \lambda \in \{1, \ldots, \Lambda_j\} \tag{3.14}$$

where M is a suitably large constant. Then, $\rho_{j,\lambda}$ must become 1 if the right side of Equation (3.14) is larger than 0, i. e., if i and j are separated by more than $\delta_{i,j,\lambda}$ time steps. The parameter $\delta_{i,j,\lambda}$ can grow very large, however it is only relevant if a dependency to another job and ramping exist. We assume that the λ-th ramping step of job j must be executed λ time steps before the start of job j. With this, the amount of power required for ramping job j at time step t can be formulated as

$$\omega_j(t) = \sum_{\lambda=1}^{\Lambda_j} \rho_{j,\lambda} \cdot s_j(t + \lambda) \cdot \mu_{j,\lambda} \quad \forall j \in \{1, \ldots, n\}, \forall t. \tag{3.15}$$

Here, $\omega_j(t)$ becomes $\mu_{j,\lambda}$, i. e., the power for j's λ-th ramping step, if and only if j is started in time step $t + \lambda$ and $\rho_{j,\lambda}$, i. e., the indicator for executing the λ-th ramping step is 1.

Total Power Requirement. The total power requirement in the system at time step t is described as the sum over the power of all running jobs at time step t and the power used of the jobs currently ramping

$$\hat{P}(t) = \sum_i \left(P_i \sum_{t-T_i<t' \leqslant t} s_i(t') \right) + \sum_j \omega_j(t) \quad \forall t. \tag{3.16}$$

Linearisation. Some of the constraints described above are not linear per se. See for example Equation (3.15), where $\rho_{j,\lambda}$ and $s_j(t)$ — both variables, not constants — are multiplied. However, for two binary variables a and b, such a multiplication can easily be linearised if the product contributes only positively to the objective function, i. e., if a solution is preferred for which the product is 0.

Let c be a third binary variable indicating whether the product $a \cdot b$ is 1. Then it is enough to introduce the constraint $c \geqslant a + b - 1$. We can replace $a \cdot b$ with c everywhere. If a and b are both 1, then c must be 1. In all other cases, c will be set to 0, since an optimum solution prefers the product to be 0.

3.4 Experimental Evaluation

We experimentally evaluate the MILP resulting from our modelling framework by generating random instances, running the MILP for 15 minutes and measuring the optimality gap, i. e., the gap between best feasible solution found and best shown lower bound. We evaluate the framework with peak shaving as objective function (see Equation (3.2)). This is due to cost minimization and appropriate weighting of the objectives being very problem specific and harder to generalize. We conduct all experiments on a machine with 16 Intel Xeon E5-2670 cores at 2.6 GHz and 64GB of RAM, using Gurobi 6.5 as a solver. [1]

We generate five separate sets of instances. For each of the five sets, Table 3.3 shows the number of jobs, number of dependencies between two jobs, number of dependencies that are associated with a drain, and the (net) *slack* jobs have in the instances. The *slack* of a job is its deadline minus the release time minus the run time of the job. The slack gives an indication of the amount of freedom one has during scheduling. The *net* slack compensates for the fact that in the presence of dependencies, the earliest possible start time of a job does not just depend on the release time, but also on the start times of its predecessors. Thus, a lower bound for the earliest start time of a job is the maximum of all its predecessors' earliest start times plus their respective run times. The *net* slack takes this lower bound *and* the release time into account.

In Table 3.3, intervals like $[a,b]$ indicate that the value is chosen uniformly at random between a and b for every instance. A set like $\{a, b, c\}$ indicates that we generate instances

1 http://www.gurobi.com

Table 3.3 Properties of the five sets of generated instances. Intervals $[a,b]$ indicate numbers chosen uniformly at random between a and b, inclusively.

Name	# Jobs	# Dep.	# Dep.with Drain	Net Job Slack
Jobs (Set A)	$\{50, 100, 150, 200, 250, 300\}$	$[0, 1000]$	0	$[0, 30]$
Dependencies (Set B)	200	$\{0, 100, 500, 1000, 2000, 3000\}$	0	$[0, 30]$
Drain (Set C)	200	1000	$\{0, 100, 200, 500, 900\}$	$[0, 30]$
Slack (Set D)	200	$[0, 10000]$	0	$\{1, 25, 50, 75, 100\}$
Slack w/few Dependencies (Set E)	200	200	0	$\{1, 25, 50, 75, 100\}$

for each of the values a, b and c. For each such value, we generate 30 instances, for a total of 810 instances. We set the objective for all instances to minimise the peak power requirement. The power requirement for every job is drawn from a normal distribution with mean 5 and standard deviation 2, where negative values are discarded.

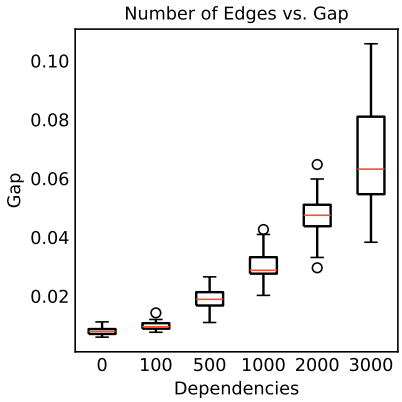
In the following, we analyse the *gap* between best found feasible solution and best lower bound. Formally, let C_{bound} be the cost of the best lower bound and C_{feasible} be the cost of the best found feasible solution, then the gap is defined as $1 - (C_{\text{bound}}/C_{\text{feasible}})$. For instances where no bound or no feasible solution was found, we set the gap to 1.

Figure 3.1a shows the effect of the number of dependencies on the gap achieved after 15 minutes. We can see that for up to 100 dependencies, all instances stay below a 2% gap. Even for 1000 dependencies, almost all instances can be solved up to a 4% gap. However, the gap increases superlinearly with the number of dependencies.

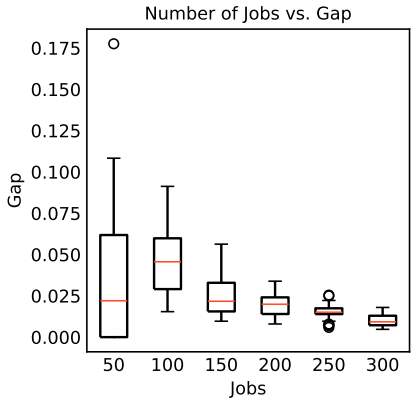
In Figure 3.1b, we present the same plot for a varying number of jobs. A counterintuitive result is that while the gap first increases from 50 to 100 jobs, it decreases from there on. An explanation for this is the fact that the gap is a relative measure. As we keep adding jobs (keeping the global deadline and release time fixed), the absolute value of the optimum solution increases. A fixed (absolute) difference between the best feasible solution and the best lower bound becomes a lower gap as the optimum solution increases, which manifests here. However, note that even in the worst case, with 100 jobs, the majority of the instances could be solved to a gap of 5% or below.

Figure 3.1c shows the *net slack* of all jobs versus the achieved gap. It is visible that large net slacks strongly increase the computational complexity of the model. Note that the mean duration of all jobs is 10, i.e., a net slack of 50 says that the net window of every job is already six times its duration. Furthermore, Figure 3.1d shows results of the same experiment where we kept the number of dependencies moderate, namely at 200. The gap again increases with the size of the net slack, however even for a net slack of 100, the gap never gets larger than 5%. Thus, a large number of dependencies combined with a lot of slack is what drives complexity here.

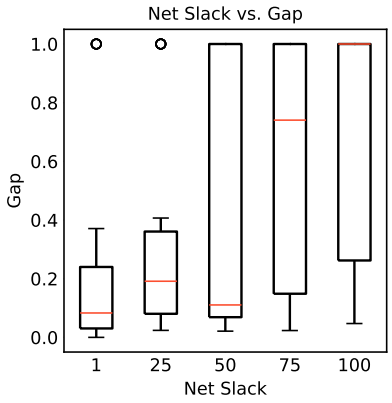
Regarding the effect of dependencies with drain, Figure 3.1e shows the gap for different numbers of dependencies with associated drains. As you can see from Table 3.3, we kept the number of dependencies constant at 1000 and vary the fraction of dependencies with



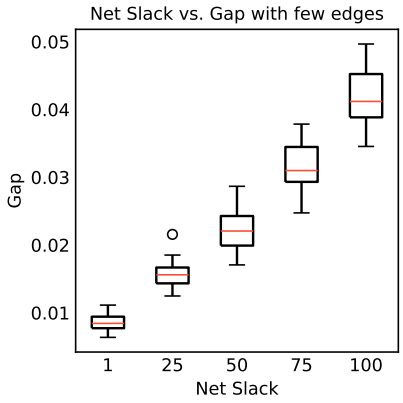
(a) Effect of varying the number of dependencies (Set B)



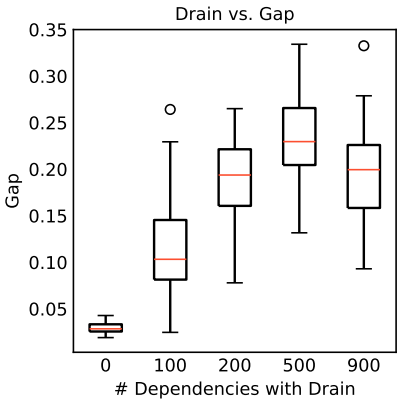
(b) Effect of varying the number of jobs (Set A)



(c) Effect of varying the amount of slack (Set D)



(d) Effect of varying the amount of slack with few dependencies (Set E)



(e) Effect of varying the number of dependencies with drain (Set C)

Figure 3.1 The effect of varying different parameters in instance generation. Red lines indicate the median of all runs. The box indicates upper and lower quartile, i.e., 75% of all results lie below the upper end of the box, and 75% of all results lie above the lower end of the box. Whiskers show the extend of the remaining results, with outliers being shown as circles.

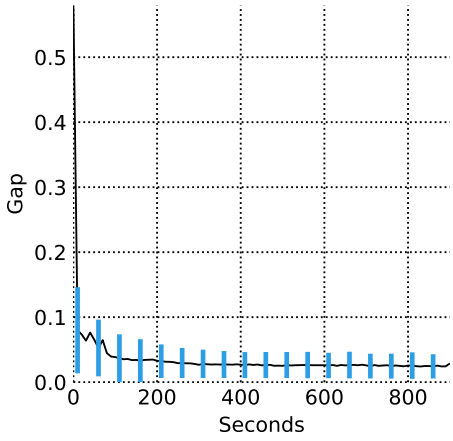


Figure 3.2 Convergence speed of the MILP solutions. The black line indicates the median over all runs. The blue bars indicate the area in which 75% of all runs fall.

drain. We can see that drain significantly raises the complexity of the model, even if just 10% of the dependencies are associated with drain. However, the complexity does not strictly increase with the number of dependencies with associated drain: If too many dependencies are incentivised to have jobs placed closely to each other, the flexibility in the model decreases and results improve slightly, as can be seen.

We finally take a look at the speed with which the MILP solver converges to the optimum solution in Figure 3.2. The black line shows the median of the achieved gap over all MILP runs at different points in time. The blue bars indicate the upper respectively lower quartiles. We can see that within the first 200 seconds, the MILP gap drops to below 10% on average. After 200 seconds further improvement is relatively slow.

Direct comparison with Petersen et al. (2014). Petersen et al. (2014) also give a MILP formulation of a problem, which is a subset of the problem our framework can solve. They state that their MILP, executed on a standard laptop, was able to solve five out of twenty generated instances before hitting memory limits, and for the five solved instances, average execution time was eight minutes. We try to generate twenty instances based on the same parameters as they did, i. e., ten instances each corresponding to their *Portfolio*$(25{,}100)$ and *Portfolio*$(50{,}100)$ settings. Petersen et al. (2014) define a *Portfolio*(N, K) "*as a randomly generated portfolio of* N *local units with* $K_{\text{Run}} \in \{2{,}3{,}4{,}5\}, \overline{P} \in \{1{,}2{,}3{,}4\}$*, and* $K_{\text{End}} \in \{1{,}2{,}\ldots{,}K\}$". A local unit is in their definition a flexible consumer, corresponding to a job in our formulation. Unfortunately, the authors do not state how P_{Dispatch}, described by $P_G(t)$ in our formulation, is selected. For the given portfolio settings, the average power consumption over the (expected) optimisation period is 4.4 respectively 8.8, thus we select $P_{\text{Dispatch}} = 5$ and $P_{\text{Dispatch}} = 9$. This should result in relatively difficult instances since, in an optimum solution, jobs must be distributed as uniformly as possible.

We solve these instances using our model on a standard laptop with 12 GB of RAM and a quadcore CPU running at 2.4 GHz. Gurobi is able to solve all instances to optimality within less than a second and a peak memory usage of less than 35 MB. The fast computation suggests that our framework indeed results in fairly tractable MILP models.

3.5 Discussion

We present a comprehensive modelling framework for demand-side flexibility incorporating most of the characteristics from Table 3.1. In the implementation, we currently do not include the features multiple runs, down-/uptime and production. Without a specific production target, scheduling all jobs exactly once seems most fitting. This results in a fixed output for all possible schedules. However, we will extend the modelling framework and include the remaining flexibility constraints. Simultaneously, we plan to evaluate the representation and interdependencies of the individual constraints theoretically and with real-life case studies. In current research, it seems unclear what realistic test instances that cover a lot of possible real-life scenarios, look like. This is a topic for further research on its own.

Considering our goal to encompass as many applications as possible with our framework, it is questionable whether the cost function represented by the objective function in Equation (3.1) is linear in real applications, as we assumed so far. Logically, (marginal) production costs for the energy to be used would be increasing rapidly with small quantities and levelling out the larger the volume. However, a realistic cost function has to be found for every case study according to the real (marginal) production costs of the case study. Thus, we use the linear cost function as a substitute and emphasise that it has to be adapted to specific use cases.

Additionally, further research should investigate the optimal degree of flexibility in production processes. In our model, flexibility has zero marginal costs. However, providing a particular level of flexibility usually incurs a certain amount of costs and resources that need to be considered. As generating and providing energy usually incurs production costs, the unused self-produced energy also needs to be further considered. Therefore, non-utilization should be penalised in the optimisation problem. A solution approach to this is the direct inclusion of energy storage capacities. Energy storage can help out by saving the otherwise unused energy for a certain amount of time. Nevertheless, storage costs will also occur and need to be considered in the optimisation problem. Currently, we are only considering costs that occur for additional consumption of electricity meaning that we minimise the absolute area difference between production and consumption, in the sense of electrical power.

As we have discussed before, not all flexibility aspects are yet included in the numerical model even though we consider them in the mathematical model. However, we expect the remaining characteristics to be of lower computational complexity as those that we have already incorporated. Thus, their influence on the optimality gap and runtime should be smaller than the impacts of the characteristics we already evaluated in Section 3.4. Furthermore, our consideration of complexity is incomplete as we have not gradually

changed complexity but evaluated inherently different scenarios. Further extensions to consider gradual changes is subject to further research.

We also point out that we use Gurobi 6.5's standard configurations to solve the Mixed-Integer Linear Programs resulting from our modelling framework. These standard configurations work adequately for our random instances. However, note that tuning these could lead to improved solutions. This approach might become useful in time-critical real-life implementation scenarios.

3.6 Conclusion

In this chapter, we presented and evaluated a holistic modelling framework which allows the universal representation of demand side flexibility. Thus, the framework addresses a gap in the modelling of flexibility as current research has introduced a variety of models which are suitable for specific problem instances but neglect the characteristics of demand side flexibility for other applications. After an extensive review of existing literature, we aggregate a coherent list of demand-side flexibility features from research. We then create a framework to integrate them into one consistent model. After introducing the modelling framework mathematically, it is evaluated using randomly created problem instances, and the performance is measured. We measure the performance as the occurring optimality gap and show that our model performs well computationally while considering a wide range of features. We focus on the minimization of externally procured electrical power and peak shaving. In future work, we will consider the economic implications of providing and investing in flexibility. Our model advances current research as it can be universally used to describe flexibility for different applications and improves the comparability of optimisation algorithms.

4 Mining Flexibility from Energy Time Series

With the new modelling framework introduced in Chapter 3, we can describe the flexibility of a given process mathematically. In order to do so, however, we have to know the process well. If a non-expert wants to find out more about the processes and their flexibility, the above framework will not help. In the remainder of this chapter, we investigate if and how one can derive flexibility information from historical data of processes alone. Two steps are necessary to find this information. First, we have to know which processes are running and second, we have to learn about their flexibility. The present chapter, aims to find a solution for the first question and find processes in time series data unsupervised. The next chapter uses the found process information to derive flexibility information from it.

As introduced in the introduction, a process is a series of actions or steps taken by one or several machines in order to achieve something. For example, one process for a machine could be heating up water. If the machine at hand is a kettle, this is most likely its only process. However, if the machine at hand is a coffee machine, heating water is just one process. Although this is an arbitrary example, it already illustrates some key features of the problem. We have a time series recording energy data from one or several machines. Each of these machines could run one or several processes. Assuming we have no more information than the time series data, we do not know how many processes to detect. Our key indicator for something being a process is thus repetition. More formally speaking, we thus search for sequences of a time series, or subsequences in a set of sequences, that occur repeatedly.

Parts of this chapter are reproduced from

N. Ludwig, L. Barth, D. Wagner, and V. Hagenmeyer (2019b). "Industrial Demand-Side Flexibility: A Benchmark Data Set". In: *Proceedings of the Ninth International Conference on Future Energy Systems - e-Energy '19*. The Association for Computing Machinery, pp. 460–473. DOI: 10.1145/3307772.3331021

N. Ludwig, S. Waczowicz, R. Mikut, and V. Hagenmeyer (2017). "Mining Flexibility Patterns in Energy Time Series from Industrial Processes". In: *Proceedings 27. Workshop Computational Intelligence*. KIT Scientific Publishing, pp. 13–32. DOI: 10.5445/KSP/1000074341

A similar question arises in other fields also dealing with sequences. In DNA sequence analysis (and also in music and art), subsequences that occur repeatedly in a set of sequences are often called *(sequence) motifs*. In other areas, such as statistics, similar data points would usually be referred to as cluster. In the remainder of this thesis, we will continue using the term motif and the corresponding task *motif discovery* when highlighting that we are looking for sequences, not individual data points. Otherwise the notion, cluster and the corresponding task *clustering* are used as well. Both motifs and cluster could be seen as pattern and we are thus generally operating in the realm of *pattern recognition*.

In the present thesis, we use a variation of the motif discovery algorithm of Chiu et al. (2003), which we introduce in the following section. The advantage of motif discovery in contrast to other time series clustering approaches is that it uses a Symbolic Aggregated Approximation (SAX) of the time series. This approximation allows for a more fuzzy matching of the compared sequences which is essential for us as we are not looking for exact pattern matches but rather patterns which are similar yet not the same. Additionally, motif discovery uses random projection (Buhler and Tompa 2002) making it a lot faster than other clustering approaches; however with similar pattern detection results (Ludwig et al. 2018b).

Motif discovery has been used before to understand energy time series, e. g. by Simmhan and Noor (2013). The authors use motif discovery to reduce the dimensionality of large time series data sets and reduce the prediction errors when forecasting energy consumption. More prominently, motif discovery is used to identify individual appliances in energy consumption data. Some examples are given in Farinaccio and Zmeureanu (1999), in which the ability is shown to disaggregate the energy consumption in a household from a single-point of entry. However, their approach needs a training phase of one week to find the characteristics in the household data. The authors of Froehlich et al. (2011) give an overview over more methods for disaggregation of end-users smart meter data which is also known under the term non-intrusive load monitoring. However, all the methods mentioned work in a supervised manner and thus need labelling of the data.

Clustering and motif discovery algorithms perform well, when comparing time series sequences of equal and known length, i. e. sequences with the same number of time steps. However, the performance drops when the sequences are of unequal length and cannot be aligned easily. When interested in finding processes in energy consumption time series, the sequences of interest might not always be of equal length. Consider the kettle example from above. If we fill the kettle with a cup of water it will take less time to bring it to a boiling point than the same kettle bringing several cups of water to a boil. The underlying process is however the same. In some scenarios, we might not be interested in the fact that the kettle is undergoing the same process just with different amounts of water. In our application of flexibility detection and modelling however, we are interested in this. We could classify the different amounts of water in the kettle as different operating modes (see the modelling framework for details on modes) and thus potential sources of flexibility.

In this chapter, we thus want to describe a motif discovery algorithm that detects processes in energy time series, even when occurrences of the same process are of different length.

Additionally, the assumption remains that there is no further information available on the processes. Hence, any algorithms relying heavily on knowledge of how many processes are running or how long they are (e. g. most fixed window clustering approaches such as presented in Waczowicz et al. (2015) and Chicco (2012)), are not practical. While there exist algorithms to find different motifs of different or unknown length such as Linardi et al. (2018). They do not help us, they do not help us in the chosen setting, in which the definition of motif is more relaxed in the sense that one motif can be described by unequal length occurrences. In a second step, we want to determine the flexibility from our found motifs with the features from the previous section in mind. The remainder of this chapter is structured as follows. Section 4.1 introduces the motif discovery algorithm, before we discuss the derived flexibility measures in Section 4.2. We then evaluate the approach on an industrial energy time series data set in Chapter 4.

4.1 Energy Time Series Motif Discovery

In the following, we introduce Energy Times Series Motif Discovery using Symbolic Aggregated Approximation (eSAX), based upon the motif discovery algorithms by Lin et al. (2003), Chiu et al. (2003), and Buhler and Tompa (2002), and adapted to cope with the specific requirements of energy time series and flexibility mining impose on the search for motifs. Initial research on motif discovery focuses on DNA sequences, thus the very neat and efficient algorithms available work for symbolic representations of sequences. To leverage the advantages of these algorithms we follow the example of Lin et al. (2003) and Chiu et al. (2003) and transform the real-valued raw time series into sequences of symbols. The whole process is summarised in Figure 4.1 and we describe the individual steps in the following.

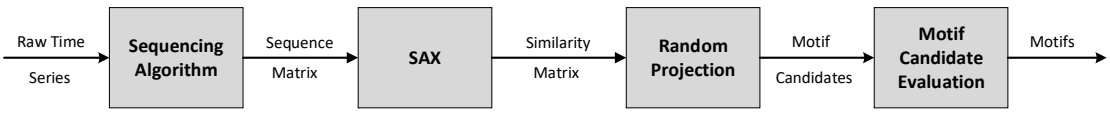


Figure 4.1 The eSAX process from a raw input time series to detected motifs via SAX transformation and random projection.

4.1.1 Sequencing Algorithms

The transformation of time series to sequences of symbols comes with several advantages, the most important one being the possibility of introducing approximations and thus reducing the noise in the raw time series. However, the sequencing of the time series poses some challenges. If we know the length of our motif, or if we can determine a range of lengths for the motif, then we can use a fixed window and move it over the time series, creating sequences of equal length that could also overlap (as done by Chiu et al. (2003)). In the case of industrial machines, however, the active phases are rarely of the same length

(see e. g., Figure 4.2). A fixed window approach would in this case too often cut the sequences at strange time points, making any comparison between the sequences useless.

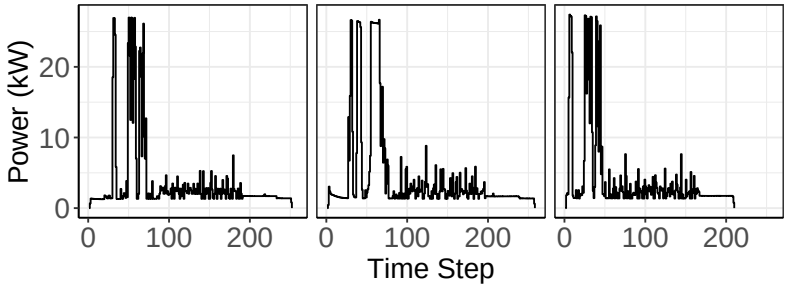


Figure 4.2 Active state sequence from machine level data with differing length.

We thus want to describe two additional simple sequencing algorithms, in order to determine the sequences of interest and give an example with energy time series. The two approaches are a sequencing based on *extreme points* and *threshold values*, while we also provide an example for *fixed windows*. For the extreme point sequencing, we search for minima or maxima within the time series by differencing the time series once and storing all zero values in a list. If necessary, one can remove all points that are less than x time steps away from each other. Secondly, we can use threshold values as cut-off points for the sequences, a typical choice for such a threshold value in energy time series is zero (either the machine is on and the energy intake is above zero, or the machine is off and the energy intake is zero). If we have any information on the time series, we might be able to specify a window size for the sequences. As mentioned before, this is especially useful, when the sequences are expected to be of equal length. In this case the time series is split as soon as the window size is reached.

Figure 4.3 shows three examples where each time a different sequencing algorithm seems to be appropriate. The left panel of the figure shows a time series of a large industrial facility. In this facility, some base processes are always running such that the time series is never zero. Additionally, although we can see a repeating pattern, it is not of equal length making any window size estimations difficult. The time series is thus sequenced by its minima, which are highlighted by red vertical lines. The middle panel shows a time series recorded on a single machine. This machine is turned on and off irregularly throughout the week. In this case, nothing happens when the machine is inactive and thus to find the sequences where the machine is active can be done with defining a threshold value. The sequence splitting points are highlighted again with red vertical lines. Lastly, the right panel shows a time series recorded on a substation connected to residential households. Again, we can see a clear pattern. This time however, the pattern is repeating every 24 hours, i. e. one day, and we thus use a fixed window size to sequence the time series. The splitting points are again highlighted with red vertical lines.

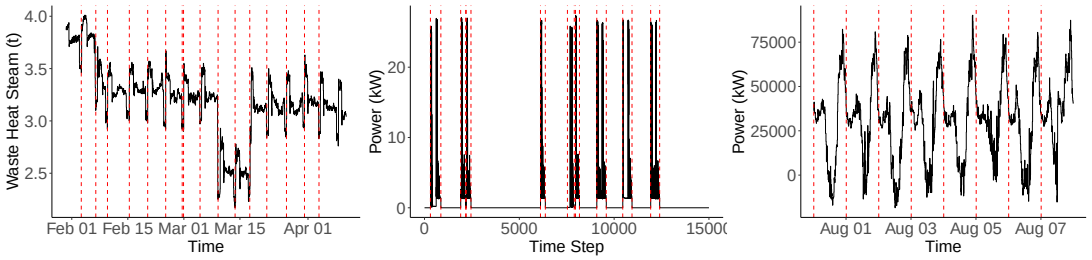


Figure 4.3 Three examples of sequencing algorithms. For each time series the sequencing points found are indicated with the red dashed lines.

4.1.2 PAA and SAX

Having found the sequences of interest, we can move on to transforming the time series to symbols. The first step to get to a symbolic representation is *piecewise aggregated approximation* (PAA). In the case of PAA, we divide the time series $\mathbf{y}$ into pieces of a predefined length p. All values which are part of one piece get aggregated, e. g. by using the mean of all values

$$\bar{y}_k = \frac{p}{m} \sum_{i=\frac{m}{p}(k-1)+1}^{\frac{m}{p}k} y_i \quad \forall k \in \left\{1, \ldots \left(\frac{m}{p}\right)\right\} \text{ and } y_k \in \mathbf{y}.$$

The whole time series is thus approximated by the mean of different short pieces of the time series. If the length of the pieces is one, we have the original time series. If the length of the pieces is greater than one, then the aggregation of the time series gets more coarse with increasing piece length. If $p = m$, the time series is described by its mean value. Having used PAA on a given time series we are still working with real-valued data points, although usually far fewer data points than before.

SAX then goes one step further than PAA. Instead of stopping at the aggregated value, we assign a letter to each piece. This letter comes from a predefined alphabet of length a and a list of breakpoints β determines which piece is assigned which letter. The list of breakpoints $B = \beta_1, \beta_2, \ldots, \beta_{a-1}$ such that $\beta_{i-1} < \beta_i$ and $\beta_0 = -\infty, \beta_a = \infty$ divides the area under $N(0,1)$ into a equal areas. By assigning a corresponding alphabet symbol α_j to each interval $[\beta_{j-1},\beta_j)$, the conversion of the vector of PAA coefficients into a string vector is implemented as follows

$$y_i^{SAX} = \alpha_j, \text{ iif, } y_i^{PAA} \in [\beta_{j-1}, \beta_j)$$

The β parameters are in the original algorithm based on the quantiles of a standard normal distribution with $\mathbb{N}(0,1)$. However, many times series, energy time series included, do not follow a normal distribution. Thus distributing the letters according to a normal distribution without this being the data distribution seem rather arbitrary. While we could

use standardisation techniques to transform the time series into a range appropriate for the normal distribution, a more data-driven approach to the alphabet distribution seems preferable. We thus use the Empirical Cumulative Distribution Function (ECDF) to distribute the letters. The ECDF is defined as follows. For observations $x = (x_1, x_2, \ldots, x_n)$, F_n is the fraction of observations less or equal to p, i. e.,

$$F_n(p) = \frac{1}{n} \sum_{i=1}^{n} \mathrm{I}(x_i \leqslant p). \tag{4.1}$$

The β parameters are then chosen as quantiles of this ECDF. Following the above approximation, we now have a string representing the initial time series, as illustrated in Figure 4.4. This string representation for each subsequence of the time series is also referred to as a word. Where each word describes one subsequence and several letters form a word. The original time series can thus also be seen as a set of words after the SAX transformation.

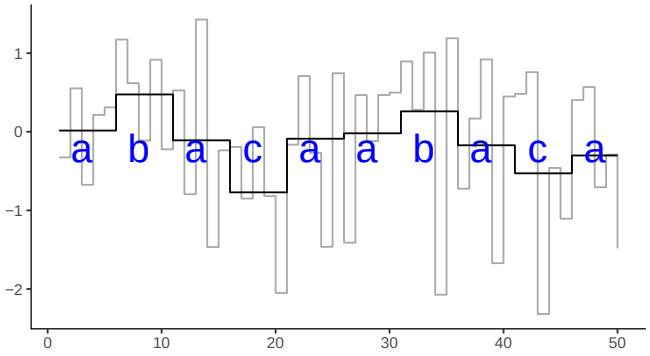


Figure 4.4 PAA (black line) and SAX (letters) approximation of a sample time series (grey).

4.1.3 Candidate Motifs

Given the symbolic aggregation of the time series and the sequence splitting points, motif discovery algorithms from DNA sequence analysis can be used to find candidate motifs. While this could be done with brute force, i. e. comparing each subsequence to all other subsequences, Chiu et al. (2003) introduce a probabilistic algorithm based on *random projection* by Buhler and Tompa (2002). Random projection is fast and robust to small changes in the motifs as not the whole sequences are compared step-by-step.

For the random projection, the SAX representation of all sequences (i. e. the words) are saved row-wise in a *similarity matrix* S^*. In every iteration of the random projection algorithm, we randomly select l of the w columns of S^*, where l is a user-defined mask length, w is the word length and $l \leqslant w$. The word built with l columns is compared to all $(n - m + 1)$ rows of S^*. If there exists a match between the letters in the mask, the corresponding entry in the so-called *collision matrix* is incremented. The entries with the highest values in the collision matrix are considered candidate motifs. These motifs are

transformed back to their piece values and then iterated over the original time series to calculate the distance to the raw sequences and thus find the motifs occurrences.

We will use two different distance measures to find occurrences of the candidate motifs, depending on the time series at hand. The first distance measure is dynamic time warping (DTW) which finds the optimal alignment given two time-dependent sequences. Hence, the sequences y_i and y_j of length n and m, DTW will align the two sequences with the help of a $n \times m$ matrix. In this matrix, every element contains the distance between two points from the sequence y_i and y_j, i. e. $d(y_i(a), y_j(b)) = (y_i(a) - y_j(b))^2$. This matrix is then used to find an optimal warping path through the distance matrix, where we want to find the path that minimizes the warping cost

$$\mathrm{DTW}(y_i,y_j) = \min \left\{ \sqrt{\sum_{k=1}^{K} w_k} \right\},$$

with w_k defined as the k-th element of the warping path w. For further explanation of dynamic time warping see e. g. Müller (2007). Alternatively to the DTW distance, we also use a simple Euclidean distance, defined as

$$d(y_i, y_j) = \sqrt{\sum_t (y_i(t) - y_j(t))^2},$$

between two sequences y_i and y_j. While the first distance is especially useful with unequal length sequences and sequences which could be shifted slightly in time, the latter is more appropriate when an alignment of the sequences is trivial.

4.1.4 Parameter Choices

While we do not need to specify a sequence length or the number of motifs we want to find, there are other important parameters which we have to choose in advance. For the approximation process, three parameters are of importance, namely the piece size p, the word length w, and the alphabet size a. As we might work with unequal length time series which are difficult to compare, we use a trick at the piecewise approximation stage. For the random projection and following candidate motif search to work efficiently, we need a similarity matrix with words of equal length w. Thus, to compare the sequences easily we approximate each one in such a way that the length of each SAX word from one machine time series is the same size w. The size of each piece in a sequence thus depends on the length of the sequence n and the predefined word size w. The word size is in our case specified as the median length of all sequences from a time series. We leave the interesting aspect of further investigation of other word size definitions for future work. Hence, we reduce the parameter space, as neither p nor w, have to be specified explicitly.

With the symbolic approximation of all time series sequences, we can apply the random projection algorithm. We determine the number of iterations based on the length of the

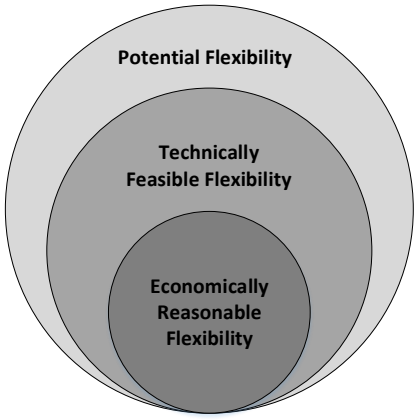
sequences, defining the number of iterations as 10% of the maximum length of sequences from one time series. The parameters which determine when a new motif candidate is found are left at their default values from the TSMining implementation in R (Fan 2015). The motif candidates found are then evaluated using the dynamic time warping distance.

The motifs and their occurrences are then real-valued time series sequences of different sizes. Each time series has at least one motif with at least two occurrences attributed to this motif.

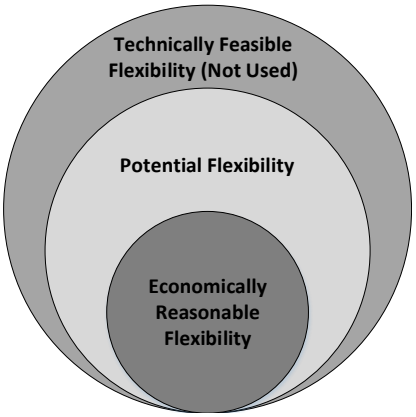
4.2 Flexibility Measures

Ideally, an expert interested in increasing the flexibility of her system would be able to describe every process with the features introduced above. Different scheduling algorithms can then be used to find a peak shaving schedule or one that best fits the renewable energy supply curve. However, such an expert is often not present or has other priorities before the value of more flexibility might be determined. In that case, detecting the features without expert input is of importance.

Overall, we call the flexibility that can be found from motif instances *potential flexibility*. The restriction to potential flexibility instead of true flexibility stems from the fact that we cannot assess which outside influences have played a role in the decision making processes that determined the past schedules. Additionally, we can only observe flexibility that has already been used, anything that is technically feasible but has not been done so far will not be observable in the data. Figure 4.5 shows how potential flexibility can be differentiated from technically and economically feasible flexibility. Whether the data-driven flexibility approach underestimates or overestimates the true flexibility largely depends on the operation of the system under investigation while the time series were recorded.



(a) Potential flexibility assuming all flexibility that is technically feasible has been used in the past.



(b) Potential flexibility assuming not all flexibility that is technically feasible has been used in the past.

Figure 4.5 Potential flexibility in comparison to what is economically reasonable and technically feasible.

We dedicate this section to determining what flexibility information can be detected from the energy demand data alone. Given processes found with eSAX, we classify their flexibility. In order to classify flexibility based on motifs, we look at the features introduced in Chapter 3 and introduce flexibility measures; two measures for time flexibility and three measures for operational flexibility. The time flexibility measures describe the flexibility features *interdependent/ interruptible jobs, earliest start time* and *deadline*, while the operational flexibility measures are used to determine the features *modes*, *base loads*, *down-/uptime* and *ramping*.

4.2.1 Time Flexibility

Time flexibility considers the potential to move a process in time. Additionally to any interdependencies between processes, the earliest start time and deadline features are restricting this flexibility the most. From a pattern recognition perspective, we can infer these features from the time steps at which the identified patterns occur. We take a look at the vector of start times for each pattern S_i as well as the vector of end times for each pattern D_i. Any variance in the start, σ_{S_i}, and end times, σ_{D_i}, of the patterns found (e. g. the hour of the day in which they start), can thus indicate potential time flexibility of a process.

To describe the interdependencies of the processes we use three Pearson correlation coefficients. One correlation coefficient, ρ_{start}, to describe the dependency among start times of two patterns, i. e. if two processes have to be started together this coefficient should be high. One correlation coefficient, ρ_{deadline}, between the deadlines of two patterns, i. e. if two processes have to finish by the same time this coefficient should be high. And the last correlation coefficient, ρ_{inter}, to describe any dependency between the start time of one process and the end time of another process, i. e. if one process needs the output from the other process this correlation should be strong. The coefficients are calculated using

$$\rho_{\text{start}}(S_i, S_j) = \frac{\text{Cov}(S_i, S_j)}{\sigma_{S_i}\sigma_{S_j}}, \tag{4.2}$$

$$\rho_{\text{deadline}}(D_i, D_j) = \frac{\text{Cov}(D_i, D_j)}{\sigma_{D_i}\sigma_{D_j}}, \tag{4.3}$$

$$\rho_{\text{inter}}(S_i, D_j) = \frac{\text{Cov}(S_i, D_j)}{\sigma_{S_i}\sigma_{D_j}}. \tag{4.4}$$

4.2.2 Operational Flexibility

Operational flexibility is concerned with variations in how a process runs, as opposed to when a process runs. With the defined features from above in mind, we are trying to find the different modes in which a process operates. From a pattern recognition perspective, we are thus interested in slight variations within the same pattern. We quantify these variations by looking at the *length*, *energy intensity* and *ramping* of a motif (and thus process).

The length is described as the difference between the start (σ_i) and the end time step (η_i) of a motif, or the occurrences within one motif,

$$n_i = \sigma_i - \eta_i. \tag{4.5}$$

Any change in length, could be due to the process being able to operate in different modes.

The energy intensity is estimated by the area under the curve (AUC) of each found pattern. As we do not have a continuous function to build an integral, we use the composite trapezoidal rule to approximate the area. Given an interval $[a,b]$, we split this interval in M subintervals $[x_k, x_{k+1}]$ of equal width $h = (b-a)/M$ by using the space nodes $x_k = a + kh$, $\forall k = 0,1,\ldots,M$. The composite trapezoidal rule for M subintervals can then be expressed as

$$\int_a^b f(x)dx \approx T(f,h) = \frac{h}{2}\sum_{k=1}^{M}(f(x_{k-1}) + f(x_k)). \tag{4.6}$$

Variations in the energy intake can describe the features for modes, with the minimum length of occurrences within one motif and minimum energy intensity determining the base load.

The ramping of a process can be determined by estimating the difference between the energy at the starting point and the maximum energy of the pattern. We denote this ramping by $\Delta(t)$ and describe it with

$$\Delta(t) := \inf(M) - s_i, \tag{4.7}$$

where

$$M := \{t | P'_t \leqslant 0\} \qquad \text{and} \qquad P'_t = P_t - P_{t-1}. \tag{4.8}$$

Evaluating all these measures, we can sufficiently describe most of the flexibility features from Section 3.2. However, the drain and losses, production, multiple resources and storage features would need external information and can as far as we know not be determined from the time series sequences alone.

4.3 Flexibility Detection for Industrial Energy Data

Given the flexibility features and how we can detect them with historical data using pattern recognition, we now want to apply these methods to detect flexibility in industrial processes from real-world data. This real-world data comes from a data set of smart meter measurements in a small scale electronics factory and is called HIPE (Bischof et al. 2018). In the following analysis, we use a subset in machines and superset in time of the originally published HIPE data set. The flexibility analysis is done based on 6

machines: a chip press, a high temperature oven, a screen printer, a soldering oven, a vacuum oven and a washing machine. We only use this subset of machines since for each of the other machines, the data quality for the selected time range (see below) was questionable for various reasons. All of the machines have been equipped with smart meters which record several quantities such as voltages, currents, frequencies etc. at a frequency of 50Hz. Out of a large number of measured quantities we only consider the active power. The first active power value we use is the last day of 2016 10 pm, while the last power value is from 31.12.2018 10:59 pm. We thus use two full years of data. We down-sample the data to one minute resolution, where the one minute values are the mean values from the original 50Hz measurements during that minute. Due to some problems during the recording of the measurements, not all machines have data for all minutes in the considered time period. For the machines with a complete set of power values we consider 1,051,260 minutes. For more information on the origin of the data and the machines we refer the interested reader to Bischof et al. (2018).

For the sake of clarity, we first present some terminology. The initial data set is a set of electrical consumption time series from *machines*, where there exists exactly one time series per machine. We assume that a machine can either run a *process* or be idle. The part of the machine time series where a process is running, is denoted a (time series) *sequence*. If several sequences are similar, we assume they describe the same process. In this case, we say they are *occurrences* of the same *motif*. The motifs are used to generate synthetic test data, which are sets of *instances*, each consisting of *jobs*. Thus one job, represents a sequence of a machine which is not idle.

As established, we are looking for sequences in the machines' time series. Determining, whether a machine is running is simple when the machines get turned on and off, one can use zero as a threshold to decide whether the machine is running or not. Some of the machines in our data set, however, are frequently in standby mode. Consequently, their power is above zero even though no process is running. To ease the analysis later and only consider the running process and not their standby times, we distinguish between an *active state* and a *passive state* of those machines. While in the active state, the machine is running a process, in the passive state, the machine is either off or in standby mode. We determine the two states for each machine individually with the help of a k-means clustering algorithm. Setting $k = 2$ leaves us with a cluster for each state. In order to detect the active sequences easily we set all data points in the passive state cluster to zero. We discard all sequences belonging to passive states and approximate the remaining ones using SAX as described in Section 4.1. The motifs found for each of these machines as well as the average energy intake and duration are summarised in Table 4.1. We assume for all machines with various motifs that each motif corresponds to a specific process running on this machine.

We characterise the time flexibility of each machine, as defined above, by the standard deviation of the start and end times (Table 4.2). This standard deviation is relatively high for all machines, indicating that both start and end times are rather flexible. To see whether these start and end times might be due to dependencies among the

Table 4.1 Characteristics of the individual machines and the patterns found in the sequences. Where $\bar{\mathrm{E}}$ is the average energy needed in a sequence per machine, $\bar{n}$ is the average length of a sequence per machine and $\sigma_{\bar{n}}$ is the standard deviation of this length.

Machine	Sequences	Motif	$\bar{\mathrm{E}}(kW)$	$\bar{n}(min)$	$\sigma_{\bar{n}}(min)$
High Temperature Oven	226	7	1.13	74.50	68.54
Screen Printer	173	1	0.32	285.51	231.54
Soldering Oven	206	4	1.89	146.32	151.90
Vacuum Oven	572	7	0.41	12.57	17.98
Washing Machine	66	4	1.95	188.47	183.83
Chip Press	51	2	1.09	433.63	220.49

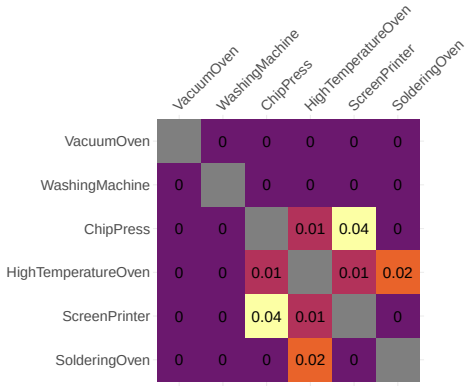
processes, we analyse the correlation coefficients introduced before. All three correlation coefficients are summarised in Figure 4.6.

Table 4.2 Time flexibility of the individual processes per machine indicated by the standard deviation of the start and end times.

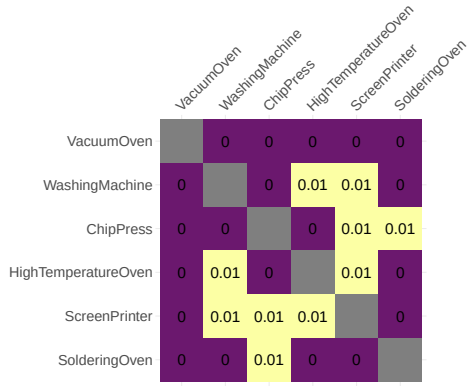
Machine	$\sigma_{\mathrm{start}}(min)$	$\sigma_{\mathrm{end}}(min)$
High Temperature Oven	199.86	198.87
Screen Printer	155.46	139.94
Soldering Oven	183.43	169.82
Vacuum Oven	163.73	164.54
Washing Machine	154.40	157.99
Chip Press	170.83	280.25

The correlation among the start times is relatively low, we can thus assume that none of the machines have to be started together. The same is true for the correlation of the end times. Thereby, the values are all around zero and we can assume that none of the machines have to be coordinated regarding their end time. The picture of dependencies starts to look slightly different, when we consider the correlation between the end and start times of the machines. Interestingly, the vacuum oven has a strong positive correlation with itself, indicating that it is often run several times. The soldering oven also seems to often be started after the vacuum oven. Overall, most of these correlations are not high and we assume that either storing the output is easy or there is only little dependency among the machines.

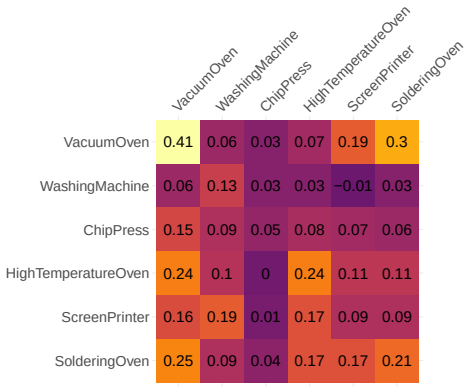
In order to get a better understanding of the differing start times and lengths of the processes, which might restrict the processes flexibility, we look at the dependency of the machine start times, as well as the length of the processes, on the hour of the day. The machine starting times are distributed throughout the day. The earliest start times occur around the start of the working day at 6 a. m., while the latest start times are around 10 p. m.. Most machines are started between 8 a. m. and 6 p. m., with a slight drop in machine operation around noon. These starting times roughly correspond to the working hours and thus indicate that most machines need to be started by hand or at least supervised.



(a) Correlation among start times ρ_{start}.



(b) Correlation among end times ρ_{end}.

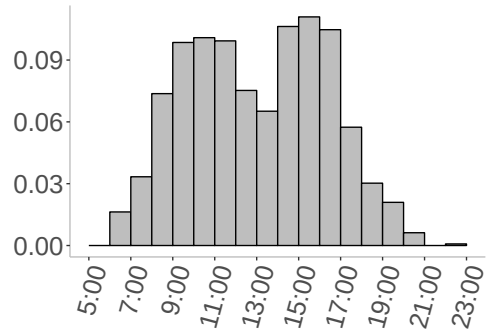


(c) Correlation between end and start times ρ_{inter}.

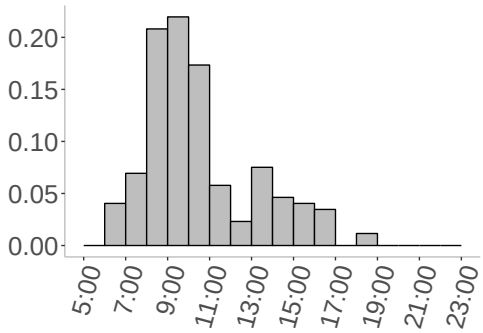
Figure 4.6 Pearson correlation among all processes.

The only machine with a different starting time pattern, is the screen printer. The screen printer is mostly started before noon which might be due to its long running time.

With these start times in mind, it seems that the flexibility is mostly restricted by the operators working hours. We hence seem to operate within the circle shown in Figure 4.5b, where more flexibility is technically feasible. However, under current circumstances it might not be economically feasible to change the working ours, or not socially accepted. The whole systems potential flexibility seem to be high and detectable with our methods.



(a) Hours during which all machines are started.



(b) Hours during which the screen printer is started.

Figure 4.7 Distribution of process start times during each hour of a day for all machines (a) and the most unusual start time distribution found for the screen printer (b).

4.4 Conclusion

The present chapter proposed a motif discovery method to find flexibility potential from industrial energy time series. We have adapted motif discovery in such a way that motifs can be detected without further knowledge about their lengths. Additionally, different occurrences of the same motif can also be found if they differ in length. From the motifs we can efficiently determine the most important flexibility characteristic needed to determine e. g. the required flexibility, as we will see in the next chapter.

The flexibility we find in the present chapter is only *potential flexibility*. As we do not know anything about the process in terms of its technical properties, we only indicate that there might be a potential as there has been some variety in the past. If this variety is only driven by technical features it would not be considered flexibility. Thus, in a next step to properly quantify the flexibility, one would have to talk to the process managers and let them rate the potentials found according to their usability. We might also not want to use the flexibility found, even if technically possible, whenever it is not economically feasible. As we can determine some kind of flexibility, but our data set is rather limited, we want to generate a proper benchmark data set for industrial flexibility in the next chapter.

5 Creating an Industrial Energy Flexibility Benchmark Data Set

Information about the flexibility of individual consumers, especially industrial ones, is not readily available leading to difficulties in testing new strategies and ideas for DSM or making different strategies comparable with each other. To test different algorithms and frameworks, one can use data from or resembling smart meters (e. g. Gottwalt et al. (2011)) or grid data (e. g. Logenthiran et al. (2012)). However, regardless of the type of data, most authors either do not publish the data their analysis is based on (Ashok 2006) or synthesize the whole data set. The synthetic data can range from being entirely made up (e. g. Petersen et al. (2014)), being modelled with specific appliances in mind (e. g. Li and Shi (2012)) or being generated based on data but without using algorithms to extract information from this data (e. g. Yaw et al. (2014)). Benchmark data sets play an essential role in making research comparable and more accessible. For general project scheduling, for example, there exists the PSBLIB benchmark data set from Kolisch and Sprecher (1997), which the related literature uses heavily (e. g. Tormos and Lova (2001)). However, this benchmark data set is not rooted in real-world data. Specifically for resource-constrained project scheduling, Kolisch et al. (1999) have published a data set. Again, the data set is not derived from real-world data. Moreover, no such benchmark data set exists for demand-side flexibility in industrial processes.

Recently, the HIPE data set, a real-world data set with smart meter measurements from industrial machines, has been published (Bischof et al. 2018). This data set contains power demand time series from a set of machines in a small-scale electronics factory. However, the data set consists only of a relatively small amount of machines and there is no readily available information about their flexibility. Hence, in the present chapter, we extract demand information from the real-world data set of industrial machines, generate more process instances based on this information and infer flexibility attributes. More

Parts of this chapter are reproduced from

N. Ludwig, L. Barth, D. Wagner, and V. Hagenmeyer (2019b). "Industrial Demand-Side Flexibility: A Benchmark Data Set". In: *Proceedings of the Ninth International Conference on Future Energy Systems - e-Energy '19*. The Association for Computing Machinery, pp. 460–473. DOI: 10.1145/3307772.3331021

specifically, we use eSAX from the previous chapter Section 4.1 to find regular process patterns in each machine and extract information on when each process starts throughout the day as well as how many different processes can be identified in each machine. Based on this information we generate instances that model real-world scenarios with available demand-side flexibility. To the best of our knowledge, we are the first to create such artificial instances based on pattern recognition through motif discovery. We evaluate the found motifs and show that the data can be used to evaluate performance-critical scheduling algorithms on workloads that resemble real-world scenarios.

The main contribution of the data set is a set of real-world-data based benchmark instances for certain scheduling problems. The problems at hand arise in smart grids when flexible electrical demands can be moved in time to optimise various objectives.

The remainder of this chapter is structured as follows. We describe how we get to non-constant power demands in Section 5.1, before describing how we generate the flexible instances, as well as the exact parameters chosen to generate the benchmark data set (Section 5.2). We then evaluate the behaviour of the processes and their block decomposition in Section 5.3.

5.1 Non-Constant Power Demands

The problems for which we want to generate the data set (i. e. most scheduling problems) assumes power demands to be constant over time. This assumption might be an especially unrealistic and simplifying one. Therefore, we describe in this section how the model from Barth et al. (2018b) can be used to model jobs with fluctuating power demand.

In the scheduling problem, every time-moveable process constitutes one *job*. Let j_i be a job. Then, j_i has a *processing time* p_i, i. e., a time during which it must be executed without interruption. The job also has a *release time* r_i, which is the earliest time during which the job can execute, and a *deadline* d_i, which is the earliest time at which the job must be finished. Finally, every job has a *usage* u_i, which is the (constant) amount of power required by j_i during its execution.

Given a set of n such jobs $J = \{j_1, j_2, \dots j_n\}$, a problem instance also has an edge-weighted directed acyclic graph $G = (J,D,w)$ on J, with edge set $D \subset J \times J$ and weight function $w : D \to \mathbb{Z}$. The edge set D defines *dependencies* between two jobs, while the weight function indicates the necessary *lag* between the two jobs. If $(j_a, j_b) \in D$, then j_b can start at the earliest $w((j_a, j_b))$ time steps after j_a has started.

The power demand function of a job is a stepwise function. To model a stepwise power demand function for a job, we perform a *block decomposition*, which is illustrated in Figure 5.1.

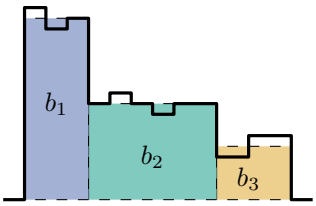


Figure 5.1 Decomposition of a stepwise power demand function into blocks. The fat black line is the original demand function with 11 steps. The three blocks b_1, b_2 and b_3 approximate this function.

Definition 5.1 (Block). For a perfect representation of a stepwise power demand function with k steps, we decompose the respective job into k blocks. From the perspective of the scheduling problem, each of the blocks again is an individual job. However, we will use the term *block* for a job with constant power demand that is part of a decomposition of a job with non-constant power demand.

It is easy to see that the k blocks of a job, when executed consecutively, behave like a single job with the appropriate stepwise power demand function. Thus, we must make sure that the blocks are executed consecutively and without any pauses between them. To this end, we use the dependencies. Let $b_1, b_2, \dots b_k$ be the k blocks that we decomposed a job j_a into. Then, for every $i \in \{1, \dots k-1\}$, we add (b_i, b_{i+1}) to D, with a weight $w((b_i, b_{i+1})) = p_i$. With this, no block can start before its predecessor has finished (but immediately after). Finally, we add (b_k, b_1) to D with $w((b_1, b_k)) = -1 \cdot \sum_{i=1}^{k-1} p_i$. This negative lag forces the last block to start *at the latest* $\sum_{i=1}^{k-1} p_i$ time steps after b_1 started. Combined, these dependencies cause the chain of $b_1, b_2, \dots b_k$ to be executed concurrently.

When modelling a real-world process (with a stepwise power demand function) in this way, the obvious k to choose is the number of steps in the respective power demand function. As mentioned before, this would result in a perfect representation of the original stepwise power demand function. However, one can also choose a smaller number. If one chooses k smaller, the job's power demand function can not reflect the original process' power demand function perfectly, but only approximate it. In Section 5.3, we take a closer look at how well this approximation works with a low value for k.

5.2 Generating Flexible Instances

We use the data and motifs found in Section 4.3 but go one step further and determine block decompositions for each detected occurrence, as necessary for the process to represent jobs with non-constant power demand introduced in Section 5.1. The block decompositions of all occurrences are part of the data set we release and we evaluate the block decomposition in Section 5.3, where we specify a measure for what we consider a good block decomposition of an occurrence.

After detecting motifs, their occurrences and their block decompositions, we generate jobs that form instances for a scheduling problem (see Ludwig et al. (2019b) for more detail on the scheduling). Several parameters influence the generation. First, the *instance size* specifies the number of jobs per instance. Also, a *time horizon* must be given, i. e., the latest deadline of any job, assuming that the earliest release time is 0. For the individual jobs, we first must specify the *block count*, i. e., into how many blocks a job should be decomposed. Also, we must specify how much flexibility we assume to be part of the instance, which is done in terms of a *window growth mean*, a *window growth standard deviation* and a *window base factor*.

Instance generation works by creating a set of *job generators*, one for each motif and each block count k, and then generating jobs from these generators up to the desired instance size. The idea behind the job generators is to fit random distributions to the respective motif's occurrences. When generating a job that should be divided into k blocks, we start with the respective motifs' occurrences that have been decomposed into k blocks as per Section 5.1. For each block, we obtain its length and total energy consumption, for a total of $2k$ values for each of the motif's occurrences. To these values, we fit a $2k$-variate Gaussian Mixture Model (GMM). We chose Gaussian Mixture Models because they are universal density approximators, as shown by e. g. Plataniotis and Hatzinakos (2000), meaning they can approximate any given probability density with arbitrary precision (under the condition that the GMM has enough components). Since we do not want to make any assumptions about the underlying distribution of the duration and energy values, GMMs seem appropriate. Drawing from this distribution results in the shape of a new job: For each of the job's blocks, we get a duration and an energy consumption, from which we determine the power demand.

To determine a release time and a deadline, we first fit a distribution to the start times of the motif's occurrences. However, the start times do empirically not fit a (mixed) normal distribution well. Thus, we instead use a mixture of uniform distributions. To do so, we first cluster the start times using the DBSCAN algorithm (Ester et al. 1996) to account for the assumption that there might be multiple separate time spans throughout a day during which the respective process is usually started. Then, we determine the 0.1 and 0.9 quantile of each determined cluster to account for outliers. These form the lower and upper limit of one uniform distribution each. We weight each uniform distribution by the number of occurrences assigned to the respective cluster. Randomly selecting one such uniform distribution by their weight and then drawing from that distribution yields a preliminary start time s.

However, we need a release time r and and a deadline d (together forming the *window* of the job). We obtain them by determining a window size w and then setting $r = s - (w/2)$ and $d = s + p + (w/2)$ (with p being the processing time). We determine w in two components, the *window base* w_b and the *window growth* w_g. The window base is meant to reflect the flexibility we see in the real-world data. However, the factory we retrieved the real-world data from was so far not managed with demand-side management in mind. We thus assume that more flexibility could be created if operations were changed to facilitate DSM, which is why we add the window growth component.

The window base w_b is determined by the span of the uniform distribution we drew the start time from multiplied by the window base factor. We assume that the more flexible a process is, the larger the spans of its uniform start-time distributions will be. The window growth is determined by drawing from a normal distribution with the specified window growth mean and window growth standard distribution.

To generate an instance with n jobs, we n times perform a weighted selection on the set of job generators. We weight the generators by the number of occurrences in the respective motif. Each time, we generate one job using the selected generator. For each job generated in this way, the release times and deadlines produced by the job generator are based on time-of-day. Thus, we finally move each generated job to a random day within the time horizon. The real-world data we obtained does not contain any satisfactory information about dependencies between processes (cf. Section 4.3), thus we do not incorporate these into the generated instance sets.

Additionally, to evaluate the effect of k, i. e., the number of blocks into which jobs are decomposed, we generate groups of instances that differ only in the value of k. We do this by first generating an instance with $k = 1$. Then, to generate an instance with $k = 2$, we iterate over all jobs in the $k = 1$ instance. For each such job, we generate a $k = 2$ job from the same motif, scale the job so that the total duration and energy consumption is the same as for the $k = 1$ job, and set the same window. We proceed in the same way for all values for k.

The instance generation process requires several parameters to be set. Table 5.1 list the parameters we chose for the instance set we generated. If a table cell contains three values like (a,b,c), that means we chose all values from a to b (inclusive) in increments of c. Between all parameters where we choose more than one value, we take the cartesian product to obtain the final parameter space. The chosen parameter space results in a total of 1764 instances.

We publish the instance set generated as above together with some auxiliary data as a separate data publication (Ludwig et al. 2019a), accessible at https://doi.org/10.5445/IR/1000094324.

Table 5.1 Generated instance sets parameters. Three values (a,b,c) in a cell indicate that a range of parameters was chosen: from minimum a to maximum b (inclusive), with steps of size c.

Parameter Name	Chosen Values
Job Count	$(200, 500, 15)$
Window Growth Mean	$(50, 200, 50)$
Window Growth Std.Dev.	20
Window Base Factor	$(0.05, 0.15, 0.05)$
Time Horizon	5 days
Time Resolution	1 minute
Block Count	$\{1, 2, 3, 4, 5, 7, 10\}$

The data archive itself contains a detailed description of its contents and the file formats. The instance file format is suitable to be use with the TCPSPSuite software package[1], which is what we used for all optimisations performed for the evaluation. The auxiliary data includes a description of the motifs discovered (as described in Section 4.3) as well as for every instance the best solution we compute during our evaluation. These solutions can be used as a baseline for benchmarks.

5.3 Evaluation

In our evaluation, we want to focus on the block decomposition and its influence on the approximation of the original power demand. Since the data has a one-minute time resolution, the power demand curve for every occurrence detected in Section 4.3 is a stepwise function, with one step per minute. However, for many algorithmic approaches, for example scheduling algorithms, the run-time complexity increases significantly if the demand functions have too many steps. Many approaches even can not deal with non-constant demand, i. e., require a demand function with exactly one step.

For this reason, in Section 5.2 we generate jobs with varying, but low numbers of steps in their power demand function. We also call the number of steps in the demand function the number of *blocks* that we decompose a job into. In this section, we analyse the effects that the number of blocks of a job has and evaluate how closely we can approximate the original power demand functions with various numbers of blocks. A detailed look into the scheduling properties of these instances can be found in Ludwig et al. (2019b).

In Section 5.1 we describe a block decomposition that allows to approximate the (stepwise) power demand curve of some original process with a varying number of steps (respectively blocks).

In this section, we analyse how close a given power demand curve with a low number of steps can be to the original curve it tries to approximate. As original curves, we take the occurrences discovered during motif discovery. First, we need a distance measure between two stepwise functions. Given the stepwise demand function of an occurrence o as P_o, and a stepwise demand function with k steps $\tilde{P}_{o,k}$ that approximates P_o, we use the measure

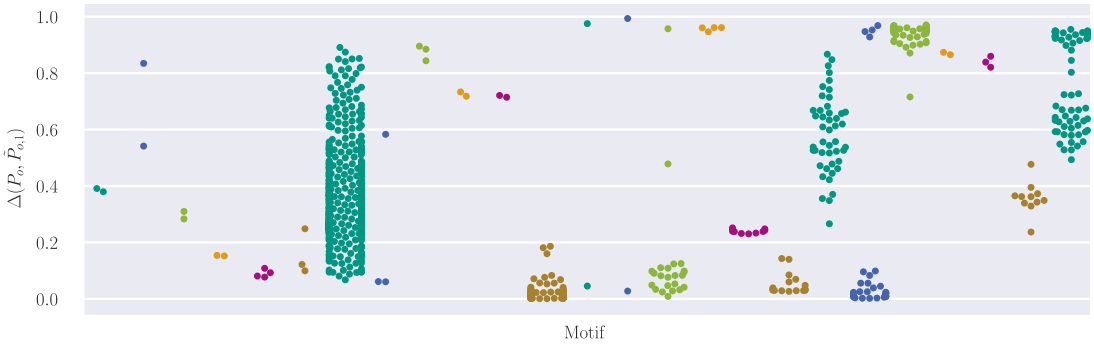
$$\Delta\left(P_o, \tilde{P}_{o,k}\right) = \frac{1}{N_o} \int_0^\infty \left(P_o(t) - \tilde{P}_{o,k}(t)\right)^2 \mathrm{d}t. \tag{5.1}$$

Here, N_o is a normalization factor determined as $N_o = \int_0^\infty P_o(t)\mathrm{d}t$, i.e., the total energy of the original occurrence. Intuitively, the distance between two stepwise functions should correlate with the area between the two curves. However, we assume that — especially for peak shaving applications — a large deviation over a short time is worse than a small deviation over a longer time, which is why we square the difference inside the integral.

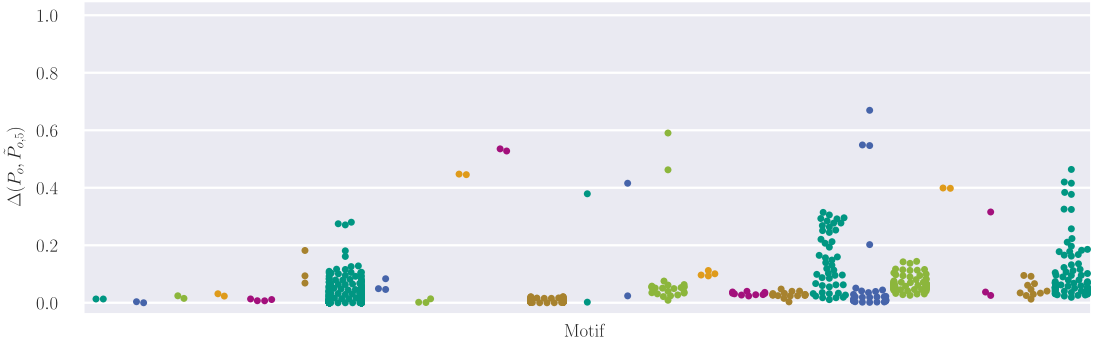
1 https://github.com/kit-algo/TCPSPSuite

We now investigate how well the detected occurrences can be approximated with a low number of steps (resp. blocks) according to this measure. Note that a job is never generated from a single occurrence, and therefore does not approximate any single occurrence's power demand function. Instead, for a given k, we compute for every occurrence of every motif a block decomposition that minimizes (Equation (5.1)). If $\Delta(P_o, \tilde{P}_{o,k})$ becomes small for a given occurrence o and its optimal k-block approximation $\tilde{P}_{o,k}$, that means that occurrence o can be approximated well using only k blocks.

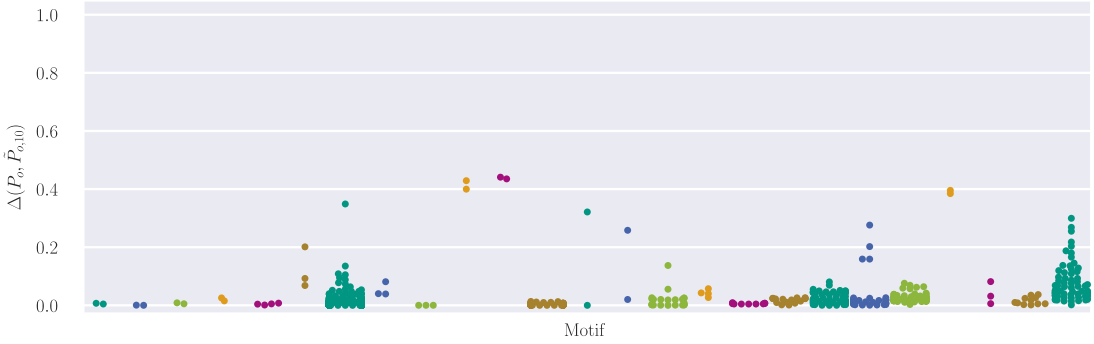
Figures 5.2a, 5.2b and 5.2c show the Δ values for k in $\{1, 5, 10\}$ and every occurrence. In the plots, every dot is one occurrence, which are arranged into columns by their motif.



(a) Δ measures for each occurrence, ordered by motif (on the x axis), for $k = 1$.



(b) Δ measures for each occurrence, ordered by motif (on the x axis), for $k = 5$.



(c) Δ measures for each occurrence, ordered by motif (on the x axis), for $k = 10$.

Figure 5.2 Δ measures for each occurrence for different k (cont.).

We see that for some motifs, the occurrences can be approximated well with only one block. However, for many motifs, one block is not enough for a good approximation. On the other hand, we can also see that the effects of using more than 5 blocks is small.

Figure 5.3 shows a line plot of the change in Δ for changing values of k (on the x axis). Here, for every occurrence o, and every $k \in \{1, \dots 10\}$, we set the y value to $\Delta(P_o, \tilde{P}_{o,k}) - \Delta(P_o, \tilde{P}_{o,20})$, giving an insight into how much one can improve the approximation for that occurrence when changing k from its respective value to 20. Here, we again see that until about $k = 5$, there is a sharp drop in Δ values, with the curve being rather flat afterwards. Thus, we can conclude that for the processes we mined from the time series data, a block decomposition into 5 blocks might be a good compromise between accuracy and instance complexity.

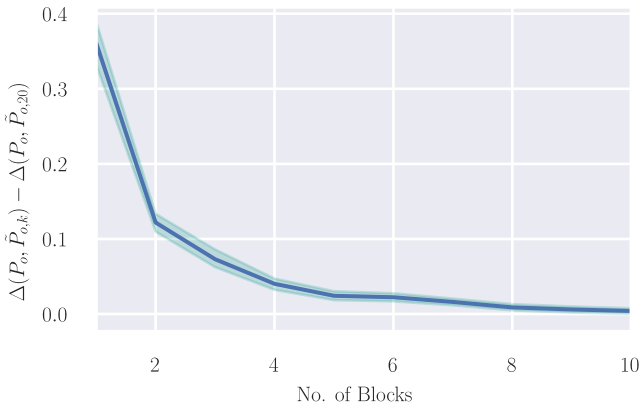


Figure 5.3 $\Delta(P_o, \tilde{P}_{o,k}) - \Delta(P_o, \tilde{P}_{o,20})$ for all occurrences, with k being on the horizontal axis. The solid blue line indicates the mean, and all lines for all occurrences fall within the green shaded area.

5.4 Conclusion

In the present chapter, we presented a new benchmark data set of industrial demand-side flexibility scheduling scenarios. The data set is based on real-world smart meter information from a small industrial facility and has been up-sampled to address large scale problems. The instances in the data set vary in terms of their size, the assumed amount of flexibility and the complexity of their processes' power demand functions, such that the data set covers a wide range of conceivable scheduling problems. We find that there is no straightforward answer to the question how good a schedule with few blocks is compared to a schedule with more blocks per job. One should decide the used block count on a case-by-case basis, based on the need for accuracy weighted against the need for performance. Please note that it is in no way necessary that all jobs are decomposed into the same number of blocks.

Our evaluation has also shown that the block decomposition we perform is to a certain extend suitable to reflect the original power demand curves, thus we may assume that our instances resemble real-world scenarios. Overall, we believe that the benchmark data set can be used to evaluate scheduling techniques dealing with demand-side management. The instances are complex enough to provide a challenge, yet because of the large parameter space diverse enough to point out strengths and weaknesses in the algorithms to be evaluated.

In the future, it would be interesting to enrich the data set with additional constraints from real-world scenarios, such as dependencies between processes, storage constraints, etc.

Summary Part II

In the present part, we have investigated the potential demand-side flexibility in a given energy system from a data-driven perspective, thus answering the [RQ1]: How much demand-side flexibility is potentially available?

We have shown that assessing the potential flexibility is possible, given defined flexibility features and measures to obtain their information from data. Additionally, we have introduced a way to detect flexible processes from energy time series and provide a benchmark data set for further research in this area.

The contributions of this part to the challenges [C1] and [C2] can thus be summarised

- We have developed a modelling framework for demand-side flexibility that can adequately describe any industrial or household process in such a way that it can be scheduled and flexibility can be represented [C1].
- We have introduced eSAX, a motif discovery algorithm based on SAX that can compare time series sequences of any length stemming from any distribution [C2].
- We have applied the modelling framework and eSAX to machine level energy consumption data and shown that it can find similar patterns and one can infer flexibility from these patterns [C2].
- We have generated a benchmark data set using the flexibility information to foster more scheduling related demand-side flexibility research.

Part III

Required Demand-Side Flexibility

Overview Part III

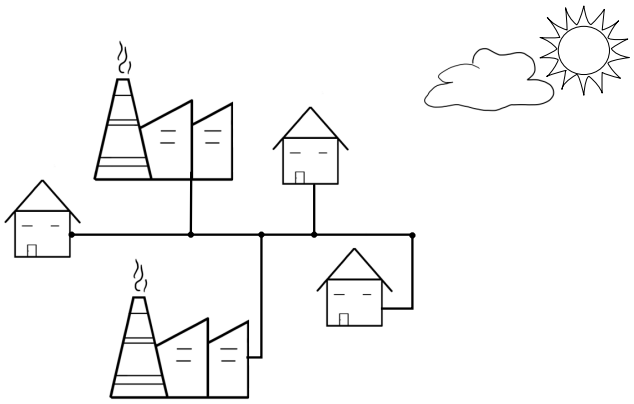


Figure 5.4 Graphical view on Part III of the present thesis: Required Demand-Side Flexibility.

Having established the potential flexibility available on the demand side in Part II, we want to analyse the need for demand-side flexibility in a given system with a high share of renewable energy sources (RES). This need stems from two main factors. On the one hand, we need to know *if* flexibility is needed, thus answering the question what the expected demand looks like and whether that fits the expected supply. If the expected demand fits the supply, there is no need to use any flexibility. If there is a mismatch between the expected demand and supply, then flexibility is needed, or can be used to solve issues arising from this mismatch. On the other hand, we need to know *how much* flexibility is needed, thus answering the question how many e. g. processes need to be moved by how much time to achieve a specific aim, such as peak shaving or energy autonomy.

In the present part, we examine both factors in order to answer [RQ2]. We start with investigating whether we need flexibility at all by evaluating demand forecasts with properly post-processed weather data in Chapter 6, including contributions to challenge [C3]. Afterwards, we assess how much flexibility we need based on a generated benchmark data set and using scheduling in Chapter 7, thus contributing to challenge [C4].

6 Proper Post-Processing of Weather Data in Demand Forecasts

When there is a mismatch of demand and supply, demand-side management (DSM) tools can be used to change the demand behaviour. However, in order to efficiently use any DSM, we need to know if and when there is a mismatch between demand and supply of energy. This mismatch, can be calculated by forecasting it directly, or forecasting demand and supply separately and then estimating the mismatch. As in the previous parts, the focus of the present thesis is on the demand side. We thus focus on electricity demand forecasts. These demand forecasts become increasingly difficult with more renewable energy sources being installed on the demand side (e. g. solar panels). Additionally, demand and renewable energy supply are uncertain and thus probabilistic energy demand forecasting has gained more attention recently.

Probabilistic forecasts aim at quantifying the uncertainty in the processes that are predicted. In the case of demand forecasts, external variables play an essential role and account for part of the inherent uncertainty. The demand, for example, depends to a certain extent on the weather. Unfortunately, the weather itself is a chaotic system where probabilistic forecasts are used to assess the likelihood of individual weather scenarios. To accurately forecast the demand, it is thus essential to propagate the uncertainty from the weather variables through the forecasting model. In the present chapter, we want to generate probabilistic forecasts of the electricity demand at the example of the United Kingdom while accurately accounting for the uncertainty coming from external weather variables.

Weather and energy in general are highly related variables. While this is obvious for power generation from RES such as wind power (Jeon and Taylor 2012) or solar power (Thorey et al. 2018), weather also plays an important role in energy demand forecasting. Several studies, for example Hor et al. (2005), show that weather variables, especially

Parts of this chapter are reproduced from

N. Ludwig, S. Arora, and J. Taylor (2020). "Probabilistic Load Forecasting Using Post-Processed Weather Ensemble Predictions". In: *Journal of the Operational Research Society (submitted)*

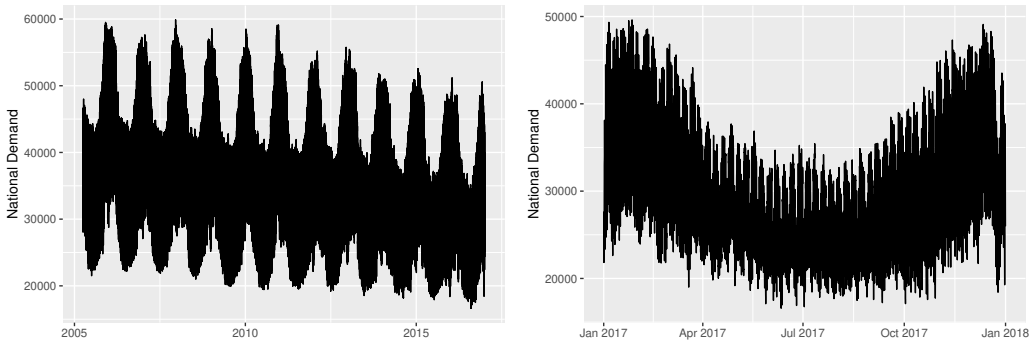
temperature, are correlated with energy demand and thus including weather information improves the forecast accuracy, regardless whether the country under observation is Greece (Mirasgedis et al. 2006), South Korea (Son and Kim 2017) or the UK (Hor et al. 2005). Therefore, forecast weather variables are important and as the weather is a chaotic system itself, probabilistic forecasts are especially advantageous to capture the inherent uncertainty in the realm of energy systems (Haupt et al. 2019).

In recent years, Numerical Weather Prediction (NWP) have become the state-of-the art in meteorology, with more computing power allowing to run complex physical models. These physical models describe the atmospheric processes in a simplified way and are due to their non-linearity and complexity solved with numerical approximations (Al-Yahyai et al. 2010). The outcome of a numerical weather prediction model highly depends on the initial state of the atmosphere. This state has to be assumed and thus several runs of the model with perturbed initial parameters are used to predict weather variables. These runs with perturbed initial conditions result in so called ensemble forecasts. Each ensemble member is then a run of the NWP with different initial parameters. Overall, such ensemble forecasts grasp the uncertainty of the initial state of the atmosphere and are thus the ideal basis for probabilistic weather forecasts.

Unfortunately, ensemble forecasts coming out of NWPs are uncalibrated and known to be under-dispersed. This knowledge resulted in a growing literature on new methods for the statistical post-processing of ensemble forecasts (e. g. Ben Bouallègue et al. (2016), Baran and Lerch (2018), Feldmann et al. (2015), Scheuerer and Büermann (2014), and Möller et al. (2015)). Most commonly used methods for the post-processing include Bayesian Model Averaging and Ensemble Model Output Statistics (EMOS). While the first method was introduced by Raftery et al. (2005), the latter is also known under the term non-homogeneous Gaussian regression (NGR) and was introduced by Gneiting and Raftery (2007). The aim of these methods, is a calibration of the ensembles to past historical data such that the true value lies within the distribution generated by the ensemble forecasts.

When each forecast variable is post-processed individually, we end up with a number of univariate and independent distributions. However, the real weather variables are not independent, and taking spatial and temporal dependencies, as well as dependencies between the variables into account is thus of importance (Schefzik et al. 2013). This problem is well known in the meteorological literature with several papers proposing new methods to create dependent distributions via different empirical copula approaches, for example, the Shaake shuffle (Clark et al. 2004) or different variants of Ensemble Copula Coupling (ECC) (Schefzik et al. 2013; Gneiting 2014).

In the energy forecasting literature, papers using ensemble weather information as input are rare, although e. g. (Al-Yahyai et al. 2010) show that NWPs are superior to station based weather information. Felice et al. (2013) use the operational forecast out of an NWP. The operational forecast is a point forecast that describes the operators best guess for the initial parameters. Even when ensembles are used to forecast variables which are not the weather, e. g. wind power (Heppelmann et al. 2015; Heppelmann et al. 2017), wind ramp events (Bossavy et al. 2013), solar power plant output (Thorey et al. 2018)



(a) Demand throughout the training period from mid 2005 to the end of 2016.

(b) Demand throughout the test period from beginning to end of 2017.

Figure 6.1 Hourly national demand of Great Britain as measured by the British system operator National Grid.

or electricity demand (Taylor and Buizza 2002; Taylor and Buizza 2003), the need for post-processing and keeping the dependency structures among the weather variables is generally neglected. Only Heppelmann et al. (2017) are using post-processing and an approach to capture the dependencies in their probabilistic wind power forecast.

In the present chapter, we include ensemble weather information from the European Centre for Medium-Range Weather Forecasts (ECMWF) to predict the electricity demand. More specifically, we use statistical post-processing including an ensemble copula coupling approach to calibrate the ensembles while keeping their spatio-temporal and multivariate dependencies. To the best of our knowledge, we are the first to include post-processing of the ensembles especially including ensemble copula coupling to get the predictive distribution of the electricity demand.

The remainder of this chapter is structured as follows. We introduce the data in Section 6.1 and describe the post-processing in Section 6.2, before introducing the forecasting models in Section 6.3. We evaluate the post-processing and forecasting on Section 6.4 and lastly, conclude in Section 6.5.

6.1 National Demand and Weather in Great Britain

We test the effect of post-processing and especially accounting for weather dependencies at the example of Great Britain. In the following we thus describe the data used to forecast the national demand of Great Britain under uncertain weather influences, while also explaining our pre-processing steps. An overview of the data sources can be found in Table 6.1.

Table 6.1 Overview of the data used in this chapter and where it can be obtained.

Type	Dates	Variables	Data Set	Data Source	Data Availability
Re-analysis (actuals)	2006-01-01 – 2017-12-31	2m Temperature, 10m u- and v-component of wind, total cloud cover	ERA-5 interim data set	Copernicus data storage	publicly available to non-commercial users
Ensembles 50 members	2015-12-01 – 2017-12-31	2m Temperature, 10m u- and v-component of wind, total cloud cover	Operational Data	ECMWF MARS archive	available to non-commercial users in member states after registration
Demand	2005-05-12 – 2017-12-31	Total demand	–	National Grid	publicly available

6.1.1 Load Data

The load data, obtained from the British system operator National Grid, describes the national demand in half hour intervals (see Figure 6.1). The data is a summary over all flows on the transmission grid in the UK and therefore does not capture all distributed energy sources at place. This form of measuring the demand also results in a very steep decline of national electricity demand over the past years. Overall, we use data from 2006 to 2017, where the first ten years are used to train the models before testing them on the last available year 2017[1].

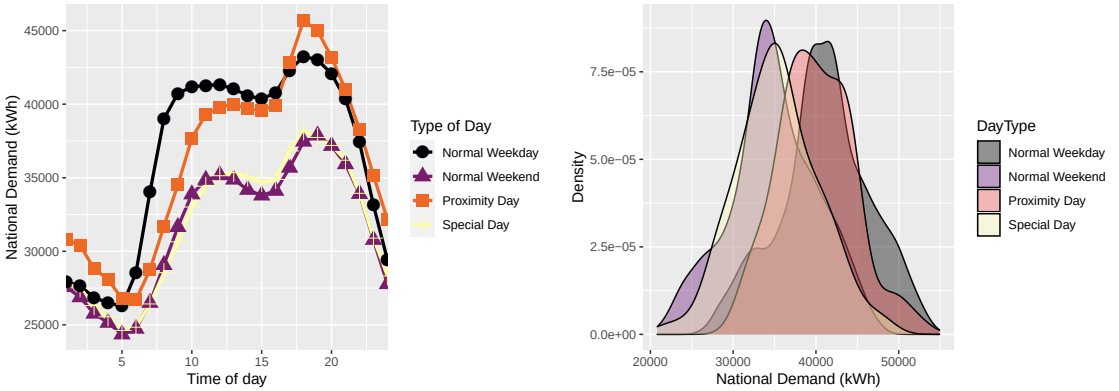


Figure 6.2 The daily UK national demand profile across the different types of day and the corresponding density.

Taking a closer look at the demand data, we see that the electricity demand behaves differently, depending on the type of day. For example, the load on weekends is usually lower than

1 Although load data for 2018 is available, the corresponding weather data is not complete and we thus restrict the analysis to the complete data set we can obtain from both sources.

on working days, and the consumption behaviour is also different on special days such as public holidays or proximity days, such as days around Christmas, as can be seen by the average daily profiles shown in Figure 6.2. These differences in demand are also represented in the correlation with weather variables, especially when comparing weekdays with weekends.

6.1.2 The Influence of Weather on Demand

The weather is known to influence the demand. To adequately capture this influence, we use the original weather variables from the NWP and additionally derive two new variables namely *effective temperature* and *cooling power of wind*. The effective temperature is added as changes in behaviour, due to outside temperatures, are usually slow and gradual. We define the effective temperature (TE) as in Taylor and Buizza (2003) using

$$\mathrm{TE}_t = \frac{1}{2}\mathrm{TO}_t + \frac{1}{2}\mathrm{TE}_{t-1},$$

where TO is an average over the past four observed temperature values. Additionally, the cooling power of wind (CP) variable, highlights the idea that wind speeds (W) change the electricity consumption behaviour only if the outside temperature is below a certain threshold. We again use the definition by Taylor and Buizza (2003), where this threshold is at 18.3 °C

$$\mathrm{CP}_t = \begin{cases} W_t^{\frac{1}{2}} \left(18.3 - \mathrm{TO}_t\right) & \text{if } \mathrm{TO}_t < 18.3\,^{\circ}\mathrm{C} \\ 0 & \text{if } \mathrm{TO}_t \geqslant 18.3\,^{\circ}\mathrm{C}. \end{cases}$$

These two new variables show a high correlation with the demand (see Figure 6.3), which is also higher than the correlation the variables temperature and wind speed have with the demand on their own.

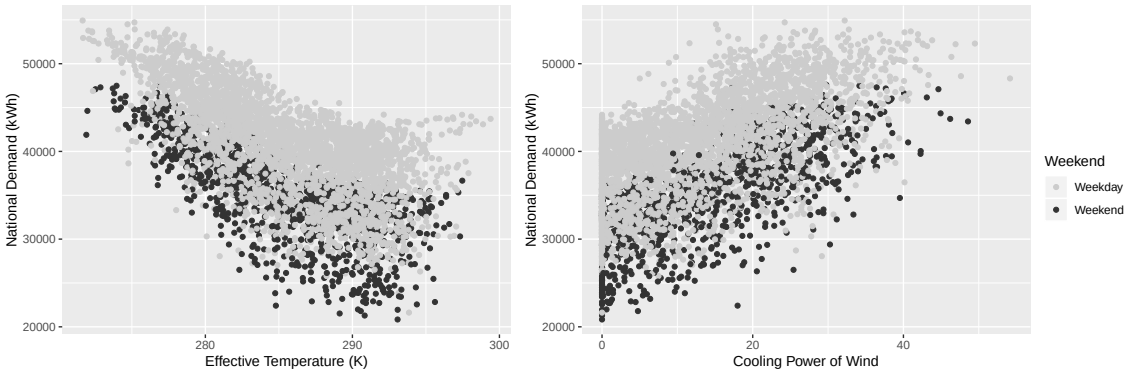


Figure 6.3 Correlation of effective temperature and cooling power of wind with the national demand, where grey dots indicate weekdays and black dots indicate weekends.

6.1.3 Weather Data

In order to capture the uncertainty in the weather, we use forecasts and actual data from ECMWF. The uncertainty in the weather is represented by the ensemble forecasts, while the actual weather is represented by the reanalysis data set. We extract the 12 noon values for each day in the test and training period for 4 different weather variables; temperature at 2m above ground, u- and v- components of wind and total precipitation. The wind speed is calculated from the u- and v-components, $\mathrm{windspeed} = \sqrt{\mathrm{u}^2 + \mathrm{v}^2}$. While the weather data is available in a grid format, we use only those grid points which are close to where energy demand is expected. The grid points used and the weights associated with their influence on the total demand are based on information from National Grid and displayed in Figure 6.4. The ensembles at each grid point are extracted and weighted to construct one set of ensembles at each time point.

Grid Point	Weight
Heathrow	0.28
Bristol	0.18
Birmingham	0.16
Hawarden	0.14
Glasgow	0.1
Leconfield	0.07
Leeming	0.07

Figure 6.4 Map of Great Britain with dots indicating the grid points that were used for the extraction of weather information. The points size indicates their corresponding weight, as also summarised in the table.

6.2 Weather Ensemble Post-Processing

The raw weather ensemble forecasts $x_1, \ldots x_M$ with M members, from ECMWF are uncalibrated and thus often biased and under-dispersed. Figure 6.5 shows the ensembles in our data set, which are biased as well. Biased ensembles could lead to biases in the demand forecasts as well, thus, in this section we describe how to overcome these biases and produce calibrated weather ensembles using EMOS. We then describe how the weather variables' dependency structures can be retained using ECC.

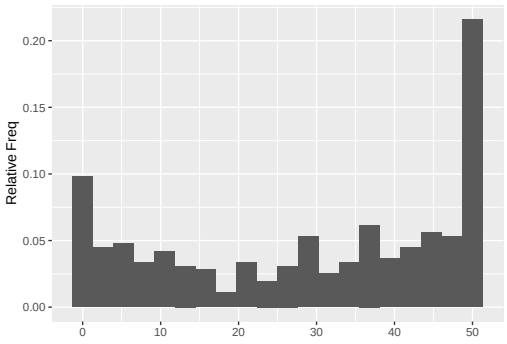


Figure 6.5 Frequency rank histogram of the raw temperature ensembles at 12 noon for a one-step ahead horizon.

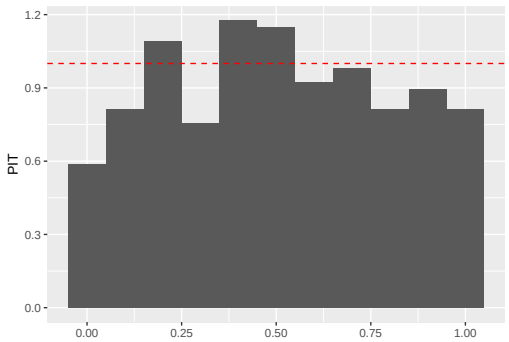


Figure 6.6 The probability integral transform for temperature after post-processing.

6.2.1 Ensemble Model Output Statistics

EMOS seeks to eliminate the bias by calibrating past ensembles to past actual data and using the estimated parameters to calibrate future ensemble members. This calibration is done by estimating a distribution for each weather variable. In our post-processing the temperature ensembles are modelled using a normal distribution $\mathcal{N}(\mu, \sigma)$, where $\mu = a + bm_{\text{ENS}}$ and $\sigma = c + ds^2_{\text{ENS}}$. Total cloud cover is modelled with a censored and shifted gamma distribution $\text{Gamma}_0(\kappa, \theta, q)$, with $\kappa\theta = a + b_1x_1 + \ldots b_mx_m$ and $\kappa\theta^2 = c + dm_{\text{ENS}}$. Lastly, wind speed is modelled using a truncated normal distribution $\mathcal{N}_0(\mu, \sigma)$, where μ and σ are calculated as for the temperature ensembles. For all weather distributions the parameters are estimated using

$$m_{\text{ENS}} = a_0 + a_{\text{ENS}} \sum_{m=1}^{51} x_m$$

and

$$s^2_{\text{ENS}} = \frac{1}{51} \sum_{m=1}^{51} \left(x_m - \frac{1}{51} \sum_{m=1}^{51} x_m \right)^2.$$

In our setting, we estimate the parameters on a training set, which is the whole year 2016, and keep them fixed throughout the test year of 2017. Usually, the EMOS parameters are re-estimated more frequently, however, we found the parameters to be very unstable and a frequent re-estimation to decrease the forecasting performance. The estimated distribution parameters from the new EMOS scheme can be found in Table 6.2. The original distributions are compared to the ones after post-processing in Figure 6.7.

To further evaluate if the EMOS and ECC approaches improve the calibration of the weather variables and demand forecast, we take a look at the *probability integral transform* (PIT). If F denotes a fixed, non-random predictive CDF for an observation Y, the probability integral transform is the random variable $Z_F = F(Y)$. It is known that if F

is continuous and $Y \sim F$ then Z_F is standard uniform. The PIT for the temperature variable is shown in Figure 6.6. Ideally all bins would end at the dashed red line. Compared to the relative frequency, this is more uniform, although it could be improved further. The estimated distribution parameters from the new EMOS scheme can be found in Table 6.2. The original distributions are compared to the ones after post-processing in Figure 6.7.

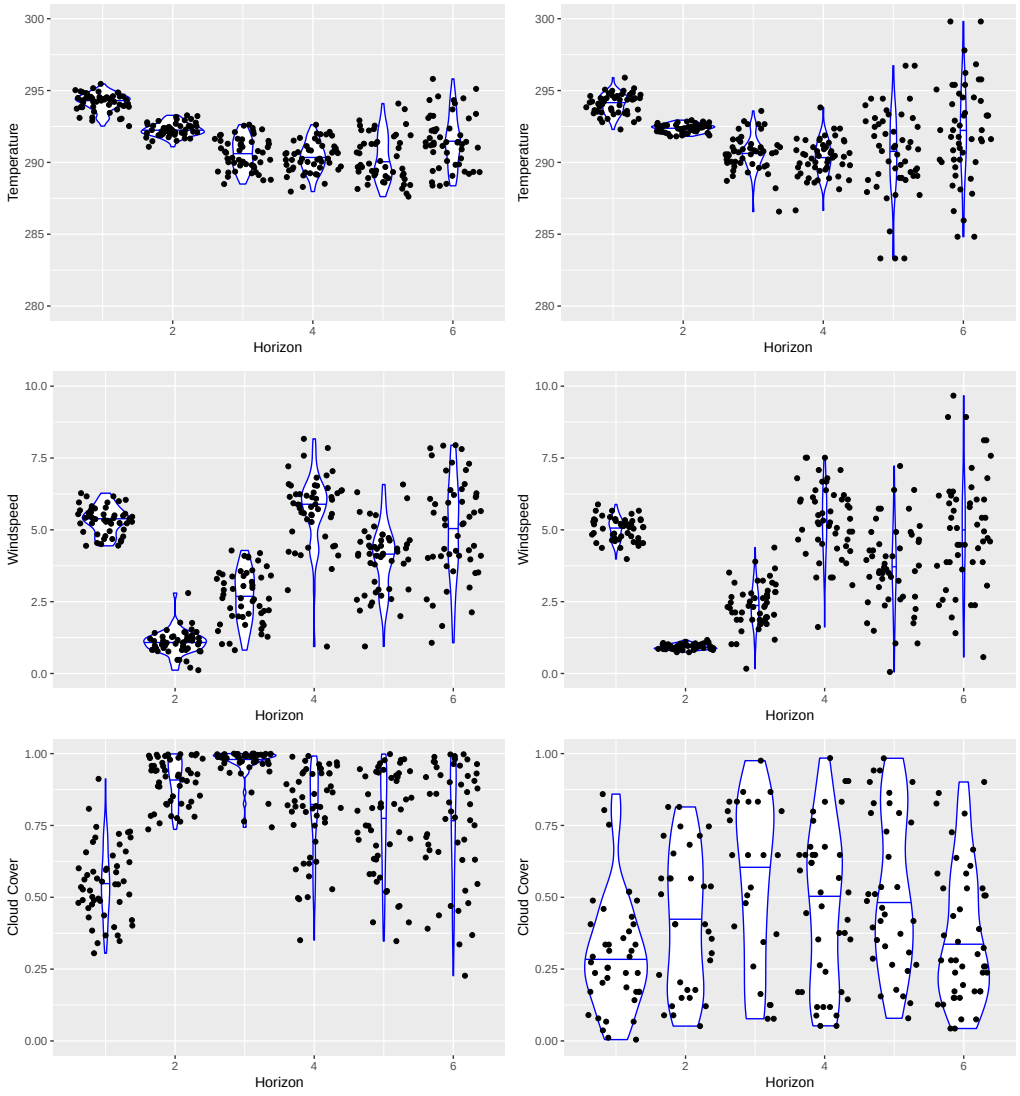


Figure 6.7 The distribution of values for temperature, cloud cover and wind speed, before post-processing (left) and after using EMOS post-processing (right) estimated once for the whole data set at one randomly selected day of 2017.

6.2.2 Ensemble Copula Coupling

After post-processing the ensembles using EMOS, we have ensembles that are calibrated to the past data. All weather variables are modelled with a univariate distribution. We could sample from each of these distributions independently and input the information

into the demand forecasting scheme. However, we could end up with combinations of weather variables that are unlikely to happen in reality as we have lost all information on dependencies among the variables as well as in space and time. The original NWP includes this information and in order to keep the dependency structure in the post-processed ensembles, we use a reordering based ensemble copula coupling scheme similar to the Shaake shuffle introduced by Clark et al. (2004) and the ECC-Q scheme by Schefzik et al. (2013).

A d-dimensional Copula is a multivariate cumulative distribution on the unit cube $[0,1]^d$ with uniform margins (Nelsen 2007). The usefulness of copulas in restoring dependency structures is based on the theorem of Sklar (Sklar 1973). Sklar (1973) state that for any multivariate cumulative distribution function F with margins $F_1, \ldots, F_M$ there exists a copula C, that is unique on the range of the margins and has the form

$$F\left(x_1 \ldots, x_M\right) = C\left(F_1\left(x_1\right), \ldots F_M\left(x_M\right)\right)$$

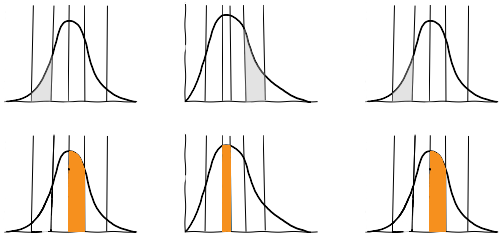
for $x_1, \ldots, x_m \in \mathbb{R}$. With regards to ensemble calibration, we already have the uniform margins $F_1, \ldots, F_M$ in the form of the EMOS univariate distribution. Therefore, Sklar's theorem states that as long as an appropriate copula is defined, univariate ensemble post-processing techniques can be used to accommodate any dependency structure.

The ECC approach is based on the mathematical concepts defined above with the appropriate copula being defined in the form of a *reordering* process. The idea is that given a dependence structure "template" (Schefzik 2017), the samples drawn from the univariate EMOS distributions can be reordered in such a way that they resemble the initial correlation structures that were given in the NWP. The templates are here based on the raw ensembles, where we assume that the raw ensembles capture the correlations sufficiently. While several variants exist, we take a closer look at quantile ECC (ECC-Q).

The procedure essentially includes three steps. In the first step, we rank the ensemble member values. Thus, the raw ensembles $x_1, \ldots, x_M$ with their order statistics $x_{(1)} \leqslant \ldots \leqslant x_{(M)}$ are used to generate a rank dependence structure at each time horizon via a permutation π, with $\pi(m) := \operatorname{rank}(x_m)$ for $m \in \{1, \ldots, M\}$. In a second step, we want to impose this rank structure on the post-processed ensembles. We do this by splitting the post-processed ensemble distribution into M equally spaced quantiles. Each quantile represents a rank in the raw ensemble. We draw from the bins of all weather variables dependently on the rank structure from the raw ensembles such that the post-processed and reordered ensemble $\hat{x}_1, \ldots, \hat{x}_M$ is given by

$$\hat{x}_1 := \tilde{x}_{(\pi(1))}, \ldots, \hat{x}_M := \tilde{x}_{(\pi(M))}.$$

We illustrate an example in Figure 6.8. In this scenario we have six ensemble members which are used to describe three distributions of weather variables at two different time steps.



	x_1	y_1	z_1	x_2	y_2	z_2
em_1	1	1	3	4	6	1
em_2	2	5	2	4	3	4
em_3	5	6	6	1	4	3
em_4	6	7	1	6	5	4
em_5	3	4	5	5	3	5
em_6	4	2	4	3	2	6

Figure 6.8 Symbolic explanation of ensemble copula coupling with the distribution of three variables at two different time steps. Each of the six ensemble members is ranked. The values for each time step are drawn from the coloured bins of each distribution to produce the forecast.

Table 6.2 The EMOS-OLS parameters describing the different distributions for each weather variable and horizon after post-processing.

Variable	Horizon	a	b	c	d
Temperature	1	3.68	0.99	0.24	0.34
Temperature	2	3.46	0.99	-0.27	1.44
Temperature	3	0.91	1.00	-0.46	1.57
Temperature	4	0.64	1.00	-0.76	2.01
Temperature	5	0.82	1.00	-1.48	2.77
Temperature	6	3.18	0.99	-2.82	3.63
Windspeed	1	-0.07	0.95	0.05	0.30
Windspeed	2	-0.11	0.95	-0.13	0.70
Windspeed	3	-0.33	0.99	-0.13	0.77
Windspeed	4	-0.50	1.02	-0.10	0.99
Windspeed	5	-0.35	0.99	-0.13	1.32
Windspeed	6	-0.50	1.01	-1.12	2.07
Cloud Cover	1	0.02	0.98	-0.00	0.67
Cloud Cover	2	0.09	0.89	-0.01	0.69
Cloud Cover	3	0.08	0.90	-0.01	0.57
Cloud Cover	4	0.20	0.76	-0.01	0.52
Cloud Cover	5	0.24	0.72	-0.01	0.50
Cloud Cover	6	0.19	0.77	-0.02	0.54

6.3 Models

We evaluate two different models according to their performance in forecasting the national demand of Great Britain; a linear regression model and a random forest. Each model is trained on the actual weather data during the training period and tested without weather data, on the uncalibrated ensemble data, the post-processed ensemble data and on the actual weather data for the unseen test set.

The training set consist of all available data from the beginning of 2006 up to the end of 2016, while the test data set is the year 2017.[2]

6.3.1 Linear Regression Model

The simplest model, which we use to forecast the national demand, is a linear regression model, which we call LM in the following. The model consists of two stages, where the first stage can be described as (for simplicity the time index t is dropped from the regression formula)

$$y = \beta_0 + \sum_{i=1}^{N} \alpha_i x_i + \sum_{j=1}^{M} \beta_j D_j + \sum_{k=1}^{K} \gamma_j C_k + \epsilon,$$

where y is the dependent variable which in this case is the national electricity demand, x_i are other time series describing the demand, D_j are dummy variables and C_k are interaction terms either between two time series or between a time series and a dummy variable. In our setting the dummy variables include Friday, Saturday, Sunday, and dummies for special days (e. g. public holidays), proximity days, which are days close to a special day (e. g. the day after Christmas), and each public and bank holiday specifically. We also include dummy variables for the summer (June, July, August) and winter period (December, January, February). The time series used to describe the load can be divided into time specific time series and weather time series. For example, the time related time series include a counter of the day in a year, and an overall day counter for the whole data set, as well as quadratic and cubic terms of these counters. Additionally we include interaction terms, such as the interaction between temperature and wind and the interaction between temperature and the weekend dummy.

In the second stage of the model the error from the first stage is modelled independently with the help of autoregressive terms in the form of

$$\epsilon_t = a_0 + a_1 \epsilon_{t-1} + \ldots + a_l \epsilon_{t-28} + \eta_t, \quad \eta_t \sim \mathcal{N}(0, \sigma).$$

6.3.2 Feature Selection using LASSO

To describe the demand, we consider a total of 56 variables. In an ordinary least squares regression this variety of features can lead to low predictive power due to problems such as multicollinearity. A common choice to overcome these problems, is the use of a regularisation technique, such as the least absolute shrinkage and selection operator (LASSO) (Hastie et al. 2013; Tibshirani 1996). The LASSO regression shrinks some

2 Although demand data for 2018 is available, the weather data for all of October 2018 is missing and we thus ended the testing with the last year completely available.

parameters towards zero. Although this shrinkage increases the bias, it improves the forecasting accuracy (Ludwig et al. 2015). We define the LASSO regression by

$$\hat{\beta}_{\text{LASSO}} = \arg\max_{\beta} \left(\frac{1}{2} \sum_{i=1}^{N} \left[y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j \right]^2 + \lambda \sum_{j=1}^{p} |\beta_j| \right).$$

In contrast to other regularisation techniques, the LASSO technique uses an L_1 penalty term, which sets some coefficients to exactly zero (Hastie et al. 2013). The LASSO can therefore be used as a feature selection method. However, to efficiently use the LASSO, the choice of λ is essential. In our case, we use a cross-validation technique to find the optimal λ parameter. Through the LASSO technique we select a total of 56 variables with coefficients greater than zero, in the model. The variables used and their corresponding coefficients can be found in Figure 6.9. The biggest effect can be found in the special and proximity days as well as special holidays such as New Year's Day and Christmas.

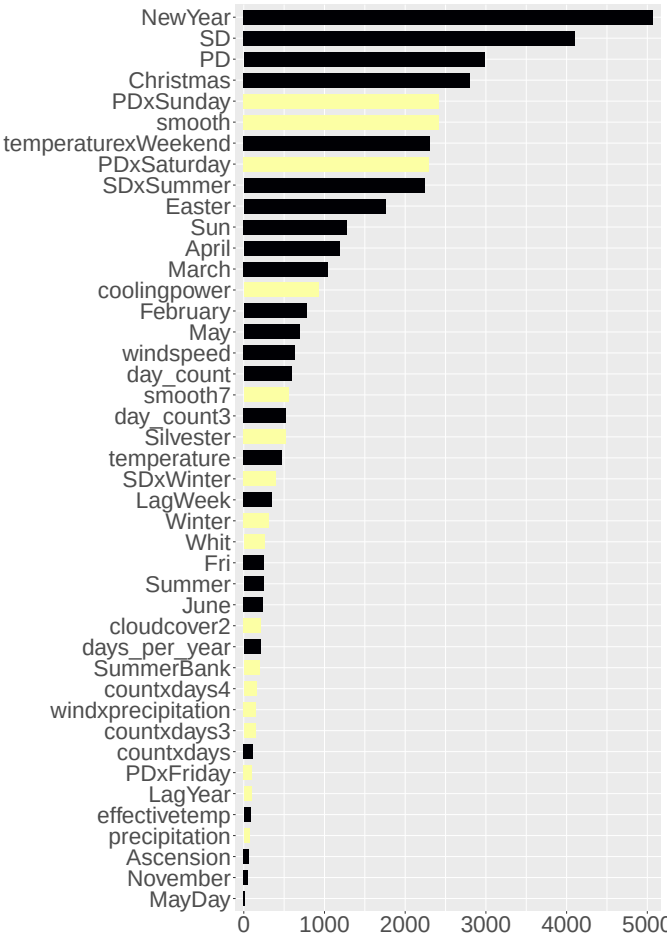


Figure 6.9 LASSO coefficients for all variables, with dark bars indicating a negative coefficient and light bars indicating a positive coefficient.

6.3.3 Monte Carlo Adapted Random Forest

Monte Carlo adapted random forests is based on quantile random forests introduced by Meinshausen (2006). The quantile random forest is adapted such that we can use data from a Monte-Carlo simulation as input as well as form a prediction distribution based on all leaf nodes.

In general, random forests (Breiman 2001; Hastie et al. 2013) is an ensemble learning technique, where several decision trees are grown using independent observations. At each tree and each node, the splits are determined randomly and each tree only learns with a specific predictor subset of the training data. To obtain a continuous prediction output, the random forest averages the output value over all individual trees.

In contrast to classical random forests, quantile random forests grows k trees $T(\theta_t), t = 1, \ldots, k$ and instead of just keeping the averages, keeps all information of every leaf of every tree. This information is used to compute a weight of each observation for every tree. The average weight $w_i(x)$ over all the k trees is than used to estimate a distribution function such that

$$F(y|X = x) = \sum_{i=1}^{n} w_i(x) \mathrm{I}_{\{Y_i \leqslant y\}}, \tag{6.1}$$

where Y_i are observations to be predicted and X_i is the input data. The quantiles are then estimated from the resulting empirical distribution function.

6.3.4 Error Measures

To properly compare the effect of post-processing the ensemble weather information, as well as the forecasting methods, we use four different forecast error measures. The Mean Error (ME), the Root Mean Squared Error (RMSE), the mean average percentage error mape and the Continuous Ranked Probability Score (CRPS). The first three error measures give us information about the point forecast accuracy with the ME being the simple difference, the RMSE putting more penalty on large deviations and the Mean Absolute Percentage Error (MAPE) being scale-independent. The ME, RMSE and MAPE are defined as

$$\begin{aligned}
\mathrm{ME} &= \frac{1}{N} \sum_{t=1}^{N} y_t - \hat{y}_t, \\
\mathrm{RMSE} &= \sqrt{\frac{1}{N} \sum_{t=1}^{N} (y_t - \hat{y}_t)^2}, \\
\mathrm{MAPE} &= \frac{1}{N} \sum_{t=1}^{N} \left| \frac{y_t - \hat{y}_t}{y_t} \right|,
\end{aligned}$$

with y_t being the real demand value at time point t and $\hat{y}_t$ being the forecast demand value at this time point. The scores are calculated at each time point and then averaged over the whole time series with length N. The last error measure is used to assess the calibration and sharpness of the probabilistic forecast

$$\mathrm{CRPS}(F, y) = \int_{\mathbb{R}} \left(F(z) - \mathbb{1}\{y \leqslant z\}\right)^2 dz,$$

with F being the demands predictive cumulative distribution function, y the verifying observation and $\mathbb{1}$ denoting an indicator function. The reported score is again averaged over all time points

$$\mathrm{CRPS} = \frac{1}{N} \sum_{i=1}^{N} \mathrm{CRPS}(F_i, y_i).$$

6.4 Evaluation

Given the forecasting and post-processing methods, we can now evaluate and compare them when forecasting the national electricity demand of Great Britain. Having post-processed and coupled all ensemble weather information, we use it as input to our forecasting scheme. For each day 12 noon, we forecast the next six days ahead using the previously selected variables including past load information, calendar variables and the weather ensembles. After each week, we re-estimate the parameters of the models.

In total we compare the performance of 9 different forecasting models. Each of the two methods (linear regression and monte-carlo adapted random forest) is trained with different ensemble weather input information on the six forecasting horizons. The linear model with actual weather information can be seen as the lower bound for the point and probabilistic forecasts. We do not expect any model with forecasted weather variables to perform better than the same model would perform with the real weather information at hand. The linear model with no weather input data can be seen as an upper bound for the forecasting performance. We expect all methods including weather information to outperform this model. In between this upper and lower bound, we compare the models which include the raw ensembles, the EMOS post-processed ensembles and the ECC post-processed ensembles. A summary of all results for the different models is given in Table 6.3.

We compare the linear model with different weather inputs in Figure 6.10. The post-processed and raw ensembles perform significantly better than the benchmark using no weather data. Additionally, they also perform similar or even slightly better than the actual weather data. While the post-processes ensembles outperform the raw ensembles regarding the CRPS and RMSE score, the raw ensembles perform better with respect to the MAPE. Overall, the forecasting performance can be improved by up to 20% for both the point and the probabilistic forecast, when the post-processed ensembles are used as weather input. The Monte-Carlo adapted random forest unfortunately performs worse than the linear model on all horizons and regardless of the weather input. The

Table 6.3 Comparison of results for different scores, models and forecast horizons.

Model	h	ME	RMSE	MAPE	CRPS
LM Actuals	1	377.94	1741.56	4.33	2.62
LM EMOS	1	405.79	2036.45	4.95	3.10
LM ECC	1	546.93	1786.61	4.45	2.58
LM Raw	1	436.61	1754.08	4.40	2.60
Forest Actuals	1	-381.32	1893.62	4.87	2.79
Forest Raw	1	-265.82	1913.40	4.90	2.81
Forest ECC	1	-215.25	1937.91	4.96	2.85
LM No Weather	1	160.61	2102.45	5.06	3.08
LM Actuals	2	364.61	1681.13	4.20	2.57
LM EMOS	2	357.39	2009.02	4.92	3.18
LM ECC	2	500.61	1777.27	4.48	2.66
LM Raw	2	390.57	1727.92	4.37	2.75
Forest Actuals	2	-327.40	1884.56	4.85	2.78
Forest Raw	2	-224.41	1918.62	4.96	2.81
Forest ECC	2	-249.51	1972.14	5.07	2.88
LM No Weather	2	145.73	2043.47	4.95	3.04
LM Actuals	3	415.25	1703.90	4.33	2.63
LM EMOS	3	392.02	2014.11	4.91	3.12
LM ECC	3	496.81	1764.54	4.42	2.60
LM Raw	3	440.99	1756.97	4.40	2.69
Forest Actuals	3	-298.14	1879.53	4.83	2.77
Forest Raw	3	-243.66	1934.71	4.97	2.82
Forest ECC	3	-239.29	1990.79	5.07	2.91
LM No Weather	3	182.11	2124.45	5.22	3.17
LM Actuals	4	421.75	1712.29	4.35	2.61
LM EMOS	4	612.87	2075.52	5.06	3.10
LM ECC	4	664.53	1856.85	4.65	2.69
LM Raw	4	668.43	1844.58	4.64	2.73
Forest Actuals	4	-393.73	1913.39	4.96	2.81
Forest Raw	4	-223.20	1994.43	5.10	2.89
Forest ECC	4	-184.65	2062.51	5.30	2.99
LM No Weather	4	187.10	2128.18	5.23	3.15
LM Actuals	5	429.58	1689.48	4.26	2.57
LM EMOS	5	310.56	1968.85	4.80	2.94
LM ECC	5	371.85	1793.92	4.48	2.61
LM Raw	5	345.91	1775.37	4.46	2.64
Forest Actuals	5	-501.52	1928.02	4.97	2.84
Forest Raw	5	-273.92	1996.03	5.07	2.90
Forest ECC	5	-257.87	2014.92	5.05	2.95
LM No Weather	5	195.31	2065.27	5.06	3.04
LM Actuals	6	426.58	1686.69	4.25	2.56
LM EMOS	6	600.65	2015.04	4.96	3.08
LM ECC	6	653.91	1901.27	4.73	2.77
LM Raw	6	596.23	1877.12	4.69	2.81
Forest Actuals	6	-449.58	1912.81	4.92	2.80
Forest Raw	6	-278.57	2058.33	5.23	2.96
Forest ECC	6	-272.60	2134.87	5.42	3.10
LM No Weather	6	192.34	2052.25	5.02	3.05

EMOS model, where the ensembles are calibrated but no ensemble copula coupling is used, performs worse than either the post-processed or raw ensembles. This result highlights, that while calibration is important, not accounting for the dependencies among the weather variables and in time makes the calibration worthless for the demand forecast. Throughout the whole testing period the performance of the best model can be seen in Figure 6.11, where we plot the 95% prediction interval (grey area) as well as the median (in orange) and the true value (in blue).

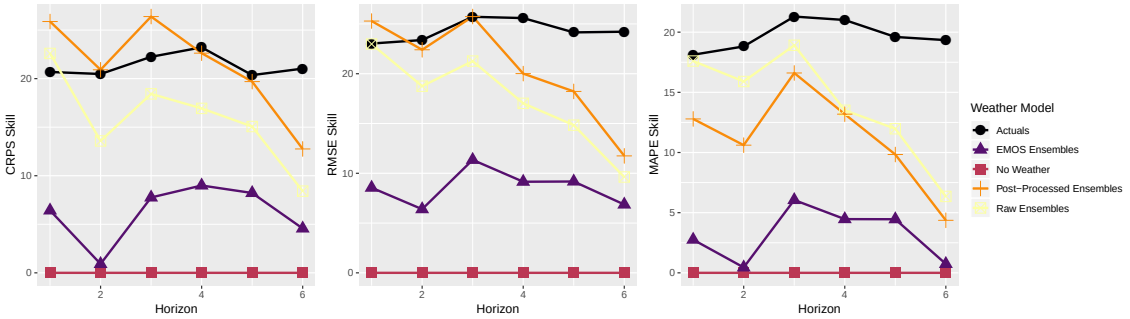


Figure 6.10 Skill scores for CRPS, RMSE and MAPE over all forecast horizons and the different weather inputs to the linear model.

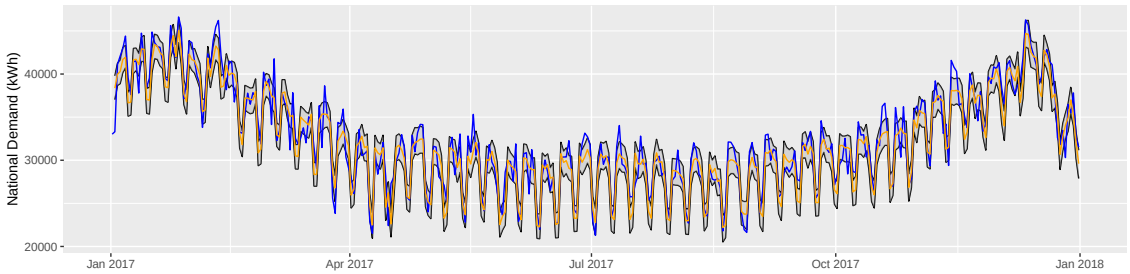


Figure 6.11 Forecasting results with 95% interval (grey area) for the linear model with post-processed and coupled weather ensembles on a one day-ahead forecasting horizon.

Additionally to the forecasting performance as measured by the different scores, we also analyse the sharpness of the predictive distribution. As can be seen in Figure 6.12, on a forecast horizon of one-step ahead, the sharpest forecast can be achieved using the post-processed ensembles. This result is in line with the previously reported CRPS score also being the best for the LM model with post-processed ensembles as input.

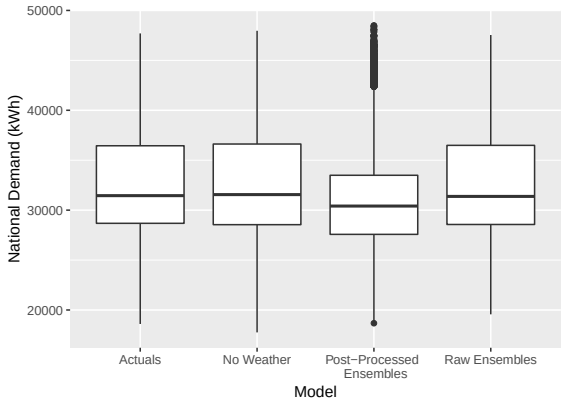


Figure 6.12 Comparison of the prediction sharpness over the different weather inputs to the linear model at forecasting horizon of one.

6.5 Conclusion

Probabilistic demand forecasts are increasingly important with a larger share of RES in the grid and are essential to establish if there is a mismatch between demand and supply and thus a need for DSM. The demand is influenced by the weather, which itself is a chaotic system. Any probabilistic demand forecast including weather information thus needs to properly account for the uncertainty in the weather variables. Numerical Weather Prediction (NWP) are probabilistic forecasts of the weather and thus an ideal input to demand forecasts. However, while it is well known to meteorologists that the weather variables out of such systems can be helpful but are uncalibrated and often biased, this information has not yet been used for energy demand forecasting.

This chapter has shown how to properly calibrate the ensemble weather information accounting for temporal correlation as well as correlation between the weather variables. We show that calibrating the ensembles, especially accounting for their dependencies with the help of a copula based coupling approach, improves the probabilistic forecast accuracy. The proper post-processing of the ensembles increases the sharpness of the forecast and results in a Continuous Ranked Probability Score (CRPS) which is up to 20% better than not including any weather information. Additionally, the post-processed ensembles also outperform the raw ensembles, highlighting the advantage of proper post-processing not only for better weather forecasts but also for forecasts depending on the weather such as the electricity demand.

In future work, we want to test other forecasting times and investigate other non-linear approaches such as neural networks. Additionally, the influence of proper post-processing when directly forecasting the mismatch between demand and supply might be of interest as well.

7 Assessing the Amount of Flexibility Needed

The previous chapter has highlighted the need for proper post-processing in demand forecasts, which are essential in order to assess if any flexibility is needed in the system. Additionally to knowing if we need flexibility at all, one further question arises, namely how much flexibility do we need.

We propose, implement and evaluate a new framework that takes power consumption time series of industrial processes as input and indicates which processes should be made flexible to optimise for several objectives. With this framework, we want to answer the question: How much flexibility do we need? Thus, given all the processes we have, how many must be made flexible (and by how much) to get improvements regarding the energy consumption? To the best of our knowledge, we are the first to propose this approach to DSM. Based on the power consumption time series from an electronics factory which we have already seen in the previous parts, we show that we can achieve notable improvement even with little newly-created flexibility.

Before describing the proposed technique, we formally define the terminology and problems used throughout this chapter in Section 7.1 and summarize related work in Section 7.2. Then, we describe the new approach and its individual steps which are based on the methods introduced in Chapter 4 in Section 7.3. We evaluate the approach in Section 7.4 and discuss the results in Section 7.5. We finally conclude in Section 7.6.

Parts of this chapter are reproduced from

L. Barth, V. Hagenmeyer, N. Ludwig, and D. Wagner (2018a). "How much demand side flexibility do we need? Analyzing where to exploit flexibility in industrial processes". In: *Proceedings of the Ninth International Conference on Future Energy Systems - e-Energy '18*. ACM Press, pp. 43–62. DOI: 10.1145/3208903.3208909

7.1 Scheduling Problem Definition

The aim of the scheduling problem is giving out a measure for how much the peak demand can be reduced, given a specific amount of flexibility. Thus, we do not seek an optimal schedule given a number of jobs, but want to analyse how a schedule changes given more or less flexibility options. Hence, we first need a schedule, which we define as

Definition 7.1 (Schedule). A schedule is a set of n triples $(c_i, p_i, u_i) \in \mathbb{N} \times \mathbb{N} \times \mathbb{R}$ for $i \in \{1,2,\dots n\}$. Each triple describes a *job* in the schedule. We identify the triple (c_i, p_i, u_i) with job i. For job i, the field c_i indicates the current *start time* in the schedule, p_i indicates the *duration* of the job and u_i specifies the *amount of power* that the job uses during execution.

Given a schedule as in Definition 7.1, we can now define the flexibility limits which we impose on this schedule.

Definition 7.2 (Flexibility Limits). Two integers $\hat{T} \in \mathbb{N}$ and $\hat{J} \in \mathbb{N}$ are used to limit the amount of flexibility that can be integrated into the schedule. Here, $\hat{J}$ specifies how many jobs may be moved away from their current start time and $\hat{T}$ specifies by how many times steps these jobs may be moved in total.

With the above two definitions in mind, we can now formally introduce the scheduling problem that we examine throughout this section of the thesis

Definition 7.3 (Flexibilisation Project Scheduling Problem (FPSP)). Given a schedule as in Definition 7.1 and flexibility limits as in Definition 7.2, find for each job $i \in \{1,2,\dots\}$ a new start time $s_i \in \mathbb{N}$ such that

1. the number of jobs i for which $s_i \neq c_i$ is at most $\hat{J}$
2. the total deviation of current and new start times is at most $\hat{T}$, i. e. $\sum_{i=1}^{n} |c_i - s_i| \leqslant \hat{T}$.

Any schedule S that meets the criteria in Definition 7.3 is a *feasible* solution of the Flexibilisation Project Scheduling Problem (FPSP) problem. Given this formulation of the problem we can investigate how inserting more flexibility changes and in an ideal case improves the schedule without having to specify anything about the flexibility of each individual job. However, one still has to specify the limits $\hat{T}$ and $\hat{J}$. In Barth et al. (2018a), it is also discussed how one can get rid of specifying these limits if the costs of flexibility can be determined. So far we have established how a feasible solution is determined, but not yet what determines an *optimal* solution. In fact, we want to evaluate two different optimisation functions for which two different schedules can be optimal. The first optimisation function we consider is peak shaving, while the second considers overshoot minimisation. For both objectives, we need to know the sum of the amount of power that is consumed at each time step. Let this total amount of power be

$$U_t = \sum \{u_i \mid s_i \leqslant t \wedge s_i + p_i > t\}.$$

If we want to optimise the flexibilisation project scheduling with peak shaving, thus minimising the maximum that occurs within the schedule by executing jobs simultaneously, the objective function which the schedule should minimise becomes

$$\hat{U} = \max_t U_t.$$

Additionally, if we take the generation into account, we can also investigate the objective function for overshoot minimisation

$$\hat{U} = \sum_{t=0}^{\infty} \max(U_t - G_t, 0),$$

where $G_t \in \mathbb{R}$ are the units of renewable energy available at time step t. With this objective function, we try to minimise generation of energy which is not already covered by the renewables. For both objective functions, any schedule minimising $\hat{U}$ is a feasible schedule.

7.2 Related Work

We focus on the analysis of *time flexibility*, hence jobs can be moved in time but have to run, they also do not change the way they demand energy (which would be the case for energy flexibility). The possibility to move jobs in time is also referred to as *shifting demand*, be e. g. Palensky and Dietrich (2011).

Although there are studies concerned with how much flexibility consumer can provide (D'hulst et al. 2015), even on a device level (Truong et al. 2016), we are not aware of anyone investigating the amount of demand-side flexibility is needed to improve the energy consumption behaviour. Instead, most studies focus on what flexibility in general could be worth (Pudjianto and Strbac 2017; Feuerriegel and Neumann 2016; Ambrosius et al. 2016; Taneja et al. 2013). Moreover, as we have seen in Part II, many papers schedule flexible demands without investigating how many of those demands need to be changed by the scheduler to improve the energy behaviour.

Improving the energy demand is a rather ambiguous aim. Many applications take a look at peak minimisation, e. g. Liu et al. (2013) try to avoid peak demands in data centres, while Zhao et al. (2017) schedule charging electric vehicles. Our formulation is inspired by an industrial setting and we thus focus on scheduling non-pre-emptive and deferrable loads, as for example O'Brien and Rajagopal (2016).

We detect the loads that we want to schedule from historical industrial load time series. While there exist many techniques to find when processes start and to monitor appliances, most prominently non-intrusive load monitoring (Hart 1992), and also Ardakanian et al. (2017) and Rollins and Banerjee (2014), these techniques work only supervised. Supervised learning techniques are not applicable to our data, as we lack any labels. We thus choose to find the pattern using motif discovery. Our approach also touches

the field of project scheduling. Research in this area is vast, many variations of problem settings have been explored, for an extensive survey, see Weglarz (2012).

7.3 Framework

In the present section, we describe the framework, as well as the data used. The framework consists of four steps. Three steps precede the scheduling, namely, sequence detection in the time series, motif discovery and the generation of the synthetic data set, which is used to evaluate our approach. All these steps are described in the following section. The model-based scheduling algorithm is run afterwards to evaluate how much flexibility we need. We omit the description of the scheduling and refer to Barth (2020) for more details. Additionally, while the generation of synthetic data could be omitted, we include it so that we can evaluate the approach on more than one data set.

For the sake of clarity, we repeat the terminology. The initial data set is a set of electrical consumption time series from *machines*, where there exists exactly one time series per machine. We assume that a machine can either run a *process* (see Definition 2.1) or be idle. The part of the machine time series where a process is running, is denoted a (time series) *sequence* (see Equation (2.2)). If several sequences are similar, we assume they describe the same process. In this case, we say they are *occurrences* of the same *motif* (see Part II for more details on motifs and their occurrences). The motifs are used to generate synthetic test data, which are sets of *instances*, each consisting of *jobs* (see Definition 3.1). Thus one job represents a sequence of a machine which is not idle.

7.3.1 Data

Our input data is the HIPE dataset (Bischof et al. 2018), also used in the previous parts and described in more detail in Section 4.3. As we have established above, we are looking for sequences in the machines' time series and proceed as in Section 4.3 using a clustering approach to determine active and passive states of a machine.

The points in time where a time series then goes from zero to non-zero power demand are the *start points* of a *sequence*. The sequence ends when the power demand goes back to zero, at its *end point*. To be able to compare sequences in a meaningful way later, we normalize the lengths of all sequences on a per-machine basis. For each machine, we determine the 80% quantile of the lengths of its sequences. We scale all sequences of this machine to this length for our motif discovery. We chose 80% because, on the one hand, stretching sequences comes with less data loss than compressing. On the other hand, taking the longest sequence would increase the impact of outliers.

7.3.2 Motif Discovery

Given the data sequences describing the machines active power intake, we now use motif discovery to find recurring patterns in these sequences (Ludwig et al. 2017). We use an equal length version of the eSAX algorithm introduced in Section 4.1, where all found subsequences have been normalised to the same length.

Figure 7.1 shows a discovered motif with all its occurrences in grey lines. The coloured lines represent the mean, 20% quantile and 80% quantile of all the occurrences. These lines can give a feeling for how a *standard* process might look like for this machine.

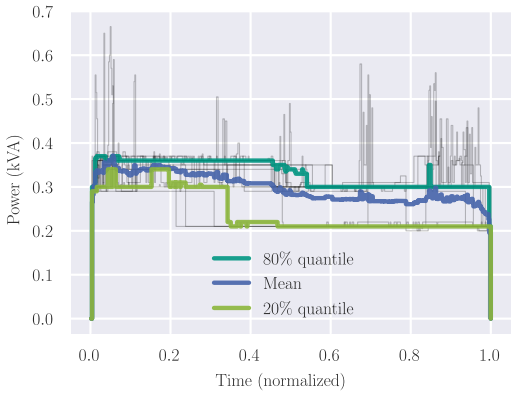


Figure 7.1 The discovered motif A with 16 occurrences. Each grey curve shows one occurrence, with the length of all occurrences being normalized to 1. Their shared common features, roughly represented by the mean and upper and lower quantiles, constitute the motif.

7.3.3 Generation of Synthetic Instances

To evaluate the feasibility and usefulness of the proposed approach, we need many instances of the FPSP problem which emulate real-world processes. As the original data set we have is rather small, we generate artificial instances for the FPSP problem from the motifs discovered as described in Section 7.3.2. In our generated instances, the start times, job durations and power requirements are statistically derived from the discovered motifs. These characteristics are key factors with regards to peak power demands during a schedule. Since we preserve these characteristics of the discovered motifs, we expect our generated instances to adequately resemble reality.

To generate the new sequences, we describe every motif by three normal distributions: One energy distribution, one start time distribution, and one duration distribution. Each of these distributions is determined by fitting a normal distribution to the lengths, start times and energy consumptions of the respective motif's occurrences. The start time distribution is actually a mixture of normal distributions: We assume that the same process might have several times within a day at which it usually starts. To factor this in, we cluster the start times of every motif's occurrences (using affinity propagation) and

generate a normal distribution for every cluster. We then create a mixture distribution for that motif's start times, weighting each normal distribution by the size of its cluster. However, for most motifs, only a single cluster was found.

We generate instances (i. e., sets of jobs) with a fixed number of jobs by repeatedly randomly selecting a motif (weighted by the number of the motif's occurrences) and then generating a job for this motif. We assume all jobs to have a non-varying power usage, which obviously cannot capture all of the complexity seen in real world power demands. However, many of our discovered motifs are actually rather block-shaped. For each job, we generate a duration by randomly drawing from the respective motif's length distribution. We discard any length of less than one time step. Similarly, we determine a start time for this job by randomly drawing from the respective motif's start time distribution. Finally, we randomly generate an energy consumption for the job by drawing from the motif's energy distribution. The job's power demand is set as energy consumption divided by duration.

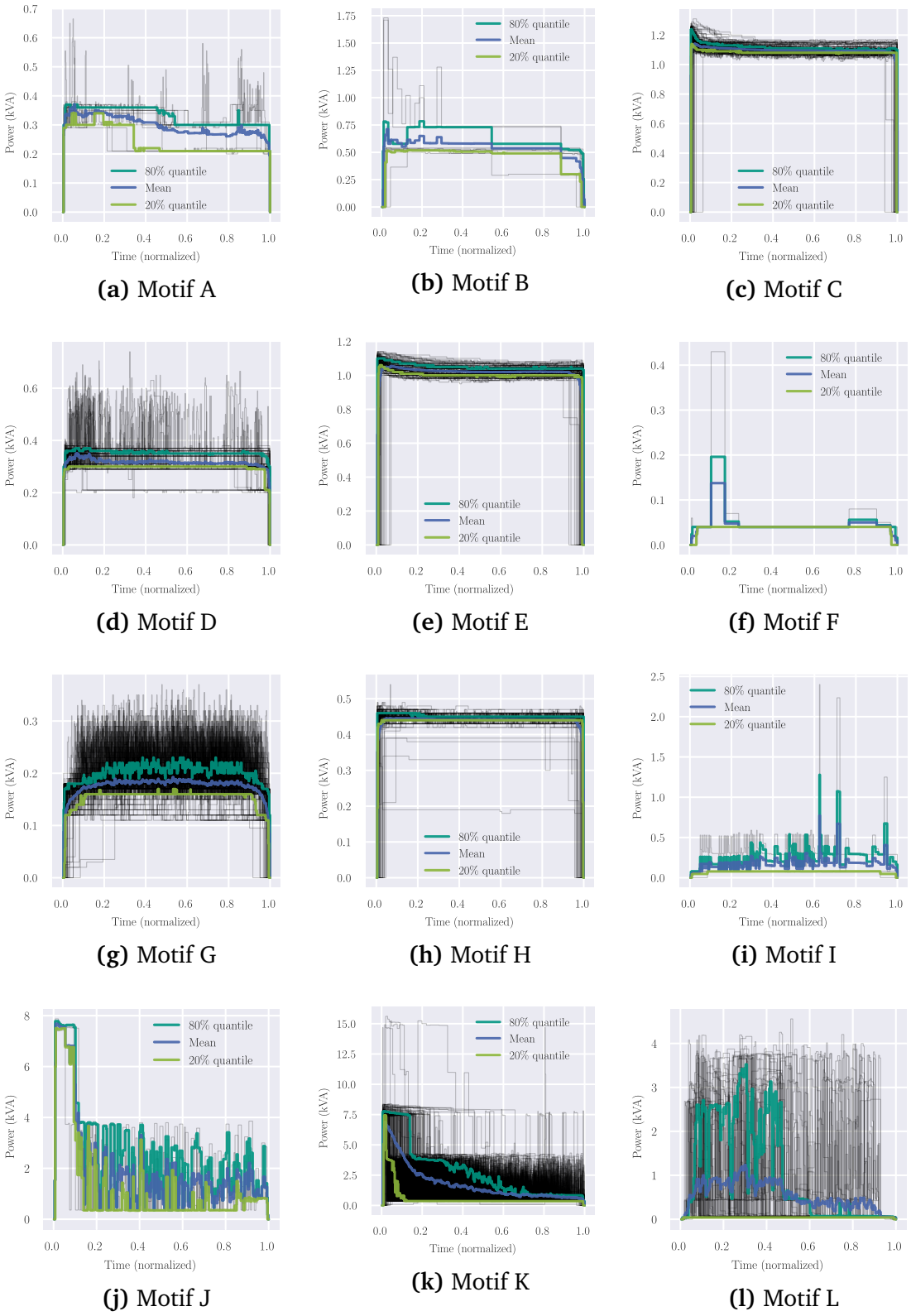
Since drawing values from normal distributions may yield extreme results in a few cases, which nonetheless are sufficient to substantially skew the complexity and results of the scheduling problem, we discard any values that deviate from the mean by more than three standard deviations.

7.4 Evaluation

Having introduced the data with which we work and our framework, we now evaluate our approach. We start with describing the motifs we find, and the instance sets generated, before assessing each of our problem sets individually.

7.4.1 Discovered Motifs

Given the machines, we find a total of 15 motifs in our time series data. The resulting motifs for each machine can be found in Figure 7.2. The parameters for the motif discovery algorithm in this section (summarised in Table 7.1) are tailored to our specific problem at hand. For example, we use a relatively small alphabet size for most machines as their variations are small. We also choose all parameters in such a way that the algorithm can classify most sequences without assigning all of them to a single motif.



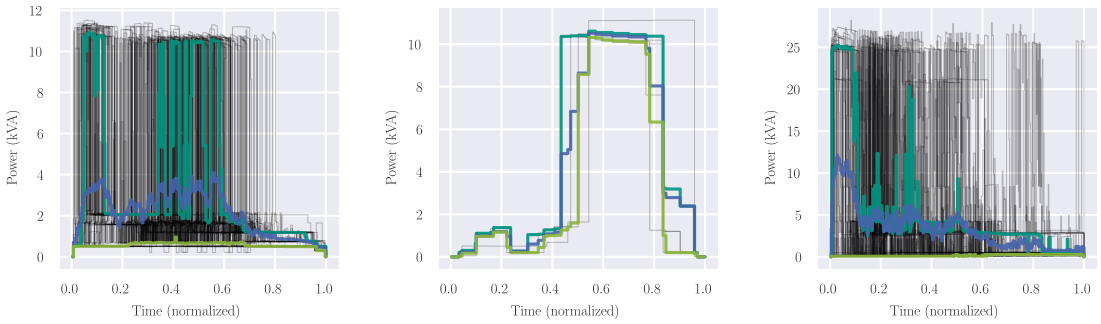
(a) Motif A **(b)** Motif B **(c)** Motif C

(d) Motif D **(e)** Motif E **(f)** Motif F

(g) Motif G **(h)** Motif H **(i)** Motif I

(j) Motif J **(k)** Motif K **(l)** Motif L

Figure 7.2 All discovered motifs. Each black line indicates one occurrence of the respective motif.



(m) Motif M **(n)** Motif N **(o)** Motif O

Figure 7.2 All discovered motifs. Each black line indicates one occurrence of the respective motif (cont.).

All sequences which are not classified as belonging to one of the motifs are classified as noise and excluded from further analysis. In future work, we might want to do an extensive evaluation of our parameter choices. However, we expect the settings we have chosen to be sufficiently useful for our problems at hand.

Machine	Motifs	Alphabet Size	Wordlength
AVT 01	A, D	4	505
AVT 02	J, K	4	403
AVT 03	B, M, N	4	267
AVT 04	H	4	433
AVT 05	C	4	543
AVT 06	E	4	406
AVT 08	F, L	4	211
AVT 09	G	4	500
AVT 10	I, O	4	426

Table 7.1 Parameter choices for the motif discovery algorithm. The alphabet size was varied between 2 and 10 words, resulting in largely the same results as presented above.

We now analyse the discovered motifs further, especially with regard to the question whether jobs with constant power demand are a reasonable approximation of the motifs, and if not, how much better the approximation becomes when we allow to split the jobs into multiple blocks.

The occurrences of motifs correspond to stepwise functions: Every point in the occurrence's power demand time series results in one step in its power demand function. Let o be an occurrence. We then call the (stepwise) function mapping a point in time to the power demand of the occurrence at that time $P_o\colon [0,1] \to \mathbb{R}$ (note that occurrences are normalized, thus a point in time is in $[0,1]$). The main question is how well we can approximate these functions with other stepwise functions *of low complexity*, i. e., with few steps. A job with a constant power demand corresponds to a stepwise function with exactly one step, a chain of two jobs corresponds to a stepwise function with up to two steps, and so on. Let $\tilde{P}_{o,k}\colon [0,1] \to \mathbb{R}$ be such a function with at most k steps, that tries to approximate P_o.

We need some notion of the difference between P_o and $\tilde{P}_{o,k}$ and thus suggest

$$\Delta\left(P_o, \tilde{P}_{o,k}\right) = \frac{1}{N_o} \int_0^1 \left(P_o(t) - \tilde{P}_{o,k}(t)\right)^2 \, \mathrm{d}t,$$

where, N_o is a normalization factor to make different motifs comparable $N_o = \int_0^1 P_o(t)^2 \, \mathrm{d}t$. Since both functions are discrete in t, we can compute the value of the integral without integrating. This metric penalizes deviations of $\tilde{P}_{o,k}$ from P_o with a quadratic term. We assume a deviation that is large in magnitude but short in time to be worse than a deviation which is small in magnitude but long in time, because deviations of large magnitude might hide exactly the peaks in power demand that we are interested in reducing.

To analyse how block-shaped our motifs really are, we fitted multiple $\tilde{P}_{o,k}$ (for multiple values of k) to the P_o of every occurrence o.[1] Our first attempt is $k = 1$, i. e., a step function with exactly one step, representing jobs with constant power demand. We see the value of $\Delta(P_o, \tilde{P}_{o,1})$ in Figure 7.3. The x axis groups the occurrences by their motif. We sort motifs by how well they are approximable for $k = 1$.

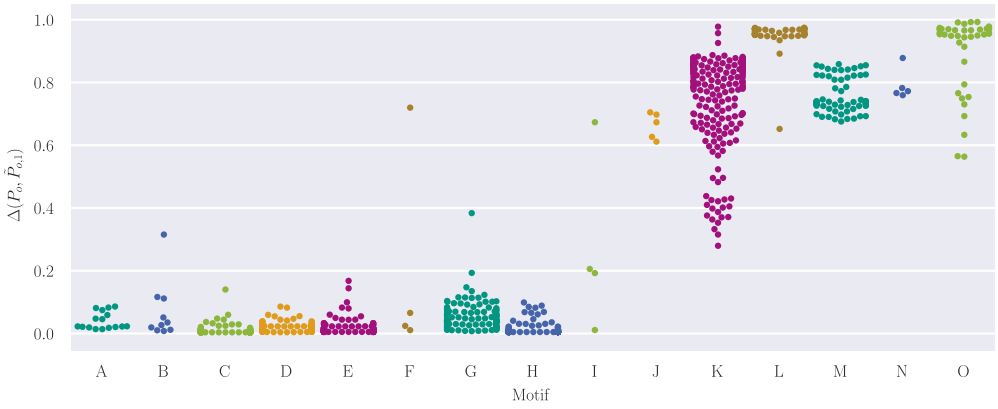
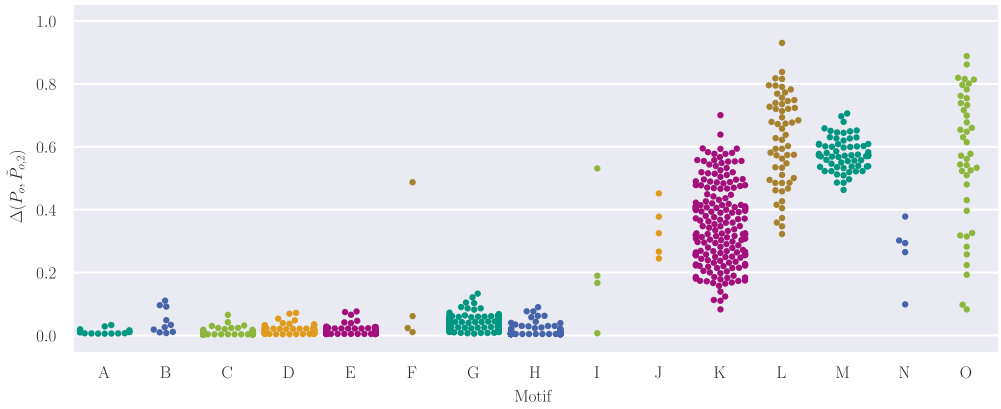


Figure 7.3 The difference between all occurrences' P_o and their respective optimal $\tilde{P}_{o,1}$

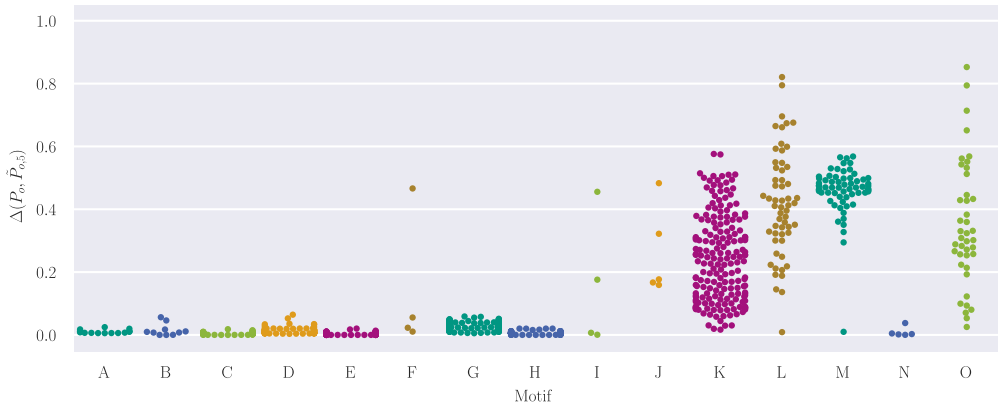
We cannot sufficiently say what values for $\Delta(P_o, \tilde{P}_{o,k})$ are good or bad, as the question of what is an acceptable approximation must be answered by the person using the framework. However, we can clearly see that nine of our fifteen discovered motifs are a lot better approximated by a job with constant power demand than the remaining six. This seems intuitively correct when looking at the motifs in Figure 7.2.

We can also see how $\Delta(P_o, \tilde{P}_{o,k})$ changes when we increase $k = 1$ to $k = 2$, which is shown in Figure 7.4a. We see that the change is substantial for the six motifs that were not well approximated before. Especially motifs L, N and O seem to profit from a two-step function. This profit of more blocks seems plausible, when looking at these motifs in Figure 7.2l, Figure 7.2n and Figure 7.2o.

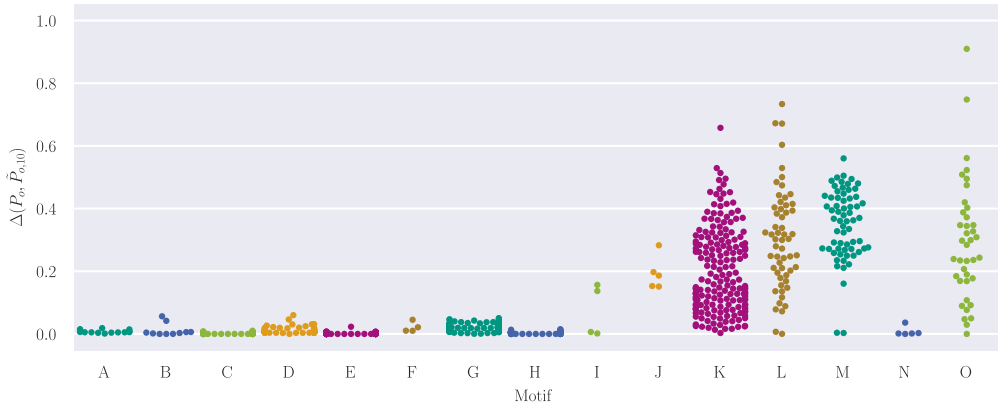
1 Using a black-box SLSQP optimiser on $\Delta(P_o, \tilde{P}_{o,k})$



(a) $k = 2$



(b) $k = 5$



(c) $k = 10$

Figure 7.4 The difference between all occurrences' P_o and their respective optimal $\tilde{P}_{o,k}$

We see a similar effect when considering $k = 5$ (see Figure 7.4b) and $k = 10$ (see Figure 7.4c): The $\Delta(P_o, \tilde{P}_{o,k})$ values shrink gradually for the six motifs which are not very block-shaped, although the improvement is less than the one going from $k = 1$ to $k = 2$.

We can thus conclude that being able to approximate the motifs with stepwise functions of more than one step most likely brings the results of the optimisation closer to reality. Especially allowing for two jobs instead of one might be a worthwhile option. However, we can also conclude that a constant function is already a good approximation for the majority of the discovered motifs, and is not completely outlandish for the rest of the motifs either.

7.4.2 Instance Sets

We generate four sets of instances from the data, resembling the input data from Section 7.3.1 to varying degrees. In the following, we introduce each of the four sets before evaluating them.

Set PS-Nonuniform The first set we evaluate is the *PS-Nonuniform* set, thus a set for peak shaving with non-uniform jobs. We proceed as described in Section 7.3.3. Job lengths, job power demands and job start times are generated from the normal distributions fitted to the discovered motifs. To keep the scenario realistic, we generate instances that span a working week (i. e. five days). Thus, for every job, we pick a start time from the corresponding start time distribution and additionally pick uniformly at random at which of the five days the job starts. Although, the data is available in a finer resolution, one time step corresponds to five minutes in our framework to keep the computational costs at bay. Each instance is then generated with 150 jobs.

With the generated instances at hand, we are left with the choice of our flexibility parameters $\hat{T}$, the amount of job movement allowed, and $\hat{J}$, the number of jobs allowed to be moved. We will discuss both in the following.

It seems reasonable to specify $\hat{T}$ relative to the instance size. We therefore introduce Θ and set $\hat{T} = \Theta \cdot \sum_i p_i$, where Θ specifies the fraction of cumulative duration that jobs may be moved. We test all values for $\Theta \in \{0.005, 0.01, 0.02, 0.03, 0.04\}$ and Table 7.2 shows statistics about the values that result for $\hat{T}$ for the various values for Θ. We see that the values for $\hat{T}$ range from about 3.5 hours to about 30 hours.

Table 7.2 Statistics of $\hat{T}$ values for all possible values of Θ.

	$\hat{T}$ (hours)	
Θ	Mean	Std. Dev.
0.005	3.6	0.20
0.01	7.2	0.39
0.02	14.3	0.77
0.03	21.5	1.16
0.04	28.7	1.54

For $\hat{J}$, the number of jobs allowed to be moved, we investigate all of $\hat{J} \in \{3, 6, 9\}$, for a total of 15 different flexibilization limits. We generate 30 sets of 150 jobs each, and pair them with every flexibilization limit from above, leading to a total of 450 instances.

Set PS-Uniform The second set we evaluate is the *PS-Uniform* set, thus a set for peak shaving with uniform jobs. The heterogeneity in power demand between the generated jobs (arising from heterogeneous power demand in the discovered motifs) has a significant influence on the computational feasibility and the possible optimisation benefits of the instances. We do not wish to bias our conclusions on the basis of this phenomenon, which may not occur in other workloads. Hence, we generate the PS-Uniform instance set in which we use a single, fixed normal distribution for all jobs' power demands (with mean 30 and standard deviation 10). Aside from this, we proceed as for the PS-Nonuniform set.

Set PSG Our third set is the *PSG* set, where we again explore peak shaving, but with fluctuating generation. This set corresponds to a setting where e. g. solar generation is available and one tries to minimize the peak residual load.

We use a solar generation curve for one day derived from total solar generation data for Germany, Austria, and Luxembourg with quarter-hourly time resolution, which was retrieved from ENTSO-E.[2] For every quarter hour, we average the production from all summer days in the year 2016. In our instances, we set available generation (i. e., G_t) on all five days based on this curve, scaling the curve such that in total, 20% of the total energy demand in each instance is provided via solar energy. Aside from that, we generate instances as described for the PS-Nonuniform set.

Set OM The last set we generate, is the *OM* set, where we evaluate the overshoot minimization objective. For the available generation, we assume that we can meet 65% of the total energy requirement in each instance by own generation. This number results from our calculations of the energy consumption and production in the summer month of BASF based on Hagenmeyer et al. 2014. As BASF's power plants are steam-controlled, the generation in the winter months is so high that they can sell excess energy. However, during the summer months, the opposite is true, and they have to buy energy from the grid, which is more expensive. We assume the generation to be a flat curve in our calculations. For a steam-controlled power plant and a time horizon of five days, this is a realistic assumption since steam demand usually fluctuates relatively little. Formally, we set

$$G_t = \frac{\sum_i (p_i \cdot u_i) \cdot 0.65}{5 \cdot 24 \cdot 60/5} \qquad \forall t.$$

For this instance set, we use the overshoot minimisation objective, but proceed as for the PS-Nonuniform set otherwise.

2 Via Open Power Systems Data: https://data.open-power-system-data.org/time_series/

7.4.3 Evaluation Environment

For all instances, we build a Mixed-Integer Linear Program (MILP) model (as described in more detail in Barth et al. (2018b) and Barth (2020)). We optimise every model using Gurobi 7.0. We also evaluated optimising using CPLEX 12.8, but achieve slightly better results for our models with Gurobi. We use a system with dual AMD EPYC 7601 CPUs (having 64 physical CPU cores) with 512 GB of RAM and optimise every model for 45 minutes, allowing for 20 threads per solver, running 6 solvers in parallel.

MILP Gap In the following, we report the glsmilp gaps after optimisation together with the solution quality. The MILP gap is the difference between the best feasible solution found and the best lower bound that the solver was able to prove, divided by the best feasible solution. A large MILP gap is an indicator that further optimisation could potentially find better solutions. Please note that we optimise with a focus on finding high-quality solutions.[3] Thus, we could have further reduced MILP gaps at the cost of solution quality. We perform parameter tuning via Gurobi's auto-tuning tool. However, the default settings produced the best results for us.

7.4.4 Evaluation of Peak Shaving Sets

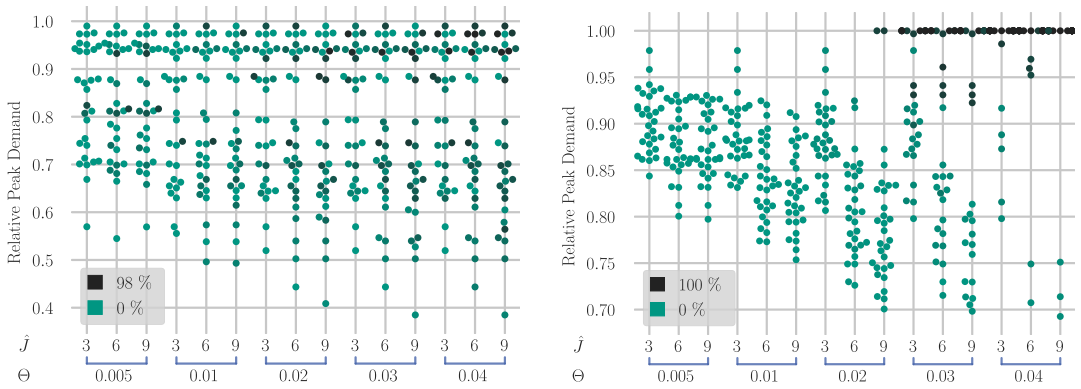
We start our evaluation by looking at the sets that represent peak shaving objectives. Figure 7.5a summarizes our results for the PS-Nonuniform set. Every dot represents one instance. Every column of dots represents one combination of $\hat{J}$ and Θ. The y-coordinate of the dot indicates the respective instance's peak demand after optimisation divided by the peak demand before optimisation, which we define as *relative peak demand*. We use the dots' colours to indicate the MILP gap achieved for the respective instance, where darker colours indicate larger MILP gaps, i. e., worse optimisation, and the lightest colour indicates that the instance was solved to optimality.

In Figure 7.5a we see that in the most extreme cases, peak demand is reduced by more than 60%. We also see that the improvement is mostly distributed evenly between 0% and about 40%, and that increasing $\hat{J}$ or Θ does not result in significant improvements. We find that increasing Θ yields larger peak reductions than increasing $\hat{J}$, and that improvements gradually diminish with larger values for Θ (respectively $\hat{J}$). However, the optimisation gets harder with increasing Θ. Since the reported MILP gaps after optimisation were large for many instances for $\Theta \in \{0.03, 0.04\}$, results may improve for those parameters if one optimises the instances further.

We perform a statistical significance test (Wilcoxon's signed-rank test [4]) on our findings. For every consecutive pair of $\hat{J}$ (respectively Θ) values, while keeping the Θ (respec-

3 We set the MIPFocus parameter to 1 for Gurobi.

4 Because we have many ties in our data, the way such ties are handled is important in our case. We use the approach suggested by Pratt (1959).



(a) Set PS-Nonuniform. **(b)** Set PS-Uniform.

Figure 7.5 Relative reduction in peak demand with one point per instance. The columns are the different settings for $\hat{J}$ and Θ. The Y-axis indicates the change in peak demand after optimisation. Colour indicates the remaining MILP gap when the optimisation was stopped.

tively $\hat{J}$) value fixed, we compute the p-value, which indicates how likely it is that the change in improvements is a random coincidence instead of an effect of altering $\hat{J}$ (respectively Θ). We report all values in Table 7.4a.

We want to assume significance with 95% confidence, i. e., say that a change is significant if the p-value is below 0.05. However, throughout the whole of this sections analysis, we perform a total of 88 such tests. With an error probability of 5%, we thus expect erroneously reporting significance for four tests. To compensate for this, one can apply a Bonferroni correction, which essentially means assuming significance only when the p-value is below $0.05/88 \approx 0.00057$.

We see that changing Θ from 0.02 to 0.03 and 0.04 does probably not result in significant improvements for the PS-Nonuniform set. This lack of improvement might be because the optimisation problem becomes too hard, but could also be because we already achieve optimal results for many instances with $\Theta = 0.02$ (see below). Aside from that, the only non-significant change is changing $\hat{J}$ from 6 to 9 at $\Theta = 0.005$ and $\Theta = 0.04$, which is likely because the little possible movement in time, can already be optimally distributed over six jobs, respectively because the instances got too hard.

To validate the very uniform distribution of improvements between 0% and 40%, we look into characteristics of the instances that allow for an unusually large or small improvement. We discover that instances which allow for almost no improvement always contain jobs with very high power demands compared to all other jobs' power demands, i. e., substantial heterogeneity in power demands. This makes sense intuitively, since the job with the largest power demand (which we call the *tallest* job) is a lower bound for the overall peak power demand. The margin for optimisation is at most the difference between the overall peak power demand and the demand of the tallest job.

Table 7.3 p-Values for the change of one parameter in the PS-Nonuniform and PS-Uniform set. Values highlighted in green indicate that changing $\hat{J}$ and Θ, while keeping the other one constant, results in a statistically significant change in improvements. Values in blue are significant before Bonferroni correction.

(a) p-Values for the change of one parameter in the PS-Nonuniform set.

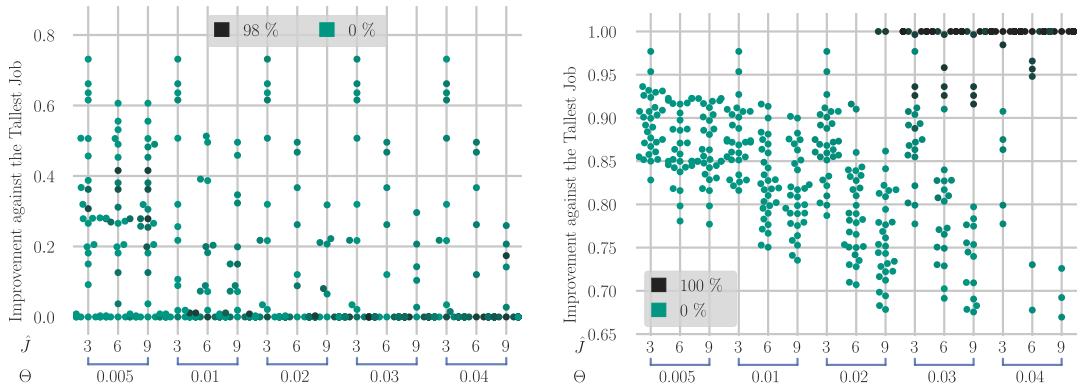
	3	→	6	→	9
0.005		$< 10^{-4}$		0.00089	
↓	$< 10^{-4}$		$< 10^{-5}$		$< 10^{-5}$
0.01		$< 10^{-4}$		0.00051	
↓	$< 10^{-4}$		$< 10^{-4}$		$< 10^{-5}$
0.02		$< 10^{-4}$		0.00039	
↓	0.0018		0.014		0.0032
0.03		0.00029		0.00014	
↓	0.012		0.047		0.035
0.04		$< 10^{-4}$		0.0018	

(b) p-Values for the change of one parameter in the PS-Uniform set.

	3	→	6	→	9
0.005		$< 10^{-5}$		0.00089	
↓	$< 10^{-5}$		$< 10^{-5}$		$< 10^{-5}$
0.01		$< 10^{-5}$		$< 10^{-5}$	
↓	0.00025		$< 10^{-5}$		0.00033
0.02		$< 10^{-5}$		0.00036	
↓	0.0008		0.03		0.041
0.03		$< 10^{-4}$		0.086	
↓	0.00011		$< 10^{-5}$		0.00022
0.04		0.00033		0.0018	

To account for this phenomenon, we also evaluate how well our optimisation performs within this margin, i. e., how the difference between peak demand and demand of the tallest job changes, which we define as the *improvement against the tallest job*. Figure 7.6a shows the results of this evaluation. A value of 0 indicates that after optimisation, the overall peak power demand equals the demand of the tallest job. The figure also shows that for the majority of instances, we are able to achieve this optimum. For all other instances, we improve by at least 20% within the margin between tallest job and original peak demand.

As mentioned above, we generate the PS-Uniform set to compensate for the property, where the peak demand is dominated by a single job. Figure 7.5b shows that even with power demands being drawn from a single distribution, peak demand reductions of 5% to 30% are realistic. However, solving these instances also becomes a lot harder



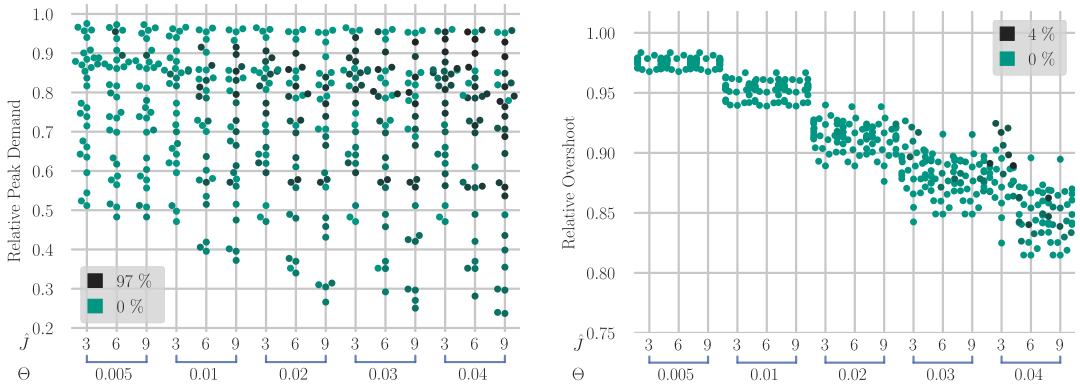
(a) Set PS-Nonuniform

(b) Set PS-Uniform

Figure 7.6 Change of the difference between the peak demand and the demand of the tallest job after optimisation.

and an interesting trend can be seen in the MILP gap. While almost all instances for $\Theta \leqslant 0.02$ could be optimised to within a 20% gap, often to optimality, the MILP gaps for $\Theta \geqslant 0.03$ are mostly above 60%. Thus, it seems like the computational complexity grows rapidly with $\hat{T}$ — this is certainly because of the larger solution space, but might also be exacerbated by numerical stability issues with constraints in the optimisation problem. The results for the PS-Uniform set are, as expected, more tightly clustered. We again evaluate the improvement towards the tallest job in Figure 7.6b. Since in PS-Uniform, the peak power demand is not dominated by single jobs any longer, this now correlates closely with the absolute improvement. We again perform significance analysis as for the PS-Nonuniform set, the results of which we report in Table 7.4b. We can see that some of the changes that were not significant for PS-Nonuniform, especially increasing Θ, have now become significant. This change supports the assumption that the non-significance in these cases for the PS-Nonuniform set is because optimal values have already been achieved for $\Theta = 0.02$.

The last peak shaving instance set, which we evaluate is the PSG set, in which we assume solar generation. As can be seen from Figure 7.7a, the distribution is similar to the PS-Nonuniform set. We can thus assume again that tall jobs have a large impact. However, the absolute values regarding improvement are much better than for PS-Nonuniform. In the most extreme cases, we are able to reduce the residual peaks by almost 80%. Most improvements are evenly distributed between 5% and 50%. This seems plausible, since in the PSG set, peak reduction cannot only be achieved by avoiding concurrent execution of jobs, but also by moving jobs towards the peaks of the solar generation curve. For this set, almost all changes in parameter choice lead to significant improvements. Overall, our approach seems to be able to exploit the benefits of a given generation curve without increasing the computational complexity.



(a) The PSG set with the Y axis indicating the change in peak demand after optimisation.

(b) The OM set with the Y axis indicating the change in overshoot after optimisation. For better readability we removed one outlier at $\Theta = 0.01, \hat{J} = 3$ with a value of approximately 0.5.

Figure 7.7 Results for the PSG and OM sets with one point per instance. The columns are the different settings for $\hat{J}$ and Θ. Colour indicates the remaining MILP gap when the optimisation was stopped.

7.4.5 Evaluation of Overshoot Minimisation

Our last evaluations analyse the overshoot minimisation objective with the OM instance set. This instance set shows the strongest clustering of values so far, indicating with e. g., $\Theta = 0.02$, one can expect the amount of energy to overshoot generation, i. e., the amount of energy that must be bought, to decrease between 6% and 12%. The choice of $\hat{T}$, apparently plays a crucial role for this optimisation problem, while $\hat{J}$ yields only minor improvements. Computational complexity increases slightly with decreasing $\hat{J}$. However, the MILP gaps are drastically smaller than for our peak shaving sets. We solve all instances to at most 4% MILP gap, and in fact, solve most of them to optimality. We again conduct a significance analysis and see that almost all parameter changes result in significant improvements.

7.5 Discussion

We can show that for all examined variations of the FPSP problems, significant improvements can be achieved with respect to the different target metrics, with a relatively small amount of flexibility. As our test data is based on energy consumption data obtained from a small scale factory, we assume our results to apply to real-world scenarios. However, real industrial processes come with more constraints than we were able to respect within the scope of the present chapter. Since the optimisation is based on an MILP framework (which supports additional constraints such as process dependencies etc.), many additional constraints should be straightforward to model.

We discovered that the possible improvements depend a lot on the heterogeneity of the process' power demands. However, even for the heterogeneous instance sets derived from our real-world data, possible improvements were promising.

For the problem variants that assume available generation, we need to choose the amount of generation. While we could obtain a solar generation curve from real-world data, we need to fix the total amount of energy available via generation somewhat arbitrarily. However, we have no reason to believe that our approach works significantly better or worse, if we choose this amount differently. We were able to show that our approach is in fact suitable to reduce the peak residual demand as well as the amount of energy that must be bought from the grid.

Another discrepancy between our test instances and real-world processes is the fact that we assume the power demand of each process to be constant over time. However, we have argued in Section 7.4.1 why this is probably a reasonably close approximation of the motifs we discovered, which is why we believe that we can even use constant-demand jobs as an approximation for many real industrial processes.

We also discover that substantial heterogeneity of power demands, while decreasing possible peak improvements, is beneficial for the complexity of the optimisation problem.

Given all that, most of the time we can optimise instances of realistic sizes, spanning a whole working week with five-minute resolution, within 45 minutes to acceptable MILP gaps. We therefore think that our approach can not just be beneficial in reality, but can also be applied to realistically sized problems.

7.6 Conclusion

We have shown a new framework that can be used to guide flexibilization efforts in industrial processes. Our method starts with consumption data obtained from the current operation of an industrial plant, and it ends with an indication which processes would be most beneficial if they were more flexible. We have shown that our framework can lead to significant improvements and should be applicable in real-world industrial processes.

Regarding our initial question [RQ2], how much demand-side flexibility do we need, our framework helps to understand which processes would benefit most from being flexible. Additionally, our results show that there is currently no need to flexibilise all processes. Starting with only a few small changes in the operation of the machines can already improve the energy consumption. This comparatively little necessary effort gives us hope that more flexibility in industrial processes is achievable and not a daunting prospect for any process manager.

In the future, we can think of several extensions of our approach. While we currently require fixed limits to be set for the amount of new flexibility to create (in terms of $\hat{T}$ and $\hat{J}$), it would be straightforward to allow for a weighting between improvement in peak demand (or overshoot) and the required new flexibility by making $\hat{T}$ and $\hat{J}$ variables and including them in the objective function. However, this requires a reasonable estimate of the (financial) costs of adding flexibility to processes compared to the costs associated with peak demand or overshoot. Also, it would be easy to price the flexibilization on a per-job basis in the objective, to account for some processes to be more costly to make flexible than others. And lastly, incorporating uncertainties into the model should be a future step that would be beneficial for practical relevance.

Summary Part III

In the present part, we have analysed the need for demand-side flexibility in a given system. We have both taken a look at the question if we need flexibility, through investigating proper demand forecasts, as well as the question how much flexibility we need, through introducing a new motif discovery and scheduling based framework.

We have shown that including weather information into forecasting is important and the question if there is a need for any flexibility measures to be taken, can be assessed with more confidence based on the more accurate forecasts. Additionally, we have also seen, that given flexibility information, such as the one obtained through the methods in Part II, allows us to determine which processes, how many process and in what way, should be changed in order to allow more RES providing the grid.

The contributions of this part to the challenges [C3] and [C4] can thus be summarised as:

- We are the first to show that including properly post-processed ensemble weather information significantly improves the forecasting accuracy, especially when dependency structures are taken into account [C3].
- We show based on an eSAX and scheduling framework, that small improvements in flexibility can reduce peak demand or energy overshooting substantially [C4].

Part IV

Balancing Potential and Required Demand-Side Flexibility

Overview Part IV

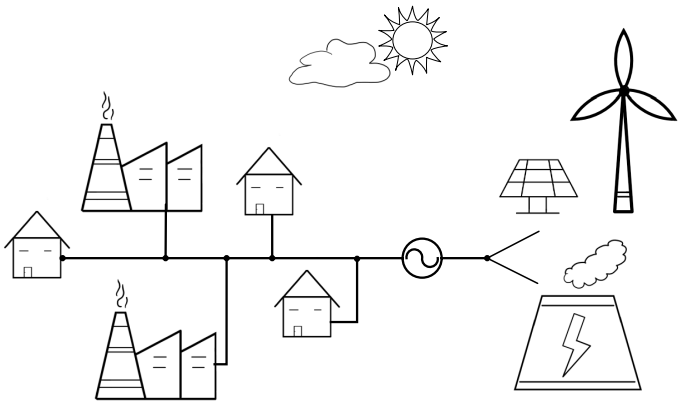


Figure 7.8 Graphical view on Part IV of the present thesis: Balancing Potential and Required Demand-Side Flexibility.

In the previous parts, we have introduced ways to detect flexibility from data and have assessed how much flexibility is needed in a given system. With this information at hand, we now want to balance the potential and required flexibility to design a system that can handle a high share of renewable energy sources. To do so, we want to use different mechanisms. We first introduce theoretical mechanism designs, before describing and training an agent-based model, in which the agents act on a market that is designed based on these mechanisms (Chapter 8). We find that the amount of conventional energy needed to supply the system can be reduced significantly with the right mechanisms at hand, at least on a local market. However, as the flexibility provided by the mechanisms might not be sufficient to completely rely on the renewable energy sources available, Chapter 9 investigates how one could size a centralised energy storage system on distribution grid level from a grid operator's persepctive. We use the pattern detection method from Part II to find the proper size for these energy storage systems.

8 Mechanism Design for Flexibility Markets

More and more research focuses on tackling the need for more flexible resources using demand response. With demand response schemes, suppliers send price signals to consumers and hope that these signals are incentive enough for the consumers to react (e. g. Torriti 2012). However, the price signals have to be very specific, as sending the same price signal to all consumers might result in too many to e. g. shut down or turn on their processes, resulting again in over- or under-capacities. Moreover, the consumer's utility for using energy at a specific point in time might be very different for each consumer and depending on the specific time, making the value of flexibility hard to assess and thus finding the right price signal very challenging (Waczowicz et al. 2016).

In the present chapter, we propose mechanisms to advance the use of flexibility potentials on a small distribution grid. More specifically, we investigate a local energy market inspired by the set-up of a large chemical industrial site, which consists of several small production facilities. Given information of the potential energy flexibility of these facilities, we analyse which mechanisms could be used to make the production managers change their operational decisions from a pure production schedule focus to a more energy usage based focus. It is our aim to change the behaviour of the demand side in such a way, that the energy can be provided mostly by renewable energy sources (RES).

Getting consumers to change their behaviour is not an easy task. One approach to use the flexibility on the demand side is often scheduling the jobs and paying or rewarding the customer for it, e. g. Setlhaolo et al. (2014). Although establishing a central planner is possible, most consumers, regardless whether households or industry, will prefer to keep full control over their consumption. An auction system allows the consumer to decide which information she wants to give to the supplier. Additionally, even if the central planner only gets some information, the consumers need incentives to provide this information.

Parts of this chapter are reproduced from

N. Ludwig, R. Mikut, and V. Hagenmeyer (2018a). "Auction Design to Use Flexibility Potentials in the Energy - Intensive Industry". In: *2018 15th International Conference on the European Energy Market (EEM)*, pp. 1–5. DOI: 10.1109/EEM.2018.8470003

In this light, an auction system can help to reveal price and flexibility information. Recent advances focus on e. g. automation of the auction system, where intelligent agents replace human beings (Ramachandran et al. 2011), or on security issues given an auction design (Wen et al. 2014). However, there are only a few approaches that want to directly enhance flexibility. The authors in Li et al. (2011) are closest to our approach and investigate an auction design for demand response scheduling. They show how a Vickrey auction could work in a household setting with a two-stage process, deciding first how much energy there is to distribute before auctioning of time and loads. They find that the auction is efficient and incentive compatible if there is a reserve price implemented by the utility. To the best of our knowledge, the only authors who explicitly incorporate flexibility in their auction design are the authors in Dauer et al. (2015). However, they only allow for flexibility by allowing specification of a minimum and maximum energy need in their bids.

We investigate the design of two different mechanisms. The first mechanism is based on the flexibility of the energy usage. Each facility can sell its potential to shift the demand, offering e. g. moving a specific job by several hours. The second mechanism is based on the energy demand itself, where the facilities bid for offers made by the supplier to reduce or increase energy at specific times. We then compare both designs theoretically, regarding their effectiveness in achieving the envisioned behaviour, the potential production costs and the costs of implementing such a system. Lastly, we implement a variant of the offering flexibility mechanism and train agents to learn how to act on this market.

The remainder of this chapter is structured as follows. We introduce the mechanisms in Section 8.1, before discussing implications for the energy-intensive industry in Section 8.2. We then present the agent-based model in Section 8.3 and evaluate it on real-world energy consumption data. We conclude the chapter with a discussion of both the theoretical and agent-based aspects in Section 8.4 and conclusion in Section 8.5.

8.1 Mechanisms

In Part II we have shown that there is potential for a more flexible energy usage. However, we cannot assess how much it would cost the facilities to provide the flexibility visible in the data. Therefore, we want to use auctions to reveal the costs for the facilities at a specific moment without having to set a price tag on the costs from the supplier side. The aim of our mechanisms is that the energy system under observation can rely mostly, or only, on renewable energy sources.

The set-up for our energy system is inspired by a large chemical industrial site and can be identified as a local market energy system. We have chosen this simple set-up as it allows us to illustrate the principles we want to show. On this site we have one supplier and N facilities using energy. The supplier can get energy from RES or can produce energy with the help of a conventional power plant (CPP). The supplier could also buy energy from the grid instead of producing it himself, however, we do not investigate this setting for the sake of brevity. We assume that the marginal costs for

using energy produced by RES is zero, $mc_{res} = 0$. In contrast to that, the marginal costs for producing energy with the CPP are constant at $mc_{cpp} = c$.

An energy unit at a specific point in time e_t is either produced by the RES, e_t^{res}, or by the CPP, e_t^{cpp}. We assume that one energy unit is not divisible and $e_t \in P_G(t)$. Thus, any energy unit is either e^{res} or e^{cpp}, but cannot be a combination of both. The supplier has no energy storage options but can shed load and she aims to maximise her profit by maximising the return from distributing the energy on the facilities grid. Any energy, which could be provided by the RES but is not used, is curtailed. Additionally, the facilities on the grid are guaranteed service. Thus, they cannot be cut off the energy sources completely. This assumption is in line with many countries regulations on energy supply, e. g. Germany.[1]

The structure of the system is the same for all our auctions and their variations. Ahead of the considered time period, the supplier publishes the forecast for the supply available by RES and the estimated demand for the industrial site. After a defined bidding time the auction system closes and those offers which help to reduce the energy needed to be produced by the CPP are accepted if the costs associated with them are not higher than the costs associated with the CPP usage.

In the following we introduce two basic designs for flexibility auctions. We describe which bids can be offered as well as who is determined the winner and what the optimisation function for the agents at the auction are. These auctions depend on three main assumptions. The first assumptions is that a shift in one process does not change any other processes. The second is that the supplier has a perfect existing forecast for the demand of the considered time period. The last assumption is that participating in the auction will not cost the facilities anything. In a second step we relax these assumptions and include several adjustments of the basic design which we think are needed for the energy intensive industry setting.

8.1.1 Flexibility

Having described the general setting of the auctions, we explain the flexibility auction in detail. We work with the flexibility definition of Neupane et al. (2014). They describe flexibility as *"the amount of energy and the duration of time to which the device energy profile (energy flexibility) and/or activation time (time flexibility) can be changed."* Considering the facilities bids we can describe flexibility in the following two ways.

1. Energy Flexibility. The facility offers to use the process in a different mode e. g. using 4kW for one hour instead of 2kW for two hours.
2. Time Flexibility. The facility offers to move a job with a predefined energy consumption in time, e. g., the supplier is free to choose the starting time of the job between 10am and 2pm.

1 In Germany this is specified in §36 EnWG.

Given these types, each facility n can offer energy flexibility, time flexibility, or both. As we assume the supplier has no information on the jobs running at the individual facilities, the bids need to contain a lot of information. First, there needs to be information on the time the process would usually start s and its energy profile $\mathbf{e}$. This information helps the supplier to update her forecast of the demand according to the new changes. Then, for the time flexibility offer, we need the offer of a new start time interval with an earliest start time s^*_{min} and latest start time s^*_{max}. Additionally, if the facility can also change the energy intake, the new energy profile $\mathbf{e}^*$ has to be part of the bid as well. Thus, a facility n can offer a sealed bid

$$b_{j,n} = \{\mathbf{e}, \mathbf{e}^*, s, [s^*_{min}; s^*_{max}], \varphi_n\},$$

where φ is the price at which the defined flexibility is offered and j indicates the consecutive number of bids in the current auction period. For the offer above, the supplier would be free to choose using either the process profile $\mathbf{e}$ or $\mathbf{e}^*$ within the new interval $[s^*_{min}; s^*_{max}]$. Each facility can run several processes. They can decide for themselves which of the processes they want to offer flexibility for. Additionally, the information here does not have to be about a single process, it could also be a group of processes, as long as one energy profile, start and end time interval can be given for this group.

Each facility wants to maximise their profits, which results from the output produced $o^n_{e,t}$ minus the costs for producing the output, $c^n_{e,t} = \sum_t e_t \cdot p^e_t - \gamma_e$. Thereby, the sum considers the energy costs and γ_e all other costs, such as labour. The output of each facility is only dependent on the energy intake, while the energy intake depends on the specific price at time t for facility n. The facilities do not care whether their energy is provided by RES or by the CPP.

However, they want to maximise their profits. Thus, a lower energy price is preferable to them as is using the energy whenever they need to. In the auction setting the facility can earn money by selling flexibility. If its flexibility offer b_j is met by the supplier, then $\xi_j = 1$ and it earns φ_j. Thus, the facility wants to maximize its return R^*_{fo}

$$\begin{aligned} \mathrm{R}^*_{fo} &= \max_{e^*,\varphi} \mathrm{R}_{fo}; \\ \mathrm{R}_{fo} &= o_e - \sum_t e^*_t \cdot p^{e^*}_t - \gamma_e + \sum_j \xi_j \varphi_j, \end{aligned} \tag{8.1}$$

where * indicates the optimum. Given rational profit maximising behaviour by the facility, it only offers flexibility in two cases. First, if the change of energy consumption does not result in a change in output or other costs, any offer of φ increases profit. Second, if a change in the energy consumption results in a change in output, the facility offers flexibility as long as the loss is offset by the auction result.

Taking a look at the supplier, we assume that she has a service task to the company. Thus, the energy costs are directly transferred into prices for the customers, but she does not maximise profit. Her optimisation function is thus the reduction of cost and can be written as

$$\begin{aligned} \mathrm{C}^{*}_{fo} &= \min_{p, e^{CPP}} \mathrm{C}_{fo}; \\ \mathrm{C}_{fo} &= c \cdot \sum_{t} e^{CPP}_{t} - \sum_{n} \sum_{t} p^{n}_{t} + \sum_{n} \varphi^{n}, \end{aligned} \tag{8.2}$$

where p^{n}_{t} is the price for facility n to use energy e at time t and $\sum_{n} \varphi^{n}$ is the total amount of payments for the used flexibility.

The supplier gets all the bids for possible flexibility from the facilities ahead of delivering the energy. Given a perfect forecast of the demand for the auctioned period, the supplier can then adjust the demand schedule based on the forecast according to the options made possible through shifts of the processes. Based on the best demand curve, which optimises her utility, the supplier then decides which of the flexibility offers she accepts.

The facilities, whose offered flexibility is used, get paid the price indicated in their bid. If the price they want for the flexibility is higher than the costs for the supplier to use CPP energy, then the flexibility is most likely not used. Note, however, that if the supplier optimises over a longer time frame, individual flexibility payments could be higher than the CPP costs, if it provides an advantage to a later time step. Therefore, it does not help a facility to bid a higher price as this would result in lower chances to be selected. Thus, paying them the indicated price if they win should reveal their true costs for the flexibility. Additionally, as the supplier only takes the flexibility she really needs which changes with the volatile supply of the RES, we expect the facilities to not have any advantages by colluding with each other.

The efficiency of this auction largely depends on the match between flexibility which can be offered by the facilities and the flexibility needed by the supplier. If the flexibility which is offered is at times where the supplier does not need it, the auction clearly fails. Thus, we will consider another auction design in the next section before investigating possibilities to make this auction design more efficient.

8.1.2 Energy

The aim of the second auction remains the same. We want to stay under a supply curve given by renewable energy generation. In this second setting however, the supplier can make offers on when the facilities could increase or reduce their energy consumption. The facilities can then bid for the increase or reduction. Thus a flexibility offer f by the supplier could be e. g. *Reduce the energy intake by 2kW from 8am to 10am*. The facility can then bid on offering this kind of flexibility with a sealed bid stating how much it would cost to supply this energy. Additionally, if it intends to not just reduce but shift the

processes to provide this flexibility, the bid would again include information on the old an new energy profile and start time. The bid of facility n for the flexibility offer f is then

$$b_{f,n} = \{\mathbf{e}, \mathbf{e}^f, s, [s^f_{min}; s^f_{max}], \varphi_{f,n}\}.$$

The optimisation functions R and C remain the same as in Equation (8.1) and Equation (8.2) and the facilities enter the auction in the same two scenarios as before. The only difference being that they change the energy profile not according to what they have specified, but according to the offer made by the supplier. The winner for each offer is the facility which is able to provide the flexibility at the lowest cost. Again, not revealing the true costs would thereby decrease the possibility to be selected and thus not be advantageous for the facility.

In contrast to the auction of Section 8.1.1, the decision making possibilities are limited for the facilities. This limitation could lead to an increase in participation as the facilities do not need to fully evaluate all their processes regarding flexibility but can limit this to the times suggested by the supplier.

8.2 Energy-Intensive Industry

We have introduced two basic auction designs, which can be used to enhance the use of flexibilities in the system with the aim of more energy provided by RES. In the following, we relax the assumptions of Section 8.1.

8.2.1 Dependencies

So far every agent acts individually on the auction market, and we assume that moving one process in a facility does not result in any changes in other facilities. However, in industry, especially if we consider a large industrial site, the facilities usually work together. One facility might depend on the output of another facility. Thus, if the first facility decides to change the output, it might trigger a whole chain reaction in the process steps of other facilities afterwards. We have two concepts, which might help to reduce the problem resulting from highly interdependent agents on our market. The first concept considers a more complex bidding procedure with complete demand curves, while the second additionally investigates the pooling of flexibility.

8.2.2 Complete Demand Curves

If the processes cannot be viewed individually, we can submit demand curves over a specified time horizon. For example, a facility n makes a bid

$$b_n = \{\mathbf{f}(\mathbf{e}), \mathbf{f}^*(\mathbf{e}), \varphi^*_n\},$$

where $\mathbf{f}(\mathbf{e})$ is the usual demand curve and $\mathbf{f}^*(\mathbf{e})$ is the possible new demand curve which can be realised at cost φ_n^*. The bids can also be aggregated, including the offer of m demand curve possibilities $\mathbf{f}^1(\mathbf{e}), \ldots, \mathbf{f}^m(\mathbf{e})$ at varying costs $\varphi_n^1 \ldots, \varphi_n^m$. The optimisation function Equation (8.1) for the facility in the case of a demand flexibility offer then changes to

$$\begin{aligned} \mathrm{R}^*_{dfo} &= \max_{e^*,\varphi} \mathrm{R}_{dfo}; \\ \mathrm{R}_{dfo} &= o_{e,t} - p_t^{e^*} \mathbf{f}(\mathbf{e}^*) + x\varphi_x. \end{aligned} \tag{8.3}$$

The auction winner is determined by which demand curves allow the supplier to use more RES. Her optimisation function Equation (8.2) thus changes to

$$\begin{aligned} \mathrm{C}^*_{dfo} &= \min_{e^{cpp}} \mathrm{C}_{dfo}; \\ \mathrm{C}_{dfo} &= c \cdot \sum_n \mathbf{f^n}(\mathbf{e}) - \sum_n \varphi^n. \end{aligned} \tag{8.4}$$

8.2.3 Pooling of Flexibility

The pooling of flexibility is inspired by Ulbig and Andersson (2015). Instead of each facility bidding by themselves, one could pool together the flexibility from all facilities using each other's products directly. One agent could than offer optimised demand curves based on the flexibility of all those facilities combined. This would facilitate calculations by the supplier as she would not have to take into account any changes of one facility affecting others in a way she cannot predict. Such an agent would also reduce the bidding costs for all facilities as they could share the payment of the pooling agent.

8.2.4 Penalties and Rewards

Participating in the auctions is so far voluntary, thereby we assume that any rational agent participates if it provides him with an additional source of income. However, in industry, additional energy costs can be transferred onto the customers. Thus, participation in the auction only becomes necessary as soon as the price becomes so high that most customers drop out of the market or move to other providers. As it is our aim to improve the usage of RES in the system, we want the facilities to start using the auction system before this happens.

One option to increase the participation in the auction is to introduce penalties. Those penalties should encourage participation in the auction or introducing more flexibility into the existing processes. However, for some facilities it might be easier to provide flexibilities than for others. This would burden those which are inflexible the most, but

would not change their short-term behaviour[2]. Thus, we consider penalties dependent on flexibility information we have gathered beforehand about the facilities.

We have seen in Part II, that given energy consumption data, it is possible to derive some information about the flexibility the facilities can offer. The auction behaviour of the facilities can then be categorised into four groups, which are shown in the matrix in Table 8.1.

Table 8.1 Auction behaviour matrix of the facilities.

	Facilities Action	
Flexibility	Known, Offer	Known, No Offer
	Unknown, Offer	Unknown, No Offer

Depending on which group a facility belongs to, the supplier can penalise or reward the behaviour. For example, facilities belonging to the first column of the above matrix could get a reward e. g. a reduction in overall energy costs, if they participate in the auction regardless of whether their offers are accepted. The facilities optimisation function Equation (8.1) would then become

$$\begin{aligned} \mathrm{R}^{*}_{pr} &= \max_{\alpha,\varphi,e} \mathrm{R}_{pr}; \\ \mathrm{R}_{pr} &= o_{e,t} - c_t + \sum \xi\varphi_{\xi} + \alpha c_t. \end{aligned} \tag{8.5}$$

The penalty or reward is represented by $\alpha \in [-1,1]$. The decision on whether to offer flexibility thus affects the overall energy costs more directly with up to 100 % increase or decrease in price. This should lead any agent on the market to faster enter the auction system, if possible. The supplier can base α on the knowledge of flexibility she already has from the historical data.

8.3 Learning to Act on Flexibility Markets

Given flexibility patterns and information on the electricity supply through renewable energy sources, we can establish an agent-based system to investigate how agents would act on a flexibility market. This market is designed similarly to the mechanism described above. In the following, we describe the agent based environment on which our agents act and the mechanisms that are evaluated. We then explain their learning strategies.

2 It can however give incentives to invest in research to make the processes more flexible in the long run, which we do not investigate here.

8.3.1 Agent Environment

Remember the small electricity system from above, the system has one supplier who wants to use renewable energy sources to supply the grid and several demand agents. Each demand agent runs several processes each day and can generate different demand curves based on these processes. In the most basic setting, each demand agent has

- A set of actions $a \in A$
  In our case each agent has a set of demand curves for a complete day. Each demand curve is one action.
- A set of states $s \in S$
  The demand agent can end up in one of two states, either energy has to be supplied with the conventional power plant or it can be supplied using renewable energy sources only.
- A reward function $R(s, a, s')$
  Given the action chosen, i. e. demand pattern used, the demand agent has to pay energy costs. These costs are higher when the overall supply available from the RES is exceeded.

The initial setting of the system does not include any bidding. At the beginning of each day, the demand agents choose the pattern they want to use. Given this decision, the supplier can either provide the energy from renewable energy sources, if enough is available, or use a conventional power plant. At the end of each day, the supplier calculates the costs for supplying energy at each hour of the day. Costs incur, when the total demand of all demand agents exceeds the available renewable energy. These costs are then distributed equally among all agents that are demanding energy in the specific hour, leading to varying hourly energy prices which are the same for all agents. One could think of a proportional energy cost, however, we leave this to future research. The energy costs thus depend on the collective behaviour of the agents. The agents cannot communicate with each other for a start and cannot observe what the other agents in the system are doing or have done in the past. The only information they observe are their pattern choices and the energy costs for each hour of the day. In order to evaluate how the agents react to different incentives, we adjust the basic setting in two ways, which we introduce in the following.

Offering Flexibility. In a first adjustment to the basic setting, the demand agents can offer the supplier an alternative demand curve for the next day. Instead of just choosing an action they rank their demand curves. The supplier can then choose to either let the demand agent use the initial choice or the second best action. As long as the costs for the demand agents to use the second best option are the same, the supplier randomly picks between them.

Advise from the Supplier. In a second adjustment, the supplier can advise the demand agents on when not to use their energy. Ahead of the demand agents choosing their action, the supplier communicates the six worst hours to demand any energy. The demand agents can then add additional costs to these hours, making any demand curve with energy consumption at these hours less likely to be chosen.

8.3.2 Learning Strategies

Throughout the learning process, the demand agent updates a matrix with cost outcomes given the choice of action. In order to find an optimal policy, the demand agents have an exploration and an exploitation phase. During the initial exploration phase, we set the choice of patterns to be completely random. After this *training phase*, the agents pick their best strategy. In order to account for seasonal and yearly changes in the behaviour, the best strategy is determined through a weighted average, which includes the best strategy overall, the best strategy throughout the past year and the best strategy throughout the past month. A discount factor also regulates when the best absolute strategy is updated, to achieve some stability over time.

Additionally to the weighted averages, the agents also do not cease to explore completely and we add an exploration parameter which determines the probability of the demand agent to pick a random strategy instead of the best performing one even though the training phase is over. This parameter is set to $\alpha = 0.01$, although this is very small it helps in case some parts of the solution space are not explored by the agent during training.

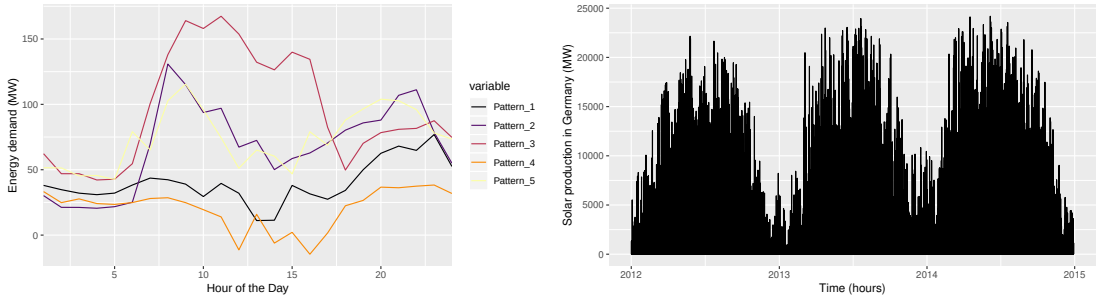
8.3.3 Evaluation

We evaluate the effect of five different numbers of demand agents, four different strategies for the demand and supply agents to act on the market, as well as four different lengths of training steps.

The number of demand agents ranges from 1 to 10000, while the training time steps range from 30 to 365 days. For each number of demand agents and training steps, the same four market setups are evaluated:

1. The demand agents can offer specific demand patterns to the supplier (*Offers*).
2. The supplier can advise the demand agents when it would be worst to use energy (*Advice*).
3. Both above strategies can be used (*Offers & Advice*).
4. None of the above strategies can be used (*None*).

The agents act by choosing one of the demand patterns shown in Figure 8.1a. Two of the patterns include negative energy demand which represents for example being able to feed back energy into the grid from an energy storage system. Another two patterns are more in line with a typical working day pattern, with higher demand in the morning (when most people get up) and in the evening (when most people come home from work). The last pattern, matches a solar production curve most closely with the highest demand between 9am and 3pm.



(a) The five daily pattern choices for each demand agent.

(b) The renewable energy, represented by the solar production in Germany, available to the supplier.

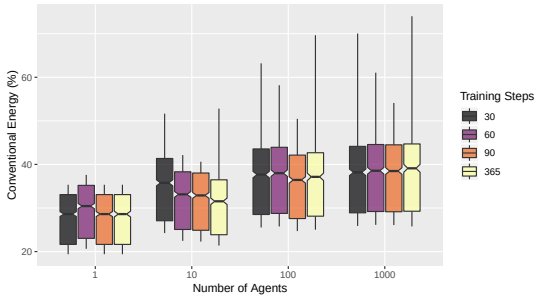
Figure 8.1 The input data for the demand and supply agents in the system.

The supplier has renewable energy from solar power at her disposal. The solar energy curve is taken from the Open Power System Data platform [3] and is the solar production in Germany in MW (see Figure 8.1b). In order to use this curve but fit it to the problem at hand, the solar production curve is scaled to the number of agents in the system.

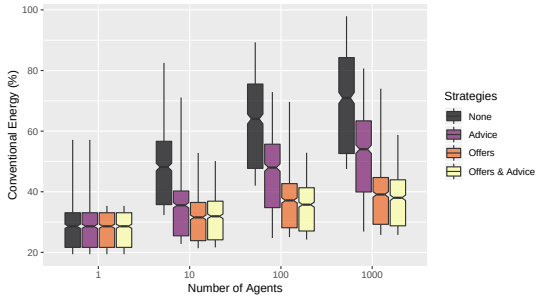
We first evaluate the influence of the different setups on the amount of conventional energy needed, for each number of agents. As can be seen in Figure 8.2b, for a single agent and on average, no additional options to communicate with the supplier are needed. The single agent gets a direct feedback via the energy price and can adjust the behaviour accordingly. As there are no interferences from other actors, the price feedback directly translates to the chosen action. As soon as several other actors join the demand side, their reactions can lead to overshooting the target (if everyone shifts to the same time point, the energy availability problem shifts with them). In this case, advice from the supplier on the six worst hours to consume energy, helps the agents to avoid these times. Even more effective is the strategy of the demand agents to offer several curves and let the supplier decide which choice to make. Combining both approaches reduced the variance and additionally slightly improves the median conventional energy usage. Thus, the *Offers* approach reduces the conventional energy demand the most, with the *Offers & Advice* approach being the strategy with the lowest median conventional energy consumption.

For the evaluation of the setups, the agents were trained using the first 90 days of the solar power time series. Figure 8.2a shows how the energy demand changes when the training is shortened to 30 or 60 days or prolonged to a full year (365 days). As can be seen from the boxplot, the percentage of conventional energy needed to supply the grid does not vary much with the amount of training.

3 https://data.open-power-system-data.org/time_series/2019-06-05



(a) Comparison of the influence of training steps on the resulting conventional energy usage over the four different numbers of demand agents.

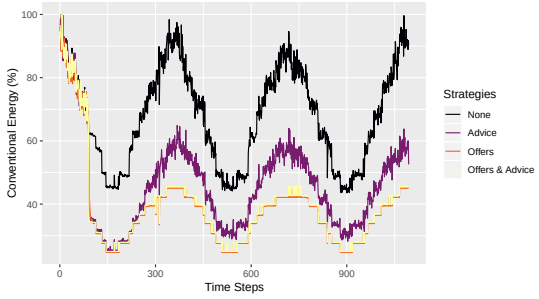


(b) Comparison of the different setup results in conventional energy usage over the four different numbers of demand agents.

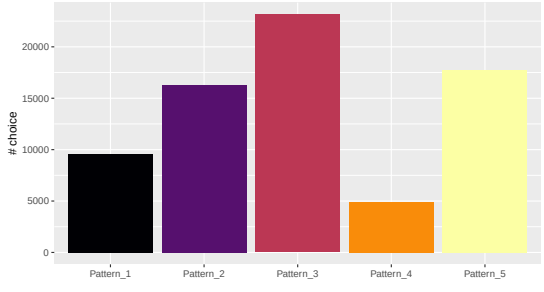
Figure 8.2 The influence of training steps and strategy options on the conventional energy usage.

Over the whole testing period using the real solar data and pattern choices, the agents can significantly reduce their need for conventional energy resources. As can be seen in Figure 8.3a for the system with 1000 demand agents and 90 days of training, the conventional energy needed drops significantly after the training phase if one of the mechanisms is used, and is consistently lower over time for all remaining test days. The lowest intake of conventional energy resources over time is needed when the demand agents can offer their patterns and the supply agent can additionally advise them.

There is not a single best pattern for the demand agents to use throughout the policy implementation phase. However, some patterns are more likely to cost less than others and are thus chosen more frequently. Figure 8.3b shows the number of times all agents combined choose which pattern with the third pattern (the one that resembles the solar curve the closest) being chosen most often. Which is in line with expectations of the agents adapting their behaviour.



(a) The conventional energy usage over time for each strategy with 1000 demand agents as well as a training period of 90 days.



(b) Number of times each pattern was chosen by all agents throughout the training and testing period.

Figure 8.3 Resulting conventional energy demand and pattern choices.

8.4 Discussion

In the present chapter, we have proposed auction systems, which can help to provide more flexibility in the energy grid and evaluated a simplified version in an agent-based simulation. However, although we have relaxed some of the assumptions from the beginning, we still simplify the system we are looking at in some points. We discuss these points in the following.

In the system we assume that energy is produced at zero marginal costs for RES. Although this is realistic, we neglect the fact that one would first need to set up enough RES to supply the industrial site with sufficient energy and those costs would probably be transferred to the consumers. Additionally, we do not investigate the case, where the site can completely rely on RES. We believe that our scenario is more realistic than assuming we have enough RES to supply a large industrial customer at all times, at least without energy storage. Omitting energy storage in our analysis is certainly limiting. The auction prices could however reveal if investing in energy storage or more production capacity is profitable.

Concerning the supplier, we assume that she does not want to make profit but only provides the energy for the sake of usage at the industrial site. This is realistic for a company setting with fair internal accounting, but might not be the case for all suppliers in general.

So far, we assume that the network capacity is sufficient for all possible demand curves realised by a more flexible production. Thus, the network capacity does not limit our options when choosing the right demand curve. Additionally, the supplier has a perfect forecast for the auction period. Including uncertainty in those forecasts might change the suppliers decision which job she wants to move. Future research should thus consider this uncertainty in the suppliers optimisation problem.

The whole auction system of flexibility assumes that none of the facilities are using a pure just-in-time (JIT) production schedule. With such a schedule any possibilities of using flexibility are marginalised and the auction would be useless. However, we believe that the majority of facilities, especially in the energy-intensive industry, do not use JIT production.

Our last point concerns connecting this system to an existing energy market. If we do not assume our site to be an island system, then the supplier is connected to an energy market. If she has her own RES, she could instead of producing the energy herself, buy energy at an energy exchange market, which adds the energy price at the exchange market to her optimisation. If the supplier shares her RES with another supplier, the optimisation function would also become more complex as both suppliers would have to coordinate themselves.

Regarding our evaluation of the mechanisms with the agent-based model, the current set-up has several flaws which we discuss in the following. All demand agents are exactly the same with respect to their action space, learning rate and learning strategies. This probably eases the learning and adjustment and cancels out some otherwise more difficult interactions among the agents. There is also only one supplier which does thus not compete with anyone or does not need to coordinate with another supplier. This greatly simplifies the choices for the demand agents as well, as they do not have to choose where to offer

their flexibility. Additionally, the renewable energy sources data we use is currently only solar power, as the whole set-up is largely simplified this is not an issue regarding the results we obtain. Nevertheless, in a more realistic scenario additional RES would be of interest.

8.5 Conclusion

The present chapter has highlighted the theoretical possibilities, mechanism design and auctions offer to increase the use of flexibility in the system. We presented two basic mechanisms for flexibility in a local energy system: one auction, in which agents can bid to provide flexibility suggested themselves, and a second auction, in which agents can bid on providing flexibility suggested by a third party. We then adjust the basic design to make it compatible with the special requirements in the energy intensive industry. More precisely, we incorporate the possibility of dependencies and penalties or rewards into our auction design.

In the very simple set-up of several demand agents interacting on a local energy market with one supplier, it can be seen that using the given flexibility greatly enhances the energy systems ability to rely on renewable energy sources.

Coupling the price to the renewable energy supply aligns the strategies of the demand agents but does not yet reduce the need for conventional energy sources. However, allowing the supplier to additionally advice the demand agents before they choose their demand curves already decreases the need for conventional energy. Even better is a system, in which the demand agents can offer a second best action and thus explicitly offer flexibility to the supplier. This indirectly coordinates the demand agents and reduces the amount of conventional energy sources needed. The agents generally adapt their behaviour and quickly learn, as there are only a few parameters to optimise for each agent this quick learning is reasonable.

In future work it would be interesting to allow communication among the agents to coordinate on the demand side. Currently, all agents base their decisions on the past data, in future work, adding forecasts, especially probabilistic forecasts, would be great to evaluate in the system.

With the results from this small distribution network in mind, we see that more usage of given flexibility helps to integrate renewable energy sources. However, in the above system, we cannot rely on the RES only, which is probably also true for larger systems. To completely rely on renewable energy sources, we might consider using storage systems from a grid operator's perspective. We thus explore how to size a storage system for a small low-voltage distribution network in the next chapter.

9 Sizing Energy Storage Systems for Distribution Grids

The previous chapter has shown that mechanisms to advance the usage of flexibility can be established theoretically and in a more practical setting. However, the chapter has also shown that these mechanisms might not be enough to rely on RES only. Therefore, we want to investigate storages as further means to enhance the flexibility in the grid. Storages as sources for flexibility have been discussed in many studies (see Qureshi et al. (2011), Soliman and Leon-Garcia (2014), Nguyen et al. (2015), Kwon and Østergaard (2014), and Atzeni et al. (2013, e. g.)). One issue with storages is that their costs rise with their size and one does therefore not usually want to buy a storage which is too large, hence unnecessary costs, nor too small and thus not useful enough. In order to find the proper size and type of an energy storage system for the low-voltage (LV) distribution grid for which we have established the mechanisms before, we employ the pattern recognition method from Part II. This chapter thus connects the data analytics work in this thesis with the more electro-technical subject of storages. While we will see that storages offer flexibility, they are also able to help a grid on a more technical level, balancing out reverse power flows for example. We will henceforth use the notation and language which is most commonly known to electrical engineers.

A high penetration level of renewable energy sources, especially photovoltaic (PV) production in LV distribution networks can lead to several challenges. For instance, the PV peak generation occurs in the middle of the day, when the demand is usually not peaking, especially in residential areas with a high full-time employment rate (Hayn et al. 2014). The generation without sufficient demand during midday can lead to a reverse power flow, which can lead to voltage-rise problems in parts of the LV network. This could be a limiting factor for the PV hosting capacity in LV networks. Furthermore, the variations in solar irradiance (e. g. caused by passing clouds) can cause rapid voltage fluctuations, which may violate the rapid voltage change thresholds, indicated in grid codes such as

Parts of this chapter are reproduced from

S. Karrari, N. Ludwig, V. Hagenmeyer, and M. Noe (2019). "A Method for Sizing Centralised Energy Storage Systems Using Standard Patterns". In: *2019 IEEE Milan PowerTech*. IEEE, pp. 1–6. DOI: 10.1109/PTC.2019.8810658

the EN 50160. Both voltage fluctuations and overvoltages occur due to the fact that LV distribution grids typically have relatively low X/R ratios, which means that active power has a great influence on the voltage, as much as or even greater than reactive power and can be shown using the voltage sensitivity formulation (see Karrari et al. (2019)).

In addition, the evening demand peak coincides with a reduced or even non-existent PV generation. This requires ramping up of large conventional power plants. If many feeders in a power system behave in a similar manner, there might be a deficiency of available power plants with fast ramping capability. Any measures using flexibility, as introduced in the previous chapters, can dampen these effects, however, the effects are most likely not completely eliminated.

Additionally to encouraging a more flexible demand side behaviour, Energy Storage System (ESS) can help to overcome these power challenges. Distribution networks can benefit from ESS for load balancing, peak shaving, voltage regulation, avoiding curtailment, energy arbitrage, loss reduction, expansion deferral and in other areas (*EASE-EERA Energy Storage Technology Development Roadmap* 2017). The usage of a properly sized ESS can also reduce reverse power flow and rapid voltage changes, which increases the dynamic and static PV hosting capacity of the grid. A centralised ESS, in comparison to distributed units, can be beneficial due to a more simplified control system, easy access to substation electrical and SCADA equipment, and availability of utility-owned land. It can also be economical, as it only requires one grid interface and there is no need for a communication infrastructure. Nevertheless, the main question that needs to be addressed when installing an ESS in a power system is choosing the right system, thus deciding, among other characteristics, which capacity, rated power, and technology should be used (Yang et al. 2014; Ru et al. 2013; Makarov et al. 2012).

While there has been a significant amount of work on optimal sizing and allocation of distributed ESS, less attention has been paid to the sizing problem for centralised ESS. Comprehensive reviews of many storage sizing methods can be found in (Hatziargyriou et al. 2016) and (Chauhan and Saini 2014). The ESS sizing algorithms can be broadly summarised into two different groups, depending on whether the algorithm uses real-world load data or not.

If the sizing algorithms do not use real data, then they are mostly based on information about the grid topology and its electrical characteristics, or on load and generation power profiles. The optimal solution in this group is usually reached through optimal power flow (AC, DC or both), or by solving a non-linear optimisation problem. The second group of algorithms then comprises those that use real load and generation data. Algorithms from this group are mostly used for centralised ESS. When data is used to size the storages, then the time resolution and chosen time frame, play a significant role. To investigate the benefits of using ESS in combination with PV generation, most papers use hourly data (Yang et al. 2014; Ru et al. 2013). However, to cover fast changing generation, as for instances caused by clouds passing over the PV, higher resolutions in the area of minutes or seconds may be required (Dugan et al. 2017). The authors in (Oudalov et al. 2007) propose a sizing strategy for a battery ESS with the purpose

of peak shaving using 15-minute data. Unfortunately, they choose one arbitrary day for the sizing algorithm, which can be misleading. (Nagarajan and Ayyanar 2015) propose a convex optimisation problem for the ESS sizing, for which they intentionally convert the 10s-data to 15-minute values to ease computations over a year, eliminating fast fluctuations of the power. The same paper also shows that when using a battery ESS to react immediately to mitigate effects of passing clouds, the lifetime of the battery diminishes over time. (Ru et al. 2013) use a similar approach, but typical load profiles are generated with constant demands for each hour and in (Xiao et al. 2014), 1-minute data is artificially generated from hourly data and the selection of a typical day is not clear.

In the present chapter, we introduce a two-step method for sizing centralised ESS in LV distribution grids. In the first step, instead of using an arbitrary day, we detect a reoccurring daily pattern with the help of Energy Times Series Motif Discovery using Symbolic Aggregated Approximation (eSAX) (as introduced in Chapter 4). This detection is executed with real measurement data from a German LV substation with resolutions from 1 second to 10 minutes. Different resolutions are used to investigate the effect of the data resolution on the sizing results. In the second step, we use a low-pass filter (LPF) designed with the help of Discrete Fourier Transform (DFT) to decompose fast and slow power variations of the daily pattern, which are then assigned to different types of ESS. ESS characteristics including nominal capacity, nominal power, maximum ramp rate and number of full cycles per day, are then derived from the data. For the present chapter, the main tasks of the ESS are assumed to be load balancing and power smoothing, which inherently leads to peak shaving, avoiding reverse power flow and reduction of voltage fluctuations. However, behind-the-meter applications of ESS such as increasing the self-consumption can also be addressed using a similar approach.

The present chapter is structured as follows. In Section 9.1, the methodology is explained in detail. A short discussion regarding the effect of the data resolution on the sizing outcome is provided in Section 9.2. Afterwards, an evaluation of the proposed method is presented in Section 9.3, followed by a conclusion and outlook in Section 9.4.

9.1 Methodology

The first requirement for sizing ESS using time-series data is to select the power profile as the input for the sizing algorithm. To reduce computation time, it is common practice to use an arbitrary day or a worst-case scenario data. However, in the present chapter eSAX is used to detect and generate daily patterns with the highest probability. In the next step, a time-frequency analysis in done before a LPF allocates different power profiles to different types of ESS. The methodology is discussed in detail in the following sections and summarised in Figure 9.1. The whole process is repeated with measurement data at different resolutions, and we compare the results in Section 9.2.

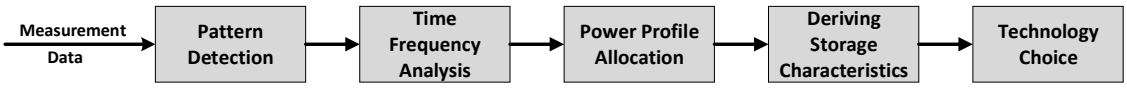


Figure 9.1 Overview of the process from raw measurement time series to storage technology choice. This process is repeated with different data resolution.

9.1.1 Getting Standard Patterns

Sizing of a ESS could be undertaken by using the raw measured data, however, this approach can lead to a very data specific storage size. With only limited data available this might not be suitable, additionally with a large amount of data this approach can be computationally intensive. For this purpose, we want to find a standard consumption pattern and size our storage with regard to this standard pattern. The approximation of the whole time series to a standard pattern helps us to focus on regular shapes of the consumption instead of single occurrences. Two weeks of measurement data with a resolution of 1 second (shown in Figure 9.2), 1 minute and 10 minutes from a German MV/LV substation is used for this purpose.

We use the eSAX algorithm from Part II with an Euclidean distance, as the sequences we compare are of equal length (i. e. one day). The word length w for eSAX changes with the resolution, but always covers a full day, for example $w = 86400$ for the 1 second data (see Section 4.1.4 for more details). The resulting motif covers most of the days in the data set. All these days grouped into the motif are used to describe the standard pattern. This standard pattern is then defined as the 80%-quantile of all the occurrences. This quantile is chosen as the storage does not need to cover the maximum consumption which has ever happened, as the LV costumers are also supported by the grid. However, choosing a higher percentage quantile can easily be done, which will lead to bigger storage units.

The resulting standard patterns are shown in Figure 9.3. As seen, the daily consumption pattern is a typical duck curve, which exhibits characteristics such as reverse power flow, high ramping and peak requirements and also significant power fluctuations during midday. These power profiles are used as the input for the second part of the sizing algorithm.

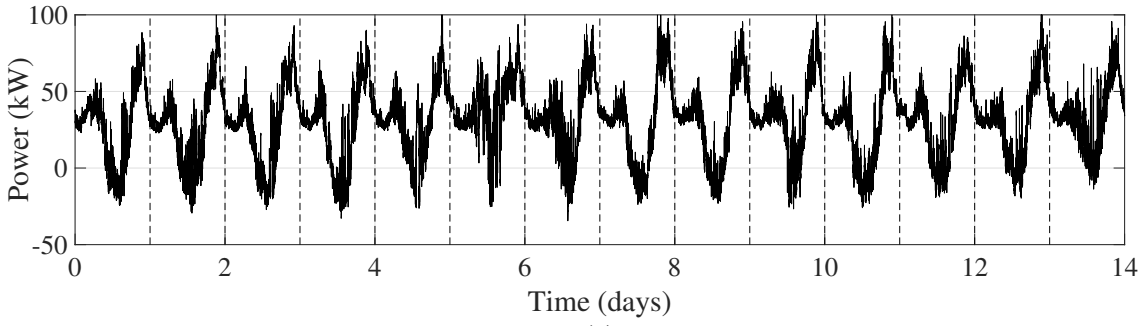
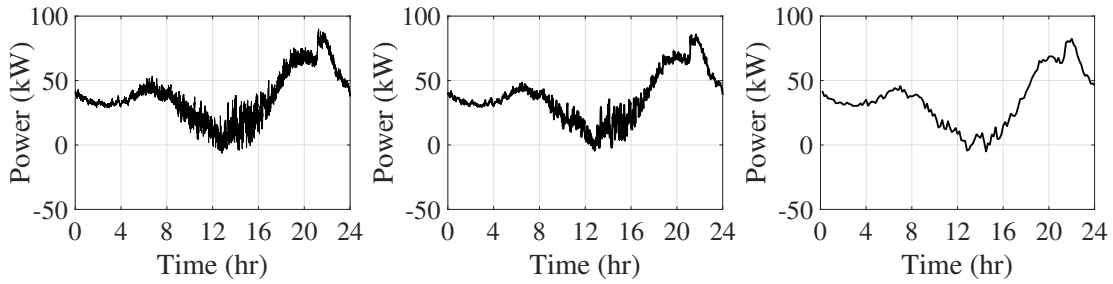


Figure 9.2 Measured power at the substation for two weeks with 1-second resolution.



(a) Data with one second resolution.

(b) Data with one minute resolution.

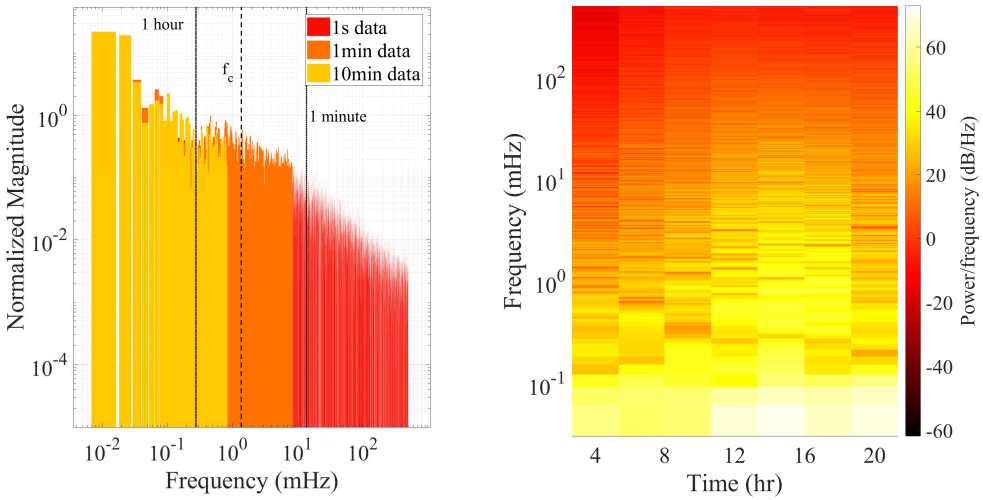
(c) Data with ten minute resolution.

Figure 9.3 The 80% quantile of all motif occurrences for the measurement time series at different temporal resolution.

9.1.2 Time-frequency Analysis

Different ESS technologies operate at different time scales. While batteries have relatively large energy density, deep and frequent cycles can effect their lifetime and capacity. On the other hand, fast ESS such as supercapacitors or flywheels have high power density and can provide a large number of cycles without degradation, but their energy content is limited.

Therefore, a frequency analysis of the power profile can be a good reference point for ESS sizing and technology choices. Using DFT, we can find how much of the power signal lies within each given frequency band. Figure 9.4a shows the frequency spectrum of the power profile for the three different time resolutions. The spectrum starts at 0.01157 mHz (corresponding to 24 hours) and ends at half of the sampling frequency, in line with the Nyquist principle that the spectrum is symmetrical around half of the sampling frequency. As a non-periodic signal, the frequency spectrum of the power profile might be less visually interpretable, compared to a periodic signal. However, we can see frequencies below 16.7 mHz, which corresponds to 1 minute, and 0.278 mHz, which corresponds to 1 hour (shown by vertical lines in Figure 9.4a). The effect of eliminating high-frequency components by using low-resolution data on the sizing outcomes are later discussed in Section 9.2. To visualise at which times the various frequencies are present Figure 9.4b shows the spectrogram of the 1 second signal using short-time DFT for every 3 hours calculated with a *Hamming window* and a 50% overlap. As expected, the higher frequencies are mostly present in the period starting at noon coinciding with the increase in PV production.



(a) Frequency spectrum of the power profile at different resolutions.

(b) Spectrogram of the daily pattern with time resolution of one second.

Figure 9.4 Frequency spectrum and spectrogram of the power profile.

9.1.3 Allocating Power Profiles

Using a LPF to allocate different power signals to different ESS technology is a common practice, in particular for sizing battery-supercapacitor hybrid ESS (Abeywardana et al. 2017). In such systems, it has been shown that a smoother power profile with the help of supercapacitors significantly improves the lifetime of the batteries (Gee et al. 2013). Thereby, the output of the LPF is used for sizing an electrochemical ESS such as Lithium-ion (Li-ion) batteries, which we refer to as *type 1* ESS. The high-frequency components can either be provided by the grid or preferably by a fast ESS such as a supercapacitor or a flywheel, which is referred to as *type 2* ESS. The cut-off frequency of the LPF influences the sizing results. However, despite many efforts there is no clear and agreed-upon method for selecting the cut-off frequency. In this work, the cut-off frequency is selected according to a typical discharge time of type 2 ESS. A LPF with a cut-off frequency of 1.33 mHz ($\mathrm{f_c}$ in Fig. 9.4a), which corresponds to 75 seconds, is selected in this study. It is assumed that the average power over a day can be provided by the grid, as an autonomous operation of the distribution grid is not intended. Fig. 9.5 shows the power profile used for sizing each type of ESS and their combination and how the State of Charge (SOC) of type 1 ESS changes over the daily pattern.

The method discussed here can also be used for sizing other Distributed Energy Resources (DER) such as fuel cells or diesel generators in combination with a type 2 ESS. In this case, the fast ESS allows a more efficient operation of the DER and a more economic sizing (Xiao et al. 2014). However, in this case, the negative power of the DER should be covered in another way, as it cannot consume energy.

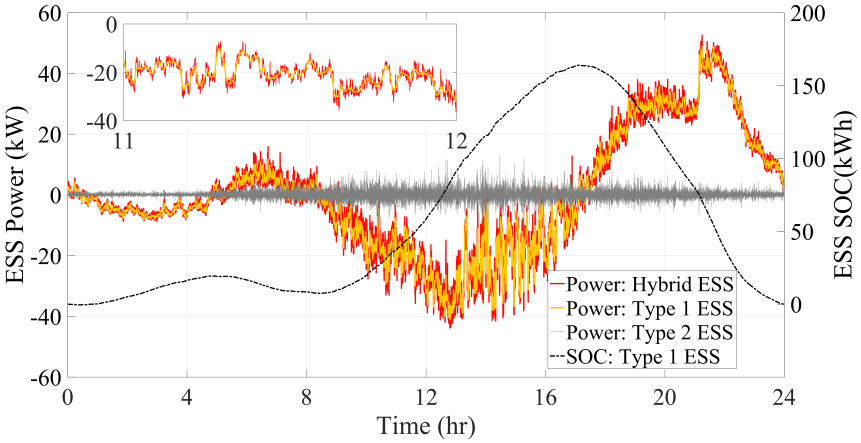


Figure 9.5 Power of both types of ESS and SOC of type 1 ESS generated from the daily pattern.

9.1.4 Deriving Energy Storage Characteristics

In the following, the power profiles derived from the previous section is used for the sizing procedure. The sizing of the ESS is carried out using data with three different resolutions to see the effect of the data resolution on the sizing outcome. Furthermore, in addition to the nominal capacity of the ESS, other characteristics of the ESS such as nominal power, ramp rate, and cycling times per day is derived from the data. Thereby, for both charging and discharging mode of the ESS, an efficiency of 90% is considered.

Nominal capacity. The nominal capacity of an ESS depends on the storage medium itself, which in case of Li-ion batteries depends on the number of cells and in case of flywheels, depends on the inertia and the maximum speed of the rotating mass. The nominal capacity ($\mathrm{E_n}$) for each type of the ESS can simply be calculated by integrating over the power allocated to each one of them and selecting its maximum, i.e.

$$\mathrm{E_n} = \frac{\max(|\int_0^t P(\tau)d\tau|))}{\eta(1-\frac{\mathrm{SOC_{min}}}{\mathrm{SOC_{max}}})}. \tag{9.1}$$

where $P(\tau)$ is the power allocated to the ESS and η is its efficiency. For Li-ion batteries, the lifetime can further be extended by operating it within a certain range of SOC (Nagarajan and Ayyanar 2015), represented by $[\mathrm{SOC_{min}}, \mathrm{SOC_{max}}]$. Therefore, this range is also taken into account when sizing the ESS.

Nominal power. The nominal power ($\mathrm{P_n}$) of the ESS is commonly limited by the power electronics interface of the ESS. Thus, increasing power ratings is generally less expensive than increasing the energy ratings. For both types of ESS, the maximum of the assigned power is chosen, considering both charging and discharging, i.e.

$$\mathrm{P_n} = \max(\max(P(t)), |\min(P(t))|). \tag{9.2}$$

Ramp rate. Ramp rates are defined in a variety of ways, in the present chapter, the ramp rate ($\mathrm{R_n}$) is considered as the maximum change of power between two consecutive points in time. If $\mathrm{T_s}$ is the sampling time of the power signal, the ramp rate is calculated as

$$\mathrm{R_n} = \max(P(t) - P(t - \mathrm{T_s})). \tag{9.3}$$

Cycling times and lifetime. The number of cycling times is defined as the number of times the SOC of the ESS falls below 10% of its maximum, which is a common limit for each technology. In electrochemical batteries such as Li-ion batteries, the lifetime deteriorates with deep cycling, due to cumulative changes of the structure and decomposition of the cells. For such batteries the following charging/discharging behaviour can significantly affect its lifetime (Nagarajan and Ayyanar 2015):

1. High rates of charging/discharging.
2. Frequent variation in the rate of charge/discharge.
3. Leaving the battery for a long time at high SOC.
4. Cycling more than once a day. This simply reduces the calendar life of the battery.

The first two factors can be avoided by allocating the high-frequency components to a type 2 ESS or the grid, as suggested earlier. The third point can be taken into account by choosing a larger battery, as shown in Equation (9.1). The last influencing factor depends on the charging and discharging algorithm and how often the batteries operation is triggered. Thereby, the control algorithm should only allow a battery to be charged and discharged once during a day. We discuss this in more detail in Section 9.3.

9.1.5 Choosing the right technology

The aforementioned characteristics for the two types of ESS are calculated and presented in Table 9.1. In each case, the ESS should have the sufficient capacity, power, ramp rate, and cycling times. For type 1 ESS, an ESS with high energy content is required with energy to power ratio of approximately three. This can easily be provided by Li-ion batteries. Also, redox flow batteries can be used, as the energy and power of redox flow batteries can be scaled independently from each other. For type 2 ESS Flywheel Energy Storage Systems (FESS) and supercapacitors are a good alternative. FESS were proposed as an optimal solution for power smoothing or other applications, where frequent cycling at high powers are required (Hearn et al. 2013; Karrari et al. 2018).

9.2 Discussion on Time Resolution of Data

In the present section, the sizing calculations are carried out using 1-second, 1-minute and 10-minute data. As shown in Table 9.1, there is no significant difference between the sizing results for type 1 ESS, when using 1-minute data and 1-second data. Therefore, it can be concluded that using 1 minute data is sufficient for sizing ESS for applications such as peak shaving and load balancing. Using 10-minute data leads to a slightly bigger storage, however, the required ramp rate cannot be accurately calculated. For power smoothing applications using type 2 ESS, 1-second data is the most advantageous, as these systems often operate within a few seconds. Even using 1-minute data can lead to oversizing the storage and underestimating the ramp rate.

Table 9.1 ESS Characteristics derived from the data.

Type of ESS	**Type 1 ESS**			**Type 2 ESS**	
Data Resolution	1s	1min	10min	1s	1min
Nom. Capacity (kWh)	163.86	162.71	171.9	0.054	2.2
Nom. Power (kW)	50.32	44.75	48.69	12.89	21.27
Ramp Rate (kW/s)	0.54	0.42	0.017	13.93	0.27
Full cycles per day	1	1	1	29	4
Suitable Technology	Li-ion batteries, Flow batteries			Flywheels, Supercapacitors	

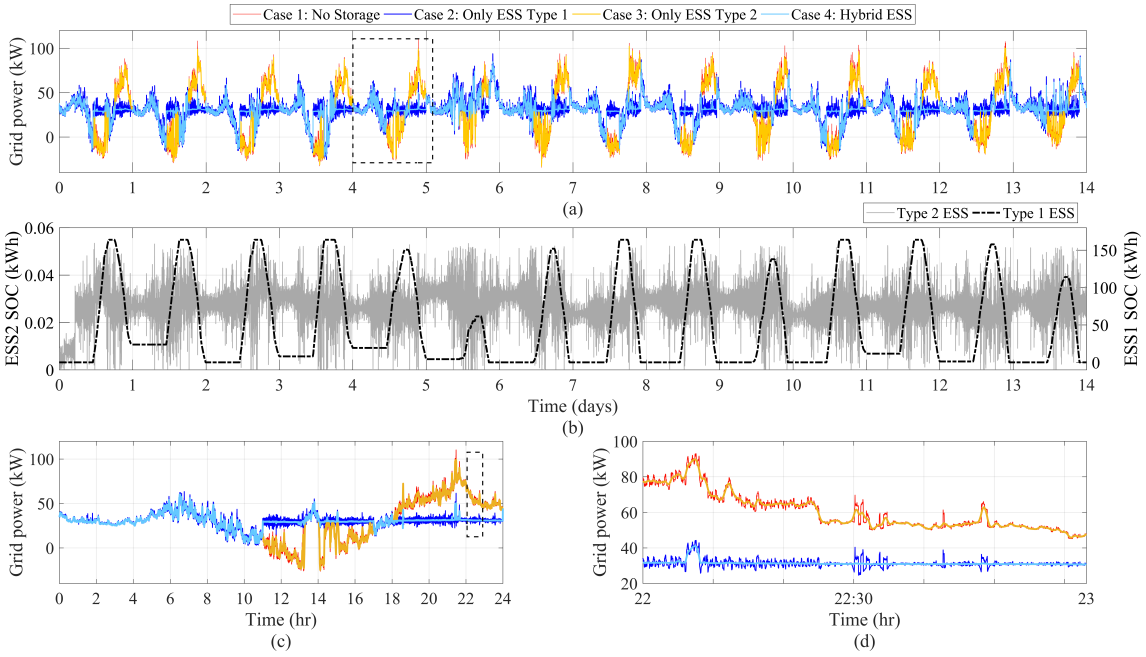


Figure 9.6 (a) Power drawn from the grid in different scenarios during two weeks. (b) SOC of type 1 and type 2 ESS during two weeks. (c) Power drawn from the grid during the 5^{th} day. (d) Power drawn from the grid from 22:00 to 23:00 during the 5^{th} day.

9.3 Evaluation

We evaluate the proposed methodology on two weeks of real world substation measurement data. As the 80% quantile is chosen for the sizing, it is clear that the ESS cannot cover the power variations at all times. However, for the intended purposes of peak shaving, load levelling and power smoothing, the effects of the ESS can be assessed. To fully observe the effect of power smoothing by the type 2 ESS, the following section presents the results using the 1-second data and its corresponding ESS characteristics. However, similar results can be presented using the 1-minute and 10-minute data. The results are obtained using the constraints presented in Table 9.1, meaning that energy content, power, and ramping rate and cycling times of each ESS are kept below calculated values. For the peak shaving implementation, a simple controller triggers the type 1 ESS to store energy from 11:00 to 16:00 and discharges it from 18:00 to 00:00. These times are also selected based on the detected daily pattern (see Figure 9.3). For type 2 ESS, a continuous operation is assumed.

Figure 9.6a compares four different cases to each other. Case 1 shows the originally measured consumption at the substation. Case 2 and case 3 show the power drawn from the grid if only one type of ESS, type 1 or type 2 respectively, is installed at the substation. Case 1 and case 3 are hard to distinguish, however case 3 has a much smoother curve, as shown in Figure 9.6d. Finally, the ideal case, in which both ESS are installed, either separately or in a hybrid structure, is presented as case 4. Analysing the results in Figure 9.6, the following observations can be made:

1. Using type 1 ESS, the evening peak is eliminated for almost everyday (see Figure 9.6a). This results in a reduction of 45.32 kW on average. The exception is the 6^{th} day, in which not enough PV generation is available to charge the type 1 ESS and cover the peak demand. Due to the peak shaving, the grid does not need to provide the power with a high ramp rate in the afternoon. Moreover, the ESS can defer expansions or relief congestion from the grid components, if that is the case. The results for the 5^{th} day are presented separately in Figure 9.6c.
2. During midday, reverse power flow is either avoided or its duration and power is significantly reduced. This can potentially increase the static hosting capacity of the LV grid for PV installations and avoids PV curtailment due to overvoltage issues. However, if a PV system is installed far away from the storage installation place, it may still require local voltage compensation.
3. As shown in Figure 9.6d, type 2 ESS reduces short-time power fluctuations. The maximum change in power during the whole period is reduced by 35.2%, from 18.8 kW/s to 12.2 kW/s, while on average it is down to 54.38%. This will increase the dynamic hosting capacity of the LV grid, as the number of rapid voltage changes due to the active power variations, such as the ones caused by passing clouds, are reduced. Moreover, in case both ESS are in operation, the power drawn from the grid is as smooth and flat as possible, taking into account the ESS constraints, such as their ramp rate.

4. By using type 2 ESS, the power drawn by or injected by type 1 ESS has significantly less variations, which can improve the lifetime of such systems significantly.
5. Energy, power and ramp rate for both types of ESS are kept within the calculated values and while type 1 ESS only has one cycle per day, type 2 ESS experiences numerous cycles per day, as expected.

9.4 Conclusion

In the present chapter, a new method for sizing centralised ESS is introduced. We first detect a reoccurring daily pattern with high probability using eSAX from Part II, before breaking this pattern down to high- and low-frequency components for sizing two types of ESS with specific applications. For each type of ESS, nominal capacity, nominal power, ramp rate, number of cycles per day and most suitable technology is derived from the data. In addition, we investigate the effects of using measurement data with different resolutions on the sizing outcome. For applications such as peak shaving, using 1 minute sampled data is still adequate, while using 10-minute sampled data can lead to minor oversizing and an inaccurate ramp rate. However, using 1 second data for power smoothing applications seems mandatory. The sizing outcomes are evaluated using 14 days of real measurement data and we show that ESS with characteristics derived from a detected daily consumption pattern can effectively reduce peak demand, reverse power flow and voltage fluctuations over the whole period.

For future work, a cost-benefit analysis and other factors influencing the sizing of the ESS, such as a more detailed modelling of the ESS and its ageing process, can be added to the methodology.

Summary Part IV

This part makes contributions to the discussion on how to advance the usage of flexibility in a given grid and thus balancing potential and required flexibility, hence answering [RQ3]. We have introduced theoretical mechanism designs which encourage the usage of flexibility, even for the energy-intensive industry. Furthermore, one of the theoretically proposed mechanisms was implemented in an agent-based simulation system of a low-voltage distribution grid. We have seen that in such a system offering flexibility in the form of alternative demand curves reduces the need for conventional energy, while leaving the consumers with their privacy intact, as they do not need to allow a centralised control system. However, we have also seen that these mechanisms might not be enough to allow the system to rely on RES completely. For this case we have investigated storages. More specifically, we have introduced a storage sizing algorithm which can – based on found patterns in historical consumption data – efficiently choose an energy storage system which not only allows for more flexibility but also stabilises the grid and avoids common electro-technical remedies such as reverse power flow.

The contributions to challenges [C5] and [C6] can thus be summarised as:

- We introduce theoretical auction designs that enhance the usage of flexibility potentials in a given energy grid [C5].
- We show by an agent-based simulation that agents can learn to use their flexibility and reduce the need for conventional energy sources [C5].
- We show that sizing storages with the help of eSAX (from Part II) is efficient and optimal storage parameters can be derived. These properly sized centralised storages increase the flexibility of a system while contribution to grid stability [C6].

Part V

Conclusions

10 Summary

It has almost become folklore in the energy community that demand-side flexibility is a key component to integrate more RES into the energy system. With a changing energy system that relies more and more on RES, a change in how we understand energy systems seems to be necessary as well. Demand-side flexibility is one component which can foster this change, together with transmission extensions and storages. Shifting the focus thus from changing the supply to changing the demand is one option to ease the pathway to a more sustainable future energy system. In the present thesis, we have discussed demand-side flexibility from a purely data-driven perspective, answering three key research questions and contributing to seven challenges resulting from these questions.

The first research question *How much demand-side flexibility is potentially available?* is approached from a modelling and pattern recognition perspective in Part II. We have developed a modelling framework that can describe the flexibility of any household or industrial process. We have also shown that with this modelling framework at hand, we can define jobs which can be scheduled to investigate for example peak shaving or overshoot minimisation. In order to describe the flexibility of processes with the modelling framework, we need very specific and detailed information. We introduce eSAX to obtain this information from any historical energy time series. A special focus is thereby on the detection of motifs, even if the processes under investigation are of unequal length. The application of eSAX and the modelling framework to real-world industrial electricity time series has shown that we can find motifs, can infer flexibility from the occurrences within each motif and use scheduling algorithms with the found motifs as input. We have also used the flexibility information to generate a large scale real-world dataset with flexible instances, to help foster more research on industrial demand-side flexibility. The eSAX approach is used throughout the thesis to extract flexibility information from time series.

The second research question *How much demand-side flexibility is required?* was investigated in Part III from a forecasting and scheduling perspective. We have shown that assessing how much flexibility is required to stabilise an energy system requires forecasts for demand and supply. These forecasts help to know if any flexibility is actually needed. We see that it is essential to include weather forecasts into the models and show that properly post-processing these weather forecasts significantly improves the forecasting accuracy, especially when the dependency structures among the different weather variables are taken into account. Additionally, with the help of new method based

on eSAX and scheduling, we analyse what amount of change is needed in a given system to achieve goals such as peak shaving. We find that small changes in the operation of processes already reduce peak demand and energy overshooting.

In Part IV, the third research question *What incentives can increase the usage of the available flexibility, when it is needed?* addresses mechanisms which help to harness the potential for flexibility when it is required. If we have an answer to the first and the second research question, we have all the information needed to balance the potential and required flexibility. However, incentives are needed for the consumers to act at the right moments in the right amount. We have therefore introduced new theoretical auction designs that enhance the usage of flexibility potentials in a given energy grid. Simulating such a system with agents learning from their past behaviour also shows that the need for conventional energy sources can be reduced significantly with the right mechanisms at hand. We also find that this need is not reduced to zero and thus investigate centralised storage solutions at the substation level from a network operator's perspective. We use eSAX and find that we can derive storage characteristics from the motifs that help find proper sizes and technologies for a low-voltage distribution grid.

11 Outlook

This thesis analyses the demand-side flexibility of a given system from three different data-driven perspectives, evaluating the potential and required demand-side flexibility as well as balancing the two. Answering the three research questions from Section 1.1, we use methods from supervised and unsupervised machine learning, as well as reinforcement learning. In the following, we want to highlight possible further research directions for each of the research questions and additionally for some of the methods used as well.

Costs of flexibility. While we highlighted that there is potential flexibility, and we have also seen that small changes in operation can have an impact on the system, we have not evaluated the costs for providing this potential. This lack of including costs is mainly because no such information was available for the given data sets. However, for future work assessing the costs for flexibility is essential to not only be able to evaluate the potential but also what is economical. From a data-driven perspective, one could assume that what is less expensive is executed more often, and one could thus, for example, assume costs for the different processes according to how often the corresponding motifs occur.

Motif Discovery. So far, the motif discovery algorithm relies on distance measures on the real-valued time series, as we only detect the candidate motifs on the letter approximation of the time series. This leaves the question open what other measures could be evaluated on the words. It seems interesting to investigate what other similarity measures one could use from language processing, primarily focusing on measures that can compare words of different length.

Active learning. The biggest issue when determining the potential flexibility from data is the evaluation. Without an expert at hand evaluating the motifs is difficult and so far only done through evaluating their usefulness in further applications such as scheduling or storage sizing. Approaches such as active learning, including experts into the training phase of the algorithm, might work to detect more accurate motifs without the need to label and characterise every subsequence in the time series.

Non-parametric post-processing. If the mismatch is forecast directly, or the supply is forecast, then the parametric assumptions we make in Chapter 6 are limiting for the analysis. In future work, it might thus be worth investigating non-parametric approaches

for the post-processing of energy or weather forecasts. Additionally, we have used straightforward forecasting methods as the focus was on the post-processing. Thus the question arises how more complex methods including probabilistic deep learning behave.

Agent interaction on flexibility markets. Concerning the agent-based simulation of a flexibility market, several aspects can be improved and analysed. One path that seems most promising is allowing for communication and interaction among the agents. Both would leave more of the decision making explicitly on the demand side and might reduce the need for intervention from the network operator.

Uncertainty in flexibility markets. As we have seen, supply and demand forecasts are not perfect, and thus probabilistic forecasts are essential. So far, the component of uncertain supply from RES and the uncertainty of a mismatch between demand and supply is not taken into account when analysing agents that act on flexibility markets. Different types of risk-takers among the agents might lead to interesting dynamics in a flexibility market, where the mismatch for tomorrow is unclear.

Overall, the purely data-driven approach to demand-side flexibility in this thesis has highlighted major aspects. First, there is flexibility potential in industrial processes, and even small changes in how processes are run can have significant effects on the possibility to mostly rely on RES to supply the energy system. These potentials can be assessed using identified motifs from historical consumption data only. Additionally, properly post-processed forecasting that includes uncertain weather information can contribute to knowing if and when any flexibility is needed. With knowledge on flexibility potentials and mismatch times at hand, it is possible to find and evaluate mechanisms that balance the system. Thus, with the tools introduced in this thesis, historical time series on the operation of the energy system, as well as weather information is sufficient to analyse the demand-side flexibility in the system and its potential to allow for more RES. Hence, this thesis establishes new data-driven tools which enable in a new way the decision-making process of a system operator, facility manager or politician when it comes to investigating demand-side flexibility without extensive a priori knowledge of the system.

Bibliography

Abeywardana, D. B. W., B. Hredzak, V. G. Agelidis, and G. D. Demetriades (2017). *Supercapacitor sizing method for energy-controlled filter-based hybrid energy storage systems*. In: *IEEE Transactions on Power Electronics*, Vol. 32, No. 2, pp. 1626–1637. DOI: 10.1109/TPEL.2016.2552198 (cit. on p. 126).

Al-Yahyai, S., Y. Charabi, and A. Gastli (2010). *Review of the use of Numerical Weather Prediction (NWP) Models for wind energy assessment*. In: *Renewable and Sustainable Energy Reviews*, Vol. 14, No. 9, pp. 3192–3198. DOI: 10.1016/j.rser.2010.07.001 (cit. on p. 66).

Alizadeh, M., A. Scaglione, A. Applebaum, G. Kesidis, and K. Levitt (2015). *Reduced-order load models for large populations of flexible appliances*. In: *IEEE Transactions on Power Systems*, Vol. 30, No. 4, pp. 1758–1774. DOI: 10.1109/TPWRS.2014.2354345 (cit. on pp. 5, 20).

Allerding, F., M. Premm, P. K. Shukla, and H. Schmeck (2012). *Electrical Load Management in Smart Homes Using Evolutionary Algorithms*. In: *EvoCOP*. Vol. 7245. Lecture Notes in Computer Science. Berlin: Springer, pp. 99–110. DOI: 10.1007/978-3-642-29124-1\textunderscore (cit. on pp. 20, 24).

Ambrosius, M., V. Grimm, C. Solch, and G. Zottl (2016). *Investment incentives for flexible energy consumption in the industry*. In: *13th International Conference on the European Energy Market (EEM)*. IEEE, pp. 1–5. DOI: 10.1109/EEM.2016.7521234 (cit. on p. 85).

Ardakanian, O., Y. Yuan, R. Dobbe, A. von Meier, S. Low, and C. Tomlin (2017). *Event detection and localization in distribution grids with phasor measurement units*. In: *2017 IEEE Power Energy Society General Meeting*, pp. 1–5. DOI: 10.1109/PESGM.2017.8273895 (cit. on p. 85).

Ashok, S. (2006). *Peak-load management in steel plants*. In: *Applied Energy*, Vol. 83, No. 5, pp. 413–424. DOI: 10.1016/j.apenergy.2005.05.002 (cit. on pp. 21, 24, 49).

Ashok, S. and R. Banerjee (2000). *Load-management applications for the industrial sector*. In: *Applied Energy*, Vol. 66, No. 2, pp. 105–111. DOI: 10.1016/S0306-2619(99)00125-7 (cit. on pp. 21, 24).

Atzeni, I., L. G. Ordonez, G. Scutari, D. P. Palomar, and J. R. Fonollosa (2013). *Demand-Side Management via Distributed Energy Generation and Storage Optimization*. In: *IEEE Transactions on Smart Grid*, Vol. 4, No. 2, pp. 866–876. DOI: 10.1109/TSG.2012.2206060 (cit. on p. 121).

Baran, S. and S. Lerch (2018). *Combining predictive distributions for the statistical post-processing of ensemble forecasts*. In: *International Journal of Forecasting*, Vol. 34, No. 3, pp. 477–496. DOI: 10.1016/j.ijforecast.2018.01.005 (cit. on p. 66).

Barth, L. F. J. (2020). *Scheduling Algorithms for the Smart Grid*. Dissertation. Karlsruhe, Germany: Karlsruhe Institute of Technology. DOI: 10.5445/IR/1000122593 (cit. on pp. 86, 95).

Barth, L., V. Hagenmeyer, N. Ludwig, and D. Wagner (2018a). *How much demand side flexibility do we need? Analyzing where to exploit flexibility in industrial processes*. In: *Proceedings of the Ninth International Conference on Future Energy Systems - e-Energy '18*. ACM Press, pp. 43–62. DOI: 10.1145/3208903.3208909 (cit. on pp. 83, 84).

Barth, L., N. Ludwig, E. Mengelkamp, and P. Staudt (2018b). *A comprehensive modelling framework for demand side flexibility in smart grids*. In: *Computer Science - Research and Development*, Vol. 33, No. 13, pp. 1865–2042. DOI: 10.1007/s00450-017-0343-x (cit. on pp. 19, 50, 95).

Beier, J., S. Thiede, and C. Herrmann (2017). *Energy flexibility of manufacturing systems for variable renewable energy supply integration: Real-time control method and simulation*. In: *Journal of Cleaner Production*, Vol. 141, pp. 648–661. DOI: 10.1016/j.jclepro.2016.09.040 (cit. on p. 5).

Ben Bouallègue, Z., T. Heppelmann, S. E. Theis, and P. Pinson (2016). *Generation of Scenarios from Calibrated Ensemble Forecasts with a Dual-Ensemble Copula-Coupling Approach*. In: *Monthly Weather Review*, Vol. 144, No. 12, pp. 4737–4750. DOI: 10.1175/MWR-D-15-0403.1 (cit. on p. 66).

Bischof, S., H. Trittenbach, M. Vollmer, D. Werle, T. Blank, and K. Böhm (2018). *HIPE - An Energy-Status-Data Set from Industrial Production*. In: *Proceedings of the Ninth International Conference on Future Energy Systems - e-Energy '18*. ACM Press, pp. 599–603. DOI: 10.1145/3208903.3210278 (cit. on pp. 44, 45, 49, 86).

Bolwig, S., G. Bazbauers, A. Klitkou, P. D. Lund, A. Blumberga, A. Gravelsins, and D. Blumberga (2019). *Review of modelling energy transitions pathways with application to energy system flexibility*. In: *Renewable and Sustainable Energy Reviews*, Vol. 101, pp. 440–452. DOI: 10.1016/j.rser.2018.11.019 (cit. on p. 5).

Bossavy, A., R. Girard, and G. Kariniotakis (2013). *Forecasting ramps of wind power production with numerical weather prediction ensembles*. In: *Wind Energy*, Vol. 16, No. 1, pp. 51–63. DOI: 10.1002/we.526 (cit. on p. 66).

Breiman, L. (2001). *Random Forests*. In: *Machine Learning*, Vol. 45, No. 1, pp. 5–32. DOI: 10.1023/A:1010933404324 (cit. on p. 77).

Buhler, J. and M. Tompa (2002). *Finding motifs using random projections*. In: *Journal of computational biology : a journal of computational molecular cell biology*, Vol. 9, No. 2, pp. 225–242. DOI: 10.1089/10665270252935430 (cit. on pp. 36, 37, 40).

Capozzoli, A., M. S. Piscitelli, and S. Brandi (2017). *Mining typical load profiles in buildings to support energy management in the smart city context*. In: *Energy Procedia*, Vol. 134, pp. 865–874. DOI: 10.1016/j.egypro.2017.09.545 (cit. on p. 6).

Castro, P., H. Matos, and A. Barbosa-Póvoa (2002). *Dynamic modelling and scheduling of an industrial batch system*. In: *Computers & Chemical Engineering*, Vol. 26, No. 4-5, pp. 671–686. DOI: 10.1016/S0098-1354(01)00792-X (cit. on p. 24).

Chauhan, A. and R. P. Saini (2014). *A review on Integrated Renewable Energy System based power generation for stand-alone applications: Configurations, storage options, sizing methodologies and control*. In: *Renewable and Sustainable Energy Reviews*, Vol. 38, No. December, pp. 99–120. DOI: 10.1016/j.rser.2014.05.079 (cit. on p. 122).

Chen, C.-C., H.-H. Juan, M.-Y. Tsai, and H. H.-S. Lu (2018). *Unsupervised Learning and Pattern Recognition of Biological Data Structures with Density Functional Theory and Machine Learning*. In: *Scientific reports*, Vol. 8, No. 557, pp. 1–11. DOI: 10.1038/s41598-017-18931-5 (cit. on p. 5).

Chicco, G. (2012). *Overview and performance assessment of the clustering methods for electrical load pattern grouping*. In: *Energy*, Vol. 42, No. 1, pp. 68–80. DOI: 10.1016/j.energy.2011.12.031 (cit. on p. 37).

Chiu, B., E. Keogh, and S. Lonardi (2003). *Probabilistic discovery of time series motifs*. In: *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, pp. 493–498. DOI: 10.1145/956750.956808 (cit. on pp. 36, 37, 40).

Chou, M. C., G. A. Chua, and C.-P. Teo (2010). *On range and response: Dimensions of process flexibility*. In: *European Journal of Operational Research*, Vol. 207, No. 2, pp. 711–724. DOI: 10.1016/j.ejor.2010.05.038 (cit. on p. 5).

Clark, M., S. Gangopadhyay, L. Hay, B. Rajagopalan, and R. Wilby (2004). *The Schaake Shuffle: A Method for Reconstructing Space–Time Variability in Forecasted Precipitation and Temperature Fields*. In: *Journal of Hydrometeorology*, Vol. 5, No. 1, pp. 243–262. DOI: 10.1175/1525-7541(2004)005<0243:TSSAMF>2.0.CO;2 (cit. on pp. 66, 73).

Coninck, R. DE and L. Helsen (2013). *Bottom-up quantification of the flexibility potential of buildings*. In: *13th Conference of International Building Performance Association* (cit. on p. 6).

Contreras-Ocaña, J. E., M. A. Ortega-Vazquez, D. Kirschen, and B. Zhang (2018). *Tractable and Robust Modeling of Building Flexibility Using Coarse Data*. In: *IEEE Transactions on Power Systems*, Vol. 33, No. 5, pp. 5456–5468. DOI: 10.1109/TPWRS.2018.2808223 (cit. on p. 6).

Dall'Anese, E., P. Mancarella, and A. Monti (2017). *Unlocking Flexibility: Integrated Optimization and Control of Multienergy Systems*. In: *IEEE Power and Energy Magazine*, Vol. 15, No. 1, pp. 43–52. DOI: 10.1109/MPE.2016.2625218 (cit. on p. 5).

Dauer, D., P. Karaenke, and C. Weinhardt (2015). *Load Balancing in the Smart Grid: A Package auction and Compact Bidding Language*. In: *Proceedings of the International Conference on Information Systems - Exploring the Information Frontier, ICIS 2015, Fort Worth, Texas, USA*. Association for Information Systems (cit. on p. 108).

Denholm, P., E. Ela, B. Kirby, and M. Milligan (2010). *The role of energy storage with renewable electricity generation*. Technical Report NREL/TP-6A2-47187 (cit. on p. 19).

D'hulst, R., W. Labeeuw, B. Beusen, S. Claessens, G. Deconinck, and K. Vanthournout (2015). *Demand response flexibility and flexibility potential of residential smart appliances: Experiences from large pilot test in Belgium*. In: *Applied Energy*, Vol. 155, pp. 79–90. DOI: 10.1016/j.apenergy.2015.05.101 (cit. on pp. 5, 85).

Du, P. and N. Lu (2011). *Appliance Commitment for Household Load Scheduling*. In: *IEEE Transactions on Smart Grid*, Vol. 2, No. 2, pp. 411–419. DOI: 10.1109/TSG.2011.2140344 (cit. on p. 20).

Dugan, R. C., J. A. Taylor, and D. Montenegro (2017). *Energy Storage Modeling for Distribution Planning*. In: *IEEE Transactions on Industrial Applications*, Vol. 53, No. 2, pp. 954–962. DOI: 10.1109/REPC.2016.11 (cit. on p. 122).

EASE-EERA Energy Storage Technology Development Roadmap (2017). https://eera-es.eu/wp-content/uploads/2016/03/EASE-EERA-Storage-Technology-Development-Roadmap-2017-HR.pdf. Accessed on: 28.04.2020 (cit. on p. 122).

Eid, C., P. Codani, Y. Perez, J. Reneses, and R. Hakvoort (2016). *Managing electric flexibility from Distributed Energy Resources: A review of incentives for market design*. In: *Renewable and Sustainable Energy Reviews*, Vol. 64, pp. 237–247. DOI: 10.1016/j.rser.2016.06.008 (cit. on p. 5).

Ester, M., H.-P. Kriegel, J. Sander, and X. Xu (1996). *A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise*. In: *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*. KDD'96. AAAI Press, pp. 226–231 (cit. on p. 52).

Fan, C. (2015). *TSMining: Mining Univariate and Multivariate Motifs in Time-Series Data*. R package version 1.0 (cit. on p. 42).

Farinaccio, L. and R. Zmeureanu (1999). *Using a pattern recognition approach to disaggregate the total electricity consumption in a house into the major end-uses*. In: *Energy and Buildings*, Vol. 30, No. 3, pp. 245–259. DOI: 10.1016/S0378-7788(99)00007-9 (cit. on p. 36).

Fehrenbach, D., E. Merkel, R. McKenna, U. Karl, and W. Fichtner (2014). *On the economic potential for electric load management in the German residential heating sector–An optimising energy system model approach*. In: *Energy*, Vol. 71, pp. 263–276. DOI: 10.1016/j.energy.2014.04.061 (cit. on p. 20).

Feldmann, K., M. Scheuerer, and T. L. Thorarinsdottir (2015). *Spatial Postprocessing of Ensemble Forecasts for Temperature Using Nonhomogeneous Gaussian Regression*. In: *Monthly Weather Review*, Vol. 143, No. 3, pp. 955–971. DOI: 10.1175/MWR-D-14-00210.1 (cit. on p. 66).

Felice, M. DE, A. Alessandri, and P. M. Ruti (2013). *Electricity demand forecasting over Italy: Potential benefits using numerical weather prediction models*. In: *Electric Power Systems Research*, Vol. 104, pp. 71–79. DOI: 10.1016/j.epsr.2013.06.004 (cit. on p. 66).

Feuerriegel, S. and D. Neumann (2016). *Integration scenarios of Demand Response into electricity markets: Load shifting, financial savings and policy implications*. In: *Energy Policy*, Vol. 96, pp. 231–240. DOI: 10.1016/j.enpol.2016.05.050 (cit. on pp. 4, 85).

Fink, J., J. L. Hurink, and A. Molderink (2014). *Mathematical modelling of devices and flows in energy systems*. https://ktiml.mff.cuni.cz/Ã\protect\unhbox\voidb@x\protect\penalty\@M\fink/publication/flow.pdf. Accessed on 28.04.2020 (cit. on p. 24).

Förderer, K., M. Ahrens, K. Bao, I. Mauser, and H. Schmeck (2018). *Towards the Modeling of Flexibility Using Artificial Neural Networks in Energy Management and Smart Grids*. In: *Proceedings of the Ninth International Conference on Future Energy Systems - e-Energy '18*. ACM Press, pp. 85–90. DOI: 10.1145/3208903.3208915 (cit. on p. 6).

Frazzetto, D., B. Neupane, T. B. Pedersen, and T. D. Nielsen (2018). *Adaptive User-Oriented Direct Load-Control of Residential Flexible Devices*. In: *Proceedings of the Ninth International Conference on Future Energy Systems - e-Energy '18*. ACM Press, pp. 1–11. DOI: 10.1145/3208903.3208924 (cit. on p. 6).

Froehlich, J., E. Larson, S. Gupta, G. Cohn, M. Reynolds, and S. Patel (2011). *Disaggregated End-Use Energy Sensing for the Smart Grid*. In: *IEEE Pervasive Computing*, Vol. 10, No. 1, pp. 28–39. DOI: 10.1109/MPRV.2010.74 (cit. on p. 36).

Gärttner, J. (2016). *Group Formation in Smart Grids: Designing Demand Response Portfolios*. Dissertation. Karlsruhe Institute of Technology. DOI: 10.5445/IR/1000059424 (cit. on p. 21).

Gee, A. M., F. V. P. Robinson, and R. W. Dunn (2013). *Analysis of Battery Lifetime Extension in a Small-Scale Wind-Energy System Using Supercapacitors*. In: *IEEE Transactions on Energy Conversion*, Vol. 28, No. 1, pp. 24–33. DOI: 10.1109/TEC.2012.2228195 (cit. on p. 126).

Gneiting, T. (2014). *Calibration of medium-range weather forecasts*. In: *European Centre for Medium-Range Weather Forecasts* (cit. on p. 66).

Gneiting, T. and A. E. Raftery (2007). *Strictly Proper Scoring Rules, Prediction, and Estimation*. In: *Journal of the American Statistical Association*, Vol. 102, No. 477, pp. 359–378. DOI: 10.1198/016214506000001437 (cit. on p. 66).

Gneiting, T. and R. Ranjan (2013). *Combining predictive distributions*. In: *Electronic Journal of Statistics*, Vol. 7, pp. 1747–1782. DOI: 10.1214/13-EJS823 (cit. on p. 11).

Goebel, C., H.-A. Jacobsen, V. del Razo, C. Doblander, J. Rivera, J. Ilg, C. Flath, H. Schmeck, C. Weinhardt, D. Pathmaperuma, H.-J. Appelrath, M. Sonnenschein, S. Lehnhoff, O. Kramer, T. Staake, E. Fleisch, D. Neumann, J. Strüker, K. Erek, R. Zarnekow, H. Ziekow, and J. Lässig (2014). *Energy Informatics*. In: *Business & Information Systems Engineering*, Vol. 6, No. 1, pp. 25–31. DOI: 10.1007/s12599-013-0304-2 (cit. on p. 19).

Gottwalt, S., J. Garttner, H. Schmeck, and C. Weinhardt (2017). *Modeling and Valuation of Residential Demand Flexibility for Renewable Energy Integration*. In: *IEEE Transactions on Smart Grid*, Vol. 8, No. 6, pp. 2565–2574. DOI: 10.1109/TSG.2016.2529424 (cit. on pp. 20, 24).

Gottwalt, S., W. Ketter, C. Block, J. Collins, and C. Weinhardt (2011). *Demand side management—A simulation of household behavior under variable prices*. In: *Energy Policy*, Vol. 39, No. 12, pp. 8163–8174. DOI: 10.1016/j.enpol.2011.10.016 (cit. on pp. 20, 49).

Hagenmeyer, V., H. Langner, and W. Hartwig (2014). *Eine Methode zur Bewertung der Energieversorgungssicherheit von komplexen Produktionsstätten*. In: *VGB-Fachtagung: Dampferzeuger, Wirbelschichtfeuerungen, Industrie- und Heizkraftwerke*. Weimar (cit. on p. 94).

Halvorsen, B. and B. M. Larsen (2001). *The flexibility of household electricity demand over time*. In: *Resource and Energy Economics*, Vol. 23, No. 1, pp. 1–18. DOI: doi.org/10.1016/S0928-7655(00)00035-X (cit. on p. 21).

Hart, G. W. (1992). *Nonintrusive appliance load monitoring*. In: *Proceedings of the IEEE*, Vol. 80, No. 12, pp. 1870–1891. DOI: 10.1109/5.192069 (cit. on p. 85).

Hastie, T. J., R. J. Tibshirani, and J. H. Friedman (2013). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. 2. ed., corr. at 7th printing. Springer series in statistics. New York, NY: Springer. DOI: 10.1007/978-0-387-84858-70 (cit. on pp. 75–77).

Hatziargyriou, N. D., D. Škrlec, T. Capuder, P. S. Georgilakis, and M. Zidar (2016). *Review of energy storage allocation in power distribution networks: applications, methods and future research*. In: *IET Generation, Transmission & Distribution*, Vol. 10, No. 3, pp. 645–652. DOI: 10.1049/iet-gtd.2015.0447 (cit. on p. 122).

Haupt, S. E., M. Garcia Casado, M. Davidson, J. Dobschinski, P. Du, M. Lange, T. Miller, C. Mohrlen, A. Motley, R. Pestana, and J. Zack (2019). *The Use of Probabilistic Forecasts: Applying Them in Theory and Practice*. In: *IEEE Power and Energy Magazine*, Vol. 17, No. 6, pp. 46–57. DOI: 10.1109/MPE.2019.2932639 (cit. on p. 66).

Hayn, M., V. Bertsch, and W. Fichtner (2014). *Electricity load profiles in Europe: The importance of household segmentation*. In: *Energy Research and Social Science*, Vol. 3, No. C, pp. 30–45. DOI: 10.1016/j.erss.2014.07.002 (cit. on p. 121).

He, X., N. Keyaerts, I. Azevedo, L. Meeus, L. Hancher, and J.-M. Glachant (2013). *How to engage consumers in demand response: A contract perspective*. In: *Utilities Policy*, Vol. 27, pp. 108–122. DOI: 10.1016/j.jup.2013.10.001 (cit. on p. 20).

Hearn, C. S., M. C. Lewis, S. B. Pratap, R. E. Hebner, F. M. Uriarte, D. Chen, and R. G. Longoria (2013). *Utilization of optimal control law to size grid-level flywheel energy storage*. In: *IEEE Transactions on Sustainable Energy*, Vol. 4, No. 3, pp. 611–618. DOI: 10.1109/TSTE.2013.2238564 (cit. on p. 128).

Heider, A. and R. Reibsch (2020). *Empirical research on flexibility options in open energy models*. Open Modelling Energy Initiative Online lightning talk mini-workshop (cit. on p. 5).

Heleno, M., M. A. Matos, and J. A. P. Lopes (2015). *Availability and Flexibility of Loads for the Provision of Reserve*. In: *IEEE Transactions on Smart Grid*, Vol. 6, No. 2, pp. 667–674. DOI: 10.1109/TSG.2014.2368360 (cit. on p. 5).

Heppelmann, T., Z. Ben Bouallegue, and S. Theis (2015). *Exploring the calibration of a wind forecast ensemble for energy applications*. In: *EGU General Assembly Conference Abstracts* (cit. on p. 66).

Heppelmann, T., A. Steiner, and S. Vogt (2017). *Application of numerical weather prediction in wind power forecasting: Assessment of the diurnal cycle*. In: *Meteorologische Zeitschrift*, Vol. 26, No. 3, pp. 319–331. DOI: 10.1127/metz/2017/0820 (cit. on pp. 66, 67).

Hor, C.-L., S. J. Watson, and S. Majithia (2005). *Analyzing the Impact of Weather Variables on Monthly Electricity Demand*. In: *IEEE Transactions on Power Systems*, Vol. 20, No. 4, pp. 2078–2085. DOI: 10.1109/TPWRS.2005.857397 (cit. on pp. 65, 66).

Jeon, J. and J. W. Taylor (2012). *Using Conditional Kernel Density Estimation for Wind Power Density Forecasting*. In: *Journal of the American Statistical Association*, Vol. 107, No. 497, pp. 66–79. DOI: 10.1080/01621459.2011.643745 (cit. on p. 65).

Joe-Wong, C., S. Sen, S. Ha, and M. Chiang (2012). *Optimized Day-Ahead Pricing for Smart Grids with Device-Specific Scheduling Flexibility*. In: *IEEE Journal on Selected Areas in Communications*, Vol. 30, No. 6, pp. 1075–1085. DOI: 10.1109/JSAC.2012.120706 (cit. on p. 5).

Junker, R. G., A. G. Azar, R. A. Lopes, K. B. Lindberg, G. Reynders, R. Relan, and H. Madsen (2018). *Characterizing the energy flexibility of buildings and districts*. In: *Applied Energy*, Vol. 225, pp. 175–182. DOI: 10.1016/j.apenergy.2018.05.037 (cit. on p. 5).

Kabelitz, S. and M. Matke (2018). *Production Process Modeling for Demand Management*. In: *Operations research proceedings 2017*. Vol. 67. Operations Research Proceedings. Springer, pp. 261–267. DOI: 10.1007/978-3-319-89920-6\textunderscore (cit. on p. 5).

Karrari, S., N. Ludwig, V. Hagenmeyer, and M. Noe (2019). *A Method for Sizing Centralised Energy Storage Systems Using Standard Patterns*. In: *2019 IEEE Milan PowerTech*. IEEE, pp. 1–6. DOI: 10.1109/PTC.2019.8810658 (cit. on pp. 121, 122).

Karrari, S., M. Noe, and J. Geisbuesch (2018). *Real-time Simulation of High-speed Flywheel Energy Storage System (FESS) for Distribution Networks*. In: *Proceedings of the Ninth International Conference on Future Energy Systems - e-Energy '18*. ACM Press, pp. 388–390. DOI: 10.1145/3208903.3212034 (cit. on p. 128).

Kolisch, R., C. Schwindt, and A. Sprecher (1999). *Benchmark Instances for Project Scheduling Problems*. In: *Project Scheduling*. International Series in Operations Research & Management Science. Springer, pp. 197–212. DOI: 10.1007/978-1-4615-5533-9\textunderscore (cit. on p. 49).

Kolisch, R. and A. Sprecher (1997). *PSPLIB - A project scheduling problem library*. In: *European Journal of Operational Research*, Vol. 96, No. 1, pp. 205–216. DOI: 10.1016/S0377-2217(96)00170-1 (cit. on p. 49).

Kwon, P. S. and P. Østergaard (2014). *Assessment and evaluation of flexible demand in a Danish future energy scenario*. In: *Applied Energy*, Vol. 134, pp. 309–320. DOI: 10.1016/j.apenergy.2014.08.044 (cit. on p. 121).

Lerch, S. (2016). *Probabilistic forecasting and comparative model assessment, with focus on extreme events*. Dissertation. Karlsruhe, Germany: Karlsruhe Institute of Technology. DOI: 10.5445/IR/1000055628 (cit. on p. 11).

Li, D., S. K. Jayaweera, and A. Naseri (2011). *Auctioning game based Demand Response scheduling in smart grid*. In: *2011 IEEE Online Conference on Green Communications (GreenCom)*. IEEE, pp. 58–63. DOI: 10.1109/GreenCom.2011.6082508 (cit. on p. 108).

Li, G. and J. Shi (2012). *Agent-based modeling for trading wind power with uncertainty in the day-ahead wholesale electricity markets of single-sided auctions*. In: *Applied Energy*, Vol. 99, pp. 13–22. DOI: 10.1016/j.apenergy.2012.04.022 (cit. on p. 49).

Lin, J., E. Keogh, S. Lonardi, and B. Chiu (2003). *A symbolic representation of time series, with implications for streaming algorithms*. In: *Proceedings of the 8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery - DMKD '03*. ACM Press, pp. 2–11. DOI: 10.1145/882082.882086 (cit. on p. 37).

Linardi, M., Y. Zhu, T. Palpanas, and E. Keogh (2018). *VALMOD: A Suite for Easy and Exact Detection of Variable Length Motifs in Data Series*. In: *Proceedings of the 2018 International Conference on Management of Data*. SIGMOD'18. Association for Computing Machinery, pp. 1757–1760. DOI: 10.1145/3183713.3193556 (cit. on p. 37).

Liu, Z., A. Wierman, Y. Chen, B. Razon, and N. Chen (2013). *Data center demand response: Avoiding the coincident peak via workload shifting and local generation*. In: *Performance Evaluation*, Vol. 70, No. 10, pp. 770–791. DOI: 10.1016/j.peva.2013.08.014 (cit. on p. 85).

Logenthiran, T., D. Srinivasan, and T. Z. Shun (2012). *Demand Side Management in Smart Grid Using Heuristic Optimization*. In: *IEEE Transactions on Smart Grid*, Vol. 3, No. 3, pp. 1244–1252. DOI: 10.1109/TSG.2012.2195686 (cit. on p. 49).

Ludwig, N., S. Arora, and J. Taylor (2020). *Probabilistic Load Forecasting Using Post-Processed Weather Ensemble Predictions*. In: *Journal of the Operational Research Society (submitted)* (cit. on p. 65).

Ludwig, N., L. Barth, D. Wagner, and V. Hagenmeyer (2019a). *Benchmark Dataset for "Industrial Demand-Side Flexibility: A Benchmark Data Set "*. KITOpen Repository. DOI: 10.5445/IR/1000094324 (cit. on p. 53).

Ludwig, N., L. Barth, D. Wagner, and V. Hagenmeyer (2019b). *Industrial Demand-Side Flexibility: A Benchmark Data Set*. In: *Proceedings of the Ninth International Conference on Future Energy Systems - e-Energy '19*. The Association for Computing Machinery, pp. 460–473. DOI: 10.1145/3307772.3331021 (cit. on pp. 35, 49, 52, 54).

Ludwig, N., S. Feuerriegel, and D. Neumann (2015). *Putting Big Data analytics to work: Feature selection for forecasting electricity prices using the LASSO and random forests*. In: *Journal of Decision Systems*, Vol. 24, No. 1, pp. 19–36. DOI: 10.1080/12460125.2015.994290 (cit. on p. 76).

Ludwig, N., R. Mikut, and V. Hagenmeyer (2018a). *Auction Design to Use Flexibility Potentials in the Energy - Intensive Industry*. In: *2018 15th International Conference on the European Energy Market (EEM)*, pp. 1–5. DOI: 10.1109/EEM.2018.8470003 (cit. on p. 107).

Ludwig, N., S. Waczowicz, R. Mikut, and V. Hagenmeyer (2017). *Mining Flexibility Patterns in Energy Time Series from Industrial Processes*. In: *Proceedings 27. Workshop Computational Intelligence*. KIT Scientific Publishing, pp. 13–32. DOI: 10.5445/KSP/1000074341 (cit. on pp. 35, 87).

Ludwig, N., S. Waczowicz, R. Mikut, and V. Hagenmeyer (2018b). *Assessment of Unsupervised Standard Pattern Recognition Methods for Industrial Energy Time Series*. In: *Proceedings of the Ninth International Conference on Future Energy Systems - e-Energy '18*. ACM Press, pp. 434–435. DOI: 10.1145/3208903.3212051 (cit. on p. 36).

Luo, Z., R. Kumar, J. Sottile, and J. C. Yingling (1998). *An MILP Formulation for Load-Side Demand Control*. In: *Electric Machines & Power Systems*, Vol. 26, No. 9, pp. 935–949. DOI: 10.1080/07313569808955868 (cit. on p. 24).

Makarov, Y. V., P. Du, M. C. W. Kintner-Meyer, C. Jin, and H. F. Illian (2012). *Sizing energy storage to accommodate high penetration of variable energy resources*. In: *IEEE Transactions on Sustainable Energy*, Vol. 3, No. 1, pp. 34–40. DOI: 10.1109/TSTE.2011.2164101 (cit. on p. 122).

Meinshausen, N. (2006). *Quantile regression forests*. In: *Journal of Machine Learning Research*, Vol. 7, No. Jun, pp. 983–999 (cit. on p. 77).

Miller, C., Z. Nagy, and A. Schlueter (2015). *Automated daily pattern filtering of measured building performance data*. In: *Automation in Construction*, Vol. 49, pp. 1–17. DOI: 10.1016/j.autcon.2014.09.004 (cit. on p. 6).

Mirasgedis, S., Y. Sarafidis, E. Georgopoulou, D. P. Lalas, M. Moschovits, F. Karagiannis, and D. Papakonstantinou (2006). *Models for mid-term electricity demand forecasting incorporating weather influences*. In: *Energy*, Vol. 31, No. 2, pp. 208–227. DOI: 10.1016/j.energy.2005.02.016 (cit. on p. 66).

Mitra, S., I. E. Grossmann, J. M. Pinto, and N. Arora (2012). *Optimal production planning under time-sensitive electricity prices for continuous power-intensive processes*. In: *Computers & Chemical Engineering*, Vol. 38, pp. 171–184. DOI: 10.1016/j.compchemeng.2011.09.019 (cit. on pp. 21, 24).

Möller, A., T. L. Thorarinsdottir, A. Lenkoski, and T. Gneiting (2015). *Spatially adaptive, Bayesian estimation for probabilistic temperature forecasts*. In: *Arxiv*. DOI: arXiv:1507.05066v3 (cit. on p. 66).

Moon, J.-Y. and J. Park (2014). *Smart production scheduling with time-dependent and machine-dependent electricity cost by considering distributed energy resources and energy storage*. In: *International Journal of Production Research*, Vol. 52, No. 13, pp. 3922–3939. DOI: 10.1080/00207543.2013.860251 (cit. on pp. 21, 24).

Müller, M. (2007). *Information Retrieval for Music and Motion*. Berlin, Heidelberg: Springer Berlin Heidelberg. DOI: 10.1007/978-3-540-74048-3 (cit. on p. 41).

Murphy, A. H. and R. L. Winkler (1987). *A General Framework for Forecast Verification*. In: *Monthly Weather Review*, Vol. 115, No. 7, pp. 1330–1338. DOI: 10.1175/1520-0493(1987)115<1330:AGFFFV>2.0.CO;2 (cit. on p. 11).

Nagarajan, A. and R. Ayyanar (2015). *Design and Strategy for the Deployment of Energy Storage Systems in a Distribution Feeder With Penetration of Renewable Resources*. In: *IEEE Transactions on Sustainable Energy*, Vol. 6, No. 3, pp. 1085–1092. DOI: 10.1109/TSTE.2014.2330294 (cit. on pp. 123, 127, 128).

Nelsen, R. B. (2007). *An introduction to copulas*. Springer Science & Business Media (cit. on p. 73).

Neupane, B., T. B. Pedersen, and B. Thiesson (2014). *Towards Flexibility Detection in Device-Level Energy Consumption*. In: *Data analytics for renewable energy integration*. Vol. 8817. Lecture Notes in Computer Science. Springer, pp. 1–16. DOI: 10.1007/978-3-319-13290-7\textunderscore (cit. on pp. 12, 109).

Neupane, B., T. B. Pedersen, and B. Thiesson (2018). *Utilizing Device-level Demand Forecasting for Flexibility Markets*. In: *Proceedings of the Ninth International Conference on Future Energy Systems - e-Energy '18*. ACM Press, pp. 108–118. DOI: 10.1145/3208903.3208922 (cit. on p. 6).

Nguyen, H. K., J. B. Song, and Z. Han (2015). *Distributed Demand Side Management with Energy Storage in Smart Grid*. In: *IEEE Transactions on Parallel and Distributed Systems*, Vol. 26, No. 12, pp. 3346–3357. DOI: 10.1109/TPDS.2014.2372781 (cit. on p. 121).

Nicolosi, M. (2010). *Wind power integration and power system flexibility–An empirical analysis of extreme events in Germany under the new negative price regime*. In: *Energy Policy*, Vol. 38, No. 11, pp. 7257–7268. DOI: 10.1016/j.enpol.2010.08.002 (cit. on p. 5).

O'Brien, G. and R. Rajagopal (2016). *Scheduling Non-Preemptive Deferrable Loads*. In: *IEEE Transactions on Power Systems*, Vol. 31, No. 2, pp. 835–845. DOI: 10.1109/TPWRS.2015.2402198 (cit. on p. 85).

Oudalov, A., R. Cherkaoui, and A. Beguin (2007). *Sizing and Optimal Operation of Battery Energy Storage System for Peak Shaving Application*. In: *IEEE Lausanne PowerTech*. IEEE Service Center, pp. 621–625. DOI: 10.1109/PCT.2007.4538388 (cit. on pp. 21, 24, 122).

Palensky, P. and D. Dietrich (2011). *Demand Side Management: Demand Response, Intelligent Energy Systems, and Smart Loads*. In: *IEEE Transactions on Industrial Informatics*, Vol. 7, No. 3, pp. 381–388. DOI: 10.1109/TII.2011.2158841 (cit. on pp. 13, 21, 85).

Paris Agreement (2015). *Paris Agreement to the United Nations Framework Convention on Climate Change*. https://unfccc.int/sites/default/files/english_paris_agreement.pdf. T.I.A.S. No. 16-1104 (cit. on p. 3).

Paulus, M. and F. Borggrefe (2011). *The potential of demand-side management in energy-intensive industries for electricity markets in Germany*. In: *Applied Energy*, Vol. 88, No. 2, pp. 432–441. DOI: 10.1016/j.apenergy.2010.03.017 (cit. on p. 21).

Petersen, M. K., K. Edlund, L. H. Hansen, J. Bendtsen, and J. Stoustrup (2013). *A taxonomy for modeling flexibility and a computationally efficient algorithm for dispatch in Smart Grids*. In: *2013 American Control Conference (ACC)*, pp. 1150–1156. DOI: 10.1109/ACC.2013.6579991 (cit. on pp. 19, 21, 24).

Petersen, M. K., L. H. Hansen, J. Bendtsen, K. Edlund, and J. Stoustrup (2014). *Heuristic optimization for the discrete virtual power plant dispatch problem*. In: *IEEE Transactions on Smart Grid*, Vol. 5, No. 6, pp. 2910–2918. DOI: 10.1109/TSG.2014.2336261 (cit. on pp. 21, 24, 32, 49).

Petersen, M., L. H. Hansen, and T. Mølbak (2012). *Exploring the Value of Flexibility: A Smart Grid Discussion*. In: *American Control Conference (ACC)*. Vol. 45, pp. 43–48. DOI: 10.3182/20120902-4-FR-2032.00010 (cit. on p. 6).

Plataniotis, K. N. and D. Hatzinakos (2000). *Gaussian mixtures and their applications to signal processing: 3*. In: *Advanced Signal Processing Handbook: Theory and Implementation for Radar, Sonar, and Medical Imaging Real Time Systems*. Ed. by S. Stergiopoulos. CRC Press, pp. 3–1 –3–35 (cit. on p. 52).

Pratt, J. W. (1959). *Remarks on Zeros and Ties in the Wilcoxon Signed Rank Procedures*. In: *Journal of the American Statistical Association*, Vol. 54, No. 287, pp. 655–667. DOI: 10.1080/01621459.1959.105015260 (cit. on p. 95).

Pudjianto, D. and G. Strbac (2017). *Assessing the value and impact of demand side response using whole-system approach*. In: *Proceedings of the Institution of Mechanical Engineers, Part A: Journal of Power and Energy*. Vol. 231, pp. 498–507. DOI: 10.1177/0957650917722381 (cit. on p. 85).

Qureshi, F. A., T. T. Gorecki, and C. N. Jones (2014). *Model predictive control for market-based demand response participation*. In: *IFAC Proceedings Volumes*, Vol. 47, No. 3, pp. 11153–11158. DOI: 10.3182/20140824-6-ZA-1003.02395 (cit. on p. 21).

Qureshi, W. A., N.-K. C. Nair, and M. M. Farid (2011). *Impact of energy storage in buildings on electricity demand side management*. In: *Energy Conversion and Management*, Vol. 52, No. 5, pp. 2110–2120. DOI: 10.1016/j.enconman.2010.12.008 (cit. on p. 121).

Raftery, A. E., T. Gneiting, F. Balabdaoui, and M. Polakowski (2005). *Using Bayesian Model Averaging to Calibrate Forecast Ensembles*. In: *Monthly Weather Review*, Vol. 133, No. 5, pp. 1155–1174. DOI: 10.1175/MWR2906.1 (cit. on p. 66).

Ramachandran, B., S. K. Srivastava, C. S. Edrington, and D. A. Cartes (2011). *An Intelligent Auction Scheme for Smart Grid Market Using a Hybrid Immune Algorithm*. In: *IEEE Transactions on Industrial Electronics*, Vol. 58, No. 10, pp. 4603–4612. DOI: 10.1109/TIE.2010.2102319 (cit. on p. 108).

Rollins, S. and N. Banerjee (2014). *Using rule mining to understand appliance energy consumption patterns*. In: *2014 IEEE International Conference on Pervasive Computing and Communications (PerCom)*. IEEE, pp. 29–37. DOI: 10.1109/PerCom.2014.6813940 (cit. on p. 85).

Ru, Y., J. Kleissl, and S. Martinez (2013). *Storage size determination for grid-connected photovoltaic systems*. In: *IEEE Transactions on Sustainable Energy*, Vol. 4, No. 1, pp. 68–81. DOI: 10.1109/TSTE.2012.2199339 (cit. on pp. 122, 123).

Schefzik, R. (2017). *Ensemble calibration with preserved correlations: unifying and comparing ensemble copula coupling and member-by-member postprocessing*. In: *Quarterly Journal of the Royal Meteorological Society*, Vol. 143, No. 703, pp. 999–1008. DOI: 10.1002/qj.2984 (cit. on p. 73).

Schefzik, R., T. L. Thorarinsdottir, and T. Gneiting (2013). *Uncertainty Quantification in Complex Simulation Models Using Ensemble Copula Coupling*. In: *Statistical Science*, Vol. 28, No. 4, pp. 616–640. DOI: 10.1214/13-STS443 (cit. on pp. 66, 73).

Scheuerer, M. and L. Büermann (2014). *Spatially adaptive post-processing of ensemble forecasts for temperature*. In: *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, Vol. 63, No. 3, pp. 405–422. DOI: 10.1111/rssc.12040 (cit. on p. 66).

Schilling, G. and C. C. Pantelides (1996). *A simple continuous-time process scheduling formulation and a novel solution algorithm*. In: *Computers & Chemical Engineering*, Vol. 20, S1221–S1226. DOI: 10.1016/0098-1354(96)00211-6 (cit. on pp. 21, 24).

Schleicher-Tappeser, R. (2012). *How Renewables will Change Electricity Markets in the Next Five Years*. In: *Energy Policy*, Vol. 48, pp. 64–75. DOI: 10.1016/j.enpol.2012.04.042 (cit. on p. 19).

Schuller, A., C. M. Flath, and S. Gottwalt (2015). *Quantifying load flexibility of electric vehicles for renewable energy integration*. In: *Applied Energy*, Vol. 151, pp. 335–344. DOI: 10.1016/j.apenergy.2015.04.004 (cit. on p. 5).

Scott, P., S. Thiébaux, M. van den Briel, and P. van Hentenryck (2013). *Residential demand response under uncertainty*. In: *International Conference on Principles and Practice of Constraint Programming*, pp. 645–660. DOI: 10.1007/978-3-642-40627-0_48 (cit. on p. 20).

Setlhaolo, D., X. Xia, and J. Zhang (2014). *Optimal scheduling of household appliances for demand response*. In: *Electric Power Systems Research*, Vol. 116, pp. 24–28. DOI: doi.org/10.1016/j.epsr.2014.04.012 (cit. on pp. 20, 107).

Shao, H., M. Marwah, and N. Ramakrishnan (2013). *A Temporal Motif Mining Approach to Unsupervised Energy Disaggregation: Applications to Residential and Commercial Buildings*. In: *Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence*. AAAI'13. AAAI Press, pp. 1327–1333 (cit. on p. 6).

Simmhan, Y. and M. U. Noor (2013). *Scalable prediction of energy consumption using incremental time series clustering*. In: *2013 IEEE International Conference on Big Data*. IEEE, pp. 29–36. DOI: 10.1109/BigData.2013.6691774 (cit. on p. 36).

Sklar, A. (1973). *Random variables, joint distribution functions, and copulas*. In: *Kybernetika*, Vol. 9, No. 6, pp. 449–460 (cit. on p. 73).

Soares, A., Á. Gomes, and C. H. Antunes (2014). *Categorization of residential electricity consumption as a basis for the assessment of the impacts of demand response actions*. In: *Renewable and Sustainable Energy Reviews*, Vol. 30, pp. 490–503. DOI: 10.1016/j.rser.2013.10.0190 (cit. on p. 21).

Soliman, H. M. and A. Leon-Garcia (2014). *Game-Theoretic Demand-Side Management With Storage Devices for the Future Smart Grid*. In: *IEEE Transactions on Smart Grid*, Vol. 5, No. 3, pp. 1475–1485. DOI: 10.1109/TSG.2014.2302245 (cit. on p. 121).

Son, H. and C. Kim (2017). *Short-term forecasting of electricity demand for the residential sector using weather and social variables*. In: *Resources, Conservation and Recycling*, Vol. 123, pp. 200–207. DOI: 10.1016/j.resconrec.2016.01.016 (cit. on p. 66).

Sou, K. C., J. Weimer, H. Sandberg, and K. H. Johansson (2011). *Scheduling smart home appliances using mixed integer linear programming*. In: *2011 50th IEEE Conference on Decision and Control and European Control Conference*. Piscataway, NJ: IEEE, pp. 5144–5149. DOI: 10.1109/CDC.2011.6161081 (cit. on p. 24).

Strähl, C. and J. Ziegel (2017). *Cross-calibration of probabilistic forecasts*. In: *Electronic Journal of Statistics*, Vol. 11, No. 1, pp. 608–639. DOI: 10.1214/17-EJS1244 (cit. on p. 11).

Strbac, G. (2008). *Demand side management: Benefits and challenges*. In: *Energy Policy*, Vol. 36, No. 12, pp. 4419–4426. DOI: 10.1016/j.enpol.2008.09.030 (cit. on pp. 4, 19, 21).

Ströhle, P. and C. M. Flath (2016). *Local matching of flexible load in smart grids*. In: *European Journal of Operational Research*, Vol. 253, No. 3, pp. 811–824. DOI: 10.1016/j.ejor.2016.03.004 (cit. on p. 6).

Ströhle, P., E. H. Gerding, M. M. DE Weerdt, S. Stein, and V. Robu (2014). *Online mechanism design for scheduling non-preemptive jobs under uncertain supply and demand*. In: *Proceedings of the 2014 international Conference on Autonomous Agents and Multiagent Systems*, pp. 437–444 (cit. on p. 21).

Taneja, J. (2014). *Growth in renewable generation and its effect on demand-side management*. In: *2014 IEEE International Conference on Smart Grid Communications (SmartGridComm)*. IEEE, pp. 614–619. DOI: 10.1109/SmartGridComm.2014.7007715 (cit. on p. 4).

Taneja, J., K. Lutz, and D. Culler (2013). *The impact of flexible loads in increasingly renewable grids*. In: *2013 IEEE International Conference on Smart Grid Communications (SmartGridComm)*. IEEE, pp. 265–270. DOI: 10.1109/SmartGridComm.2013.6687968 (cit. on p. 85).

Taylor, J. W. and R. Buizza (2002). *Neural network load forecasting with weather ensemble predictions*. In: *IEEE Transactions on Power Systems*, Vol. 17, No. 3, pp. 626–632. DOI: 10.1109/TPWRS.2002.800906 (cit. on p. 67).

Taylor, J. W. and R. Buizza (2003). *Using weather ensemble predictions in electricity demand forecasting*. In: *International Journal of Forecasting*, Vol. 19, No. 1, pp. 57–70. DOI: 10.1016/S0169-2070(01)00123-6 (cit. on pp. 67, 69).

Thorey, J., C. Chaussin, and V. Mallet (2018). *Ensemble forecast of photovoltaic power with online CRPS learning*. In: *International Journal of Forecasting*, Vol. 34, No. 4, pp. 762–773. DOI: 10.1016/j.ijforecast.2018.05.007 (cit. on pp. 65, 66).

Tibshirani, R. (1996). *Regression Shrinkage and Selection via the Lasso*. In: *Journal of the Royal Statistical Society. Series B (Methodological)*, Vol. 58, No. 1, pp. 267–288. DOI: 10.1111/j.2517-6161.1996.tb02080.x (cit. on p. 75).

Tormos, P. and A. Lova (2001). *A Competitive Heuristic Solution Technique for Resource-Constrained Project Scheduling*. In: *Annals of Operations Research*, Vol. 102, No. 1/4, pp. 65–81. DOI: 10.1023/A:1010997814183 (cit. on p. 49).

Torriti, J. (2012). *Price-based demand side management: Assessing the impacts of time-of-use tariffs on residential electricity demand and peak shifting in Northern Italy*. In: *Energy*, Vol. 44, No. 1, pp. 576–583. DOI: 10.1016/j.energy.2012.05.043 (cit. on p. 107).

Truong, N. C., T. Baarslag, S. D. Ramchurn, and L. Tran-Thanh (2016). *Interactive scheduling of appliance usage in the home*. In: *25th International Joint Conference on Artificial Intelligence (IJCAI-16)*, pp. 869–875 (cit. on p. 85).

Ulbig, A. and G. Andersson (2015). *Analyzing operational flexibility of electric power systems*. In: *International Journal of Electrical Power & Energy Systems*, Vol. 72, pp. 155–164. DOI: 10.1016/j.ijepes.2015.02.028 (cit. on p. 113).

Waczowicz, S., M. Reischl, V. Hagenmeyer, R. Mikut, S. Klaiber, P. Bretschneider, I. Konotop, and D. Westermann (2015). *Demand response clustering - How do dynamic prices affect household electricity consumption?* In: *2015 IEEE Eindhoven PowerTech*. IEEE, pp. 1–6. DOI: 10.1109/PTC.2015.7232493 (cit. on p. 37).

Waczowicz, S., M. Reischl, S. Klaiber, P. Bretschneider, I. Konotop, D. Westermann, V. Hagenmeyer, and R. Mikut (2016). *Virtual Storages as Theoretically Motivated Demand Response Models for Enhanced Smart Grid Operations*. In: *Energy Technology*, Vol. 4, No. 1, pp. 163–176. DOI: 10.1002/ente.201500318 (cit. on p. 107).

Weglarz, J. (2012). *Project Scheduling: Recent Models, Algorithms and Applications*. International Series in Operations Research & Management Science. Springer US. DOI: 10.1007/978-1-4615-5533-9 (cit. on p. 86).

Weidlich, A., H. Vogt, W. Krauss, P. Spiess, M. Jawurek, M. Johns, and S. Karnouskos (2012). *Decentralized intelligence in energy efficient power systems*. In: *Handbook of Networks in Power Systems I*. Springer, pp. 467–486. DOI: 10.1007/978-3-642-23193-3_18 (cit. on p. 19).

Wen, M., R. Lu, J. Lei, H. Li, X. Liang, and X. S. Shen (2014). *SESA: An efficient searchable encryption scheme for auction in emerging smart grid marketing*. In: *Security and Communication Networks*, Vol. 7, No. 1, pp. 234–244. DOI: 10.1002/sec.699 (cit. on p. 108).

Xiao, J., L. Bai, F. Li, H. Liang, and C. Wang (2014). *Sizing of energy storage and diesel generators in an isolated microgrid Using Discrete Fourier Transform (DFT)*. In: *IEEE Transactions on Sustainable Energy*, Vol. 5, No. 3, pp. 907–916. DOI: 10.1109/TSTE.2014.2312328 (cit. on pp. 123, 126).

Yang, Y., H. Li, A. Aichhorn, J. Zheng, and M. Greenleaf (2014). *Sizing strategy of distributed battery storage system with high penetration of photovoltaic for voltage regulation and peak load shaving*. In: *IEEE Transactions on Smart Grid*, Vol. 5, No. 2, pp. 982–991. DOI: 10.1109/TSG.2013.2282504 (cit. on p. 122).

Yaw, S., B. Mumey, E. McDonald, and J. Lemke (2014). *Peak demand scheduling in the Smart Grid*. In: *2014 IEEE International Conference on Smart Grid Communications (SmartGridComm)*. IEEE, pp. 770–775. DOI: 10.1109/SmartGridComm.2014.7007741 (cit. on p. 49).

Zhao, S., X. Lin, and M. Chen (2017). *Robust Online Algorithms for Peak-Minimizing EV Charging Under Multistage Uncertainty*. In: *IEEE Transactions on Automatic Control*, Vol. 62, No. 11, pp. 5739–5754. DOI: 10.1109/TAC.2017.2699290 (cit. on p. 85).