Dominik Egarter
Matr.-Nr.: 0760501

Load Disaggregation Applications using Active Power Measurements

Dissertation

zur Erlangung des akademischen Grades
Doktor der technischen Wissenschaften

Informationstechnik

Alpen-Adria-Universität Klagenfurt
Fakultät für Technische Wissenschaften

Begutachter:

Univ.–Prof. Dr. techn. Wilfried Elmenreich
Institut für Vernetzte und Eingebettete Systeme
Alpen-Adria-Universität Klagenfurt

Prof. Dr. Thorsten Staake
Lehrstuhl für Wirtschaftsinformatik,
insbesondere Energieeffiziente Systeme
Universität Bamberg

Klagenfurt, Dezember 2015

Declaration of honor

I hereby declare in lieu of an oath that

- the submitted academic paper is entirely my own work and that no auxiliary materials have been used other than those indicated,
- I have fully disclosed all assistance received from third parties during the process of writing the paper, including any significant advice from supervisors,
- any contents taken from the works of third parties or my own works that have been included either literally or in spirit have been appropriately marked and the respective source of the information has been clearly identified with precise bibliographical references (e.g. in footnotes),
- to date, I have not submitted this paper to an examining authority either in Austria or abroad and that
- the digital version of the paper submitted for the purpose of plagiarism assessment is fully consistent with the printed version.

I am aware that a declaration contrary to the facts will have legal consequences.

Signature Date/Place

Abstract

The Smart Grid is aiming at improving today's power grid to work more efficient and more reliable. It should enable the grid to work in an optimal and safe way even with an expected increased integration of a high number of distributed renewable energies generation units. One key factor to achieve these goals is to introduce smart metering providing fine-grained measurements letting us know when and where how much energy was consumed.

The availability of fine-grained consumption information helps to improve energy awareness of inhabitants which leads to a more efficient use of energy resources. In general, the more detailed the information of energy consumption the higher are the expected savings. Thus, feedback on the energy consumption of particular devices is beneficial to increase energy savings. One possible approach to assess consumption information on device-level is to add a measurement unit to each appliance. This introduces additional costs and increases also the energy consumption due to the additional metering units. In this context, Non-intrusive Load Monitoring (NILM) tries to break down the household consumption data to its appliance components at the grid connection point with minimum costs. The basic idea of NILM is to use statistical information of the appliance usage and to apply this knowledge for detecting running appliances in the overall power consumption.

This thesis deals with three different applications for non-intrusive load monitoring. First, a simple optimization based approach is proposed to solve the problem of aggregated power loads. Six different metaheuristic optimization techniques are used and tested on real-world data. The evaluation showed that the procedure is possible for simple setups, but cannot deal with device configurations that are typical for households.

Furthermore, to assess the complexity of NILM, the thesis is dealing with two complexity measures for classifying the load disaggregation problem. This application was inspired by the fact that there is no general common problem definition for NILM. Different NILM evaluations are using real-world datasets with different pre-processing stages and system assumptions. A fair comparison between different NILM problems is only possible with a complexity measure describing the NILM problem. The evaluations on three different real-world datasets showed that the proposed complexity measures are suitable to classify load disaggregation problems according to their complexity.

Finally, the thesis introduces a new unsupervised NILM approach. This approach is working without system information and is improving its system knowledge over time. It is working online and is suitable to run on embedded hardware. The applicability and the usefulness for NILM applications has been evaluated with synthetic and real-world data.

Zusammenfassung

Das Smart Grid hat das Ziel das heutige Stromnetz zu verbessern um effizienter und zuverlässiger zu arbeiten. Es soll dem Stromnetz ermöglichen in einem optimalen und sicheren Zustand zu funktionieren, obwohl eine steigende Integration von verteilten erneuerbaren Energieerzeugungseinheiten zu erwarten ist. Ein Schlüsselfaktor, um dieses Ziel zu erreichen, ist die Einführung von Smart Metering, welches feinkörnige Messungen bietet, um zu wissen, wann und wo, wieviel Energie verbraucht wurde.

Die Verfügbarkeit der feinkörnigen Verbrauchsinformationen hilft dabei das Energiebewusstsein der Einwohner zu verbessern, was zu einer effizienteren Nutzung von Energieressourcen führt. Im Allgemeinen sind die zu erwartenden Einsparungen um so höher je detaillierter die Informationen über den Energieverbrauch sind. Ein möglicher Ansatz Verbrauchsinformationen auf Geräteebene zu beurteilen, ist es zu jedem Gerät eine Messeinheit hinzuzufügen. Dies führt zu zusätzlichen Kosten und erhöht auch den Energieverbrauch aufgrund der zusätzlichen Messeinheiten. Non-intrusive load monitoring (NILM) versucht in diesem Zusammenhang den Hausverhaltsverbrauch auf die Gerätekomponenten mittels eines zentralen Messansatzes mit minimalen Kosten der Messeinheiten herunterzubrechen. Die grundsätzliche Idee von NILM ist es statistische Informationen des Geräteverbrauches zu nutzen und dieses Wissen zu einem Klassifizierungsmechanismus zu führen, um laufende Geräte zu erkennen.

Diese Doktorarbeit befasst sich mit drei verschiedenen Anwendungen für Non-intrusive Load Monitoring. Zunächst wird ein einfacher Optimierungsansatz vorgeschlagen, um das Problem von aggregrieten Stromverbrauchern zu lösen. Sechs verschiedene metaheuristische Optimierungsverfahren werden dafür verwendet und mit realen Daten getestet. Die Auswertung ergab, dass das Verfahren für einfache Konfigurationen möglich ist, aber nicht mit Gerätekonfigurationen, die typisch für einen Haushalt sind, umgehen kann.

Um die Komplexität zu beurteilen, befasst sich diese Arbeit mit zwei Komplexitätsmaßen zur Klassifizierung des NILM-Problems. Diese Anwendung wurde durch die Tatsache inspiriert, dass es keine generell übliche Problemdefinition für NILM gibt. Verschiedenste NILM-Ansätze verwenden reale Datensätze mit verschiedenen Vorbearbeitungsstufen und Systemannahmen. Ein fairer Vergleich zwischen verschiedenen NILM-Algorithmen ist daher ohne ein Komplexitätsmaß zur Beschreibung eines NILM-Problems nicht möglich. Die Evaluierung mittels drei verschiedenen Verbrauchsdatensätzen zeigte, dass die vorgeschlagenen Komplexitätsmaße dazu geeignet sind, das NILM-Problem entsprechend ihrer Komplexität zu klassifizieren.

Schlussendlich, führt diese Arbeit auch noch einen neuen unbeobachteten (unsupervised) NILM-Ansatz ein. Dieser Ansatz funktioniert ohne Systeminforma-

tionen und verbessert ständig die Systeminformationen. Er arbeitet online und ist fähig auf Embedded-Hardware zu laufen. Der Ansatz wurde mit künstlichen und realen Szenarien auf seine Anwendbarkeit und Nützlichkeit für NILM-Anwendungen hin überprüft.

Acknowledgements

First, I would like to thank Wilfried Elmenreich. He attended me during my entire progress of my thesis and supported me very well. With his knowledge and know-how I got a lot of advice and he helped me a lot to successfully write this thesis.

Second, I would like to thank my colleagues Andrea Monacchi and Manfred Pöchacker. Both helped me a lot with technical discussions. I really enjoyed the time doing research with them and writing scientific papers.

Third, I would like to thank Prof. Thorsten Staake for being my second reviewer and for the valuable feedback on my work.

Last but not least, I want to thank my parents, who gave me the opportunity to make the education I liked and supported me during my whole educational and research time.

List of Acronyms

ACC Accuracy

CS Cuckoo Search Algorithm

DE Differential Evolution

EA Evolutionary Algorithm

ERR Allocated Energy Error

FHMM Factorial Hidden Markov Model

FSM Finite State Machine

FA Firefly Algorithm

HMM Hidden Markov Model

HEMS Home Energy Management System

GREEND GREEND Electrical ENergy Dataset

NILM Non-intrusive Load Monitoring

PF Particle Filter

PSO Particle Swarm Optimization

PDF Probability Density Function

REDD Reference Energy Disaggregation Data Set

RMSE Root Mean Squared Error

SA Simulated Annealing

SIS Sequential Importance Resampling

Contents

List of Figures

List of Tables

CHAPTER 1 Introduction

"Verba volant, scripta manent"

– Caius Titus

The future Smart Grid aims to change and to improve the current power grid into a more efficient and reliable power grid. Enhancement in the grid should enable an easy and smooth integration of renewable energies. By knowing energy demand and production, the Smart Grid will be able to operate the grid in an optimal way as well as within safe conditions. The European Union introduces for this their 20/20/20 goals [Böh09] to improve the current grid. The goals are to reduce the green house gas emissions by 20%, to increase the number of renewable energy sources by 20% and to be more energy efficient by 20% by 2020.
Considering the goal to be more energy efficient, one key aspect is to know how energy is consumed. This could be achieved by knowing how much energy was consumed at which point in time. One solution for this is the initiative to introduce smart metering to retrieve information from energy measurements in a timely manner. Smart metering solutions provide fine-grained energy feedback for data analysis techniques to show how energy is used over time and accordingly, to improve energy awareness. One key application for smart meters are domestic environments who accounts for a major part of the world wide electricity consumption [EST12]. Thus, it is an important goal for current and future energy research to improve the energy awareness of inhabitants to decrease home energy consumption. In [For09], Ford suggested to increase energy efficiency of domestic environments by behavioral changes of inhabitants. Changing consumer behavior can be achieved by providing energy feedback. In general, energy feedback can be divided into indirect and direct feedback. Indirect feedback represents feedback given some time after the consumption. Examples for this are energy billings for each day or week in contrast to today's per year energy billing.

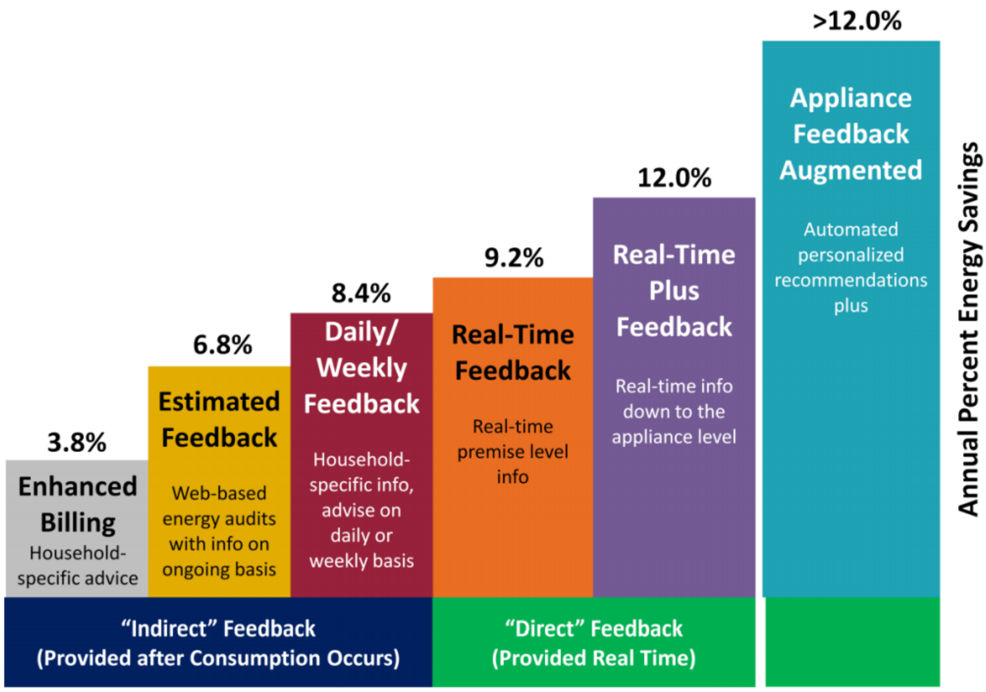

Figure 1.1: *Energy savings due to consumption feedback [Arm13]*

Direct consumption feedback means to provide consumption information in real-time. A solution is an in-home display showing the current consumption of a home. Different feedback techniques [Dar06, Fis08, Nee09, Far10] and their corresponding energy savings are presented in Figure 1.1. The best saving results have been achieved by using direct feedback on appliance level [Uen06, EM10]. In [Nee09, EM10] consumption reductions of 9 – 18% are stated based on tailored appliance level feedback.

There exist different possibilities to achieve appliance-level consumption feedback. One is to add a monitoring device to each appliance in a home tracking the electricity consumption of devices. High investment and maintenance costs are expected to be decreased in future years, but today the costs make this approach impractical and uneconomical. Additionally, the monitoring solution itself is consuming energy, thus the reduction initiated by the feedback has to be greater than the additional consumption. Another possibility for appliance-level energy feedback is the use of smart appliances [Elm12]. Smart appliances know their current operation state and consumption data and are able to communicate this information. Unfortunately, the development and standardization (e.g., interface description) of smart appliances are still in its early stages and not to be expected within the next few years.

The last possibility is to use a single meter approach named Non-intrusive Load Monitoring (NILM) performing load disaggregation. A single meter measures the total household consumption with a measurement frequency in the range of every hour to far below one second (up to 1MHz). By having a measurement point each minute or lower, characteristic appliance behaviors can be detected. Different appliances and appliance types are consuming energy in different ways making it possible to generate appliance specific patterns, models and characteristics. An example for an appliance characteristic is the amount of consumed power in operation. Accordingly, a stove has a different energy demand than a mobile phone charger. Moreover, the way an appliance is used also contributes to possible appliance characteristics. A fridge, for example, is frequently cooling over the day to control the temperature in the fridge and a water kettle is only used on demand. NILM algorithms uses these appliance characteristics to infer which appliance was used at which point in time (see Figure 1.2). However, in typical environments different appliances

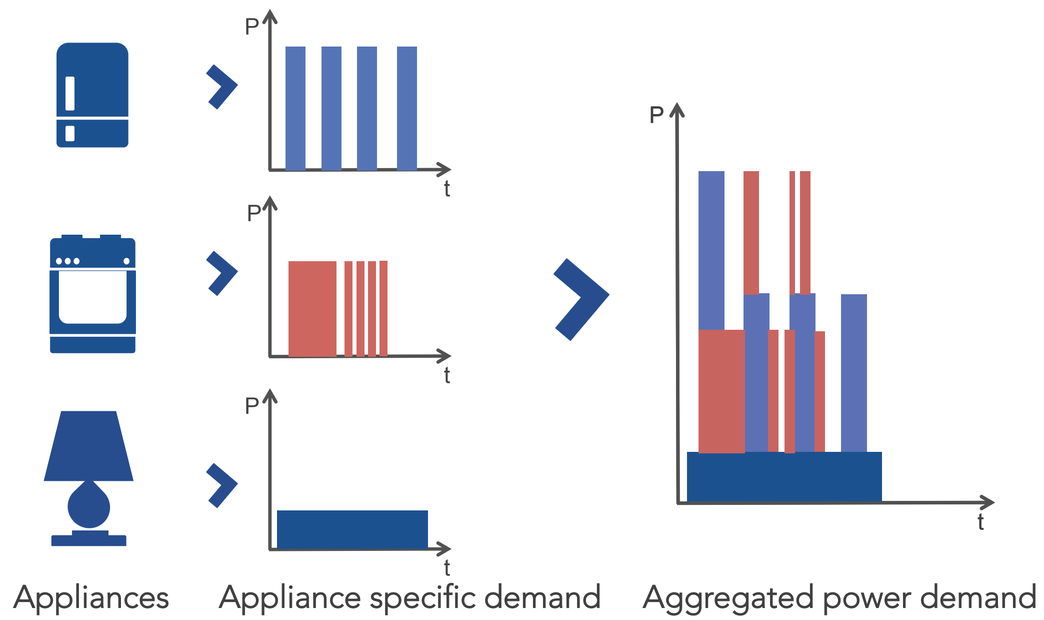

Figure 1.2: *Basic principle of the NILM process*

are used at the same time. The aggregation of appliance consumption data creates combined power values which have to be considered and disaggregated by a NILM approach. Additionally, various appliances can have also similar consumption behaviors. It is imaginable that the energy demand of a water kettle and a toaster can be in a similar range. As a consequence, the NILM problem is complex due to the fact of aggregated appliance consumption data and the similarities between appliance characteristics.

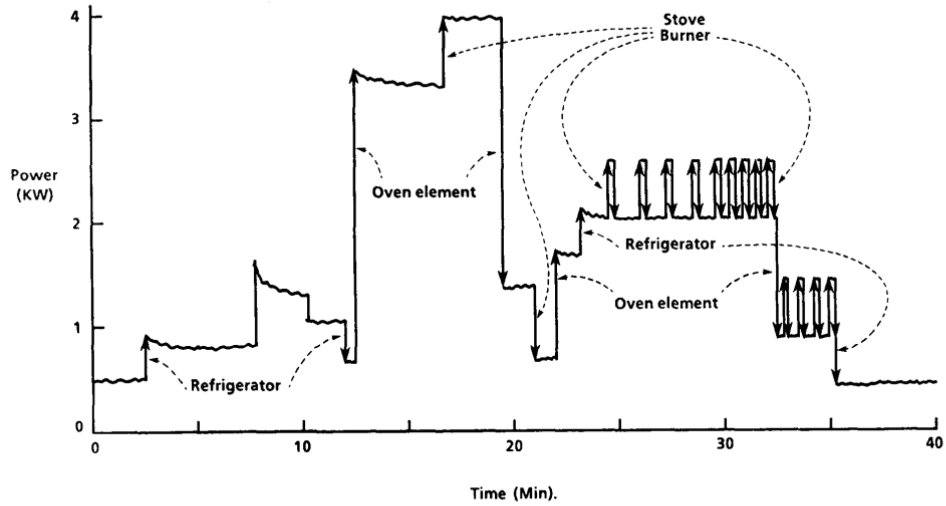

Figure 1.3: *Time series example from the original work of Hart [Har92]*

An example of a power draw of a household power consumption is presented in Figure 1.3. The figure shows different appliances with different starting and usage times in which the operations of appliances and accordingly their power draws are overlapping.

In recent years, many different disaggregation approaches were proposed opening a variety of different applications for energy feedback [Arm13], for activity recognition used in ambient assisted living [Bel13] and for load management systems [Bar14b]. Besides these positive applications to improve energy awareness and energy efficiency, NILM also opens privacy threatening issues. With $1s$ measurement granularity, NILM approaches can disaggregate around 10 different appliances [Arm13]. With information of the power demand habits on appliance level, it is possible to extract user behaviors and habits by activity recognition and user profiling [Ngu13, Lis10]. An extreme example for analysing the energy data on appliance level is shown in [Gre12]. They used a smart meter and smart algorithms to identify the multimedia content of a TV. Potentially interested stakeholders are presented in [Sko12] such as the energy utility, creditors, press and marketing/advertisements partners, and, in an extreme case, even criminals. The loss of privacy by load disaggregation and energy mining is a huge upcoming issue for society and the smart grid which calls for privacy preserving techniques such as anonymization of metering data [Eft10], privacy-preserving metering data aggregation [Li11] and masking and obfuscation of metering data [Yan12].

Considering all these aspects, NILM is a complex problem [Har92]. Algorithms are developed for certain applications. Thus, this thesis proposes approaches providing basic and novel solutions for the load disaggregation problem. In the following, we describe which applications and problems for NILM we are investigating and which research questions are solved by this thesis.

1.1 Problem Statement

In recent years, load disaggregation has become a popular research topic. In Figure 1.4 the number of publications over recent years is shown[1]. Many

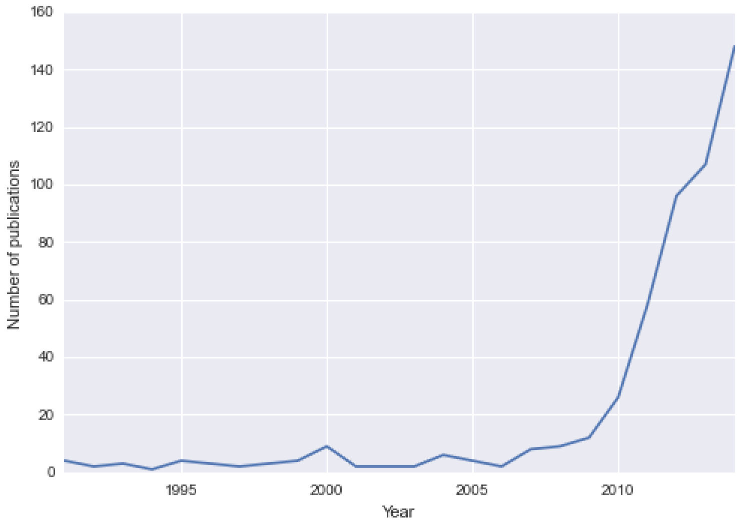

Figure 1.4: *NILM related publications over the recent years*

scientific approaches and communities contribute their work and knowledge to the topic. It has to be clearly defined which problem should be solved and which application should be stressed. Due to the roll-out of smart meters in past (e.g., in Italy) and current/upcoming years (e.g., Austria) and their characteristics to provide consumption feedback, we decided to approach applications based on active power measurements working in homes due to its availability through smart metering. In general, smart meters provide active power measurements used for billing and its general unit is Watt. In detail, we are concentrating on active power measurements with a frequency of 1 second. A frequency of 1 second means to have a measurement data point each second. We defined three different key interests, which are stressed in different chapters of this thesis:

1. *The load disaggregation problem and its original definition:* this chapter should answer the question whether it is possible and meaningful to model

[1]Presented by Oliver Parson in his blog
http://blog.oliverparson.co.uk/2015/03/overview-of-nilm-field.html

the load disaggregation problem as a stateless optimization approach (e.g., knapsack problem) and solve it with state-of-the-art optimization techniques.

2. *Description of the problem complexity:* this work aims to describe the load disaggregation problem by a complexity measure to make load disaggregation problems with different setups comparable.

3. *Introducing a novel load disaggregation approach:* the proposed load disaggregation approach should have the following novel combinations of characteristics such as i) the approach should be one-feature based, ii) the approach uses unsupervised classification, iii) the approach includes autonomous learning and continuous improvement of appliance models, iv) the approach operates online on each sample and works on restricted hardware.

The three key interests are motivated in each corresponding chapter and should provide a broad and deep look into the topic on each approach.

1.2 Outline

This thesis starts with **Chapter 1** presenting the topic and motivation of this work.
Chapter 2 provides an overview of state-of-the-art load monitoring techniques. This include intrusive load monitoring which is a distributed monitoring approach as well as non-intrusive load monitoring being the main topic of this thesis. The survey of non-intrusive load monitoring divides the process of load monitoring in their processing stages and describes the most important approaches over the last years.
Chapter 3 deals with the question whether the load disaggregation problem can be modelled as a knapsack problem. The knapsack problem is adapted to model the load disaggregation problem. Six different state of the art meta-heuristic optimization approaches are used to solve the problem. The approach is evaluated on real world data.
Chapter 4 introduces two new complexity measures describing the load disaggregation problem. Until now no metric which makes it possible to compare different load disaggregation problems independent from the used classification approach did exist. Recent evaluations are done on classification results not considering the load disaggregation problem with its preprocessing stages and characteristics. In particular, a short introduction describes the problem in detail and why other commonly used complexity measures fail to describe the

load disaggregation problem. The two measures are evaluated and discussed on three different datasets, which are commonly used by the load disaggregation community.
Chapter 5 introduces a novel unsupervised load disaggregation approach. The presented approach provides advantageous characteristics such as online capability and the ability to run on embedded hardware without the need of any system or appliance information *a priori*. The chapter provides information on how to model appliances and the aggregated power load by Hidden Markov Model (HMM)s and Factorial Hidden Markov Model (FHMM)s and how the used classification/estimation process is working in general. The proposed approach is evaluated on synthetic and/or on real world data dependent on the respective case study. At the end the approach is discussed and summarized.
Finally, **Chapter 6** summarizes the work, shows and discusses the contribution of the work, points out limitations of the introduced approaches and gives an outlook on future work.

CHAPTER 2 Background to Load Monitoring

"Awareness, when managed and directed, becomes attention. By turning into attention, awareness becomes localized, and attains a focal point. Because of this feature, attention has the power to direct energy."

– Ilchi Lee

To improve people's energy awareness and to open new energy saving and redirection opportunities a comprehensive energy monitoring solution is needed The study of Armel [Arm13] showed that energy feedback on appliance level can improve energy awareness in homes the most in comparison to other feedback approaches. In general, we distinguish between two different ways to provide appliance level feedback: i) intrusive load monitoring and ii) non-intrusive load monitoring. Intrusive load monitoring uses a distributed sensor network to track the energy consumpion of appliances via sockets/plugs in homes. The advantage of simple usage is clouded by the expensive cost to purchase, maintain and communicate the data of the system. Also the power consumption of such a distributed monitoring solution has to be considered and has its impact to the total household consumption. To map the appliance type to monitored appliances, it is necessary to label the appliances. This is done either by the human or by smart algorithms doing load identification. In general, load identification tries to identify the appliance connected to the measured socket or plug by smart algorithms and characteristics of appliances.

To overcome the disadvantage of intrusive load monitoring, George Hart introduced an approach named Non-intrusive Load Monitoring (NILM). The approach, also called load disaggregation, uses a single meter trying to detect which appliance is used at which point in time having which amount of power. In the following section, we discuss NILM and load identification in detail more since these two approaches have several correspondences and are highly related to the presented work in this thesis.

2.1 Load Identification

Load identification is based on a distributed sensor network measuring the power consumption on appliance- or plug-level as in the case of intrusive load monitoring. The aim of load identification is to detect connected appliances. One of the first approaches was presented by Ruzzelli [Ruz10] who used trained appliance signatures to detect devices. In detail, the system is called RECAP (RECognition of electrical Appliances and Profiling in real-time) and uses a trained artificial neural network to classify stored appliance signatures. The approach was tested on kitchen appliances reaching an accuracy greater than 84%. Another approach was presented by Reinhardt [Rei12]. He tested different classifiers (as for example naive Bayes, Bayesian networks, random committee or random forest) and different feature sets to identify the best classifier for load identification task. Tests were performed on the Tracebase database [Rei12]. The best results were achieved by random committee with an accuracy of 95%. More recently, Ridi published in [Rid13] an approach using K-nearest neighbours and Gaussian mixture models for classification with dynamic features based on the time derivative and the time second derivative. They used the ACS-F1 database [Gis13] reaching an accuracy of up to 93.6%. In [Rid14b] and [Rid14c], Ridi also introduced an approach using hidden Markov models. Tests were done with the ACS-F1 and the ACS-F2 database [Rid14a]. Using this approach, accuracies up to 93.9% could be reached. Finally, Barker introduced in [Bar14a] Non-Intrusive Load Identification (NILI), which automatically identifies appliances connected to an outlet or plug without any human interaction. The evaluation was done on 15 common household appliances. The classification was performed by off-the-shelf classifiers such as naive Bayes, decision tree and support vector machines reaching accuracies higher than 90%.

2.2 Non-Intrusive Load Monitoring

Overcoming the disadvantage of a distributed monitoring solution, NILM was introduced as a single monitor approach. In general, NILM stresses the problem to break down the energy consumption of a home to its appliance components by a single sensor approach. The total demand is measured at the grid-connection point. Appliance characteristics and smart algorithms are used to infer which appliance is running at which point in time. In detail, the problem to disaggregate appliance readings from the aggregated power draw is composed by overlapping appliance power draws, where each appliance has a power draw $p_i(t)$. The aggregated power $P(t)$ can be formulated as the sum of each appliance's power consumption:

$$P(t) = \sum_{i=1}^{N} p_i(t). \tag{2.1}$$

Each appliance with its power draw $p_i(t)$ exhibits a unique energy consumption pattern based on electrical and usage-based characteristics which enables the load disaggregation approach to detect appliance operations in the aggregated power draw. According to appliance consumption patterns, it is possible to assign appliances to the following appliance categories [Har92, Zei11]:

- On/Off Appliances: These appliances have two operation states
- Multi-state Appliances: These are appliances having a finite number of operation states.
- Continuous consuming appliances: These appliances have a variable consumption behavior due to their operation states.
- Permanent consuming appliances: These appliances constantly consume the same energy over time.

Each of these appliances types can be found in homes as well as in commercial buildings (e.g., schools, company buildings). The main differences for the energy consumption in private and commercial buildings areas are created by the amount of used appliances, by the amount of different appliance types and by the way to use the appliances. In general, a NILM approach can be divided into a general NILM framework [Zoh12] sketeched in Figure 2.1. In the following,

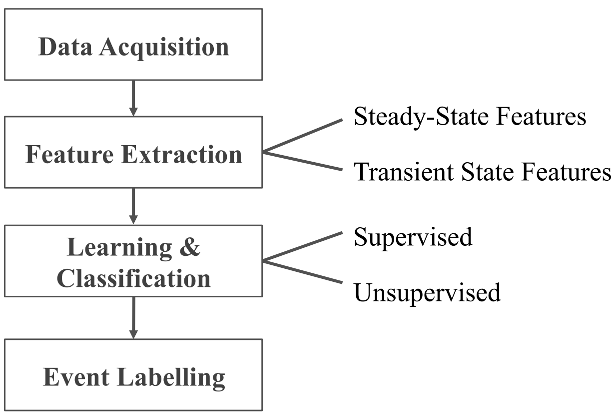

Figure 2.1: *General framework of a NILM approach*

state-of-the-art for each part of the NILM process are presented.

2.2.1 Feature Extraction and Appliance Modelling

Data acquisition is the fundamental first step in each load disaggregation approach retrieving the necessary electric data from the aggregated power consumption. The necessary monitoring units can be classified according to measurement resolution into low frequency energy meters measuring electric quantities such as active power, reactive power, apparent power, signal harmonics, etc. and high frequency energy meters capturing transient events or electric noise generated by appliances [Wan12]. According to the retrieved electric quantity, feature extraction has to be done as next step in the NILM process chain.

The task for the feature extraction is to detect appliance specific events [And12a, Cox06, Jin11, Nor96] which can be categorized into steady state and transient state features. A comprehensive overview of possible features for load signatures is provided by Liang in [Lia10].

Steady state features are considering, for example, power events switching an appliance from on to off or vice versa. Hart uses this information and recorded active and reactive power events from household appliances to model their behavior and characteristics by a simple state-machine [Har92]. We consider events as steady state events for the load disaggregation application if the sampling frequency is in the range of $1\,ms$ to $1\,h$. Many of today's smart metering entities deployed in homes fit into this range which makes load disaggregation approaches treating steady state features highly employable for future energy saving applications. Dong in [Don13] identifies power signatures of major residential loads usable for load disaggregation applications. The proposed approach uses event filtration, clustering of events, determining of authentic events and associating of different events together to reconstruct appliance characteristics, respectively. However, another example is the use of time domain characteristics of current and voltage waveforms as used in [Suz08]. He monitored the appliance waveforms for current and voltage with a sampling interval of $25\,\mu s$. In [Lau03], authors have used Fouries series analysis for detecting current harmonics with a sampling frequency of $8\,Hz$. Gupta and Chen in [Gup10] and [Che15] presented the possibility to analyse the steady state voltage noise generated by appliances equipped with motors and switch mode power supplies (SMPS). In the case of Gupta and Chen noise from $-100\,dBm$ to $-10\,dBm$ in a frequency range of $36\,kHz$ to $500\,kHz$ for static noise SMPS was considered. By using a k-means clustering a mean accuracy of 93.8% for individual device classification was stated. In contrast to [Gup10], the work of Chen [Che15] concentrates on time varying noise behaviors induced by mechanical switching (e.g., vacuum cleaner). Moreover, they used semi-supervised classification in contrast to supervised classification to decrease the required training effort. The paper stated an average accuracy of 93.8%.

Norford and Leeb [Nor96] showed that also transient events from power signals are suitable as a characteristic load disaggregation feature. Chang in [Cha10] and [Cha12] used power signatures including active, reactive power and the harmonic distortion of voltage and current signals for load disaggregation. They used a sampling frequency of $30kHz$ and showed that they achieve better classification results with transient events compared to steady state features. Another example is given by Shaw in [Sha08] who presented a transient-based approach working in the traditional AC grid as well as in a DC environment of an automobile. Finally, a high frequency approach was presented by Patel [Pat07] considering a sampling frequency up to $1\,MHz$. He used the voltage noise during switching events for load disaggregation. He tested his approach on several households and appliances (e.g., light switch, TV). He achieved an accuracy range of $85 - 90^{\%}$ with standard machine learning techniques.

2.2.2 Appliance Classification

Recent approaches solving the load disaggregation problem can be distinguished between supervised and unsupervised approaches. A good overview on supervised approaches is presented in [Zei11] and [Zoh12]. In general, supervised load disaggregation approaches need a labeled data set to train a classifier and can be divided into optimization and pattern recognition [Sha08] based algorithms. In the optimization based approaches, the problem of aggregated power profiles is modelled by an optimization problem. The total power consumption and a database of known appliance power profiles are given. With this knowledge, a random composition of appliances is selected and their power profiles are aggregated over time to approximate the total power consumption with minimal error. Baranski in [Bar04] presented a pattern detection approach based on genetic algorithm and integer programming. The algorithm was structured by the stages: event detection, fuzzy clustering, FSM creation with genetic algorithm and optimizing FSM with integer programming. The algorithm was tested with simulations and with real world data at a sampling frequency of 1s. Suzuki in [Suz08] used an integer programming approach for NILM. They used the current waveform of appliances with a sampling interval of $25\mu s$. The success rate of the waveform estimation varied between 62% (15 different appliances) and 97% (7 different appliances) according to the used appliances. In case of pattern recognition approaches, proposed methods are based on clustering methods as presented by Hart in [Har92]. Another possible example are approaches using neural networks algorithms by Srinivasan [Sri06]. Srinivasan tested several neural network based classification models including multilayer perceptron (MLP), radial basis function (RBF) network, and support vector machines (SVM) with linear, polynomial, and RBF kernels [Sri06]. He stated that

MLP and SVM both reached high accuracies in which MLP is preferable due to its low computational requirements. The classification accuracy on appliance level varied between 60% and 100% according the used number of appliances (8 or 10) and classification approach. Moreover, Lin in [Lin10] suggested a combination of Bayesian networks (including user behavior) and Bayesian filtering (classification) for NILM and compared the results to commonly used classifiers as naive Bayes, k-nearest neighbours and support vector machines. Lin showed that his approach outperforms the standard classifiers with an accuracy of 92% by using 9 different appliances and a set of 9 different features extracted from the power measurements. In general, supervised load disaggregation approaches reach a suitable and also quite high classification result. Reason for this are the necessary training phases based on *a priori* information. This fact is also the disadvantage of supervised classification approaches. In particular, in many load disaggregation problems the *a priori* information of appliances and their characteristics is available only partly or even not at all. Training is necessary, which could be costly and tedious.
Accordingly, recent research in NILM is giving focus to unsupervised algorithms, which are using unlabelled data and need no training data. Unsupervised algorithms do not need any training data and therefore, no *a priori* information of the system. In [Lia14], an unsupervised load disaggregation approach based on dynamic time warping and a supervised approach based on decision trees is presented. They used the REDD dataset and the REFIT dataset[1] for their evaluations with a sampling frequency of seconds and minutes. The results were compared with an HMM based algorithm [Par12] (success rate of 66%) and achieved a success rate of 85 – 90% with 9 appliances. A further unsupervised NILM approach was presented in [Gon11] concentrating on the clustering of power events. Recent approaches are very much interested in HMMs based algorithms. The use of the HMM based algorithm is beneficial due to their characteristic to model stationary processes with continuous valued data over discrete time. One of the first approaches was published in [Zia11]. The approach was modelled as a combined set of loads (fridge, dishwasher, microwave, computer and printer) and was evaluated with the waveforms of real and estimated waveforms. Another approach was presented by Pattem in [Pat12]. Pattem used the REDD dataset and got disaggregation results of the total energy between 56 – 67%. Due to HMMs also factorial HMMs (FHMM) are used to model the load disaggregation problem. A FHMM makes it possible to model independent appliances by decreasing the state space of problem. An approach based on FHMM is presented by Zoha in [Zoh13]. Zoha used an own set of appliances (work station, LCD, Laptop, desk lamp, table

[1]REFIT: Personalised Retrofit Decision Support Tools for UK Homes using Smart Home Technology, http://www.refitsmarthomes.org/index.php/refit-launches.

fan) with a sampling frequency of $3\,s$. Appliances were modelled as on/off and multi-state appliances reaching an accuracy of 90% for on/off appliances and 80% for multi-state appliances. Latest, different variants of FHMMs were introduced. Kolter in [Kol12] introduced additive FHMMs (the output of the FHMM is an additive function of the hidden states) and difference FHMMs (the output of the FHMM is the difference of the hidden states). By testing the REDD dataset, Kolter got an average precision of 87% and an average recall of 60% for 7 appliances. Kim in [Kim11] used so called conditional factorial hidden semi-Markov models to increase the used feature set by the features when and how appliances are used in homes. He tested his approach for 4 up to 10 appliances reaching a F-measure of 72% up to 99% dependent on the used number of appliances. Wong in [Won14] used a semi-HMM to represent more realistic appliance usage models, whose transitions are not geometrically distributed. He included state duration characteristics in the transition matrix, tested the approach on the REDD dataset (7 different appliances) and achieved an average precision of 79.9% and an average recall of 89.3%. Also modern data mining techniques are considered to solve the load disaggregation problem. Shao in [Sha12] used temporal motif mining to perform load disaggregation on private and commercial households. She used motif mining and combined her approach also with the one proposed by Kolter in [Kol12]. The combined approach reached higher performance than the individual approaches on the REDD dataset with 14 different appliances. Based on the general problem definition to separate appliance level data from the aggregated data without feedback also blind source separation techniques are recently used. Figueiredo in [Fig13] introduced an approach based on source separation via non-negative tensor factorization. She tested her approach also on the REDD dataset providing better results on the disaggregation error than the approach of Kolter in [Kol12]. Many of the presented approaches are not considering the problem how to label appliance data. Usually an unsupervised load disaggregation approach has not the knowledge of the appliance type that is tried to be detected. Approaches performing automatic labelling are conducted on Bayesian inference [Joh13] and on a semi-supervised classification [Par14, Par12]. Parson used a generic appliance model and showed that this model can be tuned to a specific appliance instance using only aggregate data.

2.3 Summary

In this chapter the related work on the load monitoring topic for homes was presented. Load monitoring can be differentiated into intrusive and non-intrusive load monitoring. These two approaches differ in the number of measurement points. Intrusive load monitoring uses a distributed sensor approach in which non-intrusive load monitoring is based on a single sensor measuring the aggregated power of a home. In the sense of intrusive load monitoring the term load identification was reviewed. The task of load identification is to identify connected appliances at a smart socket/plug with smart algorithms. This approach is highly related to NILM in which NILM tries to identify multiple connected appliances and load identification tries to identify one connected device. In this chapter, we provide a comprehensive review of the state-of-the-art for NILM. We divided the review by the general framework of a NILM approach consisting of data acquisition, feature extraction, classification and event labelling. We showed that we have to distinguish between steady-state based and transient-based approaches. The sampling frequency correlates with the number of extractable features. The higher the frequency, the more significant are the electrical features in general. This also applies if more than one electric quantity as for example active and reactive power are considered by NILM. The more features are used, the more information can be included in an appliance model and accordingly, makes the model more significant. We then provided also a review on different classification techniques divided into supervised and unsupervised load disaggregation techniques. One of these two general classification families have to be used dependent on the presence of *a priori* information (number of appliances known or not, appliance characteristics known or not, etc.). To conclude, we claim that there exist no general solution for NILM. Each proposed NILM technique is a solution for a certain application and different system characteristics. Terms such as considered appliance type (on/off, multi-state, etc.) number of appliances, degree of *a priori* knowledge (supervised vs. unsupervised) and sampling frequency (the higher the sampling frequency, the higher the extractable electrical features) are the main distinctive features and dependencies between different NILM approaches.

CHAPTER 3 Load Disaggregation as a Knapsack Model

"We can't solve problems by using the same kind of thinking we used when we created them"

– Albert Einstein

In this chapter, we are concentrating to model the problem to disaggregate appliance power data from the aggregated power draw by an optimization approach modelled as a knapsack. The aim is to optimize an objective function retrieving the information which appliance is running at which point in time. The first approach on this topic was proposed by Hart [Har92] where he stated that the load disaggregation problem can be modelled as a subset sum problem. He faced the problem that even if all power states of appliances are known small fluctuations and similarities between power states lead to a dramatical decrease in the performance of the approach. To verify the statement provided by Hart we model the load disaggregation problem as a knapsack problem. To solve this problem, we aim to use state-of-the-art metaheuristic optimization approaches such as i) the *evolutionary algorithm*, ii) *differential evolution*, iii) *particle swarm optimization*, iv) *simulated annealing*, v) the *cuckoo search algorithm* and vi) *firefly optimization* due to their ability to solve the knapsack problem. We evaluate the performance of the load disaggregator on real world data in which similar and realistic consumption behaviors are present.

The remainder of this chapter is organized as follows: Section 3.1 provides a comprehensive summary of all used metaheuristic optimization approaches. Section 3.2 describes the proposed approach and how to model the load disaggregation problem as a knapsack problem. In Section 3.3 the evaluation settings are presented. Section 3.4 provides evaluations of case studies to evaluate the proposed approach. Finally, Section 3.5 discusses the presented results and Section 3.6 summarizes the chapter. Parts of this chapter are based on the published works [Ega13c] and [Ega13b].

3.1 Meta-Heuristics

Every time we are trying to maximize the profit or to optimize a problem, we are optimizing an objective function. This objective function can be described mathematically and solved by different techniques. Modern search algorithms can be divided into deterministic and stochastic algorithms. Deterministic search algorithms are reproducible as stochastic algorithms having always some randomness. In this work, we are concentrating on stochastic algorithms based on metaheuristic algorithms. Basic operation is the random walk, which searches randomly for solutions in the vicinity of a given solution. In contrast to heuristics, which are problem dependent, metaheuristic algorithms are following a black box scheme. Metaheuristic algorithms provide problem-independent techniques in which a deterioration of a solution is acceptable and the algorithm parameters have to be adopted to the problem. Accordingly, metaheuristic algorithms are trying by trial and error to get the best solution in a reasonable time without a guarantee of the best result. They have a randomization and local search part, in which the algorithms are characterized by the ability to balance between exploitation of the best solution found and exploration of new solutions. Exploration means to generate several possible solutions for the objective function and exploitation means to find local best solutions. The best solutions are recombined and kept to ensure the convergence towards an optimal solution. In the past, several metaheuristic algorithms were introduced. This work concentrates on six different metaheuristic algorithms trying to solve a simplified load disaggregation problem as objective function. We are considering i) the *evolutionary algorithm*, ii) *differential evolution*, iii) *particle swarm optimization*, iv) *simulated annealing*, v) the *cuckoo search algorithm* and vi) *fire fly optimization*. In the following, each metaheuristic approach is described on their functioning and characteristics.

3.1.1 Evolutionary Algorithm

The Evolutionary Algorithm (EA) is a population-based optimization approach inspired by the evolution of natural life [Eib03]. A set of individuals represents a population. The algorithm aims at optimizing the population according to a fitness or objective function over several generations. The individuals are modified by the evolutionary operators *mutation* (mutation of individuals), *recombination* (combination of individuals) and *selection* (selection of individuals, which will survive, i.e., remain in the population for the next generation). The first task of an evolutionary algorithm is to encode the optimization function. In the case of the genetic algorithm an array of bits is used to encode the problem in which the evolutionary algorithm uses the decision variable and the problem function. Next step is to define a fitness function providing a feedback of the optimization process solving the given problem. The optimization process is performed several generations until the stopping criteria for the algorithm is met (e.g., max number of iterations, fitness result is optimal). Each iteration a new population is generated. The main part of the algorithm is the manipulation of the populations and the selection of solutions according to their reached fitnesses. To manipulate the populations, the operators mutation and recombination are used. The mutation operator modifies a randomly chosen bit of the population with a probability p_m. The crossover operator represents the recombination of two parent strings. Two strings of the population are chosen and two segments of the strings are created. These strings are swapped by a probability p_c between the two parent strings. The selection of solutions are based on the achieved fitness performance. In general, the best solutions are reused in the next generation. This process is called elitism strategy since the fittest solutions are surviving a generation or iteration. The pseudo code of the EA is shown in Algorithm 1 [Yan10b]. The EA has several parameters and design choices. For

Algorithm 1 Evolutionary Algorithm

Objective function $f(x)$, $x = (x_1, \dots, x_d)$
Encode the solution into chromosomes (binary strings)
Define fitness F (e.g., $F \propto f(x)$ for maximation)
Generate the initial population
Initial probilities for crossover p_c amd mutation p_m
while t < max number of generations **do**
  Generate new solution by crossover and mutation
  if $p_c > rand$ **then**
    Crossover
  end if
  if $p_m > rand$ **then**
    Mutation
  end if
  Calculate fitness
  Select best solution for next generation (elitism)
end while
Decode result

example different mutation or crossover strategies can be applied. Moreover, the number of populations and the number of generations can be modified. The parametrization and the design choices for each EA are the crucial part. The choice of parameters and operator strategies are often application and problem dependent.

3.1.2 Differential Evolution

The Differential Evolution (DE) [Das11] is a special form of the EA. Each population of a generation consists of a candidate solution of the objective function. The algorithm maintains candidate solutions and creates new solutions by combining existing ones based on simple formulas. The candidate solutions are evaluated based on a given fitness or objective function as for the evolutionary algorithm. In particular, the algorithm perform the steps mutation, recombination and selection. The first task is the mutation step choosing for each generation vector x_i three distinct vectors x_p, x_q and x_r. With these vectors a so-called donor vector is created by:

$$v_i^{t+1} = x_p^t + F(x_q^t - x_r^t). \tag{3.1}$$

The parameter F is the differential weight and usually in the range $[0, 2]$. In the crossover stage, the candidate solutions are modified as follows:

$$u_{j,i}^{t+1} = \begin{cases} v_{j.i}^t, & \text{if } r_i \leq C_r \\ x_{j,i}^t, & otherwise. \end{cases} \tag{3.2}$$

The parameter r_i is a uniform distributed random number and C_r is the crossover probability. The equation determines if a component of the candidate vector is replaced or not. Finally, the algorithm performs a fitness evaluation of the solutions to find the fittest candidates in the population. A pseudo-code description of the algorithm is presented in Algorithm 2 [Yan10b].

3.1.3 Particle Swarm Optimization

The Particle Swarm Optimization (PSO)[Wei04] is a population-based algorithm, in which the population with its candidate solutions is represented as a swarm of particles. The aim of the swarm is to move around the search space guided by their position and the position of the best candidate solution. A particle is attracted by the best particle g and its own best x_i^* in history. A movement of a particle is performed if a particle finds a better position (better result of the objective function) than the current one. Moreover, all particles are evaluated on its performance to find the global best solution g. The search of the global best particle lasts until either a maximum number of iterations is reached or the global best solution is not improving any more. The movement of the particles is determined by the current position x_i and a velocity vector v_i^{t+1} defined as:

$$v_i^{t+1} = v_i^t + \alpha\epsilon_1 \odot [g - x_i^t] + \beta\epsilon_2 \odot [x_i^* - x_i^t]. \tag{3.3}$$

The variables ϵ_1 and ϵ_2 are random vectors between 0 and 1. The operator $\odot$ represents the entrywise product of two vectors. The factors α and β are

Algorithm 2 Differential Evolution

Objective function $f(x)$, $x = (x_1, \dots, x_d)$
Initialize the population x with randomly generated solutions
Set the weight $F \in [0, 2]$ and crossover probability $C_r \in [0, 1]$
while stopping critertion **do**
  for $i = 1$ to n **do**
    For each x_i randomly choose 3 distinct vectors x_p, x_q and x_r
    Generate a new vector v
    Generate a random index $J_r \in \{1, 2, \dots, d\}$ by permutation
    Generate a randomly distributed number $r_i \in [0, 1]$
    for $j = 1$ to d **do**
      For each parameter $v_{j,i}$ (j-th component of v_j), update
      if $r_i \le C_r$ or $j = J_r$ **then**
        $u_{j,i}^{t+1} = v_{j,i}^{t+1}$
      else
        $u_{j,i}^{t+1} = x_{j,i}^{t}$
      end if
    end for
    Select and update the solution
  end for
  Increment counter t
end while

the so called learning parameter or acceleration constant. The initial values at $t = 0$ can be set to 0 for x_i and v. The pseudo code of the PSO is presented in Algorithm 3 [Yan10b].

Algorithm 3 Particle Swarm Optimization

Objective function $f(x)$, $x = (x_1, \dots, x_d)$
Initialize the location x_i and the velocity v_i of n particles
Initialize minimum $f_{min}^{t=0} = min f(x_1, \dots, x_n)$ (at $t = 0$)
while criterion **do**
  for loop over all n particles and all p dimensions **do**
    Generate new velocity v_i^{t+1}
    Calculate new location $x_i^{t+1} = x_i^t + v_i^{t+1}$
    Evaluate objective function at location x_i^{t+1}
    Find the minimum f_{min}^{t+1}
  end for
  Find current best x_i and global best g
end while

3.1.4 Simulated Annealing

The Simulated Annealing (SA)[Haj85] approach is inspired by the annealing of metallurgy, where metal is heated up and slowly cooled down to strengthen the metal structure by rearranging the crystal structure. In general, simulated annealing is random search based. It considers changes of information improving the objective function as well as information changes not be ideal. Each solution decreasing the objective function (in case of a minimization problem) is accepted. Moreover, also changes increasing the objective function are accepted with a probability p. This probability p is defined as:

$$p = \exp(\frac{-\Delta E}{k_B T}), \tag{3.4}$$

where k_B is the Bolzmann's constant, T the temperature to control the annealing process and ΔE the change in the energy level. The energy change ΔE is described by:

$$\Delta E = \lambda \Delta f, \tag{3.5}$$

where Δf is the objective function to be optimized and λ is a constant. By setting $k_B = 1$ and $\lambda = 1$, the probability p becomes:

$$p = \exp(-\Delta f / T) \tag{3.6}$$

The next step is then to define a random number r to decide if p should be accepted or not. This is done be checking:

$$p = \exp\left[-\Delta f / T\right] > r. \tag{3.7}$$

This optimization approach improves its candidate solution over a number of simulation steps in the cooldown process. In contrast to EA, DE, and PSO, there is no influence between the selection and development of candidate solutions in relation to the other candidates, thus SA is not population-based. Another important parameter of annealing process is the starting temperature T_0. The control of the annealing and cooling process is commonly done linear or geometric [Yan10b]. The final step of SA is how to set the number of used iterations N. Commonly used possibilities are to set N fixed or variable (dependent on the current temperature) to achieve a desired solution quality. The pseudo code of the simulated annealing process is presented in Algorithm 4 [Yan10b].

3.1.5 Cuckoo Search Algorithm

The Cuckoo Search Algorithm (CS)[Gan13] is inspired by the brood parasitism of cuckoos laying their eggs into nests of other birds of different species. An egg

Algorithm 4 Simulated Annealing

Objective function $f(x)$, $x = (x_1, \dots, x_d)$
Initialize initial temperature T_0 and initial guess $x^{(0)}$
Set final temperature T_f and max number of iterations N
Define cooling schedule $T \mapsto \alpha T$, $(0 < \alpha < 1)$
while $T > T_f$ and $n < N$ **do**
  Move randomly to new location $x_{n+1} = x_n + \epsilon$ (random walk)
  Calculate $\Delta f = f_{n+1}(x_n + 1) - f_n(x_n)$
  Accept the new solution if better
  if not improved **then**
    Generate random number r
    Accept if $p = \exp\left[-\Delta f / T\right] > r$
  end if
  Update the best x and f
  $n = n + 1$
end while

in a nest is representing a solution of an optimization problem and a cuckoo egg stands for an new solution. In general, each nest has one egg which could be extended to several eggs per nest dependent on the problem and representation. Based on the representation, the algorithm is following the subsequent ideas as described in [Yan10b]:

- Each cuckoo places one egg at a time, and randomly puts it in a desired nest.
- The nests are qualified according to their nest and high quality nests will survive form one generation to the next generation.
- The number of available hosts nests is defined. The egg is laid by a cuckoo and is discovered by the host bird with a probability $p_a \in (0, 1)$. In this case, the egg is discovered, the host bird either get rid off the egg or build a completely new nest leaving the old nest.

The task of CS is now to find good and better cuckoos (solution) and replace bad eggs with the better solutions. The pseudo code of the CS is presented in Algorithm 5 [Yan10b].

For the random generation of a new solution/nest $x^{(t+1)}$ a Levy Flight is performed and is represented as:

$$x_i^{(t+1)} = x_i \bigoplus Levy(\alpha). \tag{3.8}$$

Algorithm 5 Cuckoo Search

```
Objective function f(x), x = (x_1, ..., x_d)
Generate an initial population of n host nests
while (t < maxGeneration) or (stop criterion) do
    Get a cuckoo randomly/generate a solution i by Levy flights
    and evaluate its fitness F_i
    Choose a nest among n (say j) randomly
    if F_i > F_j then
        Replace nest j by the new solution i
    end if
    A fraction p_a of worse nests are abandoned and new ones/solutions are
    generated
    Keep best solutions/nests
    Rank the solutions/nests and find the current best
    Pass the current best solutions/nests to the next generation
end while
```

The variable α represents the step size and $\oplus$ a piecewise multiplications. The use of the Levy Flight is beneficial since it is more efficient to explore a search space as a simple random walk. In detail, the Levy Flight based on the random walk having a step size based on the Levy distribution:

$$Levy \sim u = t^{-\lambda}, \tag{3.9}$$

where $1 < \lambda \leq 3$.

In summary, the CS has three major characteristics. First, the algorithm is population-based. Second, the randomization is efficient because of using a step size based on Levy flights. Third, the parameters to be tuned for the optimization process are the number of nests n and the parameter p_a. The suggested choice of the parameter for the nest is $n = 15$ to 40 and for the discover probability is $p_a = 0.25$. The last characteristic represents also the benifical advantage to use CS since GA and other metaheuristic approaches are based on more parameters in which CS produces similar or better optimization results [Yan10b].

3.1.6 Firefly Algorithm

The Firefly Algorithm (FA)[Yan10a] is inspired by the flashing behavior of fireflies. The aim is to flash synchronous to attract other fireflies. The attractiveness is proportional to the brightness of a firefly in which the brightness represents the objective function. In detail, the FA is based on the following rules [Yan10a]:

- As opposed to nature, all fireflies are unisexual. One firefly will be attracted by all other fireflies.
- Attractiveness is proportional to their brightness, thus for any two flashing fireflies, the less brighter one will move towards the brighter one. The attractiveness is proportional to the brightness and the both decrease as their distance increases.
- If there is no brighter one than a particular firefly, it will move randomly.
- The brightness of a firefly is affected or determined by the landscape of the objective function

The basic principle as described is presented as pseudo code in Algorithm 6 [Yan10b].

Algorithm 6 Firefly Optimization

```
Objective function f(x), x = (x_1, ..., x_d)
Generate an initial population of fireflies x
Light Intensity I_i at x_i is determined by f(x_i)
Define a light absorption coefficient γ
while (t < maxGeneration) do
  for i = 1 : n all n fireflies do
    for j = 1 : n all n fireflies (inner loop do
      if I_i > I_j then
        Move firefly i towards j
      end if
      Vary attractivness with distance r via exp[−γr]
      Evaluate new solutions and update light intensity
    end for
  end for
  Rank fireflies and find the current best
end while
```

In detail, the brightness stands for the objective function of the optimization problem. The attractiveness is proportional to the light intensity recognized by neighboring fireflies where the attractiveness β is formulated as:

$$\beta = \beta_0 \exp(-\gamma r^2). \tag{3.10}$$

The variable β_0 represents the attractiveness at $r = 0$. The movement of the firefly i attracted by another firefly j is formulated as:

$$x_i^{t+1} = x_i^t + \beta_0 \exp(-\gamma r_{ij}^2)(x_j^t - x_i^t) + \alpha_t \epsilon_i^t. \tag{3.11}$$

The second term of formula 3.11 is the attraction from firefly j on firefly i. The third term represents the randomization where α_t is the randomization parameter and ϵ_i^t is a vector of random numbers generated by a Gaussian or uniform distribution at time t. The randomization factor α_t can be well described by:

$$\alpha_t = \alpha_0 \delta^t, \tag{3.12}$$

where $0 < \delta < 1$ and α_0 is the initial randomness factor. In [Yan10b], it is recommended to use $\alpha = 0.95$ to 0.97. The factor β_0 should be chosen between 1 and γ dependent on the problem [Yan10b]. Finally, the population size should be usually chosen as $n = 15$ to 100 [Yan10b].

By setting $\beta_0 = 0$, the optimization process becomes a simple random walk in which by setting $\gamma = 0$, the process becomes a variant of the particle swarm optimization [Yan10b].

3.2 Approach

The power demand of households depends on the used appliances. Typically, each appliance has a characteristic way of consuming energy. For example, one appliance is consuming a high amount of energy for a short period of time or another appliance behaves in a multi-state manner consuming in each running state a different amount of power. The total power load can be considered as the superposition or the aggregation of the power profiles from each appliance over time. G. Hart (et. al [Har92]) provided the idea to use the knowledge of appliance characteristics and the aggregated power demand to introduce the problem of load disaggregation or non-intrusive load monitoring (NILM). NILM breaks down the aggregated power demand to its components on appliance level. He classified the problem to disaggregate appliances as computational intractable belonging to the group of NP-complete problems. An example for the aggregated total power load and the disaggregated loads on the appliance level is shown in Figure 3.1. The aim of the load disaggregator is to find the best composition of appliance power states minimizing the error between an estimated signal and the real signal. In this context, the major problem for NILM is that appliance power states used by a load disaggregation approach are noisy (detected power states are not perfect; appliance states are varying in some ranges over time) and that the measured total power load used by the load disaggregation approach is influenced by measurement noise. Accordingly, a load disaggregation approach has to deal and to overcome noise influences to find the best composition of appliance power states. A possible approach is to perform load disaggregation based on optimization which tries to find the best solution with known information.

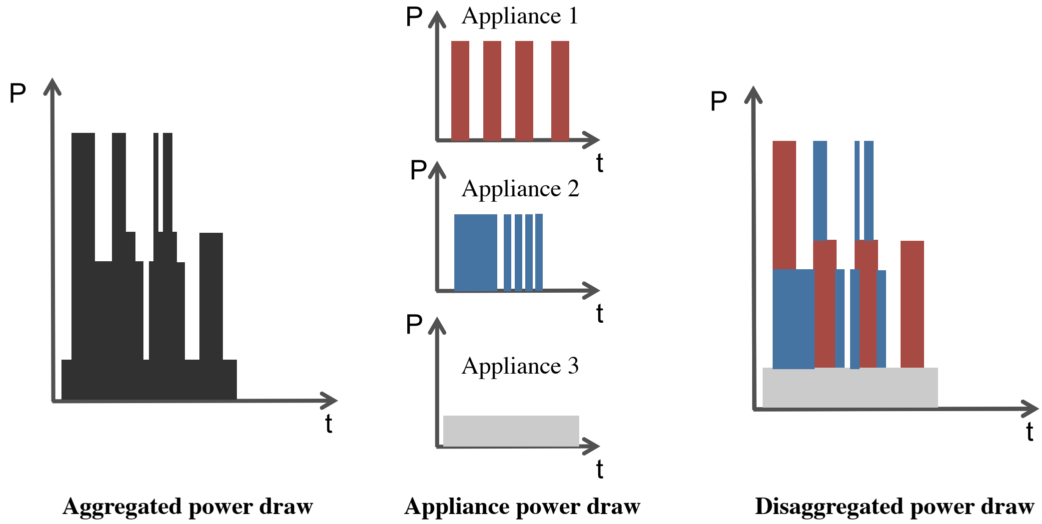

Figure 3.1: *The aggregated power draw is created by the aggregation of each appliance power profile in a house. Each appliance as in the figure has a different way of consuming power in which each appliance is operated in different times of day. The aggregated power draw corresponds to the superimposition of this appliance created power draws.*

In this chapter we concentrate on a supervised approach solved by optimization. Optimization is a feasible approach due to their ability to find solutions in a noise influenced environment. The first task is to define an objective function to be optimized. The aggregated power draw $P(t)$ can be modelled by:

$$P(t) = p_1(t) + p_2(t) + \cdots + p_n(t) \text{ for } t \in \{1, T\}, \tag{3.13}$$

where $p_i(t)$ is the power profile of each appliance in the set of N appliances and t represents the discrete time vector from 1 to T. The problem is to find the best set of appliance power profiles whereas each power profile is activated by an appliance being in a on or off state. We can model the problem as:

$$e(t) = \left| \left(P(t) - \sum_{i=1}^{N} p_i \cdot a_i(t) \right) \right| \tag{3.14}$$

where $a_i(t)$ represents the appliance state vector (e.g., appliance is on or off). In this work, we are modelling the presented optimization problem as the so-called knapsack problem [Sal75, Ega13c]. The knapsack problem is a well-known optimization problem with the aim of packing a set of n items with a certain weight w_i and profit d_i into a knapsack of capacity C in the most profitable way. If it is possible to place a item into the knapsack without exceeding the capacity C by using $x_i \in \{0,1\}$, which is responsible for whether or not a certain item is used, a profit d_i is earned. This context can be summarized as follows:

$$\text{maximize} \sum_{i=1}^{n} d_i \cdot x_i, \tag{3.15}$$

$$\text{subject to} \sum_{i=1}^{n} w_i \cdot x_i \leq C. \tag{3.16}$$

The problem of packing items into a desired shape can be adopted to the load disaggregation problem. NILM has the aim to disaggregate loads from the aggregated power demand according to their own power profile p_i in the measured total load $P(t)$. The power profiles p_i are mainly characterized by their power magnitude m_i and their time of usage. The total power load is given by:

$$P(t) = \sum_{i=1}^{n} P_i \cdot a_i(t) + e(t), \tag{3.17}$$

where n is the number of known and used appliances, $a_i(t) \in [0,1]$ represents the state vector of the appliance being on ($a_i(t) = 1$) or off ($a_i(t) = 0$). $e(t)$ describes an error term. The general optimization problem of NILM can be formulated as the minimum error $e(t)$ of the total power load and the estimated aggregation of appliance power profiles:

$$e(t) = \arg\min \left| P(t) - \sum_{i=1}^{n} P_i \cdot a_i(t) \right|. \tag{3.18}$$

The NILM system tries to find the appliance states by $a_i(t)$ to minimize the error between the sum of superimposed appliance power profiles and the total load $P(t)$. This relates to the knapsack problem, where in the case of NILM the capacity C of the knapsack corresponds to the total load $P(t)$ and the items of the knapsack correspond to the appliance power profiles P_i. We assume that the profit d_i equals 1 since we suppose that all appliances in the household are of equal importance concerning their usage. The aim of any optimization approach is to find a composition of power profiles P_i, which can be packed into the measured total load $P(t)$ with minimum error. We modify the general knapsack problem by dismissing the profit maximization with an error minimization. An illustration of the basic principle can be seen in Figure 3.2, where a collection of possible power profiles P_i and the trend of the total power load are presented. In detail, the approach tries to find for each point in time the best composition of appliance power profiles described by their power demands. The optimization approach has to optimize the vector $a_i(t)$ represented as a binary vector. The

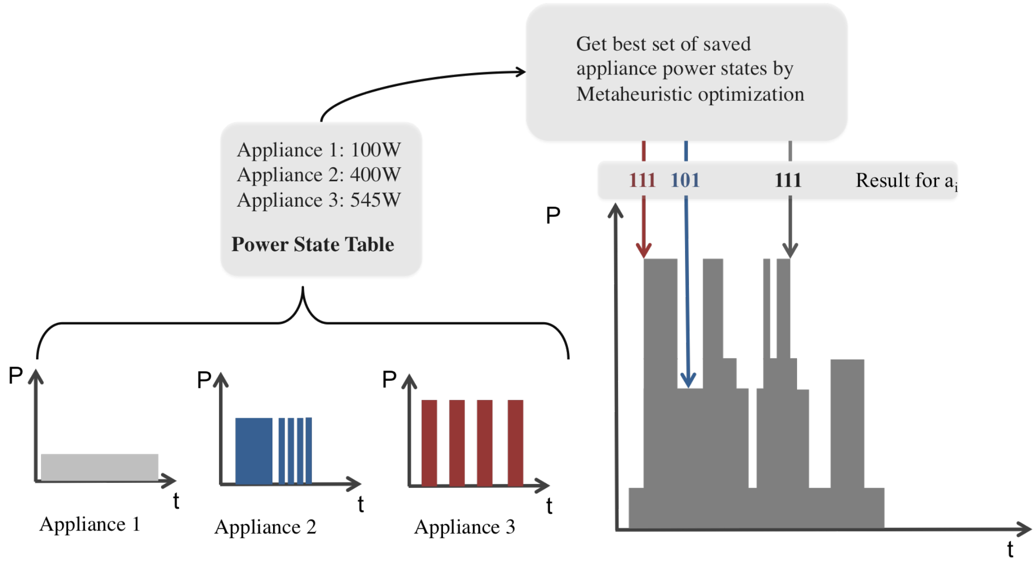

Figure 3.2: *The approach estimates the optimal set of appliance power states for each used time sample with the table of possible power states*

value 1 means an appliance is on at time t and 0 means an appliance is off at time t. The objective function is represented by:

$$F_s = -\left| P(t) - \sum_{i=1}^{N} P_i \cdot a_i(t) \right|. \tag{3.19}$$

It describes a minimization problem with an optimal fitness of 0. The optimization process of the state vector $a_i(t)$ is done on each power sample. We used six different optimization approaches to solve this problem consisting of the i) the *evolutionary algorithm*, ii) *differential evolution*, iii) *particle swarm optimization*, iv) *simulated annealing*, v) the *cuckoo search algorithm* and vi) *fire fly optimization*.

3.3 Settings

3.3.1 Algorithm Settings

All metaheuristic optimization approaches are modified to work with discrete inputs. This is mainly done by rounding the results from the continuous case to the discrete values 0 and 1. However, the used metaheuristics are dependent on different parameters. In Table 3.1 all parameters are presented. The parameters were chosen based on empirical evaluations.

Algorithm	*Parameter*	*Value*
EA, DE, PSO FA	No. of generations g	200
EA, DE, PSO	Population size p	100
EA,	Mutation operator	uniform mutation
EA,	Recombination operator	one-point crossover
EA,	Selection operator	elite selection
DE	Crossover probability	0.5
DE	Scaling factor	0.8
PSO	Coparmnitive parameter $c1$	2
PSO	Social parameter $c2$	1
PSO	Constriction parameter C	1
SA	Cooling steps	200
SA	Maximum intial temperature	100
CS	Number of nests	50
CS	Discovery rate	0.25
FA	Number of firefies	50
FA	Randomness factor	0.9
FA	Randomness reduction factor	0.95
FA	Absorption coefficient	0.2

Table 3.1: *List of parameters for each used metaheuristic optimization approach*

3.3.2 Data Settings

To test the proposed approach real-world measurements of appliances are used. There exist several public available datasets such as the REDD dataset [Kol11], the Eco-dataset [Bec14], the GREEND dataset [Mon14a] and the AMPD dataset [Mak13] (see Section 4.2), which are suitable data sources for an evaluation. We have chosen the REDD dataset as reference dataset because it is well known and one of the most popular datasets in the NILM community and meets the requirements having appliance-level measurements with low sampling rates. The dataset provides appliance level power measurements in $1s$ resolution for 6 different houses. For our evaluation, we took the first house with 6 common appliances (oven, fridge, dishwasher, kitchen outlet, microwave and washing dryer). We derived for each appliance the present power states. We distinguished between automatic detected power states and power states detected by expert knowledge. In the case of automatic detected power states, we used the identified states from Table 4.5. The states are detected from submetered power draws where similarities between power states are possible. In Table 3.2 the used appliances and their characteristics are listed. For the other case, detected power states are identified by the human. Small power states such as standby power are considered in contrast to automatic detected power states.

3.3.3 Evaluation Metric

To be able to evaluate the performance of the metaheuristic knapsack approach, we evaluate the energy consumption for each appliance and compare it to the ground truth energy data. The power draw for each power state is optimized individually. The power states and their resulting optimization results belonging to an appliance are grouped and compared to the ground truth, respectively.

3.4 Case Studies

To check the applicability of the metaheuristic based load disaggregation approach we introduce two case studies i) appliance set with ambiguous power values and ii) appliance set with unambiguous power values.

3.4.1 Appliance Set with Ambiguous Power Values

In this case study, we decided to use real world consumption data in which appliances have similar consumption behaviour. Thus, the amount of power consumed by an appliance A can be similar to the one consumed by appliance B. The case study should show if the approach is able to distinguish between appliances even if they have similar and ambiguous power demands or if the combination of power states leads to another power state. In case of the REDD dataset, we used 6 appliances of house 1 in which we identified the following appliance states for each appliance listed in Table 3.2 as utilizable. The input for each metaheuristic approach are the power states listed in Table 3.2 and we used an observation window of one day.

In Figure 3.3, 3.4 and 3.5 the energy shares of the approaches and in Figure 3.6 the ground truth energy shares are presented. The energy shares of the washing dryer are not shown since the device was turned off during the whole observation time and the algorithms always detected this fact. We claim detecting an appliance to be off is of the same difficulty as to detect an appliance to be on. There exists no preferred state for the optimization process.

However, the results show that the approach is not able to distinguish between different appliances. It is able to track and to optimize the problem.

type	states	power in Watt
A1: oven	3	[0 1690 2455]
A2: fridge	2	[0 190]
A3: kitchen outlet	5	[0 210 440 880 1100]
A4: microwave	3	[0 60 1533]
A5: stove	4	[0 260 710 1440]
A6: washing dryer	2	[0 2712]

Table 3.2: *Table of used appliance types, the number of operation states and the corresponding power values for each operation state for the case study of ambiguous power values.*

The mean error between optimized and real power draw is around $13\,W$ and is in an sufficient and satisfying range. Nevertheless, the similarity of power states, the possible representation of an power state by a combination of other states and noise effects are influencing the problem heavily. The appliances oven, fridge and kitchen outlet have similar power states heavily influencing the optimization result. This influence is presented comparing the energy share of the fridge and the kitchen outlet with real energy shares, respectively.

Also the error in total is high. As reason we claim the influence of noise effects and perfectly modelled appliance states. Moreover, the influence of different metaheuristics algorithms can be neglected. All algorithms had this problem and produced similiar results.

3.4.2 Appliance Set with Unambiguous Power States

In contrast to the previous section we are now considering appliance states which occur unambiguously. Similarities between appliances are only possible for very small power values such as the standby power. In Table 3.3 all possible power states are listed. Each state was empirically identified by human. The input for the metaheuristic load disaggregator are the power states listed in this table and we used an observation window of one day.

Figure 3.6 presents the estimated and the real energy share. The results are improved compared to the previous case study. Clearly distinguishable power states as for the oven and the fridge can be estimated very well. Nevertheless, also in this case study the effect of similar/recombined of power states and noise effects are present and affecting the results. In this evaluation we took the energy share of the evolutionary algorithm as representative case since all other approaches achieved nearly the same result. The only exception was the result

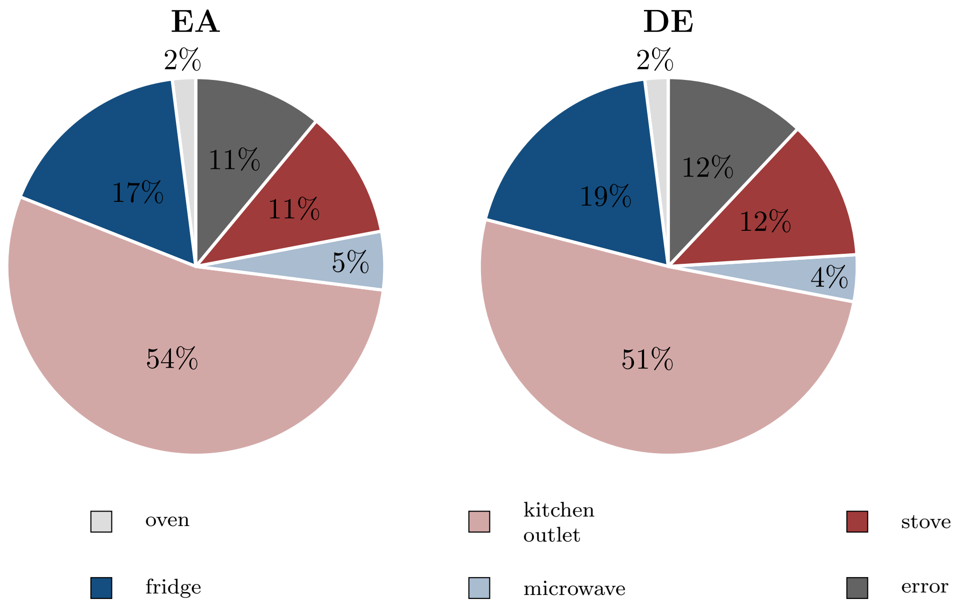

Figure 3.3: *Energy shares for the optimization results of EA and DE with similar power states*

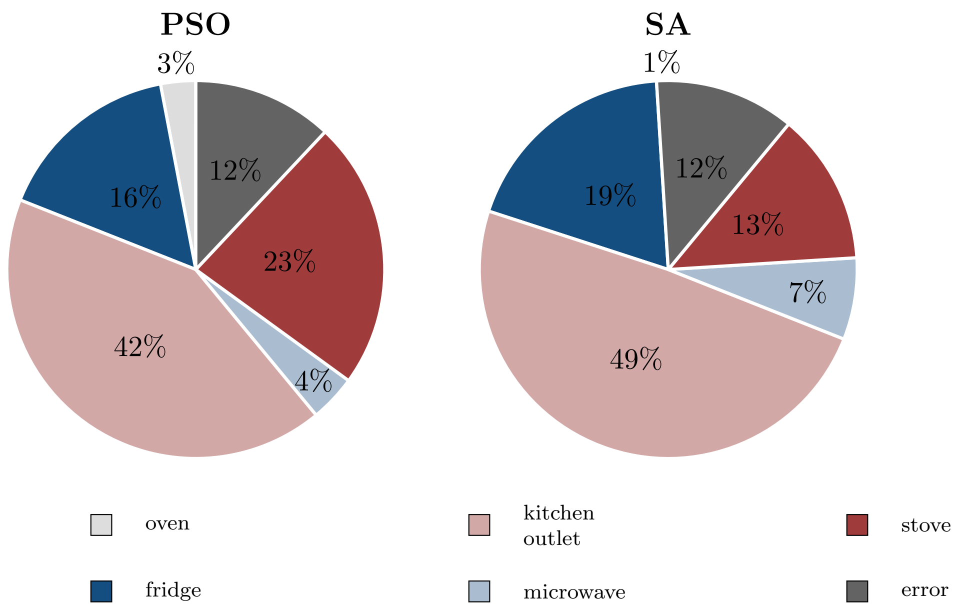

Figure 3.4: *Energy shares for the optimization results of PSO and SA with similar power states*

for the simulated annealing case study. In this case study the oven reached only 5% and the stove 14%. The other estimates were very similar. As in the previous case study, the washing dryer is not shown in this figure since it was

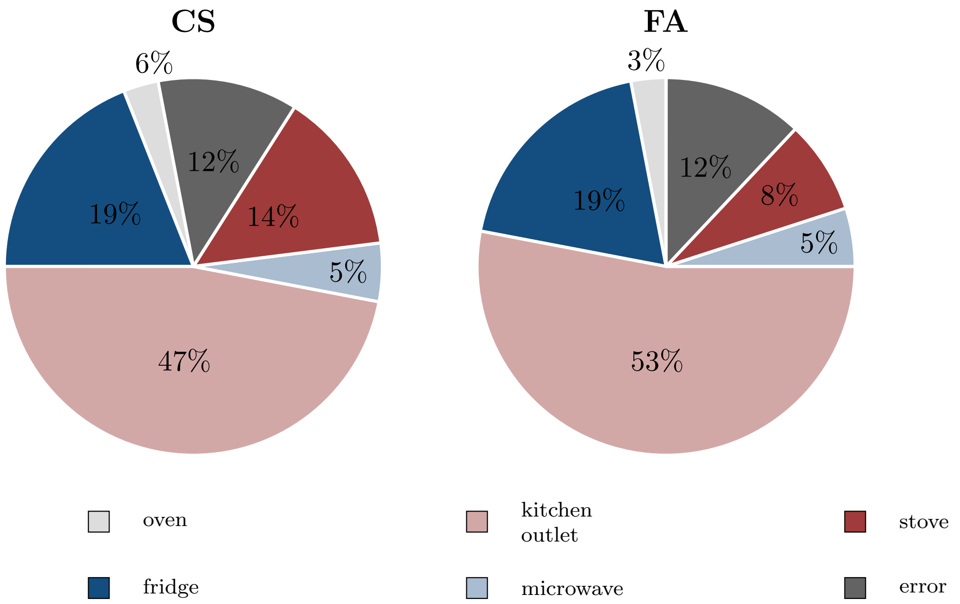

Figure 3.5: *Energy shares for the optimization results of CS and FA with similar power states*

type	states	power in Watt
A1: oven	2	[0 1600]
A2: fridge	4	[0 8 190 2000]
A3: kitchen outlet	2	[0 1080]
A4: microwave	3	[0 5 1550]
A5: stove	2	[0 1430]
A6: washing dryer	2	[0 2700]

Table 3.3: *Table of used appliance types, the number of operation states and the corresponding power values for each operation state for the case study of unambiguous power values.*

not used in the observed time and was well detected by all approaches.

Moreover, the total error of the optimization result is getting better as well as the mean error between optimized and real power draws over the time (error of $7\,W$). As reason we claim the fact of different/dissimilar power states and the lower number of possible power states. The lower the number of considered power states, the better is the result of the optimization.

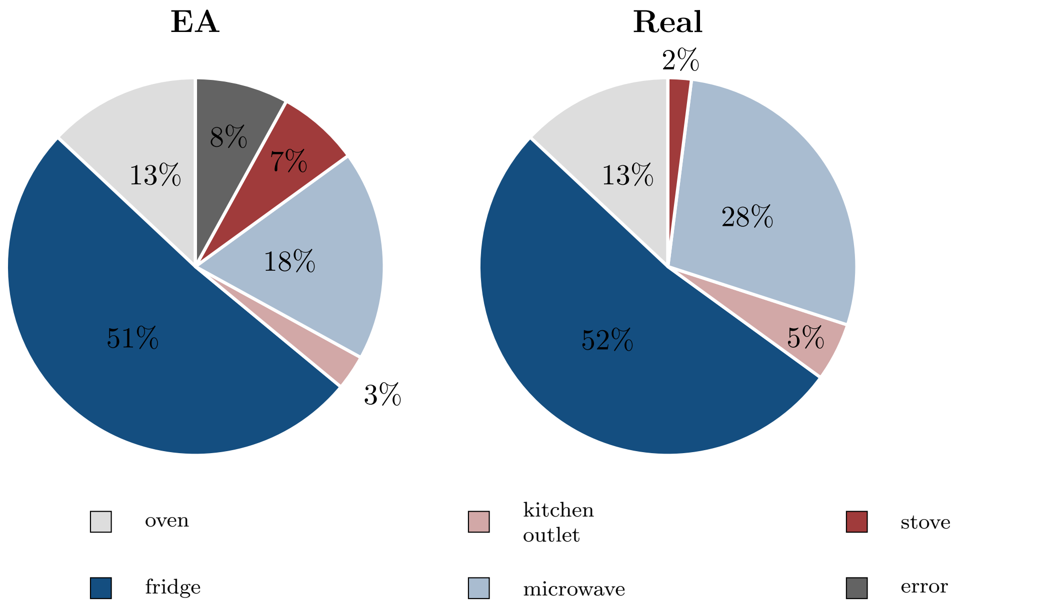

Figure 3.6: *Energy shares for the optimization results of EA with unique power states and the ground truth energy shares*

3.5 Discussion

The presented approach was not able to distinguish multiple power states which are similar to each other or can be recombined by other power states. The reason for this is expected to be the lack of information based on the use of one feature (in our case the power value of a state). From this we conclude that the problem has to be modelled by more advanced techniques including appliance structure (e.g., state machine), timing behaviors and probabilistic representation. All presented metaheuristic approaches achieved similar results. The choice of the algorithm is based on the computational time, the set of selectable parameters and the chosen applications. The approach is suitable for NILM applications with a low number of appliances with different power states. A possible application would be a power plug or multiple power plugs to detect attached appliances.

3.6 Summary

In this chapter a simple load disaggregation approach based on metaheuristic optimization modelled as a knapsack problem is presented. Six different metaheuristic algorithms (evolutionary algorithm, differential evolution, particle swarm optimization, simulated annealing, cuckoo search algorithm, firefly optimization) were tested according to their ability to disaggregate loads from the total power demand. The approach uses a simple set of appliance power set and tries to find out of this set the optimal composition of appliance power states to minimize the error between the estimated and real power draw. The approach is related to the well-known knapsack problem and modified according to problem specific characteristics. The approach was tested on real-word measurements in which we varied the used set of appliance power states. The results show that the algorithm provides satisfying results for an appliance set where almost no similar appliances are present. For similar appliances the approach was not able to disaggregate appliance power draws due to lack of further information which would be necessary to distinguish between similar appliances. The problem for the algorithm are power states and their combinations between power states. It is very probable that power states are combinations of other power states and the presented approach tries only to find these combinations. The algorithm has no knowledge of appliance characteristics such as structure or time behaviors. Load disaggregation approaches using model representation such as hidden Markov models as in [Kol11] or [Pat12] are performing better and are more flexible with noise influences. Accordingly, the presented approach is suitable for simple load disaggregation tasks. To achieve better load disaggregation results the approach has to be improved by enlarging the feature set of the optimization (e.g. using multi-objective optimization) or by a different objective function taking into account subsequent time slots.

CHAPTER 4

Complexity Analysis of Load Disaggregation

"The art of simplicity is a puzzle of complexity."

– Douglas Horton

In this chapter, two complexity measures are introduced to describe the load disaggregation problem and to make the load disaggregation results comparable in a fair way. A fair comparison between different NILM algorithms is a difficult task due to the fact that recent approaches are highly dependent on different conditions and features such as:

- sampling frequency of the measured power draw,
- number of observed appliances,
- appliance types (e.g., on/off appliances, multi-state appliances),
- the data preprocessing applied to the household power draw (the power draw feed into the NILM algorithm is usually filtered and preprocessed before evaluations) and
- set of used appliance features (e.g., steady state electrical characteristics, transient behavior, etc.).

A comparison between algorithms is possible as for example how many feature are used or on which sampling frequency is the algorithm able to work. But an algorithm comparison lacks of the ability to compare and to evaluate the results of the load disaggregator even if the same dataset and environment settings such as sampling frequency and number/types of appliances were used. Accordingly, commonly applied data pre-preprocessing stages such as noise-filtering, filling of missing data or resampling methods are highly affecting the NILM

problem and the ability to compare different load disaggregation results. A fair comparison is only possible if exactly the same data input is used by the load disaggregator. The data input consists of timer series data and of the feature set used by the load disaggregator. There is the need for a common quantitative metric for NILM which is algorithm independent and considers data assumptions, data pre-processing as well as the used feature set. Thus, the problem itself has to be made comparable which is created by the used appliances in a house, their appliance characteristics and their usage over time. A possibility to make the load disaggregation problem comparable is to describe the complexity of the problem in which the problem can be seen as a simple time series. To describe the complexity of time series different complexity measure were proposed such as entropy-based complexity measures [Pin91, Ric00, RO02], used for different applications such as DNA sequences [Mon14b, Cos05] or EEG signals [Rez98, J04, Gao12]. The problem of load disaggregation is hard due to the high variety of different appliances, their different ways to consume energy and their high time-variant behavior introduced by the appliance user. These facts are uncommon for example EEG based time series. It is necessary to involve appliances and their characteristics as well as the time dependent behavior into the evaluation of a possible complexity measure.
To the best of our knowledge, this is the first approach summarizing the disaggregation problem as a complexity value created by statistical characteristics of the appliance set and the time series behavior. A similar approach concentrating on the fundamental limits of NILM was introduced in [Don14]. In this paper, the authors derived an upper bound on the probability to distinguish scenarios for an arbitrary NILM algorithm to guarantee on when NILM is impossible to be solved. The work in [Don14] differs from our approach as we try to make the problem of superimposed loads comparable between different used NILM algorithms. In addition, Pöchacker in [Pöc15] presents a measure based on the proficiency of power values for the load disaggregation problem which can be interpreted as a complexity measure for load disaggregation. He models the problem as an information theoretical problem in which the power states are interpreted as the accessible channel for the transmission of a set of possible device states. With this assumptions, he computed the entropy, the mutual information and proficiency of synthetically generated and real-world based power values. The work in [Pöc15] is different from to the presented approach since we are considering model and measurement uncertainties and trying to reflect real world effects and challenges to be handled by a load disaggregator. In one of our complexity measures we are also considering the appliance usage behavior. However, the two proposed disaggregation complexity merits are evaluated on real-world data and compared to the disaggregation result of a state-of-the-art NILM algorithm. Parts of this chapter have been published and presented in [Ega15d].

The remainder of this chapter is organized as follows: In Section 4.1 the difficulties for disaggregation of power draws are identified and factors influencing the disaggregation complexity are identified. In Section 4.2, public known load disaggregation datasets are presented with their different parameters as for example monitored houses and electric quantities. Three different real-world datasets are chosen for further evaluation. With the chosen consumption datasets and the knowledge of complexity influencing factors, an appliance set complexity and time series complexity is defined in Section 4.3. In Section 4.4 the evaluation settings are specified and the following Section 4.5 is presenting four case studies to review the complexity measures according to their suitability and meaningfulness to describe load disaggregation problems. Section 4.6 discusses the presented results and Section 4.7 summarizes the chapter.

4.1 The Complexity of Load Disaggregation

The problem of load disaggregation is to break down the household power draw $P(t)$ to its power consumption components $p_n(t)$ and can be described as:

$$P(t) = p_1(t) + p_2(t) + \cdots + p_N(t) \text{ for } t \in \{1, T\} \tag{4.1}$$

where N represents the number of used appliances. Each power profile p_N has its own behavior to consume energy determined by the appliance power states (e.g.: on/off appliance, multi-state appliance) and the appliance usage (e.g.: fridge with periodic usage, TV with common usage times) over time. The task of a load disaggregator is to find the best combination of known appliance power profiles to minimize the error between the estimated power signal and the estimated composition of known appliance profiles. We define the complexity of load disaggregation as the hardness to disaggregate appliance profiles according to their appliance characteristics and the way of overlapping power profiles over time. Accordingly, the problem itself should be defined by a complexity measure to make the problem with its appliance characteristics and device usage comparable. There are several facts influencing the complexity of the load disaggregation problem (Figure 4.1) which are listed in the following:

1. The complexity of aggregated loads increases with an increased number of appliances due to higher probability of ambiguous power draws.

2. The higher the switching frequency (like in the case of periodic performing appliances such as a fridge), the more complex is a device set for a load disaggregation algorithm. This is because the probability of overlapping appliances in operation increases.

3. Appliances with multiple operation states (i.e., washing machine, dishwasher, etc.) make a device set more complex for a load disaggregation algorithm.

4. The higher the similarity between appliance features, the more complex is the problem. Appliance features are for example power state values or consumption shapes.

5. Measurement noise, unknown or not considered appliances and imperfect appliance model description interfere with the household power draw and increase the complexity.

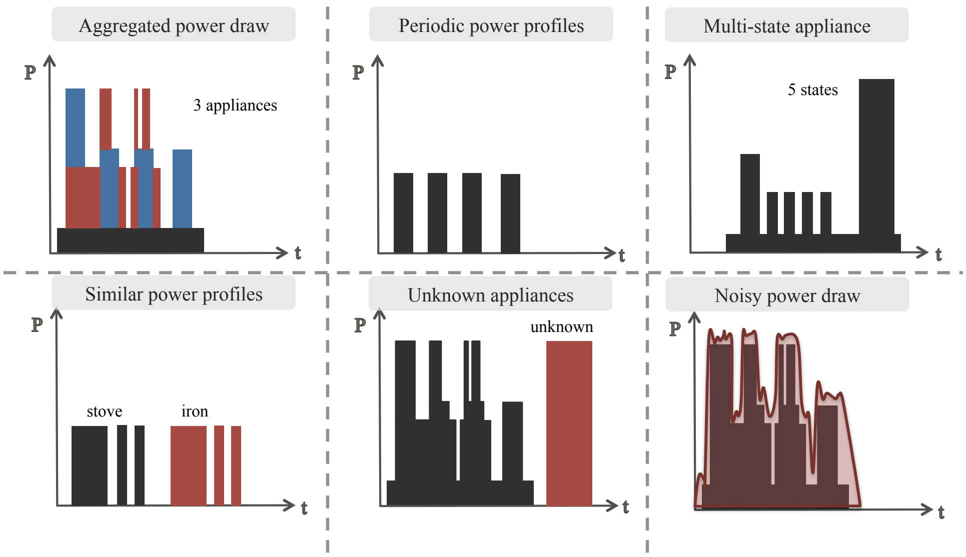

Figure 4.1: *Overview of different scenarios and characteristics of aggregate power draws increasing the complexity of a load disaggregation problem*

The aim is now to define a complexity measure describing the load disaggregation problem by a comparable quantity without taking the used load disaggregator into account.

The complexity measures should fullfill the following requirements:

1. Describe the load disaggregation problem and should not be dependent on the load disaggregation approach.

2. Include appliance descriptions as number of states and the similarities between appliances and states.

3. Reflect model and measurement uncertainties.

4. Reflect the appliance usage and those influences on the problem complexity.
5. Easy and understandable.

In this chapter, we are introducing two complexity measures fullfilling these requirements considering certain problem definitions. These problem definitions are specified by the usable appliance features (steady state features e.g., active power) and the usable appliance type (on/off and multi-state modelled appliances). These assumptions are valid for the whole chapter. Finally, we want also to clarify that the presented approaches are working only for the stated problem definitions. The approaches are not aiming to introduce a general complexity measure working for all problem definitions used by any load disaggregation approach.

4.2 Measurement Datasets

In order to enable research on load disaggregation, public available real-world datasets are necessary. Thus, in recent years several datasets were monitored and finally published to make it available to the public. In Table 4.1 a breakdown of existing datasets is provided. The datasets are categorized according to their location, their duration of the measurement campaign, the number of houses, the monitored features and the used sampling frequency. The monitored features are electric quantities such as active power (P), reactive power (Q), apparent power (S), energy (E), frequency (f), phase angle (ϕ) and current (I). Corresponding to the presented attributes, the sampling frequency and the set of electric features are the main classification attributes between the datasets. A dataset can be chosen and evaluated dependent on the load disaggregation approach and corresponding application.

To test the disaggregation complexity metric on different test cases we performed our complexity study on three different datasets. We have chosen the datasets according to electric features used (active power), to the provided measurement resolution ($1\,Hz$), to the number of monitored houses (at least 3) and to the number and type of measured appliances (common used ones such as fridge and at least 6 different appliance types). The first choice is the open available REDD dataset [Kol11]. We have chosen three different houses from this dataset where 6 appliances were selected according to characteristics to affect the household power demand in a significant way [Car13]. Furthermore, we used the open dataset GREEND [Mon14a], which documents an appliance level measurement campaign in Austria and Italy. As for the REDD dataset we have chosen 3 houses with 6 different appliances as representative for our evaluation. The ECO-Dataset [Bec14] was also used monitoring electricity consumption and occupancy in 9 Swiss houses. 3 houses with 6 different appliances were selected. For our evaluation we have used the whole observation time for the REDD dataset and two week for the GREEND and ECO-dataset. These assumptions are valid through the whole chapter if not mentioned in a different way. In Table 4.2 the appliances for each house and dataset are listed.

4.3 Proposed Approaches

We follow the idea that each possible power value produced by aggregated appliance power states is a combination out of all possible power states of appliances. The task for the load disaggregator is to find the best matching combination of power states with the measured power values. The main idea is to relate an observed power value to all possible power state combinations under the influence of measurement noise and imperfect appliance modelling.

4.3.1 Appliance Set Complexity

One of the major factors influencing the complexity of aggregated power profiles, is the set of possible power values. The more complex appliances are used (e.g., having several operation states with different power consumptions), the more complex is the problem to disaggregate the power profiles. In general, the appliance set is composed by N different appliances. With the knowledge of the appliance set and power demands of each appliance, the first step is to compute the number of possible aggregated power values M. In case of only two state devices 2^N combinations are possible. In general, there are

$$M = 2^{N_2} 3^{N_3} \cdots = \prod_{Z=2}^{Z_{max}} Z^{N_Z} \tag{4.2}$$

different power values possible. N_2 is the number of appliances with two states, N_3 with three states and so on. For the calculation of all possible aggregated power values P_i, repetitions of the same value are possible for instance if a water kettle and a coffee machine consume the same power. Exceptions for this fact are the 0W power state (all devices are off) and the all-on-state P_M (all appliances are on with their highest possible power consumption) which is the highest possible power value. The vector Π is the set of all possible (aggregated) power values P_i for a set of appliances, where i is defined as $i \in [1, M]$.

In its simplest form a NILM device observes a power value and compares it to all possible values P_i given by the device set. As long as there is one single matching power value in the set the task is solved straight forward. The problem

Dataset	Location	Duration	#Houses	Features	Resolution
ACS-F1 [Gis13]	Switzerland	1 hour session (2 sessions)	N/A	I, V, Q, f, Φ	10 s
AMPds [Mak13]	Greater Vancouver	1 year	1	I, V, pf, F, P, Q, S	1 min
BLUED [And12b]	Pittsburg, PA	8 days	1	I, V, switch events	12 Khz
GREEND *[Mon14a]*	Austria, Italy	1 year (3-6 months completed)	8	P	1 Hz
HES	UK	1 month (255 houses) - 1 year (26 houses)	251	P	2 min
iAWE [Bat13]	India	73 days	1	V, I, f, P, S, E, Φ	1 Hz
IHEPCDS	France	4 years	1	I, V, P, Q	1 min
OCTES	Finland, Iceland, Scotland	4-13 months	33	P, Energy price	7 secs
REDD [Kol11]	Boston, MA	3 - 19 days	6	Aggregate: V, P; Sub-metered: P	15 Khz (aggr.), 3 sec (sub)
Sample dataset	Austin, TX	7 days	10	S	1 min
Smart [Bar12]*	Western Massachusetts	3 months	1 Sub-metered +2 (Aggregated + Sub-metered)	P, S (circuits), P (sub-metered)	1 Hz
Tracebase [Rei12]	Germany	N/A	15	P	1-10 sec
UK-DALE [Kel14]	UK	499 days	4	Aggregated P, Sub P, switch-status	16 Khz (aggr.), 6 sec (sub.)

Table 4.1: *Publicly available energy datasets and their characteristics according to location, measurement duration, number of houses, monitored features and measurement resolution*

is harder if i) the searched power value is not in the set of power states at all and if ii) two or multiple power values are matching. Case i) we reason that it should contain something like a multiplicity or occupation number of the possible power values to reflect multiple occurrence. The case ii) does not occur in ideal NILM problems. But in reality it is likely that a measured power value does not match exactly to any of the M aggregated power values of Π. The power values are influenced by noisy measurements and imperfect appliance models. We propose to represent the possible power values by a distribution function instead of a single value. The task is to compare M different distribution functions equivalent to the matching behavior of two or more single power values. This can be achieved by evaluating the similarity of two distributions by overlapping coefficient defined as:

$$\mathrm{OVL}(f_1, f_2) = \int_x \min(f_1(x), f_2(x))\mathrm{d}x. \tag{4.3}$$

It gives the intersection area of the two distribution curves f_1 and f_2 as stated in [Inm89]. For a load disaggregation complexity measure C we propose to estimate the similarity of one power value distribution to all the other possible aggregated power valued distributions. The possible power values are expected to be between 0 and P_M. By use of the overlapping coefficient the disaggregation complexity measure for the power state P_k is defined as:

$$\begin{aligned} C_k &= \sum_{j=1}^{M} \mathrm{OVL}(f_{P_k}, f_{P_j}) \\ &= \sum_{j=1}^{M} \int_0^{P_M} \min(f_{P_k}(p), f_{P_j}(p))\mathrm{d}p \quad . \end{aligned} \tag{4.4}$$

C_k is the disaggregation complexity of the power value P_k within the set of M power state combinations. The parameter k determines the chosen reference power state combination, where $k \in [1, M]$. In case the exact distribution of the power values are not know it is reasonable to assume a normal-distributed Probability Density Function (PDF) $\mathcal{N}(\mu, \sigma)$. The mean value $\mu = P_k$ represents the observed power value and a variance σ expresses the measurement and model uncertainties. To evaluate the complexity of an appliance set, it is now possible to apply the introduced disaggregation complexity for each possible combined power value. This provides an understanding of which power values and appliance state combinations are more complex than others. Figure 4.2 sketches an example how to estimate the disaggregation complexity. For a given set of three on-off devices with {10,20,35}W we estimate the complexity for the power value $P_k = 30$. This represents the case in which device one and two are turned on. The set has $M = 8$ possible power values. Each power state is represented by the same normal distributed PDF. The final disaggregation

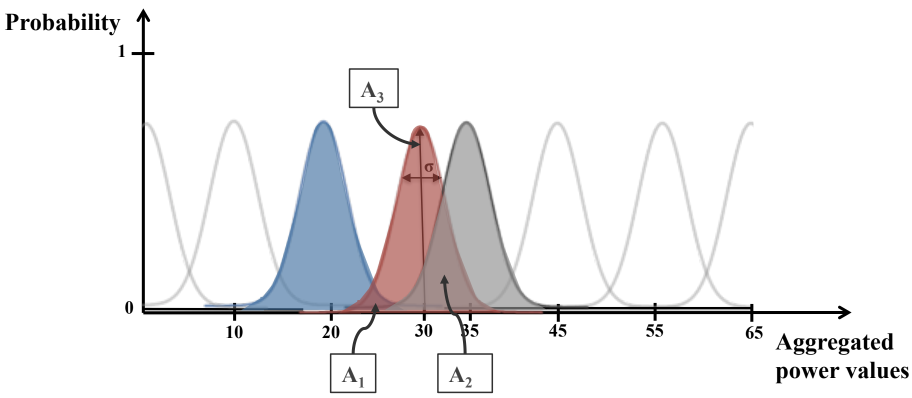

Figure 4.2: *A sketch of the different PDFs for each power value produced by the combination of all available power states of an appliance set. The appliance set consists of three on-off appliances with demands of* 10, 20 *and* $35W$.

complexity value is then the sum of all overlapping areas, such as A_1, A_2 and A_3 shown in Figure 4.2. The introduced disaggregation complexity C can be interpreted as a similarity factor of power states in the appliance set.

Accordingly, a disaggregation complexity C of 1 means that at least one solution or appliance state is equal to the wanted power value. But it can also mean that two power value distributions match with similarity 0.5. The disaggregation complexity $C = 2$ means that in the case two appliances have the same power demand. Exceptions are the all-off power state (0W) and the maximum power demand P_M. Through the limits of the complexity computation of $[0, P_M]$ these states show a value of $C = 0.5$.

Moreover, C depends on the chosen variance σ of the defined uncertainty distribution function. The higher the value of σ, the higher is the probability of intersections between power values. This means the higher σ, the higher is the appliance set complexity. In addition, σ reflects modelling errors and noise influences on the used feature set. The higher σ, the more modelling errors occurred and the higher were the noise effects on the feature set.

In summary, a whole appliance set is characterized by its power states complexity spectrum that shows the complexity value for each of the aggregated power state values. The power states complexity spectrum shows at which regions confusions of states and wrong appliance detections are more likely due to similarity and uncertainty effects.

4.3.2 Time Series Complexity of Aggregated Power Profiles

The introduced disaggregation complexity C considers the appliance set and its characteristics but does not refer to a specific aggregated power profile and accordingly, the usage of different appliances. We introduce the time series disaggregation complexity C_{total} which is a weighted average of the complexities of the power values within a time series. It considers the appliance set implicitly through the disaggregation complexity. The usage of the different appliances is reflected by the power values in the profile. We define the time series complexity of an aggregated power draw as:

$$C_{total} = \frac{1}{T}\sum_{t=1}^{T} C_t = \frac{1}{T}\sum_{t=1}^{T}\sum_{k=1}^{M} \mathrm{OVL}(f_{P_t}, f_{P_k}) \quad , \tag{4.5}$$

where T represents the number of observed power samples. The complexity C_{total} describes the averaged complexity of observed power values within all possible appliance state combinations Π for the whole observation time. Calculation of C_{total} requires knowledge of the respective appliance set, i.e., their number of states, the power values and their distribution (or reasonable assumptions about it). The time-series complexity provides feedback at which point in time a complex power state combination based on the observed power value and the possible power state combinations Π is occurring.

4.4 Evaluation Settings

To be able to evaluate the proposed complexity measures we defined the evaluation settings in the following sections including how to identify appliance power states and the used load disaggregation approach used for evaluation.

4.4.1 Identification of Appliance Power States

To be able to compute the two complexity measures, the set of occurred power states is necessary. If metadata provides this information, the data of power states could be used, but for most cases and datasets this information is either not provided or not in the desired extent. Accordingly, a straightforward approach would be to use expert knowledge to identify the appliance states and their power demand. But this process is time consuming and erroneous. An automatic state detection algorithm is presented, based on the approach published in [Ega15b] and also discussed in Section 5.2.2. The detection approach can be applied on submetered measurement data as well as on aggregated power measurements. For both scenarios different outputs are produced in which the submetered measurements can produce multi-state power states of appliances. Similarities between appliances and their power states are possible. In contrast, the aggregated power measurement data is producing a set of power states without any information of appliances and their number of states. It is detecting different power states and not different appliances. The algorithm tries to find an unambiguous set of power states. However, we want to clarify that the use of this detection approach is not necessary for the calculation of the complexity values. The complexity values can be applied to any detection approach providing a set of appliances power states in which the appliances are described as on/off or multi-state appliances.

The proposed detection approach is applied to each house and dataset chosen as described in Section 4.2. The results are listed in Table 4.2. The parameters of the algorithm are set as in [Ega15b] which are a time window of $30\,s$ for the used median filter and a threshold value of $25\,W$ to detect rising and falling edges. Changing the parameters would cause different results of appliance states. For example using a larger time window would not consider very short appliance usages.

Dataset	Appliance Type	Detected Power (submetered)	Detected power (aggregated)
REDD 1	oven, fridge, dishwasher, microwave, stove, washer dryer	[1680 2478], [200 420], [50 210 410 890 1115], [55 110 270 300 620 1405 1505], [260 710 1440], [2705]	[55], [200], [250], [410], [710], [890], [1078], [1368], [1620], [17425], [2270], [2504], [2670]
REDD 2	kitchen outlet, lighting, stove, microwave, kitchen outlet, fridge	[130 210 770], [123], [410], [40 1718 1850], [1050], [160 420]	[90], [145], [245], [310], [410], [600], [770], [937], [1060], [1752], [1885]
REDD 3	fridge, dishwasher, washer dryer, microwave, bathroom gfi, kitchen outlet	[100 400], [210 525 730], [2265], [120 540 1698], [860 960 1285 1605], [40 365 900 1220 1520]	[70], [120], [205], [270], [370], [535], [730], [960], [1274], [1676], [1835], [2197], [2367], [2630]
ECO 1	fridge, dryer, coffee machine, kettle, washing machine, PC	[40], [250 440 785], [50 1225], [1800], [90 180 250 365 21688], [72]	[105], [245], [335], [545], [900], [1232], [1800], [2170]
ECO 2	diswasher, fridge, entertainment (stereo system and TV), Freezer, water kettle, dimmable lamp	[120 2132], [70], [55 175], [50 310], [50 1840], [80 185]	[110], [190], [280], [510], [1868], [2108]
ECO 3	fridge, kitchen appliances (coffee machine, bread baking machine and toaster), lamp, freezer, entertainment (stereo and TV), microwave	[100], [67 190 280 445 650 785 1065 1545], [130], [100 175 280], [120], [40 1365 1485]	[80], [135], [195], [265], [435], [668], [841], [1007], [1185], [1386], [1565]
GREEND 1	coffee machine, washing machine, fridge, dishwasher, water kettle, vacuum cleaner	[60 148 470 570 1225 1265], [70 155 210 260 423 1898], [55 140 240], [40 1900], [1790], [1220]	[110], [239], [448], [540], [1267], [1896]
GREEND 2	fridge, dishwasher, water kettle, washing machine, dryer, bedside light	[80], [80 1725], [850], [90 173 1910], [1580], [60]	[92], [182], [845], [1583], [1775], [1900]
GREEND 3	TV, washing machine, dryer, dishwasher, kitchenware, coffee machine	[110 235 285 360], [125 245 358 1998 2100], [70 160 2358 2550], [70 2002], [120 1235], [55 125 540 882 1047 1220 1630]	[110], [295], [530], [863], [1043], [1230], [1635], [1920], [2093], [2355], [2554], [2830]

Table 4.2: *List of datasets (REDD, ECO, GREEND) with 6 chosen appliances and their appliance power states detected for submetered power draws and the aggregated power draw.*

4.4.2 Load Disaggregation Algorithms

The proposed complexity measures should describe the complexity of aggregated power loads. To get an idea of how meaningful the proposed complexity approaches are, the results of the complexity measures are compared to the results of an appropriate and suitable load disaggregation approach. This comparison should give a quantitative feedback if the complexity value is meaningful according to the used load disaggregation approach. We claim that the load disaggregation approach needs to have the same inputs as the complexity measures to be able to provide meaningful results.
We used the proposed load disaggregation approach from [Ega15a] and also discussed in Section 5.2.3 based on PF. The appliance model includes on the one hand the approximated power demand and on the other hand the general appliance structure, such as how many states a device has. The PF-based approach is suitable for our evaluation due to the fact that the algorithm can handle a set of appliances modelled as on/off or multi-state appliances and is performing load disaggregation based on a set of power states and the aggregated power draw. For the evaluation the PF is parametrized as in [Ega15a] in which the number of used particles, as most important parameter, is set to 1000 particles.

4.5 Case Study

This section presents four different case studies for the evaluation of the proposed complexity measure such as a case study i) on the appliance set complexity, ii) on the time series complexity, iii) on the influence of varying measurement and model uncertainties and iv) on the relation between a load disaggregation approach and the proposed complexity measures.

4.5.1 Appliance Set Complexity for Different Datasets and Different Sets of Power States

As described in the previous sections, the appliance set complexity is aiming to describe the complexity of the used appliance set without considering the appliance usage over time. The most relevant parameter are the used power values for each appliance power state. These power states are identified using the algorithm presented in Section 4.4 and leads to the results for aggregated and submetered measurements presented in Table 4.2.

As input for the complexity computation a vector of all possible power state combinations of the appliance set of Table 4.2 is used. The results

		Appliance Set Com.				Time Series Com.			
		submetered		aggregated		submetered		aggregated	
Dataset	**H.**	*max*	*mean*	*max*	*mean*	*max*	*mean*	*max*	*mean*
REDD ●	*1*	16.91	7.88	2.28	1.48	13.79	1.04	1.62	0.50
REDD ■	*2*	6.170	2.62	2.32	1.33	5.39	0.54	2.32	0.11
REDD ⧫	*3*	21.39	8.69	1.98	1.32	17.54	1.07	1.98	0.35
ECO ●	*1*	6.65	2.88	2.67	1.36	3.71	0.95	2.62	0.15
ECO ■	*2*	12.06	4.75	1.44	1.04	11.99	2.86	1.11	0.19
ECO ⧫	*3*	16.62	6.53	1.59	1.15	14.77	4.91	1.57	0.41
GREEND ●	*1*	18.20	7.17	2.01	1.19	7.77	0.89	1.06	0.12
GREEND ■	*2*	4.46	2.18	1.36	1.07	4.305	0.91	1.35	0.50
GREEND ⧫	*3*	48.36	24.43	1.87	1.18	45.01	3.67	1.81	0.04

Table 4.3: *List of mean and maximum of the appliance set complexity and the time series complexity for each house and dataset*

are presented in Table 4.3 using the mean and the maximum value of the appliance complexity.The complexity values for submetered data are higher and more complex than the aggregated power readings. As reason we claim that similarities between appliances are getting lost in the case of aggregated loads due to the inability to distinguish between appliances.

With aggregated power readings it is only possible to distinguish between different power states. This also leads to the fact that the problem complexity for the same house of a dataset differs between appliance sets created by aggregated or submetered power readings. This strengthens the need for a complexity measure due to different preprocessing stages of power data. We also provide Figure 4.3 presenting the appliance set complexity for each dataset over all possible power state combinations and is based on the appliance states produced by the submetered power readings. The plot shows for each possible power state combination the appliance set complexity. The color white means that the appliance set complexity is zero because this power value is not producible by a combination of saved power states for a certain dataset and house. The appliance set complexity starts from green (low complexity), blue (medium complexity) and ends at red (high complexity). The colors are normalized according to the dataset with the maximum occurred appliance set complexity. Figure 4.3 shows which dataset and house is more complex according to the used power states presented in Table 4.2. For example, house 2 of the GREEND dataset has a very low appliance set complexity while house 3 of the GREEND dataset has a very high and tight appliance set complexity.

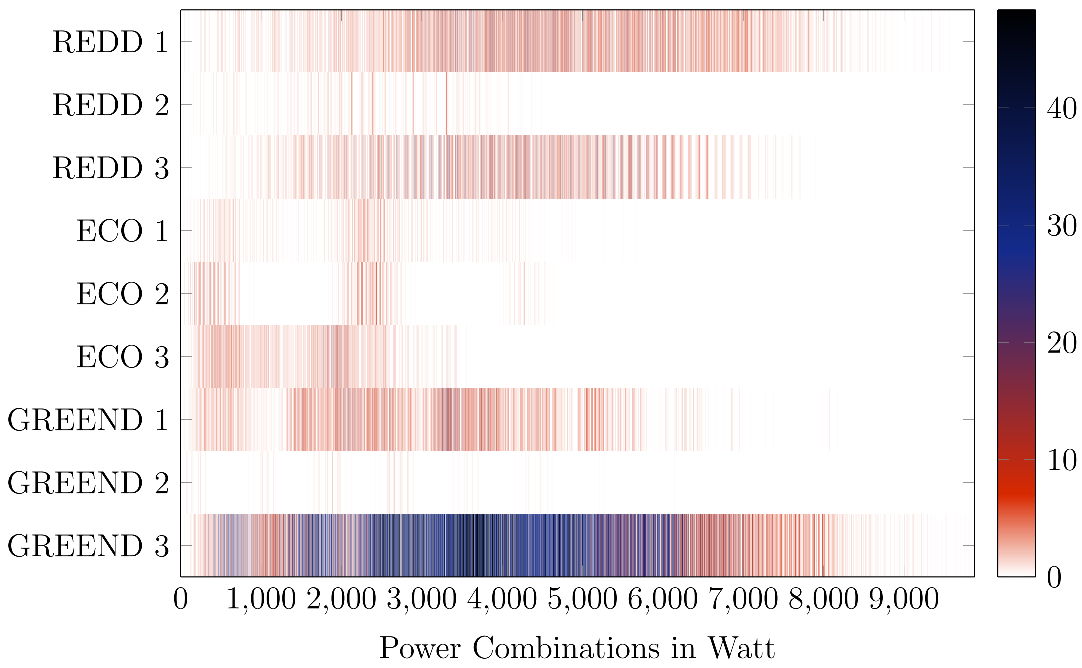

Figure 4.3: *Color map of the appliance set complexity for 3 houses of the REDD, ECO, GREEND dataset over possible power combinations of all houses and datasets*

4.5.2 Time Series Complexity for Different Datasets and Different Sets of Power States

The appliance set complexity gives feedback of the problem complexity of the used appliances by comparing their power states and appliance structures. For the load disaggregation problem another important factor is the influence of the appliance usage over time. This considers how and when appliances are operated. In this sense appliances could be operated for example user driven (e.g., coffee machine, TV) or periodically activated (e.g., fridge). The proposed time series complexity considers this circumstances in its computation. For the evaluation of this complexity measure the time series of all houses and datasets defined in Section 4.2 for an observation window of half day are considered. The input for the complexity computation are the measurement samples which are combinations of possible power states affected by noise in which the appliance set complexity considers power state combination without noise as input. Appliance states are based on aggregated and submetered power data from Table 4.2. In Table 4.3 the mean and the maximum of the time series complexity for all houses and datasets. The time series complexity is highly affected by the appliance usage. In this respect, effects on the appliance usage could be a

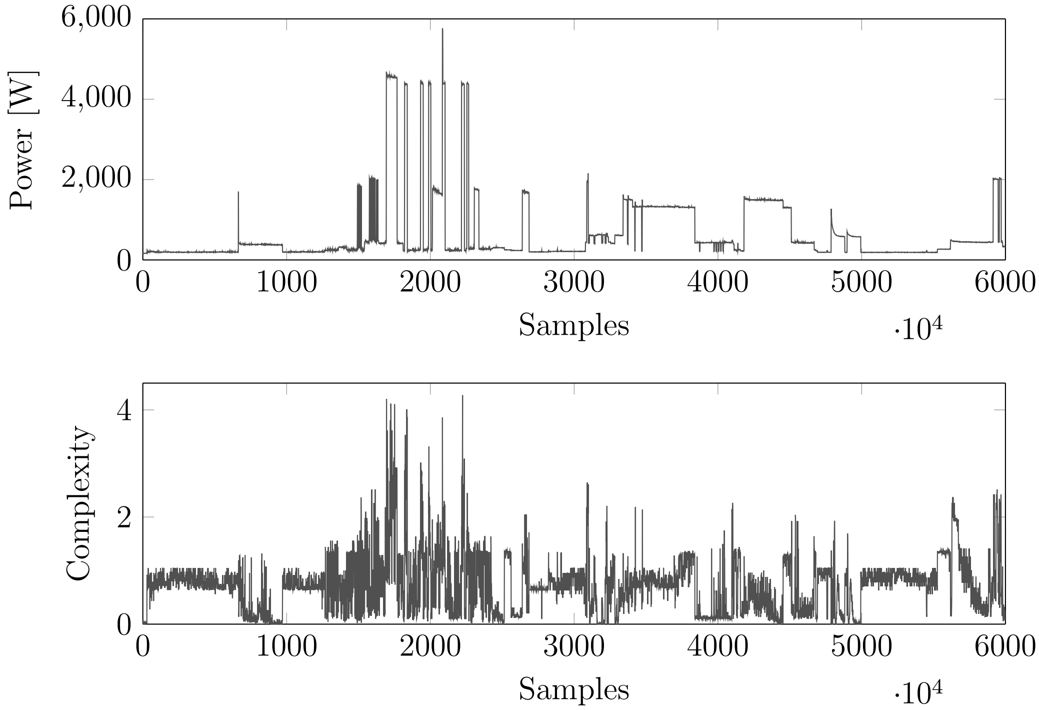

Figure 4.4: *Time snippet of the power readings for REDD house 2 with the time series complexity per sample*

high number of overlapping appliances and the unfavourable combination of appliances in time. Moreover, also the measurement error of the observed power value is highly effecting the time-series complexity. The higher this error, the more probable is an increased problem complexity. We claim that even complex appliance sets as the house 3 of the GREEND dataset can have a low time series complexity due their appliance usage over time. Thus, the appliance set complexity and the time series complexity do not correlate to each other. For example, a high appliance set complexity can lead to a low or a high time series complexity. Moreover, an overlapping behavior of appliances and their power states results in an increased and high complexity value while high power values do not necessarily results in a high complexity. A time snippet of a time series of house 2 of the REDD dataset with corresponding complexity values for each measurement sample are presented in Figure 4.4. Over time the complexity is highly fluctuating according to the measured power value. In addition, we want to clarify that even small power values can be very complex to be disaggregated and also high power values could be of low complexity which strengthens the need of a complexity measure reflecting the possible power state combinations as well as the appliance usage over time.

4.5.3 Variation of Model and Measurement Uncertainties

An important aspect to be included and represented by the complexity measures are uncertainty effects introduced by model and measurement errors. Model errors represent erroneous power state identifications as well as the normal power variation of appliance power states. Measurement uncertainties are introduced by erroneous measurements in the monitoring chain. As described in Section 4.3 the appliance set as well as the time series complexity consider this by using the value σ. Thus, the aim of this case study is to vary the uncertainty of models and measurements by this value σ and to evaluate the effect on the complexity values. The value σ is diversified by $\sigma \in \{1, 5, 10, 50\}W$. The values should represent small measurement errors (e.g., 1 to 10 W) and model definition errors of 50 W. It is assumed that all used appliances and their resulting power state combinations suffer from the same model and measurement uncertainties. We considered the appliance set and the time series complexity for our evaluations. Results are presented in Table 4.4. The higher the uncertainty and the error for the model description and the measurement environment are, the more complex is the problem. It is very probable that power states are similar to each other and cannot be considered as significant disaggregation features. The measurement and model identification stages have to be accurate to achieve sufficient and suitable appliance state dissimilarities. A high power state diversity is needed to lower the load disaggregation complexity and simplifies the load disaggregation problem in general.

4.5.4 Load Disaggregation of Complexity Marked Power Readings

In this case study the results of the complexity measures are compared with the results of a NILM approach on the same power data. The aim is not to evaluate the used disaggregation approach, but to give a feedback about the suitability and meaningfulness of the proposed complexity measures. As described in Section 4.4 we used the load disaggregation algorithm from [Ega15a] which is able to handle on/off and multi-state appliances. We used the appliance set and models identified by the submetered measurements from Table 4.5.

We assume the availability of ground truth data for the evaluation as reason to use the submetered data and not the aggregated power readings. The appliance set detected in Table 4.5 compared to the listed ones in Table 4.2 are different because the appliance state identification algorithm from Section 4.4 was considering only the most common appliance power states. We defined power states as most common if a detected power state occurred as often as

House	$\sigma = 1$	$\sigma = 5$	$\sigma = 10$	$\sigma = 50$
	Appliance Set Complexity			
REDD 1	2.21/7.67	7.88/16.90	15.20/22.94	74.30/111,7
REDD 2	1.34/3.11	2.62/6.17	4.05/8.25	15.35/21.86
REDD 3	2.65/9.87	8.69/21.39	16.31/32.36	74.77/105.15
ECO 1	1.38/4.41	2.88/6.65	4.79/9.97	21.87/42.19
ECO 2	1.97/7.02	4.75/12.06	8.42/17.44	38.02/64.75
ECO 3	2.14/9.01	6.53/16.62	12.63/26.11	61.17/106.84
GREEND 1	2.22/9.91	7.17/18.20	13.60/30.64	65.78/124.06
GREEND 2	1.30/2.43	2.17/4.46	3.19/6.19	10.72/17.92
GREEND 3	5.57/17.84	24.43/48.36	47.98/83.19	237.48/381.62
	Time Series Complexity			
REDD 1	0.11/2	0.41/4.85	0.69/7.89	3.55/24.1
REDD 2	0.01/1	0.06/1.13	0.27/1.74	1.38/3.79
REDD 3	0.08/3.2	0.42/4.63	0.82/6.11	3.51/15.96
ECO 1	0.2/1.96	0.84/2.59	1.58/3.42	6.5/9.75
ECO 2	0.33/2.03	1.13/3.92	1.79/7.12	8.1/31.41
ECO 3	0.19/2.98	1.02/6.37	2.17/9.44	10.29/29.69
GREEND 1	0.23/2.15	1.1/5.15	2.27/7.13	9.64/21.31
GREEND 2	0.17/2.98	0.86/4.63	1.14/6.71	4.95/18.25
GREEND 3	0.05/1.97	0.55/3.74	1.16/5.06	3.98/12.6

Table 4.4: *List of mean and maximum of the time series complexity for each house and dataset for a varying set of σ*

15% of the maximum occurred power state. We used power readings of a whole day to calculate the time-series complexity. The load disaggregation algorithm is evaluated according to the real and estimated energy per kWh on appliance level and to the aggregated power readings. The results for each house and dataset for all used appliances are shown in Table 4.6. Less complex time series like in REDD house 2 are easier to disaggregate than more complex time series as in ECO house 2. A lower complexity is in general easier to disaggregate as a more complex time series. Similar power states as for example in house 1 and 2 of the ECO dataset are highly affecting the load disaggregation result. In the case of similar power states the algorithm is not able to distinguish between appliances with similar power states supporting the need of a common complexity measure for load disaggregation. By using a different power state identification setting also the appliance set complexity and the time-series

Dataset	Appliance States
REDD 1	[1690 2455], [190] [210 410 880 1110], [60 1533], [260 710 1440] [2712]
REDD 2	[770], [145], [410], [1875], [1050], [160]
REDD 3	[120], [210] [2255], [130 1740], [960 1290 1610], [360 900]
ECO 1	[40], [780], [50 1205], [1795], [80], [90]
ECO 2	[120 2060 2170], [70], [55 178], [50], [1845], [160]
ECO 3	[100], [55 1085 1520], [130], [100], [120], [1330 1567]
GREEND 1	[50 1270], [55 1840], [50 140], [40 1900], [1790], [1220]
GREEND 2	[80], [80 1730], [850], [90 160 1910], [1580], [60]
GREEND 3	[60], [72 2020], [160 2415], [70], [1230], [1030]

Table 4.5: *Appliance set used by the load disaggregation approach.*

complexity compared to the previous case studies become different. This also strengthens our assumption to have a complexity measure handling the set of appliance power states independent from the used load disaggregation algorithm.

4.6 Discussion

In the previous section different case studies were presented to evaluated usefulness of the proposed complexity measures. For example the appliance set complexity is highly dependent on the used appliance set. The number of devices states and similar states between appliances are strongly affecting the load disaggregation complexity. The complexity is higher for more complex appliance sets and is not dependent on the used house or dataset. Thus, we claim that the preprocessing stage has an important effect on the problem complexity and accordingly also on the result of the used load disaggregation process. For example changing the parameter of the appliance state detection phase would cause different appliance states and accordingly, the complexity of the problem. This fact is also valid for the time-series complexity. By using an appliance dataset with complex appliances and structures also the time-series complexity is affected strongly for the same house of a dataset. Moreover, the time series complexity is highly affected by the appliance usage. We claim that even complex appliance sets as the house 3 of the GREEND dataset can have a low time series complexity due their appliance usage over time. Thus, the appliance set complexity and the time series complexity does not have necessarily the same meaning. A high appliance set complexity can lead to a low or a high time series complexity.

Dataset	App. 1 *real/est.*	App. 2 *real/est.*	App. 3 *real/est.*	App. 4 *real/est.*	App. 5 *real/est.*	App. 6 *real/est.*	**Total** *real/est.*	**AC** *mean/max*	**TC** *mean/max*
REDD 1	0.13/0.22	1.27/0.98	0.31/0.43	0.53/0.21	0.003/0.32	0.0/0.06	2.23/2.21	2.97/9.06	0.41/4.64
REDD 2	0.19/0.13	0.82/0.99	0.05/0.28	0.29/0.05	0.24/0.20	1.67/1.44	3.26/3.01	2.01/4.69	0.23/1.27
REDD 3	1.08/0.94	0.16/0.25	0.70/0.78	0.20/0.29	0.69/0.87	0.33/0.34	3.17/3.46	1.69/3.78	0.40/4.09
ECO 1	0.54/0.35	0.001/0.04	0.23/0.26	0.0002/0.02	0.002/0.34	0.49/0.26	1.27/1.27	1.469/2.69	0.84/2.59
ECO 2	0.0/0.05	0.53/0.61	0.86/0.067	0.71/0.54	0.30/0.31	0.01/0.82	2.39/2.40	2.72/5.83	0.758/3.038
ECO 3	0.66/1.18	0.48/0.32	0.073/1.55	4.18/1.26	0.54/1.46	0.42/0.48	6.30/6.25	2.34/6.45	0.54/2.66
GREEND 1	0.11/0.29	0.0/0.10	1.20/0.32	0.01/0.41	0.0/0.03	0.0/0.081	1.32/1.24	2.57/6.04	1.08/5.15
GREEND 2	0.55/0.43	0.81/0.04	0.0/0.03	0.0/0.04	0.19/0.82	0.0/0.196	1.56/1.55	1.07/1.27	1.002/3.023
GREEND 3	2.59/0.49	0.93/0.94	1.94/1.60	0.65/0.58	0.08/1.50	0.19/1.40	6.37/6.48	1.73/4.01	0.42/2.15

Table 4.6: *List of the load disaggregation result (real and estimated) on appliance level and in total for all houses and datasets. For comparison also the appliance set complexity (AC) and time-series complexity (TC) are shown.*

In addition, the presented case studies showed that the complexity measure and the load disaggregation problem is highly affected by the chosen variance σ representing model and measurement uncertainties. Imperfect appliance models and faulty power measurements lead to an increased problem complexity which lead to a higher degree of similarities and characteristic features disappear.

We also showed that the proposed complexity measures can classify the complexity of a load disaggregation problem but does not correlate to the used load disaggregation approach. The result of the load disaggregation approach cannot be estimated by the presented complexity measures. The complexity measures aims to describe and to make the problem defined by appliance modelling and data preprocessing comparable. The introduced measures should solve the inability to compare different load disaggregation approaches using different data sets and data setting. By using the complexity measure different problems can be assessed and accordingly, also the hardness to disaggregate power profiles.

4.7 Summary

This chapter defined two complexity measures for the problem of load disaggregation which deals with the task to break down the aggregated power draw of appliance to the appliance components. Appliance characteristics with smart algorithms are used to solve this task. One important aspect is the distinction between the disaggregation approach itself and the problem of aggregated power profiles. Beside clear performance measures for NILM algorithms it needs a clear definition to specify the hardness or complexity of a specific load disaggregation problem. This makes a fair comparison of different NILM approaches according to the used load disaggregation problem possible. To overcome the lack of measures/metrics to compare load disaggregation problems we introduced two novel complexity measures to assess the complexity of a load disaggregation problem based on the used appliance sets. With the proposed complexity measures the used appliance sets and the aggregated power readings are evaluated for their complexity. The two complexity measures include information such as the appliance power states for on/off and multi-state appliances and uncertainties created by appliance modeling errors and erroneous power measurements. The complexity measures are evaluated on real-world datasets and quantitatively compared with the results of a state-of-the-art NILM approach. Our evaluations show that our disaggregation complexity measure is able to assess the hardness of an appliance dataset as well as of specific time series for a NILM algorithm. We want to emphasize that the presented complexities are relative and not absolute measures for the problem complexity. Thus, knowing the disaggregation complexity is not sufficient to determine the performance of the load disaggregator. The presented measure gives meaningful results for load disaggregation problems with one feature such as the active power representing each power state of an appliance. The introduced complexity measures are thus a novel way to make NILM problems comparable.

CHAPTER 5 Unsupervised Load Disaggregation Approach

"You can have data without information, but you cannot have information without data"

– Daniel Keys Moran

Supervised load disaggregation algorithms require previous knowledge about the devices employed in the system as labelled training data. This is in most cases a crucial problem for the deployment as well as for costs of an load disaggregation system. To overcome the need of labelled appliance data and of learning phases, this chapter introduces a novel unsupervised load disaggregation approach. The unsupervised load disaggregation approach aims to minimize the amount of *a priori* information without a deduction of the information gain produced by load disaggregation. Our proposed unsupervised load disaggregation approach combines the following approach characteristics:

- The number of appliances and their model description is learned without any *a priori* knowledge. The needed information will be learned in operation and will be improved over time.
- The approach is of low computational complexity running on embedded hardware with restricted resources.
- The classification process works online on each measurement sample to provide a fast feedback usable e.g. to detect and to react on faulty appliances.

Accordingly, the introduced load disaggregation approach identifies device operations based on the characteristic power changes when devices are switched on/off or switched to a different power state. Considering that power states of devices are distinguishable, the proposed algorithm does not need *a priori* information of the system. It autonomously adapts to new and updates devices. The algorithm can be used online and is suitable for operation on low-cost embedded system hardware, for example as part of an energy management system.

The presented approach constitutes an important step towards an automatic disaggregation of electrical loads. The approach is especially suitable for household appliances, since these environments feature typically different power draws out of a device pool that is also subject to change over a larger timescale by acquisition of new devices. By presenting a working approach for automatizing the detection of devices without supervision, i.e., without the need for querying the user every time the device pool has changed, this work lays the ground for a broad application of load disaggregation.

The remainder of this chapter is organized as follows: in Section 5.1 background of the used appliance/ household modelling and used state estimation is provided. Section 5.2 concentrates on the proposed approach and their processing steps such as i) feature detection, ii) state clustering and appliance modelling, iii) state estimation and appliance classification and iv) appliance database update. In Section 5.3 the evaluation settings are presented and Section 5.4 deals with case studies to evaluate the proposed approach. The discussed case studies are on i) synthetics data evaluations, ii) real-world data evaluations, iii) transition matrix dependency, iv) appliance model detection, v) whole NILM framework and vi) the computational complexity. Finally, Section 5.5 discussed the presented results and Section 5.6 summarizes the chapter. Parts of this chapter are based on the published works in [Ega13a], [Ega15a] and [Ega15b].

5.1 Estimation and Modelling Approach

The proposed NILM approach is based on probabilistic graphical models (e.g., HMM) and on Bayesian state estimation (e.g., PF). We provide in the following sections an overview of the used techniques and knowledge used in this chapter.

5.1.1 Household and Appliance model

The total power load of a household is the aggregated sum of appliance power profiles, where each appliance is modelled by a Hidden Markov Model (HMM) [Rab89] and the total power consumption is modelled by a Factorial Hidden Markov Model (FHMM). An HMM is a probabilistic graphical model describing time series as a Markov model in which the states are not directly observable.

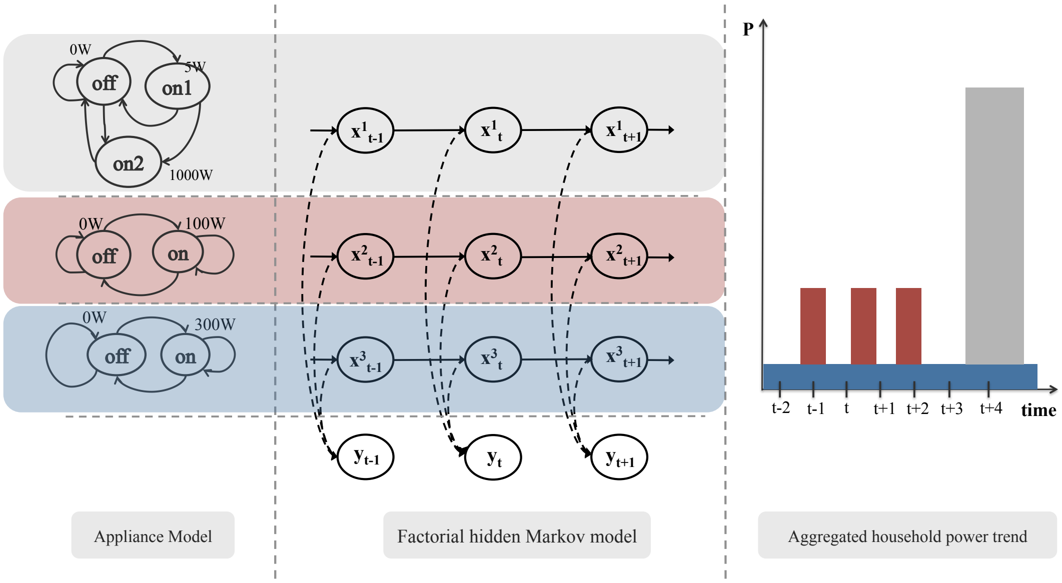

Figure 5.1: *Sketch of the appliance models for on/off and multi-state appliances, of the FHMM model and of the power draw of the aggregated power draw of three appliances*

The states of an HMM are characterized by a probability distribution function. States cannot be directly observed, but can be estimated from the available measurements. The HMM model has n hidden states $s = \{s_1, \dots s_n\}$ as well as a transition matrix $A = \{a_{i,j} \leq i, j \leq n\}$ representing the state transition from s_i to s_j. In detail, $a_{ij} = P(x_{t+1} = s_j \mid x_t = s_i))$, where $a_{ij} > 0$ and $\sum_{j=0}^{n} a_{ij} = 1$. The terms x_t are the states observable at each time slice t, which represents the power consumption of an appliance in a particular state. An emission matrix B must be defined for the HMM representing symbols in their actual states. The emission matrix of the appliance model provides all possible power values in each appliance state. Moreover, the initial probability $\pi = P(x_1 = s_i)$ of the HMM has to be defined.

A vector $z = \{z_1, z_2 \dots z_t\}$ is the result of the hidden states $x = \{x_1, x_2, \dots x_t\}$, where the next state of the HMM is dependent on the HMM's current state and is independent of past states. This is the Markov property $P(x_{t+1} \mid x_t, x_{t-1} \dots x_1) = P(x_{t+1} \mid x_t)$. In Figure 5.1, an example for a general model of an on/off appliance model is shown. In addition, on/off devices and their description can be easily extended to multi-state appliances providing several power states. In the case of an multi-state appliance, the parameter matrices $\{\pi, A, B\}$ of the HMM grow by the number of states n. In general, the definition of the matrices A and B is the crucial task of the appliance model design. The matrices A and B have to be learned online or offline with or without knowledge about the HMM and problem environment. A distinction between supervised and unsupervised learning methods or even semi-supervised learning methods is needed.

The household power profile can be observed as the aggregate power profile $Y = \{y_1, y_2, \dots y_t\}$ of N different appliances. It is generated by the state sequence of $x = \{x^{(1)}, x^{(2)}, \dots x^{N}\}$ representing the superposition of the appliance states $x^{(n)} = \{x_1^{(n)}, x_2^{(n)} \dots x_t^{(n)}\}$ at each time slice. This results in a household model based on an FHMM. An FHMM is commonly used method to model multiple independent hidden states and to decrease the number of parameters in contrast to using a standard HMM with a large set of operational states. The general structure of an FHMM is presented in Figure 5.1.

5.1.2 State Estimation

In the following sections, we discuss background information on particle filtering. We start with Bayesian estimation, explain the shortcomings of using Bayesian estimation with non-linear problems and non-Gaussian noise.

Sequential Bayesian Estimation

According to the Bayesian approach, the state of a physical system x_t at time t can be inferred from the probability density function (PDF) of a state given all the measurement $y_{1:t}$ until time t. The sequential Bayesian estimation has two primary steps at every time instance t:

- State prediction predicting the state as the expectation of the prediction PDF

$$p(\mathbf{x}_t \mid \mathbf{y}_{t-1}) = \int p(\mathbf{x}_{t-1} \mid \mathbf{y}_{t-1})\, p(\mathbf{x}_t \mid \mathbf{x}_{t-1})\, d\mathbf{x}_{t-1}, \tag{5.1}$$

where $p(\mathbf{x}_{t-1} \mid \mathbf{y}_{t-1})$ is the posterior PDF available from time t-1 and $p(\mathbf{x}_t \mid \mathbf{x}_{t-1})$ is the state transition probability given by the system process model.

- Measurement update where upon receiving the measurement, the predicted state is computed as expectation of the posterior PDF

$$p(\mathbf{x}_t \mid \mathbf{y}_t) = \frac{p(\mathbf{x}_t \mid \mathbf{y}_{t-1})\, p(\mathbf{y}_t \mid \mathbf{x}_t)}{\int p(\mathbf{x}_t \mid \mathbf{y}_{t-1})\, p(\mathbf{y}_t \mid \mathbf{x}_t)\, d\mathbf{x}_t}, \tag{5.2}$$

where the $p(\mathbf{y}_t \mid \mathbf{x}_t)$ is the likelihood PDF given by the measurement model of the system. The Kalman Filter (KF) [Aru02] can be used to solve the integrals in Eq. 5.1 and Eq. 5.2 if the system is linear with additive white Gaussian noise. In contrast, if the physical systems are non-linear, then these integrals are intractable. Often, non-linear state estimation methods such as PF are used to approximate these integrals.

Particle Filter (PF)

A PF calculates weighted particles or Monte Carlo samples to approximate the PDFs as in Eq. 5.1 and Eq. 5.2. Particles are propagated over time to obtain new particles and the weights, resulting in a series of PDF approximations. The approximation of the PDF becomes more accurate with an increasing number of samples. In many cases, the sampling of the required PDF is not possible. In such cases, the samples drawn from a different PDF (importance PDF) are used to approximate the required PDF. It is called importance sampling. Let $\{\mathbf{x}^i_{0:t}, \mathbf{w}^i_t\}^{Np}_{i=1}$ be the set of random samples, $x^i_{0:1}$, drawn form the importance density $q\left(\mathbf{x}_{0:t} \mid \mathbf{y}_{1:t}\right)$ and their associated weights, w^i_t, for $1 \ldots N_p$ where N_p is the number of particles. Then the required PDF can be approximated as:

$$p\left(\mathbf{x}_{0:t} \mid \mathbf{y}_{1:t}\right) \approx \sum_{i=1}^{Np} \mathbf{w}^i_t \delta\left(\mathbf{x}_{0:t} - \mathbf{x}^i_{0:t}\right), \tag{5.3}$$

where δ is the unit dirac function and the weights are defined as:

$$\mathbf{w}^i_t = \frac{p\left(\mathbf{x}^i_{0:t} \mid \mathbf{y}_{1:t}\right)}{q\left(\mathbf{x}^i_{0:t} \mid \mathbf{y}_{1:t}\right)}. \tag{5.4}$$

In the case of sequential importance resampling (SIS) [Aru02], the samples and corresponding weights $\{\mathbf{x}^i_{0:t-1}, \mathbf{w}^i_{t-1}\}^{Np}_{i=1}$ which approximate $p\left(\mathbf{x}_{0:t-1} \mid \mathbf{y}_{1:t-1}\right)$ are known at time t. If the importance density for approximating $p\left(\mathbf{x}_{0:t} \mid \mathbf{y}_{1:t}\right)$ is chosen in such a way that

$$q\left(\mathbf{x}_{0:t} \mid \mathbf{y}_{1:t}\right) = q\left(\mathbf{x}_t \mid \mathbf{x}_{0:t-1}, \mathbf{y}_t\right) q\left(\mathbf{x}_{0:t-1} \mid \mathbf{y}_{1:t-1}\right), \tag{5.5}$$

then the new samples $x^i_{0:t} \approx q(x_{0:t}|y_{1:t})$ can be obtained by augmenting the existing samples $x^i_{0:t-1} \approx q(x_{0:t-1}|y_{1:t-1})$ with the new state $x^i_t \approx q(x_t|x_{0:t-1}, y_{1:t})$. The corresponding weight update equation is given as:

$$\mathbf{w}^i_t = \mathbf{w}^i_{t-1} \frac{p\left(\mathbf{y}_t \mid \mathbf{x}^i_t\right) p\left(\mathbf{x}^i_t \mid \mathbf{x}^i_{t-1}\right)}{q\left(\mathbf{x}^i_t \mid \mathbf{x}^i_{0:t-1}, \mathbf{y}_t\right)}. \tag{5.6}$$

Now, the required PDF at time t can be approximated as:

$$p(x_{0:t}|y_{1:t}) \approx \sum_{i=1}^{Np} w^i_t \delta(x_t - x^i_{0:t}). \tag{5.7}$$

However, the SIS algorithm suffers from the degeneracy problem in which all but a few particles have negligible weights. Due to the degeneracy, large computational effort is expended for updating the particles with less contribution to the approximation of the required PDF. One solution to overcome degeneracy is resampling. The resampling process eliminates particles with negligible weights by replacing them with particles with large weights $\{\mathbf{x}_{0:t}^{*i}, \mathbf{w}_t^{*i}\}_{i=1}^{Np}$. Several resampling techniques are proposed in [Aru02]. Then, the PDF can be approximated as:

$$p\left(\mathbf{x}_{0:t} \mid \mathbf{y}_{1:t}\right) \approx \sum_{i}^{Np} w_t^{*i} \delta\left(\mathbf{x}_t - \mathbf{x}_t^{*i}\right). \tag{5.8}$$

The PF algorithm is given as: At time t, $\{\mathbf{x}_{t-1}^{*i}, \mathbf{w}_{t-1}^{*i}\}_{i=1}^{Np}$ are known. The new samples are generated by:

$$\mathbf{x}_t^i \sim p\left(\mathbf{x}_t \mid \mathbf{x}_{t-1}^{*i}\right)\Big|_{i=1}^{Np}.$$

The weights are updated by:

$$\mathbf{w}_t^i = p\left(\mathbf{y}_t \mid \mathbf{x}_t^i\right)\Big|_{i=1}^{Np}.$$

Resampling: The particles are resampled by using the auxiliary resampling [Aru02] as:

$$\left\{\mathbf{x}_t^{*i}, \mathbf{w}_t^{*i}\right\}\Big|_{i=1}^{Np} = \mathit{Resampling}\left\{\mathbf{x}_t^i, \mathbf{w}_t^i\right\}\Big|_{i=1}^{Np}.$$

The state estimate is given by the sample mean of the resampled particles $\mathbf{x}_t^{i*}$.

In summary, the PF estimates the posterior density of the state space based on the observation variables and the dynamic representation of a system.

5.2 Approach

This chapter proposes a novel load disaggregation approach working unsupervised with minimal amount of information. The appliances models are based on HMMs aiming to model appliance stationary processes with continuous valued data over discrete time. Moreover, the classification process using the HMMs should infere the most probable appliance states online by PF considering minimal computational complexity. The proposed principle is presented in Figure 5.2 and can be divided into the four steps i) feature detection, ii) state clustering and appliance modelling, iii) state estimation and appliance classification and iv) appliance database update:

- **Feature Detection**: Aims to detect significant power edges which can be assigned to appliance switching events. Data preprocessing as signal smoothing and denoising takes place at this processing stage.
- **State Clustering and Appliance Creation**: Power edges are formed to state clusters to identify the most important states or switching events. These states are used to create appliance models used by the load disaggregator.

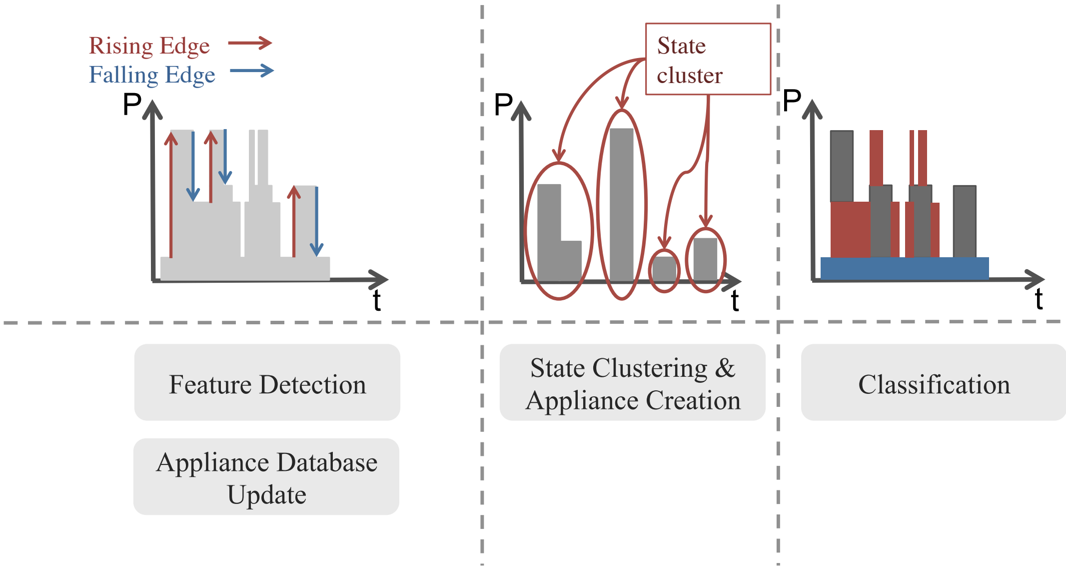

Figure 5.2: *General computation sequence of the unsupervised load disaggregation approach including the stages state detection, appliance database update,state clustering and state estimation and appliance classification*

- **State Estimation and Appliance Classification**: With the appliance models generated, appliance states should be estimated by an online load disaggregation approach using low frequency active power readings.
- **Appliance Database Update**: To add, to maintain and to update appliance models in an autonomous way, this stage is responsible to find new power states, to improve and to update the power states of existing appliance models.

In the following sections each processing stage is described in detail.

5.2.1 Feature Detection

One major task of the proposed load disaggregation approach is to detect and to identify useful appliance features. According to our assumptions, we focus on smart metering readings of active power ratings with a measurements resolution of 1 Hz due to low costs of a sensing platform and lower computation and storage costs compared to high frequency measurements. With the aggregated power readings we aim to extract appliance features based on appliance switching events. In detail, we concentrate on switching on and switching off events where all power states of an appliance are taken under consideration. The task is to produce abrupt edges with a significant change without losing important appliance related information. Power transients can last several seconds in reality which has to be considered by the proposed feature detector. Due to the fact that measurement readings are affected by noise, the readings have to be preprocessed to get sufficient and satisfying data. Thus, we de-noise the power readings by median filtering with an appropriate window size of 30 samples. The window size was set to 30 samples considering that a noted operation duration last longer than 30 seconds. The window size has to be chosen carefully since a window chosen too wide could lead to information loss by wiping out important edges. The filtering process is followed by a process to sharpen edges and to produce steady states in the signal. This is necessary to overcome both fluctuations in the readings and slow power transient from one appliance state to another one. To produce a steady state power signal, we detect the most significant edges by checking for rising and falling edges by:

$$\epsilon(t) = \begin{cases} 1 & \text{if } (P(t) - P(t+w) > (P(t) + th) \\ -1 & \text{if } (P(t) - P(t+w) < (P(t) - th) \\ 0 & \text{elsewhere.} \end{cases}$$

The variable $P(t)$ represents the power readings for each time instance, w represents the used window to overcome slowly rising transients, th is the used power threshold to detect edges in the signal and t is representing the discrete time variable. The vector $\epsilon(t)$ is used as an index vector to decide which power value should be attached to which detected power edge. With the information of $\epsilon(t)$, a new power vector $s(t)$ is generated. The initialized zero vector $s(t)$ is filled with the mean value of power samples between occurring edges. For example between consecutive rising edges all samples are taken, the mean value is calculated and the entries of s between this consecutive edges are set to the calculated power value. The resulting power vector $s(t)$ contains all steady power state with sharp edges and is used by the second edge detector. The edge detector finds edges greater or smaller than the predefined threshold th by creating the difference $p_{diff} = s(t) - s(t-d)$ and by finding positions where $p_{diff} > th$ to get a rising edge or $p_{diff} < -th$ to get a falling edge. The variable d represents the delay to calculate edges and is usually set to $1-3$. The delay is necessary to overcome long lasting power transition at second level. Finally, the pool of rising and falling edges are compared to each other to find matching edges. A falling and rising edge pair is found, if the difference between them is lower than the threshold $2 \cdot th$. In Figure 5.3 a sketch of the proposed procedure is presented. In summary, the feature detection stage is creating a pool of found power edges from a de-noised, filtered and smoothed power draw. For multi-state appliances it would be necessary to consider also the sequence of occurred edges and a logic to map them to a new list of edge pairs (multi-state appliances and its appliance models are not considered in the presented approach).

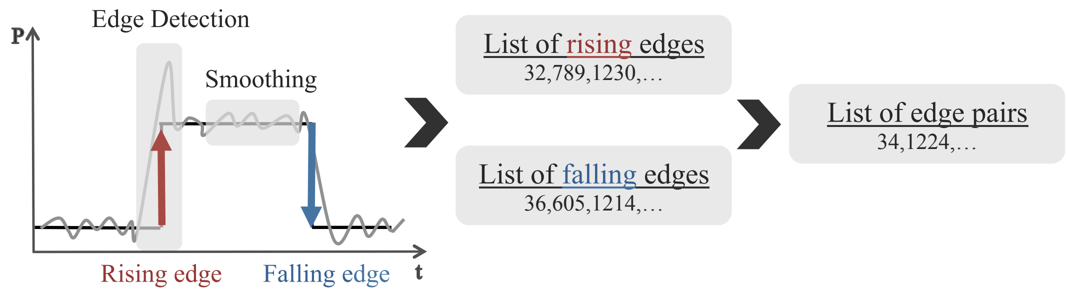

Figure 5.3: *Sketch of the sequence of the edge detection procedure used by unsupervised load disaggregation approach*

5.2.2 State Clustering and Appliance Modelling

The pool of matching power edges is the basis for the next analysing process creating appliance models based on HMMs. First, a histogram of all edge pairs detected by the feature detector is created. The histogram is defined by an upper and lower bound considering the maximum possible power demand of an appliance. Thus, the created histogram counts the occurred power edges from 0 (lower bound) to 3000 W (upper bound) each 5 W. The partition of 5 W for the histogram is a sufficient assumption due to the fact that appliance demands can vary of several Watts. Dependent on the application and the appliances present in a system, the values for the upper/lower bounds and the partition factor can be freely selected.

Next, power edges in the histogram which are occurring at least once are combined with existing neighbouring power edges. With the set of neighbouring power edges, the considered power value is created by calculating the mean value of the set of adjacent power states. The identified power states are used to create on/off appliance models by assigning them to the observation matrix of the appliance HMM. The off state (0 W) is assigned to each appliance HMM as first observation entry followed by the detected power demands in operation. An appliance is set to the off state as initial operation state. In summary, the clustering approach is detecting unique power edges and models them as single on/off appliances.

5.2.3 State Estimation and Appliance Classification

According the previously defined problem definition, the aim of the estimator is to use the detected appliance models (HMM) and to classify aggregated power readings online and of low computational complexity. Our approach is based

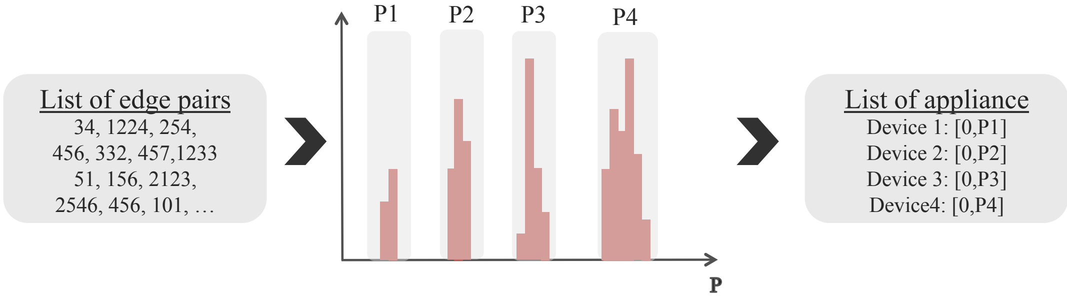

Figure 5.4: *Sketch of the power state histogram of detected edges pairs and the created appliances*

on Particle Filter (PF) with auxiliary resampling aiming to approximate the posterior density of the FHMM. The approach disaggregates each appliance power demand and appliance state from the household demand, according to the current observed consumption and the given appliance models modelled as HMMs. The PF estimates the posterior density of the FHMM state space. The output of the PF are power values for each appliance which are aggregated at each point in time. The PF has the characteristic to randomly adjust the estimated power observation for each appliance in predefined ranges. This range is defined as $2 \cdot \sqrt{p_s}$ where p_s is the saved power demand for an appliance state. The reason for that is to estimate and to compensate appliance inaccuracy in the appliance power consumption as well as imperfect model definitions. Moreover, the posterior density is resetting all states every 60 seconds. As reason we claim that the PF is in general for continuously changing signals in which the aggregated power demand in second resolution is comparatively sluggish. Significant power changes are commonly occurring not each second leading to an loss of diversity by the resampling stage.

The PF itself is not providing the information in which state an appliance is operating. It delivers power values which are given to a decision making process to classify the appliance. The decision making process has knowledge of the power demand of each appliance operation state. It decides accordingly in which state each appliance is at each point in time by a simple thresholding approach. The use of a PF as load disaggregator is beneficial for three reasons. First, PF can handle non-linear problems presented by non-linear behaving loads such as a driller or a dimmer. Second, it can handle non-Gaussian noise influences resulting from uncertainty in power trends and consumption data. Third, PF and its performance can be adjusted by the number of used particles. The more particles the PF considers, the better is the estimated posterior density and the estimation result. The number of particles is limited by the computational effort of the approximation process. Moreover, exact knowledge of the transition matrix is not necessary since the PF is independently estimating the appliance states by an appropriate number of used particles. In case of a two-state appliance represented by a two-state transition matrix, a clear trend should be visible which state is more probable than the other. This simplifies the appliance learning and modelling stage in which intensive offline processing stages are not needed. The disaggregation process is performed on each measurement sample (each second) and considers only the current power sample for the estimation process. It is performing online and is only restricted by the number of considered particles. A simplified sequence of how to the PF is used for the application to disaggregate loads is shown in Figure 5.5.

5.2.4 Appliance Database Update

As we are aiming to find appliance models which are valid for the whole uptime, we have to consider different cases of appliances, of appliance models and characteristics as wells as different ways of using them. For example, appliances can be used only once a week, can be used every 15 minutes or can be exchanged with new appliances. Moreover, the demand of an appliance can be varying over time. The proposed approach, described in the following, should be able to deal with these situations. The method should be able to deal with no system information at the start and should improve its knowledge over time. As a first step, we consider and update each saved and identified appliance model each observation time. We try to detect each observation time all possible power states and compare them with already detected power states. The comparison is done by performing a distance measure by saved power states to new detected power states. We use the absolute error between the states and evaluated according to a threshold value defined. If the power states is not known, a new appliance is added to the appliance database. If the power states is detected, the old power state is replaced by the mean power state of the old power state and the new power state. However, for the first observation time all detected power states are used to introduce new appliance models saved.

5.3 Evaluation Settings

In the following, the evaluation settings for the simulations on synthetic data and on a real-world dataset are described and the evaluation metrics for the proposed approach are defined.

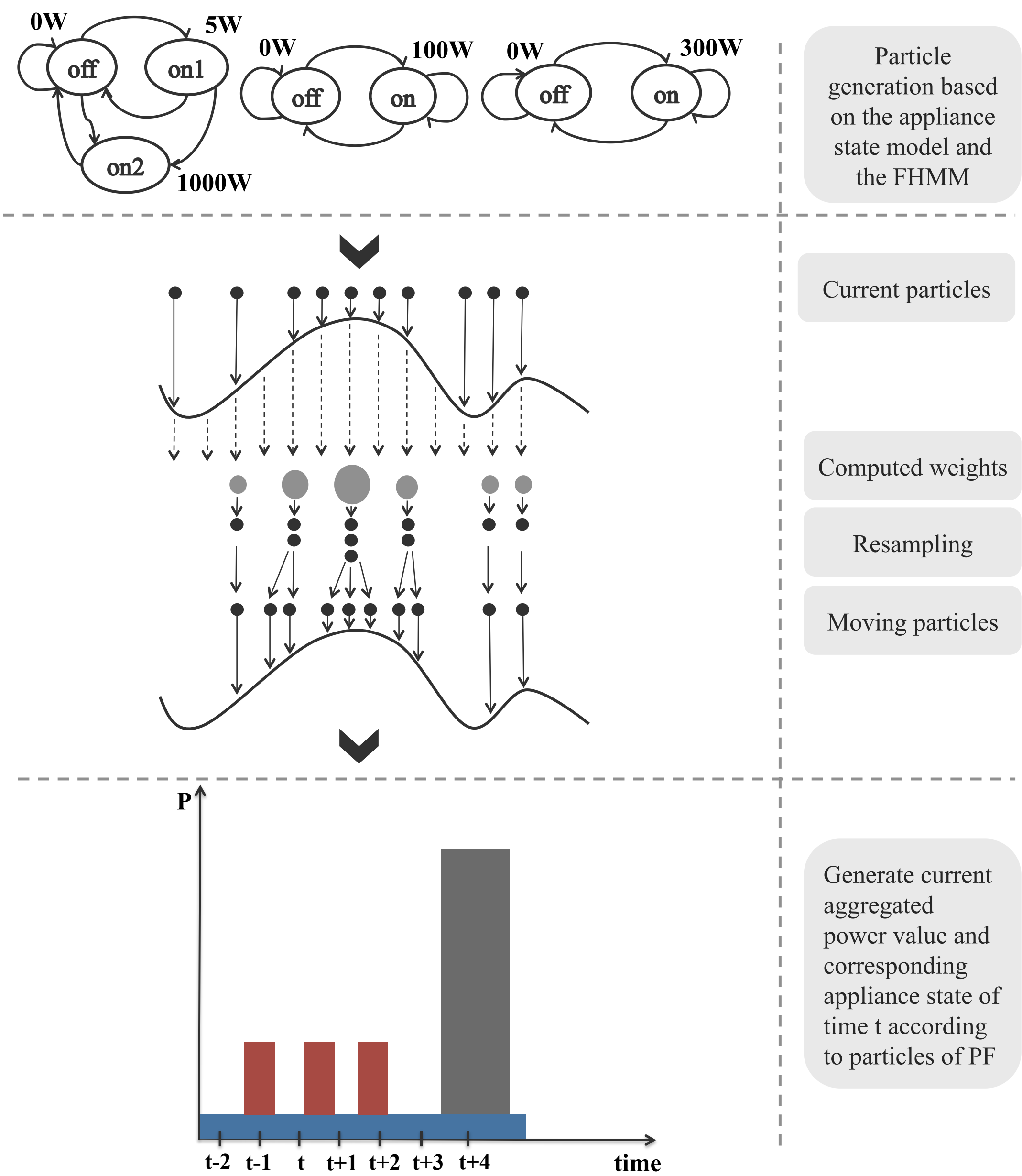

Figure 5.5: *Sketch of the general principle how the PF is used to infer appliance states.*

5.3.1 Settings on Synthetic Data

The first step to evaluate the proposed approach is to define controllable test cases based on synthetic data. A list of appliances is shown in Table 5.1. They are categorized according to their power demand p_s per operation state s, their average run time t_{on} and their average occurrence per day f_{on}. The values for p_s, t_{on} and f_{on} are empirically identified in which the chosen values are not necessarily reflecting real appliances and their characteristics. For the evaluation each appliance is modelled as an HMM defined by their transition matrix A and their observation matrix B. The transition matrix A is created by an on and off probability in which the on probability is defined as $p_{on} = f_{on}/T$ and the off probability as $p_{off} = 1/t_{on}$. The discrete variable T represents the observed time which is defined in our case as $T = 86400$. It represents an observation time of one day in second resolution. The observation matrix is built up by $B = \{0, p_d\}$, where $B = 0$ belongs to the appliance off state and $B = p_d$ belongs to the appliance on state. Multi-state appliances are defined in a similar way. The transition matrix A is defined in a way that the on probability is chosen equivalently for on/off appliances. The transition states from one state to the other state are defined by t_{on} and is the same for each transition from one state to another state. The probability of staying in the same state is calculated by 1 minus the sum of all other transition probabilities. The observation power demand matrix B_m is defined by the power demand values for each appliance state.

To create the total power demand P, a set of appliances (number of appliance is defined in advance) from Table 5.1 are chosen. The power demand of the chosen appliances is created by their HMM definitions and finally, added up to create the total power demand P.

5.3.2 Real-World Dataset

Beside the evaluations based on synthetic data, we also evaluated our results on real-world measurements. A variety of different real-world dataset exists as presented in Table 4.1. We decided to use the REDD dataset as real-world dataset. It provides several power draws of monitored appliances and houses over several days [Kol11] and is well-known in the research community. We used house $1, 2, 3$, where each appliance is defined by the recorded apparent power. We choose 6 different appliances which are common in households and are affecting the energy consumption of an household in a significant way [Car13]. The REDD dataset offers submetered power profiles in which the devices are known and the loads are already disaggregated. We calculated an overall power profile based on the submetered data which was fed into the presented classification approach from Section 5.2.3. The submetered power profiles have a varying sampling frequency and are partially out of order which makes it necessary to adjust the sampling frequency on an equal level using interpolation. The used sampling frequency is one second. The used classification approach is a model based state estimation approach using HMMs. Thus, for each appliance the transition matrix A and the observations matrix B has to be defined. In Section 5.4.2 we used the MATLAB pre-programmed HMM functions [1] to construct the matrix A. The used appliance states are set by expert knowledge. The observation matrix B is defined as well by expert knowledge and is composed by the off state consuming $0W$ and all other detected power states by the human. In all over Sections 5.4.3, 5.4.5 and 5.4.6 we used a predefined transition matrix A representing a tendency of which power state is more probably than the others. The used observation matrix is either detected by expert knowledge as before or by the proposed appliance modelling and detection approach presented in Section 5.4.4 and 5.2.2.

5.3.3 Evaluation Metric

The planned case studies of this chapter require different evaluation metrics. We distinguish in following between the evaluation metrics used for i) the appliance classification process and ii) the appliance detection process.

Appliance Classification

To evaluate the performance and the precision of the proposed load disaggregation approach, we used the accuracy matrix for binary classification, the error of the allocated energy *ERR* and the root mean squared error *RMSE* of the estimated to the real signal. To be able to formulate the accuracy of the classification process, the classification terms *TP* (number of times an appliance is correctly detected as on), *FP* (number of times an appliance is wrongly detected as on), *FN* (number of times an appliance is wrongly detected as off) and *TN* (number of times an appliance is correctly detected as off) have to be defined. The classification terms *TP*, *FP*, *FN* and *TN* are straightforward for On/Off appliances. Considering multi-state appliances we remark that we consider only the operating state if an appliance is on or off and not, if a device is in a certain operating state. With the mentioned classification terms, the overall classification result is calculated by combining *TP*, *FP*, *FN* and *TN* to the accuracy metric

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \in [0, 1], \tag{5.9}$$

where ACC represents how accurate appliance states can be detected by the proposed approach.

[1] hmmestimate - estimates the HMM based on emissions and states

The error of the allocated energy is defined as:

$$ERR(i) = \sum_{t=1}^{T} \hat{y}_t^{(i)} - y_t^{(i)}, \tag{5.10}$$

where $\hat{y}_t^{(i)}$ is the estimated power and $y_t^{(i)}$ is the ground truth data for each appliance at each point in discrete time T. The variable T considers the used observation window of the classification process. The metric $ERR(i)$ represents the energy not correctly assigned for each appliance. Therefore,

$$ERR = \sum_{i=1}^{N} ERR(i) \tag{5.11}$$

represents the energy not correctly assigned in total over all used appliances N. Finally, the $RMSE$ is defined as:

$$RMSE = \frac{\sqrt{E((\hat{\Theta} - \Theta))^2}}{max(\Theta)}, \tag{5.12}$$

where $\Theta = \sum_{i=1}^{N} \hat{y}^{(i)}$ represents the true total power load, $\hat{\Theta} = \sum_{i=1}^{N} y^{(i)}$ represents the estimated total power load produced by the classification approach and $max(\Theta)$ represents the maximum power value in the total power load.

Appliance Detection

To evaluate the performance of the appliance detection, we propose a modified accuracy according to the following definition:

$$ACC_e = \frac{As}{N_s + Unk}. \tag{5.13}$$

N_s represents the number of uniquely known power states, As describes the true positive events as correct assignable power states detected and Unk stands for false positive events where power states are detected but cannot be assigned to a known power state. A power state is assigned as As if the absolute error between the real and the detected power state is less than a predefined threshold value.

5.4 Case Studies

This section presents six different case studies for the evaluation of the proposed load disaggregation approach such as an case study i) on synthetic data evaluation, ii) on real-world data evaluation, iii) on transition matrix dependency, iv) on the appliance model detection, v) on the whole NILM framework and finally iv) on the computational complexity.

5.4.1 Synthetic Data Evaluation

This case study aims to provide an insight on the applicability of the proposed classification approach to solve the load disaggregation problem based on synthetic generated data. Thus, we define synthetically 18 typical household appliances as described in Table 5.1 and in Section 5.3.1. We generated the total power demand by a randomly chosen set of 12 appliances from Table 5.1. The set is changed every day and the whole observation time is 30 days. As evaluation metric we used the accuracy ACC defined in Section 5.3.3 feedbacking all right detected on and off states over all possible measurements and the $RMSE$ defined in Section 5.3.3. The ACC is computed on appliance level and in total. An appliance is defined as on if their power demand exceeds $10\,W$ and off otherwise. Moreover, we varied the number of used particles from $N = \{100, 1000\}$ to identify a sufficient number of particles for further evaluations.

The results for the ACC and the $RMSE$ are presented in Table 5.1. The more particles are used for the classification process, the better is the overall estimation results. Each appliance could be detected with an accuracy over 92% for 1000 particles. The improved result is impressed more by the $RMSE$ results and shows that the proposed classification process is applicable for the load disaggregation problem. However, due to the good results on synthetic data we claim to use real data for further evaluations. Real world measurements are influenced by varying and not modelled power states as well as unpredictable noise which is needed for a fair evaluation of the proposed approach.

Name	P_{state1}	P_{state2}	P_{state3}	P_{state4}	*avg. run time*	*avg. occurrence*	**Accuracy** $Np = 100$	$Np = 1000$
Water Kettle	1980	0	-	-	120	10	0.9969	0.9995
Stove	870	0	-	-	1200	5	0.9591	0.9670
Freezer	170	0	-	-	120	100	0.9272	0.9570
Iron	1430	0	-	-	1800	2	0.9794	0.9823
Fridge	78	0	-	-	300	150	0.8960	0.9348
Toaster	700	0	-	-	250	2	0.9904	0.9974
Vacuum Cleaner	1100	0	-	-	800	2	0.9791	0.9828
Air Condition	1000	0	-	-	200	200	0.9669	0.9780
Hair Dryer	1530	0	-	-	600	2	0.9867	0.9913
Boiler	1300	0	-	-	1200	4	0.9578	0.9599
Waffle Iron	950	0	-	-	600	2	0.9762	0.9770
Curling Iron	90	0	-	-	100	3	0.9849	0.9860
Mixer	80	0	-	-	180	2	0.9871	0.9848
Coffee Machine	10	1150	0	-	120	5	0.9439	0.9096
Clothes Dryer	250	1800	0	-	3600	1	0.9584	0.9481
Clothes Washer	170	650	0	-	3600	1	0.9490	0.9544
Microwave	5	1650	0	-	300	4	0.9421	0.9285
Dishwasher	5	200	1200	0	3600	2	0.9229	0.9274
						ACC	*0.9619*	*0.9745*
						RMSE	*0.0912*	*0.0494*

Table 5.1: *A selection of typical on/off and multi-state appliances described by the power demand for each state, average usage time and average occurrence per used observation window. It shows the accuracy on appliance level, and in total and the reached RMSE.*

5.4.2 Real-World Data Evaluation

Based on the promising results of the previous section, the proposed classification process is tested on real world data. As mentioned in Section 5.3.2 we used the REDD dataset from MIT for our evaluations. 3 houses with 6 different appliances (see Table 5.2) are used to generate the aggregated total power demand wherein beside on/off appliances also multi-state appliances are used by the load disaggregator. In this table, for each appliance the used power demands per operation state are presented. This information was used to train appliance HMMs. In detail, the transition matrix of the HMM were trained by a provided MATLAB function and the observation matrix was filled by the power demands listed in Table 5.2. For this case study an observation time of one week was used.

Beside the aim to provide a feasibility study of the proposed load disaggregation approach, also the PF as estimator is modified and adopted to the problem. As mentioned in Section 5.2.3 we reset the used appliance state space each predefined time to overcome the loss of diversity due to slowly changing signals. We varied this time[2] by $t_{reset} = [1\,min, \frac{1}{2}\,h]$. Moreover, we varied the used standby threshold by $p_{stby} = [0\,W, 5\,W, 10\,W]$.

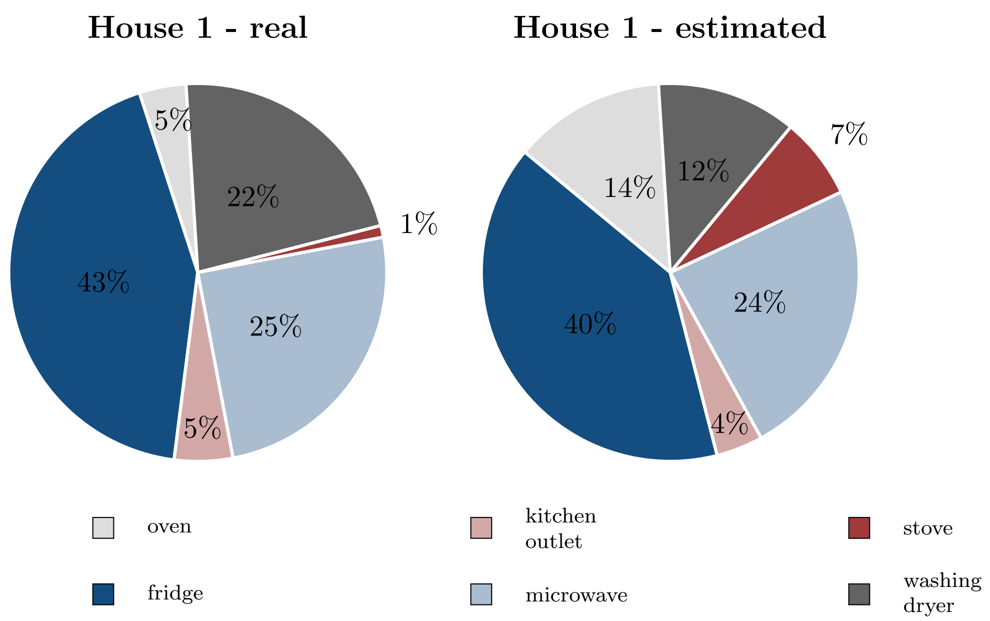

Figure 5.6: *Classification results for the energy shares of REDD houses 1 by using $t_{reset} = 1\,min$ and a power variation of $0\,W$*

[2] s stands for seconds, h represents an hour

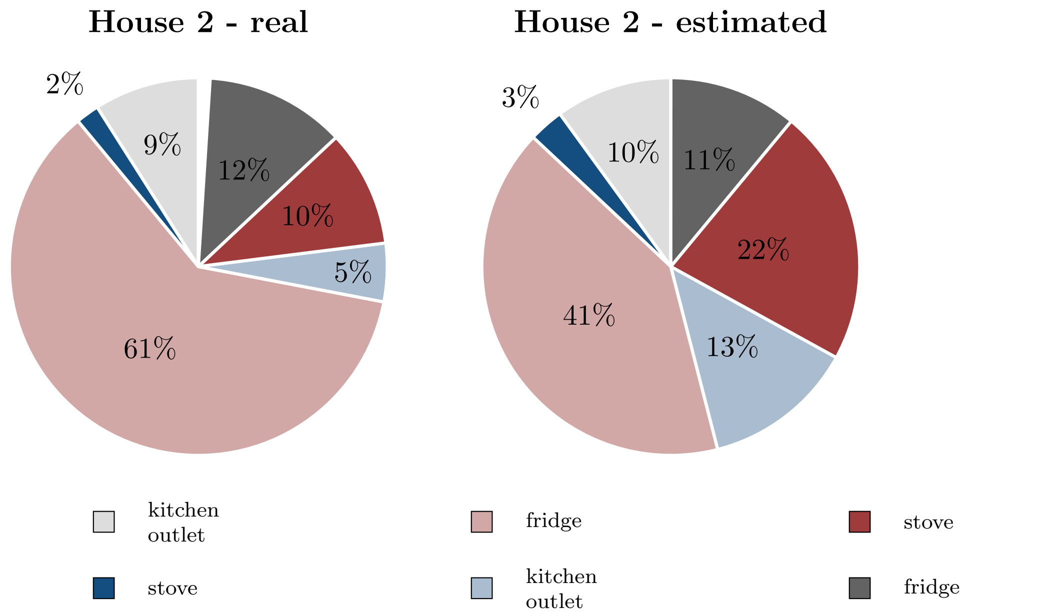

Figure 5.7: *Classification results for the energy shares of REDD houses 2 using* $t_{reset} = 1\,min$ *and a power variation of* $0\,W$

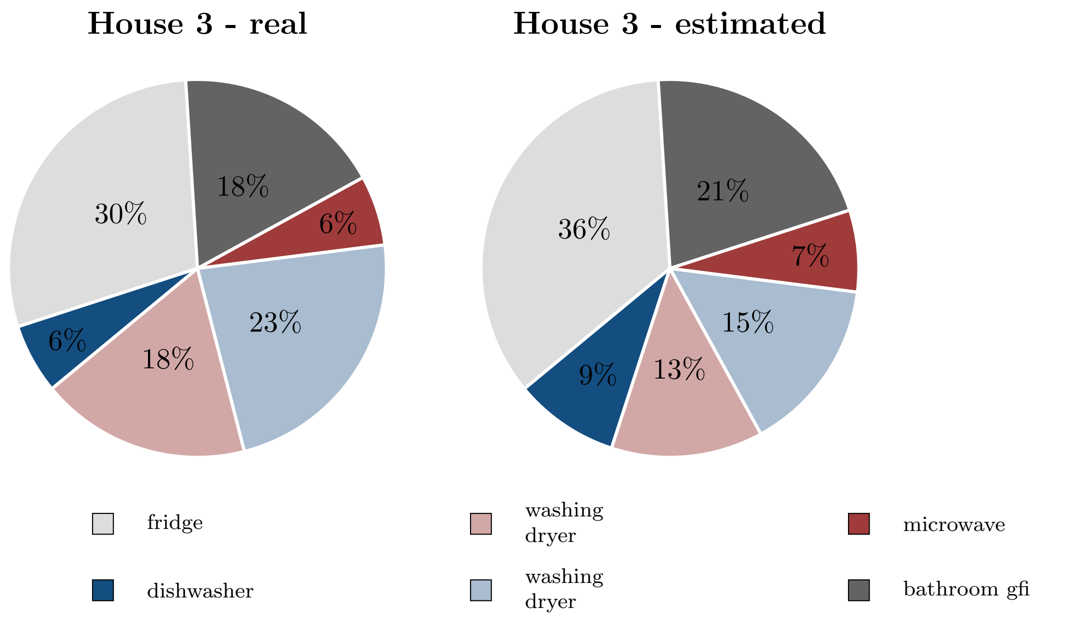

Figure 5.8: *Classification results for the energy shares of REDD houses 3 using* $t_{reset} = 1\,min$ *and a power variation of* $5\,W$

The results are evaluated according the reached ACC on appliance level and in total as well as to the reached $RMSE$ and are presented in Table 5.2.

The results for the ACC and for the $RMSE$ are the worst using no resetting behavior. Thus, we claim the need to have this simple rule to reset the used state space every predefined time. This time has to be chosen carefully according to results. If the time for resetting is chosen to big as in the case of $1/2\,h$ the results of the classification process are getting worse. We identified a time of $1\,min$ as sufficient and suitable time to reset the appliance state space of the PF.

Moreover, we also varied the standby power in this case study. The standby power is the power to decide if an appliance is off or on. The table shows slightly better results for house 1 and 2 using a standby power of $5\,W$. For these two houses the reset variation evaluation were computed with a standby power of $0\,W$. For house 3 we used already a standby power of $5\,W$ because the performance of the classification process was decreased with a standby power of $0\,W$. We claim to use a low standby power of $5\,W$ or lower to be appropriate for different households and their uncertainty situations due to noise and measurement/modeling errors.

Finally, in Figure 5.6, 5.7 and 5.8 the energy shares for the estimated and the ground truth data of house 1-3 are presented. The error ERR between the estimated power and the real power is for house 1 3.7%, for house 2 3.9% and for house 3 25.2%. The results show that some energy shares are incorrect assigned due to similarities of the power states. House 3 has a low energy tracking result, but a sufficient classification results. As reason for the mismatches of the power shares we assume the erroneous appliance modelling by the expert. The classification algorithm uses constant power states for the classification for each power state. Also standby behaviors of appliances were not modelled for this evaluation, which can have a considerable proportion of a household power demand. A solution to overcome this situation and to improve the energy tracking behavior is to model power state in predefined ranges (e.g state 1 is valid between $105\,W$ and $120\,W$) or by a normal distribution function with a desired mean and variance. Compared to other approaches as for example [Pat12] reaching up to 90% of right assigned energy or [Kol11] reaching up to 65% of right assigned energy, the presented approach reaches comparable better results depending on the problem case.

5.4.3 Transition Matrix Dependency

In Section 5.2.3 we claimed that the classification approach based on PF has not the need to know the exact transition matrix. A trend such as the probability to stay in a state is higher than to change a state should be reflected by the transition matrix and its entries. This fact is reducing the learning phases of appliance models and is beneficial for the computational complexity and also

House	**Device Type**	**States** [W]	**reset variation ACC**			**power variation ACC**	
			no	1 min	1/2h	0W	5W
1	oven	1660	98.4	99.2	99.8	-	99.1
	fridge	8, 190, 2000	95.4	97.7	98.5	-	97.8
	kitchen outlet	1080	96.3	99.7	99.3	-	99.7
	microwave	5, 1550	96.2	98.4	98.0	-	98.4
	stove	1430	97.9	99.4	98.7	-	99.2
	washing dryer	2700	99.1	99.3	99.2	-	99.3
		ACC	*95.8*	*98.9*	*98.5*	-	*98.9*
		RMSE	*0.115*	*0.033*	*0.051*	-	*0.030*
2	kitchen outlet	1060	87.3	98.8	98.4	-	99.2
	stove	410	73.6	99.0	96.1	-	97.9
	fridge	8, 160	74.8	89.5	92.1	-	86.7
	kitchen outlet	15, 780	93.5	98.5	96.9	-	99.6
	dishwasher	250, 1215	66.9	88.6	91.6	-	87.8
	microwave	5, 45, 1900	80.3	79.4	86.8	-	81.9
		ACC	*64.4*	*92.1*	*93.7*	-	*92.2*
		RMSE	*0.24*	*0.027*	*0.079*	-	*0.0270*
3	fridge	120	62.0	96.6	95.8	70.2	-
	dishwasher	215, 752	44.4	98.3	93.6	70.1	-
	washing dryer	2230	56.6	99.3	99.2	99.1	-
	washing dryer	280, 2500	56.1	98.2	98.2	97.9	-
	microwave	1760	57.7	99.6	99.6	99.4	-
	bathroom gfi	1280, 1590	51.7	99.4	97.2	99.3	-
		ACC	*45.9*	*98.3*	*96.2*	*90.8*	-
		RMSE	*0.597*	*0.062*	*0.067*	*0.069*	-

Table 5.2: *Case study results of the accuracy for the PF based classification on the REDD dataset for house 1–3 with known appliance models*

for the online capability. This case study should support our statement by testing general appliance transition matrices. Thus, we set the transition matrix constant to

$$A = \begin{bmatrix} 0.9 & \frac{0.1}{N} & \cdots & \frac{0.1}{N} \\ \frac{0.1}{N} & 0.9 & \cdots & \frac{0.1}{N} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{0.1}{N} & \frac{0.1}{N} & \cdots & 0.9 \end{bmatrix},$$

for evaluation case 1, uniformly distributed ($\mathcal{U}$) to

$$A = \begin{bmatrix} \mathcal{U}_1 & \frac{1-\mathcal{U}_1}{n-1} & \cdots & \frac{1-\mathcal{U}_1}{n-1} \\ \frac{1-\mathcal{U}_2}{n-1} & \mathcal{U}_2 & \cdots & \frac{1-\mathcal{U}_2}{n-1} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1-\mathcal{U}_n}{n-1} & \frac{1-\mathcal{U}_n}{n-1} & \cdots & \mathcal{U}_n \end{bmatrix},$$

for evaluation case 2 and equal distributed by setting all matrix entries to 0.5 for the evaluation case 3. The size of the matrix depends in the number of appliance states n. An on/off devices has a 2x2 transition matrix, a 3-state device a 3x3 transition matrix and so on. We used real world data (see Section 5.4.2) and an observation window of one day. We repeated the simulations of evaluation case 2 ten times to produce different transition matrices. The results for each test case compared to the normal case representing the appliance transition matrix learned by MATLAB are presented in Table 5.3. The MATLAB learned transition matrix are learned for each appliance and have in general a high state probability to stay in the same state (greater than 0.9) and accordingly a quite small state-change probability. Best results are achieved by using the constant transition matrix. Worst results are obtained by the equal distributed matrix entries and the learned transition matrix in some houses. As reason we assume that the learning process is highly affected by the uncertainties affected by noise or modelling errors. However, the results are strengthening our statement that our proposed classification approach has only to follow a trend for the transition matrix. Learning phases can be simplified and the computational complexity can be reduced.

5.4.4 Appliance Model Detection

Up to now the appliance models were known or learned before evaluation. The task of this case study is to test the proposed appliance detection and modelling approach presented in 5.2. First, we used synthetic power data with a random composition of on/off appliances from Table 5.1 for 10 days with 100 simulation runs. This allows to verify the result with ground truth data which is not available using real world data. Second, we used real world data to evaluate the performance on the first 3 houses of the REDD dataset on an observation window of 12 consecutive days.

In the case of real world data in which we have no information about the used power states, the first task is to find common power states in the real world measurements. We used the submetered power draws for the chosen appliances. For each power draw we identified the most common appliance power states consuming more than $50\,W$ and were running at least $100\,s$ for the last 12 days.

House	Device Type	*case 1*	*case 2*	*case 3*	*normal*
			ACC		
1	oven	99.9	99.9	99.9	99.2
	fridge	99.0	98.9	99.0	97.7
	kitchen Outlet	99.9	99.9	99.9	99.7
	microwave	99.3	99.3	99.4	98.4
	stove	99.7	99.6	99.6	99.4
	washing dryer	99.9	99.9	99.9	99.3
	ACC	*99.62*	*99.52*	*99.56*	*99.0*
	RMSE	*0.018*	*0.018*	*0.018*	*0.033*
2	kitchen outlet	99.5	99.4	99.0	98.8
	stove	98.8	98.6	99.6	99.0
	fridge	95.9	92.0	60.2	89.5
	kitchen outlet	99.6	99.2	98.6	98.5
	dishwasher	96.3	92.3	58.4	88.6
	microwave	93.3	87.3	85.9	79.4
	ACC	*97.3*	*94.8*	*83.6*	*92.3*
	RMSE	*0.009*	*0.014*	*0.036*	*0.027*
3	fridge	98.6	78.5	64.1	96.6
	dishwasher	97.6	93.3	96.1	98.4
	washing dryer	99.8	99.8	99.2	99.3
	washing dryer	97.9	97.4	97.3	98.2
	microwave	98.8	98.7	99.5	99.6
	bathroom gfi	98.9	98.7	96.2	99.4
	ACC	*98.6*	*94.4*	*92.01*	*98.6*
	RMSE	*0.015*	*0.019*	*0.047*	*0.062*

Table 5.3: *Case study results of the ACC for the PF based classification on the REDD dataset for house 1-3 with varying transition matrix definition. Case 1 represents the constant transition matrix, case 2 represents the uniformly generated transition matrix, case 3 represents an equal distributed transition matrix and the normal case represent the learned transition matrix by MATLAB*

This results in a set of power values for each appliance in which power states could be very similar to each other. According to the fact to have a unique set of occurred power states, we combined similar power states. Power states which are in a range of $50\,W$ are combined to a single power state by calculating the mean value of the set of similar power states. We used the defined metrics in Section 5.3.3. In the case using synthetic data we calculated for each metric value As, Unk and ACC_e the mean value out of all simulations runs.

The result on synthetic data are presented in Table 5.4 representing As, Unk, ACC_e for a varying number of used appliances ($N \in [6, 9, 12]$). Figure 5.9 shows As and Unk for all simulation runs and used appliance numbers ($N \in \{6, 9, 12\}$)

House	$N = 6$	$N = 9$	$N = 12$
As	5.89	8.42	10.36
Unk	0.68	1.47	1.88
ACC_e	0.98	0.93	0.86

Table 5.4: *Case study results for As, Unk and ACC_e for synthetic generated power draws with a varying number of appliances ($N \in \{6, 9, 12\}$)*

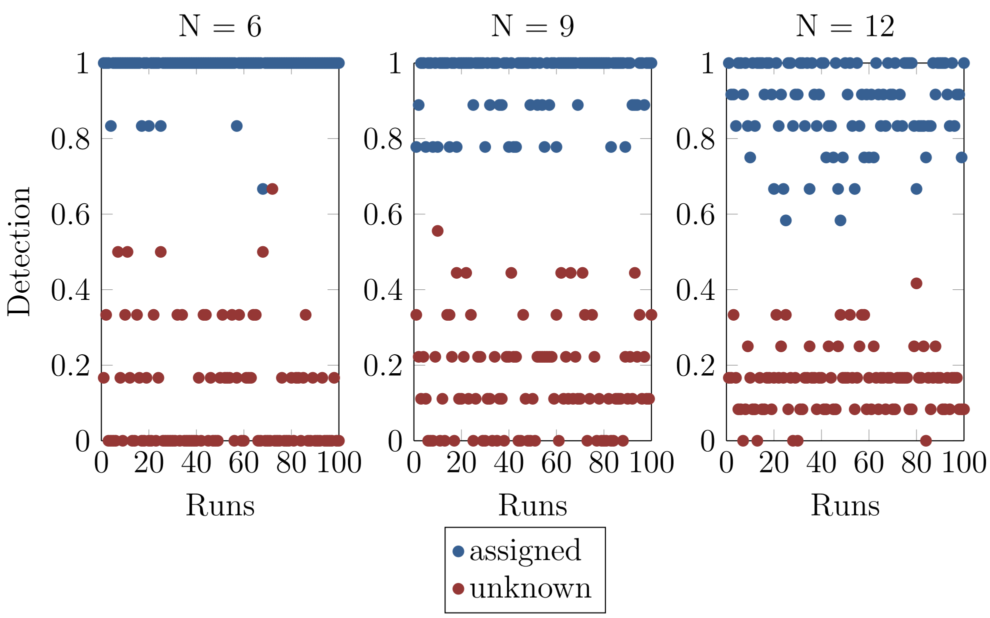

Figure 5.9: *Case study results for As and Unk for synthetic generated power draws for each simulation run*

The results show that with an increasing number of power states the ACC_e is decreasing. The number of assignable power states is sufficient and also the mean number of unknown and false detected power states is below 2.

The results of the real world data based case study is presented in Table 5.5 for house 1-3 for 12 consecutive days. The detected power states from the submetered power measurements as well as As, $Unkn$ and the ACC_e are presented to evaluate the proposed approach.

The results for house 1 and 2 are promising and sufficient. In house 3, an ACC_e of 0.64 is reached due to not assignable power states presented by $Unkn$. As sub-reason we can assume that two appliances were simultaneously operated and the power states were not detected by the detection approach. The range of this power state was out of consideration area. The power state to be detected was around $4.7\,kW$ and we considered only power values up to $3\,kW$. This explains 2 items of $Unkn$, but still 3 states were wrong detected and not assignable. Finally, we claim that the presented results of the real world data case study are varying according to used parameter set as for example the used threshold to detected power states from the submetered data.

House	power states	As	$Unkn$	ACC
1	[108, 209, 272, 413, 742, 895, 1039, 1356, 1530, 1640, 2696]	11	1	0.92
2	[79 164 238 406 463 573 760 1044 1185 1646 1834]	11	1	0.92
2	[157 232 411 479 730 864 1001 1102 1287 1478 1569 1701 2231 2464]	9	5	0.64

Table 5.5: *Case study results for As, Unk and ACC for real world based power measurements (REDD House 1-3) with the set of detected power states computed on the submetered power measurements*

5.4.5 Overall NILM framework

This case study aims to evaluate the performance of the proposed unsupervised load disaggregation approach on the whole NILM process. Our approach includes the four processing stages *feature detection*, *appliance creation with state clustering*, *state estimation and appliance classification* and *appliance database update*. In detail, we varied the observation time of event detection and appliance update phase in a way that we run the process every sixth day and we run the classification on each sixth days out of thirty days in total. Thus, we computed four classification results since the first 6 days were used to learn the initial set of appliance states. Due to the fact that the presented unsupervised approach provides information of on/off appliances, the initial task of the case study is to assign power states to "virtual" appliances (VD). Each appliance based on ground truth data has their certain power states in which one appliance can have more than one power state. Moreover, power states between appliances can be similar to each other. The proposed detection algorithm is not able to distinguish between multi-state devices and similar power states. It is necessary to group detected power states to appliance power states based on the ground truth appliance data. This is done by defining "virtual" appliances combining the power states of multi-state and similar power states to one device. For example two devices with the states $\{100, 1000\}$ for device 1 and $\{200, 1001\}$ for device 2 are combined to one VD. Finally, the detected power states are assigned to this VD by a difference measure between the detected power state and states of the VD. The threshold for the difference measure was $75\,W$. In the case that the detected power state cannot be assigned to any virtual device, this power state is forwarded to an "unknown" appliance container. In the "unknown" device container all power states not assignable are collected together.

Results of this case study are presented in Figure 5.10, 5.11, 5.12, 5.10 and in Table 5.6. In detail, the mentioned figures present the energy share for each virtual appliance and unknown appliance for the ground truth and the estimated results. The observation time was always six days, in which the detected appliance set of the previous six days was used for classification. The results are satisfying. In Table 5.6 the accuracy for the classification on virtual appliance level, the assignable detected events, the number of unknown events and the accuracy for the event detection are presented.

Because the algorithm uses a solution and evaluation approach not used by other approaches, a direct comparison with other approaches was not possible for this case study. This is because the algorithm continuously detects appliance power states and uses them by the estimation and classification stage to disaggregate the total power load. Accordingly, the results are changing and also improving from iteration to iteration. In this sense an iteration is every time the set of appliance power states are updated.

5.4.6 Computational Complexity

The proposed load disaggregation approach should be of low computational complexity to work on restricted hardware in soft realtime. Soft realtime in this context means that the algorithm should be able to produce a valid result inbetween the measurement of two consecutive power samples without providing strict guaranties on this timing. The classification process is time-critical since

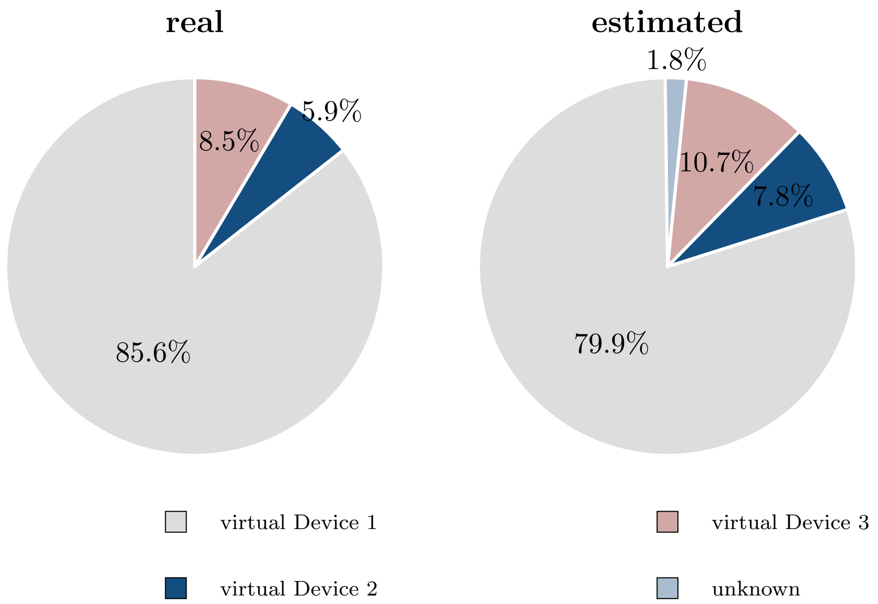

Figure 5.10: *Classification results for the energy shares of REDD houses 1 for the first 6 days*

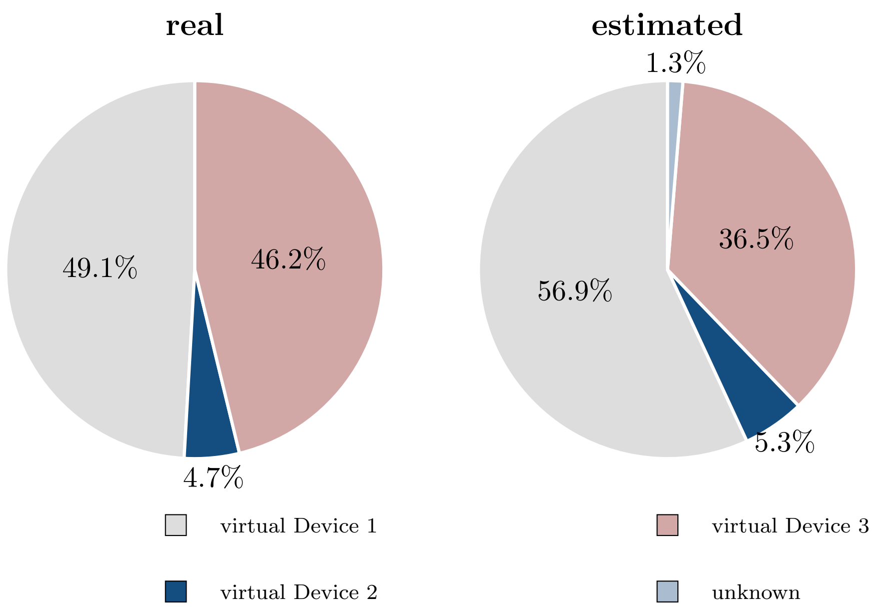

Figure 5.11: *Classification results for the energy shares of REDD houses 1 for the second 6 days*

it should work online (sample per sample). The other three processing stages are performed on an observation window of one or more days and not aiming to be online-capable.

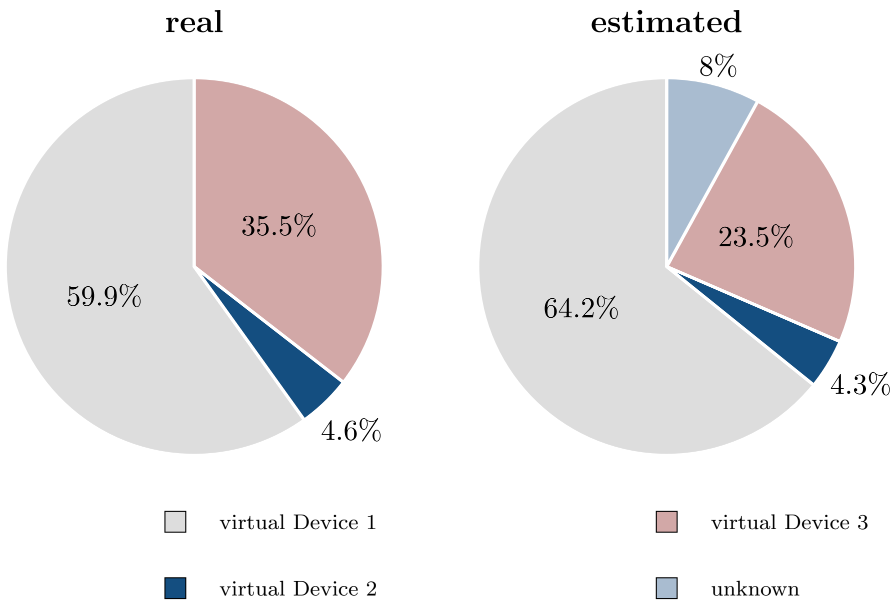

Figure 5.12: *Classification results for the energy shares of REDD houses 1 for the third 6 days*

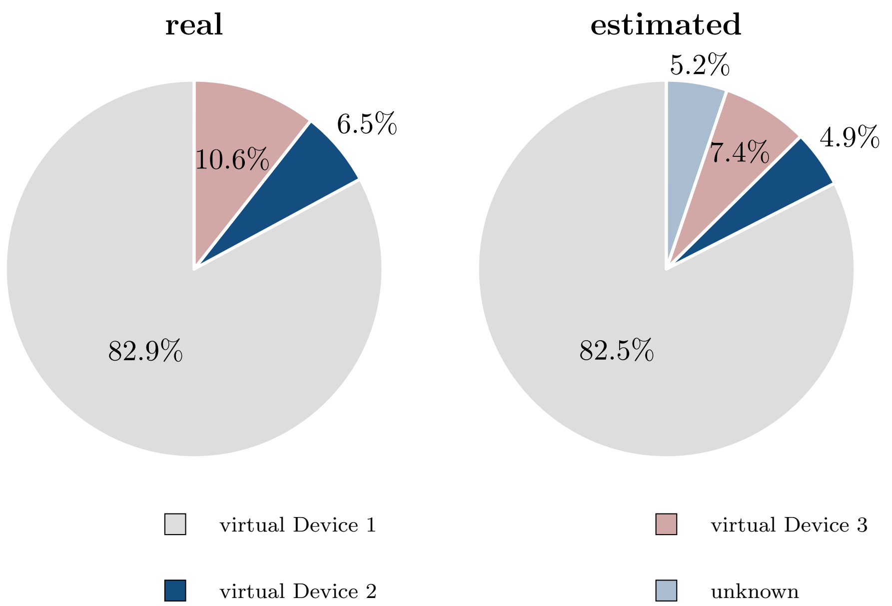

Figure 5.13: *Classification results for the energy shares of REDD houses 1 for the fourth 6 days*

We run simulations on a standard PC and a UDOO Board, which is an

Day	VD 1	VD 2 **ACC**	VD 3	*As*	*Unkn*	*ACC*
1	99.1	99.7	99.8	6	1	0.5
2	98.0	99.5	99.4	11	1	0.92
3	98.2	99.5	99.4	9	2	0.7
4	99.4	99.4	99.5	10	2	0.77
total	*98.7*	*99.5*	*99.6*	*9*	*1.5*	*0.72*

Table 5.6: *Case study results for As, Unk and ACC for real world based power measurements (REDD House 1-3) with the set of detected power states computed on the submetered power measurements*

embedded development platform based on a Freescale ARM processor[3]. We measure the time for each classification iteration and evaluate the time according to reached mean and standard deviation of the computational time. In contrast to the previous section, the implementation based on MATLAB was translated and implemented in C++ making the algorithm platform-independent working. We used synthetically generated power data in which we varied the number of appliances between $N \in \{6, 12, 18\}$ with a varying number of particles $p \in \{100, 500, 1000\}$. In Figure 5.14, the errorbar of the reached computational times for different implementations and settings on the load disaggregation problem are presented. The C++ implementation is in general at least 10 times faster than the Matlab implementation. Computational times below $10\,ms$ can be reached. The C++ implementation running on an embedded hardware as the UDOO reaches computational times below $100\,ms$ and show is applicability to run on embedded hardware. Considering a sampling frequency of $1\,s$ we claim that the approach is fulling the wanted soft-real time capability. The approach provides a valid classification result for each time instance of the sampling process. The computational time is increasing linear by the number of wanted appliances and the number of used particles. We claim that the approach is scalable to the number of used power states by the number of used particles for the estimation process.

5.5 Discussion

In the previous section we evaluated the proposed load disaggregation approach by performing different case studies. The presented algorithm is unsupervised,

[3]http://www.udoo.org/

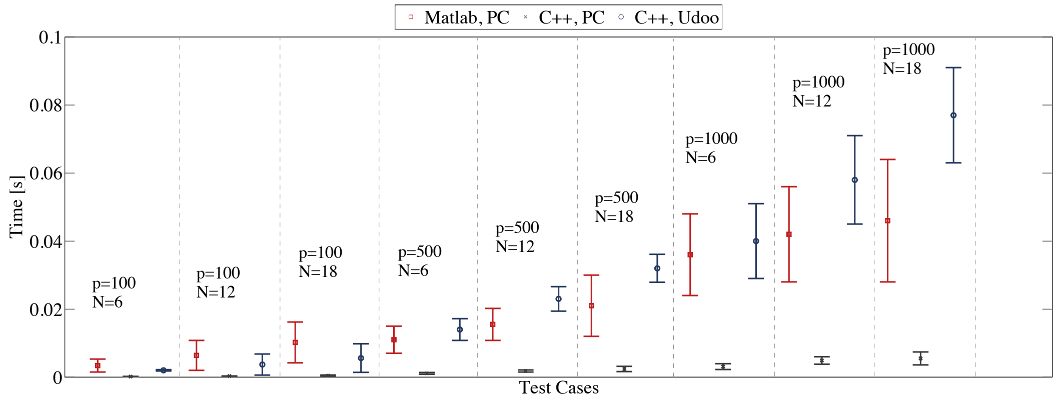

Figure 5.14: *Error bar of the computational time of the classification process on different hardware and different implementations. The number of used particles ($p \in \{100, 500, 1000\}$) and number of appliances ($N \in \{6, 12, 18\}$) is varied*

needs no device information as the usual power consumption or the structure of the device. The approach detects on/off appliances. It is able to detect unique power states in aggregated power draws. According to the fact that detected power events also include events from multi-state appliances, the presented approach has to be enhanced to multi-state appliances in future work. The power state detection phase is working sufficient to detect steady state events, but suffers on the problem of power events lasting over several seconds before remaining in steady state. This introduces errors in the detection process which can be fairly compensated by parameter tuning of the algorithm. Moreover, the algorithm shows on the one hand that it can work with inaccurate power measurements and power state models as well as with predefined transition matrices. The proposed approach is based on HMMs which are known to be good at modelling the combination of stationary processes with continuous data over discrete time but are computational expensive. The learning process is computational complex and not working online. The present load disaggregation framework solves this constraint by being able to work with predefined transition matrices without a certain learning stage. Moreover, the algorithm is proofing to be of low computational complexity based on the case study of Section 5.4.6. The algorithm is scalable by the number of used particles. The more particles are used the better is the classification results. In general the PF has also the advantage to be easily parallelizable.

The approach as presented is highly dependent on the chosen parameters. By changing the threshold value from 25 to 50 W whole NILM result is changing to some extent. Although the choice of a threshold value is highly dependent on the used household and their included appliances. Future work should perform a parameter evaluation or even introduce an automatic parameter tuning stage to make the approach applicable also for large deployments.

Moreover, we claim that the presented approach is employable for real world scenarios. This is shown by the low computation complexity able to work also on restricted hardware and the improving estimation and classification behavior over time without any knowledge of the load disaggregation environment. Finally, it is also worthy to discuss that the algorithm is currently not able to label detected appliance states to a real appliance. This and the scalability to a higher appliance number is necessary and the next step to make the algorithm general deployable for real households.

5.6 Summary

In this chapter an unsupervised approach to solve the problem to disaggregate appliance power draws from the aggregated power load was presented. The approach autonomously detects the power states of the used appliances at run-time. It improves the saved appliance models in operation and updates the appliance database by adding new appliance models and maintaining saved appliance models. The detected appliance models can be used by the load disaggregator to estimate the appliance states. The algorithm contains a preprocessing stage to de- noise and to smooth the aggregated power draw in a way to be able to detect sharp and significant power edges. Appliance models are established as on/off appliances only with the knowledge of detected power edges and are finally used by the load disaggregator based on particle filtering. The approach is working unsupervised and online, and can work on restricted hardware such as embedded computers. In detail, the approach can be split into the following four parts i) feature detection, ii) state clustering & appliance creation iii) state estimation and appliance classification and iv) appliance database update. Each stage has been evaluated on synthetic and/or real-world measurement data. The results show that the number of detected appliance states and the corresponding disaggregation result is sufficient. In summary, the work contributes to current state-of-the-art algorithms by being an unsupervised active power based (based on one feature) load disaggregation approach working online without any system information (e.g., number of appliances, power states of appliances) and being of low computational complexity able to work on embedded hardware.

CHAPTER 6 Conclusion, Limitations and Future Work

"All things are difficult before they are easy."

– Thomas Fuller

In this chapter, we conclude and summarize the proposed approaches for NILM. We provide information about limitations of the proposed approaches and show directions for future work based on the contributions of this work. Finally, we also present peer-reviewed works related to the topic of the thesis.

6.1 Conclusion

To improve the energy awareness and the energy efficiency in homes, appliance-level energy feedback could be the holy grail of energy efficiency [Arm13]. Non-intrusive load monitoring, which is a single monitor solutions aims to provide appliance-level energy feedback. It uses appliance characteristics and smart algorithms to break down the aggregated power draw to its appliance components. In this thesis, a comprehensive overview on the research of load disaggregation has been provided and three novel applications on NILM have been proposed. In detail, we have started this thesis to present a comprehensive survey of load monitoring techniques. This included distributed load monitoring techniques such as intrusive load monitoring. In this sense also load identification techniques to detect connected or plugged appliances were discussed. By identifying the disadvantages of the distributed load monitoring environment, we have focused the continued thesis on NILM approaches overcoming these disadvantages. Accordingly, we have concentrated on three different load disaggregation applications starting with an optimization-based load disaggregation approach modelled as a knapsack problem. In this sense, the problem of load disaggregation has been modelled by a modified knapsack problem. Six different metaheuristic optimization approaches have been used as load disaggregator. The main idea was to create the best composition of appliance power states to estimate the total power demand of a household for each point in time. The approaches have been evaluated on real-world data in which the used sets of appliance power states were different. One power state set consisted of unambiguous, not similar power states based on power detections on the aggregated power draws. In the other case study, we have used submetered power draws to identify the most common power states. This resulted into power states having ambiguous appliance power states. The evaluations of the proposed approach have shown that the approach was able to estimate the total power draw with acceptable error. But the approach was not able to distinguish between similar power states. The amount of information based on the used feature set was insufficient in the context of similar power states. With a unique set of power states the classification result has shown an improved behavior. We have shown that more features are needed to possibly overcome the problem of similar and noisy power states and confirms the statement of Hart [Har92] that the load disaggregation problem is highly affected by similarities and noise influences. We have shown by example that the use of a simple optimization approach as a knapsack model is not able to solve the problem as a whole.

Motivated by the problem of similar classification features and by the fact that there is no measure to compare different load disaggregation problems, we have introduced two complexity measures to classify the load disaggregation problem. Different data sets and evaluation metrics are existing, but there is no general defined view on the load disaggregation problem itself considering the used feature set and system assumptions. For example, one could use a sampling frequency of 1 second and another one 1min. Accordingly, the used feature set will be different and also the problem complexity. Thus, the introduced complexity measures are based on the appliance power states as well as on noise and appliance modelling errors. One complexity measure describes the appliance set complexity based on the appliance model description. The other complexity measure considers the appliance usage over time for a certain household power draw. The introduced complexity measures have been evaluated on three different real-world datasets and have shown that they are applicable to classify different load disaggregation problems. By using the proposed complexity measures it is possible to classify the disaggregation problem, to make load disaggregation results fairly comparable and to be independent of the used algorithm and data pre-processing computations.

Finally, in this thesis we have introduced a load disaggregation approach which is unsupervised, needs no system information, is improving and updating over time, is working online and is of low computational complexity to run on embedded hardware. The approach consists of four computational steps i) feature detection, ii) state clustering and appliance creation, iii) state estimation and appliance classification and iv) appliance database update. It has been evaluated on synthetic and/or real world consumption data. Different test scenarios such as i) synthetic data evaluation, ii) real-world data evaluation, iii) transition matrix dependency, iv) appliance model detection, v) overall NILM framework and vi) the computational complexity have been evaluated. The results are promising with high classification accuracies (over 90%, dependent on the problem) and low energy estimation errors. Additionally, we have shown that requirements such as online capability, no system information and low computation complexity were met.

In summary, this thesis has introduced and has evaluated three different perspectives on the load disaggregation problem. Each of these perspectives and corresponding approaches contribute to the research area of NILM on its own. We believe that the presented approaches and their introduced system and model assumptions are valid and applicable for many energy-related applications and results. However, we also recognized that each of the approaches could be extended and improved to make it work on more general and large-scale scenarios. The following two sections are presenting current approach limitations and accordingly, introducing future research directions based on this thesis.

6.2 Restrictions

Although the presented approach achieved several sufficient and satisfying result, this chapter aims to discuss restrictions of the presented approaches. We will describe several scenarios where the present approaches are less successful. The presented limitations should be used to motivate the future work in the next section.

The following limitations are valid for both load disaggregation approaches presented in Chapter 3 and Chapter 5:

- In today's homes many appliances (e.g. home entertainment appliances) are controlled over multiple outlets at the same time. Two or more appliances might be turned off or turned on the exact same time. The starting time of devices will be the same and the presented approaches of Chapter 3 and 5 will fail to distinguish between multiple appliances. The approaches are based on distinguishable power edges which is not possible for concurrent events.

- Another possible limitation is that home appliances can have power profiles without general steady states. Model assumptions (e.g., on/off appliances, multi-state appliance) as in this work are not applicable by the use of variable-load consuming appliances. These appliances are changing their power continuously over time due to their current operation state and load effort. Also appliances with a highly complex transition behavior for different operation cycles could lead to problems by increasing to considered appliance state space.

- It also imaginable that a household owns a multiple number of the same appliance such as a household having two identical air condition systems or fridges. The presented approaches will fail to distinguish between two devices of the same type having the same consuming characteristics.

Additionally, Chapter 4 introducing two new complexity measures for load disaggregation has limitations. The presented complexity measures are dependent on the used appliances having their predefined feature set. The used feature set consists out of the model structure of the appliance, the power demand of all appliance states and the used noise assumptions due to influence of measurement and modelling errors (for all appliances the same noise assumptions are valid). Considering the variety of different appliance dataset and their different measurement quantities (such as active power, reactive power, current waveforms, etc.) the presented approaches have to be modified.

6.3 Future Work

The topics presented in this thesis created several future research perspectives which can be followed in future research:

- The introduced complexity measures for the load disaggregation problem showed to be beneficial to classify the load disaggregation problem. The proposed measures are based on active power. Only one feature is considered which is steady state based. Future work should consider the extension of the approach by considering multiple features. The features set should either based on steady state or/and on transient state features. This enhancement would make the complexity measure applicable to classify the load disaggregation problem in general since many load disaggregation problems are using different feature sets and sampling frequencies.

- The load disaggregation community is highly dependent on published and freely available consumption datasets. In recent years, many datasets have been published which are usable on different applications. Different datasets have different houses with a different set of used appliances. Measurements from a particular house could be more complex to disaggregate than measurements from another house. Accordingly, there is the need to get a classified load disaggregation problem according to the task you want to solve. A general test case should be generated as it is done in other disciplines (e.g. objective function for optimization approaches). The proposed complexity measure should be used to produce rated load disaggregation problems which are re-producible, repeatable and divisible. It is imaginable to generate a total power draw based on a determined complexity value in which appliances power draws are generated synthetically or retrieved from real-world consumption data.

- Based on the complexity measures and the enhancement with multiple features, the measure could be used to identify the most relevant appliance features and their combinations. This will help to introduce and to improve appliance modelling approaches.

- The proposed unsupervised load disaggregation approach should be improved in the appliance clustering/modelling stages as well as in the feature detection stage. The feature detection stage should be extended to extract additional characteristic appliance features based on active power measurements. Imaginable features are time of use, usage duration and usage frequency. Due to the extended feature set also the classification approach should be modified. Moreover, the current approach uses only on/off appliances. Future work should be able to use multi-state appliances. Repeating behaviors of consecutive power states and power state combinations are grouped to create multi-state appliances.

- Current tests on the load disaggregation approaches and with the complexity measures were made on relatively small deployments. Future work should aim to test the approaches on large-scale deployments to detect algorithmic problems or wrong and mature approach assumption. This generalizes the introduced approach to work in each considered household.

- The load disaggregation approach should be adopted to work also for commercial buildings. The load disaggregation approach has to deal with a different environmental situations in which a commercial building has in general a high number of appliances. energy is consumed at almost the same times over the day. Several appliance types are appearing multiple times increasing the probability of similar appliance power profiles.

- In our opinion the metaheuristic approach can be improved by including multiple features for the evaluation and using a multi-objective optimization approach.

- Our proposed unsupervised load disaggregation approach delivers device statuses by neglecting the actual device type. It is not possible to automatically identify appliance 1 as a fridge. Therefore, a labelling approach has to be introduced. This can be done either by expert knowledge with the help of user feedback (e.g., notifications by the mobile phone which appliance was currently turned on) or by an automatic labelling process. Automatic labelling can be achieved by extracting meta data of the detected appliances such as the time of use or the usage frequency.

6.4 Related Publications

In total, the author of this thesis published 18 publications in workshops, conferences and journals. Another two manuscripts are currently under review for publication. All mentioned publications are related to energy applications. In the following, we list related peer-reviewed publications we have written and are not discussed in this thesis. The publications were presented and discussed with the scientific community. The content of the papers contributed to the background knowledge of the author related to energy applications and to NILM. The descriptions of the papers are based on the abstracts of each work.

- **Proficiency of Power Values for Load Disaggregation [Pöc15]** Load disaggregation techniques infer the operation of different power consuming devices from a single measurement point that records the total power draw over time. Thus, a device consuming power at the moment can be understood as information encoded in the power draw. However, similar power draws or similar combinations of power draws limit the ability to detect the currently active device set. We present an information coding perspective of load disaggregation to enable a better understanding of this process and to support its future improvement. In typical cases of quantity and type of devices and their respective power consumption, not all possible device configurations can be mapped to distinguishable power values. We introduce the term of proficiency to describe the suitability of a device set for load disaggregation. We provide the notion and calculation of entropy of initial device states, mutual information of power values and the resulting uncertainty coefficient or proficiency. We show that the proficiency is highly dependent from the device running probability especially for devices with multiple states of power consumption. The application of the concept is demonstrated by artificial data as well as with actual power consumption data from real-world power draw datasets.
- **Worried About Privacy? Let Your PV Converter Cover Your Electricity Consumption Fingerprints [Rei15]:** Solar power has

emerged as one of the three most widely installed renewable energy sources around the globe. Photovoltaic (PV) capacity in excess of 150 GW had been installed in 2013 already, and many more installations are connected to worldwide power grids every day; especially in the form of small-scale PV plants in domestic environments. However, in order to connect PV installations to the power grid, their dc output must be converted to the nominal mains voltage and frequency through the use of converters. In this paper, we propose a novel approach to influence the maximum power point tracking (MPPT) component of such a PV converter in order to enable two main privacy-preserving operations: Firstly, by deliberately reducing the output power through changing the converter's operating point, appliance operations can be emulated in order to pretend user presence during periods of absence. Secondly, by running the converter below optimum output power, and feeding real-time data of an appliance consumption to the device, it is able to hide the appliance's operation from the household's aggregate consumption. We present simulations results that prove how our modified converter design can hide appliance load signatures as well as how it can be used to emulate appliance signatures to falsely indicate user presence.

- **Load hiding of household's power demand [Ega14b]:** With the development and introduction of smart metering, the energy information for costumers will change from infrequent manual meter readings to fine-grained energy consumption data. On the one hand these fine-grained measurements will lead to an improvement in costumers' energy habits, but on the other hand the fined-grained data produces information about a household and also households' inhabitants, which are the basis for many future privacy issues. To ensure household privacy and smart meter information owned by the household inhabitants, load hiding techniques were introduced to obfuscate the load demand visible at the household energy meter. In this work, a state-of-the-art battery-based load hiding (BLH) technique, which uses a controllable battery to disguise the power consumption and a novel load hiding technique called load-based load hiding (LLH) are presented. An LLH system uses a controllable household appliance to obfuscate the household's power demand. We evaluate and compare both load hiding techniques on real household data and show that both techniques can strengthen household privacy but only LLH can increase appliance level privacy.

- **GREEND: An energy consumption dataset of households in Italy and Austria [Mon14a]:** Home energy management systems can be used to monitor and optimize consumption and local production from renewable energy. To assess solutions before their deployment, researchers and designers of those systems demand for energy consumption datasets. In this paper, we present the GREEND dataset, containing detailed power usage information obtained through a measurement campaign in households in Austria and Italy. We provide a description of consumption scenarios and discuss design choices for the sensing infrastructure. Finally, we benchmark the dataset with state-of-the-art techniques in load disaggregation, occupancy detection and appliance usage mining.

- **European end-users level of energy consumption and current structural barriers for smart homes: A case study of residential sectors in Austria and Italy [Kha14]:** This article presents a quantitative assessment of the level of energy consumption of inhabitants located in Carinthia and Friuli-Venezia Giulia. In addition, an analysis for the current structural barriers for smart powered homes and smart energy management systems is conducted. A questionnaire consisting of 43 questions is used to address the aforementioned issues. In particular, a sample size of 385 respondents with a confidence of 95% and marginal error of 5% is found to be representative of the adopted area. Based on the results, we modeled the average energy consumption of a typical 110 m^2 area household with 16.8 kWh/day, a 2.6 kW peak, and a load factor of 27%. Furthermore, an average of 46% of the respondents expressed the willingness to exploit tariff systems for operating their electrical appliances, and about two thirds of the respondents declared that they care about the energy efficiency at their households. However, low renewable energy utilization is observed due to some existing structural barriers. Therefore, an analysis and a discussion are carried out to investigate these barriers. Finally, some recommendations are provided according to the obtained results.

- **YOMO - The Arduino based smart metering board [Kle15a],[Kle15b]:** Smart meters are an enabling technology for many smart grid applications. This paper introduces a design for a low-cost smart meter system as well as the fundamentals of smart metering. The smart meter platform, provided as open hardware, is designed with a connector interface compatible to the Arduino platform, thus opening the possibilities for smart meters with flexible hardware and computation features, starting from low-cost 8 bit micro controllers up to powerful single board computers that can run Linux. The metering platform features a current transformer which allows a non-intrusive installation of the current measurement unit. The suggested design can switch loads, offers a variable sampling frequency, and provides measurement data such as active power, reactive and apparent power. Results indicate that measurement accuracy and resolution of the proposed metering platform are sufficient for a range of different applications and loads from a few Watts up to five kilowatts.

- **Techno-economical assessment of grid-connected photovoltaic power systems productivity in summer season in Klagenfurt, Austria [Sch14]:** This paper shows the productivity of grid-connected photovoltaic systems, to plan future investments in the region around Klagenfurt, Austria. To evaluate the usage of grid-connected PV systems, a methodology based on mainly three factors is presented. These factors are yield factor, capacity factor and cost of energy. The analysis is done by a model based on monitored PV data and meteorological data in an interval of 15 minutes. Meteorological data was recorded from May 2013 until July 2014 with a lack of data from September 2013 until January 2014. By analyzing the collected data, it is found that the daily average solar energy in Klagenfurt was 6613 Wh/m2 in the time of research. The results of the research show that future investments in grid-connected PV systems in Klagenfurt can be profitable. The yield factor over the time of research (293 days) is 1263.4 kWh/kWp, whereas the capacity factor of the proposed system is 17.97%. The cost of energy is found to be 0.2348 /kWh. This is a satisfactory result compared with values of other European counties.

- **Integrating households into the smart grid [Mon13]:** The success of the Smart Grid depends on its ability to collect data from heterogeneous sources such as smart meters and smart appliances, as well as the utilization of this information to forecast energy demand and to provide value-added services to users. In our analysis, we discuss requirements for collecting and integrating household data within smart grid applications. We put forward a potential system architecture and report stateof-the-art technologies that can be deployed towards this vision.

- **Design guidelines for smart appliances [Elm12]:** Embedded intelligence can help controlling and reducing the energy consumption of appliances to a significant amount. Such a smart appliance will consist of a communication interface, a local processing and decision unit and the appliance's actual function. Sophisticated functions for such a device will involve a notion of real-time with a respective time format, a generic database that contains energy usage logs, error messages, warnings and real-time measurements for power usage, and an embedded self-description that allows to integrate the device into a system with minimum manual configuration. While there exists concepts for smart plugs and smart outlets that can be applied to "smarten" an existing device, in general we need to assume that the variety of appliances and technologies will require the support for various architectures including software solutions that integrate into the functions of an appliance with existing computing power, e.g. a DVD player or a state-of-the-art television set. Thus there is a need for architectural services with flexibility for different hosting systems while keeping the interoperability with respect to a smart home control system.

- **Integration of legacy appliances into home energy management systems [Ega15c]:** The progressive installation of renewable energy sources requires the coordination of energy consuming devices. At consumer level, this coordination can be done by a home energy management system (HEMS). Interoperability issues need to be solved among smart appliances as well as between smart and non-smart, i.e., legacy devices. We expect current standardization efforts to soon provide technologies to design smart appliances in order to cope with the current interoperability issues. Nevertheless, common electrical devices affect energy consumption significantly and deserve consideration within energy management applications. This paper discusses the integration of smart and legacy devices into a generic system architecture and, subsequently, elaborates the requirements and components which are necessary to realize such an architecture including an application of load detection for the identification of running loads and their integration into existing HEM systems. We assess the feasibility of such an approach with a case study based on a measurement campaign on real households. We show how the information of detected appliances can be extracted in order to create device profiles allowing for their integration and management within a HEMS.

- **An Open Solution to Provide Personalised Feedback for Building Energy Management [Mon15]:** The integration of renewable energy sources in- creases the complexity in mantaining the power grid. In particular, the highly dynamic nature of generation and consumption demands for a better utilization of energy resources, which seen the cost of storage infrastructure, can only be achieved through demand-response. Accordingly, the avail- ability of energy and potential overload situations can be reflected using a price signal. The effectiveness of this mechanism arises from the flexibility of device operation, which is nevertheless heavily reliant on the exchange of information between the grid and its consumers. In this paper, we investigate the capability of an interactive energy management system to timely inform users on energy usage, in order to promote an optimal use of local resources. In particular, we analyse data being collected in several households in Italy and Austria to gain insights into usage behavior and drive the design of more effective systems. The outcome is the formulation of energy efficiency policies for residential build- ings, as well as the design of an energy management system, consisting of hardware measurement units and a management software. The Mjölnir framework, which we release for open use, provides a platform where various feedback concepts can be implemented and assessed. This includes widgets displaying disaggregated and aggregated consumption information, as well as daily production and tailored advices.

- **Smart Grid: Visionen** & **Herausforderungen [Ega14a]:** Book chapter introducing the term Smart Grid and its current and future visions and challenges

Bibliography

[And12a] K. Anderson, M. Berges, A. Ocneanu, D. Benitez, and J. M. F. Moura. Event Detection for Non Intrusive Load Monitoring. In *Proceedings of the 38th Annual Conference on IEEE Industrial Electronics Society (IECON)*, 2012.

[And12b] K. Anderson, A. Ocneanu, D. Benitez, D. Carlson, A. Rowe, and M. Berges. BLUED: A Fully Labeled Public Dataset for Event-Based Non-Intrusive Load Monitoring Research. In *Proceedings of the 2nd KDD Workshop on Data Mining Applications in Sustainability (SustKDD)*, 2012.

[Arm13] K. C. Armel, A. Gupta, G. Shrimali, and A. Albert. Is disaggregation the holy grail of energy efficiency? The case of electricity. *Energy Policy*, 52(0):213 – 234, 2013.

[Aru02] M. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *IEEE Trans. Signal Process.*, 50(2):174–188, 2002.

[Bar04] M. Baranski and J. Voss. Genetic algorithm for pattern detection in NIALM systems. In *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, 2004.

[Bar12] S. Barker, A. Mishra, D. Irwin, E. Cecchet, P. Shenoy, and J. Albrecht. Smart*: An open data set and tools for enabling research in sustainable homes. *SustKDD, August*, 2012.

[Bar14a] S. Barker, M. Musthag, D. Irwin, and P. Shenoy. Non-intrusive load identification for smart outlets. In *Proceedings of the IEEE Conference on Smart Grid Communications (SmartGridComm)*, 2014.

[Bar14b] S. Barker, S. Kalra, D. Irwin, and P. Shenoy. NILM Redux: The Case for Emphasizing Applications over Accuracy. In *Proceedings of the Second International Workshop on Non-Intrusive Load Monitoring (NILM)*, 2014.

[Bat13] N. Batra, M. Gulati, A. Singh, and M. B. Srivastava. It's Different: Insights into Home Energy Consumption in India. In *Proceedings of the 5th ACM Workshop on Embedded Systems for Energy-Efficient Buildings*, 2013.

[Bec14] C. Beckel, W. Kleiminger, R. Cicchetti, T. Staake, and S. Santini. The ECO Data Set and the Performance of Non-intrusive Load Monitoring Algorithms. In *Proceedings of the 1st ACM Conference on Embedded Systems for Energy-Efficient Buildings*, BuildSys '14, 2014.

[Bel13] C. Belley, S. Gaboury, B. Bouchard, and A. Bouzouane. Activity recognition in smart homes based on electrical devices identification. In *Proceedings of the 6th International Conference on Prrvasive Technologies Related to Assistive Environments*, 2013.

[Böh09] C. Böhringer, T. F. Rutherford, and R. S. Tol. The EU 20/20/2020 targets: An overview of the EMF22 assessment. *Energy economics*, 31:S268–S273, 2009.

[Car13] D. R. Carlson, H. S. Matthews, and M. Bergs. One size does not fit all: Averaged data on household electricity is inadequate for residential energy policy and decisions. *Energy and Buildings*, 64(0):132 – 144, 2013.

[Cha10] H.-H. Chang, C.-L. Lin, and J.-K. Lee. Load identification in non-intrusive load monitoring using steady-state and turn-on transient energy algorithms. In *Proceedings of the 14th International Conference on Computer Supported Cooperative Work in Design (CSCWD)*, 2010.

[Cha12] H.-H. Chang. Non-Intrusive Demand Monitoring and Load Identification for Energy Management Systems Based on Transient Feature Analyses. *Energies*, 5(11):4569–4589, 2012.

[Che15] K.-Y. Chen, S. Gupte, E. C. Larson, and S. Patel. DOSE: Detecting User-Driven Operating States ofElectronic Devices from a Single Sensing Point. In *Proceedings of the IEEE Conference on Pervasive Computing and Communication (PERCOM)*, 2015.

[Cos05] M. Costa, A. Goldberger, and C.-K. Peng. Multiscale entropy analysis of biological signals. *Phys. Rev. E*, 71:021906, Feb 2005.

[Cox06] R. Cox, S. Leeb, S. Shaw, and L. Norford. Transient event detection for nonintrusive load monitoring and demand side management using voltage distortion. In *Proceedings of the 21st IEEE Applied Power Electronics Conference and Exposition (APEC)*, 2006.

[Dar06] S. Darby et al. The effectiveness of feedback on energy consumption. *A Review for DEFRA of the Literature on Metering, Billing and direct Displays*, 486:2006, 2006.

[Das11] S. Das and P. Suganthan. Differential Evolution: A Survey of the State-of-the-Art. *IEEE Transaction on Evolutionary Computation*, 15(1):4–31, Feb 2011.

[Don13] M. Dong, P. C. M. Meira, W. Xu, and C. Y. Chung. Non-Intrusive Signature Extraction for Major Residential Loads. *IEEE Transactions on Smart Grid*, PP(99):1 –10, 2013.

[Don14] R. Dong, L. Ratliff, H. Ohlsson, and S. S. Sastry. Fundamental Limits of Nonintrusive Load Monitoring. In *Proceedings of the 3rd International Conference on High Confidence Networked Systems*, HiCoNS '14, 2014.

[Eft10] C. Efthymiou and G. Kalogridis. Smart grid privacy via anonymization of smart metering data. In *Smart Grid Communications (SmartGridComm), 2010 First IEEE International Conference on*, pages 238–243. IEEE, 2010.

[Ega13a] D. Egarter, V. P. Bhuvana, and W. Elmenreich. Appliance State Estimation Based on Particle Filtering. In *Proceedings of the 5th ACM Workshop on Embedded Systems For Energy-Efficient Buildings BuildSys*, 2013.

[Ega13b] D. Egarter and W. Elmenreich. EvoNILM: evolutionary appliance detection for miscellaneous household appliances. In *Proceedings of the 15th annual conference companion on Genetic and evolutionary computation*, pages 1537–1544. ACM, 2013.

[Ega13c] D. Egarter, A. Sobe, and W. Elmenreich. Evolving Non-Intrusive Load Monitoring. In *Proceedings of the Applications of Evolutionary Computation Conference*, 2013.

[Ega14a] D. Egarter, A. Monacchi, M. Pöchacker, K. Schweiger, and B. Steinwender, Editors. *Energie - Interdisziplinre Perspektiven auf eine knappe Ressource*. Profil, 2014.

[Ega14b] D. Egarter, C. Prokop, and W. Elmenreich. Load hiding of household's power demand. In *Smart Grid Communications (SmartGridComm), 2014 IEEE International Conference on*, 2014.

[Ega15a] D. Egarter, V. P. Bhuvana, and W. Elmenreich. PALDi: Online Load Disaggregation via Particle Filtering. *IEEE Transactions on Instrumentation and Measurement*, 64(2):467–477, Feb 2015.

[Ega15b] D. Egarter and W. Elmenreich. Autonomous Load Disaggregation Approach based on Active Power Measurements. In *Proc. IEEE Workshop on Pervasive Energy Services (PerEnergy)*, 2015.

[Ega15c] D. Egarter, A. Monacchi, T. Khatib, and W. Elmenreich. Integration of legacy appliances into home energy management systems. *Journal of Ambient Intelligence and Humanized Computing*, 2015.

[Ega15d] D. Egarter, M. Pöchacker, and W. Elmenreich. Complexity of Power Draws for Load Disaggregation. *arXiv preprint arXiv:1501.02954*, 2015.

[Eib03] A. E. Eiben and J. E. Smith. *Introduction to Evolutionary Computing.* SpringerVerlag, 2003.

[Elm12] W. Elmenreich and D. Egarter. Design guidelines for smart appliances. In *Proceedings of the Tenth Workshop onIntelligent Solutions in Embedded Systems (WISES)*, 2012.

[EM10] K. Ehrhardt-Martinez, K. A. Donnelly, S. Laitner, et al. Advanced metering initiatives and residential feedback programs: a meta-review for household electricity-saving opportunities. American Council for an Energy-Efficient Economy Washington, DC, 2010.

[EST12] Energy Saving Trust. Powering the Nation. *Technical Report*, 2012.

[Far10] A. Faruqui, S. Sergici, and A. Sharif. The impact of informational feedback on energy consumptionA survey of the experimental evidence. *Energy*, 35(4):1598–1608, 2010.

[Fig13] M. Figueiredo, B. Ribeiro, and A. de Almeida. Electrical Signal Source Separation Via Nonnegative Tensor Factorization Using On Site Measurements in a Smart Home. *IEEE Trans. Instrum. Meas.*, PP(99):1–1, 2013.

[Fis08] C. Fischer. Feedback on household electricity consumption: a tool for saving energy? *Energy efficiency*, 1(1):79–104, 2008.

[For09] R. Ford and C. Church. *Reducing Domestic Energy Consumption Through Behaviour Modification.* PhD Thesis, Oxford University, 2009.

[Gan13] A. H. Gandomi, X.-S. Yang, and A. H. Alavi. Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems. *Engineering with Computers*, 29(1):17–35, 2013.

[Gao12] J. Gao, J. Hu, and W.-w. Tung. Entropy measures for biological signal analyses. *Nonlinear Dynamics*, 68(3):431–444, 2012.

[Gis13] C. Gisler, A. Ridi, D. Zujferey, O. Khaled, and J. Hennebert. Appliance consumption signature database and recognition test protocols. In *In Proceedings of the 8th Internation Workshop on Systems, Signal Processing and their Applications (WoSSPA)*, 2013.

[Gon11] H. Goncalves, A. Ocneanu, and M. Berges. Unsupervised disaggregation of appliances using aggregated consumption data. In *Proceedings of the KDD Workshop on Data Mining Applications in Sustainability (SustKDD)*, 2011.

[Gre12] U. Greveler, B. Justus, and D. Loehr. Multimedia content identification through smart meter power usage profiles. *Computers, Privacy and Data Protection*, 1:10, 2012.

[Gup10] S. Gupta, M. S. Reynolds, and S. N. Patel. ElectriSense: Single-point Sensing Using EMI for Electrical Event Detection and Classification in the Home. In *Proceedings of the 12th ACM International Conference on Ubiquitous Computing*, 2010.

[Haj85] B. Hajek. A tutorial survey of theory and applications of simulated annealing. In *Proceedings of the IEEE Conference on Decision and Control*, 1985.

[Har92] G. Hart. Nonintrusive appliance load monitoring. *Proceedings of the IEEE*, 80(12):1870–1891, 1992.

[Inm89] H. F. Inman and E. L. Bradley. The overlapping coefficient as a measure of agreement between probability distributions and point estimation of the overlap of two normal densities. *Communications in Statistics - Theory and Methods*, 18(10):3851–3874, Jan. 1989.

[Jin11] Y. Jin, E. Tebekaemi, M. Berges, and L. Soibelman. Robust adaptive event detection in non-intrusive load monitoring for energy aware smart facilities. In *Proceedings of the IEEE Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2011.

[Joh13] M. J. Johnson and A. S. Willsky. Bayesian nonparametric hidden semi-markov models. *The Journal of Machine Learning Research*, 14(1):673–701, 2013.

[J04] V. Jntti, S. Alahuhta, J. Barnard, and J. W. Sleigh. Spectral entropy-what has it to do with anaesthesia, and the EEG? *British Journal of Anaesthesia*, 93(1):150–152, 2004.

[Kel14] J. Kelly and W. Knottenbelt. UK-DALE: A dataset recording UK Domestic Appliance-Level Electricity demand and whole-house demand. *ArXiv e-prints*, 59, 2014.

[Kha14] T. Khatib, A. Monacchi, W. Elmenreich, D. Egarter, S. D'Alessandro, and A. M. Tonello. European End-Users Level of Energy Consumption and Attitude Toward Smart Homes: A Case Study of Residential Sectors in Austria and Italy. *Energy Technology & Policy*, 1(1):97–105, 2014.

[Kim11] H. Kim, M. Marwah, M. F. Arlitt, G. Lyon, and J. Han. Unsupervised Disaggregation of Low Frequency Power Measurements. In *Proceedings of the 11th SIAM International Conference on Data Mining*, 2011.

[Kle15a] C. Klemenjak, D. Egarter, and W. Elmenreich. YOMO-The arduino based smart metering board. *Computer Science Research and Develop ment*, 2015.

[Kle15b] C. Klemenjak, D. Egarter, and W. Elmenreich. YOMO-The arduino based smart metering board. In *Proceedings of the Confernce on Energy Informatics*, 2015.

[Kol11] J. Z. Kolter and M. J. Johnson. REDD: A Public Data Set for Energy Disaggregation Research. In *Proceddings of the SustKDD Workshop on Data Mining Applications in Sustainability*, 2011.

[Kol12] Z. Kolter and T. Jaakkola. Approximate Inference in Additive Factorial HMMs with Application to Energy Disaggregation. In *Proceedings of the International Conference on Artifical Intelligence and Statistics*, 2012.

[Lau03] C. Laughman, K. Lee, R. Cox, S. Shaw, S. Leeb, L. Norford, and P. Armstrong. Power signature analysis. *Power and Energy Magazine, IEEE*, 1(2):56–63, Mar 2003.

[Li11] F. Li, B. Luo, and P. Liu. Secure and privacy-preserving information aggregation for smart grids. *International Journal of Security and Networks*, 6(1):28–39, 2011.

[Lia10] J. Liang, S. Ng, G. Kendall, and J. Cheng. Load Signature Study, Part I: Basic Concept, Structure, and Methodology. *IEEE Trans. Power Del.*, 25(2):551–560, 2010.

[Lia14] J. Liao, G. Elafoudi, L. Stankovic, and V. Stankovic. Non-intrusive appliance load monitoring using low-resolution smart meter data. In *Proceedings of the IEEE International Conference on Smart Grid Communications (SmartGridComm'14)*, 2014.

[Lin10] G. Lin, S. Lee, J. Hsu, and W. Jih. Applying power meters for appliance recognition on the electric panel. In *Proceedings of the IEEE Conference on Industrial Electronics and Applications (ICIEA)*, 2010.

[Lis10] M. Lisovich, D. K. Mulligan, S. B. Wicker, et al. Inferring personal information from demand-response systems. *Security & Privacy, IEEE*, 8(1):11–20, 2010.

[Mak13] S. Makonin, F. Popowich, L. Bartram, B. Gill, and I. V. Bajic. AMPds: A Public Dataset for Load Disaggregation and Eco-Feedback Research. In *Proceeding of the IEEE Conference on Electrical Power and Energy (EPEC)*, 2013.

[Mon13] A. Monacchi, D. Egarter, and W. Elmenreich. Integrating households into the smart grid. In *Proceedings of the Workshop on Modeling and Simulation of Cyber-Physical Energy Systems (MSCPES)*, 2013.

[Mon14a] A. Monacchi, D. Egarter, W. Elmenreich, S. D'Alessandro, and A. M. Tonello. GREEND: an energy consumption dataset of households in Italy and Austria. In *Proceedings of the IEEE International Conference on Smart Grid Communications (SmartGridComm)*, 2014.

[Mon14b] R. E. Monge and J. L. Crespo. Comparison of complexity measures for DNA sequence analysis. In *Proceedings of the International Conference on Bio-inspired Intelligence (IWOBI)*, 2014.

[Mon15] A. Monacchi, F. Versolatto, M. Herold, D. Egarter, A. M. Tonello, and W. Elmenreich. An Open Solution to Provide Personalized Feedback for Building Energy Management. *arXiv preprint arXiv:1505.01311*, 2015.

[Nee09] B. Neenan, J. Robinson, and R. Boisvert. Residential electricity use feedback: A research synthesis and economic framework. *Electric Power Research Institute*, 2009.

[Ngu13] T. A. Nguyen and M. Aiello. Energy intelligent buildings based on user activity: A survey. *Energy and buildings*, 56:244–257, 2013.

[Nor96] L. K. Norford and S. B. Leeb. Non-intrusive electrical load monitoring in commercial buildings based on steady-state and transient load-detection algorithms. *Energy and Buildings*, 24(1):51 – 64, 1996.

[Par12] O. Parson, S. Ghosh, M. Weal, and A. Rogers. Non-intrusive load monitoring using prior models of general appliance types. In *Proceedings of the Twenty-Sixth Conference on Artificial Intelligence (AAAI)*, 2012.

[Par14] O. Parson, S. Ghosh, M. Weal, and A. Rogers. An unsupervised training method for non-intrusive appliance load monitoring. *Artificial Intelligence*, 217:1–19, 2014.

[Pat07] S. N. Patel, T. Robertson, J. A. Kientz, M. S. Reynolds, and G. D. Abowd. *At the flick of a switch: Detecting and classifying unique electrical events on the residential power line*. Springer, 2007.

[Pat12] S. Pattem. Unsupervised disaggregation for non-intrusive load monitoring. In *Machine Learning and Applications (ICMLA), 2012 11th International Conference on*, volume 2, pages 515–520. IEEE, 2012.

[Pin91] S. M. Pincus. Approximate entropy as a measure of system complexity. *Proceedings of the National Academy of Sciences*, 88(6):2297–2301, 1991.

[Pöc15] M. Pöchacker, D. Egarter, and W. Elmenreich. Proficiency of Power Values for Load Disaggregation. *IEEE Transactions on Instrumentation and Measurement*, 2015.

[Rab89] L. R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. In *Proceedings of the IEEE*, pages 257–286, 1989.

[Rei12] A. Reinhardt, P. Baumann, D. Burgstahler, M. Hollick, H. Chonov, M. Werner, and R. Steinmetz. On the Accuracy of Appliance Identification Based on Distributed Load Metering Data. In *Proceedings of the 2nd IFIP Conference on Sustainable Internet and ICT for Sustainability (SustainIT)*, 2012.

[Rei15] A. Reinhardt, G. Konstantinou, D. Egarter, and D. Christin. Worried About Privacy? Let Your PV Converter Cover Your Electricity Consumption Fingerprints. In *Proceedings of the IEEE International Conference on Smart Grid Communications (SmartGridComm)*, 2015.

[Rez98] I. Rezek and S. Roberts. Stochastic complexity measures for physiological signal analysis. *IEEE Transactions on Biomedical Engineering*, 45(9):1186–1191, Sept 1998.

[Ric00] J. S. Richman and J. R. Moorman. Physiological time-series analysis using approximate entropy and sample entropy. *American Journal of Physiology - Heart and Circulatory Physiology*, 278(6):H2039–H2049, 2000.

[Rid13] A. Ridi, C. Gisler, and J. Hennebert. Automatic identification of electrical appliances using smart plugs. In *Proceedings of the 8th International Workshop on Systems, Signal Processing and their Applications (WoSSPA)*, 2013.

[Rid14a] A. Ridi, C. Gisler, and J. Hennebert. ACS-F2-A New Database of Appliance Consumption Analysis. In *Proceedings of the International Conference on Soft Computing and Pattern Recognition (SocPar 2014)*, 2014.

[Rid14b] A. Ridi, C. Gisler, and J. Hennebert. Appliance and state recognition using Hidden Markov Models. In *Proceedings of the International Conference on Data Science and Advanced Analytics (DSAA)*, 2014.

[Rid14c] A. Ridi and J. Hennebert. Hidden Markov Models for ILM appliance identification. *Procedia Computer Science*, 32:1010–1015, 2014.

[RO02] A. P. Rosso OA, MT Martin. Brain electrical activity analysis using wavelet based informational tools. *Physica A*, 2002.

[Ruz10] A. Ruzzelli, C. Nicolas, A. Schoofs, and G. O'Hare. Real-Time Recognition and Profiling of Appliances through a Single Electricity Sensor. In *Proceedings of the 7th International Conference on Sensor Mesh and Ad Hoc Communications and Networks (SECON)*, 2010.

[Sal75] H. M. Salkin and C. A. De Kluyver. The knapsack problem: a survey. *Naval Research Logistics Quarterly*, 22(1):127–144, 1975.

[Sch14] M. Schellander, T. Khatib, W. Elmenreich, and D. Egarter. Techno-economical assessment of grid-connected photovoltaic power systems productivity in summer season in Klagenfurt, Austria. In *Proceedings of the International Conference on Power and Energy (PECon)*, 2014.

[Sha08] S. Shaw, S. Leeb, L. Norford, and R. Cox. Nonintrusive Load Monitoring and Diagnostics in Power Systems. *IEEE Trans. Instrum. Meas.*, 57(7):1445–1454, July 2008.

[Sha12] H. Shao, M. Marwah, and N. Ramakrishnan. A temporal motif mining approach to unsupervised energy disaggregation. In *Proceedings of the 1st International Workshop on Non-Intrusive Load Monitoring, Pittsburgh, PA, USA*, volume 7, 2012.

[Sko12] F. Skopik. Security is not enough! on privacy challenges in smart grids. *Int. J. Smart Grid Clean Energy*, 1(1):7–14, 2012.

[Sri06] D. Srinivasan, W. S. Ng, and A. Liew. Neural-network-based signature recognition for harmonic source identification. *IEEE Trans. Power Del.*, 21(1):398–405, 2006.

[Suz08] K. Suzuki, S. Inagaki, T. Suzuki, H. Nakamura, and K. Ito. Non-intrusive appliance load monitoring based on integer programming. In *Proceedings of the International conference on Instrumentation, Control, Information Technology and System Integration (SICE)*, 2008.

[Uen06] T. Ueno, R. Inada, O. Saeki, and K. Tsuji. Effectiveness of an energy-consumption information system for residential buildings. *Applied Energy*, 83(8):868–883, 2006.

[Wan12] Z. Wang and G. Zheng. Residential Appliances Identification and Monitoring by a Nonintrusive Method. *Smart Grid, IEEE Transactions on*, 3(1):80–92, March 2012.

[Wei04] Y. Wei and L. Qiqiang. Survey on Particle Swarm Optimization Algorithm [J]. *Engineering Science*, 5(5):87–94, 2004.

[Won14] Y. F. Wong, T. Drummond, and Y. Rekercioglu. Real-time load disaggregation algorithm using particle-based distribution truncation with state occupancy model. *Electronics Letters*, 50(9):697–699, April 2014.

[Yan10a] X.-S. Yang. Firefly algorithm, Levy flights and global optimization. In *Research and Development in Intelligent Systems XXVI*, pages 209–218. Springer, 2010.

[Yan10b] X.-S. Yang. *Nature-inspired metaheuristic algorithms*. Luniver press, 2010.

[Yan12] W. Yang, N. Li, Y. Qi, W. Qardaji, S. McLaughlin, and P. McDaniel. Minimizing private data disclosures in the smart grid. In *Proceedings of the 2012 ACM conference on Computer and communications security*, pages 415–427. ACM, 2012.

[Zei11] M. Zeifman and K. Roth. Nonintrusive appliance load monitoring: Review and outlook. *IEEE Trans. Consum. Electron.*, 57(1):76 –84, february 2011.

[Zia11] T. Zia, D. Bruckner, and A. Zaidi. A hidden Markov model based procedure for identifying household electric loads. In *Proceedings of the Annual Conference on IEEE Industrial Electronics Society (IECON)*, 2011.

[Zoh12] A. Zoha, A. Gluhak, M. A. Imran, and S. Rajasegarar. Non-Intrusive Load Monitoring Approaches for Disaggregated Energy Sensing: A Survey. *Sensors*, 12(12):16838–16866, 2012.

[Zoh13] A. Zoha, A. Gluhak, M. Nati, and M. Imran. Low-power appliance monitoring using Factorial Hidden Markov Models. In *Proceedings of the IEEE Conference on Intelligent Sensors, Sensor Networks and Information Processing*, 2013.

List of Own Publications

Journal Publications

1. D. Egarter, V.P. Bhuvana and W. Elmenreich. PALDi: Online Load Disaggregation based on Particle Filtering. *IEEE Transactions on Instrumentations and Measurement*, 2014.

2. T. Khatib, A. Monacchi, W. Elmenreich, D. Egarter, S. DAlessandro and A. M. Tonello. European end-users level of energy consumption and current structural barriers for smart homes: A case study of residential sectors in Austria and Italy. *Energy Technology & Policy*, 2014

3. C. Klemenjak, D. Egarter and W. Elmenreich. YOMO - The Arduino based Smart Metering Board. *Computer Science Research and Development*, 2015.

4. M. Pöchacker, D. Egarter and W. Elmenreich. Proficiency of Power Values for Load Disaggregation, *IEEE Transactions on Instrumentations and Measurement*, 2015.

5. D. Egarter, A. Monacchi and W. Elmenreich. Integration of Legacy Appliances into the Home Energy Management System, *Journal of Ambient Intelligence and Humanized Computing*, 2015

Conference Publications

6. A. Reinhardt, G. Konstantinou, D. Egarter and D. Christin. Worried About Privacy? Let Your PV Converter Cover Your Electricity Consumption Fingerprints. *IEEE International Conference on Smart Grid Communication (SmartGridComm)*, 2015

7. M. Schellander, T. Khatib, W. Elmenreich and D. Egarter. Techno-economical Assessment of Grid-connected Photovoltaic Power Systems Productivity in Summer Season in Klagenfurt. *IEEE International Conference on Power and Energy (PECON)*, 2014

8. C. Klemenjak, D. Egarter and W. Elmenreich. YOMO - The Arduino based Smart Metering Board. *Energieinformatik*, 2014

9. G. Brandner, J. Klinglmayr, D. Egarter, U. Schilcher and C. Bettstetter. Precision of Pulse-Coupled Oscillator Synchronization on FPGA-Based Radios. *10th International ITG Conference on Systems, Communications and Coding*, 2015

10. A. Monacchi, D. Egarter, W. Elmenreich, S. DAlessandro and A. M. Tonello. GREEND: an energy consumption dataset of households in Italy and Austria. *IEEE International Conference on Smart Grid Communication (SmartGridComm)*, 2014.

11. D. Egarter, C.Prokop and W. Elmenreich. Load Hiding of Household's Power Demand. *IEEE International Conference on Smart Grid Communication (SmartGridComm)*, 2014.

12. D. Egarter, A. Sobe, and W. Elmenreich. Evolving Non-Intrusive Load Monitoring. *Proceedings of the European Conference on the Applications of Evolutionary Computation (at EvoStar conference)*, 2013.

Workshop Publications

13. D. Egarter and W. Elmenreich. Autonomous Load Disaggregation Approach based on Active Power Measurements. *IEEE Workshop on Pervasive Energy Services (PerEnergy)*, 2015

14. D. Egarter, V.P. Bhuvana and W. Elmenreich. Appliance State Estimation based on Particle Filtering. *5th ACM Workshop On Embedded Systems For Energy-Efficient Buildings (BuildSys 2013)* November 2013.

15. D. Egarter and W. Elmenreich. EvoNILM - Evolutionary Appliance Detection for Miscellaneous Household Appliances. *Proceedings of the Workshop on Green and Efficient Energy Applications of Genetic and Evolutionary Computation Workshop (at GECCO conference)* 2013.

16. A. Monacchi, D. Egarter, and W. Elmenreich. Integrating Households into the Smart Grid. *Workshop on Modeling and Simulation of Cyber-Physical Energy Systems (at E-Energy conference)*, 2013.

17. W. Elmenreich and D. Egarter. Design guidelines for smart appliances. *Proceedings of the 10th International Workshop on Intelligent Solutions in Embedded Systems*, 2012.

Book Chapter

18. D. Egarter, A. Monacchi, M. Pöchacker, K. Schweiger and B. Steinwender: Smart Grid: Visionen & Herausforderungen, published in Energie - Interdisziplinäre Perspektiven auf eine knappe Ressource (Profil Verlag; G. Getzinger, H. Gross)