Data-Driven Statistical Modeling and Reinforcement Learning Based Control of Flexibility in Domestic and Electric Vehicle Power Consumption

Data-gedreven statistische modellering en reinforcement-leren voor het controleren van flexibel energieverbruik in huishoudens en elektrische voertuigen

Nasrin Sadeghianpourhamami

Promotor: prof. dr. ir. C. Develder
Proefschrift ingediend tot het behalen van de graad van
Doctor in de ingenieurswetenschappen: computerwetenschappen

UNIVERSITEIT
GENT

Vakgroep Informatietechnologie
Voorzitter: prof. dr. ir. B. Dhoedt
Faculteit Ingenieurswetenschappen en Architectuur
Academiejaar 2018 - 2019

ISBN 978-94-6355-175-5
NUR 980, 984
Wettelijk depot: D/2018/10.500/93

GHENT UNIVERSITY

Ghent University
Faculty of Engineering and Architecture
Department of Information Technology

Examination Board:

prof. Filip De Turck (chair)
prof. Chris Develder (supervisor)
prof. Dries F. Benoit
prof. Phuong H. Nguyen
prof. Guillaume Crevecoeur
dr. Fred Spiessens
dr. Joachim van der Herten

FACULTY OF ENGINEERING AND ARCHITECTURE

Dissertation for acquiring the grade of
Doctor of Computer Science Engineering

Dankwoord

Writing a finishing touch to my PhD thesis and I cannot stop thinking: what an amazing journey! And there are wonderful individuals to thank for helping me to grow personally and professionally during this journey.

First and foremost my supervisor, prof. Chris Develder, for giving me the opportunity and guiding me throughout this journey with his immense knowledge, critical reviews, and constructive feedback. For always being responsive despite his busy schedule. Thank you as well for guiding me, often with big doses of patience, through the subtleties of scientific writing. It has been a privilege working with you and learning from you.

My husband, Hossein, for his unselfish love and endless support. For being so close from far away. For cheering me up and dusting me off during hard times. For celebrating every accomplishment in this journey more than me. For believing in me at times when I did not. For always showing how proud he is of me. For never raising an eyebrow when I claimed my thesis would be finished ‘in the next two months’ for nearly a year. What a blessing to have you beside me!

My Mom, dad, grandmother and mother-in-law for their never-ending encouragements.

My brother Masoud, my sister Elham, and their little bundle of joy, Rayan for their positive energy.

My amazing office mates with whom I have shared moments of deep anxiety but also of big excitement: Thomas, Johannes, and Matthias for always welcoming my questions and discussions; Lucas and Giannis, great scientists and amazing human beings for motivating me and being a listening ear; and Klim for joyful chit-chats. I enjoyed your company in the office, lunches, and fun dinners and will always remember them with a smile.

Some special words of gratitude go to my friends who have always been a major source of support: Yara and Mélissa, who always managed to make me feel special and with whom I had the best outings in Gent; Elnaz for always being kind and cheerful and making me laugh when things would get a bit discouraging; Reza for being there with an open heart whenever I needed help and discovering hidden gems of Gent to show us; Mahdiyeh, an amazing friend that can magically put everyone at ease and having her as a friend has made me a better and stronger person. Thanks guys for always being there for me.

The members of my examination board, prof. Filip De Turck, prof. Dries Benoit, prof. Phuong H. Nguyenm, prof. Guillaume Crevecoeur, dr. Fred Spiessens, and dr. Joachim van der Herten for their insightful comments, questions, and encouragement which helped me to improve the quality of my PhD dissertation.

Prof. Dries Benoit for helping me strengthen my Bayesian statistics knowledge.

Prof. Kankar Bhattacharya and prof. Ali Ghodsi for welcoming me in their team at the University of Waterloo.

Prof. Piet Demeester, an inspiring leader of the IDLab-imec research group.

Colleagues and friends in IDLab, Joeri, Leen, Dirk, Tim, Ankita, Sahel, Maryam, Mehdi, Femke, Domenico, Christof, Duygu, Steven, Andy and Laurent from whom I have learned so much.

I am also very grateful to all the staff members at IDLab and special thanks to Mike Van Puyenbroeck for his guidance during the initial admission phase even before I arrived in Belgium. Also grateful to Martine Buysse, Davinia Stevens, Bernadette Becue and the members of the A-team for their constant support.

For the past four years, Gent has never failed to mesmerize me every time I took a walk in its streets, shopped hand-made chocolates, enjoyed the festivals, and returned the smile of friendly strangers. It still remains as the most beautiful city I have ever been to.

Gent, November 2018
Nasrin Sadeghianpourhamami

Table of Contents

Dankwoord **i**

Samenvatting **xvii**

Summary **xxi**

1 Introduction **1**

1.1 Smart Grid . . . 2

1.2 Demand Side Management . . . 4

1.2.1 Demand response potential benefits . . . 5

1.2.2 Flexibility as a demand response asset . . . 6

1.2.3 Demand response methods . . . 7

1.2.3.1 Model-based DR . . . 9

1.2.3.2 Model-free DR . . . 9

1.2.4 Demand response barriers . . . 10

1.3 Motivations and contributions . . . 10

1.4 Outline . . . 12

1.5 Publications . . . 13

1.5.1 Publications in international journals (listed in the Science Citation Index) . . . 14

1.5.2 Publications in international conferences (listed in the Science Citation Index) . . . 14

References . . . 16

2 Modeling and Analysis of Residential Flexibility: Timing of White Good Usage **19**

2.1 Introduction . . . 20

2.1.1 State of the Art in Residential Flexibility Assessment . . . 21

2.1.2 Motivation and Contributions . . . 23

2.2 Methodology . . . 25

2.2.1 Quantitive Specification of Flexibility . . . 25

2.2.2 Model I: Two-Stage Single Variate Approach . . . 26

2.2.2.1 Stage I: Identification of Typical Deadlines . . . 26

2.2.2.2 Stage II: Parameterizing the Distribution of Configuration Times . . . 27

2.2.3 Model II: Single Stage Bivariate Approach 29
2.3 Results and Discussion 30
2.3.1 Model I Parameter Fitting and Analysis of Resulting Clusters 30
2.3.1.1 Stage I: Analysis of Deadline Clusters 31
2.3.1.2 Stage II: Analysis of Distribution of Configuration Times 32
2.3.2 Model II Parameter Fitting and Analysis of Resulting Clusters 36
2.3.3 Analysis of Model Efficiency for Data Regeneration . . . 37
2.3.4 Dependency Analysis Using Statistical Tests 38
2.4 Conclusion 41
2.4.1 Future Work 43
References 52

3 Quantitive Analysis of Electric Vehicle Flexibility: A Data-Driven Approach 57
3.1 Introduction 58
3.1.1 Objectives and Contributions 59
3.1.2 Related Work 60
3.2 Analysis of EV Charging Behavior 62
3.2.1 Dataset Description 63
3.2.2 Clustering of Charging Session Times 63
3.2.3 Analysis of Behavioral Clusters: Weekdays and Seasonal Impacts 65
3.2.3.1 Analysis of Arrival Times 65
3.2.3.2 Analysis of Sojourn Times 69
3.2.3.3 Analysis of Idle Times 70
3.3 Flexibility Quantification 71
3.3.1 Uncoordinated Charging: Business as Usual 72
3.3.2 Coordinated Charging: Load Flattening and Load Balancing 72
3.3.3 Measures for Quantification of Flexibility Utilization . . . 74
3.3.4 Evaluation of Flexibility Exploitation 76
3.4 Summary and Conclusion 79
References 83

4 Bayesian Modeling of Cylindrical Data Using Abe-Ley Mixtures 87
4.1 Introduction 88
4.2 Probabilistic Model Description 91
4.2.1 Probability Density Functions 91
4.2.2 Random Number Generation 92
4.3 Parameter Estimation using Bayesian Inference 93
4.3.1 Metropolis-Hastings Algorithm for Estimating Abe-Ley Mixture Parameters 93
4.3.2 Addressing the Challenges of a Bayesian Approach 95
4.3.2.1 Adaptive Metropolis-Hastings 96

4.3.2.2 Label Switching Issue 97
4.3.2.3 Bayesian Model Selection 97
4.4 Validation on a Sample Dataset with Known True Parameters . . . 98
4.5 Modeling Real-World Datasets with a Mixture of Abe-Ley Distributions 103
4.5.1 Wave Dynamics in the Adriatic Sea 103
4.5.2 Electric Vehicle Hourly Charging Requests 104
4.6 Conclusion 107
References 109

5 Definition and Evaluation of Model-Free Coordination of Electrical Vehicle Charging with Reinforcement Learning 113
5.1 Introduction 114
5.2 Related Work 116
5.3 Markov Decision Process 119
5.3.1 State Space 119
5.3.2 Action Space 121
5.3.3 Cost function 122
5.3.4 System Dynamics 123
5.3.5 Learning Objective: State-Action Value Function 123
5.4 Batch Reinforcement Learning 124
5.4.1 Fitted Q-iteration 125
5.4.2 The size of state-action space 125
5.5 Experiment setup 126
5.5.1 Data Preparation 126
5.5.2 Algorithm Settings 127
5.5.2.1 Creating set $\mathcal{F}$ 127
5.5.2.2 Neural network architecture 127
5.5.3 Performance Evaluation Measure 128
5.6 Experimental results 129
5.6.1 Learning the Charging Coordination (Q1–Q2) 129
5.6.2 Variance of performance over time (Q3) 132
5.6.3 Generalization to Larger Scales (Q4) 133
5.7 Conclusion 135
References 138

6 Conclusion 141
6.1 Summary 141
6.2 Future Work 146
References 148

A Modeling Real-World Flexibility of Residential Power Consumption: Exploring the Cylindrical WeiSSVM Distribution 149
A.1 Introduction 150

A.2 Modeling User Energy Consumption Flexibility with WeiSSVM Mixtures . . . 151
A.2.0.1 PDF of WeiSSVM Mixtures . . . 151
A.2.0.2 Model Parameter Estimation . . . 151
A.2.0.3 Measures for Model Comparison . . . 151
A.3 Analysis and Model Comparison . . . 152
A.4 Conclusion . . . 154
References . . . 155

List of Figures

1.1 Comparing the traditional and the smart grid [9] [10] 3

1.2 The components of the smart grid (reproduced from [11] with minor modification of the context and graphics). The underlined parts are areas which this thesis contributes to. 4

1.3 DR classification based on the optimization approach 8

2.1 Flexibility profiles of four selected customers for dishwasher usage (times on x- and y-axis are time of day). Note that the observations below the diagonal line denote deadlines on the day after the configuration time. 25

2.2 Clusters of deadlines from Stage I in the customer flexibility profiles for dishwasher. Each cluster is shown in a different color. . . 31

2.3 Percentage of data in each deadline cluster for dishwasher. (Note that a given cluster index for one customer may relate to a different deadline than the same index for another customer.) 31

2.4 Example of a customer flexible configurations for dishwasher (Customer 6) during holidays vs. normal days over the cluster of deadlines. 32

2.5 Point process representation of posterior draws and PDF of the best fit for two randomly selected customers for dishwasher. . . . 33

2.6 Bivariate Gaussian mixture fits and the resulting MAP clusters for *dishwasher* (in distinct symbols and colors; contour plots indicate the fitted distributions). 35

2.7 Bivariate Gaussian mixture fits and the resulting MAP clusters for *washing machine* flexible usage (in distinct symbols and colors; contour plots omitted for clarity). 35

2.8 Bivariate Gaussian mixture fits and the resulting MAP clusters for *tumble dryer* flexible usage (in distinct colors; contour plots omitted for clarity). 36

3.1 behavioral clusters of sessions in terms of EV arrival and departure times. Both X- and Y-axis denote time-of-day (i.e., we report times as $t \bmod 24\,\mathrm{h}$): points below the X=Y diagonal have departures on the day after the arrival or later. (Note that also some sessions plotted above the diagonal actually have departures $\geq 24\,\mathrm{h}$ after arrival) 64

3.2 Violin and box plots of time of arrivals for the behavioral clusters over weekends and weekdays in each season (Note that the reference is changed from midnight to 3 am (2.30 am to 3.30 am is the interval with least number of arrivals) to acount for the fact that the activities right after the midnight are continuation of the late night activities) 66

3.3 Violin and box plots of sojourn times for the behavioral clusters over weekends and weekdays in each season 67

3.4 Violin and box plots of Idle times for the behavioral clusters over weekends and weekdays in each season 70

3.5 (a) Load and renewable generation patterns from 5th to 11th Jan, (b) Amount of energy that is shifted away from each slot(for arrivals from 5th to 11th Jan 2015) in load flattening scenario and (c) Amount of energy that is shifted away from each slot (for arrivals from 5th to 11th Jan 2015) in load balancing scenario. . . . 76

3.6 Average *Tflex* and *Eflex* values for each 15 min long timeslot in a day (LB: load balancing, LF:load flattening) 78

4.1 Sampled datasets from mixture of true (top row) and estimated (bottom row) Abe-Ley distributions. Contour plots indicate two-dimensional kernel densities 99

4.2 Posterior densities and trace plots of the parameters for mixture of 3 Abe-Ley (Dataset (a)). Black vertical lines mark the true parameter values and shaded areas are the 95% Bayesian credible intervals 100

4.3 Component-wise densities for the true and the estimated parameters for Dataset (a) 102

4.4 Determining number of mixture components using elpd measure for sample datasets (the bend in each curve is used for selecting the best number of mixture components). 102

4.5 The elpd values for model selection in modeling wave dynamics. (The bend in the curve is used for selecting the best number of mixture components.) 104

4.6 Posterior densities of the parameters for best Abe-Ley mixture model for wave dynamics. 105

4.7 Component-wise densities for (a) Abe-Ley mixture model estimated by [11] and (b) Abe-Ley mixture model estimated by our proposed approach. 105

4.8 elpd values for model selection in modeling EV arrivals (the bend in the curve is used for selecting the best number of mixture components) 106
4.9 Posterior densities of the parameters for best Abe-Ley mixture model for EV arrival 107
4.10 Component-wise densities for the true and estimated parameters in modeling EV arrivals 108

5.1 A simple example for $N_{\text{max}} = 2$ charging stations: (a) state representation, (b) possible action states, (c) full decision tree over the horizon of $S_{\text{max}} = 3$ slots. 120
5.2 Normalized costs of learned policy (C_{RL}), BAU policy (C_{BAU}) and optimum solution (C_{opt}) for coordinating the charging of 10 (top row) and 50 (bottom row) EV charging stations, (a) normalized costs as a function of number of samples per day for various Δts, and (b) normalized costs as a function of Δt for various numbers of sample trajectories per training day. 130
5.3 Performance using different months as test set and different time spans of the training set (1–5 months). 132
5.4 Improvement in normalized cost of the learned policy (RL) with respect to the business-as-usual policy (BAU). 132
5.5 The effect of scaling up the group size on a normalized cost of a policy learned from 10 EV charging stations 133
5.6 The effect of scaling up N_{max} on a normalized cost of a policy learned from $N_{\text{max}} = 10$ EV charging stations for different of number of sampled trajectories (ranging from 5K to 20K). 134

6.1 Blockchain based architecture for decentralized management of energy grids (from [1]) 147

A.1 Comparison of distribution shapes of the mixture components and identified clusters with WMMs (top row) and GMMs (bottom row) for selected users of dishwashers. Note that data on bottom row is wrapped around a new x-axis reference, while for the data on top row, the x-axis is circular (i.e., 00:00 and 24:00 are the same points) 153

List of Tables

2.1 Log-marginal Likelihood, $p(\mu_k|\boldsymbol{X})$ (larger values are better). . . . 34

2.2 Analyses of model efficiency in regenerating the customer flexibility profiles: percentage of times the null hypothesis is not rejected. (Note that the same customer numbers across devices do not necessarily correspond to the same users.) 38

2.3 P-values of dependency tests in Stage I of Model I for dishwasher usage. (Bold values indicate a p-value less than 0.05 and shaded rows indicate customers affected by at least one ‘factor’.) 40

2.4 P-values of dependency tests in Stage II of Model I for dishwasher usage. (Bold values indicate a p-value less than 0.05 and shaded rows indicate customers affected by at least one ‘factor’.) 41

2.5 P-values of dependency tests for Model II for dishwasher usage. (Bold values indicate a p-value less than 0.05 and shaded rows indicate customers affected by at least a ‘factor’.) 42

2.6 Summary of estimated parameters of Model II based on customer flexibility profiles for tumble dryers 46

2.7 Summary of estimated parameters of Model II based on customer flexibility profiles for washing machine 47

2.8 Summary of estimated parameters of Model II based on customer flexibility profiles for dishwasher 48

3.1 Summary of cluster and sub-cluster fractions and average sojourn and idle times 68

3.2 Nomenclature 71

4.1 Choice of priors and proposal distributions at the input of Algorithm 3 94

4.2 Estimated and true parameters for sampled data from mixture of Abe-Ley distribution (The true values are shown in parenthesis) . 100

5.1 Nomenclature 118

A.1 *elppd* of GMM and WMM fits for dishwashers. 152

List of Acronyms

AIC Akaike Information Criteria

BAU Business as Usual

BIC Bayesian Information Criteria

BRP Balance Responsible Party

CPP Critical Peak Pricing

DR Demand Response

DSM Demand Side Management

DSO Distribution System Operator

EE Energy Efficiency

EM Expectation Maximization

EU European Union

EV Electric Vehicles

FMM Finite Mixture Models

FQI Fitted Q-Iteration

GMM Gaussian Mixture Model

ICE Internal Combustion Engine

ICT Information and Communication Technologies

LOO-CV leave-One-Out Cross-Validation

MAP Maximum-A-Posteriori

MCMC Markov Chain Monte Carlo

MDP Markov Decision Process

MH Metropolis-Hastings

ML Maximum Likelihood

MLE Maximum Likelihood Estimation

MPC Model Predictive Control

NILM Non-Intrusive Load Monitoring

OCP Optimal Control Problem

PDF Probability Density Function

PSIS Pareto-Smoothed Importance Sampling

RES Renewable Energy Sources

RL Reinforcement Learning

RTP Real Time Pricing

TCL Thermostatically Controlled Loads

TOU Time of Use

TSO Transmission System Operator

UNFCCC United Nations Framework Convention on Climate Chang

US United States

WeiSSVM Weibull and Sine-Skewed Von-Mises

WMM WeiSSVM Mixture Model

Samenvatting

Aan het einde van de negentiende eeuw ontstond het revolutionaire concept van het “elektrische net” met elektriciteitscentrales in Holborn Viaduct, Londen en Pearl Street, New York. Dit model met “centrale” productielocaties was een belangrijke stap voorwaarts in de evolutie van kleinschalige elektriciteitsproductie per site naar productie op industriële schaal en werd al snel het model voor het volledige elektriciteitsnet. Sindsdien wordt elektriciteit centraal opgewekt, vaak in steenkoolcentrales, en vandaar verdeeld naar alle gebruikers. Belangrijke traditionele principes zijn dat elektriciteit steeds in dezelfde richting stroomt, van centrale productie-eenheden naar eindgebruikers, en dat de productie steeds het verbruik volgt. Een elektriciteitsverbruik dat blijft stijgen en de groeiende aandacht voor de bijhorende impact op ons milieu hebben er in de 21e eeuw voor gezorgd dat de infrastructuur en beheer van het net geleidelijk evolueren tot een zogenaamd “slim net”. In zo’n slim net wordt informatie- en communicatietechnologie (ICT) geïntegreerd om tegemoet te komen aan het steeds stijgende wereldwijde verbruik op een economische, duurzame en veilige manier. Zo’n slim net is zo ontworpen dat het de gedistribueerde opwekking door hernieuwbare energiebronnen (zon, wind) toelaat en nieuwe elektriciteitsverbruikers (bv. elektrische voertuigen) geïntegreerd kunnen worden.

Het variabele productiepatroon van hernieuwbare energiebronnen brengt nieuwe uitdagingen met zich mee aangezien deze bronnen niet zomaar aangestuurd kunnen worden. De productie hangt af van weerspatronen, het tijdstip van de dag en de seizoenen. Echter, door de integratie van ICT technologie zijn eindgebruikers in een slim net niet langer passieve verbruikers. Ze kunnen bijdragen aan het evenwicht tussen productie en consumptie door flexibiliteit aan te bieden in hun verbruik, wat gestimuleerd kan worden met behulp van variabele energietarieven of andere financiële compensaties. Vraagsturingsalgoritmes gebruiken deze flexibiliteit op een gecoördineerde manier om zo de betrouwbaarheid van het net te verzekeren. Het realiseren van praktisch toepasbare vraagsturingsalgoritmes in een slim net is echter niet zo eenvoudig. Een dergelijk algoritme is: (i) ontworpen en gevalideerd met behulp van accurate modellen van het echte coördinatieprobleem en (ii) toepasbaar op een breed gamma van gelijkaardige coördinatieproblemen. Deze thesis focust op de realisatie van praktisch toepasbare vraagsturingsalgoritmes met bijdragen in twee domeinen: (i) analyse, modellering en kwantificering van flexibiliteit gebaseerd op een aantal concrete datasets en (ii) een modelvrij vraagsturingsalgoritme dat gebruik maakt van een “reinforcement learning” aanpak.

Vraagsturing is gebaseerd op flexibiliteit in het elektriciteitsverbruik. Het is dan ook essentieel om de eigenschappen van deze flexibiliteit te analyseren en te begrijpen. Deze thesis focust op de analyse en modellering van flexibiliteit afkomstig van residentiële witgoedtoestellen en het opladen van elektrische voertuigen. In beide gevallen wordt het tijdsaspect van de flexibiliteit (i.e., wanneer en voor hoelang het elektriciteitsverbruik uitgesteld kan worden) bepaald door de gebruikers. De beschikbare flexibiliteit wordt beïnvloed door hun voorkeuren en gewoontes. Dit maakt het modelleren van deze flexibiliteit uitdagend vanwege de grote verschillen tussen gebruikers onderling en de onzekerheden die samenhangen met hun levensstijl.

Om inzicht te krijgen in de aanwezige flexibiliteit in het verbruik van witgoedtoestellen hebben we ons gebaseerd op een dataset afkomstig van gebruikers die voorzien waren van slim aanstuurbare toestellen en die een vergoeding kregen om deze toestellen zo veel mogelijk flexibel te gebruiken. In plaats van het toestel direct te starten konden gebruikers een bepaalde deadline instellen tegen het wanneer het toestel ten laatste klaar moest zijn. De flexibiliteit wordt dus bepaald door (i) het tijdstip van instellen en (ii) de ingestelde deadline. Analyse van deze dataset laat toe de volgende onderzoeksvragen te beantwoorden: (i) vertonen gebruikers bepaalde gedragspatronen bij het flexibel instellen van hun slimme witgoedtoestellen, en (ii) welke factoren beïnvloeden de ingestelde flexibiliteit? Met behulp van clustering technieken en statistische tests concluderen we dat verschillende gebruikers substantieel verschillend gedrag vertonen en dat de beschikbare flexibiliteit beïnvloed kan worden door factoren als dag van de week, seizoenen, etc.

Merk op dat aangezien het gebruik van slimme toestellen nog in zijn kinderschoenen staat, er slechts een beperkt aantal datasets beschikbaar zijn waarbij toestellen echt flexibel ingesteld konden worden. Dergelijke datasets zijn vaak ook niet publiek beschikbaar vanwege privacy of IP aspecten. Om hieraan tegemoet te komen stellen we twee methodologieën voor om generatieve statistische modellen af te leiden van het flexibiliteitsgedrag van een gebruiker. Op die manier kan realistische data beschikbaar gemaakt worden voor een grotere groep van onderzoekers.

Vervolgens analyseren en kwantificeren we de flexibiliteit bij het opladen van elektrische voertuigen aan de hand van een dataset afkomstig van publieke laadpalen. Via clustering onderscheiden we drie gedragspatronen op basis van de aankomst- en vertrektijden: (i) dichtbij huis opladen, (ii) opladen tijdens werkuren, (iii) parkeren om te laden. Met behulp van boxplots en vioolplots analyseren we de karakteristieken van de laadsessies binnen elke cluster en identificeren we de verschillen tussen de clusters op weekend- en weekdagen in elk seizoen. Tenslotte stellen we twee flexibiliteitsmetrieken voor om de flexibiliteitsbenutting te kwantificeren en een algoritme om te bepalen hoeveel en over welke periode energieverbruik verschoven werd voor verschillende objectieven. We gebruiken de metrieken en het algoritme om na te gaan welk aspect van de flexibiliteit (tijdstip en duur van beschikbaarheid of de hoeveelheid uitstelbare energie) het meest nuttig is op verschillende momenten van de dag.

Het tijdsaspect van flexibiliteit heeft een cilindrisch karakter vanwege het periodieke tijdskarakter van een dag en de lineariteit van de aangeboden flexibiliteitsduur. Bij het analyseren en modelleren van flexibiliteit in de eerder vermelde toepassingen hebben we probabilistische modellen gebruikt met lineaire schaalverdelingen voor het modelleren van de data (bv. Gaussiaanse mixture modellen). Men kan zich de vraag stellen als probabilistische modellen die gebruik maken van cilindrische schaalverdelingen niet meer geschikt zijn. Om die vraag te kunnen beantwoorden stellen we een cilindrische Bayesiaanse aanpak voor om de parameters van de mixture distributie te schatten. We gebruiken deze aanpak om flexibiliteitsdata afkomstig van witgoedtoestellen te modelleren en vergelijken de performantie met lineaire modellen. De vergelijking toont aan dat de lineaire aanpak betere modellen oplevert, zeker vanuit het perspectief van kwantitatieve statistische metrieken.

Tenslotte, als tweede bijdrage van deze thesis, focussen we op de ontwikkeling van modelvrije vraagsturingsalgoritmes voor het gecoördineerd opladen van een verzameling van elektrische voertuigen. In deze modelvrije methode wordt het coördinatieprobleem geformuleerd als een Markov beslissingsproces bestaande uit een toestand, actie, kost/beloning en transitieprobabiliteiten. Hierbij zijn de transitieprobabiliteiten onbekend door het stochastische karakter van toekomstige aankomsten en energienoden van de elektrische voertuigen. In het model leert een agent wat de beste coördinatiestrategie is door te interageren met zijn omgeving. Dit interageren houdt in dat bepaalde acties ondernomen worden in een bepaalde toestand en de uitkomst van deze acties geobserveerd wordt (i.e. de nieuwe toestand en bijhorende kost of beloning). Eén van de uitdagingen bij deze manier van werken is om een schaalbare voorstelling te maken van de toestand en acties rekening houdend met de invloed van het laden van de andere voertuigen in de verzameling. We pakken deze uitdaging aan door het formuleren van een nieuw Markov beslissingsproces met schaalbare voorstellingen van toestand en acties. We gebruiken dan een batch-model reinforcement learning algoritme om de beste strategie te bepalen (i.e., een mapping van toestand naar meest geschikte actie) om groepen elektrische voertuigen van variërende grootte zo optimaal mogelijk op te laden. We analyseren uitgebreid het effect van verschillende instellingen (bv. grootte van de gebruikte dataset, tijdsduur van test- en trainingsets) op de prestatie van onze methodologie. We tonen ook aan dat een coördinatiestrategie geleerd op basis van data afkomstig van een beperkte verzameling elektrische voertuigen bruikbaar is om het laden van een grotere groep wagens te coördineren, met slechts een kleine verhoging van de genormaliseerde kost (in vergelijking met de kost van de optimale alwetende oplossing).

Summary

The 19th century witnessed a revolutionary inception of the "electric grid" with central power stations in Holborn Viaduct, London and the Pearl street, New York. Such "central" power station design was a paramount step from small-scale, onsite power generation to industrial-scale generation, and soon became the underlying model of the entire electric grid. Since then, the electric grid serves its costumers in a centralized fashion with one-way flow of electricity generated from mostly coal-fired stations in a "supply following the demand" manner. However, environmental concerns and the raising electricity demands of 21st century have necessitated an overhaul in the infrastructure and operation management of the traditional electric grid, transitioning it into the "smart grid" of the future. In the smart grid, the power grid is integrated with information and communication technologies (ICT) to meet the continually increasing worldwide electricity demand in an economic, sustainable, secure, and environmentally friendly manner. Particularly, the smart grid is envisioned to support distributed generation from renewable energy resources (RES) and new spectrum of energy consumers (e.g., electric vehicles).

The intermittent nature of the renewable energy resources brings new challenges to the smart grid since the amount and the timing of their generation is not controllable. However, thanks to the integration of the ICT infrastructure, smart grid customers are no longer a passive part of the grid. They can contribute to demand-supply balancing by offering flexibility in their electricity usage in response to variable energy tariffs or financial incentives. Demand response (DR) algorithms are viable solutions to exploit that customer flexibility in a coordinated way and ensure a more reliable network performance. However, the establishment of practical DR algorithms is one of the barriers to their widespread deployment in the smart grid paradigm. A practical DR is: (i) designed and assessed based on accurate models of the real-world coordination problem, and (ii) is applicable to a broad range of coordination problems of similar characteristics with minor modification. This thesis focuses on the realization of practical DR algorithms. It presents contributions in two areas: (i) analysis, modeling, and quantification of the flexibility based on real-world datasets, and (ii) proposing a model-free DR algorithm based on the reinforcement learning (RL) approach.

Energy consumption flexibility is DR's main asset. To realize a practical DR, it is vital to analyze and understand such flexibility. This thesis focuses on analysis and modeling of the flexibility stemming from residential white-good usage and flexibility in electric vehicle (EV) charging sessions. In both of these applications, the timing aspect of the flexibility (i.e., time of availability and duration of the delay in energy consumption) is determined by the owners. Hence, it is influenced by their preferences and life styles. This makes the modeling of the flexibility challenging due to heterogeneity of the customers and uncertainties associated with their lifestyle.

To sharpen the understanding and characterization of residential white-good usage flexibility, we base our studies on a real-world dataset where customers are equipped with smart white-goods and are given the opportunity to use these appliances flexibly. This means that instead of starting an appliance immediately, customers set a flexibility duration (i.e., how long the appliance operation can be delayed) when configuring the device. Hence, the flexibility is characterized by: (i) time of configuration, and (ii) deadline (latest allowed start time of the appliance). Analyzing this dataset, we answer the following questions: (i) do customers exhibit certain behavioral patterns when using their smart white-goods flexibly? (ii) what are the factors influencing the flexibility in smart white-good usage? Using clustering and statistical dependency tests, we conclude that not only do different users exhibit potentially substantially different behavior, but also that such flexibility could be influenced differently by factors such as day-of-the-week or seasons, etc.

Note that since the usage of smart appliances is at its infancy, there are limited number of such real-world datasets where customers are given the opportunity to offer their flexibility. The majority of these datasets are not publicly available due to intellectual property issues. To facilitate data availability to broader range of researchers for realistic design and assessment of DR algorithms, we propose two systematic methodologies to derive generative statistical models of a customer's flexibility behavior in offering smart devices for DR exploitation.

Next, we consider the analysis and quantification of flexibility from a real-world EV charging dataset collected from roadside charging stations. Using a clustering algorithm, we identify three behavioral patterns in terms of EV arrival and departure times: charge near home, charge near work, and park to charge clusters. We then use box and violin plots to further analyze the characteristics of the charging sessions within each cluster and highlight the differences among the clusters over weekends and weekdays in each season. Finally, we propose two flexibility measures to quantify the percentage of the flexibility utilization and an algorithm to determine the amount and duration of the shifted energy. We use the measures and the algorithm to inspect which aspect of flexibility (time and duration of availability or amount of deferrable energy) is more useful at various times of the day.

The timing aspect of the flexibility is of cylindrical nature due to the periodicity of the time of configuration and linearity of the offered flexibility duration. When analyzing and modeling the flexibility in the aforementioned applications, we used probabilistic models defined on linear scales for data modeling (e.g., Gaussian mixture models). This raises a question whether probabilistic generative models using distributions defined on cylinder are better than the linear ones in modeling energy consumption flexibility. To answer this question, we propose a Bayesian approach for estimating the parameters of a mixture distribution defined on cylinder. We then use the developed approach to model the flexibility data from white-good usage and compare the performance with the linear models. The comparison reveals that linear models are better generative models of the underlying data, at least from the perspective of quantitative statistical measures.

Finally, as the second contribution of this thesis, we focus on developing model-free DR algorithms for coordinating the charging demand of collection of electric vehicles. In the model-free approach, the coordination problem is formulated as a Markov decision process (MDP), constituting the state, action, cost/reward, and transition probabilities. Here, transition probabilities are unknown due to the stochasticity of future EV arrivals and their energy requirements. In this paradigm, a coordinating agent learns the best coordinating policy by interacting with the environment by taking actions in a particular state and observing the outcome (i.e., the next state and the immediate reward/cost). One of the challenges in this approach is to have a scalable representation of the state and action while taking the EV couplings into account. We tackle this challenge and formulate a novel MDP with scalable state and action representations. We then use a batch-model RL algorithm to learn the best policy (i.e., a mapping from a state to best action) to coordinate the charging of the EV groups of various sizes. We extensively analyze the effect of the various settings (e.g., sample sizes, time spans of test and training sets) on the performance of the proposed approach. We also show that a policy learned from data of smaller number of EVs can be applied to coordinate the charging of larger groups, with only a small increase in the normalized cost (with respect to the cost of optimum all-knowing solution).

1
Introduction

"The value of an idea lies in the using of it."

–Thomas A. Edison

Climate change and continually increasing energy demand has raised worldwide concerns for energy sustainability and environmental preservations. Various initiatives are put in action to tackle these concerns: the European union (EU) 2030 climate and energy framework [1] targets 40% reduction in greenhouse gas emissions, 27% increase of renewable energy share and 27% energy efficiency improvement by 2030; the EU road map 2050 [2] targets 80% reduction in greenhouse gas emission from 1990 levels; British Colombia's 2010 clean energy act [3] targets 80% reduction in greenhouse gas emission from 2007 levels by 2050; the Paris agreement 2015 [4] within the United Nations Framework Convention on Climate Change (UNFCCC) members has set a long-term goal of keeping the increase in global average temperature to well below 2 degree Celsius above pre-industrial levels and to limit the increase to 1.5 degree Celsius; and united states (US) Energy Independence and Security Act of 2007 [5].

Aforementioned efforts encourage, among other action plans, the use of renewable energy resources (RES) and the electrification of the energy use (with the transportation industry as a notable example). As a result, electricity consumption globally increases at a faster pace than other energy vectors [6] and this electricity would increasingly be supplied by intermittent RES (e.g., wind and solar energy) whose power output is uncontrollable. Therefore the supply can not be adjusted to meet the demand when relying on RES. This leads to mismatch in time between the supply and demand and brings a new challenge to the power grid. However, the conventional power grid was neither designed with the latest technologies in mind nor to comply with the environmental concerns and electricity demands of the 21st century. It operates in a centralized manner where the supply must follow the demand. To accommodate the needs of the 21st century, the power grid should be overhauled from a one-way grid of the past to a decentralized, interconnected and bi-directional infrastructure of the future. This transition is realized by integrating the power grid with information and communication technologies (ICT), moving it towards the smart power grid (or simply, the smart grid). The integration of the power grid with ICT makes the passive consumers of the conventional power grid to become active and adjust their energy demand via demand side management (DSM) solutions to facilitate the "demand following supply" scheme. This postpones and even avoids the costly upgrades to the power grid.

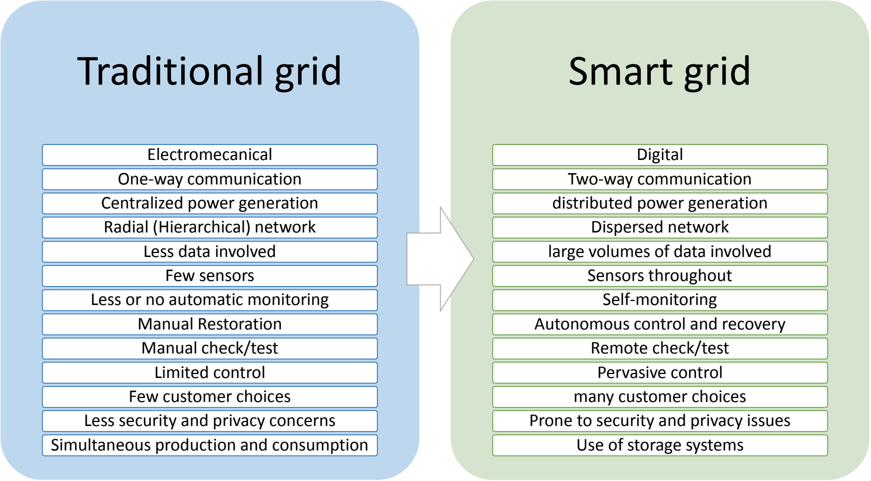
The smart grid and DSM concepts are discussed in more details next.

1.1 Smart Grid

The smart grid is not a new technology but rather an integration of the conventional power grid with ICT and computational intelligence from the generation to the consumption points of electricity [7]. The envisioned objective of the smart grid is to meet the continually increasing worldwide electricity demand in an economic, sustainable, secure and environmentally friendly manner. The US department of energy has the following functional definition for the smart grid [8]: "A Smart Grid is self-healing, enables active participation of consumers, operates resiliently against attack and natural disasters, accommodates all generation and storage options, enables introduction of new products, services and markets, optimizes asset utilization, operates efficiently, and provides power quality for the digital economy". The smart grid may be understood better in light of a comparison with the traditional grid. Figure 1.1 summarizes the differences between the traditional electric grid and the smart grid.

To achieve the aforementioned functionalities, the envisioned smart grid consists of 3 systems [11]: (i) the smart infrastructure system, (ii) the smart management system, and (iii) the smart protection system. These systems (explained next) collaboratively ensure the operation of the smart grid.

The *smart infrastructure system* is responsible for providing a platform for bi-directional flow of electricity to accommodate the injection of power from distributed generators, including customers who could also be producers of electricity. To realize such platform, not only the energy transmission and distribution infrastructure needs to evolve but also the information extracted (via data metering) from various parts of the grid needs to the exchanged though a reliable communication network. Hence, the smart infrastructure system is comprised of energy, informa-



Traditional grid	Smart grid
Electromecanical	Digital
One-way communication	Two-way communication
Centralized power generation	distributed power generation
Radial (Hierarchical) network	Dispersed network
Less data involved	large volumes of data involved
Few sensors	Sensors throughout
Less or no automatic monitoring	Self-monitoring
Manual Restoration	Autonomous control and recovery
Manual check/test	Remote check/test
Limited control	Pervasive control
Few customer choices	many customer choices
Less security and privacy concerns	Prone to security and privacy issues
Simultaneous production and consumption	Use of storage systems

Figure 1.1: Comparing the traditional and the smart grid [9] [10]

tion and communication subsystems as shown in Figure 1.2.

The *smart protection system* provides not only smarter failure protection mechanisms, but also autonomous failure identification, analysis and recovery. It further addresses cyber security issues, and preserves privacy associated with gathering and communicating the metered data.

The *smart management system* uses ICT to leverage various functionalities of the smart infrastructure via advanced management and control services. Demand-supply balance is an integral part of the energy system operation. In the smart grid paradigm, volatile RES and electro-mobility bring new challenges to this balance. A smart management system ensures a supply-demand balance in the grid in the presence of these new challenges via various actors that are responsible for supply and/or demand side management in the grid. In the European grid, these actors are transmission system operators (TSOs), distribution system operators (DSOs), balance responsible parties (BRPs) and aggregators. Figure 1.2 lists the tools the grid actors have in their disposal to perform supply and demand side management. In addition to the listed tools and resources, a market structure that ensures an effective cooperation of grid actors is also an essential component in the smart management system.

The smart management subsystem (and particularly the demand side management) is where this dissertation contributes to. The contributions are underlined in Figure 1.2 and the relevant concepts are explained in the next section.

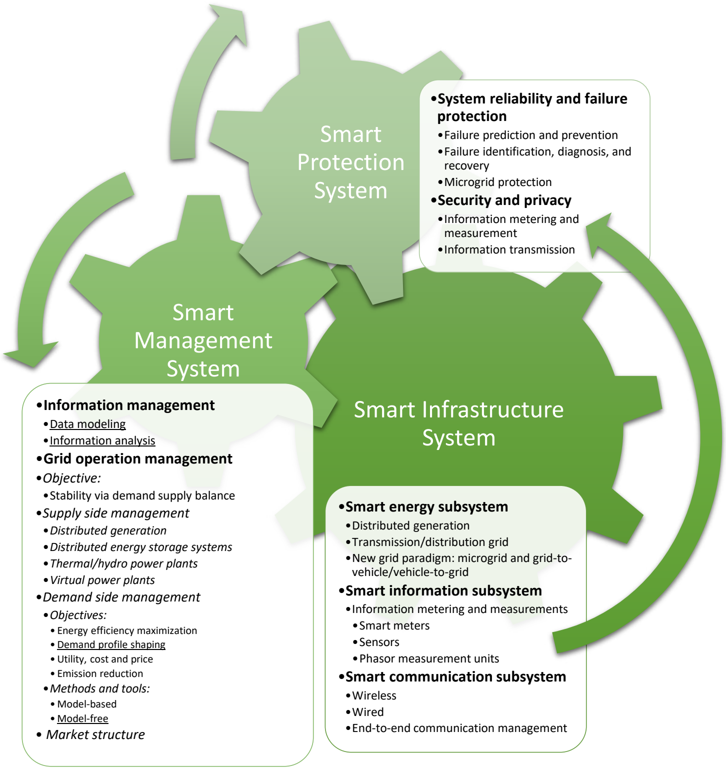

Smart Protection System
•System reliability and failure protection
•Failure prediction and prevention
•Failure identification, diagnosis, and recovery
•Microgrid protection
•Security and privacy
•Information metering and measurement
•Information transmission
Smart Management System
•Information management
•Data modeling
•Information analysis
•Grid operation management
•Objective:
•Stability via demand supply balance
•Supply side management
•Distributed generation
•Distributed energy storage systems
•Thermal/hydro power plants
•Virtual power plants
•Demand side management
•Objectives:
•Energy efficiency maximization
•Demand profile shaping
•Utility, cost and price
•Emission reduction
•Methods and tools:
•Model-based
•Model-free
• Market structure
Smart Infrastructure System
•Smart energy subsystem
•Distributed generation
•Transmission/distribution grid
•New grid paradigm: microgrid and grid-to-vehicle/vehicle-to-grid
•Smart information subsystem
•Information metering and measurements
•Smart meters
•Sensors
•Phasor measurement units
•Smart communication subsystem
•Wireless
•Wired
•End-to-end communication management

Figure 1.2: The components of the smart grid (reproduced from [11] with minor modification of the context and graphics). The underlined parts are areas which this thesis contributes to.

1.2 Demand Side Management

This section explains the demand side management (DSM) as one of the components of the smart management subsystem and motivates the need for DSM. It then focuses on demand response (DR) algorithms (one of the two DSM categories) and explains the benefits it brings to the generation, transmission, distribution and customer's sectors in power grid. It highlights the energy consumption flexibility as DR's main asset and lists the sources it stems from. Next, it presents and explains various DR categories and finally outlines the barriers to the widespread realization of DR algorithms in smart grid.

DSM plays a crucial rule in attainment of a fully functional smart grid with widespread utilization of RES and support for a wide range of energy demands.

DSM refers to planning, implementation and monitoring of activities designed to influence energy consumption at the customer side of the meter [12] to keep a demand-supply balance in the grid. DSM's primary advantage is that it facilitates a cost friendly solution to keep the demand-supply balance in the grid: it is cheaper to intelligently influence a load rather than building a new power plant or installing new storage devices. DSM programs are categorized based on the methodology for securing supply-demand balance into [13]: (i) energy efficiency (EE) programs, and (ii) demand response (DR) algorithms..

The *EE programs* communicate the electricity consumption information to the consumers in order to lower their energy consumption without modifying operating practice and only by prompting adoption of new devices or system upgrades that provide same level of service using less energy [14]. Hence, EE programs typically result in long lasting savings in energy when the end-use equipment is in operation. A popular example of a cost effective EE programs is Non-intrusive load monitoring (NILM). In the NILM paradigm, per-appliance energy consumption is inferred from a single aggregate measurement (communicated from smart meters) using machine learning techniques. Per-appliance consumption information is then communicated to the consumers to assist them to improve their efficiency (e.g., by identifying and replacing old, or faulty devices),

DR algorithms, by promoting the interaction and responsiveness of the customers, aim to instantaneously change the end-user electricity demand pattern to match the supply at a given point in time and are the focus of this dissertation. DR is more accurately defined as [8]: "changes in electric use by demand-side resources from their normal consumption patterns in response to changes in the price of electricity, or to incentive payments designed to induce lower electricity use at times of high wholesale market prices or when system reliability is jeopardized".

1.2.1 Demand response potential benefits

DR algorithms yield many potential benefits [15], spanning from (i) reduction of generation margin and increasing the penetration of intermittent RES, (ii) improving service reliability and delaying/avoiding costly network reinforcement in transmission and distribution grid, (iii) stabilizing energy market, and (iv) bringing financial benefits and improved reliability for customers of the power grid. Each of these points are explained next.

The total capacity of generation in a power grid must be larger than the system maximum demand to guarantee supply during contingencies. This is known as the generation margin (or reserve margin). DR can reduce the generation margin by offering load curtailment during the increase in demand or interruption in generation [16]. Furthermore, DR promotes the increase in the intermittent RES with uncontrollable generations by coordinating the demand to match the generation.

The capacities of the transmission and distribution networks are often underutilized as a part of a preventive measure to withstand infrequent outages. DR can help the power systems to lower the operation costs and reduce the network capacities by offering corrective actions to effectively eliminate overloads that occur, e.g., after outages of circuits. An example of such actions is reduction of demand in certain locations of the grid. In distribution networks, DR can further be used to manage network constrains and offer: reduction of network expansion investments; increase in share of distributed generations; mitigate the voltage-constrained power transfer problems and alleviate network congestions [16].

DR can also stabilize the energy market by dispersing the energy consumption and flattening the load curves, hence reducing the overall generation cost. The reduction in generation cost translates to lower marker prices and benefits all consumers, including the ones who did not participate in DR programs. In addition, the improved elasticity in demand due to DR limits the extent and frequency of price spikes and eliminates the exercise of marker power by generators in wholesale electricity markets [8].

From the customer's perspective, DR gives the customers more options for energy cost management and an opportunity to reduce their electricity bill, receive payments for participation in DR and even sell their local generations to the grid. Customers also enjoy the reliability in form of lower probability of the involuntarily curtailments [8, 14].

1.2.2 Flexibility as a demand response asset

Consumer energy consumption flexibility is the prerequisite foundation of the entire DR paradigm. Flexibility is characterized by the time, amount and duration of a deviation from normal consumption pattern that a consumer is willing to incur when participating in a DR program. Flexibility of large industrial and some of the commercial consumers was already being exploited even before the inception of smart grid. The integration of the ICT infrastructure not only allows residential consumers to also participate in DR by offering their flexibility, but also fosters participation from new commercial sectors (such as electric vehicle charging lots) whose demands are affected by their customer's preferences.

This section considers two sources of flexibility: (i) the residential households which consume 37.4% [17] and 29% [18] of the electricity in US and EU respectively, and (ii) electric vehicles (EV) (in both residential and commercial sectors) which are envisioned to reach full penetration by 2050 in Europe [19].

Residential flexibility stems from various loads:

(1) *Thermostatically controlled loads (TCL)*: This category includes the loads that store energy in a thermal buffer (e.g., air conditioners and heaters).

TCLs should be operated within a predefined temperature band, set by customer comfort and safety constraints. While respecting this temperature band, the regular operating cycle of these devices can be altered to provide flexibility for DR exploitation.

(2) *Shiftable loads (smart appliances)*: Devices whose operation can be delayed by customers. Some examples of such devices are washing machines, dishwashers and tumble dryers. The timing and the duration of the delay depends on customer preferences and lifestyle.

(3) *Storage/batteries*: Rechargeable batteries are today's most widespread electrical energy storage devices [20]. They store electrical energy in the form of chemical energy and allow bidirectional flow of energy. Batteries can be charged/discharged with consideration of the battery aging (typically staying within 20% and 80% of the state-of-charge) to contribute to DR objectives. Typical residential storage are battery packs in electric vehicles or stand alone domestic batteries. Note that charging/discharging of the electric vehicle batteries should additionally take into account the customer preferences (in terms of set deadlines to completing the charging).

EVs are not only seen in residential sectors but also in the commercial sector (e.g., EV charging points in shopping centers and roadside parking spots). In fact, EVs are rapidly growing in numbers in the transportation system, driven by sustainability and environmental initiatives. According to the European Environment Agency, the use of conventionally fueled cars in urban transport will be halved by 2030 and phased out entirely in cities by 2050 [19]. EVs have brought various challenges to the power grid but also constitute a great potential as a source of distributed storage whose flexibility can be utilized by DR.

1.2.3 Demand response methods

Since the inception of the smart grid, DR algorithms have received considerable attention. Various DR algorithm have been proposed for tackling demand-supply mismatch in various use cases. This section presents DR categorizations from two perspectives: (1) categorization in terms of the method for motivating customer's participation in DR, and (2) categorization in terms of the mathematical methods used for realizing DR.

In terms of the method for motivating customer participation, DR is characterized into: (1) price-based DR, and (2) incentive-based DR.

In the *price-based DR algorithms*, the participant's electricity demand patterns are influenced by variable electricity price signals communicated to them. There are three pricing schemes in this category: (i) Time-of-use (TOU) pricing: rates with static price blocks with minor variation over time of the day, (ii) Critical peak

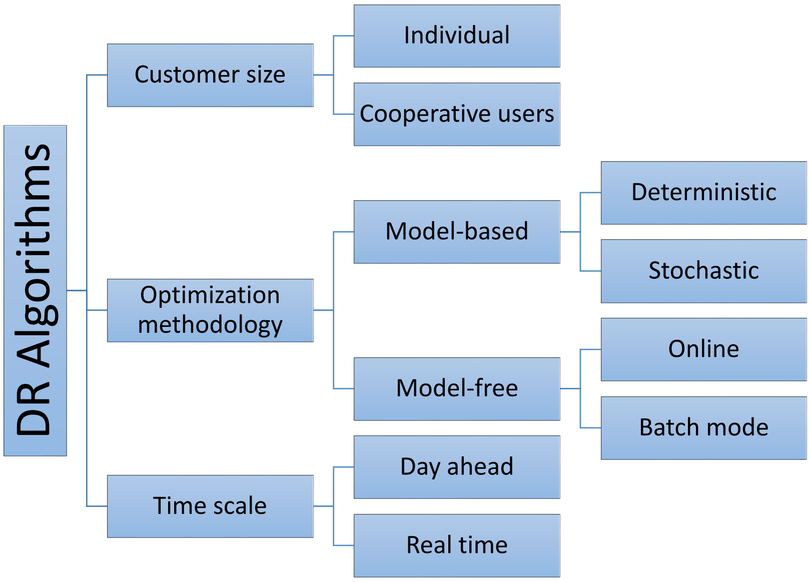


Figure 1.3: DR classification based on the optimization approach

pricing (CPP): a pre-specified higher rate triggered by the utility during the peak demand for few hours only, and (iii) Real-time pricing (RTP): dynamic rates that vary in accordance with wholesale electricity market prices.

In the *incentive-based DR algorithms*, participants are compensated for giving some level of control over their electricity consumption or being available to reduce their usage upon request. Incentive-based DR is typically used to mitigate system stress (e.g., during grid congestions or for operational reliability improvement). Several schemes exist in this DR category [21]: (i) Direct load control: participants allow a degree of control by utility over specific devices (ii) Interruptible/curtailable rates: participants offer limited load shedding when needed to ensure reliability (iii) Emergency DR programs: participants are rewarded for voluntary response to emergency signals (iv) Capacity market programs: participants offer limited load shedding as substitutes for capacity increase (v) Demand bidding programs: participants bid for curtailing at attractive prices.

Another categorization perspective separates DR algorithms based on the mathematical methods used for realizing them. Demand coordination in DR translates into an optimal control problem (OCP) [22]. In terms of the optimization control strategies, the customer scope and time-scale of the OCP, the DR algorithms are categorized as shown in Figure 1.3. The DR algorithms can either target an individual consumer or coordinate the consumption of a group of consumers. DR algorithms can also make coordinating decisions in real-time or for a predefined (typically day-long) horizon. In terms of the mathematical methods for finding the optimum solution (coordination decisions), the DR algorithms are classified into model-based and model-free approaches, which are discussed next.

1.2.3.1 Model-based DR

In initial DR studies, the demand response problem is usually cast as a model predictive control (MPC) approach where an explicit mathematical model of the process dynamics is used to predict the system behavior (e.g., physical characteristics of the devices or consumer energy consumption behavior). An optimizer then chooses the best decision in the sense of a cost (e.g., the customer's electricity bill or the energy provider's cost) while respecting the constraints (e.g., customer preferences, distributed energy resource constraints and energy market constraints). Various optimizers are used in context of the MPC approach, including convex optimization, stochastic optimization, particle swarm optimization, dynamic programming, and game theory. A review of these approaches is presented in [23, 24].

In the model-based DR approaches, selecting accurate models and estimating their parameters is crucial for the efficiency and the reliability of the resulting DR algorithm. However, the modeling task is particularly challenging in the smart grid paradigm due to uncertainty stemming from heterogeneity of the end user loads, difference in user behavioral patterns and uncertainty surrounding their behavior. Furthermore, model-based DR algorithms are difficult to transfer from one scenario to the other, since the model designed for one group of users or applications is likely to require customization/tweaking for application to different groups.

1.2.3.2 Model-free DR

Unlike model-based approaches, model-free DR is data-driven and does not require a model of system dynamics. This promotes the incorporation of the smart grid stochasticity without the need for expert knowledge and sustains generalizability of the DR algorithms.

Reinforcement learning (RL) is a popular recent approach in the model-free DR paradigm. In the RL-based approach, the DR problem is formulated as a Markov decision process (MDP). A coordinating agent interacts with the environment (i.e., DR participating customers, energy providers, energy market prices, etc.) and takes control actions while aiming to maximize the long term expected reward (or to minimize the long term expected cost). In other words, the agent learns an optimum policy (i.e., a mapping between states and actions that maximizes/minimizes the received reward/cost) by taking actions and observing the outcomes (i.e., states) and the rewards/costs in an iterative process. The DR objective (e.g., load flattening, load balancing) is achieved by appropriately designing the reward/cost signal.

One of the downsides of the RL-based approach is that it is challenged when the state space and/or action spaces of the control problem are very large or continuous [22]. In such cases, RL has to be combined with so called function-approximation techniques (e.g., artificial neural networks or tree based methods) allowing one to generalize over the state-action space based on the observed data (which is typically a very sparse sample of the full state-action space).

1.2.4 Demand response barriers

Despite the promising potential benefits of DR and availability of the monitoring and communication technologies [25], its widespread deployment in the grid has been slow due to various barriers. There barriers can be looked into from two (interrelated) perspectives:

(i) Barriers preventing the optimal use of DR resources: These barriers stem from external factors (not related to the practical modeling and methodological implementation of DR algorithm) such as lack of market and regulatory framework, lack of social understanding of the DR technology (and how to benefit from it) in the consumer side, and lack of widespread deployment of the infrastructure (e.g., smart meters smart appliances).

(ii) Establishment of practical DR algorithms: The lack of experience (due to limited availability of real-world data where consumers demonstrate flexibility) and the consequent need for unrealistic assumptions when modeling and evaluating DR algorithms hinders the implementation of practical DR algorithms [25] [26].

This dissertation focuses on the latter.

1.3 Motivations and contributions

A practical demand response (DR) is vital in the ongoing transition into smart grid. But what is a “practical” DR? The answer to this question is twofold:

(1) A practical DR is designed based on accurate models of the real-world problem (e.g., consumer preferences). It avoids unrealistic assumptions and is developed and assessed with consideration of realistic characteristics of the coordination problem.

(2) A practical DR should not be scenario specific. Instead, it should be applicable to a broad range of coordination problems with similar characteristics, requiring little or no modification.

This thesis presents two contributions to the realization of a practical DR algorithms. First, an in-depth analysis of two real-world datasets to (a) characterize and quantize the flexibility as DR's main asset, to improve the efficiency of DR design and assessment, and (b) development of probabilistic models of their data to be used for synthetic data generation for simulation propose (hence, these models are called the generative models in this thesis). Second, a model-free DR solution that learns from real-world data and generalizes to broader range of coordination problems. These contributions are explained in more detail next.

As established in Section 1.2.4, one of the challenges in realizing practical DR algorithms is the lack of real-world datasets which facilitate realistic assumptions and accurate models during the algorithm design. In fact, there is a limited amount of real-world data available where customers are given the opportunity to offer their energy consumption flexibility for DR exploitation. Moreover, the majority of these datasets are not publicly available due to intellectual property concerns. Hence, the first contribution of this thesis is:

- A realistic and insightful analysis and modeling of two real-world datasets: LINEAR (where consumers use their smart white-good appliances flexibly), and ElaadNL (a large scale set of EV charging transactions at roadside parking spots)
    - Implementation of clustering to identify behavioral patterns,
    - Analysis of influential factors (e.g., seasonal changes, weekend/weekdays) on the customer energy consumption behavior and the offered flexibility,
    - Definition of measures for quantification of energy consumption flexibility,
    - Development of probabilistic generative models to facilitate synthetic data generation.

As a second contribution, this thesis focuses on developing a model-free DR algorithm using RL. As explained in Section 1.2.3.2, model-free DR algorithms do not need accurate models of the environment and learn the optimal control policy by interacting with the environment. The latter implies taking actions and observing the outcome (i.e., the resulting state and the cost/reward), hence fostering practical DR solutions. However, one of the challenges of applying RL-based approaches in real-world applications is that the majority of such applications have high-dimensional state and action representations as well as large state-action spaces. This challenges the learning algorithms in RL-based approaches. This thesis aims to tackle the aforementioned challenge for coordination of a group of EV charging stations. The second contribution is detailed below.

- Proposing a model-free DR solution for a collection of EV charging stations:
  - Formulation of a Markov decision process (MDP) with a relatively compact aggregate state and action space representation that is generalizable to various EV collection sizes,
  - Adoption of batch reinforcement learning with function approximation to find the best EV charging policy,
  - Experimental and quantitative exploration of the performance of the proposed RL approach, using real-world data.

The next section explains the chapter specific contributions and outlines the structure of the remaining part of this thesis.

1.4 Outline

This dissertation is composed of a number of publications that were realized within the scope of this PhD. The selected publications provide an integral and consistent overview of the work performed. A full list of realized publications is given in Section 1.5. Chapters 2-4 following this introduction provide in-depth analysis of power grid customers' behavioral patterns using real-world datasets and construction of generative models to represent them. Chapter 5 represents a data driven control approach to exploit flexibility available from that behavior.

Chapter 2 focuses on characterizing the flexibility stemming from residential white-good usage. It is based on the data from the LINEAR project where customers are provided with smart white-goods and their flexible usage of these appliances is recorded in form of time of configuration of the smart appliance and the deadline (latest start time) set by the customer during the configuration. It contributes with: (1) a new quantitative specification of flexibility, (2) two systematic methodologies for modeling individual customer behavior, (3) evaluation of the proposed models in terms of how accurately the data they generate corresponds with real world customer behavior, and (4) a basic analysis of factors influencing the flexibility behavior based on statistical tests.

Chapter 3 performs an in-depth analysis of the flexibility characteristics of EVs based on a reasonably large real-world dataset that has records of roadside EV charging sessions. The EV flexibility indicates to what extent the charging load can be coordinated (i.e., to flatten the load curve or to utilize renewable energy resources). Through clustering the arrival and departure time combinations, three behavioral patterns are identified in the EV charging data. A systematic analysis of the characteristics of the charging sessions in each behavioral cluster on weekdays and weekends and across various seasons is provided. Finally, an algorithm and two measures are proposed to investigate how flexibility (in terms of amount, time and duration of the shifted energy) is exploited and which aspect of the flexibility (time and duration of availability or amount of deferrable energy) is more useful at various times of the day.

A customer's energy consumption flexibility is defined in terms of amount, time and duration of availability. It is established in Chapters 2 and 3 that the timing of flexibility is circular in nature: configuration times of real-world observations form clusters which cross over from one day to the next (across the midnight boundary). Chapters 2 and 3 avoid the cylindrical representation by defining a heuristic algorithm that identifies the middle of the largest gap on the circular axis to wrap the data around and proceed to modeling using probabilistic models defined on linear scales. However, such heuristic algorithms might fail in situations where such a reference point is challenging or impossible to find. This raises the question whether probabilistic generative models using cylindrical distribution are better than the linear ones for modeling energy consumption flexibility.

To answer the above-raised question, Chapter 4 proposes a Bayesian approach based on Markov Chain Monte Carlo (MCMC) to estimate the parameters of Abe-Ley mixture distribution. Abe-Lay distribution is a cylindrical distribution based on the combination of Weibull and sine-skewed von-Mises. The choice of the distribution is motivated by its various merits including compared to the other existing cylindrical distributions which are outlined as: (i) flexible shapes, (ii) cross-correlation among linear and circular variables, (iii) well-known marginal and conditional distributions and (iv) support of data skewness. The proposed approach in Chapter 4 is then used to model the residential white-good usage flexibility and compare it with the proposed linear models of Chapter 1 in Appendix A.

Chapter 5 has a different goal than the previous 3 chapters. Instead of data analysis and modeling, it aims to propose a model-free DR approach for coordinating the charging of a collection of EVs. It formulates the coordination problem into a scalable and generalizable MDP with compact representation of state and action. Batch-mode RL is used to learn the optimum policy. The performance of the proposed reinforcement learning approach is evaluated using real-world data, answering the following research questions: (Q1) What are appropriate durations for training period and number of sampled trajectories from the decision trees? (Q2) How does the RL policy perform compare to an optimal all-knowing oracle algorithm? (Q3) How does that performance vary over time using realistic data? and (Q4) Does a learned approach generalize to larger EV collections?

Finally, Chapter 6 concludes the thesis.

1.5 Publications

The research results obtained during this PhD research have been published in scientific journals and presented at a series of international conferences. The following list provides a complete overview of the publications during my PhD research.

1.5.1 Publications in international journals (listed in the Science Citation Index)[1]

1. **Nasrin Sadeghianpourhamami**, Thomas Demeester, Dries F. Benoit, Matthias Strobbe and Chris Develder. *Modeling and analysis of residential flexibility: Timing of white good usage.* Applied Energy, Vol. 179, Oct. 2016, pp. 790-805.

2. **Nasrin Sadeghianpourhamami**, Joeri Ruyssinck, Dirk Deschrijver, Tom Dhaene and Chris Develder. *Comprehensive feature selection for appliance classification in NILM.* Energy and Buildings, Vol. 151, Sep. 2017, pp. 98-106.

3. **Nasrin Sadeghianpourhamami**, Nazir Refa, Matthias Strobbe and Chris Develder. *Quantitive analysis of electric vehicle flexibility: A data-driven approach.* International Journal of Electrical Power & Energy Systems, Vol. 95, Feb. 2018, pp. 451-462.

4. **Nasrin Sadeghianpourhamami**, Dries F. Benoit, Dirk Deschrijver, Chris Develder. *Bayesian modeling of cylindrical data using Weibull-Sine-Skewed-Von-Mises mixtures.* Submitted to Applied Mathematical Modeling (Under Second round of revision), 2018.

5. **Nasrin Sadeghianpourhamami**, Johannes Deleu and Chris Develder. *Definition and experimental evaluation of model-free coordination of electrical vehicle charging with reinforcement learning.* Submitted to IEEE Transactions on Smart Grid, 2018.

1.5.2 Publications in international conferences (listed in the Science Citation Index)[2]

1. **Nasrin Sadeghianpourhamami**, Matthias Strobbe and Chris Develder. *Real-world user flexibility of energy consumption: Two-stage generative model construction.* In Proc. 31st ACM/SIGAPP Symp. Applied Computing (SAC 2016), Pisa, Italy, 4-8 Apr. 2016. pp. 2148-2153.

[1] The publications listed are recognized as ‘A1 publications’, according to the following definition used by Ghent University: A1 publications are articles listed in the Science Citation Index Expanded, the Social Science Citation Index or the Arts and Humanities Citation Index of the ISI Web of Science, restricted to contributions listed as article, review, letter, note or proceedings paper.

[2] The publications listed are recognized as ‘P1 publications’, according to the following definition used by Ghent University: P1 publications are proceedings listed in the Conference Proceedings Citation Index - Science or Conference Proceedings Citation Index - Social Science and Humanities of the ISI Web of Science, restricted to contributions listed as article, review, letter, note or proceedings paper, except for publications that are classified as A1.

2. Chris Develder, **Nasrin Sadeghianpourhamami**, Matthias Strobbe and Nazir Refa. *Quantifying flexibility in EV charging as DR potential: Analysis of two real-world data sets.* In Proc. 7th IEEE Int. Conf. Smart Grid Communications (SmartGridComm 2016), Sydney, Australia, 6-9 Nov. 2016, pp. 600-605.

3. **Nasrin Sadeghianpourhamami**, Dries F. Benoit, Dirk Deschrijver and Chris Develder, *Modeling real-world flexibility of residential power consumption: Exploring the cylindrical WeiSSVM distribution.* In Proc. 9th ACM Int. Conf. Future Energy Systems (e-Energy 2018), 12-15 Jun. 2018.

4. **Nasrin Sadeghianpourhamami**, Johannes Deleu and Chris Develder. *Achieving scalable model-free demand response in charging an electric vehicle fleet with reinforcement learning.* In Proc. 9th ACM Int. Conf. Future Energy Systems (e-Energy 2018), 12-15 Jun. 2018.

References

[1] *2030 climate and energy framework.* https://ec.europa.eu/clima/policies/strategies/2030_en. Accessed: 2018-08-27.

[2] *Roadmap 2050: a practical guide to a prosperous, low-carbon Europe*. Technical report, European Climate Foundation, 2010.

[3] *Clean energy act.* http://www.bclaws.ca/civix/document/id/consol24/consol24/00_10022_01. Accessed: 2018-08-27.

[4] *The Paris agreement.* https://unfccc.int/process-and-meetings/the-paris-agreement/the-paris-agreement. Accessed: 2018-08-27.

[5] *Energy independent and security act of 2007.* https://www.congress.gov/110/plaws/publ140/PLAW-110publ140.pdf. Accessed: 2018-08-27.

[6] *Global energy statistics yearbook 2018.* https://yearbook.enerdata.net/electricity/electricity-domestic-consumption-data.html. Accessed: 2018-08-25.

[7] H. Gharavi and R. Ghafurian. *Smart Grid: The Electric Energy System of the Future*. Proceedings of the IEEE, 99(6):917–921, June 2011. doi:10.1109/JPROC.2011.2124210.

[8] Q. Qdr. *Benefits of demand response in electricity markets and recommendations for achieving them.* Technical report, US Department of Energy, Washington, DC, USA, 2006.

[9] M. L. Tuballa and M. L. Abundo. *A review of the development of smart grid technologies*. Renewable and Sustainable Energy Reviews, 59:710–725, 2016. doi:10.1016/j.rser.2016.01.011.

[10] H. Farhangi. *The path of the smart grid.* IEEE Power and Energy Magazine, 8(1):18–28, January 2010. doi:10.1109/MPE.2009.934876.

[11] X. Fang, S. Misra, G. Xue, and D. Yang. *Smart grid — the new and improved power grid: a survey*. IEEE Communications Surveys Tutorials, 14(4):944–980, 2012. doi:10.1109/SURV.2011.101911.00087.

[12] L. Gelazanskas and K. A. Gamage. *Demand side management in smart grid: a review and proposals for future direction*. Sustainable Cities and Society, 11:22–30, 2014. doi:10.1016/j.scs.2013.11.001.

[13] M. Behrangrad. *A review of demand side management business models in the electricity market*. Renewable and Sustainable Energy Reviews, 47:270–283, 2015. doi:10.1016/j.rser.2015.03.033.

[14] P. Siano. *Demand response and smart grids—a survey*. Renewable and Sustainable Energy Reviews, 30:461–478, 2014. doi:10.1016/j.rser.2013.10.022.

[15] A. Safdarian, M. Fotuhi-Firuzabad, and M. Lehtonen. *Demand response from residential consumers: potentials, barriers, and solutions*. In Smart Grids and Their Communication Systems, pages 255–279. Springer, 2019.

[16] G. Strbac. *Demand side management: benefits and challenges*. Energy Policy, 36(12):4419–4426, 2008. Foresight Sustainable Energy Management and the Built Environment Project. doi:10.1016/j.enpol.2008.09.030.

[17] *Electricity explained: use of electricity*. https://www.eia.gov/energyexplained/index.php?page=electricity_use. Accessed: 2018-09-04.

[18] *Final energy consumption by sector and fuel*. https://www.eea.europa.eu/data-and-maps/indicators/final-energy-consumption-by-sector-9/assessment-1. Accessed: 2018-09-04.

[19] *Electric vehicles in Europe*. Technical report, European Environment Agency, 2016.

[20] X. Hu, C. Zou, C. Zhang, and Y. Li. *Technological developments in batteries: a survey of principal roles, types, and management needs*. IEEE Power and Energy Magazine, 15(5):20–31, Sept 2017. doi:10.1109/MPE.2017.2708812.

[21] P. Palensky and D. Dietrich. *Demand side management: demand response, intelligent energy systems, and smart loads*. IEEE Transactions on Industrial Informatics, 7(3):381–388, Aug 2011. doi:10.1109/TII.2011.2158841.

[22] D. Ernst, M. Glavic, F. Capitanescu, and L. Wehenkel. *Reinforcement learning versus model predictive control: a comparison on a power system problem*. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 39(2):517–529, April 2009. doi:10.1109/TSMCB.2008.2007630.

[23] R. Deng, Z. Yang, M. Chow, and J. Chen. *A survey on demand response in smart grids: mathematical models and approaches*. IEEE Transactions on Industrial Informatics, 11(3):570–582, June 2015. doi:10.1109/TII.2015.2414719.

[24] S. Kakran and S. Chanana. *Smart operations of smart grids integrated with distributed generation: a review*. Renewable and Sustainable Energy Reviews, 81:524–535, 2018. doi:10.1016/j.rser.2017.07.045.

[25] N. O´Connell, P. Pinson, H. Madsen, and M. O´Malley. *Benefits and challenges of electrical demand response: a critical review*. Renewable and Sustainable Energy Reviews, 39:686–699, 2014. doi:10.1016/j.rser.2014.07.098.

[26] S. Nolan and M. O´Malley. *Challenges and barriers to demand response deployment and evaluation*. Applied Energy, 152:1–10, 2015. doi:10.1016/j.apenergy.2015.04.083.

2
Modeling and Analysis of Residential Flexibility: Timing of White Good Usage

In this chapter, we focus on modeling and quantification of the energy consumption flexibility stemming from residential white-good appliances. We base our studies on a real-world dataset where customer are given the smart white-goods and have the chance to configure their appliances flexibly (i.e., set a time they expect the appliance operation to finish). We analyze the timing aspect of such flexibility (i.e., time of availability and duration of permitted delay in energy consumption), which is influenced by consumer habits and lifestyles, to identify behavioral patterns. In addition, since there are a limited number of datasets with flexible appliance usage and most of them are not publicly available due to intellectual properties issues, we construct generative probabilistic models from LINEAR dataset. These generative models can be used to generate data needed for a practical assessment of DR impact, while avoiding the intellectual property issues. Additionally, the proposed models facilitate control over data generation (e.g., data size, different mixes of types of observed behaviors).

⋆ ⋆ ⋆

N. Sadeghianpourhamami, T. Demeester, D.F. Benoit, M. Strobbe and C. Develder.

Published in Applied Energy, Oct. 2016.

Abstract Challenges that smart grids aim to address include the increasing fraction of supply by renewable energy sources, as well as plain rise of demand, e.g., by increased electrification of transportation. Part of the solution to these challenges lies in exploiting the opportunity to steer residential electricity consumption (e.g., for flattening the peak load or balancing the supply and demand in presence of the renewable energy production). To optimally exploit this opportunity, it is crucial to have insights on how flexible the residential demand is. Load flexibility is characterized by the amount of power, time of availability and duration of deferrable consumption. Residential flexibility however, is challenging to exploit due to the variation in types of customer loads and differences in appliance usage habits from one household to the other. Existing analyses of individual customer flexibility behavior in terms of timing are often based on inferences from surveys or customer load patterns (e.g., as observed through smart meter data): there is a high level of uncertainty about customer habits in offering the flexibility. Even though some of these studies rely on real world data, only few of them have quantitative data on actual flexible appliance usage, and none of them characterizes individual user behavior. In this paper, we address this gap and contribute with: (1) a new quantitative specification of flexibility, (2) two systematic methodologies for modeling individual customer behavior, (3) evaluation of the proposed models in terms of how accurately the data they generate corresponds with real world customer behavior, and (4) a basic analysis of factors influencing the flexibility behavior based on statistical tests. Experimental results for (2)–(4) are based on a unique dataset from a real-life field trial.

2.1 Introduction

The rapid integration of renewable energy sources into the power grid and their intermittent nature has created a need for flexibility in energy demand. Flexibility is generally regarded as the amount of load that is shiftable over various time scales and is quantized by 3 parameters [1]: (1) the *amount* of deferrable energy (i.e., the amount of energy that can be delayed without jeopardizing customer convenience or quality of the task to be fulfilled by a smart device), (2) the *time* of availability (i.e., the time at which a customer offers the device flexibility for exploitation), and (3) the *deadline* to exploit the offered flexibility (i.e., the maximum allowable delay for the energy consumption). Once flexibility is known and thus adequately characterized, it can be utilized by demand response (DR) algorithms to coordinate the demand-supply balance in the network. Various DR algorithms have already been proposed to exploit such flexibility: for an overview, we refer to [2] and [3]. Hence, proposing a new DR algorithm is not our focus.

Our main objective is to characterize and model the flexibility as DR's main asset, to improve the efficiency of DR assessment. One of the main challenges in the widespread deployment of DR algorithms (especially in the residential sector) is the uncertainty surrounding their impact [4, 5]. A poor understanding of flexibility characteristics as DR's main asset leads to inefficient DR assessment and uncertain conclusions (i.e., accurate evaluation of DR algorithms is impossible without in-depth analysis of the flexibility parameters). The outcome of our flexibility modeling and characterization (which is based on a unique dataset from a real-life field trial) can foster more realistic assessment of the potential impact of DR algorithms and pave the way to their realization in smart grid.

Flexibility of large industrial customers has already been extensively assessed and exploited by long standing programs (e.g., [6–10]). Since the inception of the smart grid, that industrial flexibility has increasingly been complemented by residential flexibility. Residential customers form a promising source of flexibility due to their widespread distribution and substantial share of electricity market and hence are the focus of our study here.

Residential flexibility however, is challenging to characterize due to the large variety of appliances and their diverse consumption patterns, as well as the uncertainty associated with appliance usage due to different usage habits among various households. A substantial amount of research has analyzed the flexibility potential of residential customers from various perspectives. A brief overview is presented in the next section.

2.1.1 State of the Art in Residential Flexibility Assessment

Methods to assess residential flexibility potential in literature can be categorized into two main streams, according to the objective they pursue: *DR-based* and *DR-independent* methods. The *DR-based* methods are often tailored to the underlying DR scheme (i.e., price-based or incentive-based DR) and their main objective is to model the responsiveness of customers to price signals or incentive programs. In price-based DR schemes, an elasticity matrix models customer flexibility as changes in aggregated demand in response to price changes [11–14]. However, an elasticity matrix can only measure the aggregated flexibility potential and not the appliance specific flexibility. Price-demand models based on mixed integer linear programming (e.g., [15]) or probabilistic models (e.g., [11, 16]) are proposed to predict the customer consumption patterns from the appliance level perspective in response to dynamic prices. For incentive-based DR schemes, Hu *et al.* [17] propose a stochastic model to assess the probability distribution of residential demand in response to certain incentives. The proposed residential responsive demand model is formulated with consideration of the customer portfolio and household characteristics obtained from time-of-use surveys, rather than actual measurement of real behavior.

One of the limitations of DR-based approaches is that any quantification and assessment of flexibility potential is inevitably influenced by the underlying DR algorithm. Additionally, the impact of the underlying DR algorithm on the flexibility is not measurable. In other words, customers might exhibit different flexibility behavior when assessed with other DR-based methods. Hence, the outcome of the analysis of a particular DR-based method cannot be employed to reliably assess the impact of different DR algorithms. Instead, DR-independent methods (including the modeling approaches in this paper), offer an unbiased analysis where the customer flexibility behavior is not influenced by the specifics of any DR algorithm.

In *DR-independent* methods, the main objective is to model customer flexibility potential (independent of the underlying DR scheme) and subsequently use the model to assess the potential impact of DR algorithms on peak load reduction or demand-supply balancing. Some of these methods are derived merely based on appliance energy usage patterns that are either obtained from sub-metering of household appliances [18] or assumed by studying the characteristics of the various appliances [19]. Analyzing the flexibility potential based on appliance energy usage patterns provides insights about the potential amount of deferrable energy of each appliance. However, it does not completely characterize the flexibility potential because customer behavior, affecting the time of availability and deadline to exploit the offered flexibility, is not accounted for. One of the popular means to take into account customer appliance usage habits in the flexibility model is collecting time-of-use surveys. Laicane *et al.* [20] performed a time-of-use survey on a four-person household to determine its appliance usage behaviors, particularly for washing machine and dishwasher, to quantify the flexibility potential. The model was then used to shift appliance usage accordingly for peak load reduction. Safdarian *et al.* [21] used a similar approach on 1600 Finnish households to assess the benefits of demand response on the operation of distribution networks. However, time-of-use surveys may be inaccurate in modeling the customer appliance usage habits because they indicate the self-reported behavior of the customer, which may differ from the real behavior.

Another approach taken by DR-independent methods is to obtain a time series estimate of flexibility of residential customers based on the clustering of their load profiles. Kouzelis *et al.* [22] proposed a methodology for analyzing the flexibility potential of residential heat pumps in a probabilistic way from the aggregated load profile of the customers. The proposed methodology compares the load profile of the flexible customer with electrically similar non-flexible customers by means of clustering the customer load profiles and then statistically infers the flexibility potential thereof. Labeeuw *et al.* [23] also used clustering of customer load profiles to derive a time series estimation of load curves and determine demand reduction potential of wet appliances in terms of amount of deferrable load only (without assessing the flexibility duration). They additionally incorporated attitude measurements based on questionnaires in their studies to account for customers willingness to participate in DR based on survey data. Despite valuable contributions of these approaches in terms of amount of deferrable energy and time of availability, they do not give any assessment of the deadline to exploit the flexibility due to limitations in their measurements.

In both of the aforementioned DR-based and DR-independent categories, modeling of customer responsiveness to participate in DR algorithms is not based on real-word scenarios where households are provided with smart appliances and required to configure their appliances flexibly. Hence, the uncertainty about limitations of DR algorithms due to the differences in customers' real-life (power consumption) habits remains largely unresolved. To address this gap, Kobus *et al.* [24] conducted a longitudinal study for one year over 77 Dutch households. Each household was given a smart washing machine, and an energy management system that received daily dynamic prices. The customers' behavioral changes with respect to a reference group was then studied for a full year to explore the potential role of smart appliances in shifting real electricity demand of smart washing machines in response to dynamic tariffs. Still, a limitation of this valuable work is that the analysis is tailored specifically to the underlying DR scheme.

D'Hulst *et al.* [25] also have based their analysis on a real world scenario where customers are provided with a platform to operate their smart devices and offer their flexibility for DR exploitation. The probabilistic analysis of time of configuration as well as the duration of provided flexibility is performed for various appliances. Their analysis is based on the LINEAR pilot project [26] in which 15 minute based measurements were collected from 418 appliances in 186 Belgian households. The analysis in [25] was independent of the underlying DR scheme that harnessed the flexibility potential, hence inferring a better estimation. However, the analysis of the flexibility potential is performed on an aggregated level and does not model individual household flexibility.

2.1.2 Motivation and Contributions

The residential flexibility potential has been extensively analyzed in terms of the amount of shiftable energy. However, the assessment of customer appliance usage habits in terms of when and for how long the flexible device is available for DR exploitation, is usually based on survey data and this may limit the analysis accuracy. Although D'Hulst *et al.* [25] provide a probabilistic analysis of the time of the configuration and deadline to exploit the flexibility, the analysis is at the aggregated appliance level. Also, [25] is limited to analysis only: they do not offer a generative model (that could be used, e.g., to generate data that is representative of reality, for simulation of DR). Such a generative model of (individual) customer flexibility based on real world data is missing in the literature. We have addressed this gap by modeling the residential flexibility in terms of configuration time and deadline, for individual households and appliances.

The primary motivation to develop a *generative model* of customer flexibility behavior, derived from a real dataset, is that such a model sharpens the definition of flexibility and fosters a more realistic analysis of residential flexibility potential by tackling not only the appliance load patterns, but also the uncertainties due to their owner's willingness to offer the available flexibility (for exploitation by DR algorithms). Additionally, a parametric representation of customers allows for the comparison and selection of relevant customers for energy efficiency programs or DR participation, without jeopardizing their privacy. Finally, parametric models can be utilized for synthetic data generation, hence eliminating the need for time-of-use surveys and resulting in more realistic assessment of the DR algorithms' efficiency in harnessing the residential flexibility for various objectives.

As a secondary objective, our modeling allows for examination of factors that might influence the customer flexibility behavior. Identification of the *underlying factors* influencing the customer flexibility behavior allows for more realistic assumptions about potential flexibility. It also may help to make more accurate predictions of customer flexibility behavior.

Our analysis is based on real-world smart appliance usage (washing machines, dishwashers and tumble dryers) as collected in the LINEAR pilot project [26] and offers these contributions:

1. We take a new perspective and instead of indirectly inferring flexibility from load profiles or time-of-use surveys, we use direct measurement of flexible configuration times and deadlines to characterize flexibility (Section 2.2.1),

2. We propose two systematic methodologies to derive generative models for customer flexibility behavior from data (Sections 2.2.2 and 2.2.3),

3. We test the accuracy of the proposed models in representing the real world data using sampling and hypothesis tests (Section 2.3.3) and

4. We analyze factors influencing the flexibility behavior using reliable statistical tests (Section 2.3.4).

To the best of our knowledge, we are the first to propose a generative model of individual customer flexibility behavior towards a particular appliance. Note that this paper is a substantial extension of our earlier work (which proposed a two-stage model) [1], since we now present (a) a new and more general, single-stage model for the flexibility behavior, and (b) more extensive analysis. The latter includes assessing both the accuracy of the proposed algorithms in regenerating the

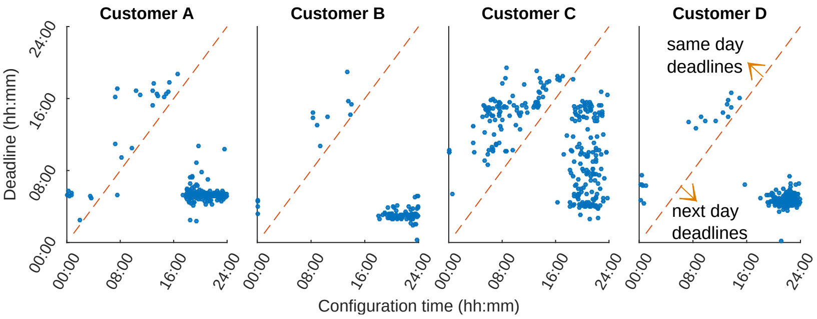


Figure 2.1: Flexibility profiles of four selected customers for dishwasher usage (times on x- and y-axis are time of day). Note that the observations below the diagonal line denote deadlines on the day after the configuration time.

data (based on statistical tests), and the factors influencing the flexibility behavior of customers. We also apply our models to customer data from washing machine and tumble dryers in addition to dishwashers.

2.2 Methodology

In this section, we first explain the input features used to model customer flexibility behavior. We then introduce two systematic methods to model the flexibility behavior of an individual residential household towards a particular smart appliance. Note that our first model is proposed specifically for modeling customer behavior that exhibits clear deadlines. We observed this behavior in dishwasher usage, an appliance that has been attributed a high demand response potential [21, 27, 28]. Our second model is a more general approach, which we found to be suitable for modeling all types of studied white goods (i.e., washing machine, tumble dryer and dishwasher).

2.2.1 Quantitive Specification of Flexibility

Each time a customer configures a smart appliance, (s)he sets the flexibility duration, i.e., how long the start of the device's operation can be deferred. We therefore represent the flexibility using two parameters: (1) *Configuration time*, which is the time at which the smart appliance is configured flexibly, and (2) *Deadline*, which is the latest allowed start time of the appliance and is calculated by adding the recorded flexibility duration to the time of configuration. Given that flexibility duration is a non-zero value without any upper bound, the deadline parameter may exceed the 24 h limit. The flexibility profile is defined for a given customer and a particular appliance, and is a model for his/her usage of that appliance. Note that we do not parameterize flexibility by the amount of shiftable power, since this attribute has already been analyzed extensively in the literature (e.g., [28]). The flexibility data for each customer and each appliance is obtained from year-long measurements in the LINEAR pilot project [26] (see further, Section 2.3). Figure 2.1 depicts flexibility profiles of four selected customers for their dishwasher. Each point on the graph of Figure 2.1 represents a single usage, with the configuration time on the x-axis and the corresponding deadline on the y-axis.

2.2.2 Model I: Two-Stage Single Variate Approach

Customer flexibility profiles for dishwashers in Figure 2.1 suggest that (unlike washing machines and tumble dryers, as we'll see later) dishwashers are usually configured with a typical deadline. For example, Customer A has 3 typical deadlines around 5 am, 10 am and 4 pm. Configuration times are different for different deadlines. For example, most of the configurations with deadline at around 5 am are set in the late evening and span until a few hours after midnight, whereas the configurations with 4 pm deadlines usually occur during daytime from 8 am onwards. Given this typical customer behavior for dishwashers, our first model proposes a two-stage univariate approach. In the first stage, we identify the typical deadlines of the customer by using a clustering algorithm and estimate the probability $P(\text{deadline})$ proportional to the size of the cluster for the respective deadline. For each cluster of deadlines resulting from Stage I, we then use parametric probability distributions to model the corresponding configuration times in Stage II and obtain $P(\text{configuration time} \mid \text{deadline})$. The joint distribution of deadlines and configuration times can then be obtained by $P(\text{configuration time, deadline}) = P(\text{deadline}) \cdot P(\text{configuration time} \mid \text{deadline})$.

2.2.2.1 Stage I: Identification of Typical Deadlines

The main objective of this stage is to use the deadline feature (y-axis in Figure 2.1) as the input to a clustering algorithm to identify the typical deadline clusters and parameterize their distributions. For this purpose, we adopt the G-means clustering algorithm [29] and change its hypothesis test (i.e., the Anderson-Darling test of normality). For completeness, the G-means algorithm and our modifications are explained below.

A wide variety of algorithms have been proposed for clustering of load profiles (e.g., see [30]). Some of the popular ones are K-means, expectation maximization, fuzzy K-means, hierarchical clustering, and self-organizing maps. An overview of these algorithms and their performance comparison is presented in [31]. G-means is a clustering algorithm based on K-means, with the capability of dynamically determining the number of clusters using hypothesis tests, hence eliminating the challenging task of choosing the optimal number of components in K-means. G-means is an iterative approach that starts with an initial value of k, i.e., the number of clusters (starting with $k = 1$ in absence of prior information). It then decides to split each cluster into two new clusters if the data in the current cluster does not pass the Anderson-Darling test of normality. After each iteration, the K-means algorithm is executed on the entire data to refine the solution. The core assumption in G-means is that the data in each cluster is normally distributed. Since deadlines are strictly positive, we instead assumed a Gamma distribution for the data in each cluster. Hence, we replaced the Anderson-Darling test of normality with a one-tailed Kolmogorov-Smirnov (k-s) test at significance level of 1% . We refer to this adaptation as Γ-means.

2.2.2.2 Stage II: Parameterizing the Distribution of Configuration Times

In this stage, we employ probability distributions to model the corresponding configuration times for each deadline cluster resulting from Stage I. Qualitative exploration of tuning data revealed that the empirical distributions of configuration times are often multi-modal, skewed, and with heavy tails. These characteristics suggested the use of finite mixture models (FMM) as parametric models to represent the unknown distributions as a mixture of known distributions. In what follows, we present the general definition of FMMs.

Let X be an r-dimensional random variable with probability density function $f(\mathbf{x})$, defined in the sample space $\mathcal{X} \subset \mathbb{R}^r$ and arising from a K-component finite mixture distribution. The probability density of X for all $\mathbf{x} \in \mathcal{X}$ is then defined as

$$f(\mathbf{x}) = \sum_{k=1}^{K} \eta_k f_k(\mathbf{x}), \tag{2.1}$$

where $f_k(\mathbf{x})$ is the component-wise probability density function and η_k is the corresponding weight of the component. $\boldsymbol{\eta} = (\eta_1, ..., \eta_K)$ is called the weight distribution. $\boldsymbol{\eta}$ takes a value in the unit simplex ε_K which is a subspace of $(\mathbb{R}^+)^K$ defined by the following constraints:

$$\eta_k \geq 0, \quad \eta_1 + \eta_2 + ... + \eta_k = 1. \tag{2.2}$$

Assuming that all the component densities arise from the same distribution family, $\mathcal{T}(\boldsymbol{\theta})$, the mixture density function is then written as,

$$f(\mathbf{x}|\boldsymbol{\vartheta}) = \sum_{k=1}^{K} \eta_k f_k(\mathbf{x}|\boldsymbol{\theta}_k), \tag{2.3}$$

where $f_k(\mathbf{x}|\boldsymbol{\theta}_k)$ is the probability density function of the k^{th} component indexed by the parameter $\boldsymbol{\theta}_k$ and $\boldsymbol{\vartheta} = (\boldsymbol{\theta}_1, ..., \boldsymbol{\theta}_K, \boldsymbol{\eta})$ is the parameter vector of the mixture model.

algorithm 1: Schnatter's MCMC algorithm

```
Input  : data points ($\mathbf{X}$, with $|\mathbf{X}| = N$); number of components ($K$)
Output: Parameter vector of the mixture model ($\boldsymbol{\vartheta}$); allocation vector ($\mathbf{S}$) denoting
        to which component each of the observations are assigned (if $x_i$ is part
        of cluster $k$, then $S_i = k$).
Initialize S (based on the K-means algorithm) repeat $M + M_0$ times
    /* Given the allocation vector $\boldsymbol{S}$, estimate the mixture
       parameters                                                  */
    Sample $\boldsymbol{\eta} = (\eta_1, \ldots, \eta_K)$ from the Dirichlet distribution
      $D(e_1(\boldsymbol{S}), \ldots, e_K(\boldsymbol{S}))$, where $e_k(\boldsymbol{S}) = e_0 + N_k(\boldsymbol{S}), k = 1, ..., K$,
      and $N_k(\boldsymbol{S})$ is the number of data points allocated to component $k$ of the
      mixture and $e_0$ is the prior of the Dirichlet distribution
    foreach $k = 1, \ldots, K$ do
        Sample the component parameter $\boldsymbol{\theta}_k$ from the complete-data posterior
          $p(\boldsymbol{\theta}_k | \boldsymbol{S}, x)$
    /* Update the allocation vector $\boldsymbol{S}$ based on the sampled
       parameter vector $\boldsymbol{\vartheta}$                                    */
    Classification of each observation $x_i$ conditional on knowing $\boldsymbol{\vartheta}$, by sampling
      $S_i$ independently for each $i = 1, \ldots, N$ from the following discrete distribu-
      tion: $p(S_i = k | \boldsymbol{\vartheta}, x_i) \propto p(x_i | \theta_k)\, \eta_k$
Disregard the first $M_0$ draws return $M$ draws for $(\boldsymbol{\vartheta}, \boldsymbol{S})$
```

To estimate the parameter vector $\boldsymbol{\vartheta}$, we employ a Bayesian approach based on data augmentation and Markov-chain Monte Carlo (MCMC) proposed by Schnatter [32]. The main difference between MCMC and the classical expectation maximization (EM) algorithm based on maximum likelihood (ML) estimation is that MCMC performs integration while EM does maximization. One of the key advantages of a Bayesian approach over ML is the direct availability of confidence regions while for EM these may be inaccurate for small data sizes because of the reliance on asymptotic approximations. Also, the Bayesian approach allows including prior information in the estimation procedure.

The data augmentation and the MCMC algorithm by Schnatter [32] are briefly explained here for completeness. The main objective of the algorithm is estimating the component parameters, $\boldsymbol{\vartheta}$, as well as the allocation vector $\boldsymbol{S} = (S_1, S_2, \ldots, S_N)$ denoting the allocation of each observation to its corresponding component in the mixture ($S_i = k$ if x_i belongs to cluster k). This problem is interpreted as a missing data problem. The augmented parameter $(\boldsymbol{\vartheta}, \boldsymbol{S})$ is estimated by sampling from the complete-data posterior distribution based on the Gibbs MCMC algorithm. Schnatter's MCMC algorithm is outlined in Algorithm 1.

First, the algorithm uses the K-means clustering method to assign each observation to one of the K mixture components and initializes the allocation vector $\mathbf{S}$

accordingly (Line 1). The algorithm then runs for $M + M_0$ iterations (Lines 1–5). In each iteration, the component weights are sampled from a Dirichlet distribution (Line 2) and the parameters of each component of the mixture are sampled from the complete-data posterior, given allocation vector and the observations (Lines 3–4). Once the samples of the weights and the component parameters are drawn, the probability of belonging to a particular component is calculated for each observation and each observation is then allocated to a component based on the calculated probabilities (Line 5). At the end of the algorithm, the first M_0 draws are disregarded (Line 6) and the remaining M draws for $(\boldsymbol{\vartheta}, \boldsymbol{S})$ are returned (Line 6).

The mixture parameters are identifiable at best up to an arbitrary permutation mainly because the component number of the mixture is not a meaningful concept. This phenomenon results in the label switching problem due to the invariance of the mixture likelihood function under relabeling the components of a mixture model [32]. To solve this problem, [32] has proposed to apply random permutations of labels on the draws of their MCMC algorithm to force balanced label switching and ensure that the sampler explores the full mixture posterior distribution.

To choose the optimum number of components, we employ marginal likelihood denoted as $P(\boldsymbol{X}|\mu_k)$ where μ_k is the k-component FMM model. The marginal likelihood measures the average fit of a model to the data, whereas likelihood based or point estimators such as Akaike information criteria (AIC) [33] or Bayesian information criteria (BIC) [34] base their decisions on the best fit of each competing model [35]. The use of the marginal likelihood also automatically penalizes the complexity of the model because complex models spread their probability mass widely by predicting various possible outcomes, hence the probability of actual data will be smaller for overly complex models.

2.2.3 Model II: Single Stage Bivariate Approach

The proposed two-stage univariate model is relatively simple since one-dimensional data is used for modeling in each stage. Yet, it is therefore not possible to model arbitrary interactions between configuration times and deadlines: Model I might not be very efficient when the customer does not make clear deadline choices (i.e., the flexibility profile does *not* amount to groups of data points with a similar y-axis value, which represents the deadline, such as Customer C in Figure 2.1 or customer flexibility profiles for washing machine and tumble dryer in Figure 2.7 and Figure 2.8 respectively). Indeed, for such cases, the clustering in Stage I of Model I will produce multiple clusters of deadlines, each one in parallel with the x-axis, but this might not be the best representation of the underlying data. Therefore, we propose a second methodology to model the customer flexibility profile using a bivariate Gaussian mixture model (GMM).

In this scheme, a bivariate GMM is fit to the flexibility profile of the customer: both of the flexibility features (i.e., deadline and configuration time) form the input to the model, and their joint probability distribution is estimated in a single step. The same methodology based on Bayesian MCMC (explained in Section 2.2.2.2) is employed to fit the bivariate mixture model and select the optimal number of components.

2.3 Results and Discussion

We have applied our methodologies on the data from year-long measurements in the LINEAR pilot project [26] for 3 types of white goods: dishwasher, washing machine and tumble dryer. The flexibility data associated with each appliance is obtained from smart meters. Measurements are taken every 15 min: a day is divided into 96 time slots and any customer configuration within the 15 min interval is reported at the end of the interval, implying that the resulting customer flexibility profile is discrete. To make our data continuous, we spread the measured data over the 15 min preceding the interval end times, by adding random noise from a uniform distribution.

For each appliance, we considered customers with at least 100 configurations to increase the reliability of the analysis, leading to a set of 15 test customers for dishwashers, 12 for washing machines and 8 for tumble dryers. It is noteworthy that the proposed methodologies are implemented in MATLAB.

The results are presented in four subsections. In Sections 2.3.1 and 2.3.2, we model the distribution of the flexibility profiles based on the test data using the proposed methodologies and present the resulting clusters. In Section 2.3.3, we evaluate the efficiency of the derived models in reproducing the customer flexibility profile when using them to generate synthetic data. We then apply χ^2 and Fisher's exact tests in Section 2.3.4 to examine the dependency of cluster membership on day-of-the-week, weekends, holidays and seasons.

2.3.1 Model I Parameter Fitting and Analysis of Resulting Clusters

Model I is suitable for flexibility profiles that have typical deadlines in their configurations (i.e., the dishwasher profiles in our dataset). It is a two-stage univariate approach in which each stage models one of the two flexibility features (i.e., the configuration time or the deadline). Hence, Model I allows to analyze each of these flexibility features separately, as presented in the following subsections.

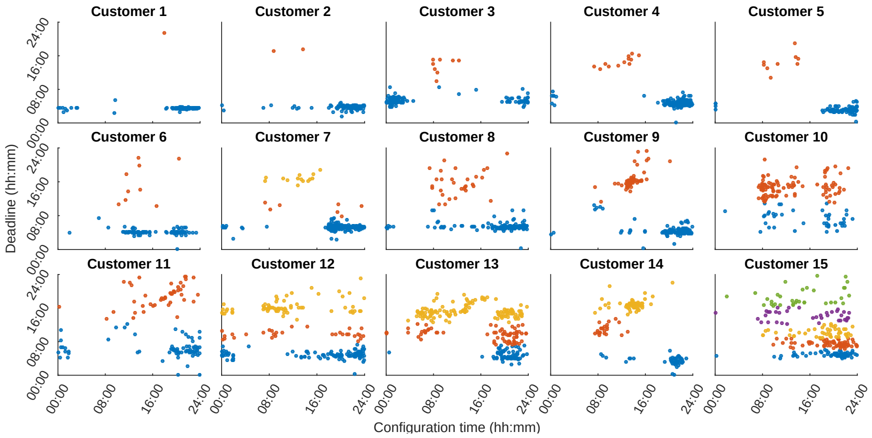


Figure 2.2: Clusters of deadlines from Stage I in the customer flexibility profiles for dishwasher. Each cluster is shown in a different color.

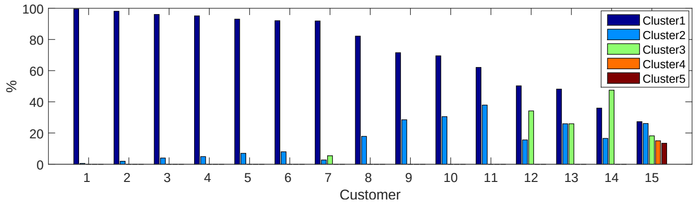


Figure 2.3: Percentage of data in each deadline cluster for dishwasher. (Note that a given cluster index for one customer may relate to a different deadline than the same index for another customer.)

2.3.1.1 Stage I: Analysis of Deadline Clusters

Figure 2.2 shows the output at Stage I of the model, where Γ-means clustering is applied to deadlines (y-axis), for the test customers. The corresponding configuration times (x-axis) are also shown in Figure 2.2 for further analysis and comparison of differences among the configuration times of various clusters. An exploratory analysis of the outcome of Stage I on test customers reveals the following characteristics of the customers' flexibility profile for their dishwashers: (i) there are typically 2 to 3 deadline clusters, early mornings (around 4-5 am), late morning/noon (10-12 pm), and evening (around 5-6 pm), but the number of deadline clusters varies among customers; (ii) for most customers, morning deadlines are more frequent and the corresponding configuration times usually lie in the afternoon and late night. Such customers show a more deterministic behavior compared to others: for example, Customer 1 has a substantial amount of data in the cluster corresponding to the early morning deadline.

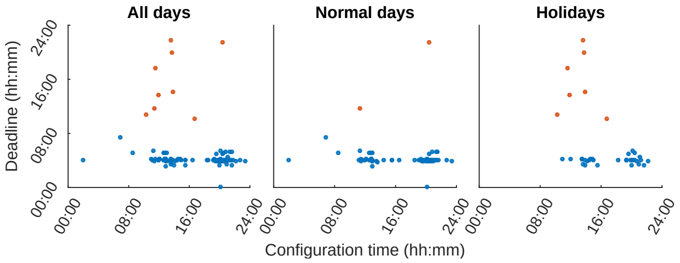


Figure 2.4: Example of a customer flexible configurations for dishwasher (Customer 6) during holidays vs. normal days over the cluster of deadlines.

Figure 2.3 shows the percentage of data in each deadline cluster of the test customers. Note that data are not evenly distributed across the clusters: for 7 out of 15 customers, one cluster contains more than 90% of the data (for most customers, one cluster has more than 50% of the data). This cluster represents the dominant habit of the customer while other clusters reflect his exceptional behavior. Figure 2.4 depicts an exemplary customer whose flexible configurations are affected by holidays. As seen from Figure 2.4, the data in the smaller cluster, representing afternoon and evening deadlines, are usually configured during holidays, although a substantial amount of configurations with early morning deadlines are still present during holidays.

However, these conclusions drawn from Figure 2.4 should not be extrapolated to all the customers, as we will show in Section 2.3.4.

2.3.1.2 Stage II: Analysis of Distribution of Configuration Times

In this stage, the parametric model is developed to represent the distribution of the corresponding configuration times of each of the resulting clusters in the previous stage. Looking back at Figure 2.2, we observe that for some clusters (e.g., those with early morning deadlines) the data in the lower left corner of the figure is related to that in the right. It is sensible to say that configurations shortly after midnight are indeed the tail of the ones at late evening and hence they are likely to belong to the same distribution. To account for this, we changed the reference point from midnight to the middle of the largest gap in the configuration time of each cluster (typically around 5-6 am). To model the configuration time distribution, we first focused on the larger clusters (with more than 100 data points) to ensure reliability of our conclusions and then applied the model to smaller clusters. In the following paragraph, we explain the three steps involved in choosing and fitting a parametric distribution to test clusters.

Step 1 (choosing the right parametric distribution): We initially fit all valid parametric distributions to the configuration times of test clusters and compared

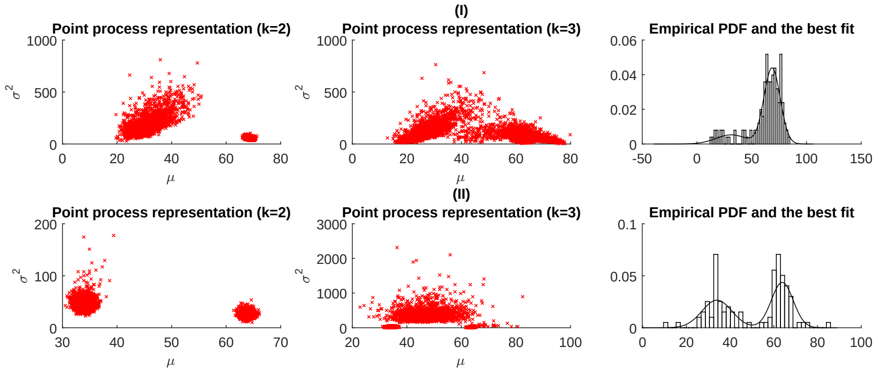


Figure 2.5: Point process representation of posterior draws and PDF of the best fit for two randomly selected customers for dishwasher.

the fit using goodness-of-fit tests (e.g., BIC, AIC, log-likelihood) as well as hypothesis tests (one tailed k-s test). From this analysis, we concluded that there is not a unique single-component parametric distribution that can represent the data for all of the test clusters. Also, for some clusters, none of the parametric distributions provides an acceptable fit. However, for those clusters that could be represented by a single-component parametric distribution, the non-central student distribution was usually the best fit. Further, the characteristics of the empirical distribution for the majority of the test clusters suggested the use of FMMs for parametric modeling. Based on the initial observation that suggested the non-central student distribution as a suitable fit, and the fact that a non-central student distribution is approximated by a Gaussian distribution for a large enough sample, we fit and compare the FMM from two families of distributions: Gaussian mixtures and student mixtures. To compare the fit of the two models, we used marginal likelihood values.

Comparing marginal likelihood values reveals that the Gaussian mixtures are a more suitable model to represent the data for all the test clusters. Hence Gaussian mixtures are employed for modeling and further analysis of the distribution of configuration times of deadline clusters.

Step 2 (choosing the number of components): To select the optimal number of components of the mixture, we employ the log-marginal likelihood together with a point process representation of the posterior draws to avoid overfitting. The point process representation is a viewpoint introduced by [36] which represents every component of the mixture in terms of its parameters. Next, we explain the overall procedure as well as the point process representations with the aid of an example.

Table 2.1 shows the log-marginal likelihood for fitting Gaussian mixtures with different values for k, i.e., the number of components of FMMs for test clusters. According to Table 2.1, the largest log-marginal likelihood corresponds to $k = 2$

Table 2.1: Log-marginal Likelihood, $p(\mu_k|\boldsymbol{X})$ (larger values are better).

Cluster	$\boldsymbol{k=1}$	$\boldsymbol{k=2}$	$\boldsymbol{k=3}$	$\boldsymbol{k=4}$
I	-553.95	-545.34	-595.17	-548.91
II	-595.53	-554.89	-554.84	-559.87

for Cluster I and $k = 3$ for Cluster II. The corresponding point process representation of posterior draws for $k = 2$ and 3, as well as the probability density function (PDF) of the best fit is shown in Figure 2.5. According to Figure 2.5(I), the point process representations of posterior draws for Cluster I form well-separated clusters around the parameters of the mixture components for $k = 2$ but increasing the number of components indicates overfitting because well-separated clusters corresponding to parameters of each component are no longer seen in the point process representation plot. Hence, $k = 2$ is the optimum choice as suggested by log-marginal likelihood values. However, for Cluster II, the $k = 3$ suggested by Table 2.1 overfits the data according to the point process representations shown in Figure 2.5(II). Therefore, $k = 2$ is selected as the optimum number of mixture components for Cluster II. It is noteworthy that in Figure 2.5, the x-axis is changed from time to timeslot of a day (i.e., a value between 0 and 95) to ease the comparison between point process representation and PDF of the best fits. Experimenting with test clusters suggests that when the difference in log-marginal likelihood of k and $k+1$ components is smaller than 1, the smaller number of components should be chosen to avoid overfitting.

Step 3 (determining the mixture component membership): Any data that is represented best with FMMs, can be clustered based on the mixture component membership and hence be analyzed to identify the potential factors influencing the component membership. We use the maximum-a-posteriori (MAP) clustering algorithm to identify, with a certain probability, the component membership of each data point. MAP identifies the allocation vector that yields the largest posterior probability. The optimum allocation vector is easily obtained by keeping track of the allocation vectors and corresponding posterior probability at each iteration of Schnatter's algorithm (explained in Section 2.2.2.2).

Application of the outlined steps to deadline clusters of the test customers reveals that: (i) the majority of the clusters are best represented with a 2-component Gaussian mixture model (and the optimum number of components never exceeds 3), and (ii) the component weights are not uniformly distributed in the majority of the clusters, meaning that there is typically one dominant cluster.

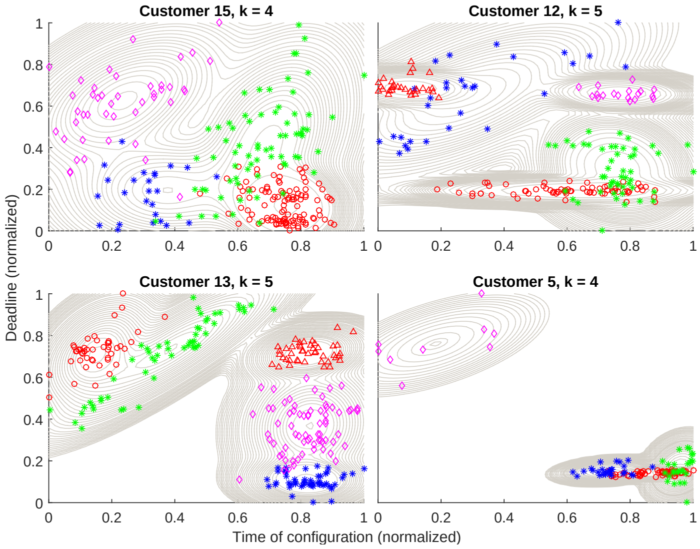


Figure 2.6: Bivariate Gaussian mixture fits and the resulting MAP clusters for *dishwasher* (in distinct symbols and colors; contour plots indicate the fitted distributions).

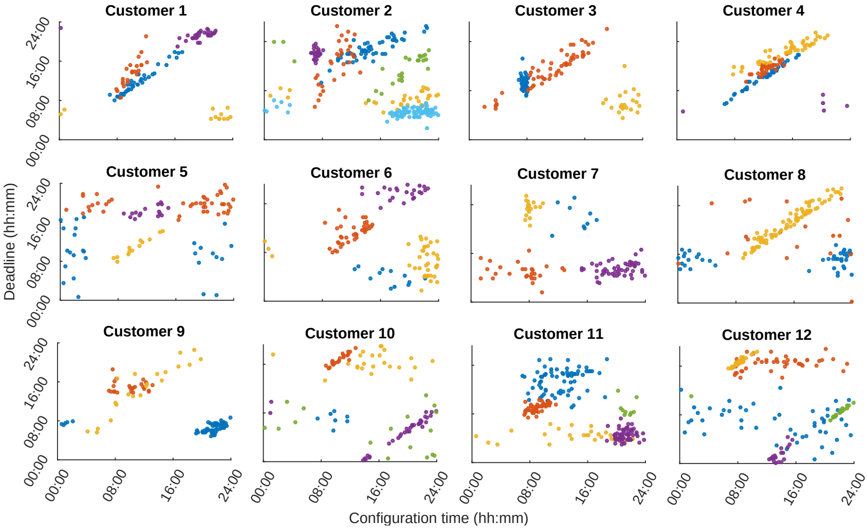


Figure 2.7: Bivariate Gaussian mixture fits and the resulting MAP clusters for *washing machine* flexible usage (in distinct symbols and colors; contour plots omitted for clarity).

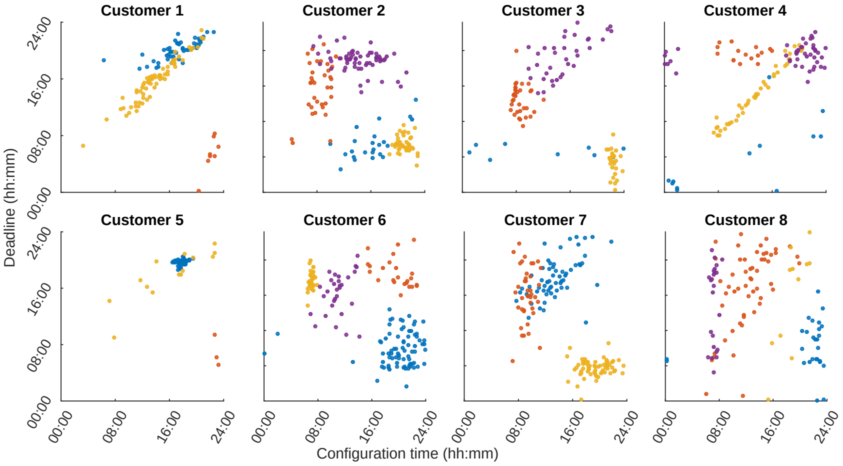


Figure 2.8: Bivariate Gaussian mixture fits and the resulting MAP clusters for *tumble dryer* flexible usage (in distinct colors; contour plots omitted for clarity).

2.3.2 Model II Parameter Fitting and Analysis of Resulting Clusters

In this section, we fit bivariate Gaussian mixture models to our test data. Similar to Stage 2 of Model I, we change the reference point to the middle of the largest gap seen in the configuration times (x-axis) of the customer's data, in order to ensure an acceptable continuity of the distribution of configuration times over midnight. The summary of the estimated model parameters are tabulated in the Appendix.

The results of fitting Model II to flexibility profiles for dishwashers are depicted in Figure 2.6 for 4 example customers. The contour plots in Figure 2.6 indicate the bivariate Gaussian mixture fits. We have also employed MAP clustering to assign each point to a single component (as explained in Subsection 2.3.1.2, Step 3). The resulting MAP clusters are shown using distinct colors and symbols in Figure 2.6. Unlike clusters in Model I, the clusters in Model II are not always aligned in parallel with the x-axis (i.e., do not have similar deadlines). Two categories of clusters are observed in Model II: (i) clusters with similar deadlines (i.e., horizontal ones), and (ii) those in which the customers configure the device with similar flexibility duration (i.e., clusters in parallel with the $x = y$ diagonal line). This shows the versatility of Model II in accommodating various types of customer behaviors (i.e., configurations with similar deadlines or similar flexibility durations).

We also applied Model II on the customer flexibility profiles for washing machines and tumble dryers (refer to the tables in Appendix for a summary of the model parameters). The resulting clusters are shown in Figure 2.7 and Figure 2.8

for washing machine and tumble dryer respectively. The aforementioned categories of clusters are also observed in the flexibility profiles for washing machines and tumble dryers. As seen from Figure 2.7 and Figure 2.8, for the majority of customers using a washing machine or tumble dryer, their configurations with similar deadlines (early morning) usually are in the late evening and span to few hours after the midnight. The configurations throughout the day all have a similar flexibility duration (resulting in clusters roughly parallel to the $x = y$ diagonal). In addition, there are clusters with similar configuration times but different deadlines (i.e., vertical ones) for some customer flexibility profiles for washing machine and tumble dryers (e.g., Customers 2, 3, 6, 7 and 8 in Figure 2.8 and Customers 2, 3 and 7 in Figure 2.7). Such vertical clusters are not observed in customer flexibility profiles for dishwashers.

2.3.3 Analysis of Model Efficiency for Data Regeneration

In this section we analyze the efficiency of the models in regenerating the customers flexibility profile. To achieve this objective, we first model a customer using our proposed methods. Each method yields a probabilistic model of the customer's data. We note that in Stage I of Model I, we model the data in each deadline cluster using a gamma distribution. Hence, we represent the overall data as mixture of gamma distributions where the parameters of each component of the mixture are equal to the underlying gamma distribution in each deadline cluster. The weight of each component in the mixture is calculated by taking the ratio of the corresponding deadline cluster size to the overall data size. We then generate 500 sample sets of the same size as the original data (from which the models are derived). Each sample set is compared to the original data using a two tailed k-s test to determine whether the sample and the original data come from the same distribution. The null hypothesis is that data and sample come from the same distribution and the alternative hypothesis is that they do not. It should also be noted that since the original k-s test is developed for one-dimensional data, we have applied the extension of the k-s test to bivariate data, as proposed by [37].

Table 2.2 shows the percentages of times the test fails to reject the null hypothesis at 5% significance level. As seen from Table 2.2, for most of the customer flexibility profiles (of all 3 categories of white goods) more than 90% of the samples generated by Model II are not rejected by the hypothesis test and hence, may be representative samples.

For dishwashers, Model II outperforms or matches Model I in re-generating the data, although for some customers they perform equally well. Yet, for some customers, Model I never generates a sample that can be considered to be representative of the original data. Investigating these cases, analysis reveals that although data in each cluster follows a Gamma distribution (due to Γ-means clustering), for

Table 2.2: Analyses of model efficiency in regenerating the customer flexibility profiles: percentage of times the null hypothesis is not rejected. (Note that the same customer numbers across devices do not necessarily correspond to the same users.)

Dishwasher			Washing Machine		Tumble Dryer	
Customer	Model I	Model II	Customer	Model II	Customer	Model II
1	0.0 %	90.4 %	1	100 %	1	100 %
2	83.2 %	95.2 %	2	100 %	2	100 %
3	0.2 %	99.2 %	3	100 %	3	100 %
4	7.8 %	99.2 %	4	98 %	4	91 %
5	31.4 %	89.8 %	5	100 %	5	100 %
6	0.0 %	97.2 %	6	100 %	6	100 %
7	0.0 %	61.2 %	7	100 %	7	100 %
8	0.6 %	99.8 %	8	100 %	8	100 %
9	48.0 %	95.4 %	9	100 %		
10	99.6 %	99.6 %	10	100 %		
11	77.2 %	100.0 %	11	100 %		
12	99.2 %	99.8 %	12	100 %		
13	99.6 %	99.8 %				
14	98.2 %	100.0 %				
15	98.2 %	99.0 %				

customers with very deterministic behavior (i.e., very narrow deadline windows in each deadline cluster), representing hard clusters of deadlines with a mixture of Gamma distributions does not regenerate a sample representative of deadline distributions. We conclude this from a one dimensional two tailed k-s test by considering deadline feature of customer flexibility data: the null hypothesis (i.e., data arising from the mixture of Gamma distribution) is always rejected for customers with very narrow spread in their deadlines.

Despite the fact that Model I often does not represent the data as successfully as Model II, it still has the benefit of simplicity, both in constructing the model and in interpreting it (e.g., clear identification of typical deadlines, which is characteristic of several dishwasher flexibility profiles).

2.3.4 Dependency Analysis Using Statistical Tests

To investigate the dependency of resulting clusters of each model on potential determining factors, including weekends, day-of-a-week, holidays and seasons, we employ χ^2 and Fisher's exact tests. Fisher's exact test is only employed when the conditions of the χ^2 test are not met, mainly due to small data sizes in some cells of the contingency table. Additionally, Fisher's exact test is only applicable for 2×2 contingency tables: for larger tables, extensions of Fisher's test are employed (e.g., Freeman-Halton's extension [38] for 2×3). We define our null and alternative hypothesis for a 'factor' as follows:

- *Null hypothesis*: cluster membership and 'factor' are independent,
- *Alternative hypothesis*: cluster membership and 'factor' are dependent.

The dependency tests are conducted at the 5% significance level. In other words, for p-values larger than 0.05, the test fails to reject the null hypothesis and hence does not conclude a dependency among the considered 'factor' and clusters under analysis. The number of columns in the contingency table corresponds to the number of deadline clusters for each customer. The contingency tables always consists of two rows. For example, to test the dependency of cluster membership on weekends, the first row of the corresponding table would contain count data for configurations on weekend days and the second row would contain the configurations on all the other days of the week. Similarly, for a day-of-the-week factor, the contingency table of each day would have count data for the particular day in one row and the configurations of the remaining six days of the week on another row.

The outcome of the tests (for dishwashers only) for clusters in Stage I and Stage II of Model I are summarized in Table 2.3 and Table 2.4 respectively.

Table 2.3(a) shows that weekends, holidays and seasons may influence the behavior of some customers in setting deadlines for their dishwasher. However, as mentioned earlier, the aforementioned factors do not affect all customers behaviors similarly. Additionally, autumn seems to have a more pronounced effect on customer deadlines than the other seasons. From Table 2.3(b), we note that 6 out of the 15 customers (shaded rows) have a dependency of dishwasher configuration times on at least one day of the week. While some customers remain unaffected by day-of-the-week, some others are greatly influenced by this factor (e.g., Customer 15, for whom the null hypothesis is rejected for 4 days of the week).

Table 2.4, shows that holidays and seasons do not significantly affect the component memberships in Stage II for the test deadline clusters from Stage I. However, weekends influence component membership for 5 out of the 10 test clusters. Additionally, 7 out of 10 test clusters have at least one day that influences the component membership of their data points.

We have also applied the dependency tests to the clusters of Model II to identify factors influencing the cluster membership of a customer's data points. The p-values of the dependency tests on dishwashers are summarized in Table 2.5. We constructed similar tables for customer flexibility profiles for washing machines and tumble dryers, however, we omit them from the paper to save space: below we summarize the main observations.

For *customer flexibility profiles for dishwashers*: from Table 2.5, 5 out of 15 customers' cluster membership might be dependent on holidays, whereas more than 50% of the customers behavior is affected by weekends, seasons and day-of-the-week factors. Among seasons, winter least influences the cluster memberships.

Table 2.3: P-values of dependency tests in Stage I of Model I for dishwasher usage. (Bold values indicate a p-value less than 0.05 and shaded rows indicate customers affected by at least one 'factor'.)

(a) Holidays, weekends and seasons

Cus-tomer	**No. of clusters**	**Test type**	**holiday**	**weekend**	**spring**	**summer**	**autumn**	**winter**
1	2	Fisher	2.9E-01	1.0E+00	6.2E-01	8.8E-02	5.7E-01	5.1E-01
2	2	Fisher	5.4E-01	5.3E-01	4.6E-01	3.4E-01	4.0E-01	3.8E-01
3	2	Fisher	2.1E-01	1.0E+00	5.6E-01	1.7E-01	5.8E-01	2.6E-01
4	2	Fisher	5.4E-01	1.0E+00	6.5E-01	5.5E-01	1.0E-01	**2.8E-02**
5	2	Fisher	1.0E+00	1.2E-01	**6.2E-03**	5.3E-01	**3.8E-02**	3.9E-01
6	2	Fisher	**1.1E-02**	4.4E-01	1.7E-01	7.6E-02	**3.7E-02**	5.9E-01
7	3	Fisher	5.2E-01	**5.2E-04**	7.8E-01	8.4E-02	**6.4E-03**	2.0E-01
8	2	χ^2	8.8E-02	**1.2E-04**	**4.3E-02**	1.6E-01	9.0E-01	5.2E-01
9	2	χ^2	3.9E-01	2.8E-01	6.32E-02	6.9E-01	2.5E-01	1.2E-01
10	2	χ^2	**2.8E-03**	3.2E-01	8.5E-02	**5.0E-02**	**4.7E-03**	5.2E-01
11	2	χ^2	9.5E-01	3.0E-01	1.03E-01	9.6E-01	1.4E-01	9.9E-01
12	3	χ^2	3.5E-01	**1.8E-02**	9.8E-01	1.8E-01	**3.1E-04**	8.6E-02
13	3	χ^2	**5.1E-03**	2.0E-01	**2.1E-04**	**2.2E-02**	**5.0E-03**	7.1E-02
14	3	χ^2	3.7E-01	7.7E-02	3.1E-01	2.4E-01	2.0E-01	2.2E-01
15	5	χ^2	7.7E-02	6.0E-02	9.8E-02	3.7E-01	**4.9E-02**	**2.6E-02**

(b) Day-of-the-week

Customer	Sun	Mon	Tue	Wed	Thur	Fri	Sat
1	1.0E+00	1.0E+00	1.6E-01	1.0E+00	1.0E+00	1.0E+00	1.0E+00
2	1.0E+00	1.0E+00	1.0E+00	2.3E-01	3.1E-01	1.0E+00	1.0E+00
3	3.6E-01	1.0E+00	6.3E-01	6.0E-01	6.1E-01	9.1E-02	2.3E-01
4	1.0E+00	7.0E-01	6.7E-01	3.7E-01	4.1E-01	3.9E-01	1.0E+00
5	6.9E-01	6.9E-01	3.5E-01	1.0E+00	3.6E-01	5.6E-01	**3.8E-02**
6	1.0E+00	5.9E-01	1.0E+00	3.5E-01	1.0E+00	9.8E-02	2.1E-01
7	7.9E-01	2.9E-01	1.6E-01	5.0E-01	6.5E-01	1.7E-01	**7.9E-05**
8	**8.8E-03**	5.3E-01	1.1E-01	**2.2E-02**	5.9E-01	7.2E-01	**2.2E-02**
9	1.7E-01	3.4E-01	8.9E-01	1.3E-01	7.9E-02	6.4E-01	9.5E-01
10	3.6E-01	1.3E-01	8.4E-01	4.7E-01	6.1E-01	1.5E-01	8.0E-02
11	8.6E-01	9.3E-01	9.9E-01	2.8E-01	6.9E-01	8.1E-01	2.5E-01
12	6.6E-01	4.1E-01	9.0E-01	6.8E-01	8.3E-01	6.5E-02	**4.8E-04**
13	9.5E-01	**2.7E-02**	6.2E-01	8.5E-01	**1.6E-02**	**6.2E-03**	**1.7E-02**
14	9.1E-02	3.0E-01	3.2E-01	1.1E-01	3.1E-01	4.8E-01	5.6E-01
15	3.6E-01	3.9E-01	2.0E-01	2.7E-01	**7.1E-04**	7.2E-02	1.1E-01

The results of Table 2.5 also indicate that while not all customers are affected in a similar manner, from an overall perspective, weekend, seasons and day-of-the-week factors influence more customers than the holiday factor.

For *customer flexibility profiles for washing machines*: only 3 out of 12 customers are influenced by holidays, whereas, weekends, seasons and day-of-the-week factors affect 50% of the customers. Among seasons, winter has the highest influence (6 out of 12) and autumn has the least (2 out of 12) influence on cluster membership.

For *customer flexibility profiles for tumble dryers*: none of the 8 test customers are affected by holiday, and only 1 customer is affected by seasons. On the other hand, weekend and day-of-the-week factors influence more than 50% of the customers.

The main conclusion drawn from the dependency tests above is that customers not only exhibit different behavior in offering the flexibility of their smart device,

Table 2.4: P-values of dependency tests in Stage II of Model I for dishwasher usage. (Bold values indicate a p-value less than 0.05 and shaded rows indicate customers affected by at least one 'factor'.)

(a) Holidays, weekends and seasons

Cluster	**No. of clusters**	**Test type**	**holiday**	**weekend**	**Spring**	**Summer**	**Autumn**	**Winter**
A	3	χ^2	6.9E-01	3.1E-01	3.0E-01	2.5E-01	2.1E-01	3.7E-01
B	3	Fisher	2.1E-01	**2.5E-03**	4.6E-01	4.6E-01	**2.7E-02**	1.7E-01
C	3	χ^2	4.9E-01	**5.2E-03**	1.0E+00	2.1E-01	8.4E-01	5.0E-01
D	3	χ^2	4.0E-01	**3.4E-02**	4.5E-01	4.3E-01	1.4E-01	5.9E-01
E	2	χ^2	6.8E-01	**7.6E-04**	2.2E-01	2.6E-01	6.0E-01	7.5E-02
F	2	χ^2	4.2E-01	6.5E-01	1.2E-01	5.0E-01	5.6E-02	6.3E-01
G	2	χ^2	3.7E-01	6.2E-02	9.5E-01	8.8E-01	3.4E-01	4.6E-01
H	3	Fisher	3.6E-01	4.7E-01	5.5E-01	6.4E-01	6.3E-01	4.4E-01
I	3	Fisher	1.7E-01	1.9E-01	**3.9E-03**	7.7E-02	**1.5E-02**	1.8E-01
J	3	χ^2	**3.4E-06**	**8.7E-04**	7.0E-01	**2.7E-02**	5.5E-01	1.5E-01

(b) Day-of-the-week

Cluster	**Sun**	**Mon**	**Tue**	**Wed**	**Thur**	**Fri**	**Sat**
A	9.9E-01	3.3E-01	4.4E-01	1.2E-01	3.5E-01	**2.9E-02**	6.9E-02
B	**6.7E-03**	8.1E-01	8.6E-02	4.2E-01	7.2E-01	8.3E-01	3.8E-01
C	6.8E-02	9.2E-01	**1.2E-04**	9.6E-01	6.9E-01	7.2E-01	5.5E-02
D	2.4E-01	**3.3E-02**	6.3E-01	2.4E-01	**2.9E-02**	1.0E+00	1.1E-01
E	**7.8E-05**	6.9E-01	**3.5E-02**	1.6E-01	2.7E-01	9.4E-01	9.4E-01
F	2.6E-01	**3.0E-02**	7.8E-01	4.0E-01	3.9E-01	**3.8E-02**	3.5E-01
G	3.1E-01	3.0E-01	2.5E-01	9.6E-01	1.6E-01	2.8E-01	1.9E-01
H	2.4E-01	6.9E-01	5.2E-01	7.8E-01	2.8E-01	6.9E-01	3.1E-01
I	1.0E-01	5.3E-01	3.1E-01	9.2E-01	1.2E-01	7.0E-01	6.5E-01
J	**1.7E-04**	8.8E-01	**6.1E-03**	**2.3E-02**	2.3E-01	**2.9E-03**	**3.9E-05**

but also that they are affected differently by the aforementioned factors. Hence, the assumption that flexibility will be offered by all customers in a similar manner, is not supported by our data. It hence seems crucial to individually model customer behavior in order to incorporate the uncertainties due to their distinct habits, and thus allow for a more realistic assessment of DR impact.

2.4 Conclusion

Flexibility is characterized by (1) amount of deferrable load, (2) time of availability and (3) deadline to exploit. Existing works have extensively studied (1). However, (2) and (3) are challenging to analyze because they depend on customer appliance usage habits and may substantially differ among households. Hence, state-of-the-art lacks the quantitative analysis on (2) and (3): they are inferred from either the load patterns or time-of-use surveys which do not depict a realistic customer behavior (i.e., inferences are not based on real scenarios where customers configure their smart appliances flexibly). This distorts the realistic assessment of flexibility and hurdles their efficient exploitation.

In this paper, we took the first step to tackle this issue and aimed to sharpen the analysis of flexibility by characterizing the individual customer flexibility profile

Table 2.5: P-values of dependency tests for Model II for dishwasher usage. (Bold values indicate a p-value less than 0.05 and shaded rows indicate customers affected by at least a 'factor'.)

(a) Holidays, weekends and seasons

Cus-tomer	**No. of clusters**	**Test type**	**holiday**	**week-end**	**spring**	**summer**	**autumn**	**winter**
1	3	Fisher	3.8E-01	7.4E-01	5.1E-01	2.0E-01	8.3E-01	3.5E-01
2	4	χ^2	6.4E-01	1.8E-01	**7.9E-03**	6.2E-01	2.9E-01	6.7E-01
3	5	χ^2	**3.8E-07**	**8.1E-03**	3.4E-01	1.6E-01	8.6E-02	4.1E-01
4	4	χ^2	**1.5E-03**	**1.7E-03**	7.0E-01	**4.0E-03**	1.1E-01	9.0E-02
5	4	χ^2	8.9E-01	1.2E-01	**4.9E-02**	8.7E-01	1.6E-01	4.1E-01
6	5	χ^2	**8.8E-03**	2.1E-01	**3.8E-02**	**1.3E-04**	7.2E-02	1.6E-01
7	4	χ^2	7.0E-01	**2.0E-02**	1.6E-01	**2.8E-02**	**1.7E-04**	3.4E-01
8	4	χ^2	1.6E-01	**1.2E-05**	3.7E-01	3.1E-01	3.7E-01	8.0E-01
9	5	χ^2	5.1E-01	3.8E-01	2.6E-01	1.2E-01	**1.7E-02**	8.5E-02
10	4	χ^2	8.2E-02	4.5E-01	**5.5E-03**	**4.8E-02**	**1.3E-02**	9.6E-02
11	4	χ^2	5.5E-01	1.6E-01	7.7E-02	1.1E-01	6.5E-02	3.4E-01
12	5	χ^2	**5.0E-02**	**3.3E-03**	8.0E-01	2.0E-01	**8.7E-04**	1.3E-01
13	5	χ^2	**9.1E-05**	**8.5E-04**	**4.0E-05**	**2.8E-03**	**2.5E-03**	5.6E-02
14	5	χ^2	6.7E-01	**5.7E-03**	3.5E-01	5.4E-01	1.2E-01	2.8E-01
15	4	χ^2	1.2E-01	8.1E-02	8.6E-01	5.0E-01	3.0E-01	**4.4E-02**

(b) Day-of-the-week

Customer	**Sun**	**Mon**	**Tue**	**Wed**	**Thur**	**Fri**	**Sat**
1	8.6E-01	8.5E-01	**3.1E-02**	8.8E-01	**1.6E-02**	3.6E-01	4.4E-01
2	1.0E-01	4.7E-01	6.1E-01	2.3E-01	3.6E-01	6.9E-01	2.6E-01
3	**3.9E-02**	8.2E-01	7.8E-01	6.0E-01	2.3E-01	**2.4E-02**	1.0E-01
4	8.6E-02	5.0E-02	**1.4E-02**	7.4E-02	6.5E-01	**4.7E-04**	**6.7E-06**
5	3.2E-01	1.9E-01	2.3E-01	3.7E-01	**3.9E-02**	8.9E-02	**2.8E-02**
6	3.9E-01	7.7E-01	3.1E-01	1.4E-01	8.0E-01	5.5E-01	2.5E-01
7	9.2E-01	3.4E-01	2.2E-01	3.4E-01	6.3E-01	9.7E-02	**6.6E-03**
8	6.1E-02	8.2E-01	2.7E-01	**2.5E-02**	7.0E-01	6.0E-02	**9.2E-04**
9	2.9E-01	**3.2E-02**	8.7E-01	6.7E-01	5.3E-01	5.7E-01	9.9E-01
10	3.5E-01	2.9E-01	9.8E-01	9.9E-01	9.5E-01	2.4E-01	1.5E-01
11	5.2E-01	5.6E-01	4.4E-01	3.6E-01	7.6E-01	8.9E-01	3.0E-01
12	4.7E-01	8.0E-01	5.8E-01	8.9E-01	1.1E-01	5.8E-01	**8.3E-05**
13	**7.6E-05**	3.5E-01	**5.1E-03**	6.9E-02	5.5E-02	**7.0E-04**	**2.0E-04**
14	1.7E-01	7.3E-01	3.1E-01	3.8E-01	5.4E-01	4.9E-01	**1.6E-02**
15	1.1E-01	**8.2E-03**	1.1E-01	5.3E-01	9.2E-01	3.6E-01	5.4E-01

in terms of time of configuration and deadline to exploit the offered flexibility. We proposed a systematic approach to derive a generative statistical model of an individual customer's flexibility behavior in offering smart devices for DR exploitation, based on real-world data. Validating two proposed models with statistical tests, we found that especially Model II (bivariate Gaussian mixture models) was efficient in re-generating the customer flexibility profile. Finally, we evaluated the dependency of customer behavior on various factors using statistical tests, to conclude that not only do different users exhibit potentially substantially different flexibility behavior, but also that such flexibility could be influenced differently by factors such as day-of-the-week or seasons, etc. More details on the outcome of our contributions are explained below.

Our Model I is a two-stage univariate approach that is suitable for flexibility profiles that exhibit typical deadlines (e.g., dishwasher usages as observed in our real-world dataset). The first stage employs a Γ-means clustering algorithm to identify the typical deadlines and the second stage uses a Bayesian model and employs the MCMC algorithm to obtain the distribution of the corresponding configuration times for deadline clusters from Stage I. The analysis of customer flexibility profiles for dishwasher based on Model I reveals the following: (i) the majority of the customers tend to configure their smart dishwasher to meet a limited set of typical deadlines and the data in each cluster follows a Gamma distribution (based on k-s hypothesis test), (ii) for most customers, the size of deadline clusters were unbalanced with one cluster representing the dominant habit of the customer, (iii) the distribution of configuration times in each cluster resulting from Stage I was best modeled by a mixture of Gaussian distributions (based on marginal likelihood criteria).

Model II is a more general approach that takes both flexibility features as the input and models their joint distribution in a single step. The proposed model fits bivariate Gaussian mixtures on the flexibility data of each customer using a Bayesian MCMC algorithm. MAP clustering is used to assign each observation to a single component. Three categories of MAP clusters are observed when modeling the customer flexibility profiles using Model II: (i) the ones with similar deadlines and different configuration times, (ii) the ones with similar configuration times but different deadlines and (iii) the ones with similar flexibility duration. Hence, this approach is more versatile in modeling various customer behaviors.

To validate the efficiency of Model I and Model II in regenerating the customer behavior for synthetic data generation purposes, we proposed a systematic approach based on a Kolmogorov-Smirnov (k-s) test. Based on our validations, Model II was identified to be an appropriate regenerative model for all 3 types of white goods in our analysis. The summary of Model II parameters fit to our real-world dataset are given in the Appendix.

Finally, to study factors influencing the customer flexibility behavior, we used χ^2 and Fisher's exact tests and examined the effect of 4 factors: holidays, weekends, seasons and days-of-the-week. Our dependency analysis of real-world data suggests that customers are not similarly affected by aforementioned factors: modeling flexibility merely based on device characteristics and assuming that all customers will utilize them in the same manner does not reflect observed behavior. This further indicates the need to model customers individually in order to incorporate the uncertainties influencing their flexibility behavior due to their appliance usage habits.

2.4.1 Future Work

The configuration time in the flexibility profile has a cyclic nature, hence, it is plausible that configurations shortly after midnight are the tails of distributions of the ones in the late evening. In order account for this, while still keeping the modeling in linear space, we adopted a transformation of the x-axis as shown in the figures throughout the paper, and changed the reference point to the middle of the largest gap in the configuration times of the customer for both Model I and Model II. However, it could be statistically meaningful (though more complex) to consider distributions of flexibility on a cylinder, with configuration times on a circle, while deadlines are on the vertical axis. Such mathematically more complex analysis is left for future work.

Finally, the comparison of test customers suggests similarities among some customers in terms of their flexibility behavior (e.g., similar deadline clusters). In future research, we will use clustering to group customers with similar flexibility profiles for each appliance. Clustering of similar customers may improve the exploitation of their flexibility by DR algorithms (e.g., by tailoring a specific DR algorithm for a group of similar customers).

Appendix

In this appendix, we summarize the estimated parameter values of Model II for the customer flexibility profiles for all three appliance types. We also give guidelines for generating synthetic data based on the summarized parameters. The notations used in the table headers are explained first:

- $[X_{min}, X_{max}]$: the minimum and maximum values of the configuration time,
- $[Y_{min}, Y_{max}]$: the minimum and maximum values of the deadline,
- Ref.: the largest gap in the configuration times of the customer (The data is shifted around this reference to ensure the continuity of the configuration times.)
- η_k: the weight of k^{th} component of the mixture
- μ_k: the normalized mean of the k^{th} component of the mixture
- Σ^k: the covariance matrix of the k^{th} component of the mixture. To compactly represent the covariance matrix in the tables, we note it as $(\Sigma_{11}, \Sigma_{12}; \Sigma_{21}, \Sigma_{22})$ which corresponds to $\begin{pmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{pmatrix}$.

The following steps explain how to re-generate a customer flexibility profile:

– *Step 1*: sample form the K-component Gaussian mixture model based on the given parameters and weights for each customer.

– *Step 2*: the data was normalized prior to the modeling (i.e., using $X_{normalized} = \frac{X - X_{min}}{X_{max} - X_{min}}$ for normalizing configuration times and $Y_{normalized} = \frac{Y - Y_{min}}{Y_{max} - Y_{min}}$ for normalizing deadlines). Hence, de-normalize the generated samples accordingly.

– *Step 3*: shift the configuration times around the “Ref.” offset.

The aforementioned steps will regenerate a customer flexibility profile with deadlines ranging from $[0\,\text{h}, 24\,\text{h}]$ and configuration times between $[0, 96]$ (which represent indices of the 15 min timeslots). Note that since we had 15 min measurements to obtain the flexibility information, the configuration times are between $[0, 96]$ in the re-generated sample and correspond to time-slot of the day. To obtain the configuration times in terms of hour of the day, multiply the values by $1/4$.

Table 2.6: Summary of estimated parameters of Model II based on customer flexibility profiles for tumble dryers

Customer (Data Size)	$[X_{min}, X_{max}]$	$[Y_{min}, Y_{max}]$	Ref.	η_k	μ_k	Σ^k=($\Sigma_{11}, \Sigma_{12}; \Sigma_{21}, \Sigma_{22}$)
1 (126)	[13.193, 92.846]	[0.134, 22.852]	0	2.97E-01	(5.87E-01, 1.78E-01)	(3.03E-03, -3.01E-04; -3.01E-04, 1.12E-03)
				5.64E-01	(6.83E-01, 1.77E-01)	(8.26E-03, -3.24E-05; -3.24E-05, 1.30E-04)
				1.38E-01	(4.16E-01, 5.38E-01)	(6.34E-02, -4.41E-02;-4.41E-02,9.87E-02)
2 (163)	[17.166, 92.075]	[3.307, 22]	0	2.01E-01	(2.19E-01, 6.21E-01)	(8.00E-03, 4.91E-03; 4.91E-03, 2.92E-02)
				1.64E-01	(6.08E-01, 1.82E-01)	(3.91E-02, 5.37E-03; 5.37E-03, 1.21E-02)
				3.72E-01	(5.27E-01, 8.13E-01)	(3.02E-02, -4.77E-03; -4.77E-03, 7.28E-03)
				2.63E-01	(8.82E-01, 1.91E-01)	(4.97E-03, 1.17E-04; 1.17E-04, 5.37E-03)
3 (104)	[4.169, 91.279]	[0.291, 23.997]	21	1.93E-01	(4.33E-01, 4.45E-01)	(5.38E-02, -2.12E-02; -2.12E-02, 8.58E-02)
				3.14E-01	(3.31E-01, 1.83E-01)	(5.36E-03, -3.08E-05; -3.08E-05, 2.83E-05)
				3.71E-01	(7.01E-01, 1.80E-01)	(4.14E-03, -7.09E-05; -7.09E-05, 1.57E-05)
				1.22E-01	(5.22E-01, 2.31E-01)	(5.03E-02, -2.40E-04; -2.40E-04, 1.88E-04)
4 (118)	[10.361, 84.917]	[0.173, 22.587]	19	1.67E-01	(2.39E-01, 6.28E-01)	(2.09E-02, 1.09E-02; 1.09E-02, 2.39E-02)
				2.78E-01	(6.98E-01, 4.23E-01)	(1.19E-02, 8.69E-03; 8.69E-03, 4.21E-02)
				1.69E-01	(3.66E-01, 1.95E-01)	(2.12E-02, 2.11E-05; 2.11E-05, 1.60E-02)
				3.86E-01	(7.55E-01, 1.16E-01)	(5.64E-03, -4.51E-04; -4.51E-04, 6.89E-03)
5 (103)	[28.775, 92.834]	[5.157, 22.305]	0	5.52E-01	(8.47E-01, 1.66E-01)	(6.04E-03, 2.45E-04; 2.45E-04, 3.76E-04)
				3.13E-01	(3.42E-01, 6.77E-01)	(1.94E-02, 1.34E-02; 1.34E-02, 1.47E-02)
				1.35E-01	(6.48E-01, 1.90E-01)	(5.26E-02, 5.57E-03; 5.57E-03, 5.12E-03)
6 (166)	[8.908, 87.13]	[2.005, 22.813]	17	3.91E-01	(7.87E-01, 2.45E-01)	(1.49E-02, 5.41E-04; 5.41E-04, 1.70E-03)
				1.71E-01	(2.25E-01, 6.31E-01)	(1.58E-02, 5.23E-03; 5.23E-03, 4.51E-02)
				2.37E-01	(6.26E-01, 8.33E-01)	(1.31E-02, 8.17E-03; 8.17E-03, 8.76E-03)
				2.02E-01	(8.03E-01, 3.93E-01)	(1.13E-02, 1.75E-03; 1.75E-03, 4.64E-02)
7 (163)	[28.38, 93.954]	[0.174, 23.301]	0	2.72E-01	(1.32E-01, 6.51E-01)	(1.04E-02, 2.33E-03; 2.33E-03, 1.36E-02)
				3.03E-01	(7.48E-01, 4.82E-01)	(1.09E-02, -3.67E-03; -3.67E-03, 4.33E-02)
				4.24E-01	(6.30E-01, 1.95E-01)	(2.92E-02, 2.00E-04; 2.00E-04, 8.27E-04)
8 (116)	[11.333, 83.927]	[0.006, 23.921]	13	1.96E-01	(9.58E-01, 1.65E-01)	(1.72E-03, 5.36E-04; 5.36E-04, 3.56E-03)
				2.25E-01	(7.33E-01, 1.53E-01)	(3.21E-03, 3.08E-04; 3.08E-04, 4.84E-04)
				4.89E-01	(8.94E-01, 1.43E-01)	(4.13E-03, 2.36E-04; 2.36E-04, 1.26E-04)
				8.99E-02	(1.83E-01, 7.60E-01)	(2.07E-02, 8.54E-03; 8.54E-03, 1.04E-02)

Table 2.7: Summary of estimated parameters of Model II based on customer flexibility profiles for washing machine

Customer (Data Size)	$[X_{min}, X_{max}]$	$[Y_{min}, Y_{max}]$	Ref.	η_k	μ_k	Σ^k=($\Sigma_{11}, \Sigma_{12}; \Sigma_{21}, \Sigma_{22}$)
1 (105)	[12.185, 82.902]	[4.184, 22.734]	16	1.33E-01	(5.74E-01, 2.39E-01)	(5.40E-02, -1.50E-02; -1.50E-02, 2.02E-02)
				3.55E-01	(5.75E-01, 1.78E-01)	(1.49E-03, -3.92E-05; -3.92E-05, 4.24E-04)
				6.53E-02	(2.68E-01, 8.82E-01)	(1.71E-02, 1.99E-03; 1.99E-03, 2.33E-03)
				4.47E-01	(7.08E-01, 1.77E-01)	(5.80E-03, -1.07E-05; -1.07E-05, 1.45E-04)
2 (253)	[4.828, 91.584]	[2.455, 23.077]	21	1.45E-01	(5.90E-01, 1.61E-01)	(3.57E-02, 1.61E-03; 1.61E-03, 7.50E-03)
				2.53E-01	(8.81E-01, 1.92E-01)	(4.96E-03, 5.29E-05; 5.29E-05, 4.71E-03)
				1.38E-01	(2.16E-01, 5.52E-01)	(1.06E-02, 7.61E-03; 7.61E-03, 2.31E-02)
				1.34E-01	(2.60E-01, 8.20E-01)	(1.11E-02, 9.08E-04; 9.08E-04, 1.12E-02)
				8.11E-02	(7.28E-01, 6.34E-01)	(2.88E-02, -4.19E-03; -4.19E-03, 1.48E-02)
				2.49E-01	(5.48E-01, 8.29E-01)	(1.72E-02, 7.79E-04; 7.79E-04, 2.43E-03)
3 (122)	[8.626, 94.296]	[4.369, 22.527]	0	3.82E-01	(7.01E-01, 1.80E-01)	(3.49E-03, -6.93E-05; -6.93E-05, 1.56E-05)
				2.96E-01	(4.87E-01, 3.41E-01)	(5.67E-02, -1.74E-02; -1.74E-02, 6.43E-02)
				3.21E-01	(3.31E-01, 1.83E-01)	(5.00E-03, -3.14E-05; -3.14E-05, 2.79E-05)
4 (133)	[11.301, 84.537]	[5.742, 21.778]	15	1.67E-01	(2.39E-01, 6.28E-01)	(2.09E-02, 1.09E-02; 1.09E-02, 2.39E-02)
				2.78E-01	(6.98E-01, 4.23E-01)	(1.19E-02, 8.69E-03; 8.69E-03, 4.21E-02)
				1.69E-01	(3.66E-01, 1.95E-01)	(2.12E-02, 2.11E-05; 2.11E-05, 1.60E-02)
				3.86E-01	(7.55E-01, 1.16E-01)	(5.64E-03, -4.51E-04; -4.51E-04, 6.89E-03)
5 (100)	[2.206, 92.976]	[0.605, 23.737]	63	5.39E-01	(8.47E-01, 1.66E-01)	(5.87E-03, 2.47E-04; 2.47E-04, 3.74E-04)
				1.20E-01	(2.69E-01, 6.62E-01)	(4.57E-02, 3.67E-02; 3.67E-02, 3.96E-02)
				1.34E-01	(6.53E-01, 1.89E-01)	(5.13E-02, 5.50E-03; 5.50E-03, 5.06E-03)
				2.07E-01	(3.66E-01, 6.74E-01)	(3.45E-03, 8.79E-04; 8.79E-04, 1.04E-03)
6 (109)	[14.609, 81.239]	[2.258, 23.972]	19	2.13E-01	(6.32E-01, 9.23E-01)	(2.69E-02, 2.16E-03; 2.16E-03, 4.19E-03)
				3.31E-01	(2.26E-01, 5.48E-01)	(1.40E-02, 6.31E-03; 6.31E-03, 1.13E-02)
				1.41E-01	(5.47E-01, 1.33E-01)	(3.21E-02, -5.03E-03; -5.03E-03, 7.68E-03)
				3.15E-01	(8.50E-01, 2.94E-01)	(6.80E-03, 3.78E-04; 3.78E-04, 1.71E-02)
7 (103)	[5.258, 95.513]	[2.025, 21.795]	0	2.61E-01	(2.59E-01, 2.12E-01)	(2.07E-02, -1.93E-03; -1.93E-03, 7.93E-03)
				4.43E-01	(8.34E-01, 2.41E-01)	(1.09E-02, 1.88E-03; 1.88E-03, 4.99E-03)
				1.07E-01	(5.88E-01, 7.62E-01)	(1.26E-02, 6.44E-05; 6.44E-05, 1.68E-02)
				1.89E-01	(3.10E-01, 8.72E-01)	(4.00E-03, 3.76E-04; 3.76E-04, 7.59E-03)

(Continued on next page)

Table 2.7: Summary of estimated parameters of Model II based on customer flexibility profiles for washing machine *(continued)*

Customer (Data Size)	$[X_{min}, X_{max}]$	$[Y_{min}, Y_{max}]$	Ref.	η_k	μ_k	Σ^k=($\Sigma_{11}, \Sigma_{12}; \Sigma_{21}, \Sigma_{22}$)
8 (139)	[4.010, 91.115]	[0.377, 23.523]	32	5.56E-01	(2.85E-01, 6.90E-01)	(2.87E-02, 2.63E-02; 2.63E-02, 2.59E-02)
				2.83E-01	(6.70E-01, 3.72E-01)	(1.07E-02, 1.29E-03; 1.29E-03, 4.98E-03)
				1.60E-01	(4.82E-01, 5.76E-01)	(7.16E-02, -1.96E-02; -1.96E-02, 5.59E-02)
9 (110)	[3.688, 91.611]	[4.907, 22.561]	13	3.84E-01	(4.30E-01, 7.52E-01)	(5.01E-03, 9.85E-04; 9.85E-04, 1.42E-03)
				3.11E-01	(8.90E-01, 1.45E-01)	(2.65E-03, 3.09E-04; 3.09E-04, 2.12E-03)
				3.06E-01	(1.81E-01, 5.21E-01)	(2.75E-02, 5.51E-03; 5.51E-03, 4.18E-02)
10 (104)	[4.023, 92.217]	[0.323, 23.781]	26	9.18E-02	(2.76E-01, 1.30E-01)	(2.77E-02, -9.61E-04; -9.61E-04, 2.60E-04)
				2.18E-01	(7.93E-01, 1.10E-01)	(1.01E-02, -1.11E-03; -1.11E-03, 2.16E-03)
				2.98E-01	(8.71E-01, 1.21E-01)	(4.04E-03, -6.91E-05; -6.91E-05, 8.53E-05)
				3.44E-01	(8.59E-01, 1.52E-01)	(5.19E-03, 7.90E-05; 7.90E-05, 6.73E-05)
				4.82E-02	(2.42E-01, 9.92E-01)	(1.99E-02, 6.14E-04; 6.14E-04, 2.02E-04)
11 (218)	[3.635, 91.896]	[3.700, 21.281]	18	1.08E-01	(7.34E-01, 2.77E-01)	(6.87E-03, -2.04E-03; -2.04E-03, 4.41E-03)
				5.36E-01	(8.26E-01, 2.59E-01)	(2.98E-03, -2.82E-05; -2.82E-05, 2.15E-04)
				6.22E-02	(9.23E-01, 3.76E-01)	(3.99E-03, 1.05E-03; 1.05E-03, 3.06E-03)
				2.31E-01	(8.60E-01, 3.10E-01)	(1.10E-03, -3.84E-06; -3.84E-06, 3.80E-04)
				6.19E-02	(2.48E-01, 8.72E-01)	(1.66E-02, 6.77E-03; 6.77E-03, 4.63E-03)
12 (173)	[3.442, 92.674]	[0.143, 23.429]	20	5.94E-01	(2.03E-01, 6.02E-02)	(1.76E-03, -1.86E-05; -1.86E-05, 2.84E-05)
				2.84E-01	(2.64E-01, 6.15E-02)	(7.07E-03, 5.14E-05; 5.14E-05, 1.99E-05)
				2.19E-02	(2.42E-02, 9.99E-01)	(8.98E-03, 1.54E-04; 1.54E-04, 9.65E-05)
				5.64E-02	(4.61E-01, 5.88E-02)	(1.17E-02, 2.02E-04; 2.02E-04, 3.64E-05)
				4.42E-02	(7.01E-01, 5.34E-02)	(6.89E-02, 6.36E-03; 6.36E-03, 2.47E-03)

Table 2.8: Summary of estimated parameters of Model II based on customer flexibility profiles for dishwasher

Customer (Data Size)	$[X_{min}, X_{max}]$	$[Y_{min}, Y_{max}]$	Ref.	η_k	μ_k	Σ^k=($\Sigma_{11}, \Sigma_{12}; \Sigma_{21}, \Sigma_{22}$)
1 (199)	[16.814, 79.651]	[2.350, 21.473]	55	6.65E-02	(5.62E-01, 5.17E-02)	(5.66E-02, 4.10E-03; 4.10E-03, 1.66E-03)
				2.36E-02	(9.71E-03, 1.00E+00)	(1.61E-02, 1.14E-04; 1.14E-04, 1.17E-04)
				9.10E-01	(2.23E-01, 6.05E-02)	(4.28E-03, 2.07E-05; 2.07E-05, 2.59E-05)

(Continued on next page)

Table 2.8: Summary of estimated parameters of Model II based on customer flexibility profiles for dishwasher *(continued)*

Customer (Data Size)	$[X_{min}, X_{max}]$	$[Y_{min}, Y_{max}]$	Ref.	η_k	μ_k	Σ^k=(Σ_{11}, Σ_{12}; Σ_{21}, Σ_{22})
2 (105)	[13.308, 82.571]	[1.634, 17.467]	15	1.87E-01	(7.92E-01, 9.49E-02)	(1.14E-02, -4.53E-04; -4.53E-04, 1.75E-03)
				5.03E-02	(2.42E-01, 9.92E-01)	(2.39E-02, 3.19E-04; 3.19E-04, 8.45E-04)
				9.68E-02	(2.86E-01, 1.29E-01)	(3.16E-02, -1.27E-03; -1.27E-03, 5.81E-04)
				6.66E-01	(8.60E-01, 1.38E-01)	(4.94E-03, -2.35E-04; -2.35E-04, 4.16E-04)
3 (202)	[9.193, 87.692]	[3.805, 15.130]	69	8.04E-02	(8.46E-02, 1.28E-01)	(3.41E-03, 1.23E-04; 1.23E-04, 1.08E-03)
				5.63E-01	(2.83E-01, 1.06E-01)	(2.29E-03, 2.85E-05; 2.85E-05, 1.01E-04)
				6.56E-02	(7.38E-01, 7.35E-01)	(1.23E-02, -9.87E-03; -9.87E-03, 5.89E-02)
				1.48E-01	(2.20E-01, 1.30E-01)	(4.39E-03, -1.02E-03; -1.02E-03, 1.14E-02)
				1.43E-01	(3.12E-01, 1.90E-01)	(1.81E-03, -1.30E-04; -1.30E-04, 2.46E-04)
4 (246)	[12.292, 82.805]	[0.175, 16.604]	17	5.23E-01	(8.24E-01, 2.58E-01)	(3.07E-03, -3.33E-05; -3.33E-05, 2.04E-04)
				1.71E-01	(7.95E-01, 3.02E-01)	(1.14E-02, 1.56E-03; 1.56E-03, 5.47E-03)
				6.11E-02	(2.49E-01, 8.73E-01)	(1.62E-02, 6.63E-03; 6.63E-03, 4.57E-03)
				2.46E-01	(8.59E-01, 3.08E-01)	(1.25E-03, 2.08E-04; 2.08E-04, 5.58E-04)
5 (129)	[15.868, 79.402]	[0.281, 18.923]	17	1.96E-01	(9.58E-01, 1.65E-01)	(1.72E-03, 5.36E-04; 5.36E-04, 3.56E-03)
				2.25E-01	(7.33E-01, 1.53E-01)	(3.21E-03, 3.08E-04; 3.08E-04, 4.84E-04)
				4.89E-01	(8.94E-01, 1.43E-01)	(4.13E-03, 2.36E-04; 2.36E-04, 1.26E-04)
				8.99E-02	(1.83E-01, 7.60E-01)	(2.07E-02, 8.54E-03; 8.54E-03, 1.04E-02)
6 (113)	[9.579, 85.951]	[0.121, 21.758]	18	3.05E-01	(3.31E-01, 1.83E-01)	(5.58E-03,-2.40E-05; -2.40E-05, 2.81E-05)
				9.94E-02	(4.83E-01, 1.89E-01)	(4.25E-02,-1.22E-05; -1.22E-05, 4.79E-05)
				1.02E-01	(5.75E-01, 1.92E-01)	(4.20E-02,4.29E-05; 4.29E-05, 6.22E-05)
				3.65E-01	(7.03E-01, 1.81E-01)	(5.10E-03,-6.01E-05; -6.01E-05, 1.58E-05)
				1.29E-01	(4.00E-01, 5.57E-01)	(5.20E-02,-1.28E-02; -1.28E-02, 7.99E-02)
7 (285)	[7.202, 88.372]	[2.383,18.716]	22	1.33E-01	(5.74E-01, 2.39E-01)	(5.40E-02, -1.50E-02; -1.50E-02, 2.02E-02)
				3.55E-01	(5.75E-01, 1.78E-01)	(1.49E-03, -3.92E-05; -3.92E-05, 4.24E-04)
				6.53E-02	(2.68E-01, 8.82E-01)	(1.71E-02, 1.99E-03; 1.99E-03, 2.33E-03)
				4.47E-01	(7.08E-01, 1.77E-01)	(5.80E-03, -1.07E-05; -1.07E-05, 1.45E-04)
8 (168)	[11.963, 84.778]	[0.345, 22.742]	16	2.01E-01	(2.62E-01, 6.53E-01)	(3.37E-02, 1.20E-02; 1.20E-02, 2.47E-02)
				1.20E-01	(7.41E-01, 2.67E-01)	(1.64E-02, -2.96E-04; -2.96E-04, 8.23E-03)
				5.50E-01	(7.81E-01, 2.20E-01)	(1.02E-02, 2.82E-04; 2.82E-04, 1.41E-04)
				1.29E-01	(3.00E-01, 2.26E-01)	(3.60E-02, 2.01E-04; 2.01E-04, 3.28E-04)

(Continued on next page)

Table 2.8: Summary of estimated parameters of Model II based on customer flexibility profiles for dishwasher *(continued)*

Customer (Data Size)	$[X_{min}, X_{max}]$	$[Y_{min}, Y_{max}]$	Ref.	η_k	μ_k	Σ^k=(Σ_{11}, Σ_{12}; Σ_{21}, Σ_{22})
9 (186)	[13.429, 82.243]	[0.390, 23.384]	16	1.02E-01	(8.77E-01,2.24E-01)	(5.10E-03, 9.73E-04; 9.73E-04, 2.10E-03)
				5.46E-02	(5.07E-01, 9.15E-01)	(1.25E-02, 1.19E-04; 1.19E-04, 3.05E-03)
				2.32E-01	(3.63E-01, 6.74E-01)	(3.87E-03, 1.05E-03; 1.05E-03, 1.31E-03)
				1.00E-01	(2.57E-01, 2.81E-01)	(4.21E-02, -2.69E-02; -2.69E-02, 2.47E-02)
				5.11E-01	(8.40E-01, 1.65E-01)	(6.31E-03, 1.85E-04; 1.85E-04, 3.67E-04)
10 (141)	[11.126, 84.580]	[4.143, 21.225]	18	1.83E-01	(6.95E-01, 1.39E-01)	(1.73E-02, 4.82E-03; 4.82E-03, 9.42E-03)
				2.47E-01	(2.23E-01, 4.66E-01)	(1.13E-02, 1.72E-02; 1.72E-02, 5.01E-02)
				2.25E-01	(6.67E-01, 6.23E-01)	(4.66E-03, 1.61E-03; 1.61E-03, 2.78E-02)
				3.45E-01	(7.46E-02, 5.91E-01)	(2.71E-03, 8.89E-05; 8.89E-05, 1.54E-02)
11 (116)	[12.960, 83.511]	[0.028, 23.573]	20	3.91E-01	(7.87E-01, 2.45E-01)	(1.49E-02, 5.41E-04; 5.41E-04, 1.70E-03)
				1.71E-01	(2.25E-01, 6.31E-01)	(1.58E-02, 5.23E-03; 5.23E-03, 4.51E-02)
				2.37E-01	(6.26E-01, 8.33E-01)	(1.31E-02, 8.17E-03; 8.17E-03, 8.76E-03)
				2.02E-01	(8.03E-01, 3.93E-01)	(1.13E-02, 1.75E-03; 1.75E-03, 4.64E-02)
12 (199)	[4.797, 91.170]	[0.381, 23.127]	23	3.61E-01	(6.10E-01, 1.93E-01)	(2.90E-02, -5.22E-05; -5.22E-05, 6.16E-04)
				1.61E-01	(9.97E-02, 6.86E-01)	(6.50E-03, -1.27E-04; -1.27E-04, 1.14E-03)
				2.28E-01	(7.59E-01, 3.02E-01)	(9.53E-03, -1.69E-03; -1.69E-03, 1.21E-02)
				1.01E-01	(7.70E-01, 6.61E-01)	(8.24E-03, -2.98E-04; -2.98E-04, 1.22E-03)
				1.49E-01	(2.83E-01, 6.45E-01)	(4.79E-02, 2.51E-02; 2.51E-02, 3.13E-02)
13 (270)	[6.595, 90.142]	[2.639, 19.398]	8	1.65E-01	(8.25E-01, 7.16E-01)	(4.81E-03, 8.75E-04; 8.75E-04, 2.68E-03)
				1.96E-01	(3.90E-01, 7.38E-01)	(2.26E-02, 1.99E-02; 1.99E-02, 2.31E-02)
				2.09E-01	(8.19E-01, 1.09E-01)	(5.31E-03, -1.33E-04; -1.33E-04, 2.21E-03)
				1.83E-01	(1.52E-01, 7.11E-01)	(4.90E-03, 2.83E-03; 2.83E-03, 1.06E-02)
				2.47E-01	(8.29E-01, 3.68E-01)	(6.68E-03, 7.99E-04; 7.99E-04, 1.33E-02)
14 (139)	[30.056, 91.676]	[0.164, 22.142]	0	6.94E-02	(2.72E-01, 1.76E-01)	(2.38E-02, -2.95E-03; -2.95E-03, 1.70E-03)
				1.65E-01	(2.96E-01, 7.17E-01)	(4.37E-02, 1.11E-02; 1.11E-02, 1.91E-02)
				2.94E-01	(8.90E-01, 1.46E-01)	(2.58E-03, 2.36E-04; 2.36E-04, 1.96E-03)
				3.35E-01	(4.25E-01, 7.50E-01)	(3.43E-03, 1.39E-04; 1.39E-04, 7.54E-04)
				1.36E-01	(8.13E-02, 4.80E-01)	(3.46E-03, 7.27E-04; 7.27E-04, 1.76E-03)

(Continued on next page)

Table 2.8: Summary of estimated parameters of Model II based on customer flexibility profiles for dishwasher *(continued)*

Customer (Data Size)	$[X_{min}, X_{max}]$	$[Y_{min}, Y_{max}]$	Ref.	η_k	μ_k	Σ^k=($\Sigma_{11}, \Sigma_{12}; \Sigma_{21}, \Sigma_{22}$)
15 (253)	[9.304, 85.805]	[3.928, 23.743]	18	1.67E-01	(2.39E-01, 6.28E-01)	(2.09E-02, 1.09E-02; 1.09E-02, 2.39E-02)
				2.78E-01	(6.98E-01, 4.23E-01)	(1.19E-02, 8.69E-03; 8.69E-03, 4.21E-02)
				1.69E-01	(3.66E-01, 1.95E-01)	(2.12E-02, 2.11E-05; 2.11E-05, 1.60E-02)
				3.86E-01	(7.55E-01, 1.16E-01)	(5.64E-03, -4.51E-04; -4.51E-04, 6.89E-03)

References

[1] N. Sadeghianpourhamami, M. Strobbe, and C. Develder. *Real-World User Flexibility of Energy Consumption: Two-Stage Generative Model Construction*. In Proc. ACM Symp. Appl. Comput. (SAC2016), pages 1–6, Apr 2016.

[2] P. Siano. *Demand response and smart grids—A survey*. Renewable and Sustainable Energy Reviews, 30:461 – 478, 2014. doi:10.1016/j.rser.2013.10.022.

[3] J. Vardakas, N. Zorba, and C. Verikoukis. *A Survey on Demand Response Programs in Smart Grids: Pricing Methods and Optimization Algorithms*. IEEE Commun. Surveys Tuts, 17(1):152–178, 2015. doi:10.1109/COMST.2014.2341586.

[4] S. Nolan and M. O'Malley. *Challenges and barriers to demand response deployment and evaluation*. Appl. Energy, 152:1–10, 2015. doi:10.1016/j.apenergy.2015.04.083.

[5] M. Alcázar-Ortega, C. Calpe, T. Theisen, and J. F. Carbonell-Carretero. *Methodology for the identification, evaluation and prioritization of market handicaps which prevent the implementation of Demand Response: Application to European electricity markets*. Energy Policy, 86(C):529–543, 2015.

[6] C. Goldman, N. Hopper, R. Bharvirkar, B. Neenan, and P. Cappers. *Estimating demand response market potential among large commercial and industrial customers: a scoping study*. Lawrence Berkeley National Laboratory, 2007.

[7] *Demand Response from Day-Ahead Hourly Pricing for Large Customers*. The Electricity Journal, 19(3):52 – 63, 2006. doi:10.1016/j.tej.2006.02.002.

[8] D. Jang, J. Eom, M. G. Kim, and J. J. Rho. *Demand responses of Korean commercial and industrial businesses to critical peak pricing of electricity*. Journal of Cleaner Production, 90:275 – 290, 2015. Available from: http://www.sciencedirect.com/science/article/pii/S0959652614012396, doi:http://dx.doi.org/10.1016/j.jclepro.2014.11.052.

[9] R. Deshmukh, G. Ghatikar, R. Yin, G. G. Das, and S. K. Saha. *Estimation of potential and value of demand response for industrial and commercial consumers in Delhi*. Presented at the India Smart Grid Week (ISGW), 2015.

[10] Z. Liu, I. Liu, S. Low, and A. Wierman. *Pricing data center demand response*. In ACM SIGMETRICS Performance Evaluation Review, volume 42, pages 111–123. ACM, 2014.

[11] V. Gomez, M. Chertkov, S. Backhaus, and H. Kappen. *Learning price-elasticity of smart consumers in power distribution systems*. In Proc. 3rd IEEE Int. Conf. Smart Grid Commun. (SmartGridComm2012), pages 647–652, Nov 2012. doi:10.1109/SmartGridComm.2012.6486059.

[12] A. van Stiphout, J. Engels, D. Guldentops, and G. Deconinck. *Quantifying the flexibility of residential electricity demand in 2050: a bottom-up approach*. In PowerTech, IEEE Eindhoven, pages 1–6, Jun 2015. doi:10.1109/PTC.2015.7232411.

[13] P. Thimmapuram and J. Kim. *Consumers' Price Elasticity of Demand Modeling With Economic Effects on Electricity Markets Using an Agent-Based Model*. IEEE Trans. Smart Grid, 4(1):390–397, Mar 2013. doi:10.1109/TSG.2012.2234487.

[14] A. Moshari, G. Yousefi, A. Ebrahimi, and S. Haghbin. *Demand-side behavior in the smart grid environment*. In Proc. 1st IEEE PES Innovative Smart Grid Technol. Conf. Europe (ISGT Europe), pages 1–7, Oct 2010. doi:10.1109/ISGTEUROPE.2010.5638956.

[15] N. Paterakis, J. Catalao, A. Tascikaraoglu, A. Bakirtzis, and O. Erdinc. *Demand response driven load pattern elasticity analysis for smart households*. In Proc. 5th IEEE Int. Conf. Power Engineering, Energy and Electrical Drives (POWERENG), pages 399–404, May 2015. doi:10.1109/PowerEng.2015.7266350.

[16] P. Siano and D. Sarno. *Assessing the benefits of residential demand response in a real time distribution energy market*. Appl. Energy, 161:533–551, 2016. doi:10.1016/j.apenergy.2015.10.017.

[17] Q. Hu, X. Fang, F. Li, X. Xu, C. fei Chen, and H. Hu. *An approach to assess the responsive residential demand to financial incentives*. In IEEE PES General Meeting, pages 1–5, Jul 2015. doi:10.1109/PESGM.2015.7286623.

[18] M. Pipattanasomporn, M. Kuzlu, S. Rahman, and Y. Teklu. *Load Profiles of Selected Major Household Appliances and Their Demand Response Opportunities*. IEEE Trans. Smart Grid, 5(2):742–750, Mar 2014. doi:10.1109/TSG.2013.2268664.

[19] F. Wattjes, S. Janssen, and J. Slootweg. *Framework for estimating flexibility of commercial and industrial customers in Smart Grids*. In Proc. 4th IEEE PES Innovative Smart Grid Technol. Conf. Europe (ISGT Europe), pages 1–5, Oct 2013. doi:10.1109/ISGTEurope.2013.6695406.

[20] I. Laicane, D. Blumberga, A. Blumberga, and M. Rosa. *Reducing Household Electricity Consumption through Demand Side Management: The Role of Home Appliance Scheduling and Peak Load Reduction*. Energy Procedia, 72:222 – 229, 2015. doi:10.1016/j.egypro.2015.06.032.

[21] A. Safdarian, M. Fotuhi-Firuzabad, and M. Lehtonen. *Benefits of Demand Response on Operation of Distribution Networks: A Case Study*. IEEE Syst. J., PP(99):1–9, 2014. doi:10.1109/JSYST.2013.2297792.

[22] K. Kouzelis, Z. Tan, B. Bak-Jensen, J. Pillai, and E. Ritchie. *Estimation of Residential Heat Pump Consumption for Flexibility Market Applications*. IEEE Trans. Smart Grid, 6(4):1852–1864, Jul 2015. doi:10.1109/TSG.2015.2414490.

[23] W. Labeeuw, J. Stragier, and G. Deconinck. *Potential of Active Demand Reduction With Residential Wet Appliances: A Case Study for Belgium*. IEEE Trans. Smart Grid, 6(1):315–323, Jan 2015. doi:10.1109/TSG.2014.2357343.

[24] C. B. Kobus, E. A. Klaassen, R. Mugge, and J. P. Schoormans. *A real-life assessment on the effect of smart appliances for shifting households' electricity demand*. Appl. Energy, 147:335–343, 2015. doi:10.1016/j.apenergy.2015.01.073.

[25] R. D'hulst, W. Labeeuw, B. Beusen, S. Claessens, G. Deconinck, and K. Vanthournout. *Demand response flexibility and flexibility potential of residential smart appliances: Experiences from large pilot test in Belgium*. Appl. Energy, 155:79–90, 2015. doi:10.1016/j.apenergy.2015.05.101.

[26] B. Dupont, P. Vingerhoets, P. Tant, K. Vanthournout, W. Cardinaels, T. De Rybel, E. Peeters, and R. Belmans. *LINEAR breakthrough project: Large-scale implementation of smart grid technologies in distribution grids*. In Proc. 3rd IEEE PES Innovative Smart Grid Technol. Conf. Europe (ISGT Europe), pages 1–8, Oct 2012. doi:10.1109/ISGTEurope.2012.6465708.

[27] I. Sajjad, G. Chicco, M. Aziz, and A. Rasool. *Potential of residential demand flexibility - Italian scenario*. In Proc. 11th IEEE Int. Multi-Conf. Systems, Signals Devices (SSD),, pages 1–6, Feb 2014. doi:10.1109/SSD.2014.6808849.

[28] K. Vanthournout, B. Dupont, W. Foubert, C. Stuckens, and S. Claessens. *An automated residential demand response pilot experiment, based on day-ahead dynamic pricing*. Appl. Energy, 155:195 – 203, 2015. doi:10.1016/j.apenergy.2015.05.100.

[29] G. Hamerly and C. Elkan. *Learning the K in K-Means*. In Proc. 17th Ann. Conf. Neural Information Processing Systems, Dec 2003.

[30] K. Mets, F. Depuydt, and C. Develder. *Two-Stage Load Pattern Clustering Using Fast Wavelet Transformation*. IEEE Trans. Smart Grid, PP(99), 2015. doi:10.1109/TSG.2015.2446935.

[31] G. Chicco. *Overview and performance assessment of the clustering methods for electrical load pattern grouping*. Energy, 42(1):68 – 80, 2012. doi:10.1016/j.energy.2011.12.031.

[32] S. Frühwirth-Schnatter. *Finite mixture and Markov switching models*. Springer series in statistics. Springer, New York, 2006.

[33] H. Akaike. *A new look at the statistical model identification*. IEEE Trans. Autom. Control, 19(6):716–723, Dec 1974. doi:10.1109/TAC.1974.1100705.

[34] G. Schwarz. *Estimating the Dimension of a Model*. The Ann. Statist., 6(2):461–464, 1978. doi:10.2307/2958889.

[35] W. Xie, P. O. Lewis, Y. Fan, L. Kuo, and M.-H. Chen. *Improving Marginal Likelihood Estimation for Bayesian Phylogenetic Model Selection*. Systematic Biology, 60(2):150–160, 2011. doi:10.1093/sysbio/syq085.

[36] M. Stephens. *Bayesian analysis of mixture models with an unknown number of components—an alternative to reversible jump methods*. Ann. Statist., 28(1):40–74, 02 2000. doi:10.1214/aos/1016120364.

[37] J. A. Peacock. *Two-dimensional goodness-of-fit testing in astronomy*. Monthly Notices of the Royal Astronomical Society, 202(3):615–627, 1983. doi:10.1093/mnras/202.3.615.

[38] G. H. Freeman and J. H. Halton. *Note on an Exact Treatment of Contingency, Goodness of Fit and Other Problems of Significance*. Biometrika, 38(1/2):141–149, 1951.

3

Quantitive Analysis of Electric Vehicle Flexibility: A Data-Driven Approach

In the previous chapter, we focused on modeling and analysis of the residential white-good usage flexibility as one of the demand response resources used to secure demand-supply balance in the smart grid paradigm. In this chapter we perform a conceptually similar analysis, but now on a substantially larger dataset of EV charging session. Electric vehicles proliferation is following a fast pace partly due to various environmental initiatives. The batteries of Electric vehicles have a great potential for providing distributed energy consumption flexibility required by demand response algorithms. Understanding and a quantification of such flexibility as a demand response asset is essential to ensure informed assumptions while developing demand response algorithms (as we will do in Chapter 5).

⋆ ⋆ ⋆

N. Sadeghianpourhamami, N. Refa , M. Strobbe, and C. Develder.

Published in International Journal of Electrical Power & Energy Systems, Feb. 2018.[1]

[1] The appendix in this chapter is not part of the published paper and is added in this dissertation to provide further clarification as requested by the examiners.

Abstract The electric vehicle (EV) flexibility, indicates to what extent the charging load can be coordinated (i.e., to flatten the load curve or to utilize renewable energy resources). However, such flexibility is neither well analyzed nor effectively quantified in literature. In this paper we fill this gap and offer an extensive analysis of the flexibility characteristics of 390k EV charging sessions and propose measures to quantize their flexibility exploitation. Our contributions include: (1) characterization of the EV charging behavior by clustering the arrival and departure time combinations that leads to the identification of type of EV charging behavior, (2) in-depth analysis of the characteristics of the charging sessions in each behavioral cluster and investigation of the influence of weekdays and seasonal changes on those characteristics including arrival, sojourn and idle times, and (3) proposing measures and an algorithm to quantitatively analyze how much flexibility (in terms of duration and amount) is used at various times of a day, for two representative scenarios. Understanding the characteristics of that flexibility (e.g., amount, time and duration of availability) and when it is used (in terms of both duration and amount) helps to develop more realistic price and incentive schemes in DR algorithms to efficiently exploit the offered flexibility or to estimate when to stimulate additional flexibility.

3.1 Introduction

Partly because of environmental constraints, electric vehicles (EVs) are increasingly being adopted as an alternative for internal combustion engine (ICE) cars. However, the load from EVs may increase the peak to average ratio of demand and hence create a need for additional generation and network capacity. That extra capacity would only be required to meet the increased peak demand and therefore is used very infrequently [1]. Integration of information technology into the power grid (in the smart grid paradigm) alleviates this challenge by enabling the exploitation of demand side flexibility to reshape the consumption to meet the supply or network constraints (i.e., by flattening demand or by balancing against renewable generation). Consequently, a substantial body of research has focused on proposing demand response (DR) algorithms to coordinate EV charging and establish their benefits (a review of various DR algorithms for charging coordination is given in [2], [3], [4], and [5]). However, one of the main limitations of such proposed DR algorithms is their potentially unrealistic assumptions about the EV owner behavior (e.g., time of availability of EV, sojourn times and the fraction of the sojourn time that is not spent for charging and is named idle time). To design an efficient and practical DR algorithm, it is necessary to accurately understand the flexibility stemming from EVs and how to influence it (through price based and incentive based schemes) to maximize DR benefits. However, despite various efforts in proposing DR algorithms, EV flexibility characteristics as DR's main asset have not been quantitatively analyzed. We believe such analysis can pave the way to more realistic demand response schemes (price-based or incentive based DR) in order to facilitate EV integration in the grid and therefore is the focus of this paper.

3.1.1 Objectives and Contributions

Understanding the flexibility characteristics, the influencing factors, and the motivation for its exploitation is an inevitable part of designing a realistic DR algorithm. Flexibility, despite its apparent simplicity, is neither straightforward to analyze nor to quantify.

We pursue two objectives in this paper. Our first objective is to perform an in depth analysis of the flexibility characteristics of EVs based on a reasonably large real-world dataset (which to the best of our knowledge amounts to the largest dataset reported in literature, see Section 3.2.1 for further details). Our second objective is to quantify the flexibility exploitation and identify how the observed flexibility is utilized for various objectives (e.g., load flattening and load balancing against renewable (energy) sources) and whether there is any typical pattern in its exploitation. More precisely, we aim to answer the following research questions:

1. *Do EV owners have specific habits to charge their cars (e.g., taking their cars to a charging station at particular times of the day)?* To answer this question, we characterize the EV charging behavior by clustering the arrival and departure time combinations, as such identifying three behavioral clusters in our EV charging data (Section 3.2.2).

2. *Are the characteristics of the charging sessions (e.g., arrival, sojourn and idle times) sensitive to seasonal changes or weekdays?* To address this question, we systematically analyze the characteristics of the charging sessions in each behavioral cluster on weekdays and weekends and across various seasons. We also characterize the flexibility stemming from the sojourn times of EVs that are longer than the time required to (fully) charge their battery (Section 3.2.3).

3. *How is flexibility (in terms of amount, time and duration of the shifted energy) exploited? Which aspect of flexibility (time and duration of availability or amount of deferrable energy) is more useful at various times of the day?* We address these questions by considering two case studies (i.e., load flattening and load balancing scenarios) to investigate to what extent the observed flexibility would be exploited. To do so, we propose two measures and an algorithm to quantitatively analyze when flexibility is used in terms of the EV load volume as well as amount of time the load is deferred(Section 3.3.3 and Section 3.3.4).

3.1.2 Related Work

Estimating the EV charging load to assess its impact on the power grid has been the primary focus of research in facilitating EVs integration to the grid. In initial studies, before the wide-spread use of EVs, probabilistic models of driving behavior (with conventional ICE cars) were used to characterize a charging session. This was done by estimating arrival and departure patterns, energy requirements and the covered distance in between trips. For example Lampropoulos *et al.* [6] derive an EV charging data profile from statistical characteristics of the driving behavior of conventional ICE cars. Clement-Nyns *et al.* [7] base their analysis on extrapolation of non-EV car usage in Belgium. Paevere *et al.* [8] model the spatio-temporal impact of EV load based on a linked suite of models of future EV uptake, their travel and charging/discharging models. Grahn *et al.* [9] derive EV charging behavior from non-EV driving behavior in Sweden. Pashajavid *et al.* [10] derive the demand profile of EVs from traveling and refueling information of non-EV in Tehran, and a more recent study [11] estimates possible states of EVs, regarding their demand, location and connection period, based on synthetic data which mimics reality.

Later studies, when EV penetration had increased, relied on the availability of EV charging datasets to use data-driven approaches to model the charging behavior of EVs and assess their impact on the grid. For instance, Xydas *et al.* [12] characterize the charging demand of EVs by statistically analyzing and clustering a dataset of 22k sessions in UK. Khoo *et al.* [13] derive the impact of EV charging on peak load based on around 5k sessions from an Australian field trial and establish the expected impact on the total power demand in 2032-33 for the state of Victoria. Brady *et al.* [14] use a probabilistic charging module to translate the travel patterns of EVs into the respective power demand of the vehicles. Quiròs-Tortòs *et al.* [15] and Navarro-Espinosa *et al.* [16] use the probability distribution of start charging time and energy demanded during a connection of charging sessions in a one-year EV trial in Ireland to obtain the EV load demand and assess their impact in the low voltage distribution grid. The aforementioned works focus mainly on analyzing the impact of EVs on the load curve and do not provide any quantitative analysis of the flexibility characteristic of EV charging sessions. The objective of our analysis presented here rather is to quantify the flexibility of the EV load, and quantitatively study user behavior.

User modeling (not focusing on flexibility) has been the subject of earlier works to assess the influence of charging behavior of different user categories on the load curve. For example, Franke *et al.* [17] examine the psychological dynamics underlying charging behavior of EV users. Spoelstra [18] aims at understanding the charging behavior of EV users and the factors constituting such behavior. Khoo *et al.* [13] have modeled the charging sessions for households and EV fleets during weekends and on weekdays in terms of arrival times and energy demands.

Quiròs-Tortòs *et al.* [19] produce probability distribution functions (PDF) of different charging features (e.g., start charging time) for both weekdays and week-ends based on 68k samples from 221 residential EV users. They further discuss the effects of the EV demand on future UK distribution networks. Similarly, Richardson *et al.* [20] produce PDF of connection times and daily energy requirements of EV based on the charging behavior of 78 users for a duration of 1 year. Helmus *et al.* [21] distinguish a priori defined different user types (residents, commuters, taxis, etc.) and characterize them in terms of EV charging session start and end times and the associated energy needs. Similarly, Aunedi *et al.* [22] characterize the charging behavior and the demand diversity of two predefined user categories: residential users and commercial users. Instead of defining the user categories a priori, Xydas *et al.* [12] cluster the observed charging sessions into distinct types of behavior. They derive aggregate models for three specific geographical areas, characterized by different clusters of "typical EV charging demand profiles". Similar characterization of charging session timing is presented by Kara *et al.* [23]. Similar to [12] and [23] (but using different clustering technique), we cluster the EV charging sessions into behavioral clusters. However, our work differs from the aforementioned papers: instead of focusing on the impact of EVs on the load curve, we characterize the flexibility stemming from the EVs as well as how such flexibility is used (in terms of both amount and duration) to flatten the load or balance against renewable energy.

Quantification of demand side flexibility and assessing its impact on alleviating the EV charging burden on the grid has been tackled before. Aunedi *et al.* [22] characterized the flexibility of EV charging demand in terms of the amount of load shifted in time from the peak consumption without compromising the ability of EV users to make their intended journeys. Their analysis suggested that it is possible to shift 70% to 100% of EV demand from peak hours towards the night. Kara *et al.* [23] defined the flexibility matrix as the fraction of total connection time that is not spent on charging. They presented the variation of this measure over different months. Teng *et al.* [1] defined the potential flexibility of EV demand as the amount of the shifted energy in the coordinated vs. the uncoordinated charging. They further establish the benefits of this flexibility in reducing carbon emissions and cost of integration of renewable energy sources (RES) through appropriate measures. Pavić *et al.* [24] estimated the EV flexibility benefits for providing spinning reserve services through matrices expressed as operational costs, environmental benefits and reduced wind curtailment. Salah *et al.* [25] used the parking data from a car park in southern Germany, which is mainly used for shopping and working. They modeled parking duration distribution for two types of parking behavior: shopping and workplace. They inferred the flexibility thereof by assuming an average EV charging time of 45 min at 11 kW per car. Kheserzadeh [26] inferred the probability of availability of EVs in the parking lots for different EV owners including: residential, industrial and commercial customers (using the statistics of their traveling habits and traveling loads). The impact of various EV owners charging behavior on flattening the micro-grid load was investigated. Schuller *et al.* [27] evaluate to what extent the charging of EVs can be accommodated using RES in two sociodemographic groups: retired vs. employed people.

The listed works give valuable insights on the benefits of EV flexibility in various aspects including load reduction, environmental benefits and RES integration. However, they characterize the flexibility only in terms of the amount of shifted energy and not the duration. We on the other hand provide a complete quantification of flexibility in terms of not only the deferrable amount but also time of availability and the deferrable duration. Furthermore, detailed analysis of how the flexibility is used is also missing in the literature. We thus present an extensive analysis on how flexibility is exploited (using our proposed measures) to meet two representative objectives: peak reduction and balancing against RES.

Note that this paper is a substantial extension of our work in [28] since we now offer a more extensive analysis of the charging session characteristics and investigate the effect of seasonal changes and weekends on the characteristics of the charging sessions. Additionally, in [28], we quantized flexibility as the maximal load that could be deferred for a specific duration at any time of the day, independent of any DR scheme. In other words, our previous analysis showed the flexibility potential that is available for utilization and not the flexibility that would be utilized to meet various DR objectives. In this paper, we complement our previous flexibility potential analysis and propose measures to quantify the actually exploited flexibility under two DR schemes: load flattening and load balancing.

3.2 Analysis of EV Charging Behavior

In this section, we address the first two research questions raised in Section 3.1.1: Do EV owners have specific habits in terms of charging their cars? Are the characteristics of the charging sessions (e.g., arrival, sojourn and idle times) sensitive to seasonal changes and weekdays? Our analysis is based on a reasonably large real-world dataset which is explained next.

3.2.1 Dataset Description

The data for our analysis was collected by ElaadNL[2] between 2011 to 2015 from public charging infrastructure deployed throughout the Netherlands. The dataset has more than 1.5M charging sessions characterized by arrival time, departure time, charging duration, and total power consumption. The EVs in this dataset are privately owned cars and thus comprise a mixture of various and a priori unknown types, without further information on their driving behavior. For our analysis, we took the subset of sessions from 22nd Dec 2014 to 21st Dec 2015 (i.e., 387,524 sessions) to ensure the observed charging behavior is not dominated by (potentially distinctive) behavior of novice users, since by that date the system had been deployed a few years already. Moreover, in this period there were not substantial extensions of the charging infrastructure: the number of deployed charging stations remained almost constant through 2015. The selected horizon effectively covers the four seasons and hence facilitates analysis of seasonal influences.

3.2.2 Clustering of Charging Session Times

The first question we address is: are there any typical behaviors in terms of arrival and departure times in the dataset? To answer this question, we have plotted the data in 2D space in terms of arrival time vs. departure time as shown in Figure 3.1. We then adopted DBSCAN [29] clustering to cluster the data in that 2D space.

DBSCAN clustering is a density based clustering algorithm and we deemed it to be more suitable than other clustering algorithms (e.g., k-means and G-means [30]) for two reasons: (1) unlike k-means, DBSCAN does not require to a priori specify the number of clusters to distinguish and (2) DBSCAN is able to identify arbitrary shaped clusters without prior assumptions about the underlying distribution of data in each cluster, as opposed to the normal distribution assumed by the G-means clustering algorithm. One of the disadvantages of DBSCAN is its sensitivity to the parameters of the algorithm (i.e., ε, which specifies how close points should be to each other to be considered part of the same cluster; and $minPts$, which specifies the minimum number of points required to form a dense region). The values of ε and $minPts$ are empirically obtained from the data. To examine the sensitivity of DBSCAN to these parameter values, we considered clustering the data separately for each month. We were able to identify 3 behavioral clusters in each month using similar values of ε and $minPts$ (i.e., $\varepsilon = 0.4$ and

[2] ElaadNL is the knowledge and innovation center in the field of charging infrastructure in The Netherlands, providing coordination for the connections of public charging stations to the electricity grid on behalf of 6 participating distribution system operators (DSOs). It also performs technical tests of charging infrastructure, researches and tests smart charging possibilities of EVs, and develops communication protocols for managing EV charging. The EV charging session data is available upon request for non-commercial research purposes, subject to signing an agreement. For more information, please contact Chris Develder (email: chris.develder@ugent.be)

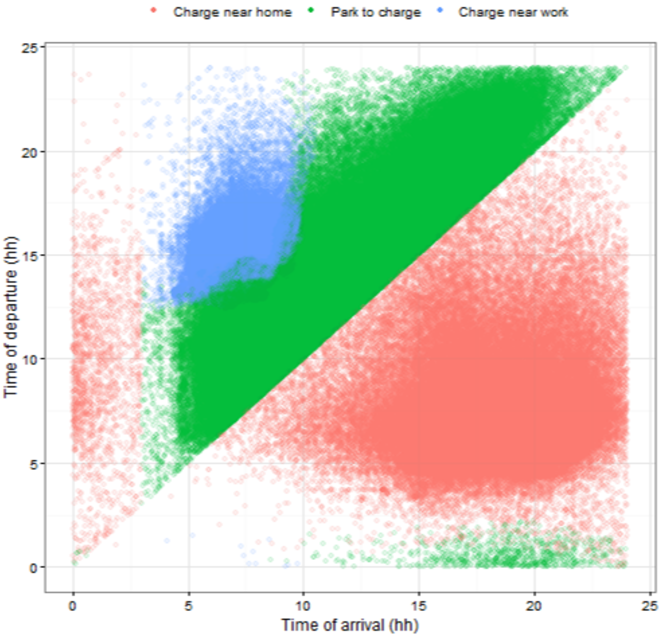


Figure 3.1: behavioral clusters of sessions in terms of EV arrival and departure times. Both X- and Y-axis denote time-of-day (i.e., we report times as $t \bmod 24\,\text{h}$): points below the X=Y diagonal have departures on the day after the arrival or later. (Note that also some sessions plotted above the diagonal actually have departures $\geq 24\,\text{h}$ after arrival)

$minPts = 90$).

Figure 3.1 shows the resulting behavioral clusters for the entire dataset. We named the clusters according to our interpretation of the observed behavior: charge near home, charge near work and park to charge clusters. The *charge near home* cluster (27.84% of the total data) has arrivals in the afternoon/evening with departures mostly in the morning of the next/subsequent days. We hypothesize these are mostly people that live nearby the public charging station and park their car until they leave for work in the morning. Hence, the charging usually occurs at night for the sessions in this cluster. The *charge near work* cluster (9.3% of the total data), which accounts for the smallest share of the data, is characterized by arrivals in the morning and departures in the evening. We assume these are people who either work near a public charging station or take their car to the station on their way to work (e.g., as a part of their commute, near a train station) and leave their car there while at work. Hence, this cluster has significantly smaller fraction of arrivals in weekends compared to the other two clusters (see Table 3.1 for fraction of weekend arrivals in each cluster). This type of behavior is absent in the datasets collected from residential charging (e.g., iMove [28]). The *park to charge* cluster (62.86% of the total data) is the largest cluster and has arrivals/departures scattered throughout the day with sojourns that last not much longer than the time required to charge the battery. We hypothesize these are people that park specifically for the sake of charging the EV battery.

The aforementioned behavioral clusters provoke questions pertaining to what factors exactly distinguish them from each other, which we analyze next.

3.2.3 Analysis of Behavioral Clusters: Weekdays and Seasonal Impacts

In this section, we further analyze the sessions within each of the behavioral clusters in terms of their arrival time, sojourn time (i.e., how long the car is connected at the charging station) and idle time (i.e., the time between the completion of the charging and departure of the car). More formally, we define:

$$\begin{aligned}
\textit{Sojourn time} \triangleq \delta_{\text{sojourn}} &= t_{\text{depart}} - t_{\text{arrive}}, && (3.1)\\
\textit{Charging time} \triangleq \delta_{\text{charging}} &= t_{\text{end charging}} - t_{\text{start charging}}, && (3.2)\\
\textit{Idle time} \triangleq \delta_{\text{idle}} &= \delta_{\text{sojourn}} - \delta_{\text{charging}}. && (3.3)
\end{aligned}$$

We also investigate the impact of weekends and seasonal changes on the aforementioned properties[3].

3.2.3.1 Analysis of Arrival Times

Figure 3.2 shows the violin and box plots of arrival times for the behavioral clusters over weekends and weekdays in each season. In general, weekends and seasons impact the shape of the distributions. Seasonal changes usually shift the arrivals to earlier times in summer and spring for all the clusters. This is possibly due to the earlier sunrise and people's preference to start their days earlier in summer and spring. Arrivals are also earlier on weekdays than in weekends. More details about the weekend and seasonal impacts on the arrival times of the cars in each behavioral cluster are listed below.

For sessions in the *charge near work cluster*, the distribution of arrival times are unimodal and right-skewed during the weekends and multi-modal in the weekdays. Arrivals on weekdays are approximately 1 hour earlier than during weekends in all seasons. Additionally, the interquartile range is slightly longer in weekends compared to weekdays. The longest interquartile range is observed for spring weekends. Across seasons, in summer and spring arrivals are earlier by around 1 hour.

[3] Note that ElaadNL is using flat rates and its impact was not significantly visible in the dataset.

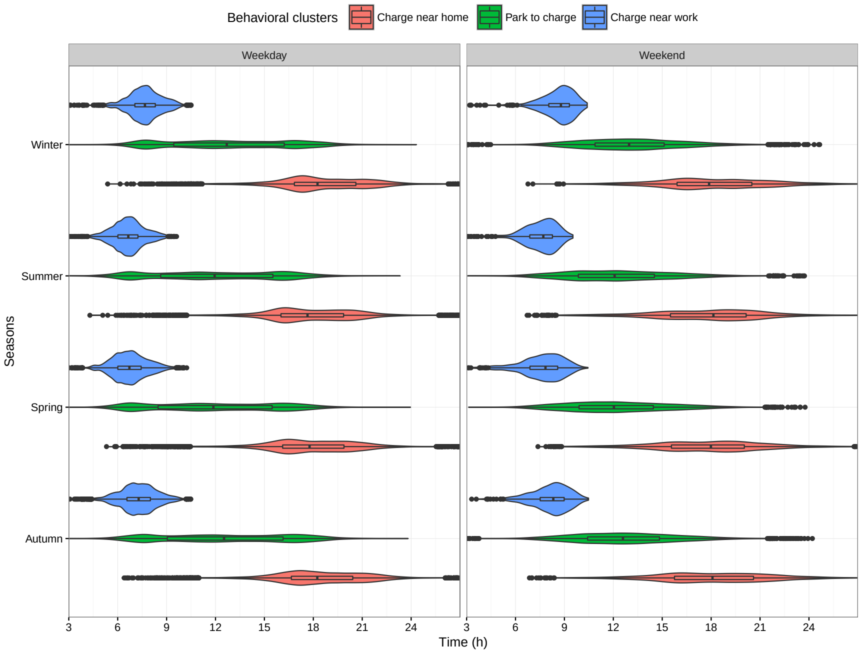


Figure 3.2: Violin and box plots of time of arrivals for the behavioral clusters over weekends and weekdays in each season (Note that the reference is changed from midnight to 3 am (2.30 am to 3.30 am is the interval with least number of arrivals) to acount for the fact that the activities right after the midnight are continuation of the late night activities)

For sessions in the *park to charge cluster*, the distribution of arrival times has a single mode and peaks around noon during the weekends, whereas on weekdays it is multi-modal with 3 peaks (in morning, noon and evening). The arrivals in this cluster are scattered throughout the day,resulting in the largest interquartile range amongst the behavioral clusters. The interquartile range is approximately an hour longer on weekdays for all seasons. Across seasons, the arrivals are typically 30 to 45 min earlier in summer and spring compared to autumn and winter.

For sessions in the *charge near home cluster*, the distribution of arrival times are uni-modal and right-skewed with a heavy tail on weekdays in all seasons. During weekends, the distributions are also uni-modal but right-skewed in summer and spring while left-skewed for winter and autumn. This can be explained by people's preferences to stay out longer during weekends to enjoy longer daylight and warmer weather in summer and spring. The interquartile ranges are longer in weekends in all seasons. Seasonal changes do not significantly affect the interquartile ranges. Finally, arrivals are typically earlier during the weekdays of summer and spring but similar in all seasons during the weekends.

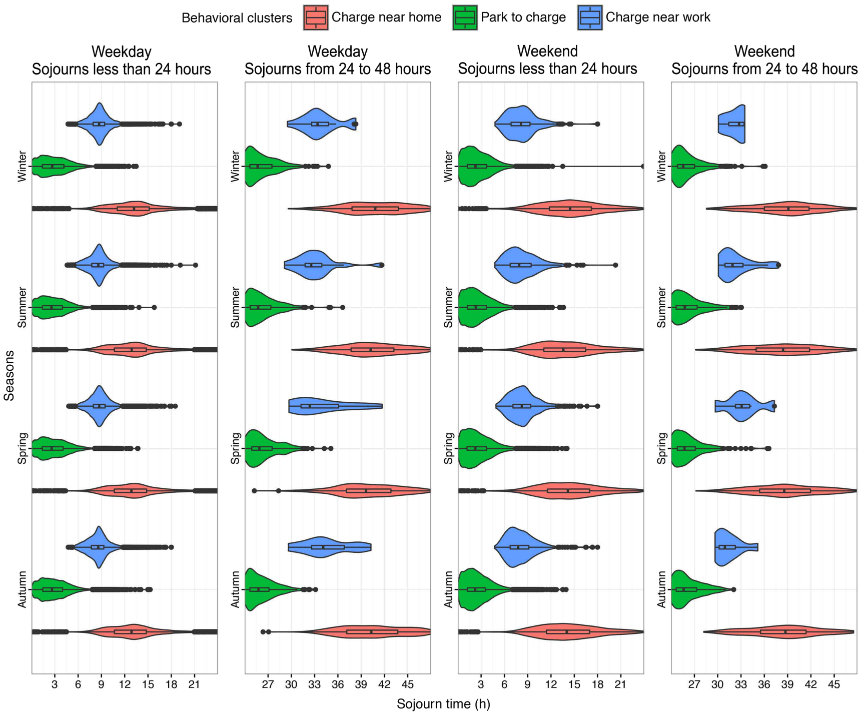


Figure 3.3: Violin and box plots of sojourn times for the behavioral clusters over weekends and weekdays in each season

Table 3.1: Summary of cluster and sub-cluster fractions and average sojourn and idle times

Cluster	Weekend arrival fraction	Sub-cluster departures	Sub-cluster fraction	Mean sojourn time	Mean idle time
Park to charge (62.86%)	28.24%	in 1st 24 h	98.9%	2 h 28 min	48 min
		in 2nd 24 h	0.85%	26 h 18 min	22 h 48 min
		in 3rd 24 h	0.21%	66 h 18 min	62 h 42 min
		in 4th 24 h and later	0.11%	105 h 24 min	101 h 42 min
Charge near home (27.84%)	23.24%	in 1st 24 h	95.09%	13 h 24 min	10 h
		in 2nd 24 h	3.44%	39 h 36 min	36 h 12 min
		in 3rd 24 h	0.9%	63 h 48 min	60 h 6 min
		in 4th 24 h and later	0.57%	113 h 30 min	109 h 54 min
Charge near work (9.3%)	6.33%	in 1st 24 h	99.51%	8 h 42 min	5 h 30 min
		in 2nd 24 h	0.33%	33 h 24 min	29 h 12 min
		in 3rd 24 h	0.4%	38 h 12 min	34 h 24 min
		in 4th 24 h and later	0.09%	119 h 42 min	115 h 18 min

3.2.3.2 Analysis of Sojourn Times

Looking at each individual behavioral cluster, we observe that a minority of sessions have sojourn times of more than 24 h (see Table 3.1). We also find that for these clusters, the sojourn time distribution is multi-modal, where the modes correspond to subsequent days and are well separated. We thus partition the data into sub-clusters based on the departure time (i.e., depending on whether it is within the first, second, etc. period of 24 h following the arrival). Figure 3.3 shows the violin and box plots of sojourn times for the behavioral sub-clusters over weekends and weekdays in each season. We only show the first 2 sub-clusters (i.e., sessions with departures within first and second 24 h from their arrivals) since the later sub-clusters constitute less than 1% of the data (see Table 3.1). In general, seasonal changes have minor effects on sojourn times in the behavioral clusters, but weekends impact the sojourn times more significantly. Further details about the weekend and seasonal impacts on the sojourn times in each behavioral cluster are listed below. Note that our explanations here are based on the 1st sub-clusters (i.e., departures within 1st 24 h) since in the second sub-clusters (i.e., departures within second 24 h), distributions of the sojourn times have similar characteristics as ones in the first sub-clusters. One interesting characteristic is the approximate shift of 24 h in the average sojourn times in the second sub-clusters from the average values of the first sub-clusters (as seen from Table 3.1)

For the sessions in the *charge near work cluster*, the distribution of sojourn times are right-skewed in weekends and symmetrical or left-skewed during weekdays. This implies that typically the sessions have shorter sojourn times in weekends (average sojourn times are 8 h 18 min and 8 h 48 min for arrivals in weekend and weekdays respectively). Additionally, the interquartile ranges are smaller in the weekdays, implying a more predictable sojourn time. The largest interquartile range is in summer weekends.

Sessions in the *park to charge cluster* typically have smaller sojourn times than sessions in other clusters. As shown in Figure 3.3, the distributions are left skewed for both weekend and weekdays, with slightly larger interquartile ranges during weekdays. This implies that sojourn times are typically shorter in weekends (average sojourn times are 2 h 36 min and 2 h 48 min for arrivals in weekend and weekdays respectively). The seasonal changes do not impact the distributions significantly in this cluster.

Sessions in the *charge near home clusters* have considerably larger sojourn times than the sessions in other clusters. The distribution of the sojourn times are symmetrical for both weekends and weekdays, with larger interquartile ranges in weekends. Unlike the other clusters, the charge near home sessions have longer sojourns during weekends (the average sojourn times are 13 h 6 min and 14 h 18 min for arrivals in weekends and weekdays respectively). This is mainly because they are night time charging sessions, and people leave home later in the morning in the

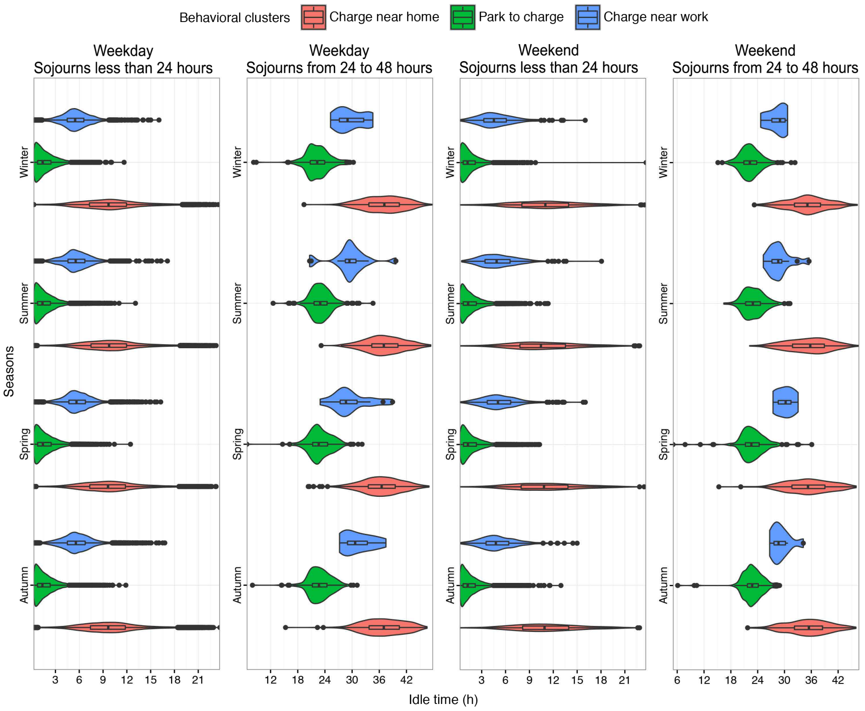


Figure 3.4: Violin and box plots of Idle times for the behavioral clusters over weekends and weekdays in each season

weekend.

3.2.3.3 Analysis of Idle Times

We have used the same sub-clustering approach to present the distribution of the Idle times in each behavioral cluster. Additionally, to improve the readability of the plots in Figure 3.4, we have removed sessions with short idle times (i.e., less than 15 min). This amounts to 43.08% and 33.58% of the data in weekends and weekdays respectively.[4] Note that the majority of the removed short idle times belong to the park to charge cluster. An overall view of Figure 3.4 suggests that seasonal changes do not influence the distribution of idle times significantly, unlike weekend impacts, which are more apparent. Further details about the impact of the weekends on the distribution of the idle times are listed below.

The sessions in the *charge near work cluster* typically have 4 to 7 h of idle time in the first 24 h sub-cluster and 27 to 28 h of idle time in the second 24 h sub-cluster during the weekends. On weekdays, idle times are typically around 30 min

[4] However, the average values in Table 3.1 do include the short idle times in their calculation.

Table 3.2: Nomenclature

Input parameters	
N	Total number of cars in the optimization window
H	The length of the optimization window (the number of 15 min time slots)
γ_{nh}	Maximum allowable energy consumption for car n in slot h
E_n	Total energy to be scheduled for car n
P_n^{avg}	Average power consumption of car n
α_n	Arrival slot of car n
β_n	Departure slot of car n
β_h	Penalty for delaying the charging by h
L_h^{RG}	Total energy from renewable generation in slot h
Decision variables	
x_{nh}	Energy scheduled to charge car n in slot h
L_h	Total energy consumed in slot h

longer than in weekends. On average (taking into account the sessions with short idle times), this cluster has 5 h 30 min of idle time in the first 24 h sub-cluster and 29 h 48 min of idle time in the second 24 h sub-cluster.

The sessions in the *park to charge cluster* typically have the shortest idle times, which suggests that the cars are usually parked with the motive of leaving as soon as the charging completes. The distribution of idle times are right skewed even after the removal of short idle times for the first sub-cluster over both weekends and weekdays. In the second sub-cluster, it looks symmetrical. On average, the park to charge sessions have 42 min of idle time in the first sub-cluster and 22 h 48 min of idle time in the second sub-cluster.

The *charge near home* sessions offer longer idle times (i.e., 10 h in the first and 36 h in the second sub-cluster) than the other clusters. The distributions of the idle times are symmetrical in all the sub-clusters and during both weekends and weekdays. The interquartile ranges span from 8 h to 14 h in the weekends and from 7 h 30 min to 12 h during the weekdays in first sub-cluster.

3.3 Flexibility Quantification

Our quantitative analysis of flexibility exploitation relies on the aforementioned EV charging data collected by ElaadNL, and renewable generation data obtained from ELIA (Belgium's electricity transmission system operator).[5] The data obtained from ELIA comprises wind and solar energy generation measurements in

[5] http://www.elia.be/en/about-elia

15 min intervals for the region of Flanders in Belgium. We rescaled the renewable energy production data to keep similar monthly wind to solar ratios as of the ones in Netherlands.[6] Additionally, we further scaled the data to ensure the total yearly generation is similar to the total yearly demand of all the EV sessions considered in our study. We provide an assessment of flexibility exploitation in coordinated charging for two scenarios: (i) load flattening and (ii) load balancing against renewable production. As a reference, we take uncoordinated charging and refer to it as a business as usual scenario without flexibility exploitation.

Each time slot is characterized by a 15 min interval $h \in \mathbf{H} = \{1, 2, ..., H\}$ and the EVs are denoted as $n \in \mathbf{N} = \{1, 2, ..., N\}$. Table 3.2 summarizes all the model parameters and the decision variables.

3.3.1 Uncoordinated Charging: Business as Usual

In the business as usual (BAU) scenario, charging starts immediately upon arrival. In the ElaadNL dataset, vehicles are charged according to this BAU scenario and the charging time as well as the total energy consumption is reported for each session. The load in each time slot (i.e., of 15 min duration) is hence calculated as $P_h = \Delta t \cdot E_n / (t_{BAU} - t_{arrive})$, where t_{BAU} is the time of the completion of charging in the BAU regime and Δt is the duration (in hours) of each slot (i.e., $\Delta t = 0.25\,\text{h}$) in our settings.

3.3.2 Coordinated Charging: Load Flattening and Load Balancing

In the coordinated charging scenario, charging decisions are optimized by an aggregator to meet a predefined objective function. We formulate such a problem as a quadratic optimization (i.e., a quadratic objective function subject to linear constraints). To make the problem scalable and solvable in close to real-time, we define an optimization window of length $H = 96$ time slots (i.e., 24 h) which starts at the present time slot (denoted as "Now") and moves one slot in each iteration. We thus consider a receding horizon control approach, where we repeatedly solve the optimization problem to find the decision variables covering the window ("Now","Now+H").

For load flattening, the objective function is defined as:

$$\underset{\mathbf{L},\mathbf{X}}{\text{minimize}} \qquad M \sum_{h=1}^{H} L_h^2 + \sum_{n=1}^{N} \sum_{h=1}^{H} \beta_h x_{nh} \tag{3.4}$$

[6] See http://en-tran-ce.org/ for yearly reports of renewable generations in Netherlands.

The first term in (3.4) is a convex quadratic cost function and reflects the total load that needs to be minimized in the optimization window. We define a second term in (3.4) as a secondary objective which penalizes charging at later slots. This ensures that charging at earlier slots is preferred when permutations of charging decisions across different slots have the same cost. Note that we multiply the first term in (3.4) by M, a large constant[7], to have the first term dominate the second term in the objective function.

For load balancing, the objective function is defined as:

$$\underset{\mathbf{L},\mathbf{X}}{\text{minimize}} \qquad M\sum_{h=1}^{H}(L_h - L_h^{RG})^2 + \sum_{n=1}^{N}\sum_{h=1}^{H}\beta_h x_{nh} \tag{3.5}$$

The first term in (3.5) models the imbalance using a convex quadratic function. Note that (similar to [31]) we account for negative imbalance to be as bad as positive imbalance. Similar to (3.4), the secondary objective function in (3.5) ensures earlier charging when charging at various slots has the same cost.

Both of the objective functions are subject to the following linear constraints:

$$L_h = \sum_{n=1}^{N} x_{nh} \qquad \forall h \in \mathbf{H} \tag{3.6}$$

$$E_n - E_a \leq \sum_{h=1}^{H} x_{nh} \leq E_n \qquad \forall n \in \mathbf{N} \tag{3.7}$$

$$0 \leq x_{nh} \leq \gamma_{nh} \qquad \forall n \in \mathbf{N}, h \in \mathbf{H}_n \tag{3.8}$$

$$x_{nh} = 0 \qquad \forall n \in \mathbf{N}, h \in \mathbf{H} \setminus \mathbf{H}_n \tag{3.9}$$

where,

$$\mathbf{H}_n = \begin{cases} \{\alpha_n, ..., \beta_n\} & \beta_n \leq H \\ \{\alpha_n, ..., H\} & \beta_n > H \end{cases}$$

and

$$E_a = \begin{cases} P_n^{avg} \cdot (\beta_n - H) & \beta_n > H \\ 0 & \text{otherwise} \end{cases}$$

Constraint (3.6) ensures that the total load consumed in slot h is equal to the summation of the loads from all the cars scheduled to charge in slot h of the optimization window. Constraint (3.7) ensures that the charging demand (i.e., E_n) is fulfilled within the car's sojourn time. When a car departs within the optimization window, (3.7) becomes an equality constraint (i.e., equals E_n). Constraint (3.8) limits the energy consumption in each slot to the car's allowable consumption level and constraint (3.9) prohibits any charging outside the sojourn time.

[7] see appendix for the lower bound of M

3.3.3 Measures for Quantification of Flexibility Utilization

As outlined in Subsection 3.1.2, the demand response potential of EVs has already been studied to some extent, but how exactly the offered flexibility is exploited in real-world scenarios has not been well clarified in literature. In this section, we address this gap and offer a quantitative analysis of the flexibility exploitation of EVs using various measures. We first define the flexibility using 3 factors [32]: (1) the *amount* of deferrable energy (i.e., the amount of energy that can be delayed without jeopardizing customer convenience or quality of the task to be fulfilled), (2) the *time* of availability (i.e., the time at which a customer offers the flexibility for exploitation), and (3) the *deadline/permissible duration* to exploit the offered flexibility (i.e., the maximum allowable delay for the energy consumption).

We define the following measures to adequately quantize the EV flexibility exploitation:

1. *Eflex* (flexibility utilization in terms of Energy): fraction of the maximum energy that could be consumed beyond t_{BAU}. More formally,

$$\mathit{Eflex} = \frac{\text{Energy consumed beyond } t_{BAU}}{\text{Maximum possible energy consumption beyond } t_{BAU}} \tag{3.10}$$

2. *Tflex* (flexibility utilization in terms of duration): fraction of the maximum delay beyond t_{BAU}. More formally,

$$\mathit{Tflex} = \frac{t_{coordinated} - t_{BAU}}{t_{depart} - t_{BAU}} \tag{3.11}$$

where $t_{coordinated}$ refers to the time of completion of charging in the coordinated charging regime.

The combination of *Eflex* and *Tflex* values quantizes the fraction of flexibility (in terms of time and amount) that was utilized for each charging session. For example, when $\mathit{Tflex} = \mathit{Eflex} = 1$, the energy consumption is deferred as much as possible (i.e., $t_{coordinated} = t_{depart}$) and the consumption beyond t_{BAU} is at its maximal level. Another interpretation is that $1 - \mathit{Eflex}$ is the fraction of state-of-charge (SoC) at t_{BAU} that has been realized in the flex scenario; for example, if $\mathit{Eflex} = 0.25$, it means that at t_{BAU}, we have $1 - 0.25 = 75\%$ of the desired SoC. Note that *Eflex* and *Tflex* definitions are only valid for charging sessions with non-zero flexibility.

Although the aforementioned measures indicate how much of the offered flexibility is effectively utilized in each charging session, they do not provide information about the volume and the precise time shift of the deferred energy. Indeed, we believe it is interesting to know what portion of energy use is shifted to what time exactly. To quantitatively evaluate this, we define the *shift profile* of a charging algorithm 2: Calculate shift profile for a charging session

```
Input  : $\mathbf{L}_{BAU}$ (with size($\mathbf{L}_{BAU}$) $= S$), a vector denoting energy consumption by
         the EV in each slot in the BAU scenario
         $\mathbf{L}_{coordinated}$ (with $L_{coordinated}| = M$), a vector denoting energy
         consumption in each slot in the coordinated charging scenario
Output: The shift profile ($\mathbf{L}_{shift}$)
7  Define $\mathbf{L}_{scheduled}$ with size($\mathbf{L}_{scheduled}$) $=$ $M$ and initialize it with zeros
      /* $\mathbf{L}_{scheduled}(s)$ is the energy scheduled from previous slots
      to $s$ */
8  $s' = 1$ /* $s'$ is used for indexing to save the caclulations in
      $\mathbf{L}_{shift}$ */
9  foreach $s = 1, \ldots, S$ do
10     $shift = \mathbf{L}_{BAU}(s) - \mathbf{L}_{coordinated}(s) + \mathbf{L}_{scheduled}(s)$ /* the amount of energy
          that needs to be shifted away from $s$ */
11     $m = 1$ while $shift \neq 0$ do
12         $capacity = \mathbf{L}_{coordinated}(s+m) - \mathbf{L}_{scheduled}(s+m)$ /* $\mathbf{L}_{coordinated}(s) \geq$
              $\mathbf{L}_{scheduled}(s)$ since this calculation is done after the
              optimization and $\mathbf{L}_{coordinated}(s)$ is the finalized load
              to be consumed in slot $s$. */
13         $actual\ shift = min(shift, capacity)$ $\mathbf{L}_{shift}(s') = (s, actual\ shift, s+m)$
            $s' = s'+1$ $\mathbf{L}_{scheduled}(s+m)$ += $actual\ shift$  $shift = shift - actual\ shift$
14         $m = m + 1$
15 return $\mathbf{L}_{shift}$
```

session: the *shift profile* indicates how the energy is shifted from the BAU scenario to obtain the load pattern in the coordinated charging regime. In other words, it shows how much energy is shifted away from a particular slot and which slot it is scheduled to. We now explain how we calculate this *shift profile*, as outlined in Algorithm 2.

Given the $\mathbf{L}_{BAU}$ and $\mathbf{L}_{coordinated}$ vectors, respectively denoting the BAU and the coordinated energy consumption values in each slot, Algorithm 2 returns a $\mathbf{L}_{shift}$ list as its output. Each element of $\mathbf{L}_{shift}$ is a triple, depicting how much energy was shifted away from a particular slot and which slot it was shifted to (e.g., if 5 kWh of energy is shifted from slot 1 to slot 3, then the triple will have the following form: $(s_{from}, E_{shifted}, s_{to}) = (1, 5, 3)$).[8] The algorithm starts by initializing $\mathbf{L}_{scheduled}$, a vector that keeps track of the amount of energy scheduled in a particular slot from the other slots (Line 7). For each slot s, starting with the first one, the amount of energy we need to shift away from it (i.e., *shift*) is calculated in Line 10. Note that to calculate the *shift* in each slot, we take the difference in energy consumption in

[8] Note that there could be several feasible shift profiles (e.g., {(1, 1, 3)} vs. {(1, 1, 2), (2, 1, 3)}) but here we calculate the one with minimal $s_{to} - s_{from}$.

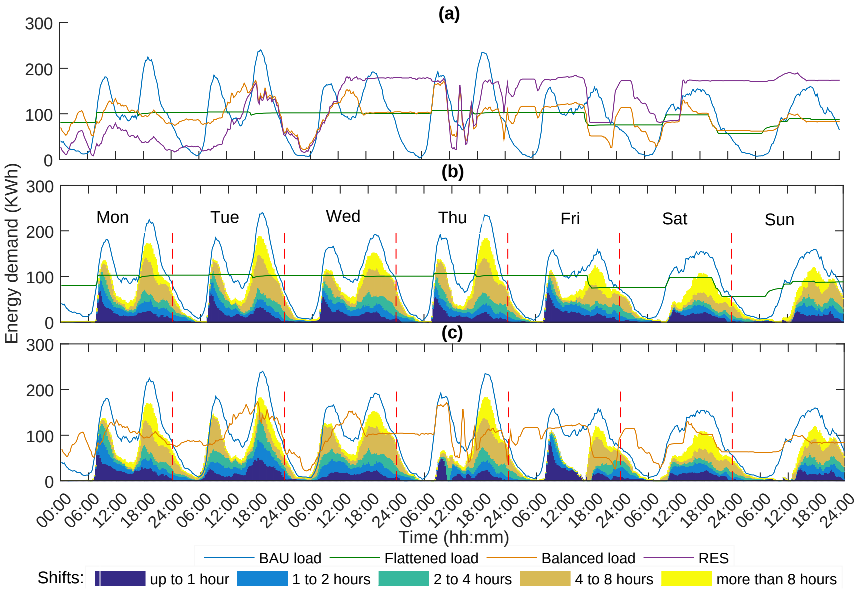


Figure 3.5: (a) Load and renewable generation patterns from 5th to 11th Jan, (b) Amount of energy that is shifted away from each slot(for arrivals from 5th to 11th Jan 2015) in load flattening scenario and (c) Amount of energy that is shifted away from each slot (for arrivals from 5th to 11th Jan 2015) in load balancing scenario.

the BAU and the coordinated charging scenario. Additionally, since any energy scheduled to be consumed in a slot also contributes to the delay of the energy consumption from that slot, we add the $\mathbf{L}_{scheduled}$ to the subtraction term. In the while loop, the *shift* is allocated to the subsequent slots following s, based on their available capacity. The amount of the allocated energy and the slot number is saved in $\mathbf{L}_{shift}$ (Line 13) and $\mathbf{L}_{scheduled}$ is updated accordingly (Line 13).

3.3.4 Evaluation of Flexibility Exploitation

In this section, we evaluate the flexibility exploitation using the measures and the algorithm proposed in the previous subsection. We implemented the optimization problem using MOSEK[9], in a MATLAB runtime environment.

Figure 3.5 shows how much energy (kWh) has been pushed away, and for how long, from BAU consumption, assuming 15 min long time slots in the optimization of the coordinated charging scenarios (i.e., load flattening and load balancing). A week long duration is selected for demonstration in Figure 3.5. Figure 3.5a shows the energy consumption patterns (in the BAU, load flattening and load balancing

[9] MOSEK is a software package for solving mathematical optimization problems, see https://www.mosek.com/.

scenarios) and the scaled renewable generation in each slot of the selected one week long time period. As seen from the figure, the BAU energy consumption patterns are multi-modal with distinct morning (around 9 am) and evening (around 8 pm) peaks on weekdays. During the weekends, the peak-to-average ratio is lower than on weekdays and energy consumption patterns have a small peak around noon and a larger peak around 6 pm.

In the **load flattening** scenario (i.e., Figure 3.5b), we observe the following:

1. The flexibility utilization is influenced by the BAU energy consumption patterns as well as the car arrival times (note that the arrival times and the BAU energy consumption patterns are also highly correlated.)

2. *During weekdays*: The load is typically shifted away from the morning peak (around 8-10 am) towards the afternoon valley (around 12-2 pm). Since the afternoon valley is not long away from the morning peak, the duration of the shift is typically lower compared to the shift from the evening peak to the midnight valley. Hence, we see more shifts of “up to 1 hour” long and less shifting of beyond “4 hours” from the morning peak. On the other hand, the shifts from the evening peaks are longer to fill up the night valley, which is deeper and further away.

3. *During weekends*: The shifts from the evening peak to the night valleys are longer in weekends (typically more than 8 hours from the Saturday evening peak and more than 4 hours from the Sunday evening peak). The longer shifts from Saturday peaks are due to the wider and deeper valley between Saturday and Sunday peaks.

In the **load balancing** scenario, clearly the flexibility utilization is not only influenced by BAU energy consumption pattern and the car arrival times, but also by the renewable generation patterns. The flexibility exploitation for load balancing is depicted in Figure 3.5c with the following key observations:

1. Although the flexibility utilization is not as consistent as for the load flattening scenario, still, longer shifts are observed in the evening peaks on weekdays. Additionally, there are still longer shifts from the Saturday peaks compared to the shifts from the Sunday peaks.

2. In general, longer shifts from the evening peaks are observed when there is substantial renewable generation in the night valleys.

The observations based on Figure 3.5 give insight in the motivation for utilization of the flexibility and, hence, how much energy is required to be shifted and for how long. This is particularly useful for price-based or incentive-based demand response programs aiming to influence the offered flexibility at various hours of

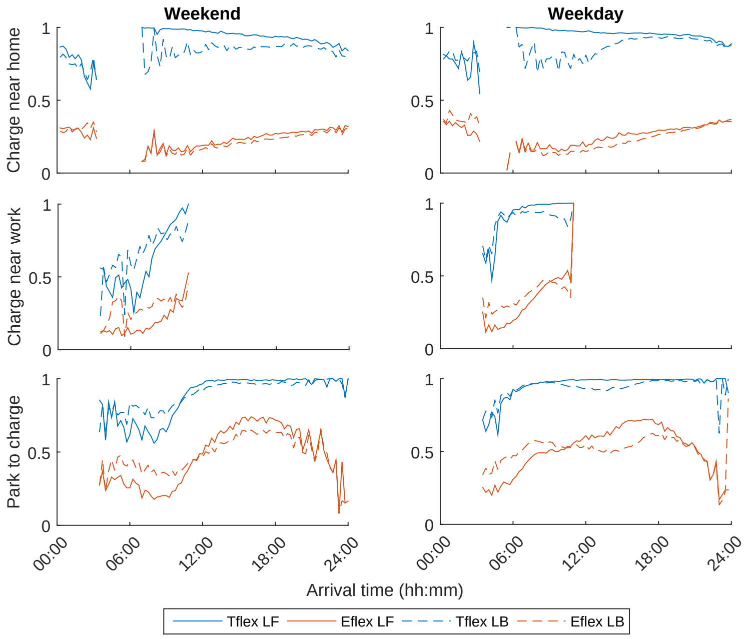


Figure 3.6: Average *Tflex* and *Eflex* values for each 15 min long timeslot in a day (LB: load balancing, LF:load flattening)

the day accordingly (using a relevant price or incentives). For example, the longer shifts from morning peak are not as frequent as the ones from the evening peak and hence, a lower incentive could be given for longer sojourn time of the cars arriving before the morning peak.

In addition, it is also useful to know how much of the offered flexibility is utilized throughout the day. To quantize the degree of flexibility utilization, we use the *Eflex* and *Tflex* measures. Figure 3.6 shows, for a given time slot, the average *Tflex* and *Eflex* values for the sessions with arrivals in that slot (note that these sessions may extend until much later slots). The values are depicted for each behavioral cluster during weekdays vs. weekends. The empty sections in the plots indicate there were either no arrivals occurred, or the arrivals had zero idle times at these times of day. We list our observations for the *Eflex* and *Tflex* in the load flattening scenario, which essentially also apply qualitatively for the load balancing case.

For *Tflex*: In general, *Tflex* close to 1 means that charging lasts almost until the end of the sojourn. Yet, this does not mean that all charging is delayed (see the *Eflex* which is reasonably low, meaning that the SoC at t_{BAU} is pretty high). We observe lower *Tflex* for arrivals at night and in the early morning (i.e., 0-6 am).

The reason is that the sessions with arrival times in those slots are responsible for the bulk of the load at those times, which is low compared to other slots, so there is a lower motivation to push their charging away and make use of flexibility. Any arrivals in the subsequent slots have their load shifted away from the peaks, hence, the *Tflex* value increases and approaches 1. *Tflex* starts to decrease again for the arrivals near midnight.

For *Eflex*: similar to *Tflex*, lower *Eflex* is observed for arrivals at night and early morning (i.e., 0-6 am) since the bulk of the load at those times is low and hence, there is little need for deferring the consumption. The *Eflex* in the late morning (9-11 am) is lower than in the afternoon/evening. Note that the arrivals in the late morning are usually used to fill the afternoon valley, but the amount of energy pushed into afternoon valley from morning peaks is lower compared to the amount of energy pushed into night valley (the night valley is deeper and requires more load to be filled). Additionally, the arrivals in the late morning are typically from the *park to charge* or the *charge near work* clusters: since their sojourn does not overlap with the night valley, their load cannot be used to fill the night valley. Another interesting observation is the bell shape of *Eflex* after 12 pm in the *park to charge* cluster for both weekends and weekdays, which peaks around 4 pm and 6 pm respectively. Note that since the sessions in this cluster have very small idle times, a larger portion of their energy consumption is deferred, but for shorter duration, to flatten the load. In the *charge near home* cluster, we see a rather linear increase in *Eflex*. The sessions in this cluster offer much longer idle times compared to the *park to charge* cluster. By observing the SoC status of the sessions in this cluster, we find that for the sessions whose sojourns overlap with the evening peak, their charging usually stops during the peak hours and resumes in the night valley. That is the main reason for *Tflex* close to one but rather small *Eflex* for sessions with arrivals in the afternoon and evening.

3.4 Summary and Conclusion

Motivated by the lack of research in characterizing the flexibility stemming from EV charging sessions, in this paper we took the first step to (1) offer an in-depth analysis of the flexibility characteristics of a nearly 390k EV charging sessions and (2) propose flexibility measures to quantify its exploitation in two scenarios, load flattening and load balancing. Our contributions in this paper pave the way to more realistic evaluation and development of DR algorithms, which aim to not only exploit the flexibility but also to influence it more efficiently (through price-based or incentive-based schemes).

To fulfill our first objective (i.e., analysis of flexibility characteristics), we clustered the EV data in 2D space in terms of arrival and departure times using the DBSCAN algorithm. As such, we identified three behavioral clusters: charge near home, charge near work, and park to charge clusters. We then used box and violin plots to further analyze the characteristics of the charging sessions within each cluster and highlighted the differences among the clusters over weekends and weekdays in each season. A summery of our observations is listed here:

1. The three behavioral clusters differ substantially in their arrival times, sojourn times and the idle times. The park to charge cluster (which is the largest in terms of number of sessions, 62.86% of all sessions) has arrivals scattered throughout the day and the sessions in this cluster are characterized by very short idle times (averaging 48 min). The charge near work cluster (27.84% of all sessions) has predictable arrival times (around 6-9 am) and their sojourn times are typically less than 9 hours (with average idle time of 5 h 30 min), hence, their charging usually takes place throughout the day. Finally, the sessions in charge near home cluster (9.3% of all sessions), with arrivals typically in the evening until midnight, offer the longest idle times among the clusters (10 h on average). The charging for these sessions usually occurs at night.

2. Weekends and weekdays as well as seasonal changes impact the arrival times in all three clusters. In general, the arrival times are earlier in summer and spring in all the clusters. The arrivals are also earlier on weekdays compared to weekends. However, seasons have no substantial impact on the sojourn and idle times. Sessions in park to charge and charge near work clusters have shorter sojourn and idle times in the weekends whereas the sessions in the charge near home clusters have longer sojourn and idle times in the weekends compared to weekdays.

To fulfill our second objective (i.e., quantification of flexibility exploitation), we proposed two flexibility measures to quantify the percentage of the flexibility utilization and an algorithm to determine the amount and duration of the shifted energy. A summary of our analysis using the algorithm and the measures is as follows.

1. The flexibility exploitation is greatly influenced by the uncontrolled business as usual (BAU) load patterns, the distribution of arrival times, and the renewable energy generation patterns. The main motivation for exploitation of the flexibility in both load flattening and load balancing is to fill the valleys of the BAU load pattern. Hence, longer shifts are observed from the evening peaks compared to the morning peaks in the weekdays (since the nighttime valley is larger and deeper). Similarly, longer shifts are seen from Saturday peaks compared to Sunday peaks because the night valley between Saturdays and Sundays is bigger.

2. For arrivals in the afternoon until midnight, flexibility in terms of deferrable time is almost fully exploited to ensure the charging takes place in the nighttime (which corresponds to the lower demand). Yet, this does not imply that all the charging is delayed since the *Eflex* values are reasonably low, meaning that the SoC at the BAU charging completion time (i.e., t_{BAU}) is pretty high. Across the behavioral clusters, the offered flexibility in charge near work cluster is often used to fill the afternoon valley since these sessions are characterized by morning arrivals and their sojourn typically does not cover the night valley. Hence, their exploitation in terms of deferrable time and energy is typically lower compared to the arrivals in the other clusters which are usually in the afternoon. The sessions in the charge near home cluster are the better candidate to fill the night valley.

We conclude that the sessions in the charge near work cluster should be targeted to provide long enough flexibility to fill the afternoon valley. Any longer idle time would not be exploited (unless it is long enough to cover the night valley). The sessions in the charge near home cluster should be targeted to fill the night valley and for arrivals after midnight in this cluster, there is less need for longer idle times. Finally, in the park to charge cluster, it is recommended to target the arrivals in the afternoon to stimulate longer flexibility durations to fill the afternoon valley.

Appendix

In this section we mathematically show the lower bound of constant M in multi objective functions defined in 3.4 and 3.5. The choice of constant M impacts the outcome of the optimization. M should be sufficiently large such that earlier charging as the cost of larger aggregate load (in 3.4) or imbalance (in 3.5) never happens. However, an extremely large M may cause the second terms in 3.4 and 3.5 to have no impact. Hence, it is useful to know the lower bound of M.

In 3.4, M should be sufficinetly large such that the first term is always larger than the second term for the largest delay (i.e., β_H) and maximum energy consumption (i.e., x_{max}), hence

$$M \sum_{h=1}^{H} L_h^2 > \sum_{n=1}^{N} \sum_{h=1}^{H} \beta_H x_{max} \tag{3.12}$$

Using 3.6,

$$M \sum_{h=1}^{H} (\sum_{n=1}^{N} x_{max})^2 > \sum_{n=1}^{N} \sum_{h=1}^{H} \beta_H x_{max} \tag{3.13}$$

Rearranging the above terms yields,

$$M > \frac{NH\beta_H x_{max}}{HN^2 x_{max}^2} = \frac{\beta_H}{N x_{max}}. \tag{3.14}$$

In 3.5, M should be sufficiently large such that the early charging does not occur at the cost of extra imbalance. Let us assume charging at maximum charging rate (i.e., x_{max}). The cost of minimum imbalance (i.e., $M x_{max}^2$) should be larger than the cost of maximum delay (i.e., $\beta_H x_{max}$). Therefore,

$$M x_{max}^2 > \beta_H x_{max} \tag{3.15}$$

$$M > \frac{\beta_H}{x_{max}}. \tag{3.16}$$

References

[1] F. Teng, M. Aunedi, and G. Strbac. *Benefits of flexibility from smart electrified transportation and heating in the future UK electricity system*. Applied Energy, 167:420 – 431, 2016. doi:10.1016/j.apenergy.2015.10.028.

[2] E. S. Rigas, S. D. Ramchurn, and N. Bassiliades. *Managing Electric Vehicles in the Smart Grid Using Artificial Intelligence: A Survey*. IEEE Transactions on Intelligent Transportation Systems, 16(4):1619–1635, Aug 2015. doi:10.1109/TITS.2014.2376873.

[3] Z. Yang, K. Li, and A. Foley. *Computational scheduling methods for integrating plug-in electric vehicles with power systems: A review*. Renewable and Sustainable Energy Reviews, 51:396 – 416, 2015. doi:10.1016/j.rser.2015.06.007.

[4] K. M. Tan, V. K. Ramachandaramurthy, and J. Y. Yong. *Integration of electric vehicles in smart grid: A review on vehicle to grid technologies and optimization techniques*. Renewable and Sustainable Energy Reviews, 53(C):720–732, 2016.

[5] J. Hu, H. Morais, T. Sousa, and M. Lind. *Electric vehicle fleet management in smart grids: A review of services, optimization and control aspects*. Renewable and Sustainable Energy Reviews, 56:1207 – 1226, 2016. doi:10.1016/j.rser.2015.12.014.

[6] I. Lampropoulos, G. M. A. Vanalme, and W. L. Kling. *A methodology for modeling the behavior of electricity prosumers within the smart grid*. In Proc. IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT Europe), pages 1–8, Oct 2010. doi:10.1109/ISGTEUROPE.2010.5638967.

[7] K. Clement-Nyns, E. Haesen, and J. Driesen. *The Impact of Charging Plug-In Hybrid Electric Vehicles on a Residential Distribution Grid*. IEEE Transactions on Power Systems, 25(1):371–380, Feb 2010. doi:10.1109/TPWRS.2009.2036481.

[8] P. Paevere, A. Higgins, Z. Ren, M. Horn, G. Grozev, and C. McNamara. *Spatio-temporal modelling of electric vehicle charging demand and impacts on peak household electrical load*. Sustainability Science, 9(1):61–76, 2014. doi:10.1007/s11625-013-0235-3.

[9] P. Grahn, K. Alvehag, and L. Söder. *PHEV Utilization Model Considering Type-of-Trip and Recharging Flexibility*. IEEE Transactions on Smart Grid, 5(1):139–148, Jan 2014. doi:10.1109/TSG.2013.2279022.

[10] E. Pashajavid and M. Golkar. *Non-Gaussian multivariate modeling of plug-in electric vehicles load demand*. International Journal of Electrical Power and Energy Systems, 61:197 – 207, 2014. doi:10.1016/j.ijepes.2014.03.021.

[11] J. Soares, N. Borges, M. A. F. Ghazvini, Z. Vale, and P. de Moura Oliveira. *Scenario generation for electric vehicles' uncertain behavior in a smart city environment*. Energy, 111:664 – 675, 2016. doi:10.1016/j.energy.2016.06.011.

[12] E. Xydas, C. Marmaras, L. M. Cipcigan, N. Jenkins, S. Carroll, and M. Barker. *A data-driven approach for characterising the charging demand of electric vehicles: A UK case study*. Applied Energy, 162:763 – 771, 2016. doi:10.1016/j.apenergy.2015.10.151.

[13] Y. B. Khoo, C.-H. Wang, P. Paevere, and A. Higgins. *Statistical modeling of Electric Vehicle electricity consumption in the Victorian EV Trial, Australia*. Transportation Research Part D: Transport and Environment, 32:263 – 277, 2014. doi:10.1016/j.trd.2014.08.017.

[14] J. Brady and M. O'Mahony. *Modelling charging profiles of electric vehicles based on real-world electric vehicle charging data*. Sustainable Cities and Society, 26:203 – 216, 2016. doi:10.1016/j.scs.2016.06.014.

[15] J. Quiròs-Tortòs, L. F. Ochoa, S. W. Alnaser, and T. Butler. *Control of EV Charging Points for Thermal and Voltage Management of LV Networks*. IEEE Transactions on Power Systems, 31(4):3028–3039, July 2016. doi:10.1109/TPWRS.2015.2468062.

[16] A. Navarro-Espinosa and L. F. Ochoa. *Probabilistic Impact Assessment of Low Carbon Technologies in LV Distribution Systems*. IEEE Transactions on Power Systems, 31(3):2192–2203, May 2016. doi:10.1109/TPWRS.2015.2448663.

[17] T. Franke and J. F. Krems. *Understanding charging behaviour of electric vehicle users*. Transportation Research Part F: Traffic Psychology and Behaviour, 21:75 – 89, 2013. doi:10.1016/j.trf.2013.09.002.

[18] J. Spoelstra. *Charging behaviour of Dutch EV drivers*. Master's thesis, 2014.

[19] J. Quiròs-Tortòs, L. F. Ochoa, and B. Lees. *A statistical analysis of EV charging behavior in the UK*. In 2015 IEEE PES Innovative Smart Grid Technologies Latin America (ISGT LATAM), pages 445–449, Oct 2015. doi:10.1109/ISGT-LA.2015.7381196.

[20] P. Richardson, M. Moran, J. Taylor, A. Maitra, and A. Keane. *Impact of electric vehicle charging on residential distribution networks: An irish demonstration initiative*. In 22nd International Conference and Exhibition on Electricity Distribution (CIRED 2013), pages 1–4, June 2013. doi:10.1049/cp.2013.0873.

[21] J. Helmus and R. van den Hoed. *Unraveling user type characteristics: Towards a taxonomy for charging infrastructure*. In Proc. 28th Int. Electric Vehicle Symp. and Exhibition (EVS 28), pages 1211–1226, Goang, Korea, 3–6 May 2015.

[22] M. Aunedi, M. Woolf, G. Strbac, O. Babalola, and M. Clark. *Characteristic demand profiles of residential and commercial EV users and opportunities for smart charging*. In Proc. 23rd Int. Conf. Electricity Distribution (CIRED 2015), pages 1–5, Lyon, France, 15–18 Jun. 2015.

[23] E. C. Kara, J. S. Macdonald, D. Black, M. Bérges, G. Hug, and S. Kiliccote. *Estimating the benefits of electric vehicle smart charging at non-residential locations: A data-driven approach*. Applied Energy, 155:515 – 525, 2015. doi:10.1016/j.apenergy.2015.05.072.

[24] I. Pavić, T. Capuder, and I. Kuzle. *Value of flexible electric vehicles in providing spinning reserve services*. Applied Energy, 157:60 – 74, 2015. doi:10.1016/j.apenergy.2015.07.070.

[25] F. Salah, A. Schuller, M. Maurer, and C. Weinhardt. *Pricing of demand flexibility: Exploring the impact of Electric Vehicle customer diversity*. In 2016 13th Int. Conf. on the European Energy Market (EEM), pages 1–5, June 2016. doi:10.1109/EEM.2016.7521202.

[26] M. Khederzadeh. *Inherent Potential of Electrical Vehicles to Flatten the Daily Load Curve in a Microgrid.* In Proc. 23rd Int. Conf. Electricity Distribution (CIRED 2015), pages 1–5, Lyon, France, 15–18 Jun. 2015.

[27] A. Schuller, C. M. Flath, and S. Gottwalt. *Quantifying load flexibility of electric vehicles for renewable energy integration*. Applied Energy, 151:335 – 344, 2015. doi:10.1016/j.apenergy.2015.04.004.

[28] C. Develder, N. Sadeghianpourhamami, M. Strobbe, and N. Refa. *Quantifying flexibility in EV charging as DR potential: Analysis of two real-world data sets*. In Proc. 7th IEEE Int. Conf. Smart Grid Communications (SmartGridComm 2016), pages 1–6, Sydney, Australia, 6–9 Nov. 2016.

[29] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu. *A density-based algorithm for discovering clusters in large spatial databases with noise.* In Kdd, volume 96, pages 226–231, 1996.

[30] G. Hamerly and C. Elkan. *Learning the K in K-Means*. In In Neural Information Processing Systems, page 2003. MIT Press, 2003.

[31] K. Mets, F. De Turck, and C. Develder. *Distributed smart charging of electric vehicles for balancing wind energy*. In Proc. 3rd IEEE Int. Conf. Smart Grid Communications (SmartGridComm 2012), pages 133–138, Tainan City, Taiwan, 5-8 Nov. 2012. doi:10.1109/SmartGridComm.2012.6485972.

[32] N. Sadeghianpourhamami, T. Demeester, D. F. Benoit, M. Strobbe, and C. Develder. *Modeling and analysis of residential flexibility: Timing of white good usage*. Applied Energy, 179:790–805, Oct. 2016. doi:10.1016/j.apenergy.2016.07.012.

4

Bayesian Modeling of Cylindrical Data Using Abe-Ley Mixtures

Chapters 2 and 3 focused on analysis, characterization, and modeling of two sources of flexibility: residential white-good appliance usage flexibility and electric vehicle charging session flexibility. In both cases, the time of the available flexibility (time of appliance configuration and arrival time of an electric vehicle) are of cyclic nature, while the flexibility duration (how long the consumption can be delayed) is a linear quantity. Hence, it is natural to adopt circular distributions to model such data. In Chapter 2 however, we used distributions on a linear scale (Gaussian mixture models) to model the flexibility data. This leads to the following question: can cylindrical distributions model such data better than the linear distribution? To answer this question, we first need to select appropriate mixture models based on a cylindrical distribution. In this chapter, we use a recently proposed distribution (called WeiSSVM or Abe-Ley interchangeably) and develop a Bayesian approach to estimate the parameters of the mixture model. The proposed models are then used in the Appendix A to answer the above-raised question.

⋆ ⋆ ⋆

N. Sadeghianpourhamami, D. F. Benoit, D. Deschrijver, and C. Develder.

Submitted to Applied Mathematical Modeling, Dec. 2017. (Under review, 2$^{\text{nd}}$ round)

Abstract This paper proposes a Metropolis-Hastings algorithm based on Markov chain Monte Carlo sampling, to estimate the parameters of the Abe-Ley distribution, which is a recently proposed Weibull-Sine-Skewed-von Mises mixture model, for bivariate circular-linear data. Current literature estimates the parameters of these mixture models using the expectation-maximization method, but we will show that this exhibits a few shortcomings for the considered mixture model. First, standard expectation-maximization does not guarantee convergence to a global optimum, because the likelihood is multi-modal, which results from the high dimensionality of the mixture's likelihood. Second, given that expectation-maximization provides point estimates of the parameters only, the uncertainties of the estimates (e.g., confidence intervals) are not directly available in these methods. Hence, extra calculations are needed to quantify such uncertainty. We propose a Metropolis-Hastings based algorithm that avoids both shortcomings of expectation-maximization. Indeed, Metropolis-Hastings provides an approximation to the complete (posterior) distribution, given that it samples from the joint posterior of the mixture parameters. This facilitates direct inference (e.g., about uncertainty, multimodality) from the estimation. In developing the algorithm, we tackle various challenges including convergence speed, label switching and selecting the optimum number of mixture components. We then (i) verify the effectiveness of the proposed algorithm on sample datasets with known true parameters, and further (ii) validate our methodology on an environmental dataset (a traditional application domain of Abe-Ley mixtures where measurements are function of direction). Finally, we (iii) demonstrate the usefulness of our approach in an application domain where the circular measurement is periodic in time.

4.1 Introduction

Various scientific fields consider bivariate measurements that have a linear and a circular component. This amounts to data that can naturally be represented on a cylinder. A main challenge in modeling such cylindrical data is accounting for cross-correlation between the circular and linear variables. Additionally, to capture skewness and heterogeneity in the data, mixture models are required, which aggravates the modeling difficulty.

Various parametric distributions have been proposed to jointly model bivariate cylindrical data. Some early examples are models by Mardia and Sutton [1], based on the conditional distribution of a trivariate normal distribution, as well as Johnson and Wehrly [2], based on the principle of maximum entropy, subject to constraints on certain moments. Further extensions of Mardia and Sutton's models have been proposed by Kato and Shimizu's team, based on the conditional of a trivariate normal [3] and the conditional of a trivariate t-distribution [4]. An extension to Johnson and Wehrly's model is defined by Wang [5], with a distribution generated from a combination of the von Mises and transformed Kumaraswamy distributions. A semi-parametric extension to Johnson and Wehrly's model is introduced by Fernández-Durán [6] using non-negative trigonometric sums. Finally, non-parametric models for cylindrical data, based on kernel density estimation are explored by García-Portugués *et al.* [7] and Carnicero [8] .

Recently, Abe and Ley [9] have defined a new cylindrical distribution (now commonly referred to as the Abe-Ley distribution) that is based on the combination of the sine-skewed von Mises [10] and the Weibull distributions. Compared to the other aforementioned cylindrical models, the merits of Abe-Ley are highlighted as having (i) flexible shapes, (ii) cross-correlation among linear and circular variables, (iii) well-known marginal and conditional distributions and (iv) support for data skewness.

Mixtures of Abe-Ley distributions have been used successfully to model environmental data (e.g., for analyzing dynamics of waves [11] and marine currents in the Adriatic Sea [12]) where measurements are a function of the direction (represented by an angle). One of the challenges in estimating parameters for such mixture models is that

(L1) it is difficult to give closed-form expressions for the maximum likelihood estimates (MLEs), which is typically addressed by resorting to numerical methods [9].

Effectively, in current literature the parameters of Abe-Ley mixtures are estimated using expectation-maximization (EM) based on MLE. However, these EM-based methods for parameter estimation of Abe-Ley mixture models (e.g., [11], [12], [13]) have the following limitations:

(L2) the EM methods are based on optimizing the log-likelihood and hence are susceptible to converging to local maxima, and

(L3) being a point estimate, the uncertainties of the estimates (e.g., confidence intervals or standard errors) are not directly available in EM methods.

To address limitation (L2), typically a short run strategy [14] is used to avoid converging to a local maximum while estimating the parameters of the Abe-Ley mixture models [11–13]: the EM algorithm then runs multiple times using different random initialization and stops without waiting for full convergence. However, converging to the global optimum is still not guaranteed in case of a mixture of Abe-Ley distributions, due to the high dimensionality and the complexity of the likelihood function.

To alleviate limitation (L3), further calculations are needed to approximate the uncertainty of the estimation (e.g., confidence intervals). One approach is to approximate the sampling distribution of the estimated parameters via bootstrap methods, and use that approximated distribution to compute confidence intervals.

Approximating the sampling distribution involves randomly sampling from the data with replacement, to create a so-called bootstrap sample (typically of similar size as the original data). For each bootstrap sample, the parameters of interest (in our case, parameters of the Abe-Ley mixture) are estimated using the EM algorithm. The retained EM estimate's instances from each bootstrap sample are then used to approximate the parameter distributions. When using bootstrapping to approximate the sampling distribution for a mixture distribution, we note that one also needs to tackle the label switching issue, caused by the invariance of the likelihood of a K-component mixture model to any permutation of its component indices (see Section 4.3.2.2 for further explanation of the label switching issue).

To circumvent aforementioned limitations of EM-based methods, we propose a Bayesian approach based on Markov Chain Monte Carlo (MCMC) to estimate the parameters of a Abe-Ley mixture distribution. MCMC methods perform the integration of the posterior distribution of the parameters by sampling from it, rather than optimizing the likelihood, thus circumventing aforementioned limitations (L1) and (L2) of EM [15, 16]. Additionally, MCMC-based approaches give joint posterior *distributions* of the parameters. Such distributional info captures the multi-modality (i.e., provides information on both local and global maxima) of the posterior distributions, and also offers insight into the uncertainty of the parameter values (which can be estimated directly by inference from the posterior distribution, without the need for additional calculations). Such Bayesian approaches have been previously successfully applied for modeling circular data (e.g., [17] and [18]), as well as estimating the parameters of finite mixture models for linear variables (e.g., [19] and [20]). Yet, to the best of our knowledge, we are the first to effectively apply a Bayesian approach to estimate the parameters of a (quite complex) bivariate circular-linear distribution and its mixture models.

In the next Section 4.2, we describe the Abe-Ley distribution and the mixture model. Subsequently, we discuss the following contributions:

1. We propose a Metropolis-Hastings (MH) algorithm to estimate the parameters of the Abe-Ley mixture model. Given that we are dealing with a mixture, we note that the MH algorithm is complicated by the need to sample the component weights, in addition to the model parameters for each of the components themselves (Section 4.3.1).

2. We successfully tackle the challenges of the proposed Bayesian MH approach, including (i) convergence speed, (ii) the label switching issue (due to the invariance of the likelihood to permutations of mixture component parameters) and (iii) determining the optimal number of mixture components (Section 4.3.2).

3. We first validate the effectiveness of our approach by showing we can successfully estimate the model parameters for datasets sampled from an a pri-

ori known Abe-Ley mixture distribution, i.e., with a known number of mixture components and known parameter values (Section 4.4).

4. We then apply the proposed approach to two real-world datasets (one traditional application domain with measurements as function of angles, and one new application domain where the circular measurement is periodic time). We show that the Abe-Ley mixture models, estimated using the proposed MH algorithm, effectively capture the heterogeneity in the data under consideration (by referring to the previous analysis of those datasets in literature, see Section 4.5).

5. We illustrate the existence of multi-modality and skewness in the posterior density of the parameters of the Abe-Ley distribution (Section 4.5), which makes EM methods susceptible to converging to local optima. From this, we conclude that the proposed Bayesian approach is more reliable than EM methods in estimating the parameters of the Abe-Ley mixtures (Section 4.6).

4.2 Probabilistic Model Description

In this section, we first introduce the Abe-Ley density function and the Abe-Ley mixture model. We then explain the sampling process from the Abe-Ley distribution as proposed in [9, §3.3], with a minor correction. (We later use the sampling in Section 4.4 to test the effectiveness of our estimation.)

4.2.1 Probability Density Functions

The Abe-Ley distribution is a combination of the Weibull distribution and the sine-skewed von Mises distribution. Its probability density is defined as:

$$f(\theta, x|\boldsymbol{\zeta}) \mapsto \frac{\alpha\beta^{\alpha}}{2\pi\cosh(\kappa)}(1+\lambda\sin(\theta-\mu))x^{\alpha-1}\exp[-(\beta x)^{\alpha}(1-\tanh(\kappa)\cos(\theta-\mu))], \tag{4.1}$$

with random variables $(\theta, x) \in [0, 2\pi) \times [0, \infty)$, and distribution parameters $\boldsymbol{\zeta} = (\alpha, \beta, \mu, \kappa, \lambda)$ [9]. The parameters of the Abe-Ley distribution comprise $\alpha, \beta > 0$, which are linear shape and scale parameters respectively, a circular location parameter $0 \leq \mu < 2\pi$, the parameter $\kappa \geq 0$ that controls the circular concentration and regulates the dependence structure, and finally $-1 \leq \lambda \leq 1$ that controls the circular skewness.

The mixture of a K-component Abe-Ley distribution has the following density function:

$$f(\theta, x|\boldsymbol{\vartheta}) = \sum_{k=1}^{K} \tau_k f_k(\theta, x|\boldsymbol{\zeta_k}) \tag{4.2}$$

where $f_k(\theta, x|\boldsymbol{\zeta_k})$ denotes the probability density of the k^{th} component characterized by parameter set $\boldsymbol{\zeta_k}$, and τ_k is the weight of the k^{th} component. Thus, $\boldsymbol{\tau} = (\tau_1, \tau_2, ..., \tau_K)$ is the weight distribution that takes a value in the unit simplex ε_K which is a subspace of $(\mathbb{R}^+)^K$ defined by the following constraints:

$$\tau_k \geq 0, \quad \tau_1 + \tau_2 + ... + \tau_K = 1. \tag{4.3}$$

Hence, $\boldsymbol{\vartheta} = (\boldsymbol{\zeta}_1, ..., \boldsymbol{\zeta}_K, \boldsymbol{\tau})$ is the parameter vector of the mixture model.

4.2.2 Random Number Generation

One of the strong assets of the Abe-Ley distribution is that it has well-known conditional and marginal distributions, which simplifies the random number generation process. The marginal density of the circular component θ is a sine-skewed wrapped Cauchy distribution and the conditional density $f(x|\theta)$ is defined as

$$\begin{aligned} f(x|\theta) = \alpha \cdot & \left[\beta \left\{1 - \tanh(\kappa)\ \cos(\theta - \mu)\right\}^{1/\alpha}\right]^{\alpha} \\ & \cdot x^{\alpha-1} \cdot \exp\left[-\left\{\beta\left(1 - \tanh(\kappa)\ \cos(\theta - \mu)\right)^{1/\alpha} x\right\}^{\alpha}\right] \end{aligned} \tag{4.4}$$

Abe and Ley [9] state (4.4) to be a Weibull distribution with shape parameter $\beta\ (1 - \tanh(\kappa)\ \cos(\theta - \mu))^{1/\alpha}$, whereas according to the standard definition of the Weibull distribution, (4.4) has shape parameter α and scale parameter $\beta(1 - \tanh(\kappa)\ \cos(\theta - \mu))^{-1/\alpha}$ (see [9] for mathematical details). Accounting for this corrected terminology, randomly generating numbers following the Abe-Ley distribution [9] can be achieved as follows:

Step 1: Generate a random variable[1] Θ_1 from a wrapped-Cauchy distribution with location parameter μ and concentration $\tanh(\kappa/2)$.

Step 2: Generate U from a uniform distribution on $[0, 1]$ and define

$$\Theta = \begin{cases} \Theta_1 & \text{if } U < (1 + \lambda\ \sin(\Theta_1 - \mu))/2 \\ -\Theta_1 & \text{if } U \geq (1 + \lambda\ \sin(\Theta_1 - \mu))/2 \end{cases}$$

to ensure Θ follows the sine-skewed wrapped Cauchy distribution.

Step 3: Generate X from a Weibull with shape parameter α and scale parameter $\beta(1 - \tanh(\kappa)\ \cos(\Theta - \mu))^{-1/\alpha}$.

To draw N samples from a K-component Abe-Ley mixture, we repeat the aforementioned 3 steps for each mixture component k characterized by $\boldsymbol{\zeta}_k$, where the expected number of samples from the k^{th} component is $\tau_k\ N$.

[1] Capital letters indicate sampled data instances (as opposed to lowercase variable notations).

4.3 Parameter Estimation using Bayesian Inference

In this section, the proposed Metropolis-Hastings algorithm for estimating the parameters of a mixture of Abe-Ley distributions is explained and the associated challenges are addressed.

4.3.1 Metropolis-Hastings Algorithm for Estimating Abe-Ley Mixture Parameters

Let us assume $\boldsymbol{S} = \{(\theta_1, x_1),\, (\theta_2, x_2),\, \ldots,\, (\theta_N, x_N)\}$ is a set of N observations from a K-component Abe-Ley mixture distribution defined by (4.2) where $\boldsymbol{\vartheta} = (\boldsymbol{\zeta}_1, \ldots, \boldsymbol{\zeta}_K, \boldsymbol{\tau})$ is the unknown parameter vector.

Calculating the posterior density by solving analytical equations is impossible, since it involves calculating intractable integrals. To overcome this challenge, typically Markov-Chain Monte-Carlo (MCMC) methods are used to generate samples from the posterior distribution. A well-known MCMC-based algorithm is Metropolis-Hastings (MH).

To estimate a parameter $y \in \boldsymbol{\zeta}$ of the target distribution (Abe-Ley in our case) from a set of observations $\boldsymbol{S}$, MH iteratively refines its estimate y^i in iteration i, starting from an initial value y^0 for $i = 0$. Given y^i in each iteration i, a new draw y^* is obtained from a predefined proposal distribution $q(y^*|y^i)$. Then, y^* is accepted with acceptance probability

$$A = \min\left\{1,\ \frac{p(y^*|\boldsymbol{S})\, q(y^{(i)}|y^*)}{p(y^{(i)}|\boldsymbol{S})\, q(y^*|y^{(i)})}\right\} \tag{4.5}$$

where p denotes the target density.

Finally, the accepted draws are returned as the output of MH algorithm. Note that, when estimating the components of the mixture models, the MH algorithm should also estimate the component allocations and weight distributions of each component (τ_k) in addition to its parameters ($\boldsymbol{\zeta}_k$) .

Algorithm 3 summarizes our approach, which basically adapts the MH algorithm to estimate the parameters of the K-component Abe-Ley mixture and obtain the component membership of each observation. Similar to the original MH, the input to our algorithm comprises (i) the target distributions (i.e., the Abe–Ley mixture model defined by (4.2)), (ii) the observations $\boldsymbol{S}$, with $|\boldsymbol{S}| = N$, and (iii) the prior and proposal distributions for each parameter. Table 4.1 summarizes the choice of the priors and the proposal distributions for each parameter of the Abe-Ley distribution. The priors are non-informative (i.e., the hyper-parameter selection is such that the resulting priors are almost uniform across the parameter domains). The choices of proposal distributions (used for sampling the new value for parameters) are (truncated) normal distributions defined on the permitted

Table 4.1: Choice of priors and proposal distributions at the input of Algorithm 3

Parameter	Prior (hyper-parameters)	Proposal*
α	Gamma(shape = 0.001, scale = 1000)	Truncated normal on $[0, \infty)$
β	Gamma(shape = 0.001, scale = 1000)	Truncated normal on $[0, \infty)$
κ	Gamma(shape = 0.001, scale = 1000)	Truncated normal on $[0, \infty)$
μ	von Mises(mean = 0, precision = 0.001)	normal *mod* 2π
λ	Beta(shape1 = 1, shape2 = 1) (rescaled)	Truncated normal on $[-1, 1]$
$\boldsymbol{\tau}$	Dirichlet(vector of ones with length K)	Not-applicable

* The means of the proposal distributions at each iteration are the draws in the previous iteration (or the initial component values at the first iteration); the variances of the proposal distributions are adjusted every 50 iterations (see Section 4.3.2.1 for details.)

domain of the parameters. The Abe-Ley density function is used for determining the component allocation in each iteration of the algorithm. The output of the algorithm comprises draws for the parameter vector $\boldsymbol{\vartheta}$ and the allocation vector $\boldsymbol{l} = (l_1, \ldots, l_N)$ that indicates to which component each observation belongs.

Algorithm 3 starts by randomly initializing the component parameters on their permitted domains to obtain $\boldsymbol{\vartheta}^0 = (\boldsymbol{\zeta}_1^0, \ldots, \boldsymbol{\zeta}_K^0, \boldsymbol{\tau}^0)$ and assigns each observation $(\theta_n, x_n) \in \boldsymbol{S}$ to a component with probability $\boldsymbol{\tau}^0$ to obtain the initial allocation vector $\boldsymbol{l}^0$ (Line 16). The algorithm then runs for $(M_0 + M)$ iterations, where M_0 is the number of initial samples to disregard (burn-in samples). Each iteration consists of two parts. In the first part (Lines 18-20), the parameters $\boldsymbol{\zeta}_1^i, \ldots, \boldsymbol{\zeta}_K^i$ of each component of the mixture are drawn using the MH algorithm: for each component, new parameter values are sampled from the proposal distributions defined in Table 4.1 (Line 20) and are accepted with probability A defined by (4.5) (Line 20).

In the second part of the iteration, the allocation vector $\boldsymbol{l}$ and component weight vector $\boldsymbol{\tau} = (\tau_1, \ldots, \tau_K)$ are sampled (Lines 21-22). To identify $\boldsymbol{l}^i$ (the allocation vector in iteration i), first the probability of each observation (θ_n, x_n) belonging to a component k of the mixture is calculated independently using $p((\theta_n, x_n)|\boldsymbol{\zeta}_k^i)$, where $\boldsymbol{\zeta}_k^i$ is the parameter vector of component k drawn at iteration i. Note that $p((\theta_n, x_n)|\boldsymbol{\zeta}_k^i)$ is an Abe-Ley distribution defined by (4.1) and not an Abe-Ley mixture model. In other words, $p((\theta_n, x_n)|\boldsymbol{\zeta}_k^i)$ denotes the probability of observation n coming from an Abe-Ley distribution with parameters $\boldsymbol{\zeta}_k^i$. The observation is then assigned to a component k with probability $p((\theta_n, x_n)|\boldsymbol{\zeta}_k^i)$. (Line 21). Once $\boldsymbol{l}^i$ is identified, the number of observations allocated to each component of the mixture is counted to calculate the parameter vector of a Dirichlet distribution. The component weight vector $\boldsymbol{\tau}^i = (\tau_1^i, \ldots, \tau_K^i)$ is then sampled from that Dirichlet distribution (Line 22). Finally, the first M_0 draws are discarded and M remaining draws are returned for $(\boldsymbol{\vartheta}, \boldsymbol{l})$ (lines 23-23).

algorithm 3: Metropolis-Hastings

```
Input  : The target distribution $f(\theta, x|\boldsymbol{\vartheta})$ with unknown parameter vector $\boldsymbol{\vartheta}$ (defined by (4.2)); data samples $\boldsymbol{S}$ with $|\boldsymbol{S}| = N$; priors and proposal distributions for each parameter in $\boldsymbol{\vartheta}$ (see Table 4.1)
Output: the draws for $\boldsymbol{\vartheta}$ (i.e., $\boldsymbol{\vartheta}^i$); the allocation vector $\boldsymbol{l} = (l_1, \dots, l_N)$ /* $\boldsymbol{l}$ denotes to which component each of the observations are assigned (if $(\theta_n, x_n)$ is part of component $k$, then $l_n = k$) */
Initialize $\boldsymbol{\vartheta}^0 = (\boldsymbol{\zeta}_1^0, \dots, \boldsymbol{\zeta}_K^0, \boldsymbol{\tau}^0)$ and $\boldsymbol{l}^0$; /* allocate samples in $\boldsymbol{S}$ to each component with probability $\boldsymbol{\tau}^0$ to obtain $\boldsymbol{l}^0$ */
foreach $i = 0, \dots, M_0 + M - 1$ do
    /* Given $\boldsymbol{l}^i$, estimate the mixture parameters */
    foreach $k = 1, \dots, K$ do
        foreach $y \in \boldsymbol{\zeta}_k$ do
            /* update parameters of each component in this loop */
            Sample the component parameter $y^*$ from a proposal distribution $y^* \sim q(y^*|y^{(i)})$ Sample $u$ from a Uniform distribution on $[0, 1]$ Calculate the acceptance ratio as $A = \min\{1, \frac{p(y^*)\, q(y^{(i)}|y^*)}{p(y^{(i)})\, q(y^*|y^{(i)})}\}$ $y^{(i+1)} = \begin{cases} y^* & u < A \\ y^{(i+1)} & u \geq A \end{cases}$
    Classify each observation $(\theta_n, x_n) \in \boldsymbol{S}$ conditional on knowing $\boldsymbol{\zeta}_1^i, \dots, \boldsymbol{\zeta}_K^i$, by sampling $l_n$ independently for each $n = 1, \dots, N$ from: $p\left(l_n = k|\boldsymbol{\zeta}_1^i, \dots, \boldsymbol{\zeta}_K^i, (\theta_n, x_n)\right) \propto p\left((\theta_n, x_n)|\boldsymbol{\zeta}_k^i\right)$ to obtain $\boldsymbol{l}^i$
    Sample $\boldsymbol{\tau}^i = (\tau_1^i, \dots, \tau_K^i)$ from the Dirichlet distribution $D(e_1(\boldsymbol{l}^i), \dots, e_K(\boldsymbol{l}^i))$, where $e_k(\boldsymbol{l}^i) = e_0 + N_k(\boldsymbol{l}^i), k = 1, \dots, K$, and $N_k(\boldsymbol{l}^i)$ is the number of data points allocated to component $k$ of the mixture at iteration $i$ and $e_0$ is the prior of the Dirichlet distribution
Disregard the first $M_0$ draws  return $M$ draws for $\boldsymbol{\vartheta}$ and the allocation vector $\boldsymbol{l}$
```

4.3.2 Addressing the Challenges of a Bayesian Approach

In this section, we explain how we tackle three computational aspects in our proposed approach: (1) improving the convergence rate of the MH algorithm via an *adaptive* Metropolis-Hastings algorithm, (2) addressing the label switching issue (caused by invariance of the mixture likelihood to a permutation of component parameters) and (3) Bayesian model selection (for determining the optimal number of mixture components). Note that the second and third challenges are inherent to parameter estimation for mixture models in general, both in Bayesian approaches as well as EM-based methods.

4.3.2.1 Adaptive Metropolis-Hastings

A crucial factor in developing an efficient MH algorithm is the definition of the proposal distribution, $q\left(y^*|y^i\right)$. In most applications, a symmetric, unimodal distribution such as the Gaussian distribution is chosen. The MH algorithm requires that the variance of the proposal distribution is preset and does not change during the execution of the MCMC procedure. However, the choice of this variance parameter has an important impact on the efficiency of the MCMC algorithm. When the variance of the proposal distribution is too large, the acceptance probability (defined by (4.5)) will tend to 0. As a result, the Markov Chain will retain its current value and only jump to new values with vanishing small probability. On the other hand, when the variance of the proposal distribution is too small, the acceptance probability will become close to 1. In this case, the Markov Chain is constantly sampling new values, but these values are very close to the current value, and it will take an excessive amount of time before the entire posterior distribution is sampled.

The variance of the proposal distribution has to be set so that both inefficient behaviors outlined above are avoided. Standard practice [21] is to sample from the posterior distribution for some time and then evaluate the acceptance probabilities. The variances of the proposal distributions should then be adjusted in order to achieve an acceptance probability of $0.3 - 0.5$. These values ensure that the algorithm does not output the same value, while still making reasonably large jumps. However, in the current mixture model, this approach is difficult to implement. As shown in Algorithm 3, the parameters of each component of the mixture are sampled in blocks. Each component has five parameters, $\boldsymbol{\zeta} = (\alpha, \beta, \mu, \kappa, \lambda)$. In the MH algorithm we have to set a variance for each parameter of the K components, so $5 \cdot K$ in total. Tuning these variances is tedious and time consuming.

To overcome the aforementioned challenge, we use a modified version of the MH algorithm proposed by Haario *et al.* [22] that automatically adjusts the variance of the proposal distribution to maximize efficiency. The basic idea of the adaptive Metropolis-Hastings algorithm is that every R iterations (e.g., $R = 50$), the acceptance probabilities are calculated and evaluated. Whenever the acceptance probabilities are above (below) some threshold (e.g., 0.44), the variance of the proposal distribution is increased (decreased) with an amount $s = \min(0.01, \sqrt{R/i})$ for the next R iterations. At that point, the acceptance probabilities are re-evaluated, and the variances are adjusted again, if necessary. Note that the adjustment amount s is a function of the current iteration i of the MCMC chain, such that at the beginning of the chain larger adjustments are possible, while subsequent adjustments are forced to become continuously smaller.

A detailed description of the adaptive Metropolis-Hastings algorithm and the implications for the mathematical foundations of the algorithm can be found in [22].

4.3.2.2 Label Switching Issue

Note that the likelihood of the K-component mixture model in (4.2) is invariant to any permutation of its component indices (which amounts to a total of $K!$ permutations). Under symmetric priors (i.e., when exchangeable priors are chosen for the component parameters), the resulting posterior will also be invariant to $K!$ permutations in the labeling of the component parameters. In other words, the posterior will have $K!$ symmetric modes. As a result, labels of the components can permute multiple times in subsequent iterations of the MCMC sampling, resulting in a label switching problem. Since in our Bayesian approach, that posterior is used (as distribution for component parameters) for inference of the model parameters, the label switching issue makes such inference very challenging.

Early attempts to solve the label switching issue focus on proposing identifiability constraints via prior distributions to force a unique labeling (e.g., [19, 23]). However, as shown by Stephens [24], identifying such constraints is not always feasible, especially when systematically separating the posterior modes is not possible. Hence, two categories of relabeling methods are proposed to post-process the MCMC samples: (1) deterministic relabeling methods that find the optimal permutation in each iteration of the MCMC sampler by minimizing a loss function (e.g., Stephens' algorithm [24], the pivotal reordering algorithm [25, 26], default [27] and iterative versions [28] of algorithms for equivalence class representatives, data-based algorithms [28]), and (2) probabilistic relabeling methods that treat permutation of the parameters as missing data with associated uncertainty and estimate its density using an EM type approach (e.g., [29]).

We refer the interested reader to [30] for an explanation and performance comparison of the relabeling methods in terms of CPU times as well as to what extent they agree on the component labels. We note that the data-based relabeling algorithm [28] performs better in terms of both performance criteria (as shown in [28, 30]), and hence use it for the relabeling of the samples from Algorithm 3. The data-based relabeling is based on the key idea that in a converged MCMC, while the labels of each cluster might change from one iteration to the other, the clusters remain almost the same. Leveraging such minute difference between the clusters of each iteration, one may keep track of the k clusters throughout each MCMC iteration to identify cluster movements. Further details of the relabeling algorithm are outlined in [28, Algorithm 5].

4.3.2.3 Bayesian Model Selection

In many modeling problems, the number of mixture components is not known and needs to be identified as a part of the model selection process. Earlier attempts tried to estimate the true number of components either by calculating the marginal likelihoods (e.g., [31, 32]) or by trans-dimensional MCMC samplers (e.g., reversible jump MCMC [33]). In recent approaches however, the view of model selection is changed from *identifying the true model* to *finding a useful model* [34]. In the latter case, model usefulness is seen as its predictive ability for future or unseen data (i.e., out-of-sample prediction accuracy [35]). Vehtari *et al.* [36] quantify the out-of-sample prediction accuracy as expected log pointwise predictive density (elpd). However, since future data is not available, to calculate elpd, first a log point-wise prediction density (lpd) is calculated using the observed samples. Here, lpd is an over-estimation of elpd for future data, which can be corrected with a bias term [35]. The lpd measure is calculated from the posterior samples using leave-one-out cross-validation (LOO-CV) as

$$lpd_{LOO\text{-}CV} = \sum_{i=1}^{n} \log \left(\frac{1}{S} \sum_{s=1}^{S} p\left(y_i | \theta^{is}\right) \right) \tag{4.6}$$

where n is the number of observations, S is the number of samples from the posterior, θ^{is} is a sample s from the posterior samples drawn based on all but observation y_i, and $p(y_i|\theta^{is})$ is the probability of observation y_i given posterior parameter θ^{is}.

The above calculations are based on $n-1$ observations. If n is large, the overestimation is negligible, otherwise it is corrected using a bias b that denotes the improvement of an estimation when n observations are considered.

Note that the calculations of $lpd_{LOO\text{-}CV}$ are computationally expensive for a large number of observations. Hence, Vehtari *et al.* [36] also propose an efficient approximation of $lpd_{LOO\text{-}CV}$ using Pareto-smoothed importance sampling (PSIS). Still, the approximations by PSIS-LOO are not reliable when the estimated shape parameter of the generalized Pareto distribution exceeds 0.7 [36]. In that case, 10-fold cross-validation is used to estimate the elpd values as outlined in [36, Section 2.3].

For every modeling endeavor there is a trade-off between the interpretability of a model and the predictive performance. Here we focus mainly on the latter, hence we use elpd values to find the optimum number of mixture components. To avoid overfitting, we use graphical inspection of the elpd measure's evolution for increasing the number of mixture components. We identify a knee point in that graph as a point beyond which the increase in the number of components K results in a smaller, or at least not better, elpd value compared to that for smaller K.

4.4 Validation on a Sample Dataset with Known True Parameters

Before moving to applying our approach to real-world data, we first validate its capability of correctly estimating the parameters from synthetic samples generated using a known Abe-Ley mixture model. We generate such data using the random

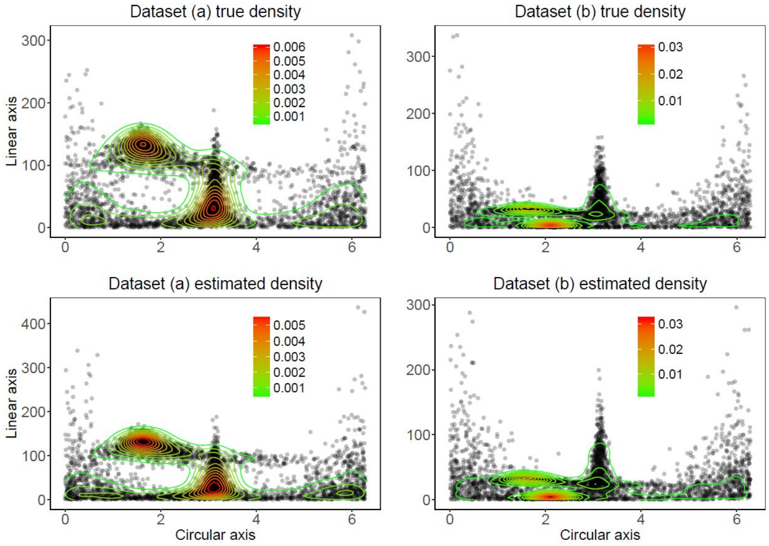


Figure 4.1: Sampled datasets from mixture of true (top row) and estimated (bottom row) Abe-Ley distributions. Contour plots indicate two-dimensional kernel densities

number generation process explained in Section 4.2.2. We then run Algorithm 3 for a total of $M_0 + M = 100,000$ iterations, from which we disregard the $M_0 = 20,000$ initial draws that are considered as burn-in. Additionally, to reduce auto-correlation between the samples, we use thinning by a factor 5, (i.e., only keeping every 5th draw of the MCMC chain) in Algorithm 3. Hence, we finally retain 20,000 draws with a burn-in of 5,000 initial samples. The choice of 5 is based on the auto-correlation plots of the posterior draws.[2] We use trace plots to examine the convergence and mixing performance of an MCMC chain. As a spread measure of the posterior distribution, we use a 95% Bayesian credible interval.

The top row of Figure 4.1 shows the data sampled from a mixture of three (dataset (a)) and four (dataset (b)) Abe-Ley distributions along with the true Abe-Ley mixture densities in the form of contour plots. Both datasets contain 4,500 samples. The number of samples from each component of the mixture is the same in dataset (a) but different per component in dataset (b) (see component weights from Table 4.2).

As mentioned earlier, one of the defining advantages of the MH algorithm is that — unlike likelihood-based estimations, which are point estimates — the MH

[2] Note that the auto-correlation plots are excluded from this paper to maintain a reasonable paper length.

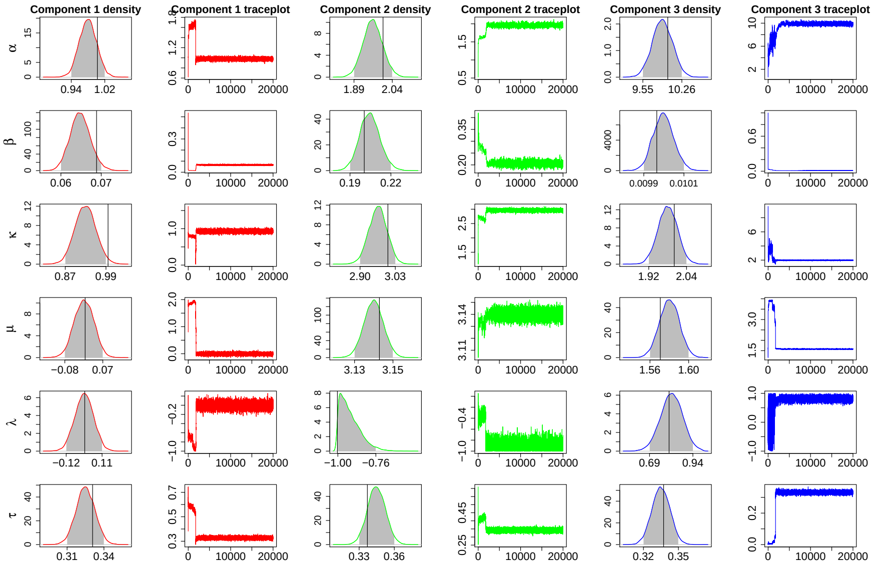


Figure 4.2: Posterior densities and trace plots of the parameters for mixture of 3 Abe-Ley (Dataset (a)). Black vertical lines mark the true parameter values and shaded areas are the 95% Bayesian credible intervals

Table 4.2: Estimated and true parameters for sampled data from mixture of Abe-Ley distribution (The true values are shown in parenthesis)

Dataset	k	α	β	κ	μ	λ	τ
(a)	1	0.98 (1)	0.07 (0.07)	0.93 (1)	0.03 (0)	0.00 (0)	0.33 (0.33)
	2	1.96 (2)	0.21 (0.2)	2.97 (3)	3.14 (3.14)	-0.98 (-1)	0.34 (0.33)
	3	9.91 (10)	0.01 (0.01)	1.97 (2)	1.58 (1.57)	0.82 (0.8)	0.33 (0.33)
(b)	1	1.00 (1)	0.07 (0.07)	0.95 (1)	0.06 (0)	-0.06 (0)	0.3 (0.31)
	2	1.99 (2)	0.19 (0.2)	3.00 (3)	3.14 (3.14)	-0.79 (-1)	0.18 (0.18)
	3	10.10 (10)	0.04 (0.04)	2.01 (2)	1.58 (1.57)	0.85 (0.8)	0.24 (0.24)
	4	2.97 (3)	1.02 (1)	3.00 (3)	2.09 (2.09)	0.53 (0.5)	0.27 (0.27)

algorithm outputs samples from the posterior distribution of the model parameters, making the uncertainty of the estimates directly inferable from the posterior densities, without the need for extra calculations (such as in bootstrapping). The posterior densities of the parameters for a mixture of 3 Abe-Ley distributions (dataset (a)) are shown in Figure 4.2. The accompanying trace plot for each parameter is used to analyze the convergence and mixing performance of the MH algorithm. As seen from the trace plots, the burn-in of 5,000 initial samples is sufficient to disregard the unstable initial draws of the algorithm. The retained draws are from the higher probability region of the posterior and are close to the true values of the parameters, indicating that the chain has converged. The density plots in Figure 4.2 are based on the 15,000 retained draws. Finally, the trace plots also confirm that the draws among various iterations are not identical: the chain is mixing well and effectively exploring the posterior. For some parameters however, the mixing of the chain is not apparent: this is due to the scaling issue, caused by a large difference between the starting and the true value of the parameter. Also, very low auto-correlation is observed among the draws, which confirms efficient exploration of the posterior.

The shaded regions in the density plots of Figure 4.2 indicate the 95% Bayesian credible intervals. The boundaries of the credible intervals are marked with numeric values on the horizontal axis, while the black vertical lines show the true values of the parameters. For some parameters, the true values are in the tails of the posterior. This is only natural, since there is always a 5% chance that the true value will be outside this credible interval, and we have 5 parameters per component plus the component weights (in this case totaling 18).

To demonstrate the estimated predictive density for both datasets, we use the last 4,500 MCMC draws of $\boldsymbol{\vartheta}^i$ and for each draw, sample a data point from the Abe-Ley mixture distribution parameterized by $\boldsymbol{\vartheta}^i$. The generated data points are composed and presented in Figure 4.1. We then use a two-dimensional kernel density estimate to obtain the estimated predictive density (shown in the form of contour plots in Figure 4.1). As seen from Figure 4.1, the estimated and true predictive density of the Abe-Ley mixtures for both datasets are very similar. This comparison further validates the effectiveness of the proposed approach in estimating the parameters of the Abe-Ley mixture distributions.

While noting that Bayesian estimation is not a point estimate, still, to be able to numerically compare the component-wise densities for the true and the estimated parameters, we summarize the posterior distributions in point forms. To do that, we use maximum a posteriori (MAP) estimation, which corresponds to the mode of the empirical distribution of the posterior. An alternative summarization would be taking the mean of the posterior distribution, but that Bayes estimate is not suitable for multi-modal posteriors. The true and estimated parameters we thus obtain are summarized in Table 4.2. These results indicate that our proposed algorithm can effectively estimate the mixture model parameters for both datasets. For dataset (a), we further demonstrate component-wise densities for the true and the estimated parameters in Figure 4.3. This figure shows that the true and estimated densities for the mixture of 3 Abe-Ley distributions are very similar.

To validate the effectiveness in model selection of elpd, approximated by PSIS-LOO, we have calculated the elpd for a varying number of mixture components for both datasets as shown in Figure 4.4. The location of the bend (knee) in Figure 4.4 indicates the most suitable number of components, which is the same as the number of true mixtures for both datasets.

The examples presented above demonstrate the effectiveness of our proposed

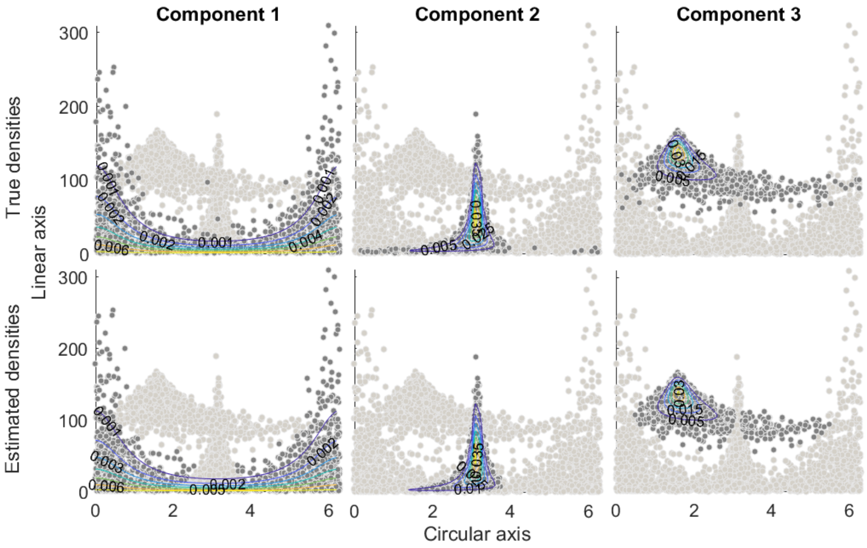


Figure 4.3: Component-wise densities for the true and the estimated parameters for Dataset (a)

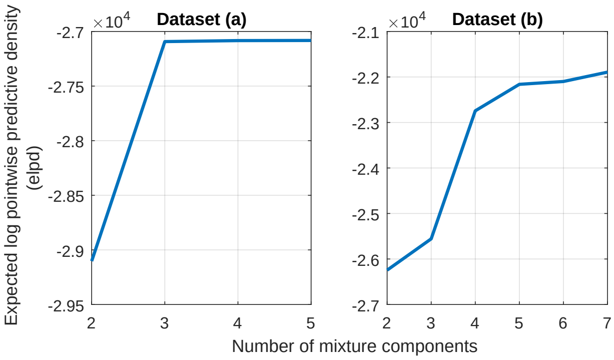


Figure 4.4: Determining number of mixture components using elpd measure for sample datasets (the bend in each curve is used for selecting the best number of mixture components).

algorithm for estimating the parameters of a mixture of Abe-Ley distributions and of elpd as suitable model selection measure. Next, we apply our approach to model the data for two different real-world applications.

4.5 Modeling Real-World Datasets with a Mixture of Abe-Ley Distributions

We now apply our methodology to fit an Abe-Ley mixture to (1) a wave dynamics dataset, and (2) an EV charging dataset. The first application is a traditional application domain of mixtures of cylindrical distributions where the circular measurement is the direction (angle). The dataset in the second example on the other hand is a new one, where the circular measurement is periodic time. For both applications, we run Algorithm 3 for $M_0 + M = 100,000$ iterations and discard a burn-in of $M_0 = 20,000$ draws. Further, we use thinning by a factor 5, i.e., we only keep every 5th draw of the MCMC chain) to reduce the auto-correlation among subsequent draws.

4.5.1 Wave Dynamics in the Adriatic Sea

In this section, we consider a dataset of wave dynamics, which is a well-studied application of the Abe-Ley distribution. The dataset comprises semi-hourly wave directions and heights in the Adriatic Sea, recorded in the period 15 February 2010 to 16 March 2010 as reported by [11]. Lagona *et al.* [11] also approximate this data with a mixture of Abe-Ley distributions, whose parameters depend on the states of a latent Markov chain. However, their proposed estimation algorithm is based on the EM method (and thus a point estimate), whereas our proposed approximation algorithm is based on the MCMC method (thus giving a posterior distribution for the mixture parameters). Additionally, the procedure by Lagona *et al.* [11] includes a temporal dependence for the data, based on a hidden Markov model. In this work, we focus on fitting the distribution only, while addressing temporal dependence could be interesting as future work.

Figure 4.5 shows the elpd values for different numbers of Abe-Ley mixtures and suggests $K = 4$ as the optimum number of mixtures because the improvement in elpd when increasing the number of mixtures from 2 to 4 is significantly larger than for the increase from 4 to 7 and beyond. Hence, $K = 4$ is a knee point. Note that in [11] the best number of states (mixtures) is deemed to be $K = 3$ using the BIC measure. To represent and compare the distribution of each component of the mixture models, we use MAP estimation to summarize the estimated parameters from our approach in a point form. The resulting mixtures are depicted in Figure 4.7(b) and are compared with the fits from Lagona *et al.* [11] shown in Figure 4.7(a).

Figure 4.7 suggests that both approaches identify similar heterogeneity in the data. The first component of Lagona *et al.* models the high waves coming from the north [11], which in our model are represented by the first and the second components. The second component of Lagona's model (and our third component)

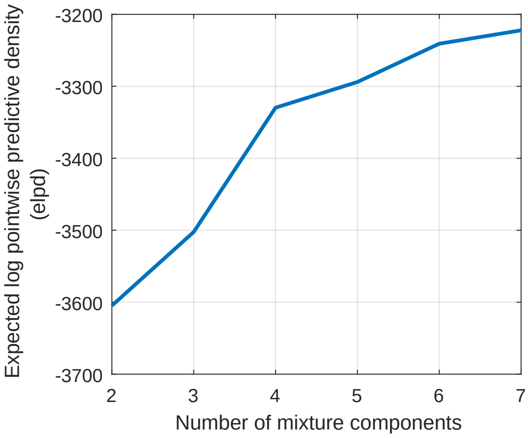


Figure 4.5: The elpd values for model selection in modeling wave dynamics. (The bend in the curve is used for selecting the best number of mixture components.)

is associated with calm sea and finally, their third component (fourth in our model) is associated with Sirocco episodes (caused by wind blowing southeasterly, along the major axis of the Adriatic Sea). However, the posterior densities of the 4 mixture components depicted in Figure 4.6 confirm that the Bayesian approach is more reliable than the EM method. The posterior densities for component 1 and 2 in Figure 4.6 are multi-modal, making the EM based approaches susceptible to premature convergence to a local maximum.

4.5.2 Electric Vehicle Hourly Charging Requests

Our second study is motivated by the increasing use of EVs and the need to analyze their impact on the power grid. Initial studies only presented empirical distributions of the arrival times of EVs from real-world measurements [37–40]. Here, we take the first step to model the arrival times of EVs using a mixture of Abe-Ley distributions. We use an EV charging session dataset (collected by ElaadNL[3]) that includes the arrival times of electric vehicles at public roadside charging stations across the Netherlands from January to March 2015.

We divide a day into hour-long slots and count the number of EV arrivals in each slot. We also take the mean time-of-arrival of EVs in each slot to charac-

[3] ElaadNL is the knowledge and innovation center in the field of charging infrastructure in the Netherlands, providing coordination for the connections of public roadside charging stations to the electricity grid on behalf of 6 participating distribution system operators (DSOs). It also performs technical tests of charging infrastructure, researches and tests smart charging possibilities of EVs, and develops communication protocols for managing EV charging. The EV charging session data is available upon request for non-commercial research purposes, subject to signing an agreement. For more information, please contact Chris Develder (email: chris.develder@ugent.be).

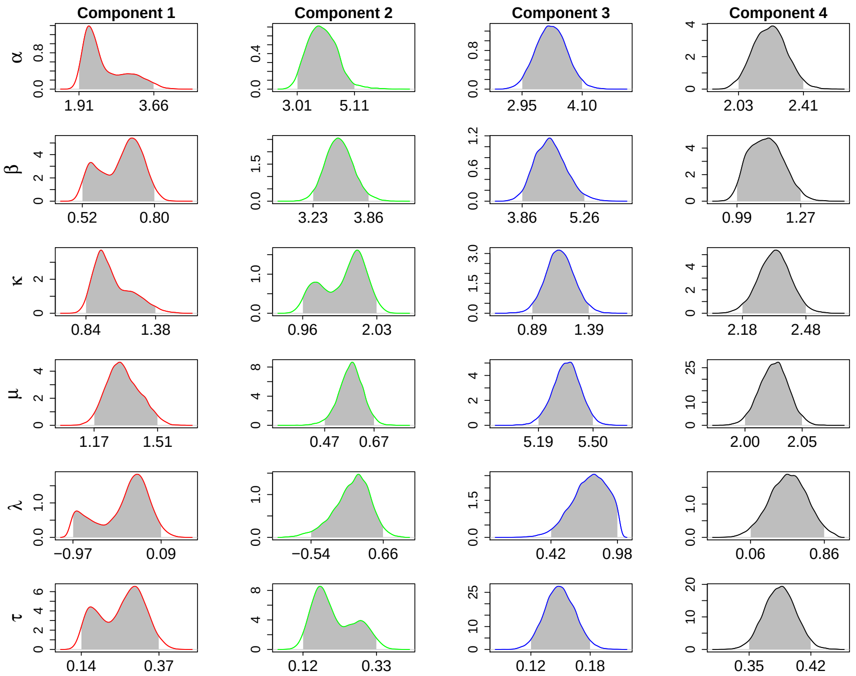


Figure 4.6: Posterior densities of the parameters for best Abe-Ley mixture model for wave dynamics.

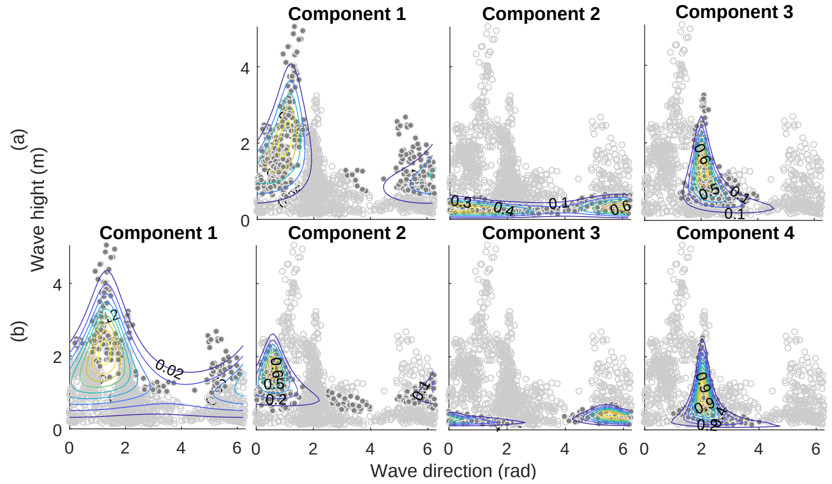


Figure 4.7: Component-wise densities for (a) Abe-Ley mixture model estimated by [11] and (b) Abe-Ley mixture model estimated by our proposed approach.

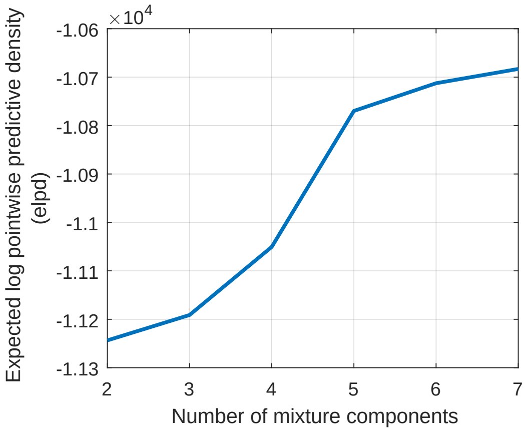


Figure 4.8: elpd values for model selection in modeling EV arrivals (the bend in the curve is used for selecting the best number of mixture components)

terize the timing aspect of the measurement. Therefore, the resulting data points have one linear (number of EV arrivals) and one circular (average time-of-arrival) measurement and are best represented on a cylinder. Note that the linear measurements in this dataset are of discrete nature. However, due to unavailability of the cylindrical distributions that effectively take into account the cross-correlation of the circular and linear measurements for discrete data, we have modeled this dataset with an Abe-Ley distribution as the best existing candidate. To prevent over-fitting, we add a random value generated from a uniform distribution on (0,1) to the EV counts in each slot.

We fit the mixture of Abe-Ley distributions with a varying number of mixture components to the EV dataset and use the elpd measure to select the best number of mixture components, as illustrated in Figure 4.8. The bend in the elpd values suggests $K = 5$ mixtures to model the EV arrival distribution. The estimated posterior densities are shown in Figure 4.9, where the shaded regions indicate the 95% Bayesian credible intervals. We also use MAP estimation to numerically summarize the posteriors in point forms and show the component densities in Figure 4.10.

Next, the resulting mixtures are compared with our previous studies on this dataset. In [39, 40], we clustered this data on a 2-dimensional surface (i.e., time-of-arrival vs. time-of-departure) into 3 clusters: charge-near-work (characterized by early morning arrivals, mainly on weekdays), charge-near-home (with late afternoon arrivals), and park-to-charge (with arrivals throughout the day). We also found that the EV arrivals have different empirical distributions on weekdays compared to weekends: weekday arrivals have two peaks (mornings and evenings), whereas the weekend arrivals only peak around noon.

This heterogeneity is very well captured by the mixture of the 5-component

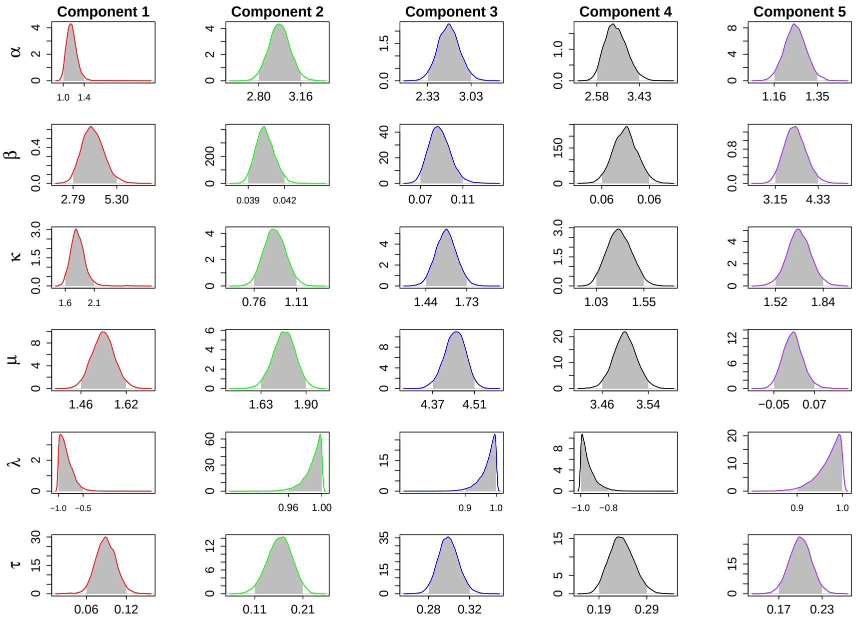


Figure 4.9: Posterior densities of the parameters for best Abe-Ley mixture model for EV arrival

Abe-Ley distribution. As seen in Figure 4.10, the second component models the morning peaks in EV arrivals, which are typically arrivals on weekdays and from the charge-near-work cluster. The third component of the mixture comprises weekday arrivals, from charge-near-home sessions. The fourth component models the weekend arrivals and day-time charging during the weekdays, which are typically park-to-charge sessions. Finally, the probability of having a very small number of EV arrivals is modeled by the first (for very early morning arrivals) and fifth (for arrivals around midnight) components of the mixture.

4.6 Conclusion

In this paper, a Metropolis-Hastings algorithm based on MCMC sampling was proposed for estimating the components of Abe-Ley mixture models. A dynamic Metropolis-Hastings algorithm was used to adjust the variance of the proposal distribution to improve both convergence and exploration of the posterior distribution in the proposed algorithm. Two challenges associated with estimating the mixture parameters were also tackled: the label switching issue and the selection of the optimal number of components.

By referring to the posterior distributions of the parameters of the Abe-Ley

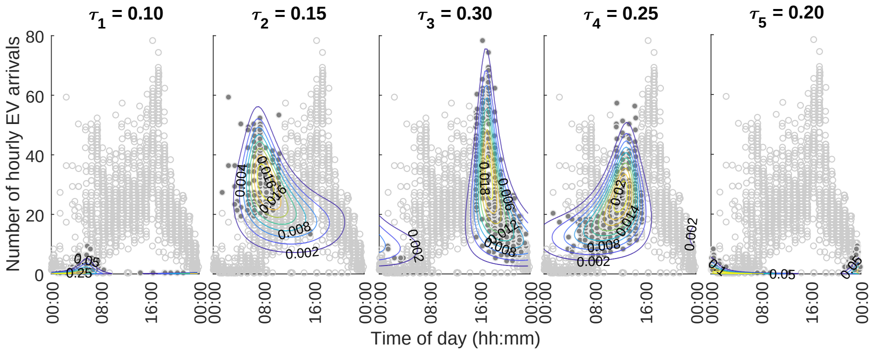


Figure 4.10: Component-wise densities for the true and estimated parameters in modeling EV arrivals

mixture, we illustrated that a Bayesian based estimation is more reliable than EM methods for estimating these parameters, because: (1) the multi-modality of the posteriors can be inferred directly from the output of the MH algorithm, hence, local maxima are avoided without a need for additional calculations (which also do not guarantee arriving at a global optimum), and (2) in EM-based estimation, the uncertainty of the estimates is not available and further calculations are needed to approximate them. While bootstrapping could address this, in our proposed Bayesian approach, the uncertainty of the estimation is directly inferred from the posteriors.

We validated the effectiveness of our proposed approach as well as the model selection measure by estimating the parameters of actual Abe-Ley mixture models. Further, we applied our proposed approach to model the data from two different real-world application domains: wave dynamics and electric vehicle (EV) arrivals. In both applications, the Abe-Ley mixtures captured the data skewness, the correlation between the circular and linear variables and the data heterogeneity (multi-modality). We found that the resulting mixtures were intuitively appealing and qualitatively in accordance with previous studies of the real-world datasets.

Acknowledgements

We thank ElaadNL for providing the data and relevant insights on EV charging in the Netherlands. Special thanks goes out to Nazir Refa (ElaadNL) for preparing the raw data for analysis. We also thank Prof. Francesco Lagona for sharing the wave dynamics data of the Adriatic Sea. Finally, we thank Dr. Thomas Demeester for his expert advice on the bootstrapping approach.

References

[1] K. V. Mardia and T. W. Sutton. *A Model for Cylindrical Variables with Applications*. Journal of the Royal Statistical Society. Series B (Methodological), 40(2):229–233, 1978.

[2] R. A. Johnson and T. E. Wehrly. *Some Angular-Linear Distributions and Related Regression Models*. Journal of the American Statistical Association, 73(363):602–606, 1978.

[3] S. Kato and K. Shimizu. *Dependent models for observations which include angular ones*. Journal of Statistical Planning and Inference, 138(11):3538 – 3549, 2008. Special Issue in Honor of Junjiro Ogawa (1915 - 2000): Design of Experiments, Multivariate Analysis and Statistical Inference. doi:10.1016/j.jspi.2006.12.009.

[4] S. Sugasawa, K. Shimizu, and S. Kato. *A flexible family of distributions on the cylinder*. ArXiv e-prints, January 2015. arXiv:1501.06332.

[5] M. Wang. *Extensions of Probability Distributions on Torus, Cylinder and Disc*. PhD thesis, Keio University, 2013.

[6] J. J. Fernández-Durán. *Models for Circular-Linear and Circular-Circular Data Constructed from Circular Distributions Based on Nonnegative Trigonometric Sums*. Biometrics, 63(2):579–585, 2007.

[7] E. García-Portugués, R. M. Crujeiras, and W. González-Manteiga. *Exploring wind direction and SO2 concentration by circular–linear density estimation*. Stochastic Environmental Research and Risk Assessment, 27(5):1055–1067, Jul 2013. doi:10.1007/s00477-012-0642-5.

[8] J. A. Carnicero, C. Ausín, and M. Wiper. *Non-parametric copulas for circular-linear and circular-circular data: An application to wind directions*. Stochastic Environmental Research and Risk Assessment, 27:1991–2002, 12 2013.

[9] T. Abe and C. Ley. *A tractable, parsimonious and flexible model for cylindrical data, with applications*. Econometrics and Statistics, 4(Supplement C):91 – 104, 2017. doi:10.1016/j.ecosta.2016.04.001.

[10] T. Abe and A. Pewsey. *Sine-skewed circular distributions*. Statistical Papers, 52(3):683–707, Aug 2011. doi:10.1007/s00362-009-0277-x.

[11] F. Lagona, M. Picone, and A. Maruotti. *A hidden Markov model for the analysis of cylindrical time series*. Environmetrics, 26(8):534–544, 2015. env.2355. doi:10.1002/env.2355.

[12] F. Lagona and M. Picone. *Model-based segmentation of spatial cylindrical data*. Journal of Statistical Computation and Simulation, 86(13):2598–2610, 2016. doi:10.1080/00949655.2015.1122791.

[13] M. Ranalli, F. Lagona, M. Picone, and E. Zambianchi. *Segmentation of sea current fields by cylindrical hidden Markov models: a composite likelihood approach*. Journal of the Royal Statistical Society: Series C (Applied Statistics). doi:10.1111/rssc.12240.

[14] J. Bulla, F. Lagona, A. Maruotti, and M. Picone. *A Multivariate Hidden Markov Model for the Identification of Sea Regimes from Incomplete Skewed and Circular Time Series*. Journal of Agricultural, Biological, and Environmental Statistics, 17(4):544–567, Dec 2012. doi:10.1007/s13253-012-0110-1.

[15] Y. Tang, J. Fu, W. Liu, and A. Xu. *Bayesian analysis of repairable systems with modulated power law process*. Applied Mathematical Modelling, 44(Supplement C):357 – 373, 2017. doi:10.1016/j.apm.2017.01.067.

[16] S. Ali. *Mixture of the inverse Rayleigh distribution: Properties and estimation in a Bayesian framework*. Applied Mathematical Modelling, 39(2):515 – 530, 2015. doi:10.1016/j.apm.2014.05.039.

[17] R. McVinish and K. Mengersen. *Semiparametric Bayesian circular statistics*. Computational Statistics & Data Analysis, 52(10):4722 – 4730, 2008. doi:10.1016/j.csda.2008.03.016.

[18] G. Nuñez-Antonio, E. Gutiérrez-Peña, and G. Escarela. *A Bayesian regression model for circular data based on the projected normal distribution*. Statistical Modelling, 11(3):185–201, 2011. doi:10.1177/1471082X1001100301.

[19] S. Frühwirth-Schnatter. *Markov Chain Monte Carlo Estimation of Classical and Dynamic Switching and Mixture Models*. Journal of the American Statistical Association, 96(453):194–209, 2001.

[20] J. Diebolt and C. P. Robert. *Estimation of Finite Mixture Distributions through Bayesian Sampling*. Journal of the Royal Statistical Society. Series B (Methodological), 56(2):363–375, 1994.

[21] G. O. Roberts, A. Gelman, and W. R. Gilks. *Weak convergence and optimal scaling of random walk Metropolis algorithms*. Ann. Appl. Probab., 7(1):110–120, 02 1997. doi:10.1214/aoap/1034625254.

[22] H. Haario, E. Saksman, and J. Tamminen. *An adaptive Metropolis algorithm*. Bernoulli, 7(2):223–242, 04 2001.

[23] S. Richardson and P. J. Green. *On Bayesian Analysis of Mixtures with an Unknown Number of Components (with discussion).* Journal of the Royal Statistical Society: Series B (Statistical Methodology), 59(4):731–792, 1997. doi:10.1111/1467-9868.00095.

[24] M. Stephens. *Dealing with label switching in mixture models.* Journal of the Royal Statistical Society: Series B (Statistical Methodology), 62(4):795–809, 2000.

[25] J.-M. Marin, K. Mengersen, and C. P. Robert. *Bayesian Modelling and Inference on Mixtures of Distributions.* Handbook of Statistics, 25:459 – 507, 2005. doi:10.1016/S0169-7161(05)25016-2.

[26] J.-M. Marin and C. Robert. *Bayesian core: a practical approach to computational Bayesian statistics.* Springer Science & Business Media, 2007.

[27] P. Papastamoulis and G. Iliopoulos. *An Artificial Allocations Based Solution to the Label Switching Problem in Bayesian Analysis of Mixtures of Distributions.* Journal of Computational and Graphical Statistics, 19(2):313–331, 2010. doi:10.1198/jcgs.2010.09008.

[28] C. E. Rodríguez and S. G. Walker. *Label Switching in Bayesian Mixture Models: Deterministic Relabeling Strategies.* Journal of Computational and Graphical Statistics, 23(1):25–45, 2014. doi:10.1080/10618600.2012.735624.

[29] M. Sperrin, T. Jaki, and E. Wit. *Probabilistic relabelling strategies for the label switching problem in Bayesian mixture models.* Statistics and Computing, 20(3):357–366, Jul 2010. doi:10.1007/s11222-009-9129-8.

[30] P. Papastamoulis. *label.switching: An R Package for Dealing with the Label Switching Problem in MCMC Outputs.* Journal of Statistical Software, Code Snippets, 69(1):1–24, 2016. doi:10.18637/jss.v069.c01.

[31] S. Chib. *Marginal Likelihood from the Gibbs Output.* Journal of the American Statistical Association, 90(432):1313–1321, 1995.

[32] S. Chib and I. Jeliazkov. *Marginal Likelihood from the Metropolis-Hastings Output.* Journal of the American Statistical Association, 96(453):270–281, 2001.

[33] P. J. Green. *Reversible jump Markov chain Monte Carlo computation and Bayesian model determination.* Biometrika, 82(4):711–732, 1995.

[34] J. Piironen and A. Vehtari. *Comparison of Bayesian predictive methods for model selection.* Statistics and Computing, 27(3):711–735, May 2017. doi:10.1007/s11222-016-9649-y.

[35] A. Gelman, J. Hwang, and A. Vehtari. *Understanding predictive information criteria for Bayesian models*. Statistics and Computing, 24(6):997–1016, Nov 2014. doi:10.1007/s11222-013-9416-2.

[36] A. Vehtari, A. Gelman, and J. Gabry. *Practical Bayesian model evaluation using leave-one-out cross-validation and WAIC*. Statistics and Computing, 27(5):1413–1432, Sep 2017. doi:10.1007/s11222-016-9696-4.

[37] J. Brady and M. O'Mahony. *Modelling charging profiles of electric vehicles based on real-world electric vehicle charging data*. Sustainable Cities and Society, 26:203 – 216, 2016. doi:10.1016/j.scs.2016.06.014.

[38] Y. B. Khoo, C.-H. Wang, P. Paevere, and A. Higgins. *Statistical modeling of Electric Vehicle electricity consumption in the Victorian EV Trial, Australia*. Transportation Research Part D: Transport and Environment, 32:263 – 277, 2014. doi:10.1016/j.trd.2014.08.017.

[39] C. Develder, N. Sadeghianpourhamami, M. Strobbe, and N. Refa. *Quantifying flexibility in EV charging as DR potential: Analysis of two real-world data sets*. In Proc. 7th IEEE Int. Conf. Smart Grid Communications (SmartGridComm 2016), pages 600–605, Sydney, Australia, 6–9 Nov. 2016. doi:10.1109/SmartGridComm.2016.7778827.

[40] N. Sadeghianpourhamami, N. Refa, M. Strobbe, and C. Develder. *Quantitive analysis of electric vehicle flexibility: A data-driven approach*. International Journal of Electrical Power and Energy Systems, 95:451 – 462, 2018. doi:10.1016/j.ijepes.2017.09.007.

5
Definition and Evaluation of Model-Free Coordination of Electrical Vehicle Charging with Reinforcement Learning

As established in the introductory chapter, a practical demand response should be based on realistic models of the problem and should be able to generalize to various scenarios of similar characteristics. In Chapters 2-4 we focused on analysis of the energy consumption flexibility and developing generative models to pave the way for such realistic and accurate models for use in demand response algorithms. In this chapter, we pursue a different direction and instead, propose a model-free demand response algorithm for coordinating the charging of a collection of electric vehicles. The proposed approach does not require accurate models of the environment and generalizes to coordinating the charging of various number of electric vehicles.

⋆⋆⋆

N. Sadeghianpourhamami, J. Deleu, and C. Develder.

Submitted to IEEE Transactions on Smart Grid , Sep. 2018

Abstract With the envisioned growth in deployment of electric vehicles (EVs), managing the joint load from EV charging stations through demand response (DR) approaches becomes more critical. Initial DR studies mainly adopt model predictive control and thus require accurate models of the control problem (e.g., a customer behavior model), which are to a large extent uncertain for the EV scenario. Hence, model-free approaches, especially based on reinforcement learning (RL) are an attractive alternative. In this paper, we propose a new Markov decision process (MDP) formulation in the RL framework, to jointly coordinate a set of EV charging stations. State-of-the-art algorithms either focus on a single EV, or perform the control of an aggregate of EVs in multiple steps (e.g., aggregate load decisions in one step, then a step translating the aggregate decision to individual connected EVs). On the contrary, we propose an RL approach to jointly control the whole set of EVs at once. We contribute a new MDP formulation, with a scalable state representation that is independent of the number of EV charging stations. Further, we use a batch reinforcement learning algorithm, i.e., an instance of fitted Q-iteration, to learn the optimal charging policy. We analyze its performance using simulation experiments based on a real-world EV charging data. More specifically, we (i) explore the various settings in training the RL policy (e.g., duration of the period with training data), (ii) compare its performance to an oracle all–knowing benchmark (which provides an upper bound for performance, relying on information that is not available or at least imperfect in practice), (iii) analyze performance over time, over the course of a full year to evaluate possible performance fluctuations (e.g, across different seasons), and (iv) demonstrate the generalization capacity of a learned control policy to larger sets of charging stations.

5.1 Introduction

Demand response (DR) algorithms aim to coordinate the energy consumption of customers in a smart grid to ensure demand-supply balance and reliable network performance. In initial DR studies, the demand response problem usually is cast as a model predictive control (MPC) approach (e.g., [1, 2]), typically formulated as an optimization problem to minimize the customer's electricity bill or maximize the energy provider's profit, subject to various operating constraints (e.g., physical characteristics of the devices, customer preferences, distributed energy resource constraints and energy market constraints). However, the widespread deployment of such model-based DR algorithms in the smart grid is limited for the following reasons: (i) heterogeneity of the end user loads, difference in user behavioral patterns and uncertainty surrounding their behavior makes the modeling task very challenging [3]; (ii) model-based DR algorithms are difficult to transfer from one scenario to the other, since the model designed for one group of users or applications is likely to require customization/tweaking for application to different groups.

Recently, reinforcement learning (RL) has emerged to facilitate model-free control for coordinating the user flexibility in DR algorithms. In RL-based approaches, the DR problem is defined in the form of a Markov decision process (MDP). A coordinating agent interacts with the environment (i.e., DR participating customers, energy providers, energy market prices, etc.) and takes control actions while aiming to maximize the long term expected reward (or minimize the long term expected cost). In other words, the agent learns by taking actions and observing the outcomes (i.e., states) and the rewards/costs in an iterative process. The DR objective (e.g., load flattening, load balancing) is achieved by appropriately designing the reward/cost signal. Hence, reinforcement learning based approaches do not need an explicit model of user flexibility behavior or the energy pricing information a priori. This facilitates more practical and generally applicable DR schemes compared to model-based approaches.

One of the main challenges of RL-based DR approaches is the curse of dimensionality due to the continuity and scale of the state and the action spaces: this hinders the applicability of RL-based DR for large-scale problems. In this paper, we focus on formulating a scalable RL-based DR algorithm to coordinate the charging of a group of electric vehicle (EV) charging stations, which generalizes to various group sizes and EV charging rates. In fact, current literature only offers a limited amount of model-free solutions for jointly coordinating the charging of multiple EV charging stations, as surveyed briefly in Section 5.2.

Such existing RL-based DR solutions are either developed for an individual EV or need a heuristic (which does not guarantee an optimum solution) to obtain the aggregate load of multiple EV charging stations during the learning process. Indeed, a scalable Markov decision process (MDP) formulation that generalizes to a collection of EV charging stations with different characteristics (e.g., charging rates, size) does not exist in current literature. In this paper we take the first step to fill this gap by proposing an MDP and explore its performance in simulation experiments. Note that the model we present is a further refinement of our initially proposed state and action representation listed in [4] (which did not consider sizable experimental results yet, and merely proposed a first MDP formulation). More precisely, in this paper:

- We define a new MDP with compact state and action space representations, in the sense that they do *not* linearly scale with the number of EV charging stations (thus EVs), they can generalize to collections of various sizes and they can be extended to cope with heterogeneous charging rates (see Section 5.3),

- We adopt batch reinforcement learning (fitted Q-iteration [5]) with function approximation to find the best EV charging policy (see Section 5.4),

- We quantitatively explore the performance of the proposed reinforcement learning approach, through simulations using real-world data to run experiments covering 10 and 50 charging stations (using the setup detailed in Section 5.5), answering the following research questions (see Section 5.6):

> **(Q1)** What are appropriate parameter settings[1] of the input training data?
>
> **(Q2)** How does the RL policy perform compare to an optimal all-knowing oracle algorithm?
>
> **(Q3)** How does that performance vary over time (i.e., from one month to the next) using realistic data?
>
> **(Q4)** Does a learned approach generalize to different EV group sizes?

We summarize our conclusions and list open issues to be addressed in future work in Section 5.7.

5.2 Related Work

With growing EV adoption, also the amount of available (and realistic) EV data increased. Hence, data-driven approaches to coordinate EV charging gained attention, with reinforcement learning (RL) as a notable example. For example, Shi *et al.* [6] adopt an RL-based approach and phrase an MDP to learn to control the charging and discharging of an *individual EV* under price uncertainty for providing vehicle-to-grid (V2G) services. Their MDP has (i) a state space based on the hourly electricity price, state-of-charge and time left till departure), (ii) an action space to decide between charging (either to fulfill the demand or provide frequency regulation), delaying the charging and discharging for frequency regulation[2], and (iii) unknown state transition probabilities. The reward is defined as the energy payment of charging and discharging or the capacity payment (for the provided frequency regulation service). Chis *et al.* [7] use batch RL to learn the charging policy of again an *individual EV*, to reduce the long-term electricity costs for the EV owner. An MDP framework is used to represent this problem, where (i) the state space consists of timing variables, minimum charging price for a current day and price fluctuation between the current and the next day, while (ii) the action is the amount of energy to consume in a day. Cost savings of 10%-50% are reported for simulations using real-world pricing data. Opposed to these cost-minimizing approaches assuming time-varying prices, as a first case study for our joint control

[1] The parameters of interest are (i) time span of the training data, and (ii) number of sampled trajectories from the decision trees. For details see Section 5.4.2 and Section 5.5.2.

[2] Frequency regulation is a so-called ancillary service for the power grid, and entails actions to keep the frequency of the alternating current grid within tight bounds, by instantaneous adjustments to balance generation and demand.

of a group of EV charging stations, we will focus first on a load flattening scenario (i.e., electricity prices are assumed constant, but peaks need to be avoided).

In contrast to [6] and [7], which consider the charging of a single EV, Claessens *et al.* [8] use batch RL to learn a collective charging plan for a *group of EVs* in the optimization step of their previously proposed three step DR approach [9]. Their three step DR approach constitutes an aggregation step, an optimization step, and a real-time control step. In the aggregation step, individual EV constraints are aggregated. In the optimization step, the aggregated constraints are used by the batch RL agent to learn the collective charging policy for the EV fleet, which is translated to a sequence of actions (i.e., aggregated power consumption values for each decision slot) to minimize energy supply costs. Finally, in the real-time control step a priority based heuristic algorithm is used dispatch the energy corresponding to the action determined in the optimization step from the individual EVs. Vandael *et al.* [10] also use batch RL to learn a cost-effective day-ahead consumption plan for a *group of EVs*. Their formulation has two decision phases, (i) day-ahead and (ii) intra-day. In the first decision phase, the aggregator predicts the energy required for charging its EVs for the next day, and purchases this amount in the day-ahead market. This first decision phase is modeled as an MDP. In the second decision phase, the aggregator communicates with the EVs to control their charging, based on the amount of energy purchased in the day-ahead market. The amount of the energy to be consumed by each connected EV is calculated using a heuristic priority-based algorithm and is communicated to the respective EV. The local decision making process by each EV is modeled using an MDP where the state space is represented by the charged energy of the EV, the action space is defined by charging power and the reward function is based on the deviations from the requested charging power. The fitted Q-iteration (FQI) algorithm is used to obtain the best policy.

Note that our work is different from [8] and [10] in two aspects: (i) unlike [8] and [10], our proposed approach does not take the control decisions in separate steps (i.e., taking aggregate energy consumption in one step and coordinating individual EV charging in a second step to meet the already decided energy consumption) and instead it takes decisions directly and jointly for all individual EVs using an efficient representation of an aggregate state of a group of EVs, hence (ii) our approach does not need a heuristic algorithm, but instead learns the aggregate load while finding an optimum policy to flatten the load curve. We now describe our MDP model, and subsequently the batch reinforcement learning approach to train it.

Table 5.1: Nomenclature

s	State
s'	The next state from s
Δt^{depart}	Time left until departure
Δt^{charge}	Time needed for charging completion
Δt^{flex}	Flexibility (time charging can be delayed)
N_s	Number of connected EVs in state s
$\mathcal{V}_t$	Set of EVs in the system at time t
$\mathbf{x}_s$	Aggregate demand in state s
t	Timeslot
Δt^{slot}	Duration of a decision slot
S_{max}	Maximum number of decision slots
H_{max}	Maximum connection time
N_{max}	Number of charging stations jointly being coordinated
$\mathbf{u}_s$	Action taken in state s
$\mathbf{U}_s$	Set of possible actions from state s
$\mathbf{x}_s^{\text{total}}(d)$	Total number of EVs on the d^{th} upper diagonal of $\mathbf{x}_s$
C^{demand}	Cost of total power consumption
C^{penalty}	Penalty cost for unfinished charging
$C(s, \mathbf{u}_s, s')$	Instantaneous cost of state transition
$\mathcal{B}^{\text{test}}$	Test set
$\mathcal{B}^{\text{train}}$	Training set
Δt	Training data time span
C_π	Normalized cost of policy π
C_{BAU}	Normalized cost of business-as-usual policy
C_{RL}	Normalized cost of the learned policy
C_{opt}	Normalized cost of optimum solution

algorithm 4: Binning algorithm for creating the aggregate state representation.

```
Input  : $\mathcal{V}_t = \{(\Delta t_1^{\text{depart}}, \Delta t_1^{\text{charge}}), \ldots, (\Delta t_{N_s}^{\text{depart}}, \Delta t_{N_s}^{\text{charge}})\}$
Output: Aggregate state $\mathbf{x}_s$, matrix of size $S_{\text{max}} \times S_{\text{max}}$
Initialize $\mathbf{x}_s$ with zeros
foreach $n = 1, \ldots, N_s$ do
    // count number of EVs in each $(i, j)$ bin
    $i = \left\lceil \frac{\Delta t_n^{\text{depart}}}{\Delta t^{\text{slot}}} \right\rceil$
    $j = \left\lceil \frac{\Delta t_n^{\text{charge}}}{\Delta t^{\text{slot}}} \right\rceil$
    $\mathbf{x}_s(i, j) \leftarrow \mathbf{x}_s(i, j) + 1$
return $\mathbf{x}_s / N_{max}$
```

5.3 Markov Decision Process

The high-level goal of the proposed EV charging approach is to minimize the long term cost of charging a group of EVs for an aggregator in a real-time decision-making scenario. In this paper, we focus on the scenario of load flattening (i.e., more advanced DR objectives are left for future work): we aim to minimize the peak-to-average ratio of the aggregate load curve of a group of EVs. Technically, we adopt a convex cost function that sums the squares of the total consumption over all timeslots within the decision time horizon. We regard this problem as a sequential decision making problem and formulate it using an MDP with unknown transition probabilities.

5.3.1 State Space

An EV charging session is characterized by: (i) EV arrival time, (ii) time left till departure (Δt^{depart}), (iii) requested energy and (iv) EV charging rate. We translate the requested energy to time needed to complete the charging (Δt^{charge}), implicitly assuming the same charging rates for all the EVs in a group. Thus, if we have N_s electric vehicles in the system, the (remaining times of) their sessions are represented as a set

$$\mathcal{V}_t = \{(\Delta t_1^{\text{depart}}, \Delta t_1^{\text{charge}}), \ldots, (\Delta t_{N_s}^{\text{depart}}, \Delta t_{N_s}^{\text{charge}})\}.$$

Note that we do not assume a priori knowledge of future arrivals, and hence do not include the arrival time to characterize the (present) EVs.

Each state s is represented using two variables: timeslot (i.e., $t \in \{1, \ldots, S_{\text{max}}\}$) and the aggregate demand (i.e., $\mathbf{x}_s$), hence $s = (t, \mathbf{x}_s)$. Inspired by [11], aggregate demand at each given timeslot is obtained via a binning algorithm (i.e., Algorithm 4) and is represented using a 2D grid, thus a matrix, with one axis representing Δt^{depart}, the other Δt^{charge}. As time progresses, cars will move towards lower

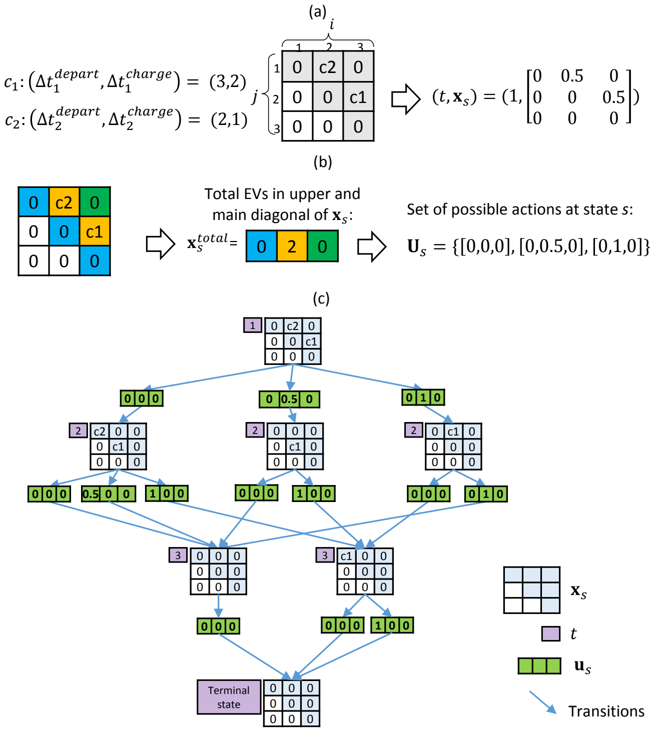

(a)
c_1: $(\Delta t_1^{depart}, \Delta t_1^{charge}) = (3,2)$
c_2: $(\Delta t_2^{depart}, \Delta t_2^{charge}) = (2,1)$
i
j
$(t, \mathbf{x}_s) = (1, \begin{bmatrix} 0 & 0.5 & 0 \\ 0 & 0 & 0.5 \\ 0 & 0 & 0 \end{bmatrix})$
(b)
Total EVs in upper and main diagonal of $\mathbf{x}_s$:
$\mathbf{x}_s^{total}$ = 0 2 0
Set of possible actions at state s:
$\mathbf{U}_s$ = {[0,0,0], [0,0.5,0], [0,1,0]}
(c)
Terminal state
$\mathbf{x}_s$
t
$\mathbf{u}_s$
Transitions

Figure 5.1: A simple example for $N_{\max} = 2$ charging stations: (a) state representation, (b) possible action states, (c) full decision tree over the horizon of $S_{\max} = 3$ slots.

Δt^{depart} cells, and (if charged) lower Δt^{charge} and Δt^{depart}.[3] Given that time-of-day is likely to influence the expected evolution of the state $\mathbf{x}_s$ (and hence the required response action we should take), we do include the timeslot t as explicit part of the state.

Formally, the process to convert the set of sessions $\mathcal{V}_t$ (associated with EVs connected at a given time t) to the matrix $\mathbf{x}_s$ is given by Algorithm 4. The size of the matrix, $S_{\max} \times S_{\max}$ depends on the maximal connection time $H_{\max}$, i.e., the longest duration of an EV being connected to a charging station: $S_{\max} \triangleq H_{\max}/\Delta t^{\text{slot}}$.

Each row/column of $\mathbf{x}_s$ represents equidistant bins with edges on $\{0, \Delta t^{\text{slot}}, 2 \cdot \Delta t^{\text{slot}}, \ldots, S_{\max} \cdot \Delta t^{\text{slot}}\}$ and each matrix element in $\mathbf{x}_s$ represents the number of EVs binned into it. $\mathbf{x}_s$ is initialized with zeros at the beginning of Algorithm 4. Lines 24–25 count the EVs with Δt^{depart} and Δt^{charge} values of the corresponding

[3] An extension to consider the variable charging rate is possible by binning the EVs in a 3D grid with charging rate as the third dimension.

(i, j)-cell in the matrix. Finally, $\mathbf{x}_s$ is normalized by N_{max} (Line 26). This normalization makes the state representation scale-free, i.e., independent of the absolute group size N_{max}, thus aiming to generalize the formulated MDP (and the learned control policy) to a differently sized group of EV charging stations.

For illustrative purposes, in Figure 5.1 we sketch a simple scenario of $N_{\text{max}} = 2$ charging stations with a horizon of $S_{\text{max}} = 3$ slots. Let us assume that at time $t = 1$ we have $N_s = 2$ connecting cars: $\mathcal{V}_1 = \{(\Delta t_1^{\text{depart}}, \Delta t_1^{\text{charge}}) = (3, 2), (\Delta t_2^{\text{depart}}, \Delta t_2^{\text{charge}}) = (2, 1)\}$, with no other arrivals during the control horizon. Figure 5.1 illustrates the resulting state space using the binning algorithm at the first time slot. The EVs are binned according to their Δt^{depart} and Δt^{charge} to a 2D grid of size 3×3. The resulting matrix is normalized by N_{max} ($= 2$ in this example). The shaded cells in the 2D grid of Figure 5.1 indicate bins with $\Delta t^{\text{charge}} \leq \Delta t^{\text{depart}}$. EVs in these bins have enough time to complete their charging.

Note that $\mathbf{x}_s$ not only summarizes the aggregated demand of connecting EVs (in terms of Δt^{depart} and Δt^{charge}), but also the flexibility in terms of how long the charging can be delayed at state s (denoted as $\Delta t^{\text{flex}} = \Delta t^{\text{depart}} - \Delta t^{\text{charge}}$) is inferred from the diagonals of $\mathbf{x}_s$ using

$$\Delta t^{\text{flex}}(i, j) = j - i \qquad \forall i, j \in \{1, \ldots, S_{max}\} \tag{5.1}$$

Equation (5.1) indicates that EVs binned into cells on the main diagonal of $\mathbf{x}_s$ (i.e., $i = j$) have zero flexibility while the ones binned into cells on the upper diagonals of $\mathbf{x}_s$ are flexible charging requests. Negative Δt^{flex}, corresponding to lower diagonals of $\mathbf{x}_s$ (i.e., the white cells in the 2D grids of Figure 5.1), indicates EVs for which the requested charging demand cannot be fulfilled. In our formulation, we will ensure that EVs charging demands are never violated, using a penalty term in our cost function (see Section 5.3.3).

Finally, the size of $\mathbf{x}_s$ and hence the size of the state s is independent of N_{max} and is only influenced by S_{max}, thus H_{max} and Δt^{slot}. This ensures scalability of the state representation to various group sizes of EV charging stations: the maximal number of cars N_{max} does not impact the state size.

5.3.2 Action Space

The action to take in state s is a decision whether (or not) to charge the connecting EVs with same Δt^{flex} in the $\mathbf{x}_s$ matrix. Such EVs are binned into the cells on the same diagonal of $\mathbf{x}_s$ as explained in the previous section. We indicate each diagonal of $\mathbf{x}_s$ as $\mathbf{x}_s(d)$ with $d = 0, \ldots, S_{\text{max}} - 1$ where $\mathbf{x}_s(0)$ is the main diagonal, $\mathbf{x}_s(d)$ is the upper d^{th} diagonal, and $\mathbf{x}_s(-d)$ is the lower d^{th} diagonal of $\mathbf{x}_s$. We denote $\mathbf{x}_s^{\text{total}}(d)$ as the total number of EVs in the cells on the d^{th} diagonal.

An action taken in state s is defined as a vector $\mathbf{u}_s$ of length S_{max}. For each individual car, we take a discrete action, i.e., we either charge it at full power or not at all for the next timeslot. This results in the element d of the action vector $\mathbf{u}_s$ being a number between 0 and 1: it amounts to charging the fraction of EVs in the corresponding d^{th} diagonal of $\mathbf{x}_s$. The set of possible actions from state s is denoted as $\mathbf{U}_s$.

Figure 5.1(b) illustrates how $\mathbf{U}_s$ is constructed at state s using a color-coded representation of matrix $\mathbf{x}_s$ and the corresponding vector $\mathbf{x}_s^{\text{total}}$. Note that we define the action vector for charging/delaying the cars on the main and upper diagonals of $\mathbf{x}_s$ only (colored cells in the 2D grids representing $\mathbf{x}_s$ in Figure 5.1). This is a design choice to keep the action space relatively small and therefore easier to explore. In the next section, we define our cost function such that the EV charging is always completed before departure: no cars will end up in any of the lower diagonals, i.e., the white cells in the 2D state grid of the figures.

5.3.3 Cost function

The goal we envision in this paper is to flatten the aggregate charging load of a group of EVs while ensuring each EV's charging is completed before departure.[4] Hence, our cost function associated with each state transition $(s, \mathbf{u}_s, s')$ has two parts:

(1) $C^{\text{demand}}(\mathbf{x}_s, \mathbf{u}_s)$: the cost of the total power consumption from all the connected EVs for a decision slot, and

(2) $C^{\text{penalty}}(\mathbf{x}_s, \mathbf{u}_s)$: the penalty for unfinished charging.

To achieve the load flattening objective, we choose the C^{demand} to be a quadratic function of the total power consumption for a decision slot. The total power consumption for a decision slot is proportional to the number of EVs being charged, since we assume the same charging rate for all the EVs in a group. Hence, the first term of the cost function at state $s = (t, \mathbf{x}_s)$ is defined as

$$C^{\text{demand}}(\mathbf{x}_s, \mathbf{u}_s) = \left(\sum_{d=0}^{S_{max}-1} \mathbf{x}_s^{\text{total}}(d)\, \mathbf{u}_s(d) \right)^2 \tag{5.2}$$

The second term of the cost function is a penalty proportional to the unfinished charging in the next state $s' = (t_{s'}, \mathbf{x}_{s'})$ due to taking action $\mathbf{u}_s$ in $s = (t, \mathbf{x}_s)$ and is defined as

$$C^{\text{penalty}}(\mathbf{x}_s, \mathbf{u}_s) = M \sum_{n \in \mathcal{V}_{t+1}} |\min(0, \Delta t_n^{\text{charge}} - \Delta t_n^{\text{depart}})| \tag{5.3}$$

The summation in (5.3) counts the amount of charging request that is impossible to complete (for EVs with $\Delta t_n^{\text{depart}} < \Delta t_n^{charge}$) as a consequence of taking action $\mathbf{u}_s$

[4] We assume only feasible requests are presented to the system, i.e., $\Delta t^{\text{charge}} \leq \Delta t^{\text{depart}}$ for each EV.

at state $s = (t, \mathbf{x}_s)$. M is a constant penalty factor, which we set to be greater than $2N_{\text{max}}$ to ensure that any EV's charging is always completed before departing (i.e., one incomplete EV is costlier than charging all EVs simultaneously). Summing (5.2) and (5.3), the total cost associated with each state transition $(s, \mathbf{u}_s, s')$ is:

$$C(s, \mathbf{u}_s, s') = \left(\sum_{d=0}^{S_{max}-1} \mathbf{x}_s^{\text{total}}(d)\, \mathbf{u}_s(d) \right)^2 + M \sum_{n \in \mathcal{V}_{t+1}} |\min(0, \Delta t_n^{\text{charge}} - \Delta t_n^{\text{depart}})| \tag{5.4}$$

Note that in Equation (5.4) the cost is independent of the timeslot variable of the state space (i.e., t) and depends only on the aggregate demand variable of the state (i.e., $\mathbf{x}_s$). Indeed, the cost of a demand to be is set as a quadratic function of the total consumption to achieve the load flattening objective, and is time-independent. Still, we include the time component in the definition of the state to ensure that our formulations can easily be extended to other objectives (e.g., reducing the cost under the time-of-use or pricing schemes). Also, we use the time component for the function approximator of Algorithm 5 (see further, Section 5.5.2).

5.3.4 System Dynamics

In the MDP framework, system dynamics (via the environment) are defined using transition probabilities $P(s'|s, \mathbf{u}_s)$. The transition probabilities from one state s to the next s' are unknown in the EV group charging problem due to the stochasticity of the EV arrivals and their charging demands. Perfect knowledge of EV arrivals and their charging demands during the control horizon would translate the problem into a decision tree depicted in Figure 5.1(c), where the cost of taking each action can be determined recursively using dynamic programming. However, in absence of such knowledge, the transition probabilities need to be estimated through interactions with the environment by taking actions and observing the instantaneous cost of the resulting state transitions. The next section explains this approach.

5.3.5 Learning Objective: State-Action Value Function

Note that $C(s, \mathbf{u}_s, s')$ is the instantaneous cost an aggregator incurs when action $\mathbf{u}_s$ is taken at state $s = (t, \mathbf{x}_s)$ and leads to state $s' = (t+1, \mathbf{x}_{s'})$. The objective is to find an optimum control policy $\pi^* : \mathbf{S} \to \mathbf{U}$ that minimizes the expected T-step return for any state in $\mathbf{S}$. The expected T-step return starting from state $s = 1$ and following a policy π (i.e., $\mathbf{u}_s = \pi(s)$) is defined as:

$$J_T^{\pi}(1) = \mathbb{E}\left[\sum_{s=1}^{T} C(s, \mathbf{u}_s, s') \right] \tag{5.5}$$

algorithm 5: Fitted Q-iteration using function approximation for estimating the T-step return

```
Input : $\mathcal{F} = \{(s, \mathbf{u}_s, s', C(s, \mathbf{u}_s, s'))|s = 1, \dots, |\mathcal{F}|\}$
Initialize $\widehat{Q}_0$ to be zero everywhere on $\mathbf{X} \times \mathbf{U}$
foreach $N = 1, \dots, T$ do
    foreach $(s, \boldsymbol{u}_s, s', C(s, \boldsymbol{u}_s, s'))) \in \mathcal{F}$ do
        $Q_N(s, \mathbf{u}_s) \leftarrow C(s, \mathbf{u}_s, s') + \min_{\mathbf{u} \in \mathbf{U}} \widehat{Q}_{N-1}(s', \mathbf{u})$
    Use function approximator to obtain $\widehat{Q}_N$ from $\mathcal{T}_{reg} = \{((s, \mathbf{u}_s), Q_{N,s})|s = 1, \dots, |\mathcal{F}|\}$
return $\widehat{Q}_T$
```

The policy π is commonly characterized using a state-action value function (or Q-function):

$$Q^{\pi}(s, \mathbf{u}_s) = \mathbb{E}\left[C(s, \mathbf{u}_s, s') + J_T^{\pi}(s')\right] \tag{5.6}$$

where $Q^{\pi}(s, \mathbf{u}_s)$ is cumulative return starting from state s, taking action $\mathbf{u}_s$, and following policy π afterwards. The optimal $Q^{\pi}(s, \mathbf{u}_s)$, denoted as $Q^*(s, \mathbf{u}_s)$, corresponds to:

$$Q^*(s, \mathbf{u}_s) = \min_{\pi} Q^{\pi}(s, \mathbf{u}_s) \tag{5.7}$$

The $Q^*(s, \mathbf{u}_s)$ satisfies the Bellman equation:

$$Q^*(s, \mathbf{u}_s) = \min_{\mathbf{u} \in \mathbf{U}} \mathbb{E}\left[C(s, \mathbf{u}_s, s') + Q^*(s', \mathbf{u})\right] \tag{5.8}$$

However, solving (5.8) requires the knowledge of the transition probabilities — defining how the system moves from one state s to the next s' — which are unknown in our setting. Hence, a learning algorithm should be used to obtain approximation $\widehat{Q}^*(s, \mathbf{u})$. This can then be used to take control action $\mathbf{u}_s$, following:

$$\mathbf{u}_s \in \operatorname*{argmin}_{u \in \mathbf{U}_s} \widehat{Q}^*(s, \mathbf{u}) \tag{5.9}$$

5.4 Batch Reinforcement Learning

We adopt batch mode RL algorithms to approximate $\widehat{Q}^*(s, \mathbf{u})$ from past experience instead of online interactions with the environment. In the batch mode RL approach, data collection is decoupled from the optimization. In other words, we use the historical EV data (i.e., arrival/departures and energy demands) and a random policy to collect the experiences (i.e., the state transitions and the associate costs) in form of $(s, \mathbf{u}_s, s', C(s, \mathbf{u}_s, s'))$ tuples. We use Fitted Q-iteration to approximate $\widehat{Q}^*(s, \mathbf{u})$ from the collected tuples, detailed next.

5.4.1 Fitted Q-iteration

Fitted Q-iteration (FQI) is a batch mode RL algorithm, listed in Algorithm 5. As input, FQI takes a set of past experiences, $\mathcal{F}$, in the form of tuples $(s, \mathbf{u}_s, s', C(s, \mathbf{u}_s, s'))$ where $C(s, \mathbf{u}_s, s')$ is the immediate cost of a transition and in our case is calculated using Eq. (5.4). The tuples are used to iteratively estimate the optimum action value function.

The state-action value function Q is initialized with zeros on the state-action space (Line 27) hence, $Q_1 = C(s, \mathbf{u}_s, s')$ in the first iteration. In subsequent iterations, Q_N is calculated for each tuple in $\mathcal{F}$ using the latest approximation of action-value function (Q_{N-1}) from the previous iteration (Line 4) to form a labeled dataset $\mathcal{T}_{reg}$. This dataset is then used for regression, i.e., by function approximation we estimate Q_N for all possible state-action pairs (Line 30).

We will adopt a fully connected artificial neural network (ANN) as our function approximation. Further details on the ANN architecture used in our experiments are given in Section 5.5.2.2.

5.4.2 The size of state-action space

The input to FQI (i.e., set $\mathcal{F}$) is constructed from past interactions with the environment (i.e., randomly or deterministically taking actions from the action space of state $s = (t, \mathbf{x}_s)$ and recording the tuple $(s, \mathbf{u}_s, s', C(s, \mathbf{u}_s, s'))$. The number of all possible actions from a given state s is given by

$$|\mathbf{U}_s| = \prod_{d=1}^{S_{max}} \left(\mathbf{x}_s^{total}(d) + 1\right) \tag{5.10}$$

since for each flexibility $\Delta t^{\text{flex}} = d$ we can choose to charge between $[0, x_s^{\text{total}}(d)]$ cars.

The goal of the RL algorithm (hence the goal of the FQI) is to estimate the T-step return for every possible action from every possible state in the environment. Estimating the T-step return starting from a state s leads to exploring a tree with an exponentially growing number of branches at the next steps. Hence, while the state and action representations are independent of the group size (N_{max}), the state-action space still grows exponentially with a growth rate given in Eq. (5.10). Let us consider a charging lot of capacity $N_{\text{max}} = 50$ and control horizon with $S_{\text{max}} = 10$. In a state where all EV charging stations are occupied ($N_s = N_{\text{max}} = 50$), there are at least 51 possible actions from that state, corresponding to a scenario where all the EVs have similar flexibility, hence located on the same diagonal of the state matrix (i.e., $\mathbf{x}_s^{\text{total}}) = [50, 0, 0, 0, 0, 0, 0, 0, 0, 0]$. For a state with $\mathbf{x}_s^{\text{total}}) = [5, 5, 5, 5, 5, 5, 5, 5, 5, 5]$, there will be $|\mathbf{U}_s| = (6)^{10}$ possible actions from that state only. This indicates that it is not feasible to include the entire state-action space in set $\mathcal{F}$ as the input to the FQI and only a subset of the state-action space is provided. We will therefore randomly sample trajectories from the decision tree with a branching factor of $|\mathbf{U}_s|$. This leads to the research question **Q1** (which is answered in Section 5.6.1): How many sample trajectories from the state-action space are sufficient to learn an optimum policy for charging a real-world group of EVs with various group sizes?

5.5 Experiment setup

In this section, we outline the implementation details of the proposed RL-based DR approach.

5.5.1 Data Preparation

We base our analysis on real-world EV charging session transactions collected by ElaadNL since 2011 from 2500+ public charging stations deployed across Netherlands, as described and analyzed in [12]. For each of the over 2M charging sessions (still growing), a transaction records the charging station ID, arrival time, departure time, requested energy and charging rate during the session. The EVs in this dataset are privately owned cars and thus comprise a mixture of different and a priori unknown car types.

To represent the EV transactions in ElaadNL as state transitions $(s, \mathbf{u}_s, s',$ $C(s, \mathbf{u}_s, s'))$, we first need to choose a reasonable size for the state and the action representations. We set the maximum connection duration to $H_{max} = 24\,\mathrm{h}$, since more than 98% of the EV transactions in the ElaadNL dataset cover sessions of less than 24 hours [12].

We further set the duration of a decision timeslot, i.e., the time granularity of control actions (i.e., $\Delta t^{\mathrm{slot}} = 2\,\mathrm{h}$), resulting in $S_{\mathrm{max}} = H_{\mathrm{max}}/\Delta t^{\mathrm{slot}} = 12$. Hence, a state s is represented by a scalar variable t and a matrix $\mathbf{x}_s$ of size $S_{\mathrm{max}} \times S_{\mathrm{max}} = 12 \times 12$. The corresponding action $\mathbf{u}_s$ taken from state s is a vector of length 12 (with 1 decision for each of the upper diagonals, one per flexibility window Δt^{flex}). The motivation of choosing $\Delta t^{\mathrm{slot}} = 2\,\mathrm{h}$ is to limit the branching factor $|\mathbf{U}_s|$ (which depends on S_{max} in Eq. (5.10)) at each state, thus yielding a reasonable the state-action space size and allowing model training (specifically, the $\min$ operation in Line 4 of Algorithm 5) in a reasonable amount of time given our computation resources.[5]

Furthermore, we make the ElaadNL dataset episodic by assuming that all the EVs leave the charging stations before the end of a day, thus yielding an empty

[5] We use an Intel Xeon E5645 processor, 2.4 GHz, 290 GB RAM.

car park in between two consecutive days.[6] We define such an episodic 'day' to start at 7 am and end 24 h later (the day after at 7 am). The empty system state in between two episodes is always reached after $S_{\text{max}} + 1$ timeslots and is represented with aggregate demand matrix $\mathbf{x}_s$ of all zeros. This ensures that while each day has a different starting state (depending on the arrivals in the first control slot and their energy demand), traversing the decision tree always leads to a unique terminal state (see Figure 5.1(c) for an exemplary decision tree). This is motivated by Riedmiller [13], who shows that, when learning with FQI and adopting a neural network as function approximator, having a terminal goal state stabilizes the learning process. It ensures that all trajectories end up in a state where no further action/transition is possible and hence is characterized by an action-value of zero.

To create a group of N_{max} EV charging stations, we select the busiest N_{max} charging stations (based on the number of recorded transactions in each station). For the analysis in this paper, we use two different subsets, one with the top-10, the other with the top-50 most busiest stations.

5.5.2 Algorithm Settings

Since $S_{max} = 12$ in our settings, fitted Q-iteration (FQI) needs to estimate the 12-step return and we thus have 12 iterations in Algorithm 5 .

5.5.2.1 Creating set $\mathcal{F}$

To create set $\mathcal{F}$, we begin from the starting state of a day characterized by $(t_1, \mathbf{x}_1)$ and randomly choose an action from the set of possible actions in each state and observe the next state and the associated state transition cost until the terminal state[7] is reached (i.e., $(t_T, \mathbf{x}_T)$). The state transitions in each trajectory are recorded in the form of a tuple $(s, \mathbf{u}_s, s', C(s, \mathbf{u}_s, s'))$ in set $\mathcal{F}$. For our experiments, we randomly sample more than a single trajectory from each day to analyze the effect of the number of sampled trajectories on the performance of the proposed approach. The notion of a *sample* in the following thus refers to a full trajectory from initial to terminal state of a day.

5.5.2.2 Neural network architecture

We use an artificial neural network (ANN) that consists of an input layer, 2 hidden layers with ReLU activation function and an output layer. There are 128 and 64 neurons in the first and second hidden layers respectively. Since the ANN is used for regression, the output layer has a single neuron and a linear activation function.

[6] The charging demands of EVs are adjusted to ensure the requested charging can be fulfilled within 24 hours.

[7] Recall that we consider an episodic setting, i.e., case where the system empties (definitely after S_{max} timeslots).

Each state-action pair is fed to the input layer in form of a vector of length $S_{\text{max}}^2 + S_{\text{max}} + 1$, by reshaping the state (t, s_t) and concatenating it with the action vector $\mathbf{u}_s$ (of size $S_{\text{max}} = 12$). Recall that the state representation has a scalar time variable t and an aggregate demand matrix $\mathbf{x}_s$ of size $S_{\text{max}} \times S_{\text{max}}$, thus reshaped to a vector of size $S_{\text{max}}^2 + 1$. In our settings each state s thus is represented as a vector of length 145 and each action u_s as a vector of length 12. Inspired by Mnih *et al.* [14], we also found that using Huber loss [15] instead of mean-squared-error stabilizes the learning in our algorithm.

5.5.3 Performance Evaluation Measure

To evaluate the performance of the proposed approach, we take ElaadNL transactions of 2015 and select the last 3 months as the test set, i.e., $\mathcal{B}^{\text{test}} = \{e_i | i = 274, \ldots, 365\}$ containing $|\mathcal{B}^{\text{test}}| = 92$ days.

We consider training sets of varying lengths (to determine the impact of training set size, see research question Q1), with $\Delta t \in \{1, 3, 5, 7, 9\}$ months. For a given Δt (i.e., training data time span), we randomly pick 5 contiguous periods within the range of Jan. 1, 2015 until Sep. 30, 2015 (except for the case $\Delta t = 9$ months, since that covers the whole training data range). We define the training set for time span Δt and run j as $\mathcal{B}_{\Delta t,j}^{\text{train}} = \{e_i | i = e_{\Delta t,j}^{\text{start}}, \ldots, e_{\Delta t,j}^{\text{start}} + \Delta t - 1\}$, where $e_{\Delta t,j}^{\text{start}}$ is the randomly selected starting date of the training set.

To evaluate the performance of the learned policy, we define the metric of *normalized cost* relative to the cost of the optimum policy. For each Δt and j we define it as

$$C_{\pi(\Delta t_j)} = \frac{1}{|\mathcal{B}^{\text{test}}|} \sum_{e \in \mathcal{B}^{\text{test}}} \frac{C_{\pi(\Delta t_j)}^e}{C_{\text{opt}}^e}, \tag{5.11}$$

where $\pi(\Delta t_j)$ is a policy learned from the training data time span of Δt at run j. Further, $C_{\pi(\Delta t_j)}^e$ is the cost of day e under policy $\pi(\Delta t_j)$ and C_{opt}^e is the cost of day e using optimization (obtained from formulating the load flattening problem as a quadratic optimization problem). A cost of a day e under policy π is calculated by summing the instantaneous cost (defined by Eq. (5.4)) of state transitions encountered when taking action according to the policy being evaluated (using Eq. (5.9)).

Clearly, if a learned policy achieves the optimum policy, then $C_{\pi(\Delta t_j)} = 1$. Further, we compare the performance of the learned policy not only with the optimum policy but also with the business-as-usual (BAU) policy where the charging of an EV starts immediately upon arrival. In the next section, we present our analysis using the normalized cost of BAU, optimum and learned policies denoted as C_{BAU}, C_{opt} and C_{RL} respectively.

5.6 Experimental results

In this section, we present experiments answering the aforementioned research questions Q1–Q4. More specifically, we first evaluate the performance of the RL-based approach in coordinating the charging demand of $N_{\text{max}} = 10$ and 50 charging stations as a function of training data time span and number of randomly sampled trajectories per day (Q1), comparing it to an uncontrolled business-as-usual scenario but also to the optimum strategy (Q2). We then investigate how well the method works across various seasons, i.e., whether performance varies strongly throughout the whole year (Q3). Finally, we check the scalability by training an agent on a group of $N_{\text{max}} = 10$ EV charging stations and testing it on upscaled group sizes N_{max} (Q4).

5.6.1 Learning the Charging Coordination (Q1–Q2)

To answer **Q1** (i.e., what are appropriate training data time span and number of sampled trajectories from the decision trees?), we study how the performance of the proposed RL approach varies in function of (i) the time span covered by the training data (i.e., Δt), and (ii) the number of sample trajectories per day of training data. Figure 5.2 compares the normalized cost of a learned policy with that of a BAU and optimum policy for varying Δt and number of samples per training day, for the case of $N_{\text{max}} = 10$ and 50 charging stations respectively.

Influence of the time span covered by the training data: Figure 5.2(b) shows that increasing Δt from 1 month to 3 months and beyond reduces the normalized cost of the learned policy for both 10 and 50 charging stations. Additionally, the performance gain when increasing Δt from 1 to 3 months is bigger than for increasing Δt beyond 3 months. This suggests that the RL approach needs at least 3 months of training data to reach maximal performance (in case of ElaadNL).

Influence of the number of sample trajectories per day of training data: Figure 5.2(a) shows that when $\Delta t \geq 3$ months, increasing the number of samples does not result in significant reduction in normalized cost of the learned policy (i.e., C_{RL}) for both 10 and 50 charging stations.

The above analysis suggests that a training data time span of at least 3 months is needed to have a comparable performance over various number of samples per day and that when training data time span is at least 3 month long, smaller number of samples (of the order of 5K trajectories) can still achieve a comparable performance (with respect to training with larger samples per day). This answers **Q1**.

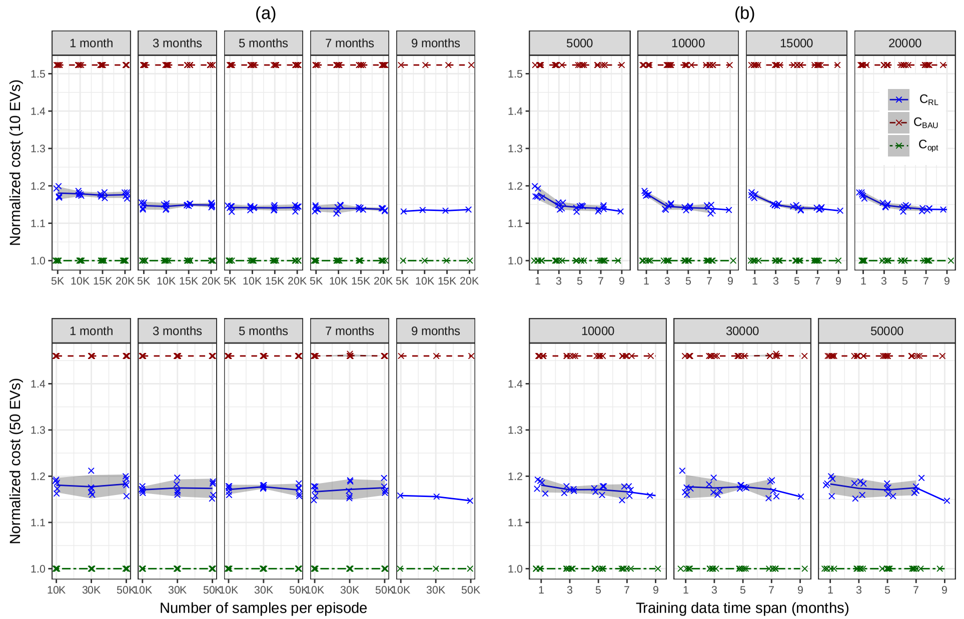


Figure 5.2: Normalized costs of learned policy (C_{RL}), BAU policy (C_{BAU}) and optimum solution (C_{opt}) for coordinating the charging of 10 (top row) and 50 (bottom row) EV charging stations, (a) normalized costs as a function of number of samples per day for various Δts, and (b) normalized costs as a function of Δt for various numbers of sample trajectories per training day.

Next, we answer **Q2** (i.e., how does the RL policy perform compare to an optimal all-knowing oracle algorithm?) by referring to the best performance measures in Figure 5.2, for coordinating 10 and 50 EV charging stations. We observe that the best performance is achieved when $\Delta t = 9$ months for both scenarios. The relative improvement in terms of reduction of normalized cost, compared to a business-as-usual uncontrolled charging scenario, C_{BAU}, amounts to 39% and 30.4% for 10 and 50 charging stations respectively. Note that C_{RL} is still 13% and 15.6% more expensive than the optimal policy cost C_{opt} (the optimal policy would achieve 52% reduction in cost with respect to C_{BAU}) for 10 and 50 charging stations respectively. Still, it is important to realize that to find the optimal policy, we assume perfect knowledge of future EV charging sessions, including arrival and departure times and the energy requirements. Clearly, having such complete knowledge of the future is not feasible in a real-world scenario: the proposed RL approach, which does not require such knowledge, thus is a more practical solution.

Finally, comparing the variance of the different runs (shaded regions in Figure 5.2 for 10 vs. 50 EV charging stations reveals that there is an increase in the variance between simulation runs when the group size is increased. Note that the same training horizons are used for both groups for a given Δt and simulation run. After observing the distributions of EV arrivals, EV departures and energy requirements, we conclude that high variability between the runs in Figure 5.2 does not stem from differences in the distributions among the various charging stations. We rather hypothesize that this increased performance variance among runs is caused by the fact that the state-action space for coordinating the charging of 50 cars is considerably bigger than the one of 10 cars, given Eq. (5.10). The performance of the fitted Q-iteration is indeed greatly influenced by the training set $\mathcal{F}$ at the input of the algorithm. With random sampling, there is no guarantee that most crucial parts of the state-action space (e.g., best and worst trajectories) will be included in the training set $\mathcal{F}$. With larger trees, such a possibility is even more limited. Re-exploration of the state-action space with a trained agent and retraining is one way to improve the performance. Efficient exploration of large state-action spaces is one of the active research domains in reinforcement learning and many algorithms are proposed to tackle the exploration problem (e.g., [16] and [17]). A summary of the exploration algorithms is presented in [18]. Such tackling of efficient exploration of the state-action space is left for future research.[8]

[8] As indicated previously, we limit this paper's focus to proposing the (scalable/generalizable) MDP formulation and experimentally exploring the resulting RL-based EV charging performance using realistic EV data.

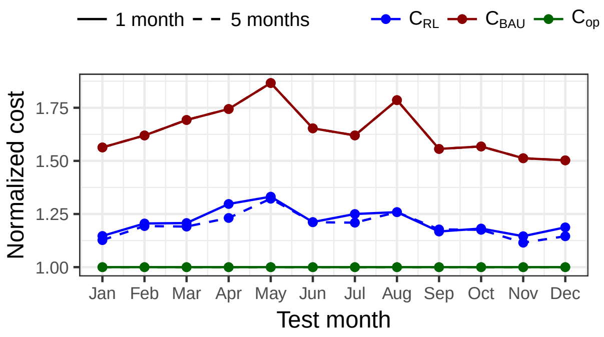


Figure 5.3: Performance using different months as test set and different time spans of the training set (1–5 months).

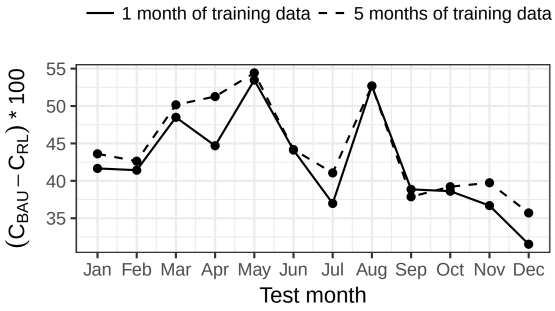


Figure 5.4: Improvement in normalized cost of the learned policy (RL) with respect to the business-as-usual policy (BAU).

5.6.2 Variance of performance over time (Q3)

In the analyses presented in Figure 5.2, the days in the last quarter of 2015 from the ElaadNL dataset were used to construct the test set. Now, we investigate whether changing the test set influences the performance of the learned policy, as to answer the question how performance of our RL approach would vary over time throughout the year. More specifically, we use each month of 2015 as a separate test set, using the preceding months as training data. We also vary the training data time span from 1 to 5 preceding months. Figure 5.3 shows the normalized costs for coordinating $N_s = 10$ charging stations. This is complemented in Figure 5.3 with the relative cost improvement compared to the business-as-usual scenario, C_{BAU}.

Figure 5.3 shows that C_{BAU} varies across the test months: for some months (e.g., May and Aug), the difference $C_{\text{BAU}} - C_{\text{opt}}$ is larger than for others. This indicates that the charging sessions in these test months have higher flexibility,

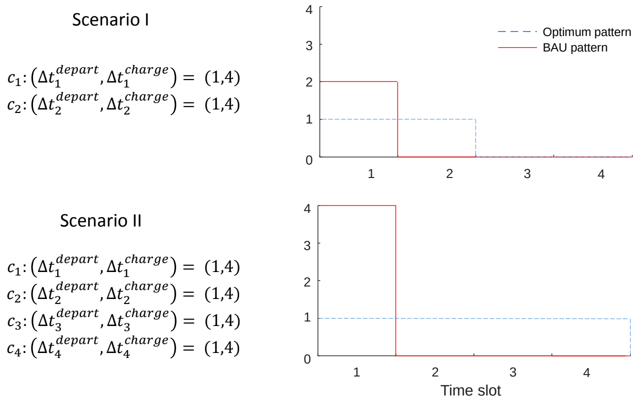


Figure 5.5: The effect of scaling up the group size on a normalized cost of a policy learned from 10 EV charging stations

which is exploited by the optimum solution. For such months with higher $C_{\text{BAU}} - C_{\text{opt}}$, our proposed RL approach also achieves a higher reduction in normalized cost compared to C_{BAU}, as seen in Figure 5.4. Still, the achieved C_{RL} is more expensive than C_{opt} compared to the months who offer less flexibility. We found that the days in which the optimal charging pattern requires the exploitation of larger charging delays are more challenging to learn by RL approach, in the sense that RL has greater difficulty in approaching the optimum (i.e., obtaining higher C_{RL}). One reason is the scarcity of such days in the training set, which results in imbalanced training data. Another reason is the random sampling of the large state action space, which does not guarantee inclusion of the scarce (but crucial) parts of the state-action space in the training set that is fed to the FQI algorithm.

We further investigate the effect of increasing the training data time span from 1 preceding month to 5 preceding months for each test set. We find that for majority of the months, this results in improvement with respect to C_{BAU} as depicted in Figure 5.4.

The analysis in this section reveals the following answer to **Q3** (i.e., How does the performance vary over time using realistic data?): the RL algorithm performance depends on the available flexibility, with greater flexibility (expectedly) leading to larger cost reductions compared to the BAU uncontrolled charging, but greater difficulty in approaching the optimum performance.

5.6.3 Generalization to Larger Scales (Q4)

While model-free approaches based on RL eliminate the need for accurate knowledge of the future EV session characteristics (as opposed to optimization based

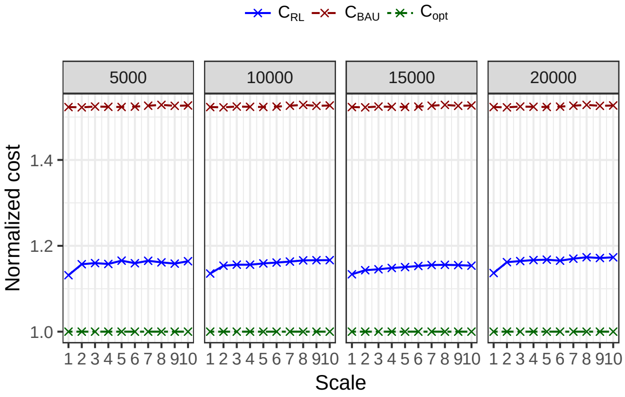


Figure 5.6: The effect of scaling up $N_{\max}$ on a normalized cost of a policy learned from $N_{\max} = 10$ EV charging stations for different of number of sampled trajectories (ranging from 5K to 20K).

approaches), they still require a reasonably long training time to be able to efficiently coordinate the EV charging sessions. The runtime for the largest training set size (covering 9 months, with 5K sample trajectories per day) is approximately 3 hours for 10 EV charging stations, while that of 50 charging stations is approximately 48 hours.[9]

Since our proposed formulations are independent of the number of EV charging stations ($N_{\max}$), it is interesting to investigate how a policy learned based on training with a small number of EV charging stations performs when applied to coordinate a larger group of stations. To do this, we use the policy learned from data of 10 EV charging stations with $\Delta t = 9$ months. We use the EV sessions in the last quarter of 2015 as our test set. To investigate the effect of the increase in the number of EV charging stations without changing other system characteristics, we duplicate the EV charging sessions by a factor *scale* to create a test set of larger $N_{\max}$.This still changes the optimum solution as illustrated with a simple example in Figure 5.5 where the length of the control horizon is $S_{\max} = 4$ slots. In Scenario I of Figure 5.5, at time $t = 1$ we have 2 connecting cars: $V = \{(\Delta t_1^{\text{depart}}, \Delta t_1^{\text{charge}}) = (1, 4), (\Delta t_2^{\text{depart}}, \Delta t_2^{\text{charge}}) = (1, 4)\}$ and no other arrivals during the control horizon. The best action is to charge 50% of the cars at $t = 1$ and 2 to flatten the load curve. In Scenario II of Figure 5.5, set V is duplicated once and the best action now is to charge 25% of the cars in each of the control timeslots.

The normalized costs (i.e., relative to the optimum C_{opt}) of the learned policy for scaled-up group sizes are shown in Figure 5.6 for various scales and number

[9] Running on an Intel Xeon E5645 processor, 2.4 GHz, 290 GB RAM.

of samples per day in the training set. The scale of 1 corresponds to the original test set without any duplication. The largest jumps in normalized costs C_{RL} are observed when the group size is doubled (i.e., scale factor $2\times$). Further increases in N_{max} (i.e., more than $2\times$), only lead to marginal increase in normalized cost for any number of sample trajectories per day (ranging from 5K to 20K). These analyses further confirm that our proposed MDP formulations are generalizable to various group sizes and that a policy learned from a smaller group of EV charging stations can be used to coordinate the charging of a larger group, at least provided that the distribution of EV arrivals, departures and energy demands are similar.

5.7 Conclusion

In this paper, we took the first step to propose a reinforcement learning based approach for jointly controlling a charging demand of a group of EV charging stations. We formulated an MDP with scalable representation of an aggregated state of the group which effectively takes into account the individual EV charging characteristics (i.e., arrival time, charging and connection duration). The proposed formulations are also independent of the number of charging stations and charging rates, hence, they generalize to varying number of charging stations. We used a real-world EV charging dataset to experimentally evaluate the performance of the proposed approach compared to an uncontrolled business-as-usual (BAU) policy, as well as an optimum solution that has a perfect knowledge of the EV charging session characteristics (in terms of arrival and departure times). The summary of our analyses (in form of answer to 4 research questions) and the conclusions thereof, for a realistic 1-year long dataset (from ElaadNL) [12], are as follows:

(1) While the representation of the state and action are independent of the group size (i.e., number of charging stations), the resulting state-action space is still relatively large. Hence, feeding the entire state-action space to the learning algorithm (i.e., FQI) is not feasible. This raised the question **Q1**: What are appropriate training data time span and number of sampled trajectories from the decision trees? We investigated the effect of the training data time span and the number of sample trajectories per day on the performance of the learned policy and concluded that when the training data time span is longer than 3 months, a smaller number of samples (order of 5K) from each of the training days achieve similar performance as the larger number of sampled trajectories from those training days.

(2) We investigated how the RL policy performs compared to an optimal all-knowing oracle algorithm (i.e., **Q2**). We show that our proposed approach learns a policy which can reduce the normalized cost of coordinating charging across 10 and 50 EV charging stations by 39% and 30.4% respectively

from the normalized cost of the uncontrolled BAU charging policy. The achieved reduction in performance by our approach does not require future knowledge about EV charging sessions and it is only 13% (for N_{max} =10 charging stations) and 15.6% (for N_{max} =50 charging stations) more expensive than the optimum solution cost with has a perfect knowledge of future EV charging demand.

(3) We then analyzed how the performance of our proposed RL approach varies over time using realistic data (i.e., **Q3**) by checking whether the learned policy performs similarly when various months of the year are used as test set while the agent is trained on the preceding months. The results indicate that the flexibility — hence reduction in the normalized cost — varies across various months. In particular, the months with larger flexibility have larger reduction in cost by the learned policy with respect to the normalized cost of the BAU policy. Still, the cost gap between the learned policy and the optimal one is larger for those higher flexibility months. This is due to the scarcity of the days with larger flexibility in the training set as well as the random sampling of the state-action space, which does not guarantee inclusion of the rare but crucial parts of the state-action space in the training set that is fed to the FQI algorithm.

(4) Finally, we trained an agent using an experience from 10 EV charging stations and applied the learned policy to control a higher number of charging stations (up to a factor of $10\times$ more arrivals) to check whether the learned approach generalizes to different group sizes (question **Q4**). These analyses further confirmed that our proposed MDP formulations are generalizable to groups of varying sizes and that a policy learned from a small number of EV charging stations may be used to coordinate the charging of a larger group, at least provided that the distribution of EV arrivals, departures and energy demands are similar.

In our future research, we will study four possible improvements to the presented approach:

(1) We used random exploration of state-action space to collect the experience (in form of tuples) as an input to our learning algorithm. We will investigate whether incorporating an efficient exploration strategies to perform a more informed sampling of the state-action space improves the performance?

(2) We used a fully connected neural network for function approximation in the FQI algorithm. Since we represent our aggregate demand in a state using a matrix, it is relevant to investigate whether using convolutional neural networks (similar to the function approximation adopted in [11]) will further improve the performance.

(3) We represented the aggregate demand in each state using a 2D grid with one axis being Δt^{depart}, the other Δt^{charge}. This approach efficiently represents the aggregate demand while retaining the individual EV charging characteristic. However, the discretization during the binning process introduces an approximation error in the cost function. The error can be minimized by increasing the time-granularity (i.e., using shorter Δt^{depart} and Δt^{charge}) in the aggregate demand representation of a state. However, increasing time-granularity results in a larger state space, affecting the scalability and the learning speed of the proposed approach. Hence, as a next step, we will analyze how this approximation error may influence the optimization result at the aggregation level. We will also investigate the possibility of identifying an optimum time-granularity that results in an acceptable approximation error at the aggregated level without jeopardizing the scalability and the learning speed of the proposed approach.

(4) The learning algorithm in our proposed approach is based on the value iteration approach where the state-action value is estimated for various state-action pairs and an optimum policy is deduced form the estimated action-values. We will investigate whether learning the policy directly using policy iteration methods improves the performance.

Acknowledgment

The authors would like to thank professor Pascal Poupart for his expert advise on the reinforcement learning algorithms, and dr. Bert Claessens for providing constructive feedback on a draft of this paper.

References

[1] A. Afram and F. Janabi-Sharifi. *Theory and applications of HVAC control systems–A review of model predictive control (MPC)*. Building and Environment, 72:343–355, 2014.

[2] J. Ma, J. Qin, T. Salsbury, and P. Xu. *Demand reduction in building energy systems based on economic model predictive control*. Chemical Engineering Science, 67(1):92–100, 2012. Dynamics, Control and Optimization of Energy Systems. doi:10.1016/j.ces.2011.07.052.

[3] N. Sadeghianpourhamami, T. Demeester, D. Benoit, M. Strobbe, and C. Develder. *Modeling and analysis of residential flexibility: timing of white good usage*. Applied Energy, 179:790–805, 2016. doi:10.1016/j.apenergy.2016.07.012.

[4] N. Sadeghianpourhamami, J. Deleu, and C. Develder. *Achieving scalable model-free demand response in charging an electric vehicle fleet with reinforcement learning*. In Proc. 9th ACM Int. Conf. Future Energy Systems (e-Energy 2018), 12–15 Jun. 2018.

[5] M. Riedmiller. *Neural fitted Q iteration-first experiences with a data efficient neural reinforcement learning method*. In 16th European Conference on Machine Learning, volume 3720, pages 317–328. Springer, 2005.

[6] W. Shi and V. W. S. Wong. *Real-time vehicle-to-grid control algorithm under price uncertainty*. In 2011 IEEE International Conference on Smart Grid Communications (SmartGridComm), pages 261–266, Oct 2011. doi:10.1109/SmartGridComm.2011.6102330.

[7] A. Chis, J. Lundén, and V. Koivunen. *Reinforcement learning-based plug-in electric vehicle charging with forecasted price*. IEEE Transactions on Vehicular Technology, 66(5):3674–3684, May 2017. doi:10.1109/TVT.2016.2603536.

[8] B. J. Claessens, S. Vandael, F. Ruelens, K. D. Craemer, and B. Beusen. *Peak shaving of a heterogeneous cluster of residential flexibility carriers using reinforcement learning*. In IEEE PES ISGT Europe 2013, pages 1–5, Oct 2013. doi:10.1109/ISGTEurope.2013.6695254.

[9] S. Vandael, B. Claessens, M. Hommelberg, T. Holvoet, and G. Deconinck. *A scalable three-step approach for demand side management of plug-in hybrid vehicles*. IEEE Transactions on Smart Grid, 4(2):720–728, June 2013. doi:10.1109/TSG.2012.2213847.

[10] S. Vandael, B. Claessens, D. Ernst, T. Holvoet, and G. Deconinck. *Reinforcement learning of heuristic EV fleet charging in a day-ahead electricity market*. IEEE Transactions on Smart Grid, 6(4):1795–1805, July 2015. doi:10.1109/TSG.2015.2393059.

[11] B. J. Claessens, P. Vrancx, and F. Ruelens. *Convolutional neural networks for automatic state-time feature extraction in reinforcement learning applied to residential load control*. IEEE Transactions on Smart Grid, 9(4):3259–3269, July 2018. doi:10.1109/TSG.2016.2629450.

[12] N. Sadeghianpourhamami, N. Refa, M. Strobbe, and C. Develder. *Quantitive analysis of electric vehicle flexibility: a data-driven approach*. Int. J. Electr. Power Energy Syst., 95:451–462, Feb. 2018. doi:10.1016/j.ijepes.2017.09.007.

[13] M. Riedmiller. *10 steps and some tricks to set up neural reinforcement controllers*, pages 735–757. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012. doi:10.1007/978-3-642-35289-8–39.

[14] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis. *Human-level control through deep reinforcement learning*. Nature, 518:529–533, Feb 2015. doi:10.1038/nature14236.

[15] P. J. Huber. *Robust estimation of a location parameter*. Ann. Math. Statist., 35(1):73–101, 03 1964. doi:10.1214/aoms/1177703732.

[16] H. Tang, R. Houthooft, D. Foote, A. Stooke, O. X. Chen, Y. Duan, J. Schulman, F. DeTurck, and P. Abbeel. *Exploration: a study of count-based exploration for deep reinforcement learning*. In Advances in Neural Information Processing Systems, pages 2753–2762, 2017.

[17] V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. Lillicrap, T. Harley, D. Silver, and K. Kavukcuoglu. *Asynchronous methods for deep reinforcement learning*. In International conference on machine learning, pages 1928–1937, 2016.

[18] R. McFarlane. *A survey of exploration strategies in reinforcement learning*. McGill University, http://www. cs. mcgill. ca/ cs526/roger. pdf, accessed: August 2018, 2018.

6 Conclusion

"When you have exhausted all the possibilities, remember this: you haven't"

–Thomas Edison

6.1 Summary

This thesis focused on paving the way for realization of practical demand response (DR) algorithms via two types of contributions: (i) analysis, characterization, quantification and modeling of the flexibility stemming from residential white–good usage and roadside EV charging based on real-world data, and (ii) proposal of a model-free approach based on RL to learn the best coordination policy for charging the collection of EVs from a real-world dataset.

Flexibility is the main asset for DR. Hence, a clear understanding of the flexibility stemming from various applications is crucial to design a practical DR algorithm and realistically assess its impact. As a part of its first contribution, this thesis provided insights on the flexibility characteristics from white-good usage and EV charging sessions and answered the following research questions.

For the flexibility stemming from residential white-good usage (characterized and modeled in Chapter 2) :

(1) *Do customers have specific behavioral patterns in using their smart appliances flexibly? Does this behavior vary among different households?* Three

behavioral patterns were observed when clustering the customer flexibility profiles: (i) similar deadlines and different configuration times (a dominant behavioral pattern in smart dishwasher usage), (ii) similar configuration times but different deadlines, and (iii) similar flexibility duration.. Not all of these patterns necessarily exist across all customers and smart appliances: customer behavioral patterns are different form one appliance to the other and also across the different households. This further suggests that assuming a unique behavioral patterns across appliances in different households when designing and assessing DR algorithms is not realistic. Hence, to achieve a practical DR algorithm, the behavioral variations among the participants should be taken into account.

(2) *Do customers' usage habits, hence the offered flexibility, change over time (e.g., during weekends, holidays, and change of seasons?)* To answer this question, we statistically tested the effect of 4 factors on the behavior of observed customers for smart appliances: holidays, weekends, seasons and days-of-the-week. The dependency analysis of real-world data suggests that customers are not similarly affected by aforementioned factors. For example, the weekends influence the behavior of at least 50% of the customers with a smart dishwasher. Seasons and day-of-the-week factors influence the behavioral patterns of more than 50% of the customers in using their smart washing machines.

For the flexibility stemming from EV charging sessions (analyzed and quantized in Chapter 3):

(1) *Do EV owners have specific habits to charge their cars (e.g., taking their cars to a charging station at particular times of the day)?* To answer this question, we clustered the EV data in a 2D space in terms of arrival and departure times. As such, we identified three behavioral clusters: charge near home, charge near work, and park to charge clusters. The three behavioral clusters differ substantially in their arrival times, sojourn times and the idle times as illustrated in Chapter 3.

(2) *Are the characteristics of the charging sessions (e.g., arrival, sojourn and idle times) sensitive to seasonal changes or weekdays?* Weekends and weekdays as well as seasonal changes impact the arrival times in all three clusters. In general, the arrival times are earlier in summer and spring in all the clusters. The arrivals are also earlier on weekdays compared to weekends. However, seasons have no substantial impact on the sojourn and idle times. Sessions in park to charge and charge near work clusters have shorter sojourn and idle times in the weekends whereas the sessions in the charge near

home clusters have longer sojourn and idle times in the weekends compared to weekdays.

(3) *How is flexibility (in terms of amount, time and duration of the shifted energy) exploited? Which aspect of flexibility (time and duration of availability or amount of deferrable energy) is more useful at various times of the day?* The flexibility exploitation is greatly influenced by the uncontrolled business as usual (BAU) load patterns, the distribution of arrival times, and the renewable energy generation patterns. The main motivation for exploitation of the flexibility in both load flattening and load balancing is to fill the valleys of the BAU load pattern. Hence, longer shifts are observed from the evening peaks compared to the morning peaks in the weekdays (since the nighttime valley is larger and deeper). Similarly, longer shifts are seen from Saturday peaks compared to Sunday peaks because the night valley between Saturdays and Sundays is bigger. For arrivals in the afternoon until midnight, flexibility in terms of deferrable time is almost fully exploited to ensure the charging takes place in the nighttime (which corresponds to the lower demand). Yet, this does not imply that all the charging is delayed, meaning that the state of charge of the battery at the BAU charging completion time is pretty high. Across the behavioral clusters, the offered flexibility in charge near work cluster is often used to fill the afternoon valley since these sessions are characterized by morning arrivals and their sojourn typically does not cover the night valley. Hence, their exploitation in terms of deferrable time and energy is typically lower compared to the arrivals in the other clusters which are usually in the afternoon. The sessions in the charge near home cluster are better candidates to fill the night valley.

The aforementioned analysis using real-world datasets and answering various questions pertaining the customers behavioral patters fosters more realistic assumptions for the design and implementation of DR algorithms. In addition to the provided insights on the flexibility characteristics, availability of the real-world datasets to broader spectrum of researchers also promotes a realistic design and assessment of the DR algorithms. However, the deployment of the smart appliances in the residential sector is still at its infancy, hence, the flexible appliance usage data is typically only available from pilot projects and testbeds. The majority of these datasets are not public due to intellectual property issues. It is also very expensive to collect such data. This hinders the widespread development of practical and realistic DR algorithms. To alleviate this issue, this dissertation proposed (in Chapter 2) two systematic methods based on a real-world data to model the flexibility behavior of an individual residential household towards a particular smart appliance. These models can be used for generating the flexible white-good appliance usage data.

The first method (denoted as Model I in Chapter 2) is a two-stage approach. In the first stage, we identify the typical deadlines of the customer by using a clustering algorithm and estimate the probability $P(\text{deadline})$ proportional to the size of the cluster for the respective deadline. For each cluster of deadlines resulting from Stage I, we then use parametric probability distributions (GMMs in particular) to model the corresponding configuration times in Stage II and obtain $P(\text{configuration time} \mid \text{deadline})$. The joint distribution of deadlines and configuration times can then be obtained by $P(\text{configuration time, deadline}) = P(\text{deadline}) \cdot P(\text{configuration time} \mid \text{deadline})$. In the second method (denoted as Model II in Chapter 2), a bivariate GMM is fit to the flexibility profile of the customer: both of the flexibility features (i.e., deadline and configuration time) form the input to the model, and their joint probability distribution is estimated in a single step. The Bayesian MCMC is employed to fit the bivariate mixture model and select the optimal number of components. To validate the efficiency of Model I and Model II in regenerating the customer behavior for synthetic data generation purposes, we proposed a systematic approach based on a Kolmogorov-Smirnov (k-s) test. Based on our validations, Model II was identified to be an appropriate regenerative model for all 3 types of white goods in our analysis.

The timing aspect of the flexibility is of cylindrical nature (i.e., time of availability is a circular measure while the duration of shift in energy is a linear quantity). This raises a question whether probabilistic generative models using cylindrical distributions are better then the linear ones for modeling energy consumption flexibility.

To answer the above-raised question, Chapter 4 proposes a Bayesian approach based on MCMC to estimate the parameters of Abe-Ley mixture distribution. Abe-Ley distribution is a cylindrical distribution based on the combination of Weibull and sine-skewed von-Mises. The choice of the distribution is motivated by its various merits compared to the other existing cylindrical distributions which are outlined as: (i) flexible shapes, (ii) cross-correlation among linear and circular variables, (iii) well-known marginal and conditional distributions and (iv) support of data skewness. The proposed approach in Chapter 4 is then used to model the residential white-good usage flexibility and compare it with the proposed linear models of Chapter 1 in Appendix A. In terms of the predictive accuracy, the linear models however are still found to be better models than the cylindrical ones.

The second contribution of this thesis (Chapter 5) is the development of a model-free DR algorithm for for jointly coordinating a charging demand of a group of EV charging stations. We formulated an MDP with scalable representation of an aggregated state of the group which effectively takes into account the individual EV charging characteristics (i.e., arrival time, charging and connection duration). The proposed formulations are also independent of the number of charging stations and charging rates, hence, they generalize to varying number of charging stations.

We adopt batch reinforcement learning (fitted Q-iteration) with function approximation to find the best EV charging policy. While the representation of the state and action are independent of the group size (i.e., number of charging stations), the resulting state-action space is still relatively large. Hence, feeding the entire state-action space to the learning algorithm (i.e., FQI) is not feasible. This raised the following question:

(1) *What are appropriate training data time span and number of sampled trajectories from the decision trees?* To answer this question, we investigated the effect of the training data time span and the number of sample trajectories per day on the performance of the learned policy and concluded that when the training data time span is longer than 3 months, a smaller number of samples (order of 5K) from training days achieve similar performance as the larger number of sampled trajectories from those training days.

Furthermore, we used a real-world EV charging dataset to experimentally evaluate the performance of the proposed approach compared to an uncontrolled business-as-usual (BAU) policy, as well as an optimum solution that has a perfect knowledge of the EV charging session characteristics (in terms of arrival and departure times). The following questions where raised and answered.

(2) *How does the RL policy perform compare to an optimal all-knowing oracle algorithm?* We show that our proposed approach learns a policy which can reduce the normalized cost of coordinating charging across 10 and 50 EV charging stations by 39% and 30.4% respectively from the normalized cost of the uncontrolled BAU charging policy. The achieved reduction in performance by our approach does not require future knowledge about EV charging sessions and it only 13% (for 10 charging stations) and 15.6% (for 50 charging stations) more expensive than the optimum solution cost which has a perfect knowledge of future EV charging demand.

(3) *How does that performance vary over time using realistic data?* We answer this question by checking whether the learned policy performs similarly when various months of the year are used as test set while the agent is trained on the preceding months. The results indicate that the flexibility, hence reduction in the normalized cost varies across various months. In particular, the months with larger flexibility have larger reduction in cost by the learned policy with respect to normalized cost of BAU policy.

(4) *Does a learned approach generalize to different EV group sizes?* To answer this question, we trained an agent using an experience from 10 EV charging stations and applied the learned policy to control a higher number of charging stations (up to a factor of 10× more arrivals). The analyses further

> confirmed that our proposed MDP formulations are generalizable to groups of varying sizes and that a policy learned from a small number of EV charging stations may be used to coordinate the charging of a larger group, at least provided that the distribution of EV arrivals, departures and energy demands are similar.

In summary, DR algorithms have received very substantial attention as a viable and cost effective solution of guaranteeing supply-demand balance in the smart grid. However, widespread deployment of DR algorithms in the grid has been slow due to various barriers. This thesis focused on the paving a way to establishing practical DR algorithms (which is one of the DR barriers). The work presented in this thesis comprises (i) analysis and characterization of energy consumption flexibility, (ii) development of generative probabilistic models to facilitate availability of realistic data to broader range of researchers, and (iii) proposal of a model-free DR algorithm for the coordination of the charging of EV groups of various sizes.

6.2 Future Work

Development of practical DR algorithms is the key to progress towards realizing the full potential of the smart grid. With advances in deep reinforcement learning, model-free DR solutions are proliferating in the recent literature. As established in this thesis, model-free approaches do not require accurate models of the coordination problem, hence are a promising approach in realizing practical DR algorithms. Specifically, in the residential sector, as shown in Chapter 2, the uncertainty associated with consumer lifestyle makes the modeling task a challenging one.

The model-free DR algorithms are data-driven, hence, the need for real-world energy consumption data is more critical than ever. This thesis presented generative probabilistic models based on the real-world flexible energy usage by residential costumers. Such modeling could also be applied to model customer behavior towards other flexibility sources (e.g., TCLs).

Furthermore, the majority of the existing model-free DR algorithms can coordinate only a very small number of flexible devices due to the scalability issues when formulating the joint coordination problem as an MDP. The large state-action spaces associated with many of the real-world coordination problems further challenge the exploration of the environment for the learning agent. Some of the promising directions to foster model-free DR solutions for large-scale coordination problems include: (i) the use of function approximation (e.g., deep neural networks) to generalize the learning from the explored states to unexplored states, (ii) development of efficient exploration strategies, (iii) the use of policy iteration algorithms to learn the best policy directly instead of approximating the action–value function. Note that in context of the reinforcement learning, various algo-

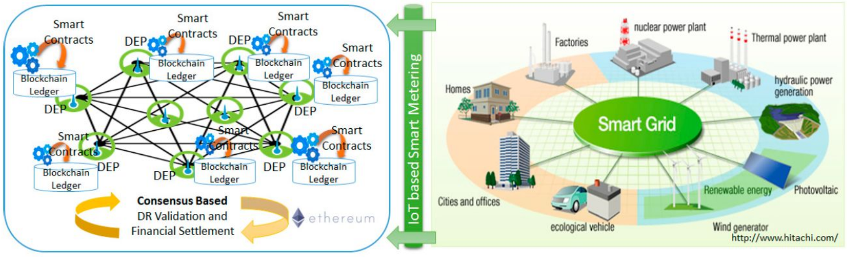


Figure 6.1: Blockchain based architecture for decentralized management of energy grids (from [1])

rithms are already available to tackle the aforementioned challenges. However, they need to be effectively adopted for the coordination problems in smart grid paradigm. The use of multi-agent reinforcement leaning is also a promising direction for DR since it is a decentralized approach where the learning agent only needs local information to learn the best coordination policy (although, a global reward/cost still needs to be communicated with each agent).

Another promising (and very recent) direction for DR algorithms is the use of smart contracts within the blockchain technology. Blockchain technology has received a great deal of attention in the energy sector because it flourishes a new peer-to-peer market for trading not only the locally produced energy but also the demand flexibility. Introduction of such peer-to-peer trading brings many changes to the traditional DR schemes. For example, it eliminates the need for a central entity (e.g., distribution grid operator) to promote DR participation. Pop *et al.* [1] demonstrate the use of decentralized blockchain mechanisms for delivering transparent, secure, reliable, and timely energy flexibility. Their envisioned decentralized management of smart grid is depicted in Figure 6.1. One of the challenges the blockchain based DR algorithms need to address is the scalability issue due to well-known “consensus problem” [2] in the blockchains which limits the number of transactions per second.

References

[1] C. Pop, T. Cioara, M. Antal, I. Anghel, I. Salomie, and M. Bertoncini. *Blockchain based decentralized management of demand response programs in smart energy grids*. Sensors, 18(1):162, 2018.

[2] S. Keshav. *How blockchain ban democratize green power*. https://wise.uwaterloo.ca/news1/commentary/how_blockchain_can_democratize_green_power. Accessed: 2018-09-12.

A
Modeling Real-World Flexibility of Residential Power Consumption: Exploring the Cylindrical WeiSSVM Distribution

In this appendix, we compare the generative models developed in Chapter 1 (using Gaussian mixture models on linear scale) for modeling the flexible white-goods usage with the generative models developed using Abe-Ley mixtures in Chapter 4. Note that Abe-Ley mixtures and WeiSSVM mixtures are two names used in the literature to refer to the same distribution. In this literature, we denote the distributions developed and estimated in Chapter 4 as WeiSSVM mixture models (WMM).

⋆ ⋆ ⋆

N. Sadeghianpourhamami, D. F. Benoit, D. Deschrijver, and C. Develder.

Published in Proceedings of 9th ACM International Conference on Future Energy Systems (e-Energy 2018), Jun. 2018.

Abstract A user's power consumption flexibility is defined in terms of amount, time and duration of availability. The timing of flexibility is circular in nature.

Therefore, it is natural to adopt circular distributions to model this data. This paper investigates the key research question whether that leads to better generative models than using conventional linear distributions. In particular, it fits Gaussian mixture models and a very flexible recent cylindrical (WeiSSVM) distribution mixture to real-world field trial data. Using a predictive accuracy performance measure, it is found that the latter does not provide substantially better fits. Shortcomings of both models are pointed out and it is concluded that research for appropriate statistical models for the observed data is still open.

A.1 Introduction

User energy consumption flexibility is typically characterized by the amount and the duration of the deferrable energy at various times of the days. The timing aspect of the flexibility is greatly influenced by user lifestyle and energy consumption habits. Hence, to derive generative models of user flexibility, its timing aspect is quantified using configuration time and deadline [1]. The *configuration time* indicates when the users configure their smart appliances flexibly and the *deadline* is the latest possible start time of the appliance. The configuration time is of cyclic nature while the deadline is a linear quantity. Hence, flexibility measurements can naturally be regarded as bivariate cylindrical data.

Initial studies modeling the energy consumption flexibility avoid the cylindrical representation by defining a heuristic algorithm that identifies the middle of the largest gap on the circular axis to wrap the data around and proceed to modeling using probabilistic models defined on linear scales (e.g., [1] [2]). However, such heuristic algorithms might fail in situations where such a reference point is challenging or impossible to find. On the other hand, most of the existing cylindrical distributions are limited in terms of flexibility in modeling cross-correlation between the cylindrical and linear variables as well as modeling skewness and heterogeneity in the data (which requires mixture models). Recently, Abe and Ley proposed a cylindrical distribution (named WeiSSVM) which is tractable, flexible, has well-known conditional and marginal distributions and models the skewness and cross-correlation between the cylindrical and linear variables [3].

This paper investigates how well WeiSSVM mixtures can model the energy consumption flexibility compared to using distributions defined on the linear scale. The analysis is based on data from year-long measurements in the LINEAR pilot project [4], where [1] previously modeled the user behavior towards smart wet-appliances with Gaussian mixture models (GMM).

A.2 Modeling User Energy Consumption Flexibility with WeiSSVM Mixtures

A.2.0.1 PDF of WeiSSVM Mixtures

The WeiSSVM distribution is a combination of a Weibull distribution and the sine-skewed Von-Mises distribution. Its probability density is defined as [3]:

$$\begin{aligned} f(\theta, x|\boldsymbol{\zeta}) \mapsto \frac{\alpha\beta^{\alpha}}{2\pi\cosh(\kappa)} \cdot (1 + \lambda\sin(\theta - \mu)) \cdot x^{\alpha-1} \cdot \\ \exp[-(\beta x)^{\alpha}(1 - \tanh(\kappa)\cos(\theta - \mu)], \end{aligned}$$

with random variables $(\theta, x) \in [0, 2\pi) \times [0, \infty)$, and distribution parameters $\boldsymbol{\zeta} = (\alpha, \beta, \mu, \kappa, \lambda)$. The parameter vector of the WeiSSVM distribution comprises $\alpha, \beta > 0$, which are linear shape and scale parameters respectively, $0 \leq \mu < 2\pi$ is a circular location parameter, $\kappa \geq 0$ controls the circular concentration, and $-1 \leq \lambda \leq 1$ controls the circular skewness. The mixture of a K-component WeiSSVM distribution has the following density function:

$$f(\theta, x|\boldsymbol{\vartheta}) = \sum_{k=1}^{K} \eta_k f_k(\theta, x|\boldsymbol{\zeta_k})$$

where $f_k(\theta, x|\boldsymbol{\zeta_k})$ is the probability density of the k^{th} component indexed by parameter set $\boldsymbol{\zeta_k}$; η_k is the weight of the k^{th} component, thus $\boldsymbol{\eta} = (\eta_1, \eta_2, ..., \eta_K)$ is the weight distribution constrained by:

$$\eta_k \geq 0, \quad \eta_1 + \eta_2 + ... + \eta_K = 1.$$

Hence, $\boldsymbol{\vartheta} = (\boldsymbol{\zeta}_1, ..., \boldsymbol{\zeta}_K, \boldsymbol{\eta})$ is the parameter vector of the mixture model.

A.2.0.2 Model Parameter Estimation

Note that the likelihood of the WMMs is a high dimensional function. Hence, Bayesian methods are more reliable than point-estimates (e.g., expectation maximization) in estimating the parameters of the WMMs since they output the entire posterior distribution. The Metropolis-Hastings algorithm [5] is used to estimate the parameters of the WMMs for the user flexibility.

A.2.0.3 Measures for Model Comparison

One of the use cases of generative models of the energy usage flexibility is to generate data samples (e.g., to simulate scenarios for assessing DR impact). Hence, it is natural to compare the generative models in terms of their out-of-sample predictive accuracy. For comparing the models, we use log point-wise prediction density

Table A.1: *elppd* of GMM and WMM fits for dishwashers.

User	WMM	GMM	User	WMM	GMM
1	-73.0(3)	**-45.1(3)**	9	-496.9(5)	**-469.8(5)**
2	-173.0(2)	**-158.0(2)**	10	-579.0(4)	**-556.1(4)**
3	-247.8(5)	**-229.3(4)**	11	-486.7(4)	**-464.7(4)**
4	-341.3(5)	**-314.7(4)**	12	-807.6(3)	**-786.3(5)**
5	-198.0(5)	**-157.5(5)**	13	-1054.3(6)	**-1011.7(5)**
6	-245.6(4)	**-219.3(4)**	14	-450.8 (4)	**-413.7(4)**
7	**-401.5(5)**	-411.5(4)	15	-1031.9(5)	**-1015.6(5)**
8	-488.6(4)	**-456.6(4)**			

(*elppd*), calculated from the posterior samples, as a popular method of quantizing the out-of-sample predictive accuracy of a model [6]. When modeling data using mixture models, one also needs to identify the optimal number of mixtures. This value is determined by finding a knee point in the plot of the *elppd* measure vs. the number of mixtures.

A.3 Analysis and Model Comparison

This section presents the results of fitting WMMs as a generative model of the energy usage flexibility for the households that participated in the aforementioned LINEAR pilot project [4]. This data (comprising 15 dishwashers, 12 washing machines and 8 tumble dryers) was previously modeled using GMMs [1]. The models are compared next. In terms of the resulting clusters: GMMs typically identify either clusters in parallel with the x-axis or clusters along a diagonal as seen from the example users depicted in the bottom row of Figure A.1. The former indicates configurations with similar deadline while the latter maps to configurations with similar flexibility duration. WMMs also identify clusters in parallel with the x-axis. As opposed to GMMs, WMMs find clusters in parallel with the y-axis, indicating similar configuration time but varying deadlines and hence different flexibility duration. Also, due to the inherent nature of the distributions, each GMM component is symmetrical, but WMMs have skewed distributions with increasing concentration along the linear axis.

To compare the predictive accuracy, we calculated the *elppd* values for the users of all three appliances and summarized the values for dishwashers in Table A.1. The numbers between parentheses are the optimum number of mixtures. The bold values in Table A.1 indicate a better generative model. As seen from Table A.1, WMMs perform comparable or worse than the GMMs for the majority of users of dishwashers. Similar results are obtained for users of washing machines and tumble dryers. This is due to inherent characteristics of the WMMs

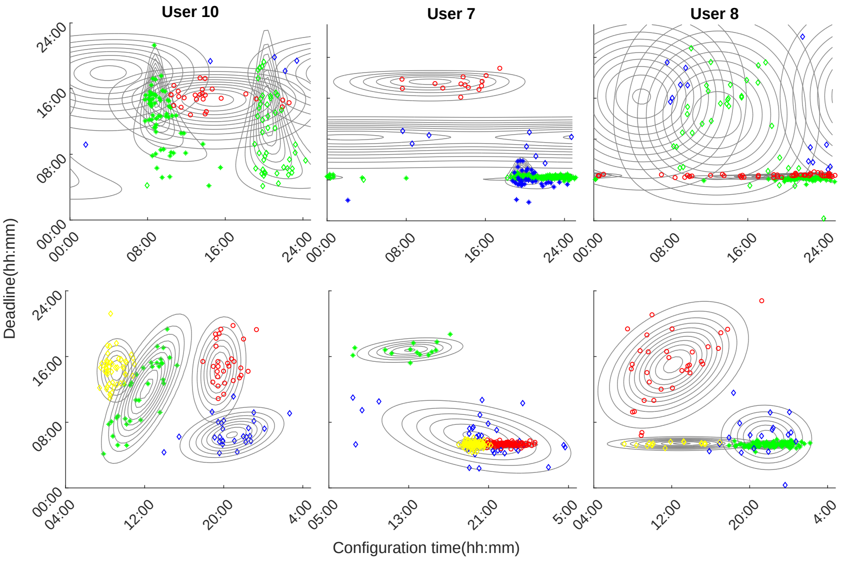


Figure A.1: Comparison of distribution shapes of the mixture components and identified clusters with WMMs (top row) and GMMs (bottom row) for selected users of dishwashers. Note that data on bottom row is wrapped around a new x-axis reference, while for the data on top row, the x-axis is circular (i.e., 00:00 and 24:00 are the same points)

and GMMs. WMMs are more suitable in modeling datasets in which the circular concentration increases along the linear axis, while the measurements in LINEAR show that users do not exhibit such trait. Instead, the areas of high densities indicating a consistent daily behavior is seen in configuration patterns of the users for all three appliances. This characteristic is well modeled with GMMs.

A.4 Conclusion

An analysis of the LINEAR dataset shows that GMMs are preferred over WMMs, because they compute better generative models. Despite having better predictive accuracy than WMMs, GMMs still suffer from the following limitations: 1. GMMs are defined for linear scale and are not suitable for modeling scenarios where an adequate reference on the circular axis is impossible or challenging to find, 2. GMMs are defined on both negative and positive values, but the flexibility characteristics are positive quantities. Hence, GMMs are prone to generating meaningless samples.

Hence, defining a suitable distribution for modeling user energy consumption behavior cylindrically is still an open research issue. Given that (compared to real-life trials) simulating scenarios for testing DR algorithms' impacts is highly cost effective, it is also a highly relevant issue.

References

[1] N. Sadeghianpourhamami, T. Demeester, D. Benoit, M. Strobbe, and C. Develder. *Modeling and analysis of residential flexibility: Timing of white good usage*. Applied Energy, 179(Supplement C):790 – 805, 2016. doi:10.1016/j.apenergy.2016.07.012.

[2] C. Develder, N. Sadeghianpourhamami, M. Strobbe, and N. Refa. *Quantifying flexibility in EV charging as DR potential: Analysis of two real-world data sets*. In Proc. 7th IEEE Int. Conf. Smart Grid Communications (SmartGridComm 2016), pages 600–605, Sydney, Australia, 6–9 Nov. 2016. doi:10.1109/SmartGridComm.2016.7778827.

[3] T. Abe and C. Ley. *A tractable, parsimonious and flexible model for cylindrical data, with applications*. Econometrics and Statistics, 4(Supplement C):91 – 104, 2017. doi:10.1016/j.ecosta.2016.04.001.

[4] B. Dupont, P. Vingerhoets, P. Tant, K. Vanthournout, W. Cardinaels, T. D. Rybel, E. Peeters, and R. Belmans. *LINEAR breakthrough project: Large-scale implementation of smart grid technologies in distribution grids*. In 2012 3rd IEEE PES Innovative Smart Grid Technologies Europe (ISGT Europe), pages 1–8, Oct 2012. doi:10.1109/ISGTEurope.2012.6465708.

[5] W. K. Hastings. *Monte Carlo sampling methods using Markov chains and their applications*. Biometrika, pages 97–109, 1970. doi:10.1093/biomet/57.1.97.

[6] A. Vehtari, A. Gelman, and J. Gabry. *Practical Bayesian model evaluation using leave-one-out cross-validation and WAIC*. Statistics and Computing, 27(5):1413–1432, Sep 2017. doi:10.1007/s11222-016-9696-4.