INSTITUT POLYTECHNIQUE DE PARIS

NNT : 2020IPPAT007

Thèse de doctorat

On Solving the Non Intrusive Load Monitoring Problem in Large Buildings: Analyses, Simulations and Factorization Based Unsupervised Learning.

Thèse de doctorat de l'Institut Polytechnique de Paris
préparée à Télécom Paris

École doctorale n°626 Institut Polytechnique de Paris (ED IP Paris)
Spécialité de doctorat : Signal, Images, Automatique et Robotique

Thèse présentée et soutenue à Palaiseau, le 16 septembre 2020, par

SIMON HENRIET

Membres du jury :

Georges Hebrail Professeur et chercheur, EDF R&D	Président, Rapporteur
Christian Jutten Professeur, Université Grenoble Alpes (Gipsa-Lab)	Rapporteur
Aapo Hyvärinen Professeur, Université d'Helsinki	Examinateur
Lina Stankovic Maitre de conférence, Université de Strathclyde	Examinatrice
Mario Bergès Professeur, Université de Carnegie Mellon	Examinateur
Gaël Richard Professeur, Télécom Paris	Directeur de thèse
Umut Şimşekli Maitre de conférence, Télécom Paris	Co-directeur de thèse
Benoit Fuentes Chercheur, Smart Impulse	Co-directeur de thèse

Abstract

With the increasing awareness about the problem of climate change and the high level of energy consumption, a need for energy efficiency has emerged especially for electric power consumptions in buildings. To spur energy savings, industrials have been looking for measurement methods to monitor power consumptions. Appliance load monitoring has thus become an active research field. Monitoring and understanding the electrical consumption of appliances can also be useful for predictive maintenance, power quality analyses, demand forecasting or occupancy detection. Thirty years ago, a method called Non Intrusive Load Monitoring (NILM) has been introduced. It consists of estimating individual appliance energy consumptions from the measurement of the total consumption of the building. Its main advantage over traditional sub-metering methods is to use a single electric power meter at the main breaker of the building and then use a disaggregation algorithm to separate the contributions of each appliance.

The goal of this thesis is to address the algorithmic challenge offered by NILM. The NILM problem can be formulated as a source separation problem, where the sources are the individual electric consumptions and the mixed observation is simply the sum of individual consumptions. Its main difficulties are: (i) the standardization of the formulation, (ii) the ill-posedness of the problem, (iii) the lack of knowledge and (iv) the machine learning algorithm design. All our contributions follow from the principal objective that is to solve the NILM problem for huge systems such as commercial or industrial buildings using high frequency current and voltage measurements. However, houses and the specific equipment found inside these buildings are not excluded of the study. This thesis is split into two parts.

In the first part, we tackle the lack of knowledge and datasets for NILM in commercial buildings. First of all, the NILM community has mostly focused on both residential NILM application and using low frequency data provided by power meter installed by utility providers. To tackle the lack of knowledge on higher frequency data and on other kind of buildings such as commercial or industrial installations,

we propose a statistical analysis based on public and private datasets. Our study on the rank of current matrix conducted for individual devices will serve as the base of a new device taxonomy and to prior assumptions on the rest of this thesis. Secondly, we address the lack of datasets especially for commercial buildings by developing an algorithm for generating synthetic current data based on a modelization of the current flowing through an electrical device. To encourage research on commercial buildings we release a synthesized dataset called SHED that can be used to evaluate NILM algorithms.

In the second part, we deal with the NILM software challenges by exploring unsupervised source separation techniques. To overcome the unaddressed difficulties of processing high frequency current signals that are measured in large buildings, we propose a novel technique called Independent-Variation Matrix Factorization (IVMF), which expresses an observation matrix as the product of two matrices: the "signature" and the "activation". Motivated by the nature of the current signals, it uses a regularization term on the temporal variations of the activation matrix and a positivity constraint, and the columns of the signature matrix are constrained to lie in a specific set. To solve the resulting optimization problem, we rely on an alternating minimization strategy involving dual optimization and quasi-Newton algorithms. IVMF is the first proposed algorithm especially designed for high frequency NILM in huge buildings. We finally show that IVMF outperforms competing methods (Independent Component Analysis, Semi Non-negative Matrix Factorization) on NILM datasets.

Résumé

La prise de conscience des conséquences du réchauffement climatique a permis de lancer un mouvement de réduction de l'utilisation d'énergie. Sans pour autant stopper toute utilisation d'énergie, le faire de façon la plus efficace possible en réduisant le gaspillage apparaît comme une solution évidente. L'électricité utilisée dans les bâtiments représente une part importante de la consommation d'énergie et doit donc être utilisée de manière efficace. Pour cela, il est nécessaire de pouvoir mesurer et suivre la consommation électrique de chaque appareil au sein d'un bâtiment. Depuis 30 ans, une méthode de suivi des consommations électriques, Non Intrusive Load Monitoring (NILM), propose à partir d'un unique compteur mesurant la consommation totale du bâtiment, de déterminer la contribution de chaque appareil électrique. Cette méthode est basée sur un algorithme de désagrégation des consommations électriques et permet de s'affranchir de l'utilisation d'un compteur de mesure pour chaque appareil électrique du bâtiment.

Cette thèse aborde les problèmes algorithmiques que présente le NILM. De manière générale, la problématique est celle de la séparation de sources. Les différentes sources à estimer correspondent ici à la consommation électrique des différents appareils branchés sur un même réseau. La mesure réalisée, aussi appelée observation mélangée, correspond à la somme de toutes les consommations. Ainsi, les principales difficultés du NILM sont : (i) la standardisation de la formulation, (ii) le caractère mal-posé du problème (perte d'information), (iii) les connaissances insuffisantes sur les signaux et (iv) l'implémentation d'un algorithme d'apprentissage. L'objectif principale de cette thèse est de traiter le NILM dans le cadre des grands bâtiments (commerciaux, bureaux, industriels) en utilisant des mesures hautes fréquences du courant et de la tension. Cependant les maisons individuelles et leurs propres types d'appareils électriques ne sont pas exclus de cette étude. Cette thèse est structurée en deux grandes parties.

Dans une première partie nous abordons le problème du manque de connaissance des signaux de consommation électriques, à la fois ceux des grands bâtiments et ceux

des différents appareils utilisés. La littérature concernant le NILM est principalement orienté sur l'étude des mesures basses fréquences de consommations dans les maisons. Nous proposons ici une analyse statistique des mesures de consommations. Nos résultats nous permettent de proposer une nouvelle classification des appareils électriques en fonction de leur caractéristiques de courant et également de définir des hypothèses pour la résolution du problème de séparation des sources. Le manque de données de consommations disponibles est également un frein pour le développement du NILM. Pour répondre à cela nous développons un modèle génératif permettant de simuler des données hautes fréquences de courant électrique de bâtiments. A partir d'un nombre limité de données réelles nous réalisons des simulations de bâtiments que nous partageons dans la base de données SHED.

Dans une seconde partie, nous abordons le problème de la séparation de source. Grâce à nos résultats d'analyse et par manque de données, nous traitons ce problème à l'aide de techniques d'apprentissage non-supervisées. Pour proposons une nouvelle méthode appartenant à la famille des factorisations de matrice appelée Independent-Variation Matrix Factorization (IVMF), qui permet d'exprimer une matrice d'observation de courant comme le produit de deux matrices: les signatures et les activations. IVMF est le premier algorithme décrit pour le traitement du NILM dans le cadre de données hautes fréquences et de grands bâtiments. Enfin, nous montrons que IVMF atteint de meilleurs résultats pour le problème du NILM que des méthodes classiques de séparation de source comme l'Analyse en Composantes Indépendantes ou encore la Factorisation de Matrice Semi Non-négative.

Remerciements

En premier lieu, je voudrais remercier Smart Impulse et en particulier ses trois fondateurs Dorian, Henri et Charles. Merci de m'avoir donné l'opportunité de commencer cette thèse. Ce sont surement les discussions techniques et scientifiques en tout genre sur le canapé du Prine qui m'auront donné envie d'aller plus loin. Je réalise le chemin parcourru, j'espère que la suite sera encore plus belle.

J'aimerais chaleureusement remercier mes encadrants de thèse. Gaël, merci mille fois d'avoir été le capitaine du bateau qui m'a mené à bon port presque 4 ans après notre première rencontre. Ton analyse et ton calme pour traiter tous les problèmes m'ont sauvé plusieurs fois. Umut, thank you so much for having taught me how to write a paper and for having us speak in english in our meetings. Your good mood is always appreciated, either at a meeting room or in a bar. Benoit, en plus d'avoir été mon encadrant de thèse je dois aussi te remercier d'avoir été un collègue inspirant pendant 6 ans; par ta pédagogie et par ta créativité. Travailler à tes côtés m'a donné la motivation finale pour débuter cette aventure. Ton soutient et tes conseils m'auront permis de la finir.

I would like to thank to all the jury members whose comments made this manuscript a better one and whose possible questions made me prepare my defense contentiously.

J'aimerais ensuite remercier tous mes collègues de Télécom et de Smart Impulse avec qui j'ai passé du temps ces dernières années et qui m'ont aidé d'une façon ou d'une autre à réaliser ce travail. Antoine, on ne s'est pas beaucoup croisé mais ce travail est bati sur les bases que tu as posées. Sergio, merci d'avoir apporté ta pierre à cette édifice. Ghassene merci pour ta motivation et tes idées pour faire avancer le NILM. Un tout particulier merci à Pierre A., Tom, Pierre L., Thomas, Mathurin pour les nombreuses discussions et vos avis toujours éclairés, même dans l'obscurité de la Butte-aux-Cailles. J'espère que j'aurai reussi à vous faire comprendre ce que NILM voulait dire. Vous m'aurez montré l'excellence. Un grand merci également à toutes les personnes qui se sont succédées dans le labyrinthe de bureaux de Télécom

en commencant par le B412. A tous mes collègues de Smart Impulse, merci de faire vivre ce projet ambitieux et éco-responsable.

J'aimerais également remercier tous les enseignants que j'ai pu avoir depuis la primaire au secondaire puis dans l'enseignement supérieur à l'INSA, l'Université de Rouen, l'Université Paris 7 et finalement à Télécom Paris. Je mesure la chance que j'ai d'avoir eu un enseignement de qualité.

Je souhaiterais aussi vivement ne pas remercier la covid-19 pour avoir chamboulé la fin de ce travail.

J'aimerais remercier toute ma famille et tous mes amis de m'avoir soutenu, encouragé et supporté dans les bons comme dans les mauvais moments. Vous n'entendrez plus la sempiternelle: "je ne peux pas je dois finir ma thèse". A tous ceux qui partagent mon autre passion, pour les balles et le gazon: j'espère que vous me supporterez encore longtemps sur les terrains et dans les stades.

Enfin, Papa, merci de m'avoir montré que tout était faisable. Maman, merci de m'avoir toujours poussé à bien faire et à choisir le chemin de la qualité. Stéphane merci d'avoir toujours été un exemple pour moi en ouvrant la voie. Frédérique, merci d'avoir choisi Pissy-Poville. Le mot de la fin est pour toi Joana, qui m'accompagne depuis presque toujours. Ton dévouement pour la médecine, tes sacrifices et ton amour m'ont donné la force et le courage d'entreprendre cette magnifique et exigente aventure qu'est la thèse de doctorat. Sans toi je n'aurais pas osé.

Contents

List of Figures

List of Tables

Nomenclature

Abbreviations

AC Alternating Current

ALM Appliance Load Monitoring

ARMA AutoRegressive Moving Average

DFT Discrete Fourier Transform

FHMM Factorial Hidden Markov Model

HF High Frequency

ICA Independent Component Analysis

IVMF Independent-Variations Matrix Factorization

LED Light-Emitting Diode

LF Low Frequency

NILM Non Intrusive Load Monitoring

PFC Power Factor Correction

RMS Root Mean Square

SC Sparse Coding

SHED Synthetic High-frequency Energy Disaggregation dataset for commercial buildings

SNMF Semi Non-negative Matrix Factorization

SNR Signal to Noise Ratio

VSD Variable-Speed Drive

Electricity Notations

f The voltage frequency in Hertz (Hz)

$\mathbf{u}$ Voltage in Volt (V)

$\mathbf{i}$ Current in Ampere (A)

$\mathbf{p}$ Power in watt W or kilowatt (kW)

$\mathbf{E}$ Energy in kilowatt hour (kWh)

τ Time

t, T Index of a voltage period and total number of period

$\mathbf{P}$ Average real power

$\mathbf{Q}$ Average reactive power

n, N Sampling index within a period and total number of sampling point

$\mathbf{I}$ Current in matrix representation $\in \mathbb{R}^{N \times T}$

$\mathbf{S}$ Average apparent power

$\mathbf{U}$ Voltage in matrix representation $\in \mathbb{R}^{N \times T}$

c, $\mathcal{C}$ A category index and the set of all the categories

d, $\mathcal{D}$, $\mathcal{D}_c$ A device index, the set of all the devices and the set of devices in category c

$\Delta\mathbf{P}$ The power variations (also called derivatives)

$\mathbf{S}$, $\mathbf{s}$ A signature matrix and a column of the matrix called a signature

$\mathbf{A}$, $\mathbf{a}$ An activation matrix and a row of the matrix called an activation

General Notations

$[\![1, K]\!]$ The set of integers between 1 and K

$\mathcal{N}$, $\mathcal{B}$er The Normal and Bernoulli distributions

$P[\dots]$ The probability of

$\mathbb{R}^{N\times T}$ The set of $N \times T$ real valued matrices

$\mathbb{1}_{[cond]}$ The indicator function, equals 1 if *cond* is true else 0

$\{0,1\}^{K\times T}$ The set of $K \times T$ matrices with binary entries

$X^\top$, $x^\top$ The transpose matrix and a row vector.

$X^\dagger$ The Moore-Penrose pseudo inverse of matrix X

X^{-1} The inverse of an invertible matrix X

∇f The gradient of f

Chapter 1

Introduction

In this very first chapter, we will set the context of this thesis dissertation by introducing the concept of Non Intrusive Load Monitoring (NILM). This active field of research has received attention for almost 3 decades now. Let see what it is!

1.1 Context and preliminaries on electricity

To start with, we explore the reasons why NILM has emerged. Secondly we treat the electric foundation of this method.

1.1.1 Appliance Loads Monitoring

NILM is related to the notion of monitoring electric appliances consumptions. First of all, we review applications of monitoring the electric consumption of devices.

1.1.1.1 Applications and usage

Appliance Load Monitoring (ALM) is the first step towards **energy efficiency** in electric power systems. Measuring and understanding the different electric consumption in an electric network or building is essential to save energy. The first question one should ask oneself is: *which equipments consume the most and may I reduce their consumption?* The first part of the question may be answered by monitoring the consumption of every devices and then computing the share of the electric consumption across all of them. The second part of the question may also be addressed by device monitoring looking for a better operation of the equipment: stopping heaters during building vacancy is one example. This particular example may seem irrelevant for a person living in his own house due to the proximity with the devices, but for an

energy manager in charge of saving energy in large office buildings, knowing the operation of a particular equipment may be impossible without remotely monitoring it. These questions and their respective answers are called energy feedbacks. The importance and nature of the feedback needed to ensure energy savings in households have been extensively studied. A detailed review [Ehrhardt-Martinez et al., 2010] of more than 60 studies, reveals that direct feedbacks (provided near real-time in contrast with indirect ones given after consumptions) of disaggregated or contextual information can help achieving maximal savings.

Predictive maintenance or appliance's health monitoring is the task of preventing faults in devices such as induction engine in an industrial plant. Reliability studies have been conducted to establish the prospective life or the mean time between failures for electrical machines. To integrate individual information for a particular motor, people have started monitoring temperature, chemicals concentration, vibration and finally electric currents. Specialists show that high frequency spectral analysis can detect faults in rotating engines [Tavner et al., 2008]. Load monitoring is thus a promising tool for predictive maintenance.

Power quality in electrical networks is more and more of research interest. The quality of an alternating current network is defined by its ability to keep the voltage stable in terms of fundamental frequency, amplitude and low distortion from a sinusoidal waveform. With the increasing development of non-linear loads, voltage distortions, sags or swells in networks can affect the operation of other devices [Sivakumar et al., 2016]. Then, there is a need for reliable and accurate monitoring of electric power systems.

From a power producer point of view, **forecasting the power demand** is important to adjust production units [Hippert et al., 2001, Alfares and Nazeeruddin, 2002]. Accessing detailed information such as individual consumption via load monitoring can simplify the forecasting problem.

A final application of load monitoring is **occupancy detection** in buildings. It consists in estimating the presence or the number of persons in a building via the electrical monitoring of certain appliances (light bulbs, computers, etc). The main purposes of occupancy detection is efficient control of Heating Ventilation and Air-Conditioning systems or presence detection in homes [Hattori and Shinohara, 2017].

1.1.1.2 Intrusive versus Non-Intrusive

Intrusive load monitoring refers to the technique of installing an electric meter on every devices one needs to track. Although very accurate, this approach suffers from

two main limitations:

(i) the economic cost of installing one meter per electric device in a building containing ten to thousands of them,

(ii) the infrastructure complexity for centralizing all the measured data and maintaining the quality over a long period of study.

To address these issues, researchers and industrials have come up with the idea of estimating individual consumptions with only access to the main breaker measurements (i.e. the total consumption of the network). This method is interesting as only one meter is needed, the difficulty then lying in the estimation. Essentially this approach trades hardware complexity and high accuracy for software complexity and estimation. This task, named Non Intrusive Load Monitoring (NILM) has been introduced by Massachusetts Institute of Technology and Electronic Power Research Institute researchers G. Hart, E. Kern and F. Schweppe in a US patent [Hart et al., 1989]. NILM and its software task are the main focus of this dissertation.

1.1.2 Physical preliminaries of electrical networks

We now recall the properties of electric networks with a focus on Alternating Current (AC) and buildings network.

1.1.2.1 Electrical energy

We restrict our explanation to utility-scale electric power systems, it means the power systems that buildings are connected to. From one side, electrical energy is generated in power station by converting mechanical (hydro-electric dam), chemical (hydrocarbons) or nuclear energies. On the other side, electrical devices are converting electrical energy to another kind of energy (heat, light, motion). We say that they *consume* the electrical energy. Physicists have defined quantities to measure the amount of electrical energy transformed by electric devices. Electrical energy (denoted **E**) is supplied by generators to the end-users by the combination of electric current (**i**) and electric potential differences (**u**) via an electric circuit. The electric power (**p**) is then the rate, per unit time, at which electrical energy is transferred by the electric circuit. Let us denote the absolute time by τ, all these quantities are related

by the following equations:

$$\mathbf{p}(\tau) = \mathbf{u}(\tau)\,\mathbf{i}(\tau) \tag{1.1}$$

$$\mathbf{E}(\tau_0, \tau_1) = \int_{\tau_0}^{\tau_1} \mathbf{p}(\tau) d\tau \tag{1.2}$$

with τ_0 and τ_1 referring to the time window considered for the energy calculation. Note that, on most electric meters displaying energy, τ_0 is implicitly set to the installation time of the meter and τ_1 is the current time.

1.1.2.2 Alternating current power systems

In AC power systems, the electricity generation is done by using rotating electric generators (the first of this kind is the well known dynamo that uses electromagnetic induction). The generated electric potential is an alternating voltage, meaning that the voltage is alternately positive and negative. The AC power system is favored over a Direct Current (DC) system because of transmission efficiency. As AC voltage can be increased or decreased using transformers, it allows the transmission of energy at high voltage which reduces the losses due to heat. The schema of a classic electric system, given in Figure 1.1, involves a generating station, followed by high voltage transmission lines and finally transformers distributing electricity to individual customers. As shown in Figure 1.1, we will focus later on the electrical network inside secondary customers buildings.

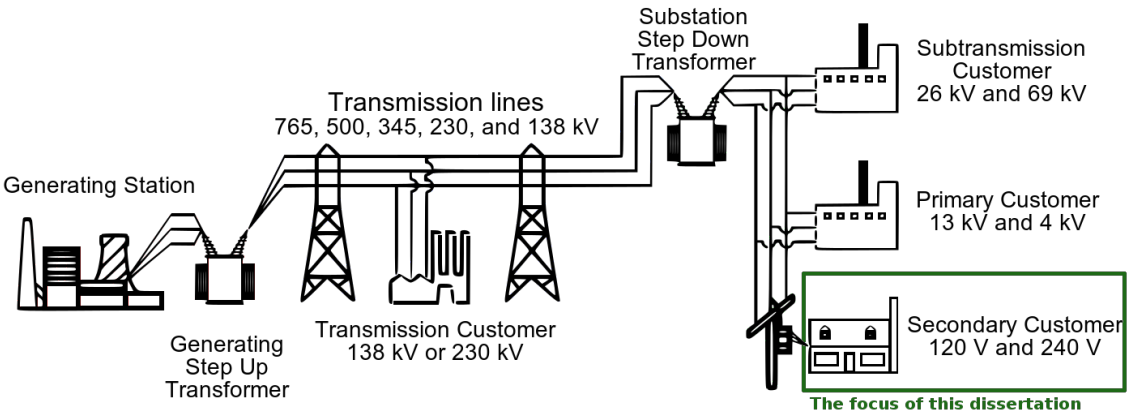


Figure 1.1: Basic structure of the US-Canada electric system (modified from [Muir and Lopatto, 2004]): generating station → step-up transformers → high voltage transmission lines → step down transformers → end-users.

We present here AC systems under perfect conditions. The first condition is the purely periodic sinusoidal voltage whose usual shape is a sine wave with constant frequency (f) and constant amplitude (V_{peak}). However they may vary from one country to another:

$$\mathbf{u}(\tau) = V_{peak} \sin(\omega\tau) \tag{1.3}$$

with $\omega = 2\pi f$. The most used frequencies are $f = 50$ or 60 Hz.

For characterizing electric systems in AC mode, one prefers using average quantities over one or several periods of the voltage. For instance, the amplitude of an AC voltage is quantified with the Root Mean Square (RMS) value (V_{rms}):

$$V_{rms}(t) = \sqrt{\frac{1}{\mathcal{T}} \int_t^{t+\mathcal{T}} \mathbf{u}^2(\tau) d\tau} \tag{1.4}$$

with t the index of period and $\mathcal{T} = 1/f$ is the period in seconds. Notice that, under perfect conditions, $V_{peak} = \sqrt{2} V_{rms}$. The most commonly found voltage (V_{rms}) values are 120/220/230/240 V. RMS current (I_{rms}) is defined in the same fashion.

Regarding the power consumptions, the classic quantity is the average power (over a period), also called real power:

$$\mathbf{P}(t) = \frac{1}{\mathcal{T}} \int_t^{t+\mathcal{T}} \mathbf{p}(\tau) d\tau = \frac{1}{\mathcal{T}} \int_t^{t+\mathcal{T}} \mathbf{u}(\tau)\, \mathbf{i}(\tau) d\tau \tag{1.5}$$

Electrical engineers have furthermore introduced two complementary quantities called apparent ($\mathbf{S}$) and reactive ($\mathbf{Q}$) powers, mathematically defined by:

$$\mathbf{S}(t) = V_{rms} I_{rms} \tag{1.6}$$

$$\mathbf{S}^2(t) = \mathbf{P}^2(t) + \mathbf{Q}^2(t) \tag{1.7}$$

Apparent power is used to design the size of conductors and transformers. Reactive power is introduced to quantify the *amount* of instantaneous power ($\mathbf{p}$) that *disappears* when integrated over a period. It can be interpreted as some kind of power *stored* in the electrical network and not *consumed*. Reactive power appears in electrical network with the introduction of reactive loads such as capacitors and inductors (these devices are studied in Chapter 2). It is a quantity of interest for power suppliers because even if reactive power is not consumed by end-users, it is still provided through the electrical network.

A final characteristic of the power transmission and distribution is the use of multiple wires or lines (as shown on Figure 1.1). Most modern power station generate three AC voltages, called phase conductors (or just phases), with same frequency and V_{rms} values relative to the same reference, called neutral (or ground depending on the earthing system). The only difference between phases is a time lag in the electric potentials. There is a third of period time lag between each phase conductors. Therefore, the power delivered to an end-user come as 4 wires: 3 phase conductors and the neutral. In perfectly balanced condition:

$$\mathbf{u}^1(\tau) = \sqrt{2}V_{rms}\sin(\omega\tau) \tag{1.8}$$

$$\mathbf{u}^2(\tau) = \sqrt{2}V_{rms}\sin(\omega\tau - 2\pi/3) \tag{1.9}$$

$$\mathbf{u}^3(\tau) = \sqrt{2}V_{rms}\sin(\omega\tau + 2\pi/3) \tag{1.10}$$

where $\mathbf{u}^i(\tau)$ denote the potential difference between the i^{th} phase conductor and the neutral wire. Figure 1.2 shows the phase shift between phase line voltages.

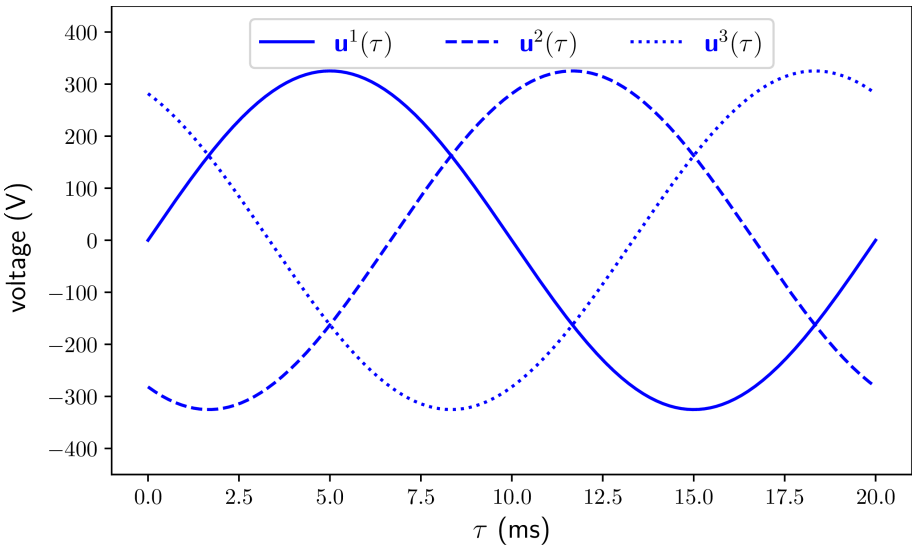


Figure 1.2: Three-phase voltage waveforms.

The strength of this system is that if the electric equipment is connected to two phase conductors instead of to one phase and the neutral, the voltage is multiplied by $\sqrt{3}$ without affecting the voltage frequency or the voltage sinusoidal shape. Three-phase electric power is also particularly efficient for rotating motors. For more details on electric power systems, one can refer to [Grainger et al., 2003].

Notation and units are summarized in Table 1.1.

Table 1.1: Physical quantities and units.

Quantity name	Symbol	Unit	Abbreviation
current	**i**	ampere	A
voltage	**u**	volt	V
instant. power	**p**	watt	W
real power	**P**	watt	W
reactive power	**Q**	volt-ampere reactive	var
apparent power	**S**	volt-ampere	VA
energy	**E**	kilowatt hour	kWh

1.1.2.3 Building's electrical network

It is worth mentioning that electric power is often supplied with only one phase and the neutral wire to small electric network such as individual houses. Unlike residential homes, large buildings are most of the time provided with three-phase electric power. Figure 1.3 illustrates the three-phase electric circuit of a building. In a building, the main breaker also referred as electric panel or distribution board is a component of an electricity supply system that divides an electrical power feed into subsidiary circuits. The utility electric meter is traditionally placed just before the main breaker for billing purposes. All the subsidiary circuits are then parallel circuits only connected at the main breaker. Parallel circuits are used so that a failure on one circuit does not affect the others. It also decreases the voltage drops of series circuits.

1.2 Non Intrusive Load Monitoring

Non Intrusive Load Monitoring (also called *Non Intrusive Appliance Load Monitoring* or *Energy Disaggregation*) is made of two inherent tasks:

(i) the Hardware task: designing a device capable of **sensing electric quantities** in a circuit and capable of **communicating** its measurements to a centralized system,

(ii) the Software task: designing an algorithm capable of **estimating** one or more individual power consumptions from the measurements.

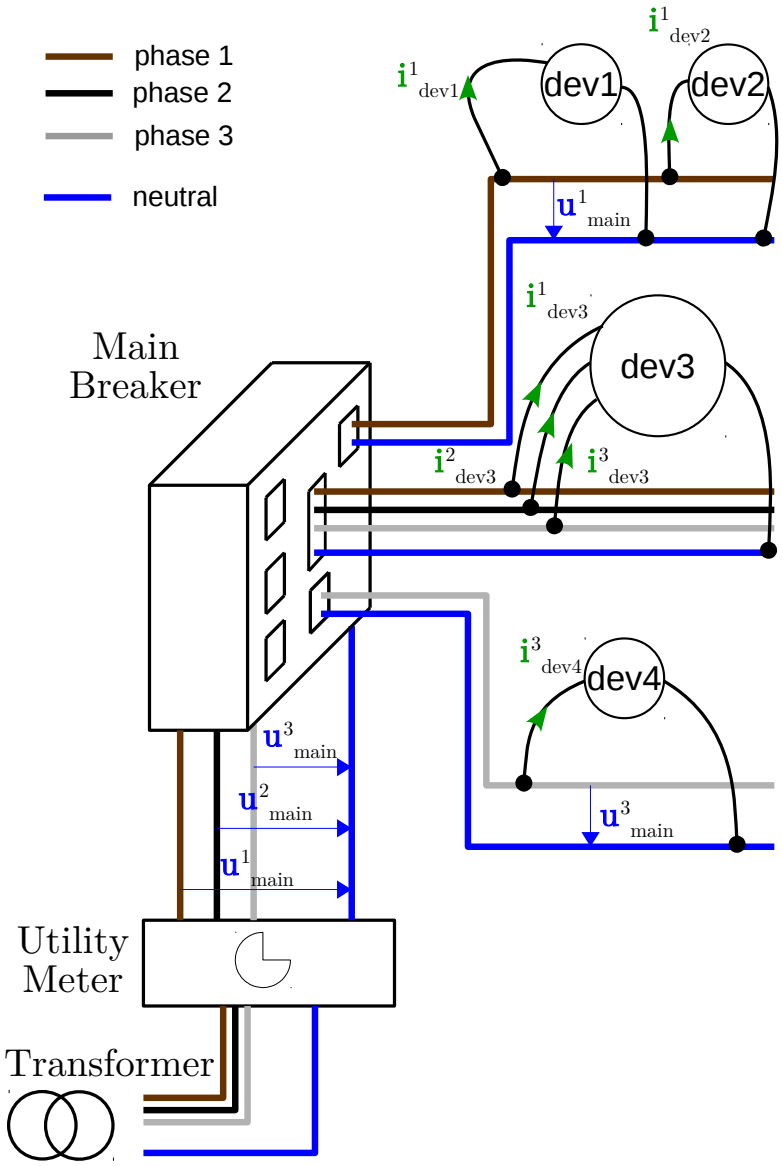


Figure 1.3: Schema of the the electric circuit of a building. The electric power is provided by the nearest transformer using 4 wires: 3 phase lines and one neutral (a ground wire may also be provided). At the entrance of the building, one may find the power meter furnished by the utility provider and then the main circuit breaker dispatching electric power to sub-circuit. The devices are connected in parallel and plugged to a single phase (device 1, 2 and 4) or to the three phases (device 3).

1.2.1 Hardware task

This section intends to introduce the hardware task of NILM by defining the physics of electrical circuits in common buildings and by reviewing different hardware systems available. The role of a hardware device, called meter, is to measure one or several electrical quantities amongst the ones defined in Table 1.1. A secondary role of the hardware device may also be to embed the NILM algorithm even if it is usually run on a remote server. The main features of a meter are the measured quantity, the processing, the sampling frequency and the measurements errors.

The hardware and software tasks are obviously closely linked. At the beginning of NILM history, software solutions have adapted to the kind of data provided by available general purpose hardwares. With the increasing interest in NILM, hardware devices have started to incorporate specifications needed to overcome some of the software limitations.

Power providers have installed electric meter on end-users circuit to quantify the amount of energy consumed in order to charge fees for it. For this application, the very first meters, called revenue meters, were simple discrete analog transducers converting AC voltage and current to an energy index. [Hart et al., 1989] are the first to design a NILM specific measurement apparatus . It sampled at a high frequency ($\approx$ 3kHz) voltage and current before digitizing it in order to compute interesting quantities (RMS voltage and current, real and reactive power). Even if the sampling capability of the device is at several kHz, the final output frequency was at 1 Hz.

Due to the absence of international norms, a multitude of hardware devices and their own data format have been developed over the last three decades (type of sensor, processing used to calculate quantities and output data formats). Notice that the International Electrotechnical Commission (IEC) has initiated the standardization of equipments for the measuring and the monitoring of steady state and dynamic quantities in power distribution systems (TC 85 / WG 20). Nevertheless, one can yet identify two categories of hardware/data format:

(i) Low Frequency (LF) data,

(ii) High Frequency (HF) data.

HF hardware or data provide information about electrical quantities at a higher sampling rate than the voltage frequency. This includes voltage (**u**), current (**i**), instantaneous power (**p**). In opposition, LF measurements represent information at a lower sampling rate than the voltage frequency. It may be an aggregation (for

instance integration) of HF data. It includes real (**P**) or reactive (**Q**) power, RMS voltage (V_{rms}) or current (I_{rms}) and energy (**E**).

With the increasing deployment of smart meters by energy providers, LF data have been made available at a high scale. The advantage of a LF hardware is its moderate cost and its lower need for data communication.

Conversely, HF data is only available nowadays from custom hardwares that need to be installed on top of the energy meter. While this category of hardware is more expensive and needs more data communication and storage, it provides also more accurate and richer information.

1.2.2 Software task

Thirty years ago, Ed Kern suggested a very simple definition of the NILM problem as *"a [program] that could monitor loads by identifying a signature at metering panel level"* (related in [Sultanem, 1991]). During three decades, researchers and industrials have formulated NILM software problems according to the electric data available, the desired quantity to be monitored and the application it was used for. As a consequence of the multitude of choices for each item, a lot of different formulations can be found in the literature.

1.2.2.1 Conservation of energy

Let us start our explanation with the fundamental concept behind all the NILM formulations: the conservation of energy. Established by Kirchhoff in 1845, the current law states that: the algebraic sum of currents in a network of conductors meeting at a point is zero:

$$\sum_k \mathbf{i}_k(\tau) = 0, \tag{1.11}$$

where k is the index of conductors. In other words the Kirchhoff's current law says that for any node or junction in an electrical circuit, the sum of currents flowing into that node is equal to the sum of currents flowing out of that node. In electrical circuits for modern buildings, all the devices are plugged in parallel (see Figure 1.3 for an explicative schema).

The main breaker is a node where the Kirchhoff's current law applies. This results in the NILM equation on current:

$$\mathbf{i}_{main}(\tau) = \sum_{d \in \mathcal{D}} \mathbf{i}_d(\tau), \tag{1.12}$$

where $\mathbf{i}_{main}(\tau)$ is the current at the main breaker, d is the index of an electric device and $\mathcal{D}$ is the set of all the device in the electric network.

Using Equation (1.12) and the fact that all the devices share the same voltage (potential at the main breaker - potential of the neutral wire, i.e. $\mathbf{u}_{main}(\tau) = \mathbf{u}_d(\tau)$), we can establish the NILM equation on power:

$$\mathbf{p}_{main}(\tau) = \sum_{d \in \mathcal{D}} \mathbf{p}_d(\tau). \tag{1.13}$$

One can notice that due to the linearity of their calculation, the same equation stands for real ($\mathbf{P}$) and for reactive ($\mathbf{Q}$) power and for energy ($\mathbf{E}$). On the contrary, apparent power ($\mathbf{S}$) is not linear and there is thus no such conservation equations.

1.2.2.2 Formulations of the problem

In this section, we first enumerate four examples of formulation in the literature.

The most commonly used formulation of the NILM software problem is following the lead of Hart's definition in [Hart, 1992]. It may be expressed as:

Example 1. *From the real power measurements acquired at the breaker panel of a house at a 1 Hz sampling rate:*

$$\mathbf{P}_{main}(t), \tag{1.14}$$

estimate, the real power consumptions of the bigger equipments ($\mathcal{D}_{big}$) in the house:

$$\mathbf{P}_d(t) \quad \forall\, d \in \mathcal{D}_{big}, \tag{1.15}$$

Figure 1.4 illustrates this example by showing the total power consumption and the disaggregated loads during one day for a house.

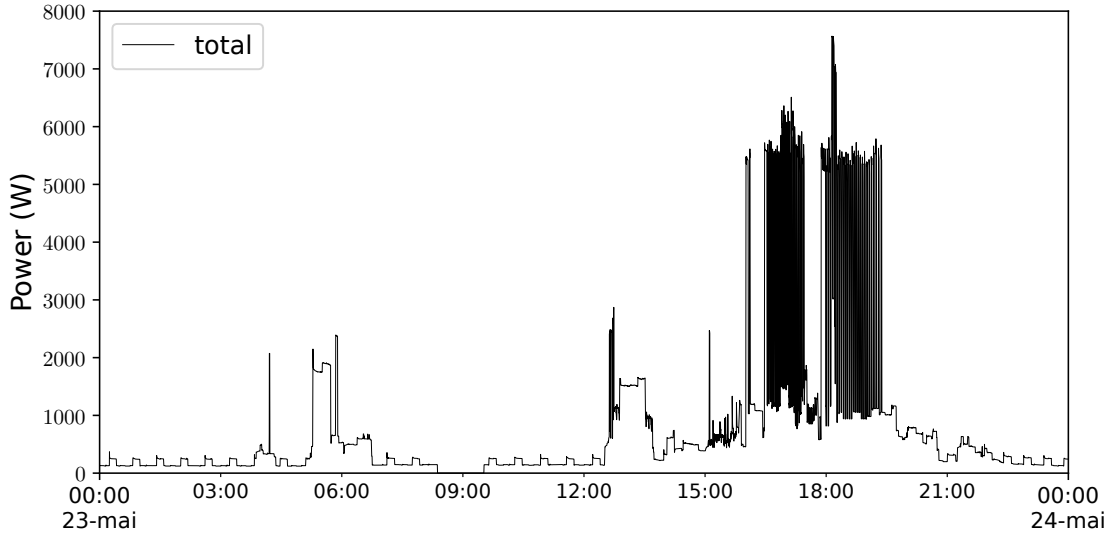
Later on, [Lee et al., 2005] defined a NILM problem in a commercial building.

Example 2. *From a transformation of current and voltage measurements, called power harmonics, acquired at the breaker panel of a commercial building:*

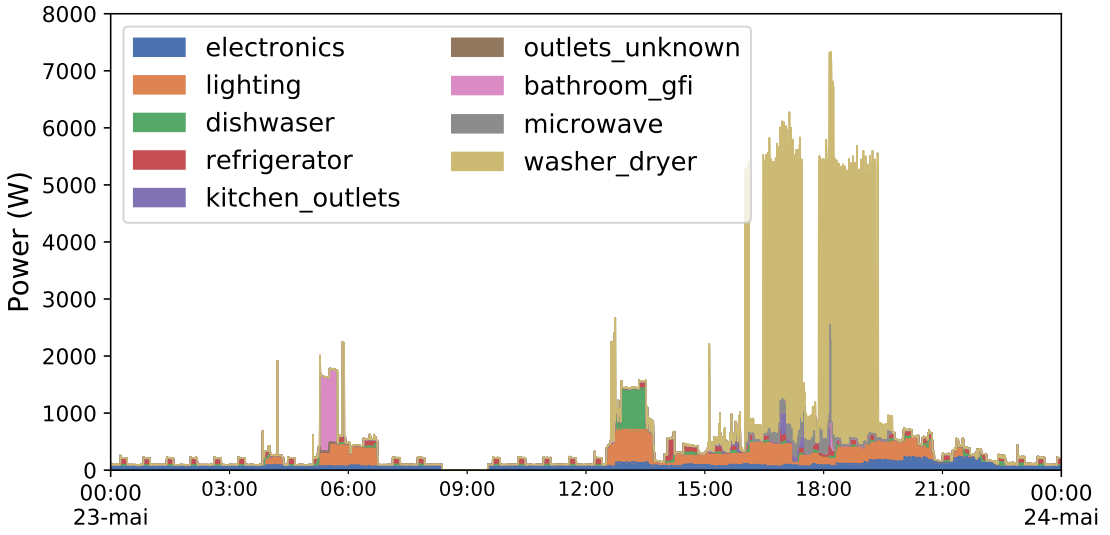
$$\mathbf{P}^{(k)}_{main}(t) = \frac{1}{\mathcal{T}} \int_t^{t+\mathcal{T}} \mathbf{i}(\tau) V_{peak} \sin(\omega k \tau) d\tau, \tag{1.16}$$

$$\mathbf{Q}^{(k)}_{main}(t) = \frac{1}{\mathcal{T}} \int_t^{t+\mathcal{T}} \mathbf{i}(\tau) V_{peak} \cos(\omega k \tau) d\tau, \tag{1.17}$$

and noticing that $\mathbf{P}^{(1)}_{main}(t) = \mathbf{P}_{main}(t)$, estimate the real power consumptions of the



(a) Total consumption



(b) Disaggregated consumption

Figure 1.4: Total and disaggregated consumption for the REDD dataset [Kolter and Johnson, 2011], house 3 on the 23rd May, 2011. Among labeled devices such as dish washer, we can notice unknown or circuit level label (bathroom gfi). The lighting label corresponds to a group of several light bulbs. A loss of data can be observed around 9 am.

Variable-Speed Drive (VSD) in the building:

$$\mathbf{P}_{vsd}(t), \tag{1.18}$$

such that:

$$\mathbf{P}_{main}(t) \geq \mathbf{P}_{vsd}(t). \tag{1.19}$$

Sometimes found as a subproblem [Kelly and Knottenbelt, 2015a] or as the main application [Tabatabaei et al., 2016], the NILM problem can be formulated as the problem of finding which devices are active at each time step.

Example 3. *From the real power measurements acquired at the breaker panel of a house:*

$$\mathbf{P}_{main}(t), \tag{1.20}$$

estimate which equipment of a list ($\mathcal{D}$) are active in the house:

$$\mathbb{1}_{[\mathbf{P}_d(t)>0]} \quad \forall\, d \in \mathcal{D}. \tag{1.21}$$

As a final example, we would like to stress on a formulation where high frequency current and voltage measurements are used to estimate real power consumption in houses [Lange and Bergés, 2016].

Example 4. *From the current and voltage measurements acquired at the breaker panel of a house at a 12 kHz sampling rate:*

$$\mathbf{i}_{main}(\tau), \mathbf{u}_{main}(\tau), \tag{1.22}$$

estimate the current waveforms and the real power consumptions of all the equipments ($\mathcal{D}$) in the house:

$$\mathbf{i}_d(\tau), \tag{1.23}$$

$$\mathbf{P}_d(t) = \frac{1}{\mathcal{T}} \int_t^{t+\mathcal{T}} \mathbf{u}_{main}(\tau)\, \mathbf{i}_d(\tau) d\tau \quad \forall\, d \in \mathcal{D}, \tag{1.24}$$

such that:

$$\mathbf{i}_{main}(\tau) = \sum_{d \in \mathcal{D}} \mathbf{i}_d(\tau). \tag{1.25}$$

To conclude on the NILM software problem, it can be seen as a **source separation problem**. In such a problem one intends to separate unknown sources

from observed mixed signals. In our case the unknown sources are the individual consumptions (either power or current signals) and the mixed observation is the total consumption of the system (either power or current). Notice that for NILM, the mixing process is simply the sum of the sources. The source separation is said *single channel* because we dispose of only one observation signal (corresponding to one mixing process).

In its common form, the source separation problem is referred to as *blind* if it is addressed without information (on the number or the type of the sources for instance). In this scenario the NILM software problem can be traditionally split into two subtasks: (i) the disaggregation and (ii) the classification steps. The disaggregation step aims at separating the total consumption into sub-components while the classification step intends to identify a sub-component to an electric device. However, the disaggregation and classification steps can also be merged into a single task.

1.3 Challenges

He have just presented what is the NILM problem and we can now enter into the details of the challenges that arise from it. We also classify the challenges as *hardware* and *software*.

1.3.1 Hardware challenges

Data compression. An important task in the hardware design is the data compression. Indeed, the amount of collected data can explode very quickly especially for HF hardware. There is no standard compression methods for electrical measurements, thus many different procedures can be found in the commercialized hardwares or in public datasets. In this situation the most encountered techniques are lossy compression: decimating (keeping only 1 sample every x sample) or averaging. Audio compression methods (lossless or not) can often be found in the public NILM datasets but a NILM specific method is still to be designed. For a complete review of the compression methods tried in the NILM context, refer to [Kriechbaumer et al., 2019].

Cost versus resolution. The cost of hardware meters is still a problem, especially for high resolution and high frequency meters. Indeed, NILM solutions are mostly implemented using low frequency data from already installed utility meters as

installing new metering devices is expensive. Then, designing low cost and high precision (resolution and sampling frequency) hardwares is a challenge to provide NILM algorithms with better data.

Even if these hardware challenges are interesting problems, it is beyond the scope of this thesis.

Data collection. Another challenge that is at the intersection of hardware and software is the dataset collection. The first public dataset for NILM has been released in 2011, it is called REDD [Kolter and Johnson, 2011]. The authors stressed the fact that access to massive public datasets has considerably spurred the research in other domains (the Wall Street Journal corpus [Marcus et al., 1993] in natural language processing; MNIST [LeCun et al., 1998] in image recognition). NILM data can be broken down into two categories: (i) aggregated (or total) measurements at the main breaker level and (ii) individual equipment measurements. When the individual measurements come from the same building and at the same period as the aggregated measurement we call it a sub-metering information or a ground truth. Even if a dozen of NILM datasets have been released in the last decade, the community still suffers from a lack of data. First, the quality of the dataset is still an issue since for most of them either the sampling rate is very low or the sub-metering is incomplete. Furthermore, datasets are acquired at different sampling rates due to the lack of standards in the NILM measurement procedures. Secondly, the quantity of data is very restricted. Almost all the publicly available datasets deal with household data or equipments usually found in houses. For most of the datasets the measurements are acquired at a limited number of houses (from one to tens). However we can notice the release of an important dataset called Dataport of more than 1200 houses sub-metered in the United States of America but the sampling rate is low (1 second to 1 minute).

1.3.2 Software challenges

Standardization. As introduced in Sections 1.1.1.1 and 1.2.2, the diversity of NILM applications and types of data induce a multitude of problem formulations. This diversity makes it difficult for researchers from general domains such as machine learning to tackle the NILM problem.

Ill-posed problem. All the examples presented in Section 1.2.2 are considered as ill-posed problems since it is obvious that an infinity of solutions exists to these

problems. This is due to the fact that the current at the main breaker is the sum of all the currents coming from the devices. This summation creates a loss of information and because of this, researchers have struggled to find good assumptions and hypotheses to reduce the number of possible solutions. This challenge can eventually be split into 2 sub-problems: (i) advancing the knowledge on electrical devices consumptions and on buildings behavior, (ii) developing machine learning algorithms capable of taking advantage of the data and prior assumptions.

Knowledge discovery. The literature has mainly focused on analyzing residential buildings consumptions using low frequency power readings. It has implied a lack of knowledge on larger systems such as commercial (office buildings, schools campus, shopping malls, etc) or industrial buildings. The consequence of this focus is that methods specifically designed for residential buildings perform badly when applied to other kinds of buildings. Furthermore, high frequency current and voltage measurements have not received a lot of attention due to acquisition difficulties even if it is widely acknowledged that rising the sampling frequency should improve NILM performances.

Machine learning algorithms. Choosing the algorithmic approach for solving the NILM software problem is a real challenge. For more than 30 years, researchers have investigated supervised and unsupervised learning techniques. The main groups of methods are chronologically: (i) pattern recognition (or event detection), (ii) Markovian models, (iii) matrix factorization (or dictionary learning) and finally (iv) deep learning. The evaluation of a NILM algorithm shall take into account the estimation precision but also its ability to process large amount of data.

1.4 Contributions

The goal of this thesis is to address the challenges offered by the NILM problem. All the contributions follow from the principal objective that is to solve the NILM problem for huge systems such as commercial or industrial buildings using high frequency current and voltage measurements. However, houses and the specific equipment we can find inside these buildings are not excluded of the study. This thesis is articulated around three main contributions.

Data Analyses and New Assumptions The main purpose of this work is to answer two questions:

(i) What are the differences between residential and commercial buildings ?

(ii) What are the common and useful characteristics of electrical devices ?

The first question is addressed by gathering aggregated measurements (acquired at the main breaker level) of residential and commercial buildings. We used both public and private datasets with low frequency (power measurement at a 30 second sampling rate) and high frequency (current waveforms at least at 10 kHz) measurements. In this first analysis we show important differences between residential and commercial buildings for the seasonality of power, the distribution of power derivatives (or power variations) and finally in the spectral content of current waveforms. The second question is tackled using a data analysis of high frequency measurements of individual devices from both public and private datasets. We use the concept of semi-nonnegative rank of a matrix. This makes it possible to introduce a low rank assumption for individual devices current measurements. This analysis is also used to propose a new device taxonomy for NILM applications.

SHED: a synthetic dataset We have developed a building model for high frequency current waveforms and a synthetic data generation algorithm that is able to learn parameters on real measurements and then use it to produce new realistic data. In the light of the results obtained from our statistical analyses and by making use of both the publicly available datasets and a private one, we conduct various experiments for evaluating the quality of our new device model and building simulations. To finally foster the NILM research for commercial buildings, we release a synthetic dataset, called SHED[1], that is generated by our algorithm.

Unsupervised learning approaches for NILM Finally, we cope with the NILM software problem by exploring unsupervised matrix factorization techniques applied to matrix of current waveforms. In a first part we explore existing general purpose factorization techniques and explain their limitations to solve the NILM software problem. In a second part, we develop a matrix factorization technique, called Independent-Variation Matrix Factorization (IVMF), that aims at addressing NILM in commercial buildings. The novelty of the method resides in the physically-inspired constraints and the regularization applied to the signature and activation matrices:

[1] https://nilm.telecom-paristech.fr/shed/

(i) we set linear and quadratic inequalities constraints on the signature columns; (ii) we add a positivity constraint on the activations and a regularization over the time variation of the activation. While the proposed optimization method accurately captures the underlying physical behavior of current signals, the proposed regularization function and the constraints make the problem highly non-trivial. To solve the resulting constrained non-convex optimization problem, we develop an alternating minimization strategy involving dual optimization, a quasi-Newton and an accelerated proximal gradient descent algorithms. We finally investigate the behavior and performance of our algorithm compared to two related methods (Independent Component Analysis (ICA) and Semi Nonnegative Matrix Factorization (SNMF)) on a synthetic source separation problem and on a realistic NILM application for commercial buildings. We show that IVMF is particularly adapted to recover positive sources that exhibit a strong temporal dependency and sources whose variations are independent from each other. To the best of our knowledge this constitutes the first attempt to solve the NILM problem for commercial buildings using current measurements factorization.

1.5 Organization of the document

The document is organized into two parts with a first part on analyzing the problem and a second where we propose a new solution to the NILM problem.

Part I aims at analyzing data in order to find insights for simulations and for developing NILM algorithms. This part is then made of 3 chapters:

In Chapter 2, we start with remainders of the theoretical field of network analysis from an electrical engineering perspective. Although this field has extended records, it is not really used in the NILM community. We then review the data analyses for residential buildings and enumerate the few studies on commercial buildings. We finally review simulation approaches in the NILM literature.

In Chapter 3, we use two private datasets from commercial buildings on top of all the public datasets available to present our own data analyses. Firstly, we analyze aggregated data from both commercial and residential buildings and present discriminative metrics. Secondly, we study individual device consumptions using matrix analyses techniques. This particular analysis be concluded by a new taxonomy of the individual equipments present in buildings and a fundamental property for future NILM algorithms: the low-rank assumption.

In Chapter 4, we develop a physically-inspired data model that enable us to reproduce and simulate the behavior of the electrical network of a building in a bottom-up procedure. We finally release a new NILM dataset called SHED, made of simulations of high frequency current data of 8 buildings along with the power consumptions of the individual devices constituting the buildings.

Part II is dedicated to the task of solving the NILM software source separation problem for huge systems such as commercial buildings. It is divided into 4 chapters:

In Chapter 5, we review the development history of NILM algorithms: (i) Pattern Recognition, (ii) Markovian models and (iii) Matrix Factorization. We see that, on top of the choice of the mathematical technique, NILM solutions differ from each other due to the type of data used (as in put and/or output) and to the learning strategy (supervised or unsupervised learning).

In Chapter 6, we introduce our framework of unsupervised learning technique suing high frequency current and voltage data. We first propose a generic formulation of the NILM software problem and instantiate the specific problem we want to solve. Based on our low rank assumption, we set our problem as a Matrix Factorization problem. Then, we detail the limitations for the problem of NILM of existing general purpose techniques (Semi Nonnegative Matrix Factorization (SNMF), Independent Component Analysis (ICA) and Sparse Coding).

In Chapter 7, to overcome the unaddressed difficulties of processing high frequency current signals, we propose a novel technique called Independent-Variation Matrix Factorization (IVMF), which expresses an observation matrix as the product of two matrices: the *signature* and the *activation*. Motivated by the nature of the current signals, it uses a regularization term on the temporal variations of the activation matrix and a positivity constraint. The columns of the signature matrix are constrained to lie in a specific set.

Finally, in Chapter 8, we use IVMF to solve the NILM problem on three public datasets: 8 commercial buildings (SHED) and 2 residential houses (REDD and BLUED). We show that IVMF outperforms competing methods such as SNMF and ICA on the commercial buildings. Although our method has been designed for commercial buildings, the qualitative results on residential datasets suggest that it can also perform well on that kind of buildings.

1.6 Publications

The previously presented contributions resulted in the following communications and publications:

(i) Simon Henriet, Umut Şimşekli, Benoit Fuentes, Gaël Richard. Energy Disaggregation for Commercial Buildings: A Statistical Analysis. In *Proceedings of the 4th International Workshop on Non-Intrusive Load Monitoring, March 7-8, 2018, Austin, USA*. (**best poster award**)

(ii) Simon Henriet, Umut Şimşekli, Benoit Fuentes, Gaël Richard. Synthetic dataset generation for non-intrusive load monitoring in commercial buildings. In *Proceedings of the 4th ACM International Conference on Systems for Energy-Efficient Built Environments (BuildSys), November 8-9, 2017, Delft, Netherlands.*

(iii) Simon Henriet, Umut Şimşekli, Benoit Fuentes, Gaël Richard. A Generative Model for Non-Intrusive Load Monitoring in Commercial Buildings. In *Energy and Buildings, Elsevier, Volume 177, 15 October 2018, Pages 268-278.*

(iv) Simon Henriet, Umut Şimşekli, Sergio Dos Santos, Benoit Fuentes, Gaël Richard. Independent-Variation Matrix Factorization with Application to Energy Disaggregation. In *IEEE Signal Processing Letters, Volume 26, Issue 11 , November 2019.*

Part I

On Analyzing and Simulating

Motivations In this part, we focus on analyzing and simulating electrical measurements, either from entire buildings (often called aggregated data) or from individual devices. On the first hand, this part is of particular interest for understanding the problem of Non Intrusive Load Monitoring (NILM). The analysis of electrical devices and electrical network is a broader endeavour than just NILM and has been extensively studied. However, such analyses have not been used that much in the NILM community. Another issue with the existing studies in NILM is the focus on residential buildings and thus on the type of equipments herein. There is also a lack of knowledge on high frequency data from a NILM perspective mainly due to a lack of available data. On the other hand, the biggest obstacle to the development of NILM algorithm is the lack of data to learn algorithms. Even if unsupervised algorithms require less data to be developed, they still need data of good quality for evaluation purposes. Therefore, the main goals of the following chapters are:

(i) to *understand* the differences between small and huge buildings,

(ii) to *find* interesting properties of individual appliances consumptions,

(iii) to *set prior information* and assumptions for future unsupervised NILM algorithms,

(iv) to *generate* synthetic datasets in order to evaluate NILM algorithms.

Organization This part is organized in 3 chapters:

In Chapter 2, we start with a recall of the theoretical field of network analysis from an electrical engineering perspective. Although this field has extended records, it is not really used in the NILM community. We then review the data analyses for residential buildings and enumerate the few studies on commercial buildings.

In Chapter 3, we use two private datasets from commercial buildings on top of all the public datasets available to present our own data analyses. Firstly, we analyze aggregated data from both commercial and residential buildings and present discriminative metrics. Secondly, we study individual device consumptions using matrix analyses techniques. This particular analysis be concluded by a new taxonomy of the individual equipments present in buildings and a fundamental property for future NILM algorithms: the low-rank assumption.

In Chapter 4, we develop a physically-inspired data model that enable us to reproduce and simulate the behavior of the electrical network of a building in a bottom-up procedure. We finally release a new NILM dataset called SHED, made

of simulations of high frequency current data of 8 buildings along with the power consumptions of the individual devices constituting the buildings.

Chapter 2

State of the Art of the NILM Problem Knowledges

Contents

[

]

The main goal of this chapter is to review different analysis and simulation

approaches in the context of NILM. Indeed, it is of first importance to analyze electric data in order to find important characteristics that will be useful either for simulating data or for developing NILM algorithms. Then, there are two ways of analyzing current and voltage measurements in any electric network, would it be a simple electric device or an entire building. The first kind of analysis is called network analysis and is *model driven*. It means that electrical engineer used ideal component and Physic's laws to analyze and understand the relationship between physical quantities such as voltage and current. The second kind of analysis is called data analysis and is more *data driven*. It means that one directly analyzes the measured quantities and tries to infer properties, structure, laws or simple models. In this chapter, we will review the state of the art of both methods and discuss existing methods to simulate data.

2.1 Network analysis

In electrical engineering, the field of network analysis is devoted to finding the voltage across, and the current through, all network components. An electronic component is defined as a basic device that lets the current flow through it. The basis of network analysis is to approximate the behavior of a network by replacing real components (the ones that are manufactured) with idealized elements (the ones that Physic's laws can explain the behavior). Electric laws result in equations (Kirchhoff's laws, Ohm's law, Norton's theorem, Thevenin's theorem) and mathematical tools (complex analysis, Laplace transform, differential calculus) enable the evaluation of current and voltage quantities in any electric network.

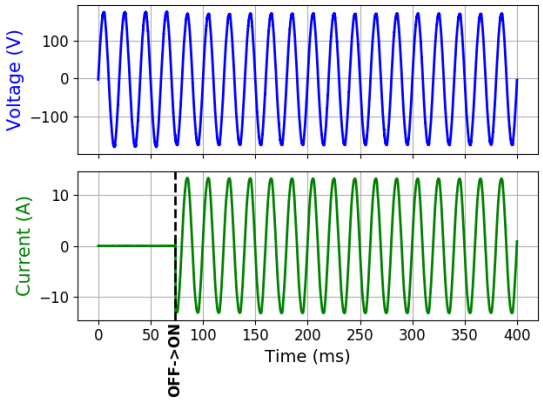
We briefly review the theory of network analysis and illustrate the link with our work from the simplest case (ideal components, linear circuit) to more complex analyses (real components, electric devices, non linear loads).

2.1.1 Resistor

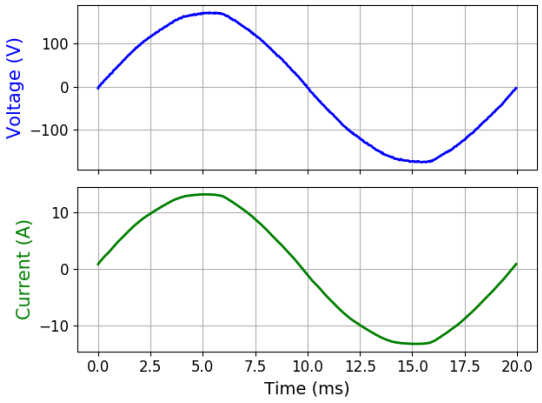
The resistor is the most simple electronic component. Resistors are part of every electrical circuit and reduce the current flows in the circuit by dissipating heat. The relationship between the current $\mathbf{i}$ thats flows through a resistor and the voltage $\mathbf{u}$ across it is given by Ohm's law [Ohm, 1827]:

$$\mathbf{u}(\tau) = R \times \mathbf{i}(\tau), \tag{2.1}$$

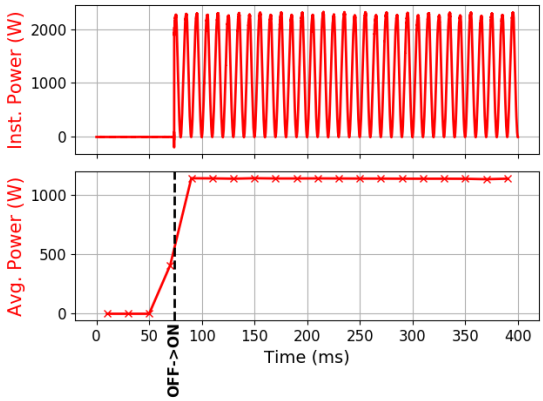
where R is called the resistance. If the voltage is a sine wave at 50Hz, the current will be a similar sine wave whose amplitude will be inversely proportional to R.



(a) Heater's switch ON



(b) Zoom on one voltage period



(c) Heater's power load

Figure 2.1: Voltage and current measurements for a heater from the PLAID dataset [Gao et al., 2014].

Every conductive wire is equivalent to a resistor. Big electric devices, such as heaters or incandescent light bulbs, are electrically equivalent to resistors. Figure 2.1(a) shows a heater's switch ON. The voltage is a periodical sine wave and once switched ON, the current looks similar. Figure 2.1(b) clearly illustrates the proportional relationship between current and voltage and shows the equivalence between a resistor and this electric heater. Finally, Figure 2.1(c) shows the instantaneous and real power load curves. The average power (see Equation 1.5) presents a constant shape after the switch ON. Note that this property of constant consumption has been one of the most important assumption in the early NILM literature.

Resistors are called linear loads. It means that the relationship between current and voltage is linear: $\mathbf{i} = f(\mathbf{u})$ with $f(x) = \frac{1}{R}x$ (for a resistor).

Other types of linear loads exist, namely capacitors and inductors. Both of them, in ideal form do not dissipate energy but store it. The storing details are beyond the topic of this manuscript. The important information is the current/voltage relationship and its expression in an Alternating Current (AC) circuit:

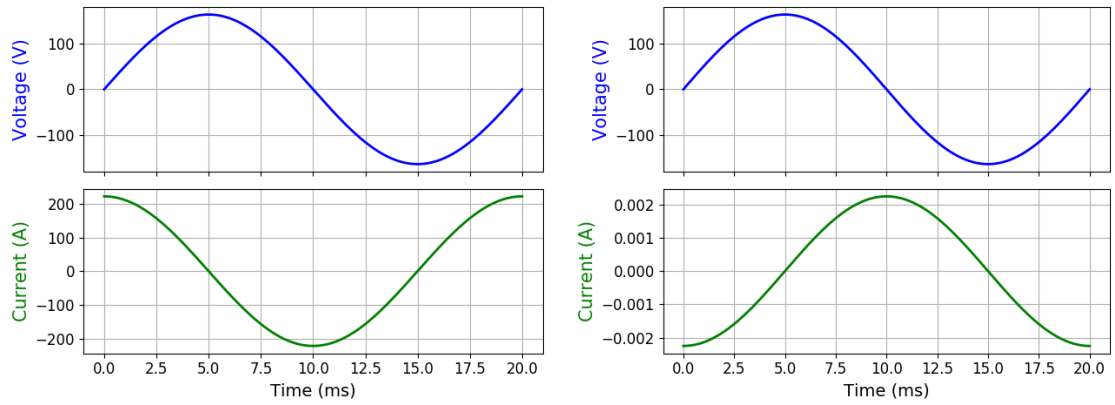
(i) capacitor:

$$\mathbf{i}(\tau) = C\frac{d\,\mathbf{u}(\tau)}{d\tau}, \tag{2.2}$$

(ii) inductor:

$$\mathbf{u}(\tau) = L\frac{d\,\mathbf{i}(\tau)}{d\tau}, \tag{2.3}$$

with C the capacitance and L the inductance. Due to the linearity of the derivative operator, we can verify that inductors and capacitors are linear loads.



(a) Capacitor ($C = 1$F)

(b) Inductor ($L = 1$H)

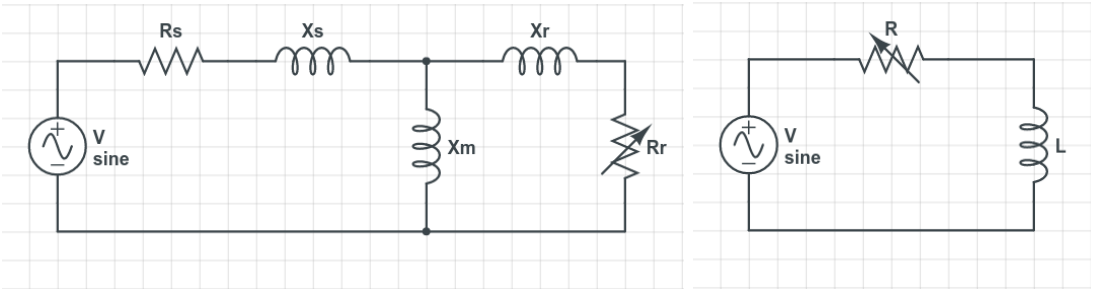
Figure 2.2: Solution of the current/voltage equations for ideal capacitors and inductors.

Figure 2.2 shows the voltage/current relationship during one voltage period. The inductor current is said to be lagging the voltage whereas the capacitor current is leading the voltage (in advance).

When connected to a voltage or current generator, in series or in parallel, linear loads constitute a linear circuit. There is a strong literature for analyzing linear circuit [Chua et al., 1987]. It is driven by linear differential equations and one may use Fourier analysis or Laplace Transform to solve it.

2.1.2 Induction Motor

An induction motor is an electric motor made of two parts: the stator (the fixed cage) and the rotor (the torque is produced around its axis). The induction motor equivalent circuit is a linear circuit made of resistors and inductors: it is called the Steinmetz equivalent circuit [Hubert, 1990] (see Figure 2.3).



(a) Steinmetz (b) Simplified circuit

Figure 2.3: Steinmetz equivalent circuit for an induction motor. The arrow on resistors mean a variable resistor.

To solve for the total current, one can further simplify the circuit, by first finding the equivalent impedance of the parallels inductance and resistance. Then, one can use the series equivalent formula for 2 impedances in series. The voltage/current equation can thus be expressed as:

$$\begin{aligned} \mathbf{u}(\tau) &= L\frac{d\,\mathbf{i}(\tau)}{d\tau} + R\,\mathbf{i}(\tau) \\ D &= R_r^2 + (X_r + X_m)^2 \\ L &= X_s + (X_r^2 X_m + X_r X_m^2 + R_r^2 X_m)/D \\ R &= R_s + (R_r X_m^2)/D \end{aligned} \tag{2.4}$$

This equation is typical of a LR circuit (a circuit comprised of a resistor and an inductor). Solving this equation enables us to calculate the current response to a sinusoidal voltage generator:

$$\text{if} \quad \mathbf{u}(\tau) = V_{peak}\sin(\omega\tau), \tag{2.5}$$

$$\text{then} \quad \mathbf{i}(\tau) = \frac{V_{peak}}{A}\sin(\omega\tau + \phi), \tag{2.6}$$

with: $\phi = \tan^{-1}(\frac{-Lw}{R})$ and $A^{-1} = R\cos\phi - Lw\sin\phi$.

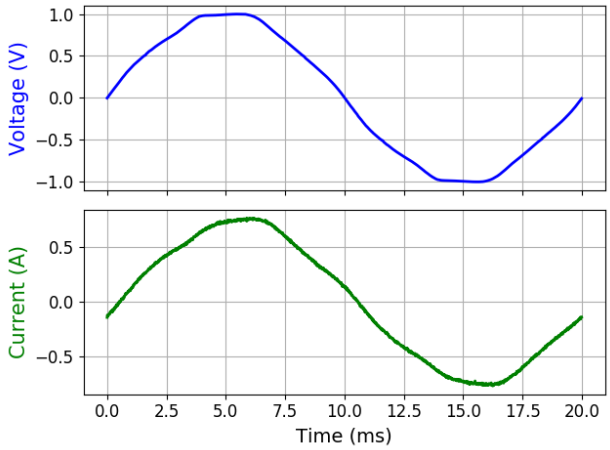


Figure 2.4: Measurements of current and voltage during the operation of a fan made of an induction motor.

Figure 2.4 clearly shows the lag due to ϕ. More generally, every linear circuit can be reduced to a simple RLC circuit (resistor + inductor + capacitor). And the response in current to a sinusoidal voltage is always a sinusoidal function at the same frequency but with a possible time lag (ϕ).

2.1.3 Diode

An electric component that does not respect the linear property is called non-linear. For instance, if a pure sine wave voltage with a 50Hz frequency (called the fundamental frequency) is applied to that equipment, the resulting current may contain several shifted sine waves with frequency multiples of the fundamental (100Hz, 150Hz, ... , called harmonics).

A diode is such a non linear load. It is an electronic component that ideally conducts the current only in one direction. The voltage/current relationship has been studied and the theoretical model is given by the Shockley diode equation:

$$\mathbf{i}(\tau) = I_s(\exp(\frac{\mathbf{u}(\tau)}{n\mathcal{V}}) - 1) \tag{2.7}$$

with I_s is the saturation current, n the ideality factor and $\mathcal{V}$ the thermal voltage. The simulated current is given in Figure 2.5. A light-emitting diode (LED) has a similar behavior as a diode and can emit lights when current flows through it. We have seen that a diode only conducts current in one direction so that the LED will produce light only during the positive voltage semi period. In order to produce light also during the second semi period, manufacturers have designed 2 parallel circuits

with diodes in the reverse side.

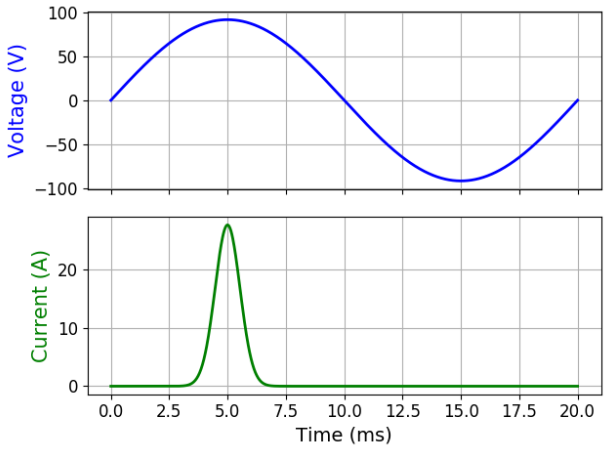


Figure 2.5: Shockley ideal diode model showing the non-linearity of the current/voltage relationship.

Many electric devices present non-linear features. We do not intend to be exhaustive in the listing, but we can name: switched-mode power supplies, compact fluorescent lamps, variable-frequency drives. We have seen that we can solve the voltage/current equation for simple ideal diodes. It becomes harder or even impossible to analytically solve for more complex circuits. In this case, one will prefer to use numerical solvers developed for the electric devices manufacturers.

2.1.4 Electronic loads

The difference between an electric load and an electronic load is that the former turns electricity into an another kind of energy (heating, lighting, mechanical energy) and the latter use electricity to a more elaborated task such as computing calculus or printing images. Such equipments have been extensively studied in [He et al., 2012]. The front-end power circuit is the electric circuit aiming at transforming the input AC voltage into a DC voltage used by the equipment. Depending on the average power consumption of the appliance, this circuit usually comprises different filters (including an electromagnetic interference filter and a rectifier), a power factor correction (PFC) and a DC-DC converter. All those components make the overall circuit complicated and also induce a highly non-linear current response.

2.1.5 Electrical circuit based taxonomy

[He et al., 2012] designed a 2-level device taxonomy based on the electrical circuit characteristics. They only focused on Miscellaneous Electric Loads (MELs) defined as all non-mains connected electrical loads in a building, including a variety of electric devices such as refrigerators, computers, food preparation appliances, space heaters/fans, etc. Their taxonomy consists of a hierarchical breakdown of electrical devices. The first level contains 7 categories depending on the different *modules* of the electric circuit (resistors, inductors, rectifier, filters, converters, transformers, etc). This level is called *front-end circuit topology*. The second level represents the usage or nature of the electric device. Table 2.1 shows the repartition of MEL devices into each level.

Table 2.1: A device taxonomy from [He et al., 2012].

Category Name (level 1)	**Circuit modules**	**Examples** (level 2)
Resistive	• resistors	• heaters, • coffeemakers, • incandescent lamps.
Reactive	• inductors • motors	• refrigerator • fans • washers
Electronic without PFC	• rectifier • filter • DC-DC converter	• personal computer • LED television • game console
Electronic with PFC	• rectifier + filter • DC-DC converter • PFC	• cell phone chargers • PC monitors • scanners
Linear	• transformer • rectifier	• battery chargers
Phase Angle Controlled	• rectifier • thyristor	• light dimmers
Complex	• mix of all categories	• microwave • laser printer

2.1.6 Discussion

This analytic approach is precious for studying the characteristics of particular devices in perfect operating conditions. Reading the equivalent or front-end circuits of an electric device can help understanding its current waveforms. In opposite, this theory can help recognizing an electric element from its current and voltage measurements. [He et al., 2012] have based their device taxonomy on the study of the electrical circuit of small appliances in buildings.

However, it is difficult to use this approach for studying an entire building. First note that electrical devices may be very different from one manufacturer to an other. So writing down the entire equivalent electric circuit may be an extremely hard task. It would require to know in advance all the electric devices with their equivalent electric circuit. Even if we dispose of such an equivalent circuit setting all the parameter values (such that the electromechanical load for an induction motor, the switch on times for lights, etc) would still be very complex, or even unfeasible.

This approach is said *model-driven*, it means that we analyze a building by starting from individual components and models in order to describe the measured quantities (voltage and current). The opposite approach, said *data-driven*, aims at extracting information or structure directly from the measurements. We will explore this latter approach in the following section.

2.2 Data analysis

In this section, we review data driven analyses. We start by listing the public datasets available. Then we introduce the matrix representation of current and voltage measurement. Finally, we discuss analysis on either individual device measurements or on aggregated data (the total consumption of a system).

2.2.1 Datasets

With the increasing development of measurement devices, measuring the electric consumptions of equipment or buildings has become easier. In the last decade, we have witnessed the release of multiple publicly available datasets of different quality and with different sampling strategies. The public datasets used for this study range from low frequency to high frequency sampling and correspond to measurements of houses (except for one which comes from an university building) and of individual small equipments. We have selected datasets with at least a 1/30Hz

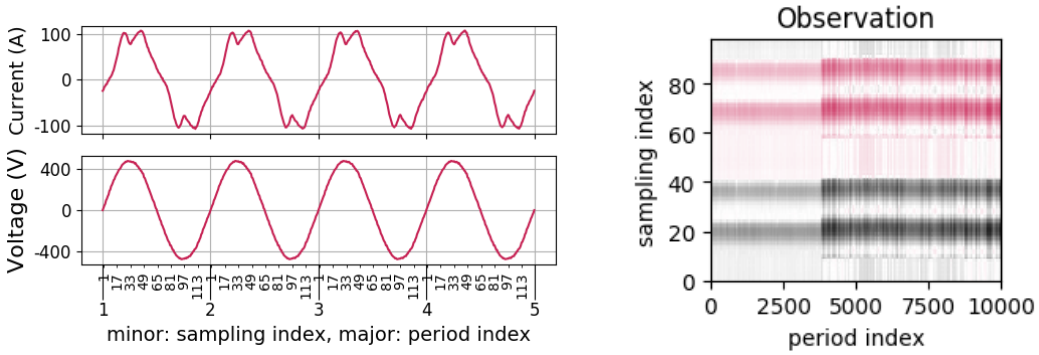
sampling frequency and from each dataset the houses, buildings or devices whose measurements last longer than a week, without distinguishing the weekdays and the weekends. All those datasets are shown in Table 2.2.

Table 2.2: The public NILM datasets: COOLL ([Picon et al., 2016]), WHITED ([Kahl et al., 2016]), PLAID ([Gao et al., 2014]), REDD ([Kolter and Johnson, 2011]), UK-DALE ([Kelly and Knottenbelt, 2015b]), BLUED ([Filip, 2011]), TRACEBASE ([Reinhardt et al., 2012]), ECO ([Beckel et al., 2014]), IAWE ([Batra et al., 2013]), REFIT ([Murray et al., 2017]), RAE ([Makonin et al., 2017]), COMBED ([Batra et al., 2014]).

Name	Data	Number	Phases	Frequency	Type
COOLL	current	840	1	100 kHz	devices
WHITED	current	1259	1	44 kHz	devices
PLAID	current	1074	1	30 kHz	devices
REDD	current	2	2/1	16.5 kHz	residential
UK-DALE	current	1	1	16 kHz	residential
BLUED	current	1	2	12 kHz	residential
TRACEBASE	power	1270	1	1 Hz	devices
REDD	power	6	2	1 Hz	residential
ECO	power	6	1	1 Hz	residential
IAWE	power	1	1	1 Hz	residential
UK-DALE	power	5	1	1/6 Hz	residential
REFIT	power	20	1	1/8 Hz	residential
RAE	power	1	2	1/15 Hz	residential
COMBED	power	1	1	1/30 Hz	commercial

2.2.2 Data Representation

In this section we introduce a very important data representation for current and voltage measurements. Recall that the digitized voltage and current waveforms are denoted: $\mathbf{u}(\tau)$ and $\mathbf{i}(\tau)$, with τ the time. Since the voltage signal is supposed to be a periodic sine wave, it is acknowledged to study the relation between current and voltage during one voltage period. For real measurements, one classically defines a voltage period as the time window between two zero-crossings. A zero-crossing corresponds to the moment at which the voltage crosses zero from negative to positive values. In perfect condition, the number of samples within a period of the voltage sine wave is constant and noted N. The second step to transform the voltage time serie is to construct the voltage matrix using the previously sliced voltage periods: each voltage period ($\in \mathbb{R}^N$) is put into a column of the voltage matrix. Thus the



(a) Time serie representation (b) Matrix representation (current)

Figure 2.6: Data representations: (a) the original time serie representation of the voltage (top) and current (bottom). The voltage zero-crossings are indicated by the major ticks (period index). (b) illustrates the matrix representation of the current used in the following of the dissertation. The colormap is black for positive values and red for negative ones. Each column of (b) corresponds the original current time serie in (a) during one voltage period. The number of rows in (b) corresponds to the number of sampling index (minor ticks) within a period in (a).

number of columns of the voltage matrix is equal to the number of voltage period and is denoted T. $\mathbf{U}(n,t)$ is an entry of the voltage matrix. Notice that, in perfect condition we can write: $\tau = n + t \times N$ and thus $\mathbf{U}(n,t) = \mathbf{u}(n + t \times N)$. This matrix representation can thus be seen as a simple reshaping of the data. The transformation of the current time serie into a matrix is slightly different. As the current time serie is not periodical but only pseudo-sinusoidal (meaning that it alternates positive and negative values), the slicing of the current serie is done according to the voltage periods. It means that the voltage and current slices are time synchronous. Once the current time series is sliced, each pseudo-period ($\in \mathbb{R}^N$) is put into a current matrix similarly to the voltage matrix. Formally, time series $\mathbf{u}(\tau)$ and $\mathbf{i}(\tau)$ are transformed into matrices $\mathbf{U}$ and $\mathbf{I}$, where the rows are indexed by n the sampling index and the columns are indexed by t the period index. Figure 2.6 illustrates this data transformation.

This matrix representation does not affect the calculation of average. It is then given by:

$$\mathbf{P}(t) = \frac{1}{N} \sum_{n=0}^{N-1} \mathbf{U}(n,t)\, \mathbf{I}(n,t). \tag{2.8}$$

For further details on the relation between current, voltage and power see Section 1.1.2 in the Introduction.

It is also possible to down-sample or aggregate these current or voltage signals by averaging several consecutive periods:

$$\mathbf{U}(n,s) = \frac{1}{S} \sum_{t=s\times S}^{(s+1)\times S-1} \mathbf{U}(n,t), \tag{2.9}$$

where S is the number of consecutive periods aggregated.

Transformation of this representation can also be conducted, such as Discrete Fourier Transform (DFT). For instance, DFT can be calculated on each column of this matrix representation. This matrix representation will be at the heart of our statistical analysis of high frequency current measurements in Chapter 3 and of the development of NILM algorithms in Chapter 6.

2.2.3 Individual device analysis

In this section we review the analysis of individual equipment measurements. We first review two proposed device taxonomies based on data analysis. Secondly, we discuss study of data feature extraction driven by the task of recognizing a device from electric measurements.

2.2.3.1 Taxonomies

On a first analysis and attempt to model the power consumption of electrical devices, [Hart, 1992] has proposed a device taxonomy of 3 types of devices: (i) ON/OFF, (ii) Finite State Machine and (iii) continuously variable. It is purely based on the shape of power consumption. The first category ON/OFF is comprised of devices that have a constant consumption once switched ON. The second category, Finite State Machine considers device with multiple states with a constant consumption for each of the states (for example: dish washer, multiple speed fan). The third category includes all the other equipment that have a varying consumption (motor, variable speed drives). A fourth category is often added: (iv) still ON. It consists of devices always ON.

Quickly after this first taxonomy, [Sultanem, 1991] developed an analysis of electrical devices based on the high frequency current waveforms. Table 2.3 lists the 6 categories of the taxonomy.

Table 2.3: A device taxonomy from [Sultanem, 1991].

Category Name (level 1)	**Features**	**Examples** (level 2)
Resistive	• zero harmonic • reactive power	• panel heaters, • cookers, • incandescent lamps.
Pump operated	• current harmonic • high reactive power	• refrigerator • deep-freeze • washer drain pumps
Motor driven without PFC	• same as pump operated • less switching ON	• washing machine • mixers • fans
Electronically fed	• current harmonic • low consumption	• TV • computers • HiFi equipments
Electronic power control	• varying consumption	• halogen lights • vacuum cleaners
Fluorescent lighting	• high 3^{rd} current harmonic • high current/voltage shift	

2.2.3.2 Feature extraction

The main question addressed in the literature using data analysis is: *Is it possible to recognize a device from its consumption measurements?* Formally it is a statistical classification problem where one disposes of observations associated to categories and tries to figure out which category a new observation belongs to. For such a problem, when the available database is limited, a traditional approach consists in designing signal features that would be discriminative between categories. Such a process is called features engineering. It consists of a first research step where one is looking for data transformations or statistics and then applies a classification algorithm on the resulting features. We review here the most commonly used features and their interpretations (when one is available). A good state of the art review for NILM classification can be found in [Sadeghianpourhamami et al., 2017]. The features can be broken down into 2 main categories: low frequency and high frequency depending on the sampling frequency of the input data. Low frequency features are extracted from power or energy measurements whereas high frequency features are computed on current and voltage waveform measurements.

Low frequency features The most common feature family uses *power* consumption. Power step (Δ **P**) is the constant amount of power that a device consumes when it is switched on. It is meaningful only for devices that have a constant consumption. ON duration and frequency of switching ON are also two features related to the power consumptions [Powers et al., 1991, Farinaccio and Zmeureanu, 1999, Marceau and Zmeureanu, 2000]. Although power step expresses the electric design of an equipment, duration and frequency illustrate the operation of the appliance.

Real power features can also be augmented by reactive power to create the **P**-**Q** features family. As the power step, the reactive step can be computed. Reactive power helps differentiating two equipments with the same power consumption [Cole and Albicki, 1998]. **P**-**Q** features are known to fail at discriminating non-linear loads.

High frequency features High frequency measurements consist of current and voltage time series. A common transform used is the Discrete Fourier Transform (DFT) applied to a window of one or several voltage periods (a typical window size is 20 ms for 50 Hz AC voltage). Harmonics values during the steady state is a widely used feature. Approaches using up to the 7^{th} harmonic can be found in the literature [Nait-Meziane et al., 2016, Srinivasan et al., 2005, Sultanem, 1991, Meziane et al., 2017]. One can also find aggregation of the harmonics vector, such as the Total Harmonic Distortion, defined by the ratio of the energy of the first harmonic and the energy on higher harmonics [Dong et al., 2013]. This entire family of features may be called *harmonics*.

On top of this instantaneous value, [Leeb et al., 1995] also used the DFT on the matrix representation introduced in Section 2.2.2 to characterize the time varying loads. It thus maps the current time serie to a two-dimensional function of time and harmonic. This approach is suitable for identifying and discriminating varying devices.

Another interesting approach of characterizing load varying devices is the family of *wavelet* features ([Chan et al., 2000, Oukrich et al., 2017]). In a similar fashion as the Fourier Transform, Wavelet Transform represents a signal using an orthonormal basis.

Some authors have investigated the relationship between current and voltage through another kind of feature called **U**-**I** *trajectory*. It is a two-dimensional representation of the current and voltage as a scatter plot. The scatter plot may be transformed to a two dimension image and image processing techniques may then be

used for the analysis ([Gao et al., 2015, De Baets et al., 2018]).

2.2.4 Aggregated data analysis

Aggregated consumption (i.e. the consumption of the entire house) has been well studied outside the NILM community. The most important feature used for NILM application in residential buildings is the so-called *one-at-a-time* feature. It states that between two following measurements in a building, only one equipment changes state [Hart, 1992, Kolter and Jaakkola, 2012].

In [Mei et al., 2017], Non-negative Matrix Factorization is used to model total power consumptions using the fact the total power consumption is made of repetitive patterns.

[Batra et al., 2014] have studied the difference between commercial and residential buildings using the measurements from 2 buildings in an educational campus in India. They present qualitative figures suggesting that commercial buildings consumptions present more time dependency than residential ones. They show that the number of events (power variations of at least 100 W) is higher in their buildings compared to residential datasets. They conclude that the *one-at-a-time* hypothesis is not expected to be valid in commercial buildings.

2.3 Simulations

In the section, we finally review approaches to simulate new data.

2.3.1 Low frequency data

Substantial efforts have been made to model electrical devices consumption and simulate datasets in order to evaluate NILM algorithms. [Fischer et al., 2015] used "ON/OFF" models with a probability of a device to be switched ON depending on the time of the day. Other approaches ([Buneeva and Reinhardt, 2017, Barker et al., 2013]) defined more complicated models that can take into account uniform randomness during operation time, multi-state devices or exponentially decaying load curves. Even though these models are efficient for electrical devices in residential buildings, they are too simple to be used in commercial settings which contain smoothly varying devices. The higher complexity of commercial buildings also implies a need for higher frequency data.

2.3.2 High frequency data

It is worth mentioning that high frequency current measures have been studied in several papers [Sultanem, 1991, Lee et al., 2004]. [Lam et al., 2007] used high frequency current/voltage trajectories to classify electrical devices. Public datasets of high frequency current measurements of residential equipments has also been released [Gao et al., 2014, Picon et al., 2016]. Finally, [Liang et al., 2010] developed a simulator for high frequency current measures but without considering long term modeling of current dynamics.

2.4 Conclusion

On the first hand, we have seen that network analysis approaches are precious for studying the characteristics of particular devices in perfect operating conditions. Reading the equivalent or front-end circuits of an electric device can help understanding its current waveforms. However, it is difficult to use this approach for studying an entire building. On the other hand, data analysis techniques have lead to interesting taxonomies and helped finding features useful for NILM algorithms. Finally, we have discussed different approaches to simulate new data.

Although these study on analysis are complete for NILM in residential buildings and especially on low frequency data such as power measurements, it is very scarce concerning commercial buildings and high frequency data.

In the following chapter, we will develop our own data analyses for commercial buildings data and will especially focus on high frequency current measurements. In Chapter 4, we will propose a new generative procedure to simulate NILM datasets.

Chapter 3

Statistical Analyses

Contents

[

] As previously said, there is a lack of knowledge on high frequency electric data measured from huge systems such as commercial buildings. Analyzing data is primordial in order to find key features that are useful for developing simulators and NILM algorithms. Thus, in this chapter, we analyze total consumption data (or aggregated) and individual consumptions (either power or current and voltage measurements). We first propose a new study on comparison between residential and commercial buildings using two private datasets in addition to public data. It consists of both low frequency total power data (named SILF) and high frequency total current measurements (named SIHF) from 7 commercial buildings. Finally, we study current data from individual equipment using matrix analyses techniques in order to extract useful properties.

3.1 Residential versus commercial buildings

In this section, we explore the statistical differences of electrical signals in commercial and residential buildings. Our goal is to develop quantitative metrics that can describe electrical power signals and differentiate both types of buildings. The physical quantities used have been defined in Section 1.1.2 and the datasets presented in the Section 2.2.1. We study the temporal structure of the power, the distribution of the temporal derivative of the power, and the harmonic content of current waveforms. This will guide us to develop the tools for understanding and differentiating the signals coming from residential and commercial buildings.

3.1.1 Power measurements (low frequency)

In this part we study the statistical properties of power measurements. A particular attention is taken to the dynamic behavior of power changes distribution and to seasonal effects. In order to discriminate residential from commercial buildings, we are particularly interested in state change events (switching on or off, different speed or heating levels) or continuous variations of electrical devices present in the building. These events result in total current signal variations and therefore in a time-varying power consumption. In this section, we used all the power and current datasets presented in Table 2.2. Power values have been calculated from current and voltage for current datasets. Power time series exhibit a temporal structure defined here by high first-order autocorrelation [1](0.92 and 0.99 for respectively residential and commercial buildings at 1/30 Hz, in average over all datasets). This can be explained by the fact that, when a device is switched on it often remains active for several periods. This motivates us to study the power derivatives (or power variations) rather than the power consumptions:

$$\Delta \mathbf{P}(t) = \mathbf{P}(t) - \mathbf{P}(t-1), \tag{3.1}$$

and to characterize its structure at different time scales. To enable the comparison between buildings, the power derivative is normalized so that the mean is zero and the standard deviation is one.

One important structure in time series is the seasonality. It is a weak assumption

[1]For a stationary signal x_t, $t = 0, \ldots, T-1$, the autocorrelation at lag τ is defined as: $R_\tau(x) = \frac{\sum_{t=0}^{T-1-\tau}(x_t-\bar{x})(x_{t+\tau}-\bar{x}_\tau)}{\sqrt{\sum_{t=0}^{T-1-\tau}(x_t-\bar{x})^2}\sqrt{\sum_{t=0}^{T-1-\tau}(x_{t+\tau}-\bar{x}_\tau)^2}}$, where $\bar{x}$ and $\bar{x}_\tau$ are the sample means of respectively x_t and $x_{t+\tau}$.

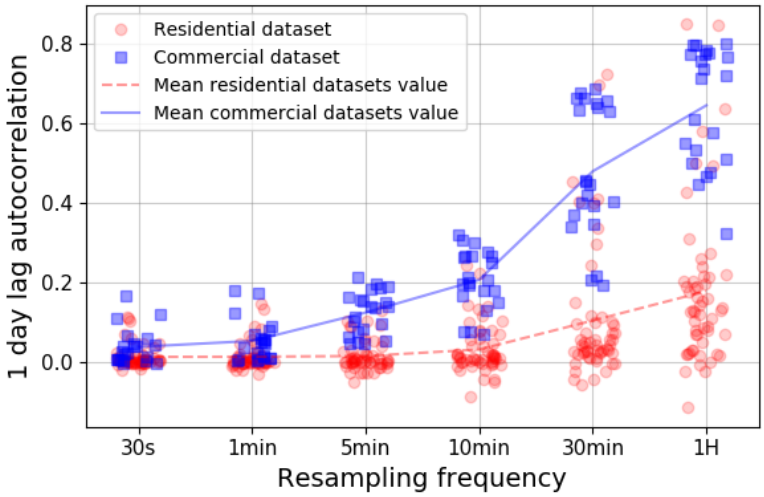


Figure 3.1: Estimation of the 1 day lag autocorrelation for the power derivatives at different re-sampling frequencies for all the datasets (see Table 2.2)

to state that the power consumption and thus its derivative can show daily seasonality due to the habits of the people and time-scheduled equipments. The seasonal effect is characterized here by the autocorrelation with a lag of 1 day of the power derivative (presented in Figure 3.1). It first shows that the derivative of hourly aggregated power discriminates the two kinds of buildings, since the seasonal effect is higher for the commercial ones than for the residential ones (0.65 vs 0.18, in average over all datasets). This can be interpreted by the fact that the consumption patterns are more periodical in commercial buildings than in residential: (i) many equipments are programmed and have recurrent patterns, (ii) the average behavior of occupants is more recurrent than individual behaviors. Figure 3.1 also shows that the seasonal effect is more intense at higher time scale.

At a 1/30 Hz sampling frequency, the power derivative has almost no temporal structure (zero first-order autocorrelation) and can thus be studied as realizations of independent and identically distributed random variables. It can be observed in Figure 3.2 that the distribution of the power derivative for a residential building can be more peaky around zero and has a heavier tail than the one of a commercial building. In order to reflect the statistical differences between residential and commercial buildings, we have selected three statistics that can provide an accurate summary for these distributions: (i) the kurtosis, (ii) the entropy and (iii) the scale parameter of Laplace distribution.

Firstly, the kurtosis is based on a scaled version of the fourth moment of a random

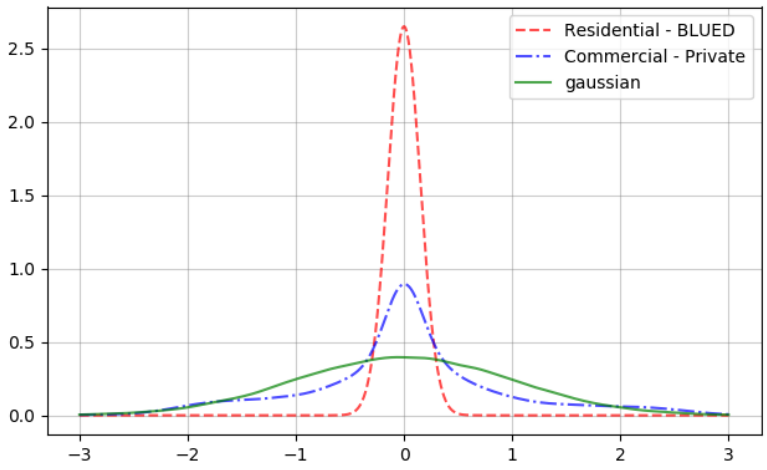


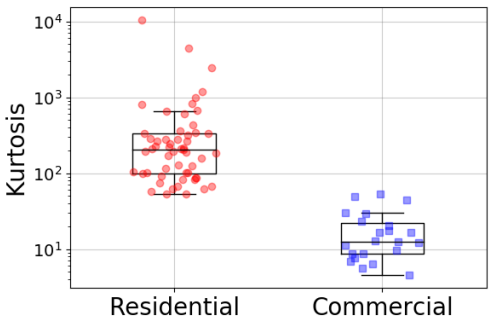
Figure 3.2: Distribution of power derivatives @ 1/30Hz for all the datasets (see Table 2.2)

variable X:

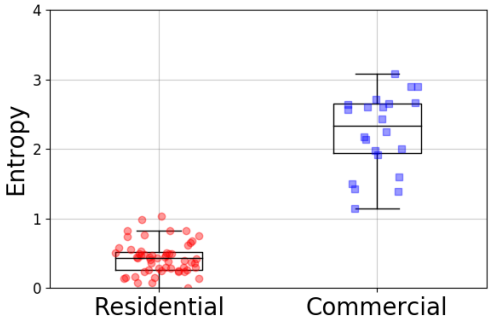
$$\text{Kurt}[X] = \frac{\mathbb{E}\left[(X - \mathbb{E}[X])^4\right]}{\mathbb{E}\left[(X - \mathbb{E}[X])^2\right]^2}, \tag{3.2}$$

where $\mathbb{E}$ is the mathematical expectation. In practice, we used the estimation of the kurtosis of the power derivatives given by the SciPy library [Jones et al., 2001]. The kurtosis has often been used as a measure of impulsiveness: impulsive signals typically have a high kurtosis value [Liang et al., 2008]. Figure 3.3 shows a clear difference in kurtosis for the two types of building. On one hand, high kurtosis value for residential buildings can be explained by the low number of devices and the relative simplicity of the devices (ON/OFF or multi-state) which result in more impulsive power derivative signals. On the other hand, when the number of independent devices increases, the distribution of the sum of the power derivative becomes closer to a Gaussian distribution, as stated by the central limit theorem. It explains why kurtosis values for commercial buildings are closer to the standard Gaussian kurtosis value (3) than kurtosis values for residential buildings. It can however be observed that the kurtosis for commercial buildings remains high compared to the kurtosis of the standard Gaussian distribution, and this characteristic can still be used in NILM algorithms.

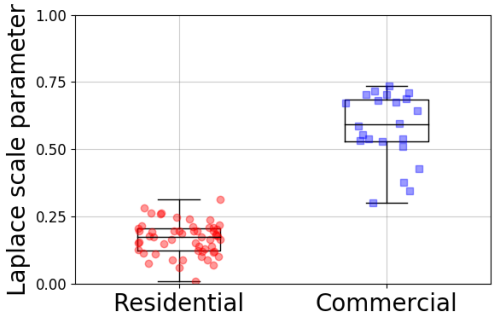
Secondly, entropy is defined as the average amount of information produced by a stochastic source of data. It is based on the logarithm of the probability density



(a) Kurtosis



(b) Entropy



(c) Laplace scale parameter

Figure 3.3: Statistical analysis of power derivatives at a 1/30 Hz sampling frequency for all the datasets (see Table 2.2)

(noted P) of a random variable X:

$$\mathrm{H}[X] = \mathbb{E}\left[-\ln(P(X))\right], \tag{3.3}$$

In practice, we used the estimation of the entropy of the power derivative given by the SciPy library [Jones et al., 2001]. Figure 3.3 shows that entropy values are higher for commercial buildings. This results from the fact that commercial datasets contain more devices and thus more information, which is more complex to encode. This can also come from the fact that there are much more devices with varying power in commercial buildings than in residential ones.

Finally, we analyze the high-kurtosis of the power derivative data by using the Laplace distribution, which is well-known for having a high kurtosis as well and is popular for modeling non-Gaussian data [Kotz et al., 2012]. The Laplace distribution has two parameters: a location (μ) and a scale (b). The location parameter equals the mean of the distribution and is of less interest because it is

0 for our normalized power derivatives. The scale parameter is proportional to the variance of the random variable. In order to compare the datasets, we estimate the scale parameter considering the distributions as Laplace and then compare the estimated parameters. A maximum likelihood estimator of the scale parameter is given by:

$$\hat{b} = \frac{1}{T}\sum_{t=0}^{T-1}|x_t - \mu|, \tag{3.4}$$

where x_t represents in our case the power derivatives. As shown in Figure 3.3, the estimated scale parameters are higher for commercial buildings. We can finally remark that these 3 criteria promote sparseness in the data.[2]

3.1.2 Current measurements (high frequency)

In buildings the voltage can be considered as a pure sine wave. In the frequency domain this is characterized by a signal with energy entirely concentrated on the fundamental frequency. On the contrary, the current signal shows relatively important energies on harmonic frequencies due to non linear devices present on the network. This property can be measured with the Total Harmonic Distortion (THD). It is based on the coefficients of the Discrete Fourier Transform (DFT) of the current signal. The DFT and the THD are computed for every period:

$$\text{THD}(t) = 100 \times \frac{\sqrt{\sum_{h=2}^{N}\mathcal{I}(h,t)^2}}{\sqrt{\sum_{h=1}^{N}\mathcal{I}(h,t)^2}}, \tag{3.5}$$

where $\mathcal{I}(h,t) = \sum_{n=0}^{N-1}\mathbf{I}(n,t).\exp\left(-\frac{2j\pi}{N}hn\right)$ is the h^{th} coefficient of the DFT of $\mathbf{I}(.,t)$. Figure 3.4 shows lower values for commercial buildings that may be explained by an important proportion of linear induction motors (heating, ventilation or air conditioning) which do not create current harmonics, since in these devices, the current becomes a linear transformation of the voltage, and the voltage typically does not contain any harmonics.

[2]For Laplace distributed random variable, entropy and the scale parameter are linked: $\text{H}[X] = \log(2\text{be})$.

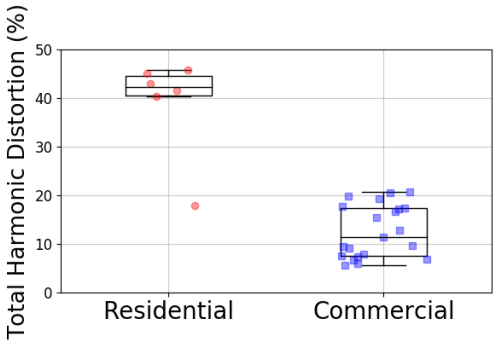


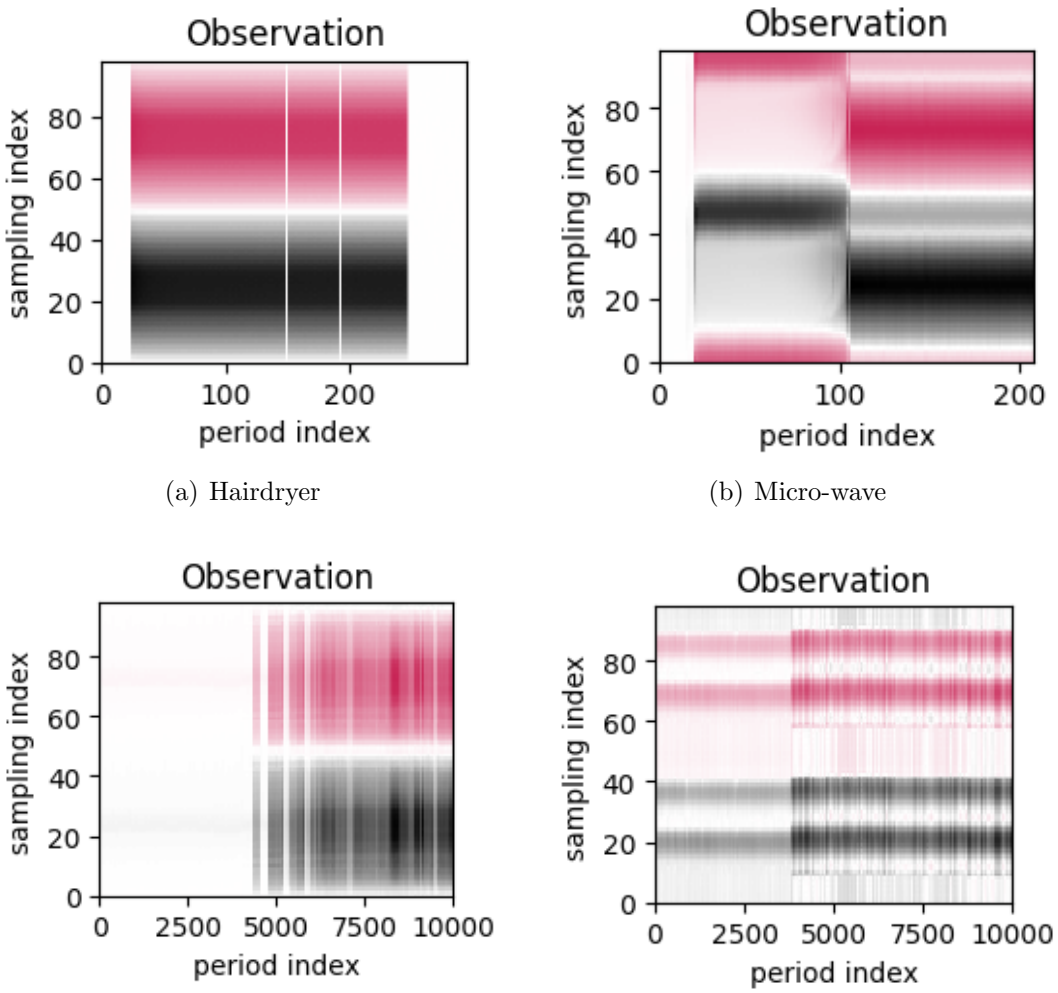
Figure 3.4: Total Harmonic Distortion of current signals for all the "current" datasets (see data column in Table 2.2)

3.2 The low rank assumption

In this section we study the structure and the properties of current matrices measured from individual equipment (see Section 2.2.2 for the construction process of current matrices). For this study we use 3 different databases: PLAID [Gao et al., 2014] and COOLL [Picon et al., 2016] for typical equipments of residential buildings and a private database of 12 devices measured on commercial buildings (half from an office building and half from a shopping mall). Our goal is to find a common structure to all the different devices in order to use it for energy disaggregation algorithms and also for characterizing the differences between devices.

From Figure 3.5, we can first see that the current matrices are very different from one device to another. However, we can also see that they all have a certain structure. The columns of the matrices seems to be highly correlated and only a few columns seems to be able to represent well the overall matrix. More formally these remarks are related to the notion of matrix rank. In the following, we will study the rank of these matrices and compute low-rank approximations.

A low rank approximation of a matrix may be constructed by the matrix product of lower rank matrices. Figure 3.2 illustrates this principle.



(a) Hairdryer

(b) Micro-wave

(c) Lightings

(d) Air handling unit

Figure 3.5: Current matrix measurements from 4 different devices.

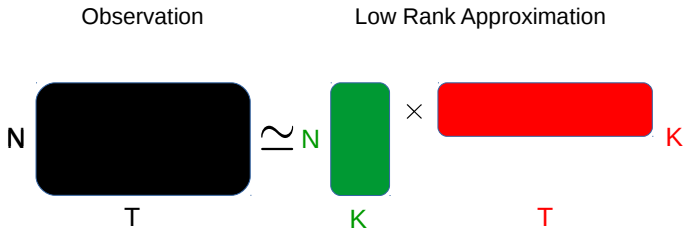


Figure 3.6: Illustration of low rank matrix approximations.

We propose to use a matrix factorization technique called Semi Non-negative Matrix Factorization (SNMF) [Ding et al., 2010] which aims to approximate an observation matrix as the product of a real-valued matrix (**S** called signature matrix) and a non-negative matrix (**A** called activation matrix): $\mathbf{I} \approx \mathbf{S}\,\mathbf{A}$. A detailed explanation of SNMF can be found in Chapter 6, Section 6.2. For this application, all we need to know is that SNMF is able to approximate a $N \times T$ real matrix **I** as the product of lower rank matrices **S** ($N \times K$) and **A** ($K \times T$), where K is a parameter we need to fix and that gives the rank of the approximation.

Figure 3.7 shows 4 examples of low rank approximations. It shows that with only a few columns for the signature matrix, one can get good matrix approximations. For instance, an approximation of rank 1 means that the current matrix **I** is simply made of one column signature **s** which is multiplied by a different amplitude at each time step. An electric heater (made of a simple resistor) is a typical example of such an equipment.

Our goal now is to find the lowest rank such that the approximation error is below a certain threshold. To do so, we need to define an *error* metric. For measuring the approximation error, we use the Signal to Noise Ratio (SNR) between the true current signals and the residual given by the approximation. The SNR is formally defined as follows:

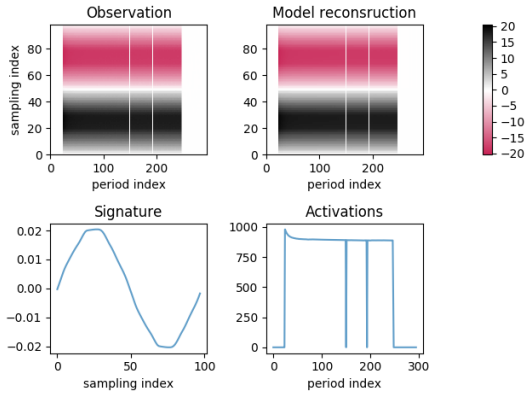
$$\mathrm{SNR} = 10 \times \log_{10}\left(\frac{\sum_{n,t}\left(\hat{\mathbf{I}}(n,t)\right)^2}{\sum_{n,t}\left(\mathbf{I}\,(n,t) - \hat{\mathbf{I}}(n,t)\right)^2}\right), \tag{3.6}$$

where $\mathbf{I}\,(n,t)$ is the real current measurement and $\hat{\mathbf{I}}(n,t)$ is the SNMF approximation.

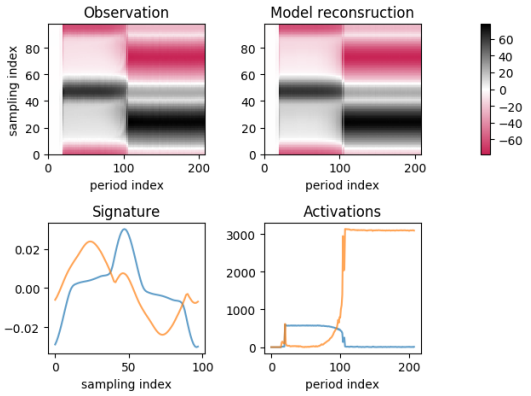
Figure 3.8 shows that for several device categories only one component ($K = 1$) results in very high values of the SNR which means a good approximation. Figure 3.9 shows the required rank to reach a SNR value of at least 50 dB between the model and the error). It can be noticed that devices with higher rank are for the majority found in commercial buildings and not in residential ones (e.g. air handling unit, lift, split, inverter).

3.3 A new device taxonomy

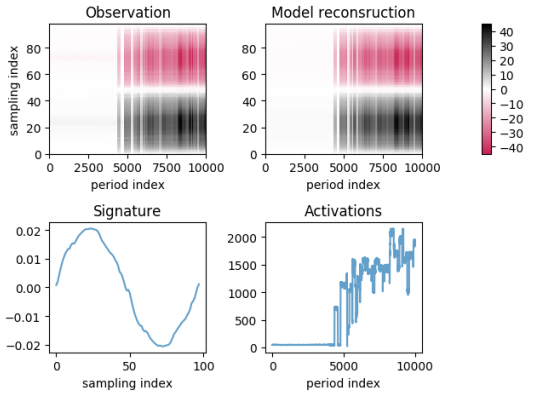
In this section, motivated by the result of our approximate rank analysis, we propose a new device taxonomy. The rank and the nature of the activations (**A**) enable us to classify the devices into 4 main classes. It is based on a certain definition



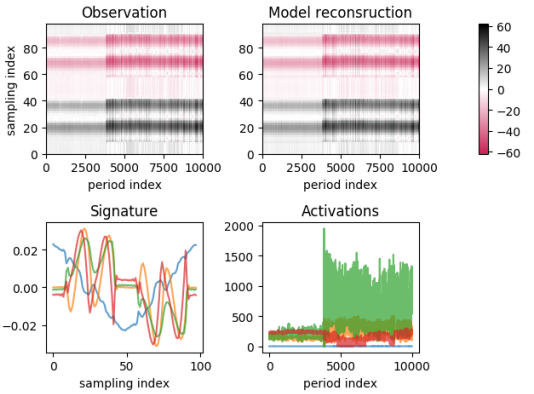
(a) "on/off", hairdryer: rank=1



(b) multi-state, micro-wave: rank=2



(c) varying load, lightings: rank=1



(d) varying signature, air handling unit: rank=4

Figure 3.7: Learned factorizations for the 4 device classes, each of them is composed of (top left) the observations in matrix shape (sampling index × period index), (top right) the model reconstruction (sampling index × period index), (bottom left) the signature matrix: each line corresponds to a column of the matrix and (bottom right) the activation matrix: each line corresponds to a row of the matrix.

of the *complexity* of a device. We can say for instance that the complexity of the device is directly linked to the rank of its approximation. A device signature matrix is considered to be complex if it has more than one column (or equivalently one signature component) and simple otherwise. We also take into account the shape of the activations. It is clear that a continuously varying activation is more complex than a constant or piecewise constant activation. Considering these two characteristics, we propose a new device taxonomy as illustrated by Table 3.1.

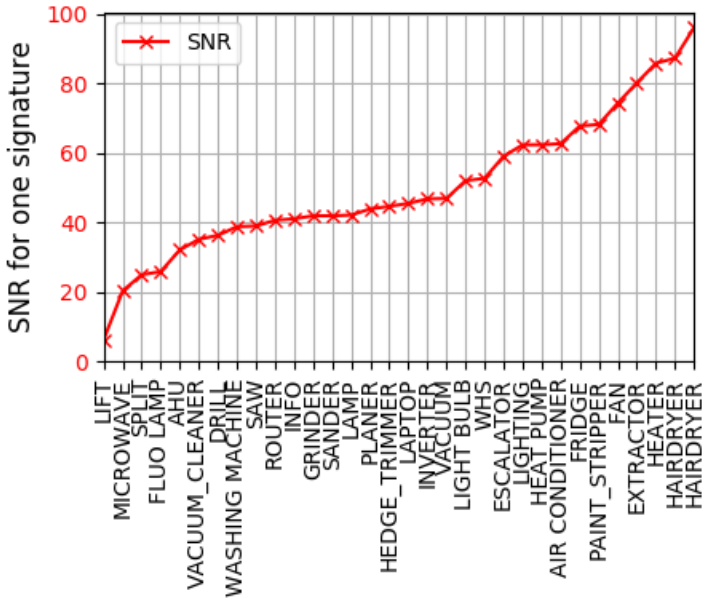


Figure 3.8: SNR values for rank 1 approximations.

Table 3.1: A new device taxonomy based on high frequency current features.

Activation Signature	Simple	Complex
Unique	On/Off or Constant	Varying load
Multiple	Multi-state	Varying signature

In the literature [Hart, 1992, Klemenjak and Goldsborough, 2016], the common devices' taxonomy includes only 3 classes: (i) ON/OFF or constant device, (ii) multi-state and (iii) continuously varying. This approach is based on low frequency features of load curves whereas we take high frequency characteristics into account. We can see in Table 3.1, that the main difference is that the original continuously varying class has been divided into two classes depending on the number of signatures used to model it. Figure 3.7 illustrates the factorizations learned on our four kinds of devices. Figure 3.7(a) shows a hairdryer with one *resistive* signature and a *ON/OFF* like activation. The microwave in Figure 3.7(b) contains 2 different signatures which are not activated together. The most *consuming* signature is the orange one which is the second signature to activate after the switch ON. Figure 3.7(c) presents a group of similar lights with a single signature but a very varying activation. Finally, 3.7(d) illustrates the more complex device category: varying signature. This air handling includes a predominant signature (in green) but the other signatures have

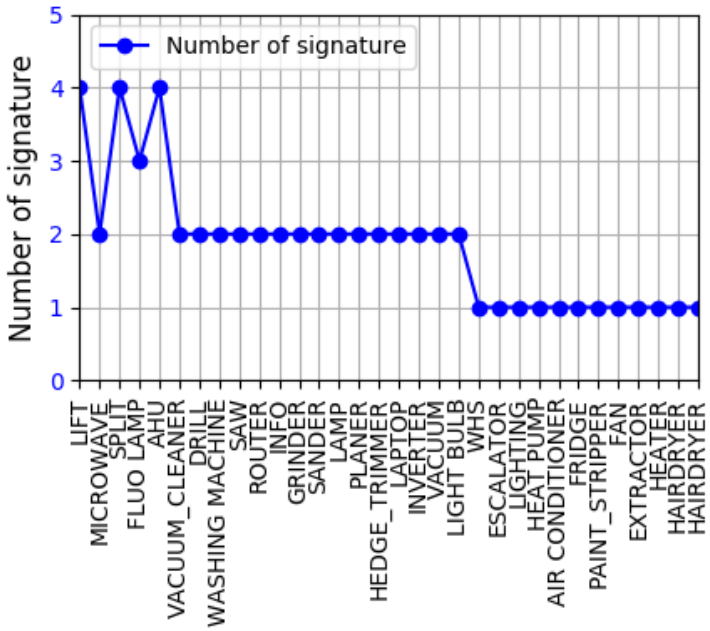


Figure 3.9: Minimum number of signatures to use for reaching a SNR of at least 50 dB.

non negligible activations.

3.4 Conclusion

In this section we have conducted various data analyses. We produced an extensive data analysis on public and private datasets that showed that commercial and residential buildings have significantly different characteristics. The study of the power derivative distribution illustrated that the residential distributions are more peaky at zero than the commercial ones. On top of this, we showed that the kurtosis, the entropy and the Laplace scale parameter of the power derivative are good discriminative indicators for residential and commercial buildings. We explained this difference by a higher amount of devices in commercial buildings and the presence of complex categories of devices (continuously varying equipment, multitude of similar devices). In Chapter 4, we will use this metrics to evaluate the quality of simulations.

Furthermore, as we will see in Chapter 5, these statistical characteristics are in contradiction with the hypothesis used for residential NILM algorithms ('one-at-a-time' and 'constant load'). In this context, detecting a single event on the power curve is a difficult task and this explains why residential NILM algorithms usually

fail when applied to commercial buildings. The statistical metrics used in our study suggest that using a soft version of the "one-at-a-time" hypothesis such as "few at a time" (only a few devices are responsible for the power variations at every instant) would be more realistic. This approach will be enhance in Chapters 6 and 7 to design new NILM algorithms.

Finally, low rank approximations of current matrices have been investigated. We have shown that very low rank (< 5) matrices are good approximations to current measurements. This low rank approximation has been used to propose a new device taxonomy based on the rank and the *complexity* of the activation matrix. These approximations are at the heart of both our simulators in Chapter 4 and our NILM algorithms approach in Chapter 6.

Chapter 4

Simulation

Contents

As already mentioned, the biggest obstacle to the development of NILM algorithms is the lack of data to learn or evaluate them. Even if unsupervised algorithms require less data to be developed, they still need data of good quality for evaluation purposes. In this chapter, we develop a physically-inspired data model that will enable us to reproduce and simulate the behavior of the electrical network of a building in a bottom-up procedure. In our modeling strategy, we first break the overall modeling problem into simpler subproblems by making use of the hierarchical structure that an *individual electric device* belongs to a *category* of devices, and a *building* contains

multiple devices that correspond to different categories. In agreement with this taxonomy, we first develop a generative model for an *individual building*, and for each building we then develop a model for each *device category*. These two models are based on Kirchhoff's current law (see physical preliminaries 1.1.2). Subsequently, we propose a generative model for each device by imposing a low-rank structure on the current waveforms. Finally, we use analytical tools to illustrate that our device model fits well real current measurements.

4.1 A model of consumption for buildings

4.1.1 The building model

The model that we put forward in this section relies on several hypotheses. First, all the electrical devices are supposed to be plugged in parallel on the network: the current waves observed at the main breaker of the network are then the sum of the currents of all devices. This is a direct application of the Kirchoff's current law. Then, the electrical network is supposed to be in ideal conditions: the voltage is considered to be identical on each node of the network and independent from the current. Moreover, in the following, the current signals of devices do not depend on the current signals of other devices. This assumption only holds if the voltage signal is purely periodic since the current waveform depends on the voltage waveform for most devices: $\forall t,\ \mathbf{U}_{main}(n,t) = \mathbf{u}_0(n)$.

Finally, for the sake of simplicity, only single-phase electrical networks are considered here, but three-phase networks can be simulated in a similar fashion.

These assumptions lead us to the following model for total current:

$$\mathbf{I}_{main}(n,t) = \sum_{c \in \mathcal{C}} \mathbf{I}_c(n,t) + \epsilon(n,t) \tag{4.1}$$

where $\mathbf{I}_{main}$ is the total current measured at the main node of the network, $\mathbf{I}_c$ is the current signal of a category c of appliances, $\mathcal{C}$ is the ensemble of category indices, and $\epsilon(n,t)$ is a zero-mean Gaussian noise.

4.1.2 The category model

Since the number of identical equipments can be high in large buildings (*e.g.* corridors light bulb, computers or resistive heaters), it is often more important and easier to evaluate a whole category consumption instead of each single device consumption

(especially for specific NILM applications such as energy management). We then define herein a category as the aggregation of one to many similar equipments:

$$\mathbf{I}_c(n,t) = \sum_{d \in \mathcal{D}_c} \mathbf{I}_{c,d}(n,t) \tag{4.2}$$

where $\mathbf{I}_{c,d}$ is the current of device d belonging to category c. $\mathcal{D}_c$ corresponds to the set of devices belonging to category c.

4.1.3 The device model

Finally, the current of a particular device is modeled using a low rank representation introduced in Section 3.2. Let us start with a rank one formulation given as follows:

$$\mathbf{I}_{c,d}(n,t) = \mathbf{s}_{c,d}(n)\,\mathbf{a}_{c,d}(t) \tag{4.3}$$

where $\mathbf{s}_{c,d}$ and $\mathbf{a}_{c,d}$ are respectively called the *current waveform signature* and the *activation* of device d in category c. The waveform signature corresponds to a fixed pattern that describes the typical current response to the voltage. The activation is a nonnegative magnitude and its nature depends on the type of devices (0 / 1 function or continuously varying). As it has been demonstrated in Section 3.2, the current matrix rank may be greater than one. Thus, we extend our model for more complex devices by enabling the use of more than one signature in the factorization:

$$\mathbf{I}_{c,d}(n,t) = \sum_{k=1}^{K_{c,d}} \mathbf{S}_{c,d}(n,k)\,\mathbf{A}_{c,d}(k,t) \tag{4.4}$$

$K_{c,d}$ is the number of signatures and activations used to model device d. $\mathbf{S}$ and $\mathbf{A}$ are now the signature matrix and the activation matrix.

To fix the inherent scale ambiguity of the multiplicative model expressed in equation (4.4) (every scalar multiplication of a column of the signature can be canceled out by the same scalar division of the corresponding row of the activations), we normalize the signatures such that:

$$\forall\, c,d,k \quad \frac{1}{N}\sum_{n=1}^{N} \mathbf{S}_{c,d}(n,k).\,\mathbf{u}_0(n) = 1 \tag{4.5}$$

It has the double advantage to fix the multiplicative ambiguity and to directly link the activations to the consumed power. Indeed, the power consumption of device d

is given by:

$$\begin{aligned}\mathbf{P}_{c,d}(t) &= \frac{1}{N}\sum_{n=1}^{N}\mathbf{I}_{c,d}(n,t).\,\mathbf{u}_0(n)\\ &= \sum_{k=1}^{K_{c,d}}\mathbf{A}_{c,d}(k,t)\frac{1}{N}\sum_{n=1}^{N}\mathbf{S}_{c,d}(n,k).\,\mathbf{u}_0(n)\\ &= \sum_{k=1}^{K_{c,d}}\mathbf{A}_{c,d}(k,t)\end{aligned} \tag{4.6}$$

We can notice that in the case of a device with a single component ($K_{c,d} = 1$), the activation becomes the power consumption. Otherwise, the power equals the sum of the activations of each component.

4.1.4 The overall model

Combining the individual models expressed in equations (4.1), (4.2) and (4.4) gives the model for the total current:

$$\mathbf{I}\,(n,t) = \sum_{c\in\mathcal{C}}\sum_{d\in\mathcal{D}_c}\sum_{k=1}^{K_{c,d}}\mathbf{S}_{c,d}\,(n,k)\,\mathbf{A}_{c,d}\,(k,t) + \epsilon\,(n,t)\,. \tag{4.7}$$

Finally, we obtain the following formula for the power per category:

$$\mathbf{P}_c\,(t) = \sum_{d\in\mathcal{D}_c}\sum_{k=1}^{K_{c,d}}\mathbf{A}_{c,d}\,(k,t)\,. \tag{4.8}$$

4.2 A generative procedure for dataset simulations

In order to be able to simulate new datasets, we need to solve two more problems. First of all, the SNMF model used to estimate factors (signatures and activations) is analytical and do not provide any generating procedure to simulate new data. Secondly, the lack of publicly available high frequency datasets of individual equipments makes it difficult to learn both signature and activations on the same dataset. To circumvent these issues, we first propose separate generative models for signatures ($\mathbf{S}$) and activations ($\mathbf{A}$) matrices. Then, we estimate their parameters and simulate new data independently for signatures and activations using different datasets: (i) short high frequency current measurements for signatures and (ii) long low frequency power measurements for activations.

4.2.1 Signature Sampling Algorithm

In order to generate signatures, we first decompose the current waveforms that correspond to an individual device, by using the SNMF algorithm that was defined in Section 3.2. Then, we use the output of the SNMF algorithm for generating random signatures. More precisely, we generate a signature $\mathbf{S}^{new}$, by using the following approach:

$$\mathbf{S}^{new}(n,k) \sim \mathcal{N}(\hat{\mathbf{S}}(n,k), \sigma^2) \tag{4.9}$$

where $\mathcal{N}(\mu, \sigma^2)$ denotes the univariate Gaussian distribution with mean μ and variance σ^2, and $\hat{\mathbf{S}}$ is estimated using the SNMF algorithm on the high frequency current signals belonging to individual devices similarly as in Section 3.2. For this task we use the datasets [Gao et al., 2014, Picon et al., 2016]. The variance σ^2 is chosen as an hyperparameter. Figure 3.7 shows four examples of learned signatures for different devices.

4.2.2 Activation Sampling Algorithm

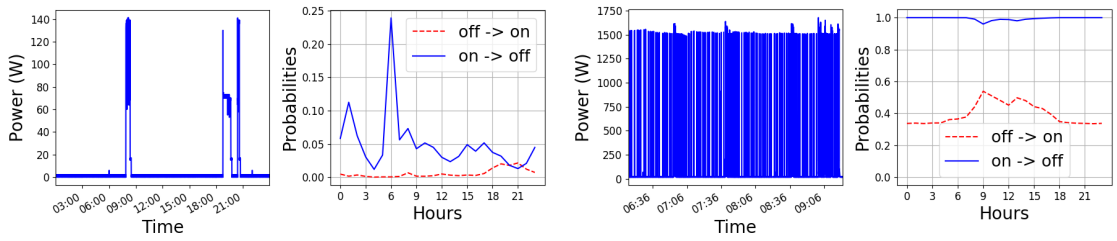
We describe here two different algorithms to simulate the activations: one for simple activations (on/off) and one for complex activations (continuously varying devices).

4.2.2.1 Simple activations

As mentioned in Section 2, a key feature of the activations is their temporal structure. Dinesh et al. [Dinesh et al., 2017] introduced a time-of-day usage pattern for a device defined by the probability of being activated at different periods of the day. These 'periods of the day' are defined as subsets of a partition of the time. In this study, we follow a similar procedure and partition the time in hours. For instance, one subset (a *period of the day* noted $\mathcal{S}_\tau$) may correspond to the slot 10 am to 11 am for every day. The total number of subsets is hence 24.

The approach that was proposed in [Dinesh et al., 2017] assumes that the activations only depend on the time of the day, and therefore it does not take the temporal dependence of the activations into account. We extend that approach by providing a generative model for on/off device activations that take into account the previous state of the device. We are considering here 0 or 1 activations and use a deterministic switching mode 2-state Markov chain to model the device's activation where the transition probability is defined as:

$$\forall \tau, \forall t \in \mathcal{S}_\tau, \forall i,j \in \{0,1\}^2 \quad P[\mathbf{a}(t) = i | \, \mathbf{a}(t-1) = j] = \gamma_\tau(i,j), \tag{4.10}$$



(a) TV-LCD (b) Coffee maker

Figure 4.1: Activation probabilities learned on public dataset (right) and a few hours of the measurements (left).

where t is the time index, $\mathcal{S}_\tau$ is a period of the day and γ_τ the transition matrix for period of the day $\mathcal{S}_\tau$. This model enables us, first, to infer the transition probabilities depending on the period of the day from databases and, second, to generate new activations. Using maximum likelihood inference, the γ parameter is estimated by the following equation:

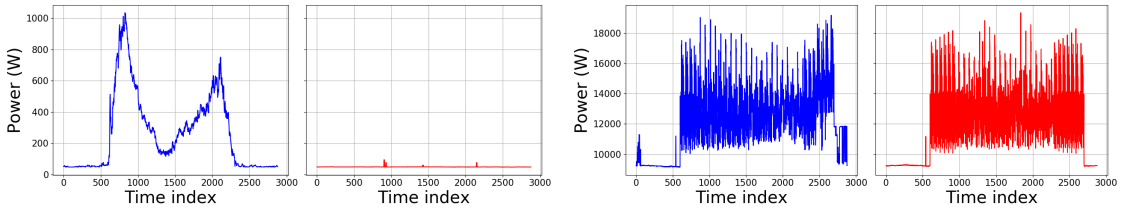
$$\hat{\gamma}_\tau(i,j) = \frac{\sum_{t \in \mathcal{S}_\tau} \mathbb{1}_{[\mathbf{a}(t)=i, \mathbf{a}(t-1)=j]}}{\#\mathcal{S}_\tau}, \tag{4.11}$$

where $\#\mathcal{S}_\tau$ is the size of subset $\mathcal{S}_\tau$. Intuitively, this estimation corresponds to counting the number of ON-to-OFF and OFF-to-ON events occurring during the period of the day $\mathcal{S}_\tau$. We are using the TRACEBASE dataset [Reinhardt et al., 2012]gathering power measurements for individual devices for several days to estimate the parameters. Firstly, we transform the power time series into on/off time series using a simple thresholding mechanism: $\tilde{x}(t) = \mathbb{1}_{[x(t)>20]}$. Secondly, we estimate the model parameters using (4.11). Finally, the learned parameters are used to generate new data:

$$\mathbf{a}^{new}(t) \sim \mathcal{B}\text{er}(\hat{\gamma}_\tau(1, \mathbf{a}^{new}(t-1))) \tag{4.12}$$

where $\mathcal{B}$er is the Bernoulli distribution (which is the natural choice in two state Markov chains).

Figure 4.1 shows two examples of simple activations data and the learned activations parameters. The learned activations show that the probability of switching ON is highest at 8 am and 7 pm for the TV. It also shows that for the coffee maker, the probability of switching ON is quite high all day long and that once ON it immediately switches OFF.



(a) IT devices (b) Heat pump

Figure 4.2: Activation templates learned on the private dataset, the templates correspond to one day (timestep = 30 sec): week-day (left) and day-off (right).

4.2.2.2 Complex activations

In this part, we are considering generating activations by learning 'activation templates' on a private dataset due to the lack of public dataset for complex devices (load varying and signature varying, see Table 3.1). The private data is collected from two large commercial buildings in two different cities in France. It contains 11 device categories and is recorded during several weeks at low sampling frequency. The goal of the templates is to catch the typical power consumption of a device category during a *period of the day* and thus account for the daily seasonal effects shown in commercial buildings (see Section 3.1). Since many equipments are programmed to switch on or off on particular days (air handling unit, heaters) or depend on building occupancy (computers), we distinguish the week days and the days off. In this part the partition of the time is made with period of 30 seconds. The total number of subsets is then 5760 (2880 periods of 30 seconds per week days and days off). In order to compute such templates, we simply average the power consumptions of individual devices over several weeks of data per period of the day:

$$\hat{\mathbf{a}}(\tau) = \frac{\sum_{t \in \mathcal{S}_\tau} \mathbf{p}(t)}{\#\mathcal{S}_\tau} \tag{4.13}$$

The learned templates are illustrated in Figure 4.2. We can observe that IT devices are switched off during day off and have smooth load curves whereas the heat pump has a more noisy consumption.

To generate new data, we multiply a positive noise with the templates to take the day to day variability into account:

$$\forall \tau, \forall t \in \mathcal{S}_\tau \quad \mathbf{a}^{new}(t) = \hat{\mathbf{a}}(\tau) \times \exp(\epsilon(t)), \tag{4.14}$$

where ϵ is a noise variable chosen to add variations to the templates from one day

to another. Indeed, as we use fixed template of one day, for simulating multiple days, one need to add variations not to have exactly the same template concatenated. To do so we chose an AutoRegressive Moving Average (ARMA) process for the ϵ noise [Box et al., 2015]. For ensuring positivity we added the exponential function. ARMA is largely used in time series modeling because of its stationary property and its ability to model autocorrelation at different lags. It is defined as follows: $\epsilon(t) = \sum_{i=1}^{p} \theta_i \epsilon(t-i) + \sum_{i=1}^{q} \beta_i \gamma(t-i) + \gamma(t)$, where $\gamma(t)$ is a white noise and $\theta = (\theta_1, \ldots, \theta_p)$, $\beta = (\beta_1, \ldots, \beta_q)$ are respectively the autoregressive and moving average parameters, which we consider as hyperparameters of the model.

In Section 4.1.3, we defined two kinds of devices with complex activations: (i) single signature or (ii) multiple signatures. While the former has just been addressed, we need to find a generative process for the latter. The proposed generative method uses the same process as before and considers a random convex combination of the activations. Indeed as we already of templates of global activation for the device, all we need to do is to find a procedure to split this global activation into the sub-activations (one for each signature):

$$\forall \tau, \forall t \in \mathcal{S}_\tau \quad \mathbf{A}^{new}(k,t) = \hat{\mathbf{a}}(\tau) \times \exp(\epsilon(t)) \times \delta(k), \tag{4.15}$$

where δ split the global activation into sub-activations. We chose to use a K-dimensional Dirichlet-distributed random variable. Indeed this kind of random variable is widely used to simulate random vectors whose entry sum to 1, which is exactly our expected characteristics. The parameter of such the Dirichlet distribution are $\alpha = (\alpha_1, \ldots, \alpha_K)$ and controls the activation components proportion. α is considered as an hyperparameter.

4.3 The SHED dataset

In order to enable high frequency NILM algorithm evaluation, we release a synthetic dataset called SHED. SHED stands for a Synthetic High-frequency Energy Disaggregation dataset for commercial buildings. The purpose of our simulations is to evaluate the disaggregation performance of NILM algorithms (i.e. the capability to separate individual consumptions from a mixture).

4.3.1 The simulations procedure

Let us start by describing how the simulations are conducted. We first start by learning the model parameters presented in Section 4.2 from public datasets of individual equipment measurements ([Picon et al., 2016, Gao et al., 2014, Reinhardt et al., 2012]) and from one private dataset. We then simulate new current data for the different kind of device by selecting model parameters amongst the learned ones. We have previously seen that we have 4 kind of electric devices and models: On/Off, Multi-state, Varying load, Varying signature. Once the individual current are simulated, we use the building model 4.1.4 to compute the total current measurement of the building.

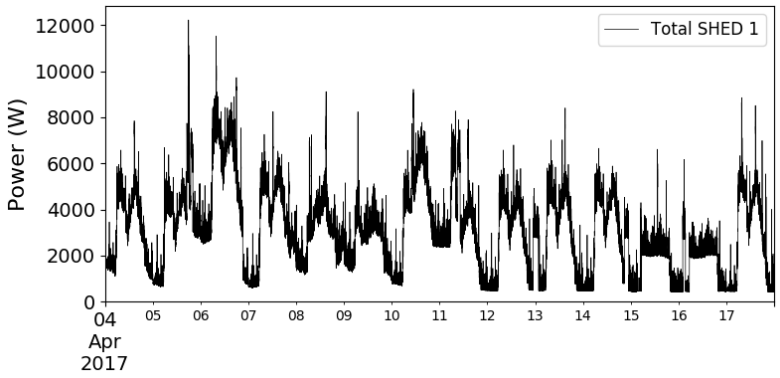
4.3.2 The SHED dataset composition

Table 4.1: Devices used to simulate the buildings in the SHED dataset: On/Off (A), Multi-state (B), Varying load (C), Varying signature (D).

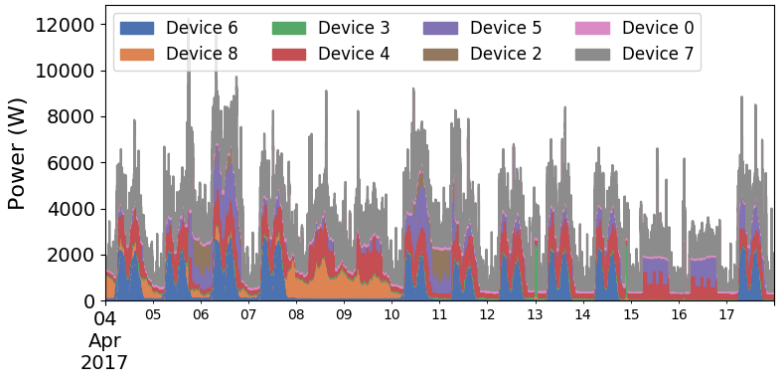
Class	A	B	C	D	Total
building 1	4	0	2	3	9
building 2	1	4	2	3	10
building 3	0	2	2	3	7
building 4	2	0	4	3	9
building 5	0	3	4	1	8
building 6	3	0	3	4	10
building 7	0	0	3	2	5
building 8	0	0	4	4	8

The SHED dataset consists of 8 buildings. For each building, it includes the total current consumption, as well as the individual consumptions corresponding to different categories. For buildings 1 to 6, the individual consumptions consist of low frequency power measurements and for buildings 7 and 8 they consist of high frequency current measurements. One current waveform is recorded at every 30 seconds and for every current waveform 200 points are sampled. Power measurements are also sampled at $1/30Hz$. The choice of the classes of the devices and the number of categories enables us to control the complexity of each building: the buildings are described in Table 4.1.

Figure 4.3 illustrates the power and current data of building 1 in SHED. Figure 4.3(a) shows clearly the daily recurrence of the total power, with high value during the day and low values at night. Figure 4.3(b) is a stacked plot of individual consumptions. It shows that certain devices have a very recurrent consumption



(a) Total power



(b) Disaggregated power

Figure 4.3: Total and disaggregated power consumptions of buildings 1 of the SHED dataset.

(devices 4 or 6) and others have more random one (devices 7 or 8). Finally, Figure 4.4 shows that the current waveforms may be very different during the two weeks of data. Notably, harmonics may appear.

Detailed plots of Buildings 1 to 8 of the SHED dataset can be found in the Appendix A.

4.3.3 Dataset quality evaluation

The quality of the device model has been evaluated in Section 3.2. We now evaluate the total current of the building. We use the metrics introduced in Section 3.1 to check the quality of the simulations. Figure 4.5 shows clearly that the simulated datasets share very similar statistical properties as real commercial datasets. It provides a strong justification that our simulations are realistic. We can however notice that the THD values of simulations are more spread than for commercial

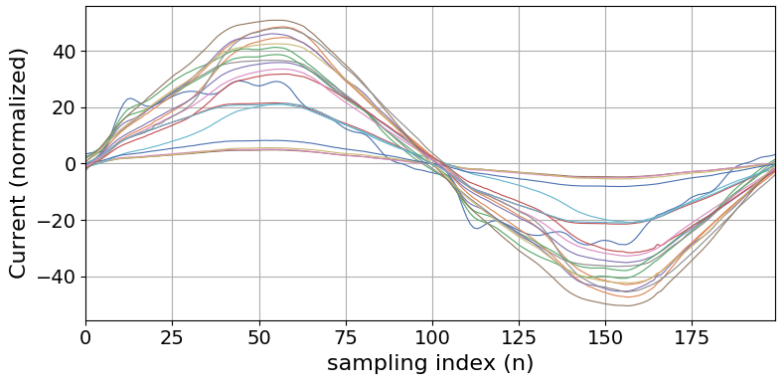
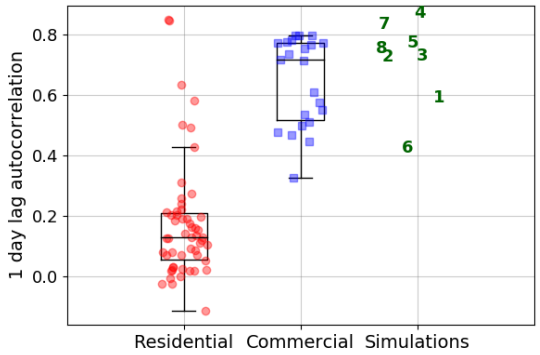
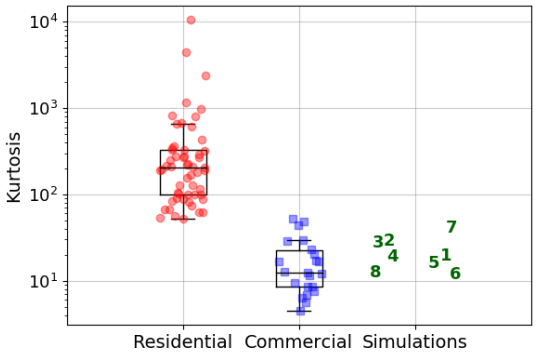


Figure 4.4: Current waveforms randomly sampled for buildings 1 of the SHED dataset.

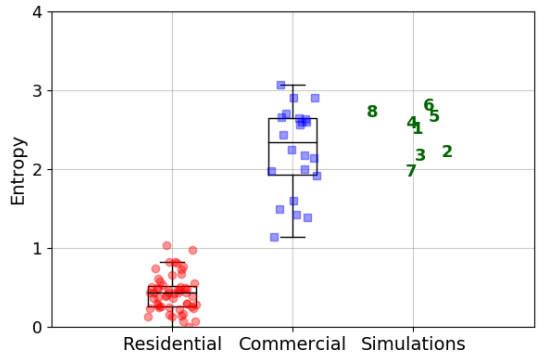
buildings. It may be explained by the fact that the public datasets used for simulating signatures mostly correspond to residential equipments.



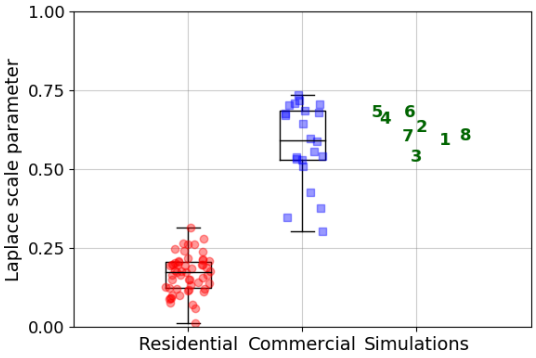
(a) Autocorrelation



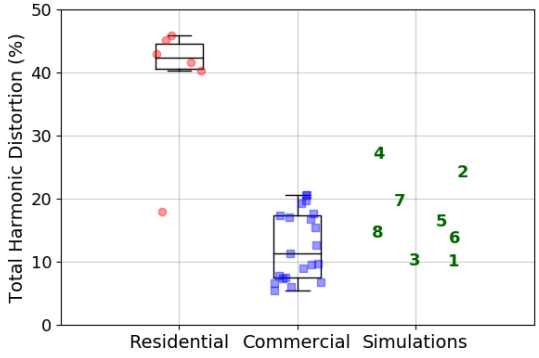
(b) Kurtosis



(c) Entropy



(d) Laplace scale parameter



(e) THD

Figure 4.5: Quantitative evaluation of the simulated datasets: comparison of the statistical metrics of simulations and real datasets. Every circle or square corresponds to one building. Numbers in simulations columns correspond to building indexes in the SHED dataset.

4.4 Conclusion

Motivated by the lack of data for commercial buildings, we developed a generative model for synthesizing high frequency current waveforms. Inspired by physical realities, it is compound of three layers: devices, categories and buildings. Our device model is based on a matrix factorization approach and a low rank assumption, breaking down high frequency current waveforms into signatures and activations components. The model efficiency has been validated with real data. Finally, we proposed a simulation procedure that enables us to learn parameters on real data and then simulate new synthetic data. Our quantitative evaluation experiments showed that the simulated datasets share the same statistical properties as real datasets. To enable algorithms testing and comparison, a simulated dataset called SHED is released at https://nilm.telecom-paristech.fr/shed/.

The SHED dataset will be used in Chapter 8 for benchmarking the performance of our different NILM algorithms.

Part II

On Solving

Motivations After having introduced the Non Intrusive Load Monitoring (NILM) problem, analyzed the data and proposed a simulation procedure, we can now address the task of solving the NILM software problem. A first difficulty is that the NILM literature is scattered across a wide variety of algorithms. Secondly, the availability of high quality data unable us to use supervised learning algorithms which have proven to be very efficient in other domains. Thus the main goals of this part are:

(i) to review the different approaches in the NILM literature,

(ii) to develop a new unsupervised technique.

Organization This part is split into 5 chapters:

In Chapter 5, we review the development history of NILM algorithms: (i) Pattern Recognition, (ii) Markovian models and (iii) Matrix Factorization. We see that, on top of the choice of the mathematical technique, NILM solutions differ from each other due to the type of data used (as in put and/or output) and to the learning strategy (supervised or unsupervised learning).

In Chapter 6, we introduce our framework of unsupervised learning technique suing high frequency current and voltage data. We first propose a generic formulation of the NILM software problem and formulate the specific problem we want to solve. Based on our low rank assumption (Section 3.2), we set our problem as a Matrix Factorization problem. We review existing Matrix Factorization applicable to such a structure of data (Semi Nonnegative Matrix Factorization (SNMF), Independent Component Analysis (ICA) and Sparse Coding) and detail their limitations for the problem of NILM.

In Chapter 7, to overcome the unaddressed difficulties of processing high frequency current signals, we propose a novel technique called Independent-Variation Matrix Factorization (IVMF), which expresses an observation matrix as the product of two matrices: the *signature* and the *activation*. Motivated by the nature of the current signals, it uses a regularization term on the temporal variations of the activation matrix and a positivity constraint. The columns of the signature matrix are constrained to lie in a specific subspace.

Finally, in Chapter 8, we use IVMF to solve the NILM problem on 3 public datasets: 8 commercial buildings (SHED) and 2 residential houses (REDD and BLUED). We show that IVMF outperforms competing methods such as SNMF and ICA on the commercial buildings. Although our method has been designed for commercial buildings, the qualitative results on residential datasets suggest that it can also perform well on that kind of buildings.

Chapter 5

State of the Art of the NILM Solutions

Contents

In this chapter we aims at reviewing the algorithms developed along the history of NILM. As seen in the Introduction (Section 1.2.2), the general NILM software problem can be broken down into two sub-tasks: (i) the disaggregation and (ii) the classification. This two subproblems have often been tackled together. However,

we can note that the classification problem, recognizing an electric device from its electric consumption (or from some features) is a more general electric problem and can be tackled outside the NILM community. In this Chapter we will focus on disaggregation as the classification task is beyond the scope of this thesis dissertation.

During the last thirty years, researchers have addressed the NILM software problems resulting in solutions that we will classify according to four axes:

(i) Learning strategy (supervised or unsupervised): being a general machine learning problem, the NILM software problem can be treated either as a supervised or as an unsupervised learning task. The main difference between the two paradigms is that for a supervised learning task one disposes of the expected output (called ground truth) for each input data during the learning phase. Whereas one only considers input data for unsupervised learning tasks and have access expected output only for evaluation purposes.

(ii) Types of building (residential, commercial, industrial, other): the type of building considered is essential for the desired application. As presented in Chapter 3, residential and commercial buildings are very different.

(iii) Data sampling frequency (low or high): as described in the Introduction (Section 1.2.1), the sampling frequency of the measurements is extremely important for the considered NILM application. We consider here two categories of measurements. Low frequency data consists of power (real and/or reactive) measurements with a sampling frequency lower than the fundamental voltage frequency (50 or 60Hz). High frequency data corresponds to current and voltage waveform measurements with a sampling frequency higher than the fundamental frequency. We can already note that low frequency data is dominant in the literature.

(iv) The mathematical method (Pattern recognition, Hidden Markov Model, Matrix Factorization or Deep Learning): the mathematical method used for addressing the NILM Software problem can also be found in other types of application.

In the rest of this chapter, and with regard to the type of building, the data sampling frequency and the learning strategy, we will present NILM solutions.

5.1 Pattern Recognition

5.1.1 Residential Buildings

Low Frequency data The first algorithm proposed by [Hart, 1992] tried to estimate the major loads power consumption from the aggregate consumption of the house. It is based on two major assumptions:

(i) *ON/OFF* device model: once switched ON, a device has a constant power consumption.

(ii) *Switch Continuity Principle*: in a small time interval, only a small number of appliances is expected to change state.

Using these two hypotheses, the author has developed a total consumption model as:

$$\mathbf{P}_{main}(t) = \sum_{d \in \mathcal{D}} a_d(t)\, \mathbf{P}_d + \epsilon(t) \tag{5.1}$$

where $a_d(t)$ is a boolean function describing the state of device d at time t (0 or 1 for switched OFF or ON), $\mathbf{P}_d$ is the constant power consumption of device d while switched ON and ϵ is a noise or error term. Then, a two steps inference algorithm has been designed for estimating a and $\mathbf{P}_d$ from $\mathbf{P}_{main}$. The first step is to disaggregate the total power into individual power consumption. The disaggregation algorithm is implemented as an unsupervised power change detection involving time serie filtering and peak detection. For each detected power change, a vector of features is calculated (including the active and reactive power difference). The feature vectors are then clustered and each cluster is associated to one generic ON/OFF device. The second step is to classify the load. This step is a supervised task, using either prior knowledge from other buildings or using a training phase on the particular building.

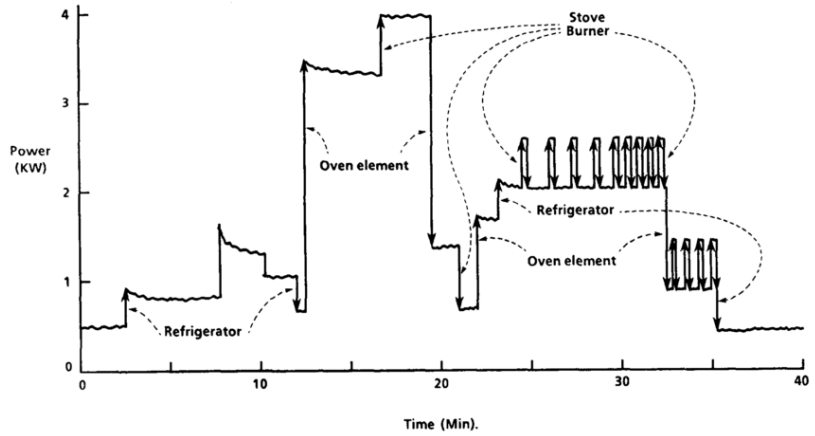


Figure 5.1: Illustration of the two assumptions used by Hart in its algorithm (Figure from [Hart, 1992]).

[Baranski and Voss, 2004] have extended this first approach by enhancing the device model. Due to limitations of ON/OFF models, the authors have developed the concept of Finite State Machine already introduced (but not addressed) in [Hart, 1992]. It consists of modeling the device power consumption as a multiple state engine, each state having a constant power consumption. The difficulty here is to match the different states of each device by constructing a sequence of event (ON $\rightarrow$ state 1 $\rightarrow$ state 2 $\rightarrow$ OFF). These sequences are constructed by maximizing a quality criterion using a genetic algorithm.

High Frequency data The pattern recognition approach has also been extended using higher frequency data. For instance, [Liang et al., 2010] have used current waveforms information (harmonics, temporal waveforms, instantaneous admittance, etc) into the feature vector associated to the events in order to better discriminate equipments. [Lam et al., 2007] have used a feature made of voltage/current trajectory.

5.1.2 Commercial Buildings

Low Frequency data For commercial buildings, most of the proposed approaches using event detection techniques try to solve a partial NILM Software problem, where the consumption of only one equipment is estimated from the total consumption (chillers, variable speed drive, air conditioner, lights, rooftop units). [Norford and Mabey, 1992, Norford and Leeb, 1996] were the first to adapt pattern

recognition using low frequency data from residential to commercial buildings. In a first approach, they tried to estimate only one equipment from the total aggregate by adding a filtering step to reduce the variations caused by other devices.

High Frequency data [Lee et al., 2005] have proposed extensions of this approach using high frequency current measurements. Their method apply a Fourier transform to current waveforms to further analyze the harmonic coefficients. They finally use this information in their pattern recognition technique to filter the consumption of Variable Speed Drive (a special type of motor with controlled speed).

5.1.3 Limitations

The main problem with the pattern recognitions approaches is that it fails at estimating the consumption of all the equipment in a building. An important limitation is the fact that it is highly dependent on the assumption that the power consumption of devices would be constant. We have seen in Chapter 3 that this assumption does not hold especially for big equipment found in commercial buildings.

5.2 Markovian Models

5.2.1 Low Frequency data

In the beginning of the 2010's, [Kim et al., 2011] have started to address the NILM software problem using stochastic approaches such as Markov Models. They have developed a fully unsupervised method using low frequency data. The main difference with the pattern recognition approach presented in Section 5.1, is the addition of a hard constraint:

(i) *Total power conservation*: the total consumption shall be equal to the sum of the consumptions of individual equipments.

This research line is based on the Factorial Hidden Markov Model (FHMM) introduced by [Ghahramani and Jordan, 1996]. While conserving the Finite State Machine model for the devices, they proposed a formulation of the NILM software problem using FHMM. The individual consumptions are modeled as latent Hidden Markov Models ($\{\mathbf{P}_d(t)\}_{t\geq 0}$ where d is the index of a device with a latent state $\{S_d(t)\}_{t\geq 0}$) while the observed total consumption ($\{\mathbf{P}_{main}(t)\}_{t\geq 0}$) is modeled using a probability distribution (called the emission distribution) conditionally on the sum of

individual consumption. To adapt to the specific NILM problem, they extended the model in two directions: (i) state occupancy distribution and (ii) external information. One property of Markov Chains is that the state occupancy duration is exponentially distributed. [Kim et al., 2011] showed that in practice the duration is closer to a Gamma distribution. To deal with this problem, Hidden Semi-Markov Models have been developed by explicitly modeling the state duration and not only the transition probabilities [Yu, 2010]. The second extension concerns the integration of external information in the hidden state modeling. For instance the author insisted on the fact that the state transition probability may depend on time of day, day of week, or input from other sensors. They combine this two extensions in a new Markovian model called Conditional Factorial Hidden Semi-Markov Model and they proposed a learning procedure based on an EM algorithm and a Gibbs sampling strategy.

At the same period, [Kolter and Jaakkola, 2012] introduced the additive FHMM, where the emission distribution is Gaussian and only depends on the sum of average values associated to the hidden states. The Emission distribution now reads:

$$\mathbf{P}_{main}(t)_{|\{\mathbf{P}_d(t)\}_{d=1}^{D}} \sim \mathcal{N}(\sum_{d=1}^{D} \mathbf{P}_d(t), \sigma) \tag{5.2}$$

where $\mathbf{P}_d(t)$ corresponds to the power consumption of a device d at time t, σ is a model noise parameter. On top of the additive model they also proposed a *difference* one being a FHMM applied to the power difference. To avoid outliers problems due to unexpected devices and noise, they added a special latent variable taking real values with a regularization of its variation. The joint density function is then given by:

$$p(\{\mathbf{P}_{residual}(t)\}_{t=0}^{T}\}) \sim \exp(-\lambda \sum_{t=0}^{T-1} |\, \mathbf{P}_{residual}(t) - \mathbf{P}_{residual}(t-1)|) \tag{5.3}$$

The focus of their development is on inference rather than learning, in the sense that they consider they have access to the parameter of their state models. They have access to individual power consumption in order to learn the Hidden Markov Models parameters. This strategy can be referred as *supervised* because the ground truth is needed, at least for a pre-training step. The inference problem of estimating of the states using the aggregated total consumption is done via a method called Maximum A Posteriori. The idea is to find the hidden states that maximize the state distribution conditionally on the observation and the known parameters. This

problem is computationally expensive and there is no way to perform exact inference without enumerating all the possible states M^{DT} (if M is the number of state per devices). The proposed approach is to transform the optimization problem over the latent states into a problem over binary variables. Finally they develop an algorithm to perform an approximate inference called AFAMAP (see Figure 5.2 for an example of results).

Other extensions are given in [Parson et al., 2012, Johnson and Willsky, 2013, Zhong et al., 2014, Shaloudegi et al., 2016].

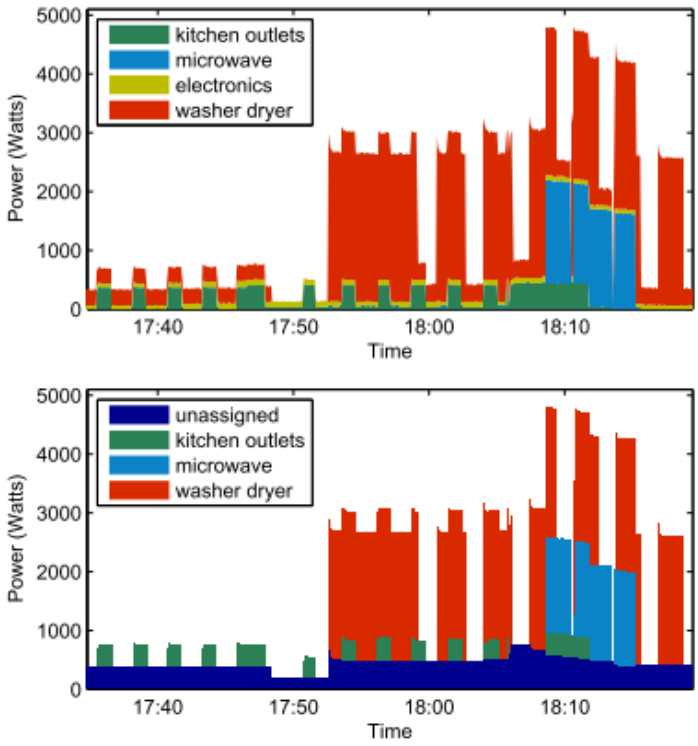


Figure 5.2: AFAMAP results on the REDD dataset. On top the ground truth an at bottom the estimated power consumption. The *unassigned* category represent the residual device introduced in Equation (5.3). Illustration taken from [Kolter and Jaakkola, 2012]

.

5.2.2 High Frequency data

[Lange and Bergés, 2018] introduced the use of FHMM on high frequency data. Instead of using low frequency power measurements (**P**), they used high frequency instantaneous power measurements (**p**). Contrary to the previously mentioned

FHMM approach, they used an unsupervised strategy, where parameters of the hidden individual consumptions are learned directly on the total consumption.This learning problem is known to be difficult due to the exponential number of possible states. They used a variational method to address this problem. The variational distribution is chosen to be a neural network function.

5.2.3 Limitations

We would like to point out that there is no trace of an application of FHMM to commercial buildings data (either low or high frequency). This is obviously due to the fact that FHMM models the power consumption as M-state Markov Chains which is not realistic at all.

It is however interesting that this important limitation has been addressed in [Kolter and Jaakkola, 2012] by considering a residual device that can model the residuals consumption. Although this residual device has been introduced to handle a certain noise in the measurement or in the model, we will further show that it can be much more interesting than this. In Chapter 7, we will explain why the power consumption of **all** the devices can be modeled using more flexible models like the one defined in Equation (5.3).

5.3 Matrix Factorization

In the last decade, Matrix Factorization techniques have been applied to NILM. Let us first introduce this generic method.

Matrix Factorization refers to the wide ensemble of techniques that can decompose observation matrix $X \in \mathbb{R}^{N \times T}$ into the product of two matrices $S \in \mathbb{R}^{N \times K}$ and $A \in \mathbb{R}^{K \times T}$, called factors. Most of the time the decomposition is qualified as *approximated* since:

$$X \approx SA, \tag{5.4}$$

but in some cases the decomposition is qualified as *exact* and:

$$X = SA. \tag{5.5}$$

The factors learning problem is traditionally cast into an optimization problem where a *fit* function of the observation X and the factorization SA is minimized. The

fit function $\mathcal{D}$ is here a function of 2 variables, from $\mathbb{R}^{N\times K} \times \mathbb{R}^{N\times K}$ to $\mathbb{R}_+$. It equals 0 if and only if the 2 variables are equals. It can be a divergence or a distance. Classical examples are based on the euclidean distance or the Kullback-Leibler divergence for instance. Moreover, regularizer functions are usually added to enforce particular characteristics to the factors (sum of vector norms, matrix norms) or to reduce the number of solutions. Finally, the factors may be constrained to lie in a specific space such as the space of positive valued matrices, the orthogonal group or the unit ball defined by a norm. Then, the generic matrix factorization optimization takes the form of:

$$\hat{S}, \hat{A} = \underset{S\in E_S, A\in E_A}{\operatorname{argmin}} \mathcal{D}(X, SA) + \lambda_S \mathcal{R}_S(S) + \lambda_A \mathcal{R}_A(A) \tag{5.6}$$

where $\mathcal{D}$ is a fit function, $\mathcal{R}_S$ and $\mathcal{R}_A$ the regularizer functions, λ_S and λ_A are the regularizer parameters, E_S and E_A are the subspaces of S and A.

Matrix factorization has a long and successful history for solving mathematical and signal processing problems (image, audio, neuroscience, recommender systems). Famous techniques such as Principal Component Analysis, Dictionary Learning [Olshausen and Field, 1997, Aharon et al., 2006, Mairal et al., 2010], Non-negative Matrix Factorization [Lee and Seung, 2001], Semi Non-negative Matrix Factorization [Ding et al., 2010] or Independent Component Analysis [Jutten and Herault, 1991, Hyvärinen and Oja, 2000] lie into this framework.

At that point we would like to make things clear concerning the vocabulary. Factor S will be designated as the signature matrix and one column of S is often named a signature. In the literature it may be called a dictionary (one column being an atoms in this case), a basis (with basis vectors) or finally a mixing matrix. Oppositely, we call A the *activation matrix* where each row is *an activation*. An activation may also be named a code or a source.

5.3.1 Low Frequency data

[Kolter et al., 2010] were the first to propose the use of matrix factorization techniques for low frequency power data **P**, in a supervised way. As seen previously, pattern recognition or Markovian methods consider the observation data as a one-dimensional time serie whereas for matrix factorization techniques, the observed data is first transformed into a matrix. The authors sliced the power time serie into chunk of data of one week so that one column of the observation matrix corresponds to one week of power consumption. For a non-negative observation matrix $X \in \mathbb{R}_+^{N\times T}$, N corresponds to the number of sampling point in one week and T corresponds to the

number of weeks. In a first attempt to adapt matrix factorization to NILM, they proposed a 2 step supervised learning procedure.

In a first learning step, individual equipment measurements (denoted $X^{(d)}$) are used to learn a representation $X^{(d)} \approx S^{(d)}Z^{(d)}$ where $S_d \in \mathbb{R}_+^{N\times K}$ and $A^{(d)} \in \mathbb{R}_+^{K\times T}$ with a sparsity inducing regularization. $S^{(d)}$ contains a set of K basis vectors and is called the dictionary. $A^{(d)}$ contains K rows of activations. The factors are constrained to be positive because by nature the power consumption is positive.

$$\hat{S}^{(d)}, \hat{A}^{(d)} = \underset{S^{(d)}\in\mathbb{R}_+^{N\times K}, A^{(d)}\in\mathbb{R}_+^{K\times T}}{\operatorname{argmin}} \|X^{(d)} - S^{(d)}A^{(d)}\|_{Fro}^2 + \lambda_A \sum_t (\|A_t^{(d)}\|_1) \quad (5.7)$$
$$\text{such that} \quad \|S_k^{(d)}\|_2 \leq 1.$$

In a second step, the disaggregation step, the learned dictionaries $\{\hat{S}_d\}_{d=1}^D$ (with D the number of different equipments) are used to decompose the total power measurement X:

$$\hat{A} = \underset{A\in\mathbb{R}_+^{KD\times T}}{\operatorname{argmin}} \|X - [\hat{S}^{(1)} \dots \hat{S}^{(D)}]A\|_{Fro}^2 + \lambda_A \sum_t (\|A_t\|_1) \quad (5.8)$$

In the aim of learning dictionaries $\{\hat{S}_d\}_{d=1}^D$ that are best at decomposing signals from aggregate measurements (and not only from individual measurements), another intermediate learning step is introduced. Indeed, it uses a training period where both individual measurements $X^{(d)}$ and the total consumption X are available. The purpose of this step is to adapt the dictionaries in (5.8) so that the activation factors ($\hat{A}$) learned on aggregated data equal to the activation factors learned on individual measurements during the pre-learning step in Equation (5.7).

On top of this discriminative learning step, the authors also proposed a regularization of the energy (sums of power consumption) to match prior information one might have on the expected energy.

In [Elhamifar and Sastry, 2015], the authors use a smaller slicing window size to construct the power matrix and added different kinds of regularization (variation of the activations, correlations of activation) and different constraints on the factors (binary activation matrix A, unconstrained dictionary S).

Finally, Figure 5.3 illustrates factorization results using a day by day matrix representation.

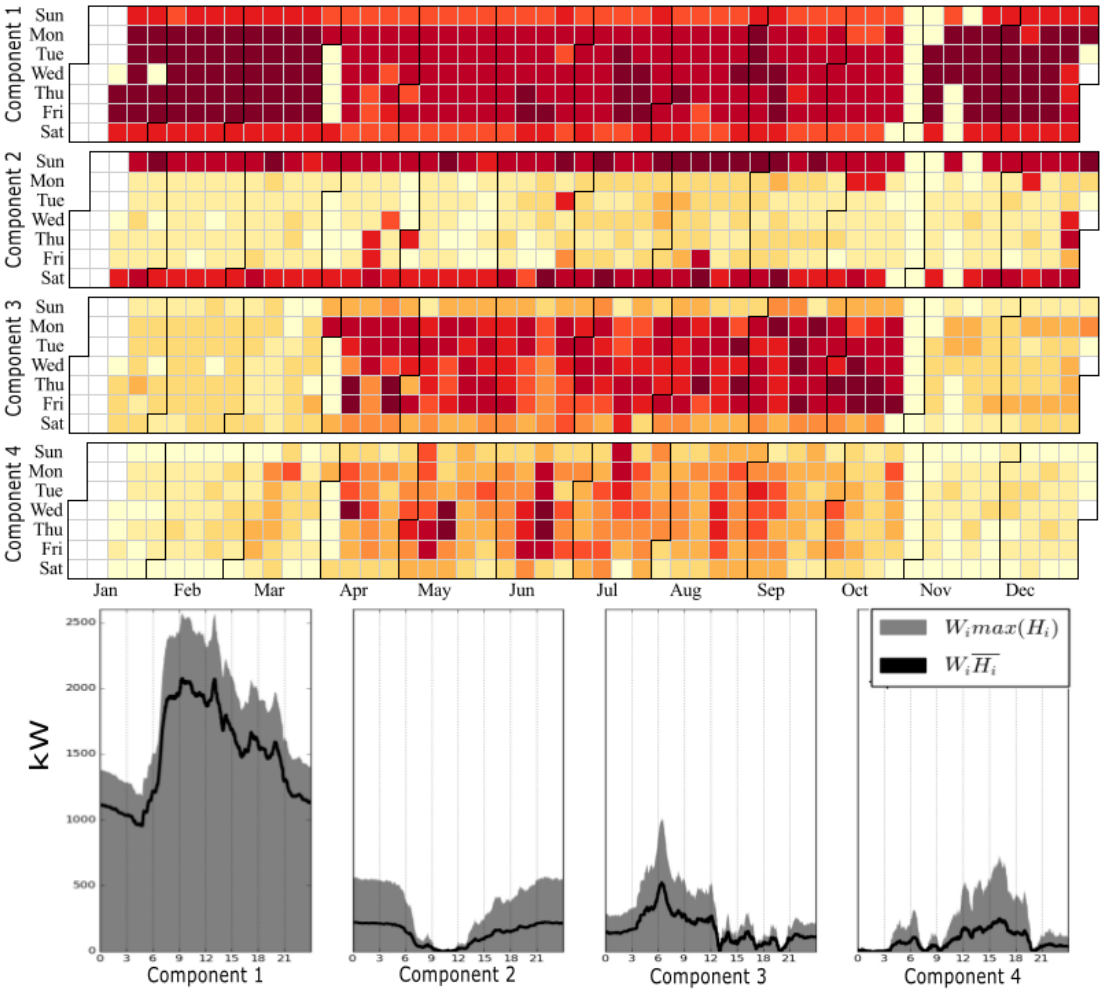


Figure 5.3: Illustration of the dictionary components (bottom) and their respective activations along the time (top). Each dictionary component is interpreted here as a typical daily consumption and is associated to on equipment. The activation shows which day the component is activated and with which intensity. This illustration comes from [García et al., 2017].

5.3.2 High Frequency data

[Lange and Bergés, 2016] designed a matrix factorization technique to deal with high frequency current measurements (**I**). This method is based on the a matrix representation where each column of the observation correspond to the current measurement during one voltage period (see Chapter 3, Section 2.2.2 for more details).

They used matrix factorization in an unsupervised setting. The main idea is to factorize the current matrix so that it can be expressed as the sum of sub-components representing *individual* current matrices. These sub-components are afterwards used to infer devices activity (which device is ON and when). Inspired by FHMM approaches, the authors chose to use 2-state sub-components (as it is done to model ON/OFF devices). This choice results in a binary constraint on the activation matrix A, while the signature matrix S is left unconstrained. The optimization problem reads:

$$\underset{S,A}{\text{minimize}} \quad \|X - SA|_{Fro}^2 \tag{5.9}$$

$$\text{subject to} \quad A \in \{0,1\}^{K \times T} \tag{5.10}$$

From an algorithmic point of view such a problem is said combinatorial and no polynomial time algorithm can solve it. Interestingly, the authors introduced an additional constraint on the factor S (called signature matrix) so that $S = f_\theta(X)$ where f is defined as a neural network with parameters θ. This constraint can be seen as a mean of casting the binary matrix factorization into a deep neural network framework and then efficiently optimize Problem (5.9).

Once the factorization is learned and the sub-component defined, a re-aggregation step is used to infer the activity of the electric devices. A supervised and an unsupervised approaches are proposed and are beyond the scope of this presentation.

This high frequency matrix factorization approach will be extensively developed in the following chapter as it constitutes the same research lead as our own. To conclude with [Lange and Bergés, 2016], we can say that its main drawback is that the binary constraint may be too strong and too far from the reality of electric devices current matrices, as developed in Chapter 3, Section 3.2.

5.4 Deep Learning

In the last decade, augmentation of computing capabilities, access to huge datasets and a high interest in Deep Learning techniques has led to impressive improvements in task automation such as in Computer Vision, in Music Information Retrieval or in Natural Language Processing.

5.4.1 Low Frequency data

As datasets are complicated to acquire in the NILM domain, the first use of Deep Learning only happened in 2015 with the work of [Kelly and Knottenbelt, 2015a]. The idea is to express NILM as a regression (or denoising) problem:

$$\mathbf{P}_{main}(t) = \mathbf{P}_i(t) + \epsilon(t) \tag{5.11}$$

where $\mathbf{P}_{main}(t)$ is the observed total power consumption, $\mathbf{P}_i(t)$ is the power consumption of the targeted device and $\epsilon(t)$ is the remaining consumption. In this scenario, one neural network is learned for each targeted device. The power estimation is given by a mapping function $f(.;\theta)$ such that:

$$\hat{\mathbf{P}}_i(t) = f(\mathbf{P}_{main}(t);\theta) \tag{5.12}$$

In [Kelly and Knottenbelt, 2015a], the authors tried both Recurrent Neural Networks, Long Short Term Memory (LSTM) and Denoising Autoencoders to learn the mapping f. We can also cite the work of [Bonfigli et al., 2018] on Denoising Autoencoders, [Kaselimi et al., 2019] on bidirectional LSTM and [Murray et al., 2019] on Gated Recurrent Units and Convolution Neural Networks.

We can note that several works [Harell et al., 2019, Martins et al., 2018, Jiang et al., 2019] have been based on the WaveNet architecture [Oord et al., 2016] which has been developed to generate audio waveforms. It is principally based on dilated causal convolutions, gated activations and skip connections. This architecture is known to model well long term and multi-scale time dependencies.

This kind of approach is called sequence-to-sequence as a sequence is taken as input and a sequence is returned as an output. In [Zhang et al., 2018], the authors used a sequence-to-point technique where a sequence of measurement is used to estimate only the mid-point of the sequence and showed better performance than sequence-to-sequence methods.

Another approach to Low Frequency NILM using Deep Learning is to use only

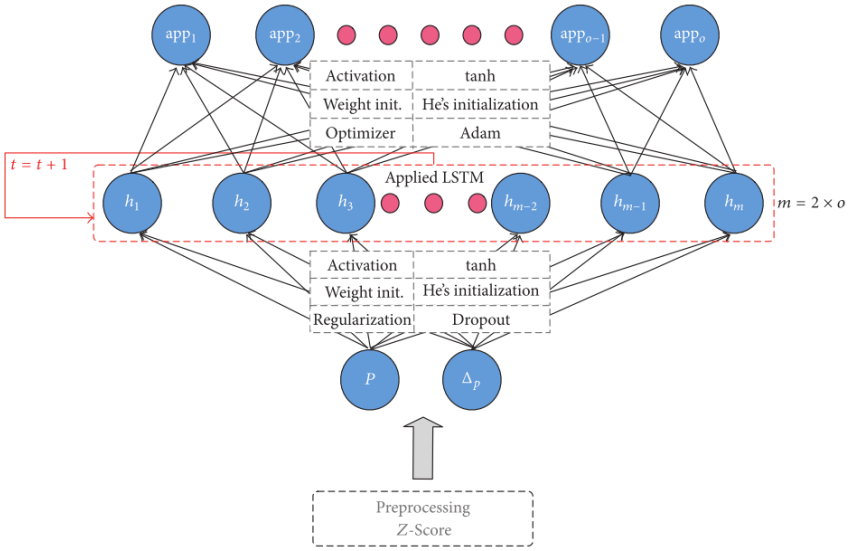


Figure 5.4: Illustration of the use of Deep Learning techniques to solve a NILM problem. Here, a LSTM unit is used to finally predict appliance's state (ON or OFF). This illustration comes from [Kim et al., 2017].

one Neural Network to estimate the consumption of all the devices. This kind of approached has been developed in [Kim et al., 2017] using a combination of LSTM and RNN to predict which devices are active or not (see Figure 5.4). In [Shin et al., 2019], the authors used to Neural Network in parallel: one for estimating the power consumption of the devices and another to estimate the status of the devices (on or off); the two branches are finally multiplicated to produce the final estimate.

Finally we can remark that all but one of previously cited work address the problem of residential NILM. In [Martins et al., 2018] the authors tackled the problem of industrial NILM, i.e. estimating the consumption of industrial machine such as big fans, milling machine in a plant.

5.4.2 Limitations

The principal limitation of applying Deep Learning for NILM is the generalization power of learned models. Indeed, as the available datasets are limited, it is common to see a model learned and tested on the same building just choosing different period of time for each set. When a careful attention is paid to the selection of training dataset and testing dataset, the number of different buildings in both dataset is very limited (from a few to tens).

5.5 Conclusion

We have chosen not to present the recent development on Deep Learning because, to the best of our knowledge, it is only focusing on low frequency power data. It is also of less interest for now because we do not dispose of a publicly available dataset big enough for such methods, especially for high frequency data.

As presented in this literature review, the majority of the approach is devoted to methods using low frequency data and applied to residential buildings. This can be explained by the fact that publicly available data are mostly of this kind. The learning strategy employed is most of the time supervised. This is a particular limitation of the presented methods because the learning phase is done on a limited number of training samples and then the generalization power may be low. Eventhough supervised learning problem are known to be easier to deal with, it is a way more easy to acquire aggregated data with no ground truth. Another striking point is the fact that most of the methods uses low frequency power data. However, it is widely acknowledged that increasing the sampling frequency of data help solving the NILM software problem.

We have seen that [Lange and Bergés, 2016] have leveraged the power of matrix factorization in an unsupervised setting to address the NILM software problem in residential buildings using high frequency current measurements.

Consequently to all these facts, the topic of the next Chapter will be to develop unsupervised learning methods using high frequency current data to solve the NILM software problem in big systems such as commercial buildings. We will rely on Matrix Factorization as our main building block.

Chapter 6

Matrix Factorization for High Frequency NILM

Contents

We have just presented the state of the art of NILM solutions and we have also stated that the specificity of each problem formulation makes it complicated to understand what is the intrinsic problem. We have seen that the quantity and quality of available datasets make it difficult or even impossible to adopt a supervised learning approach.

In this Chapter, we first formulate the NILM software problem as a generic source separation problem. We then show how this generic source separation problem can be transformed into a matrix factorization problem. After having introduced existing and applicable matrix factorization techniques, we will discuss their limitations. These limitations will serve as the basis for our own method in Chapter 7.

6.1 Our formulation of the problem

6.1.1 A generalization of the NILM Software Problem

As already explained in the Introduction (Section 1.3), the diversity of applications and data type makes the problem formulation very heterogeneous in the literature. In an attempt to generalize the NILM software problem formulation, we propose the following definition:

Definition 6.1.1 (The General NILM Software Problem)**.** *Considering the following assumptions:*

(i) Let $\mathcal{F}_{in}$ be a bi-linear transformation taking a voltage and a current signal as input and whose output represents the hardware meter process.

(ii) Let $\mathcal{F}_{out}$ be a transformation taking a voltage and a current signal as input and whose output represents the desired electrical quantity to be monitored.

(iii) Let $\mathbf{i}^p_{main}(\tau)$ and $\mathbf{u}^p_{main}(\tau)$ bet the current and voltage quantities at the main breaker of an electrical circuit on the phase line indexed by p (for p in $\mathcal{P}$ the set of phase line indices).

(iv) Let a category (indexed by c) be a set of electric devices ($\mathcal{D}_c$). The sets of all categories and devices are denoted as $\mathcal{C}$ and $\mathcal{D}$.

Then, the generic NILM software problem reads:

From the measurements, $\forall p \in \mathcal{P}$:

$$\mathcal{F}_{in}(\mathbf{i}^p_{main}(\tau), \mathbf{u}^p_{main}(\tau)), \tag{6.1}$$

estimate $\forall c \in \mathcal{C}$ and $\forall p \in \mathcal{P}$:

$$\mathcal{F}_{out}(\sum_{d \in \mathcal{D}_c} \mathbf{i}^p_d(\tau), \mathbf{u}^p_{main}(\tau)), \tag{6.2}$$

such that the energy conservation holds for every phase line $p \in \mathcal{P}$:

$$\mathbf{i}^p_{main}(\tau) = \sum_{d \in \mathcal{D}} \mathbf{i}^p_d(\tau). \tag{6.3}$$

Definition 6.1.1, contains four important concepts:

(i) The input transformation $\mathcal{F}_{in}$: it defines precisely the sensing process of the hardware metering device. It is a function of a current signal and a voltage signal. For a simple smart meter, this transformation may correspond to the calculus of real power from current and voltage (see Equation (1.5)). This transformation needs to be linear in current and voltage so that conservation equations still hold under the transformation.

(ii) The output transformation $\mathcal{F}_{out}$: it represents the desired quantity to estimate and thus is the illustration of the load monitoring application. This transformation may not be linear. In that case, it will not be possible to establish a conservation equation using this quantity. An example of such a nonlinear transformation is the indicator function that returns 1 if a device is switched on ($\mathbb{1}_{[\mathbf{P}_{main}>0]} \neq \sum_{d \in \mathcal{D}} \mathbb{1}_{[\mathbf{P}_d>0]}$).

(iii) The set of categories $\mathcal{C}$: the notion of device category is essential since one may be interested in monitoring the power consumption of a group of device (i.e., all the light bulbs: $\text{Card}(\mathcal{D}_c) > 1$) or only one particular device (i.e. an air handling unit: $\text{Card}(\mathcal{D}_c) = 1$).

(iv) The set of devices for each category $\mathcal{D}_c$: it is closely related to the set of categories and defines the number and the index of each device present in each category. In a full monitoring approach the sets of device $\mathcal{D}_c$ are a partition of all the device in the electrical circuit ($\bigcup_{c \in \mathcal{C}} \mathcal{D}_c = \mathcal{D}$ and $\bigcap_{c \in \mathcal{C}} \mathcal{D}_c = \emptyset$)

6.1.2 From the NILM Software Problem to Matrix Factorization

In the rest of this Chapter, we focus on a special instance of this previously presented generic NILM software problem, using high frequency current and voltage measurements. Note that we now consider only one phase line.

Definition 6.1.2. *From the current measurements acquired at the breaker panel of a building at a high frequency sampling rate ($> 50Hz$):*

$$\mathbf{i}_{main}(\tau), \mathbf{u}_{main}(\tau) \tag{6.4}$$

Estimate, the real power consumptions of categories of equipments (indexed by $c \in \mathcal{C}$) in the building:

$$\mathbf{P}_c(t) \quad \forall\, c \in \mathcal{C} \tag{6.5}$$

Such that:

$$\mathbf{P}_{main}(t) = \sum_{c \in \mathcal{C}} \mathbf{P}_c(t) \tag{6.6}$$

In this case, we can see that our *input transformation* is nothing but the identity function. Our *output transformation* corresponds the power calculation function: $\mathcal{F}_{out}(\mathbf{i}(\tau), \mathbf{u}(\tau)) = \mathbf{P}_{main}(t) = \frac{1}{\mathcal{T}} \int_t^{t+\mathcal{T}} \mathbf{i}_{main}(\tau)\, \mathbf{u}_{main}(\tau) d\tau$, with τ the time and $\mathcal{T}$ the voltage period. We also use category indexes to group all the devices with similar electric or electronic components.

We can now show how to transform this single-channel source separation problem into a Matrix Factorization problem. We use here the matrix representation defined in Chapter 2 (Section 2.2.2). To makes things clear, from a unidimensional time serie $\mathbf{i}(\tau) \in \mathbb{R}^{NT}$, where N is the number of samples during one voltage period and T is the number of voltage periods in the measurement, we cut slices of size N (one voltage period) which are then set as the columns of a matrix. The beginning of a voltage is defined at the time where of the voltage crosses zero from negative to positive value. Let us denote by $\mathbf{I}_{main} \in \mathbb{R}^{N \times T}$ this current matrix observation. Due to the pseudo sinusoidal shape of current it is often also referred as the current waveform matrix.

This transformation being only a reshaping of the current time serie, the current conservation equation (1.12) still holds. We denote by $\mathbf{I}_c$ the unobserved current matrix of a category of equipments indexed by c:

$$\mathbf{I}_{main}(n,t) = \sum_{c \in \mathcal{C}} \mathbf{I}_c(n,t) \tag{6.7}$$

Using this matrix representation for the voltage $\mathbf{u}$, the power calculations are given by:

$$\mathbf{P}_i(t) = \frac{1}{N} \sum_n \mathbf{I}_i(n,t)\, \mathbf{U}_i(n,t), \quad \forall i \in \mathcal{C} \cup \{main\} \tag{6.8}$$

In Chapter 3, we showed that current matrices of individual equipment or group of same equipment (such as lights or computers) can be accurately approximated by low rank matrices. In the following we make the assumption that a rank one matrix can approximate well the individual current matrices. We recall that a rank one matrix can be defined as the multiplication between a column vector and a row vector:

$$\mathbf{I}_c(n,t) \approx \mathbf{s}_c(n)\,\mathbf{a}_c^\top(t), \quad \forall c \in \mathcal{C}, \tag{6.9}$$

where, $\mathbf{s}_c \in \mathbb{R}^N$ is called a *signature* and $\mathbf{a}_c \in \mathbb{R}_+^T$ is called an *activation*.

Merging Equations (6.7) with (6.9) results in a matrix factorization equation:

$$\mathbf{I}_{main} \approx \mathbf{S}\,\mathbf{A} \tag{6.10}$$

where $\mathbf{S} \in \mathbb{R}^{N\times C}$ is called the *signature* matrix and its columns contain the signatures $\mathbf{s}_c$ for each category. The other factor $\mathbf{A} \in \mathbb{R}_+^{C\times T}$ is called the *activation* matrix and its rows correspond to the activation $\mathbf{a}_c$ of each category.

6.1.3 Learning Strategy

The learning problem defined in this Matrix Factorization framework is then defined as follows:

Definition 6.1.3 (Learning problem)**.** *From the total current* $\mathbf{I}_{main}$ *and voltage* $\mathbf{U}_{main}$ *measurements (in matrix shape), estimate* $\hat{\mathbf{S}}$ *and* $\hat{\mathbf{A}}$*, such that:*

$$\mathbf{I}_{main} \approx \hat{\mathbf{S}}\hat{\mathbf{A}} \tag{6.11}$$

$$\hat{\mathbf{P}}_c(t) = \frac{1}{N}\sum_n \hat{\mathbf{S}}(n,c)\hat{\mathbf{A}}(c,t)\,\mathbf{U}(n,t) \tag{6.12}$$

$$\sum_c \mathcal{L}(\mathbf{P}_c(t)\|\hat{\mathbf{P}}_c(t)) \quad \textit{is minimized.} \tag{6.13}$$

where $\mathbf{P}_c$ *are the true power consumptions per category (often referred as ground truth) and* $\mathcal{L}$ *is a divergence between the ground truth and the estimation.*

A supervised learning strategy would involve an important amount of observation/ground truth couples: $(\mathbf{I}_{main}^b, \{\mathbf{P}_c^b\}_{c\in\mathcal{C}})$ where b is the index of a building. In such a setting one uses a training set of couples observation/ground truth to estimate a mapping function from observation to the outputs by minimizing the loss function $\mathcal{L}$. In this case the factorization $\hat{\mathbf{S}}\hat{\mathbf{A}}$ is an intermediate quantity.

Due to the unavailability of such a training set in our particular application, supervised learning approach are not possible. Oppositely, an unsupervised learning approach would try to infer $\hat{\mathbf{S}}$ and $\hat{\mathbf{A}}$ without having access to the ground truth ($\mathbf{P}_c$). Estimating the performance of an unsupervised approach can be accomplished in two ways. First, one can use a limited size testing set to quantify the estimating error using the same loss function as in the supervised case. Secondly, one can theoretically analyze the method and demonstrate conditions under which the method guarantees estimation of the unknown sources. This includes the notion of identifiability developed in the following section.

In the following section, we will explore classic matrix factorization techniques such as Semi Non-negative Matrix Factorization, Sparse Dictionary Learning or Independent Component Analysis. We will discuss their limitations in the task of Non Intrusive Load Monitoring. In the rest of the dissertation we denote by X the observation matrix instead of $\mathbf{I}_{main}$ for generality and simplicity purposes.

6.2 Matrix Factorization for NILM

Matrix Factorization refers to the wide ensemble of techniques that can decompose a real valued observation matrix $X \in \mathbb{R}^{N \times T}$ into the product of two matrices $S \in \mathbb{R}^{N \times K}$ and $A \in \mathbb{R}^{K \times T}$, called factors. Most of the time the decomposition is qualified as *approximated* since:

$$X \approx SA, \tag{6.14}$$

but in some cases the decomposition is qualified as *exact* and:

$$X = SA. \tag{6.15}$$

The factors learning problem is traditionally cast into an optimization problem where a *fit* function of the observation X and the factorization SA is minimized. The fit function $\mathcal{D}$ is here a function of 2 variables, from $\mathbb{R}^{N \times K} \times \mathbb{R}^{N \times K}$ to $\mathbb{R}_+$. It equals 0 if and only if the 2 variables are equals. It can be a divergence or a distance. Classical examples are based on the euclidean distance or the Kullback-Leibler divergence for instance. Moreover, regularizer functions are usually added to enforce particular characteristics to the factors (sum of vector norms, matrix norms) or to reduce the number of solutions. Finally, the factors may be constrained to lie in a specific space such as the space of positive valued matrices, the orthogonal group or the unit ball defined by a norm. Then, the generic matrix factorization optimization takes the

form of:

$$\hat{S}, \hat{A} = \underset{S \in E_S, A \in E_A}{\operatorname{argmin}} \mathcal{D}(X, SA) + \lambda_S \mathcal{R}_S(S) + \lambda_A \mathcal{R}_A(A) \tag{6.16}$$

where $\mathcal{D}$ is a fit function, $\mathcal{R}_S$ and $\mathcal{R}_A$ the regularizer functions, λ_S and λ_A are the regularizer parameters, E_S and E_A are the subspaces of S and A.

Matrix factorization has a long and successful history for solving mathematical and signal processing problems (image, audio, neuroscience, recommender systems). Famous techniques such as Principal Component Analysis, Dictionary Learning [Olshausen and Field, 1997, Aharon et al., 2006, Mairal et al., 2010], Non-negative Matrix Factorization [Lee and Seung, 2001], Semi Non-negative Matrix Factorization [Ding et al., 2010] or Independent Component Analysis [Jutten and Herault, 1991, Hyvärinen and Oja, 2000] lie into this framework.

At that point we would like to make things clear concerning the vocabulary. Factor S will be designated as the signature matrix and one column of S is often named a signature. In the literature it may be called a dictionary (one column being an atoms in this case), a basis (with basis vectors) or finally a mixing matrix. Oppositely, we call A the *activation matrix* where each row is *an activation*. An activation may also be named a code or a source.

We insist on the fact that we interest ourself to the case of real valued observations and not to the non-negative case. Let us focus on these methods that will be applied to NILM in the following chapters.

6.2.1 Semi Non-negative Matrix Factorization

[Ding et al., 2010] have introduced Semi Non-negative Matrix Factorization (SNMF). In SNMF, the observation matrix X is approximated by the matrix product of two factors: a real valued factor $S \in \mathbb{R}^{N \times K}$ and a nonnegative factor $A \in \mathbb{R}_+^{K \times T}$. The divergence is chosen to be the squared Euclidean distance defined by the Frobenius matrix norm and there is no regularization function. SNMF can formally be written down as the following problem:

$$\begin{aligned} \underset{S,A}{\text{minimize}} \quad & \frac{1}{2}\|X - SA\|_{Fro}^2 \\ \text{subject to} \quad & A \geq 0. \end{aligned} \tag{6.17}$$

Identifiability. Such a structure in Matrix Factorization introduce indeterminacies, i.e. an infinite number factorization respecting the definition (6.17) may exist. The two classic indeterminacies are the permutation and the scale ambiguities. Indeed,

let $\hat{S}$ and $\hat{A}$ be a solution of (6.17). By permuting columns i and j of $\hat{S}$ and rows i and j of $\hat{A}$, the permuted matrices are still solutions of (6.17). Moreover, if we now multiply column i of S by a non-negative scalar and divide with the same scalar row i of A, we end up with a new solution. More generally, let P be a $K \times K$ permutation matrix (a matrix with exactly one 1 on every rows and columns), and C a $K \times K$ diagonal matrix with non-negative entries, then $\tilde{A} = PCA$ and $\tilde{S} = SC^{-1}P^{-1}$ are also solution of (6.17).

Specifically to SNMF, another kind of indeterminacy exists. Figure 6.1 shows 3 different factorizations that give the same observation. To go further, let us consider an observation matrix X and a solution ($\hat{S} = [\hat{s_1}, \hat{s_2}]$, $\hat{A} = [\hat{a_1}, \hat{a_2}]^\top$) of (6.17), one can find an infinite number of admissible solutions as:

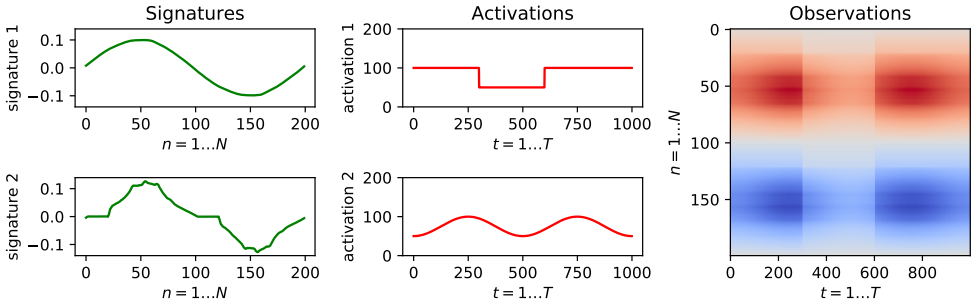
$$\tilde{s}_1 = s_1 + \alpha s_2, \quad \tilde{s}_2 = s_2, \tag{6.18}$$

$$\tilde{a}_1 = a_1, \quad \tilde{a}_2 = a_2 - \alpha a_1, \tag{6.19}$$

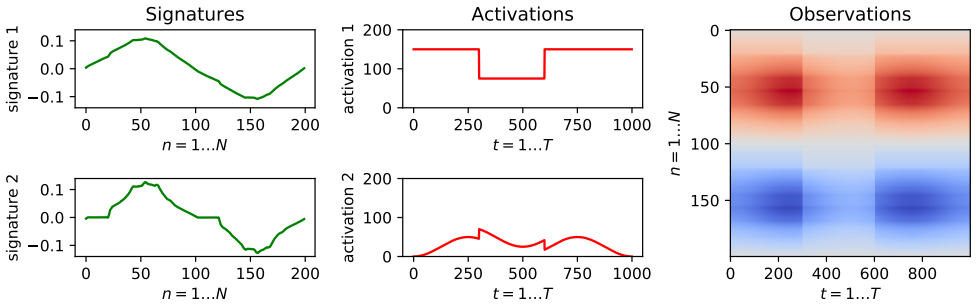
$$0 \leq \alpha \leq \min_{t=1\ldots T} \frac{a_2(t)}{a_1(t)}. \tag{6.20}$$

We can verify that $\hat{S}\hat{A} = \hat{X} = \tilde{S}\tilde{A}$. The condition on α ensures that $\tilde{a_2} \geq 0$.

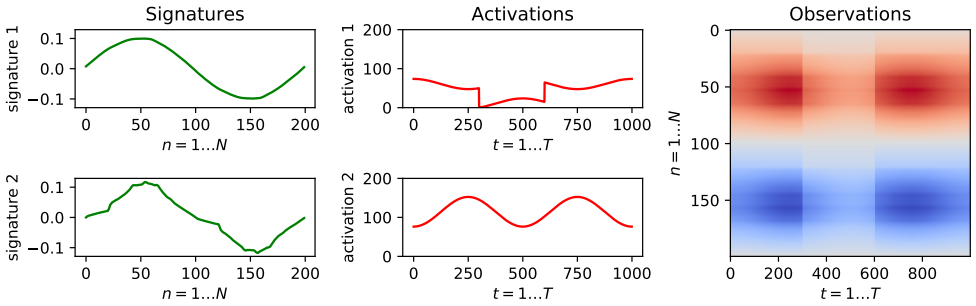
This is major problem of SNMF, indeed for a same observation, several *equivalent* (in terms of fit function) factorization coexists. We will see how this indeterminacy problem can be resolved under further assumptions on the factors.



(a) Original factors



(b) Solution 2



(c) Solution 3

Figure 6.1: Matrix factorization indeterminacies. The 3 sub-figure are 3 equivalent solutions (meaning that their product SA are equal to each other). On top, it present the original factors used for calculating the observation X. The two other sub-figures show activations and signatures that are mixtures of the original ones, while the product SA is still the same. Note that the activations are still non-negatives.

After having discussed the identifiability of SNMF, let us investigate now algorithmic approaches to solve it.

Algorithms. In [Ding et al., 2010], the authors use an alternating optimization

scheme. Indeed, Problem (6.17) is a non-convex non-linear optimization problem. However, a closer look at it shows that the problem is convex on S and A separately. Indeed alternating minimization over S and A will produce a local minimum [Bertsekas, 1997]. The minimization problem over S is an unconstrained quadratic optimization problem and has a closed form solution given by: $\hat{S} = XA^\top(AA^\top)^{-1}$. In opposite, minimizing (6.17) with respect to A is a non-negative quadratic optimization problem that has no closed form solution. In [Ding et al., 2010], a multiplicative update rules is demonstrated to reduce the cost function at each iteration. In [Gillis and Kumar, 2015], they use a block coordinate descent strategy on the rows of A. They point out that the minimization problem over one row of A while fixing the others has a closed form solution: the minimization with respect to a row of A is in fact separable into T scalar optimization. This block coordinate descent on the rows of A enables a faster convergence rate in practice. It also prevents the algorithm to be *locked at zero* due to a zero multiplication in multiplicative updates.

One way to reduce the number of solution to Problem (6.17) is to use regularization function. The idea is to choose among the infinity of equivalent solutions the one that minimizes a certain quantity. The regularization defined by the ℓ_1 norm of the activations is widely used. We will develop it in the next section.

6.2.2 Sparse Coding

Sparse Coding (SC) or Sparse Dictionary Learning is a method introduced by [Olshausen and Field, 1997] in neuroscience. The principle is to learn basis vectors such that the observations have a sparse representation in such a basis. We recall that a vector or a matrix is said to be sparse if it has a limited number of non zero elements. The ℓ_1 norm is widely acknowledge to induce sparsity when used as a regularizing function such as in the well studied Lasso regression [Tibshirani, 1996].

Algorithm. The classic SC optimization formulation reads:

$$\underset{S,A}{\text{minimize}} \quad \frac{1}{2}\|X - SA\|_{Fro}^2 + \lambda \sum_{k,t} |A(k,t)| \tag{6.21}$$

$$\text{subject to} \quad \|S_k\|_2^2 \leq 1, \qquad \forall k \in [\![1, K]\!], \tag{6.22}$$

The constraint on the columns of S is essential for the regularization to operate. Indeed, without this constraint, multiplying S by a scalar and dividing A by the same value would artificially decrease the penalization term without changing the *data fitting* term or the shape of the solution. [Lee et al., 2007] have proposed an efficient algorithm to solve Problem (6.21) using an alternating optimization strategy. Updating S results in a quadratic optimization under quadratic constraints solved using the dual problem. [Mairal et al., 2010] designed an online algorithm . They also proposed extensions to constraint A to be positive like in SNMF.

6.2.3 Independent Component Analysis

[Jutten and Herault, 1991] have introduced Independent Component Analysis (ICA). This subject has been extensively studied in these two books [Comon and Jutten, 2010, Hyvärinen and Oja, 2000]. ICA can be viewed as a special case of matrix factorization. Its main particularity is that in contrary to previously presented techniques, ICA constraints the factorization to be exact:

$$X = SA \tag{6.23}$$

Be careful with the notation as in classic ICA notations, the matrices S and A are inversed. We denote here by S the *signature* matrix corresponding to the mixing matrix in ICA and by A the *activation* matrix corresponding to the *sources* in ICA.

The fundamental principle of the ICA model is that the rows of A represent realizations of statistically independent random variables, also called sources. Unfortunately these sources are unobserved and one has only access to linear mixtures of the sources $X = SA$. We usually assume $K = N$ and thus constraint $S \in \mathbb{R}^{N \times N}$ to be invertible. Thanks to the fact that ICA ensures exact factorization, if $T > N$ one needs only to estimate S or A and can automatically find the other factor using: $A = S^{-1}X$ and $S = XA^{\dagger}$ where $A^{\dagger}$ is the Moore-Penrose pseudo inverse of A.

Identifiability. We have previously seen that matrix factorization models suffers from a number of indeterminacies. An important result in [Comon, 1994] stipulates that, if the original sources are independent and the density of at most one source is Gaussian, then, expect from a scale and permutation indeterminacy, the model is identifiable. This fundamental theorem shows why ICA is able to recover original sources from a linear mixture. Unfortunately, independence is very complex to measure in practice and we will see next how ICA algorithms are designed.

Algorithms. While ICA has a strong literature involving statistics and information theory, we concentrate here on the optimization formulation of ICA. ICA algorithms can be viewed either as maximizing the likelihood of a couple model/observation or maximizing an approximation of the independence of samples via entropy like measures. [Hyvarinen, 1999] has developed the FastICA algorithm which reduces to an iterative procedure that finds extremal points (minimizers and maximizers) of a certain non-linear and non-convex function under orthogonality conditions:

$$\begin{aligned} \underset{S}{\text{maximize}} \quad & \|\hat{\mathrm{E}}\{\mathrm{G}(S^\top X)\} - \mathbb{E}\{\mathrm{G}(\nu)\}\|_2^2 \\ \text{subject to} \quad & X = SA \quad \text{and} \quad S^\top S = I. \end{aligned} \tag{6.24}$$

where X has been centered and whitened, I is the identity matrix, $\mathcal{G}$ is a non quadratic function, $\mathbb{E}$ is the expectation, ν is a multivariate Gaussian variable with identity covariance matrix and $\hat{\mathrm{E}}$ represents the mean (over the columns of a matrix).

FastICA solves Equation (6.24) by finding the fixed points of the first order optimality conditions using a quasi Newton method. [Bell and Sejnowski, 1995] developed Infomax to solve the maximum likelihood formulation. Finally, note that [Ablin et al., 2018] developed a fast (second order) iterative algorithm to solve the same problem using a preconditioned quasi Newton method.

6.2.4 Discussion on Matrix Factorization

Matrix factorization is a powerful tool for extracting or recovering structure from observations. It is a wide field of research and we have restricted our introduction to methods that can be used to solve the NILM software problem.

We have seen that matrix factorizations can be prone to indeterminacies but regularization and assumptions can help handling it. The concepts of independence and sparsity are also at the heart of regularization techniques. In the following section we will detail the limitations of existing method when directly applied to our NILM problem (Definition 6.1.2).

6.3 Limitations of existing Matrix Factorization methods

6.3.1 Semi Non-negative Matrix Factorization

In practice, the activation rows estimated by SNMF exhibit high correlations which is a non expected property of individual equipment power in buildings. Another weakness is that the positivity constraints on A is not sufficient to ensure positivity of the estimated power consumption. As introduced in Equation (6.12) in previous section, the estimated power consumption given by the matrix factorization can be rewritten as $\hat{\mathbf{P}}_c(t) = \alpha_c(t)\hat{\mathbf{A}}(c,t)$ with $\alpha_c(t) = \frac{1}{N}\sum_n \hat{\mathbf{S}}(n,c)\,\mathbf{U}(n,t)$. One can see that constraining the entrance of $\hat{\mathbf{A}}$ to be positive is not sufficient to ensure that $\hat{\mathbf{P}}_c(t) \geq 0$.

6.3.2 Independent Component Analysis

ICA has important advantages over other Matrix Factorization technique which mainly include its identifiability and the high rate of convergence of the developed algorithms. However, the independence hypothesis of the devices consumption is not reasonable since in big buildings many devices are more likely to consume energy during the opening hours than during the night for instance. As explained in Chapter 3, a refined assumption is that the power variations (or sometimes called derivatives) of the devices are independent. It is then usual (see [Feng and Kowalski, 2018]) to apply ICA to a transformation of the data such that the independence assumption is fulfilled in this new domain. Another weakness of ICA in our problem resides in the positivity of estimated consumptions. A Semi Non-negative ICA has been developed, constraining the sources/activations A to be positive [Plumbley, 2003]. Like SNMF, it suffers from the fact that this positivity constraint is not enough for ensuring positivity of estimated power consumption.

6.3.3 Sparse Coding

As we have already seen, Sparse Coding is based on ℓ_1 norm regularization on the activation A to promote sparsity. However, this sparsity hypothesis does not hold at all for our signals. It is obvious that many different equipment are ON at the same time in a big building. However, as shown in Chapter 3, the power differences are more sparse, that is to say, only a few devices switched ON or change their

consumption state at the same instant. As done for, ICA, one can try Sparse Coding on a transformation of the original data. Unfortunately, Sparse Coding will suffer from the same positivity problem as ICA and SNMF.

6.4 Conclusion

In this chapter we have defined the NILM Software Problem in a generic framework that can address all the different formulation found in the literature. We also showed how to transform the single channel source separation problem into a matrix factorization problem using the rank one assumption on current matrices. We have finally reviewed existing Matrix Factorization techniques such as Semi Non-negative Matrix Factorization, Independent Component Analysis and Sparse Coding. Eventhough these methods present interesting properties they suffer from limitations to address the NILM software problem. The main limiting issue is that these methods can not ensure the positivity of the estimated power while promoting independence or sparsity in the variations of the estimated power consumptions.

In Chapter 7 we will address these limitations. Although we have seen that ICA and SNMF have important theoretical limitations for the NILM task, we will use it as baseline methods in Chapter 7 and 8.

Chapter 7

IVMF: Independent-Variations Matrix Factorization

Contents

We have previously seen that our NILM software problem can be transformed into a Matrix Factorization problem. Unfortunately, existing methods do not directly apply. Therefore, in this chapter, we develop a new matrix factorization method to overcome all the limitations of well known techniques such as Semi Nonnegative Matrix Factorization (SNMF), Independent Component Analysis (ICA) or Sparse

Coding (SC). This chapter is organized as follows. We first recall the desired property of the future factorization and propose a new formulation. We then focus on designing an efficient algorithm to solve this newly formulated problem. Finally, we conduct a set of experiment to test the intrinsic performance of our algorithm.

Let X be an observed current matrix, its factorization is given by $X \approx SA$. Let $\mathbf{U}$ be a voltage matrix. Making the assumption that the voltage signal is purely periodic, every column of $\mathbf{U}$ are equals to a denoted voltage vector $\mathbf{u}_0$. In perfect condition, $\mathbf{u}_0$ is simply the sine wave. The estimated power consumption by device is then given by:

$$\begin{aligned} \mathbf{P}_k(t) &= \mathbf{A}(k,t)\frac{1}{N}\sum_n \mathbf{S}(n,k)\,\mathbf{u}_0(n) \\ &= \frac{\mathbf{s}_k^\top \mathbf{u}_0}{N}\,\mathbf{a}_{k.} \end{aligned} \tag{7.1}$$

$$\forall t: \quad \mathbf{U}(n,t) = \mathbf{u}_0(n) \tag{7.2}$$

The desired properties of the estimated power $\hat{\mathbf{P}}_k(t)$ are:

(i) positivity:

$$\forall k,t: \quad \mathbf{P}_k(t) \geq 0 \tag{7.3}$$

(ii) variations sparsity:

$$\forall t, \text{ for lots of } k: \quad \Delta\,\mathbf{P}_k(t) \approx 0 \tag{7.4}$$

where $\Delta\,\mathbf{P}_k(t) = \mathbf{P}_k(t) - \mathbf{P}_k(t-1)$ are the power variations.

(iii) statistical independence, device per device, of their variations:

$$\forall k,k': \quad \Delta\,\mathbf{P}_k \perp\!\!\!\perp \Delta\,\mathbf{P}_{k'} \tag{7.5}$$

7.1 Formulation

We extend SNMF, ICA and SC by introducing: (i) a specific regularization and a positivity constraint over the activation matrix; (ii) linear and quadratic constraints

on the signature matrix. The IVMF optimization problem is then given by:

$$\begin{aligned} \underset{\mathbf{S},\mathbf{A}}{\text{minimize}} \quad & \frac{1}{2}\|X - \mathbf{S}\,\mathbf{A}\,\|_{Fro}^2 + \lambda\mathcal{G}(\mathbf{A}) \\ \text{subject to} \quad & \|\,\mathbf{s}_k\,\|_2^2 \leq 1, \qquad \forall k \in [\![1, K]\!], \\ & \mathbf{s}_k^\top\,\mathbf{u}_0 \geq \alpha_0, \quad \forall k \in [\![1, K]\!], \\ & \mathbf{A} \geq 0, \end{aligned} \tag{7.6}$$

where $\mathcal{G}(\mathbf{A}) = \sum_{k,t} G(\mathbf{A}(k, t+1) - \mathbf{A}(k, t))$ and G is a non-quadratic scalar function and $\lambda > 0$ is the regularization parameter. To induce sparsity on the variation, we propose two choices for G:

(i) the absolute value:

$$G_{abs}(x) = |x| \tag{7.7}$$

(ii) a *smooth* absolute value:

$$G(x) = \sqrt{x^2 + \epsilon} - \sqrt{\epsilon} \tag{7.8}$$

where ϵ is a small positive constant.

Note first that if we take $\epsilon = 1$, the smooth absolute value is equivalent to the classic logcosh$(x) = \log(\cosh(x))$ function in ICA (both of them being equivalent to $\frac{x^2}{2}$ near 0). When ϵ decrease to 0 the limit of the *smooth absolute value* is the absolute value. A second remark is that $G_{abs}(x)$ correspond to the widely used total variation regularization [Rudin et al., 1992]. Eventhough total variation has been introduced for denoising [Beck and Teboulle, 2009a] or inducing piecewise constant shapes [Seichepine et al., 2014], we use it to perform separation of independent sources which can be neither noisy nor piecewise constant.

The *quadratic constraint* on the columns of $\mathbf{S}$ ($\mathbf{s}_k$) enables us to fix the inherent scaling ambiguity in such factorization problems. As explained for Sparse coding, the normalization is essential for the regularization to operate. Indeed, without this constraint, multiplying $\mathbf{S}$ by a scalar and dividing $\mathbf{A}$ by the same value would artificially decrease the penalization term without changing the *data fitting* term or the shape of the solution.

The *linear constraint* on $\mathbf{s}_k$, on top of the positivity constraint on $\mathbf{A}$, enable us to ensure the positivity of the power estimation (with $\mathbf{u}_0$ being the voltage vector and α_0 a fixed positive parameter). The interpretation of this constraint is that every signature has to be in the same direction as the voltage signal.

Proof. If $\mathbf{A} \geq 0$, $\mathbf{s}_k^\top \mathbf{u}_0 \geq \alpha_0$ and $\alpha_0 > 0$, then: $\mathbf{P}_k(t) = \frac{\mathbf{s}_k^\top \mathbf{u}_0}{N} \mathbf{a}_{k\cdot} \geq 0$ ■

7.2 A full-batch alternating optimization

Let us now derive an efficient algorithm to solve Problem (7.6). Unfortunately the problem is not convex in both variables $\mathbf{S}$ and $\mathbf{A}$. Instead, we propose to use an alternating minimization strategy where $\mathbf{S}$ and $\mathbf{A}$ are updated iteratively resulting in two convex problems [Bertsekas, 1997]. Updating $\mathbf{S}$ results in a least squares problem under quadratic and linear constraints whereas updating $\mathbf{A}$ is a least squares optimization problem with a smooth or non-smooth penalization (depending on the choice of function $\mathcal{G}$) on variations and a positivity constraint. In this section we will use optimization tool such as duality, quasi-newton methods or proximal gradient descents. For more details on these methods, see [Boyd and Vandenberghe, 2004, Nocedal and Wright, 2006, Parikh and Boyd, 2014].

7.2.1 Updating the signatures

We can see that updating $\mathbf{S}$ is equivalent to solving:

$$\begin{aligned} \underset{S}{\text{minimize}} \quad & \| X - \mathbf{S}\,\mathbf{A} \|_{\text{Fro}}^2 \\ \text{subject to} \quad & \| \mathbf{s}_k \|_2^2 \leq 1, \qquad \forall k \in [\![1, K]\!], \\ & \mathbf{s}_k^\top \mathbf{u}_0 \geq \alpha_0, \quad \forall k \in [\![1, K]\!], \end{aligned} \tag{7.9}$$

An illustration of the linear and quadratic constraints is given in Figure 7.1 (each colored area correspond to a constraint). The two areas correspond to each one of the constraints. The intersection of the areas is the feasible set of our constrained problem.

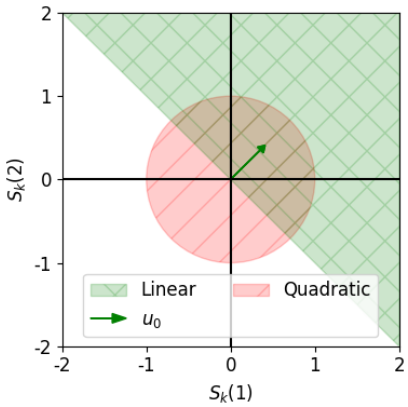


Figure 7.1: Constraints over the columns of $\mathbf{S}$.

We propose to solve the equivalent dual problem, as done for Sparse Coding in [Lee et al., 2007]. Note that our problem is different with the addition of a linear inequality constraint. The convex dual problem is solved using a quasi-Newton method .

The dual formulation of this problem is given by:

$$\begin{aligned} &\underset{\mu,\nu}{\text{maximize}} \quad \inf_{\mathbf{S}} \quad \mathcal{L}(\mathbf{S},\mu,\nu) \\ &\text{subject to} \quad \mu,\nu \in \mathbb{R}_+^K \end{aligned} \tag{7.10}$$

where $\mathcal{L}(\mathbf{S},\mu,\nu) = \|X - \mathbf{S}\,\mathbf{A}\,\|_{\text{Fro}}^2 + \sum_k \mu_k(\|\,\mathbf{s}_k\,\|_2^2 - 1) + \sum_k \nu_k(\alpha_0 - \mathbf{s}_k^\top\,\mathbf{u}_0)$ is the Lagrangian function of problem (7.9) and μ and ν are the dual variables.

The dual function is then given by:

$$\mathcal{D}(\mu,\nu) = \inf_{\mathbf{S}} \mathcal{L}(\mathbf{S},\mu,\nu) = \mathcal{L}(\mathbf{S}^*(\mu,\nu),\mu,\nu) \tag{7.11}$$

and $\nabla_{\mathbf{S}}\mathcal{L} = 0$ leads to: $\mathbf{S}^*(\mu,\nu) =$

$$(X\,\mathbf{A}^\top + \frac{1}{2}\,\mathbf{u}_0\,\nu^\top)(\mathbf{A}\,\mathbf{A}^\top + M(\mu))^{-1} \tag{7.12}$$

where $M(\mu)$ is the diagonal matrix with μ as diagonal. We used the fact that at optimum the derivatives of the Lagrangian with respect to $\mathbf{S}$ equal 0. Consequently the derivatives of the dual function are given by:

$$\forall\, k, \quad \nabla_{\mu_k}\mathcal{D}(\mu,\nu) = \mathbf{S}_k^{*\top}(\mu,\nu)\,\mathbf{S}_k^{*}(\mu,\nu) - 1 \tag{7.13}$$

$$\forall\, k, \quad \nabla_{\nu_k}\mathcal{D}(\mu,\nu) = \alpha - \mathbf{S}_k^{*\top}(\mu,\nu)\,\mathbf{u}_0 \tag{7.14}$$

These gradients are then used in L-BFGS-B [Byrd et al., 1995], a quasi-Newton solver to find optimal values of μ and ν. Finally, the optimal signature matrix $\mathbf{S}^*$ is obtained using equation 7.12. Note that in practice, $\mathbf{S}^*$ is calculated by solving the linear system which is more efficient than computing the inverse.

7.2.2 Updating the activations for smooth regularization

In this section we handle the update of matrix $\mathbf{A}$ when the regularization function is smooth (G_{smooth} as defined in Equation (7.8)). Optimizing over $\mathbf{A}$ while fixing $\mathbf{S}$ is here a non negative least square problem with a smooth non-linear penalization:

$$\begin{aligned} \underset{\mathbf{A}}{\text{minimize}} \quad & \frac{1}{2}\|X - \mathbf{S}\,\mathbf{A}\,\|_{Fro}^2 + \lambda\mathcal{G}_{smooth}(\mathbf{A}) \overset{def}{=} \mathcal{F}(\mathbf{A}) \\ \text{subject to} \quad & \mathbf{A} \geq 0 \end{aligned} \tag{7.15}$$

Since it is a smooth and convex problem with box constraints (7.15), we can also optimize it with L-BFGS-B [Byrd et al., 1995]. The derivatives with respect to every element of the activation matrix ($\mathbf{A}$) are given by:

$$\nabla_{\mathbf{A}_{k,t}}\mathcal{F}(\mathbf{A}) = B_{k,t} + \lambda(\mathbb{1}_{[t>1]}g(\Delta\,\mathbf{A}_{k,t-1}) - \mathbb{1}_{[t<T]}g(\Delta\,\mathbf{A}_{k,t})) \tag{7.16}$$

where $B = -\,\mathbf{S}^\top(X - \mathbf{S}\,\mathbf{A}) \in \mathbb{R}^{K\times T}$ and $g(x) = x/\sqrt{x^2+\epsilon}$.

7.2.3 Updating the activations for non-smooth regularization

The main problem with using non-smooth regularization is the fact that the non differentiability of the function to optimize makes it impossible to use efficient smooth convex optimization tools such as quasi Newton methods. Instead we use an accelerated proximal gradient method [Parikh and Boyd, 2014, Beck and Teboulle, 2009a]. The optimization problem of interest is the following:

$$\begin{aligned} \underset{\mathbf{A}}{\text{minimize}} \quad & \frac{1}{2}\|X - \mathbf{S}\,\mathbf{A}\,\|_{Fro}^2 + \lambda \mathcal{G}_{abs}(\mathbf{A}) \\ \text{subject to} \quad & \mathbf{A} \geq 0 \end{aligned} \tag{7.17}$$

with $\mathcal{G}_{abs}(\mathbf{A}) = \sum_{k,t} |\,\mathbf{A}(k, t+1) - \mathbf{A}(k,t)|$.

In the image processing community, this subproblem is called deblurring with anisotropic total variation [Beck and Teboulle, 2009a]. One minor difference is that our total variation regularization is only on one direction (the time t) of the activation matrix $\mathbf{A}$, whereas in image processing it is calculated on both directions. Eventhough the total variation regularization is separable in the rows of $\mathbf{A}$, the euclidean data fit term makes the overall problem not separable in the general case.

Our algorithm to solve this problem is greatly inspired by [Beck and Teboulle, 2009a] and uses two iterative loops. The full derivation of the algorithm is reserved in Appendix B for ease of lecture of this Chapter.

7.2.4 IVMF algorithms

The choice of the non-linear function $\mathcal{G}$ results in two different algorithms. Our implementation, in Python, uses Scipy's wrapper for L-BFGS-B ([Jones et al., 2001, Byrd et al., 1995]) and our own FISTA implementation [Beck and Teboulle, 2009b]. The alternating optimization algorithm guarantees to decrease iteratively the cost function but since the problem is not convex in both variables simultaneously, the reached solution can only be a local optimum [Bertsekas, 1997]. In such a situation, the solution strongly depends on the initialization of the algorithm, which will be discussed in Section 7.5. It can finally be noticed that the computational complexity of one update of $\mathbf{S}$ and one update of $\mathbf{A}$ are respectively $\mathcal{O}(NK^2)$ and $\mathcal{O}(NKT)$. The complexity of the algorithm is thus driven by the update of $\mathbf{A}$ (since $T \gg K$) but remains linear in time. We summarize the IVMF algorithm in Algorithm 1.

7.3 Preprocessing

Preprocessing the data is an important concept in signal processing and especially in blind source separation. The FastICA algorithm [Hyvarinen, 1999], for instance, works under the *white* assumptions. Let us recall that $X \in \mathbb{R}^{N \times T}$ where N is the dimension of the features (or sensors) and T represents the different samples. The

Algorithm 1: IVMF with alternating minimization

input : X, $\mathbf{u}_0$, λ, I_{max} the maximal number of iterations
Initialize $\mathbf{S}^{(0)}$, $\mathbf{A}^{(0)}$, $\mu^{(0)}$ and $\nu^{(0)}$
for $i = 0$ *to* I_{max} **do**
  /* Smooth A-update, see Section 7.2.2 */
  $\mathbf{A}^{(i+1)} = \text{L-BFGS-B}(\mathbf{A}^{(i)}, \mathcal{G}_{smooth}, \nabla_{\mathbf{A}}\mathcal{G}_{smooth})$
  /* Or Non-smooth A-update, see Section 7.2.3 */
  $\mathbf{A}^{(i+1)} = \text{FISTA}(\mathbf{A}^{(i)}, \mathcal{G}_{abs}, \nabla_{\mathbf{A}}\mathcal{G}_{abs}, \text{prox}_{\mathcal{G}_{abs}})$
  /* S-update, see Section 7.2.1 */
  $\mu^{(i+1)}, \nu^{(i+1)} = \text{L-BFGS-B}(\mu^{(i)}, \nu^{(i)}, \mathcal{D}, \nabla_{\mu_k,\nu_k}\mathcal{D})$
  /* Solve the linear system */
  $\mathbf{S}^{(i+1)} = (X\mathbf{A}^{(i+1)^\top} + \frac{1}{2}\mathbf{u}_0\,\nu^{(i+1)^\top})(\mathbf{A}^{(i+1)}\,\mathbf{A}^{(i+1)^\top} + 2M(\mu^{(i+1)}))^{-1}$

first assumption is the centering: each rows of X has a zero mean. The second assumption is the whitening: the covariance matrix of X is the identity matrix. It means that the variance of each row is one and that each row is decorrelated from an all other rows. This assumption is essential for ICA since independence implies decorrelation. Ensuring decorrelation of the estimated sources is simple from white data. It results in the constraint that the unmixing matrix is orthogonal. However, the centering assumption is non compatible with the nonnegative constraints of the sources or activations. In methods like Nonnegative ICA, the preprocessing is reduced to only whitening and the centering step is skipped.

We propose here a new whitening procedure. The data X is transformed so that the covariance matrix of the *variations* of the new data (as defined in the regularization function of IVMF) is the identity matrix.

$$\tilde{X} = WX \tag{7.18}$$
$$\text{such that:} \quad \text{Cov}\left[\tilde{X}D\right] = I_N$$

where D is a sparse $T-1 \times T$ real matrix with non zero values defined by: $\forall t \in [\![1, T-1]\!]$, $D_{t,t} = -1$ and $D_{t,t+1} = 1$. D is the operator to calculate the variations.

Lemma 7.3.1. *If: (i)* $Cov[XD] = XDD^\top X^\top = E\Delta E^\top$, *(ii)* $W = \Delta^{-\frac{1}{2}}E^\top$ *and (iii)* $\tilde{X} = WX$; *then:* $Cov\left[\tilde{X}D\right] = I_N$.

Proof. Using assumptions (i), (ii) and (iii), one can write the covariance matrix of

$\tilde{X}D$ as follows:

$$\begin{aligned}
\text{Cov}\left[\tilde{X}D\right] =& \tilde{X}DD^\top\tilde{X}^\top && (7.19)\\
=& WXDD^\top X^\top W^\top && (7.20)\\
=& \Delta^{-\frac{1}{2}}E^\top E\Delta E^\top E\Delta^{-\frac{1}{2}} && (7.21)\\
=& \Delta^{-\frac{1}{2}}\Delta\Delta^{-\frac{1}{2}} && (7.22)\\
=& I_N && (7.23)
\end{aligned}$$

■

7.4 Probabilistic Interpretation

Like ICA, IVMF has a probabilistic interpretation. Indeed, it can be seen as the Maximum A Posteriori (MAP) estimation of a probabilistic model:

$$x_t = \mathbf{S}\,\mathbf{a}_t + \Gamma_t \tag{7.24}$$

where $x_t \in \mathbb{R}^N$ are the observations and $\mathbf{S} \in \mathbb{R}^{N\times K}$ and $\mathbf{a}_t \in \mathbb{R}^K$ the random variables of interest. $\Gamma_t \in \mathbb{R}^N$ are independent and identically distributed Gaussian variables with covariance I (the identity matrix). $\mathbf{S}$ is a random matrix whose columns follow a uniform prior distribution over the closed convex set denoted $\mathcal{C}_S$ and defined by the intersection of $\{\mathbf{s}_k \in \mathbb{R}^N |\ \|\mathbf{s}_k\|_2^2 \leq 1\}$ and $\{\mathbf{s}_k \in \mathbb{R}^N |\ \mathbf{s}_k^\top \mathbf{u}_0 > \alpha_0\}$. $\mathbf{a}_t$ are positive random vectors whose joint prior density is proportional to $e^{-\lambda\mathcal{G}(A)}\mathbb{1}_{\{\mathbf{A}\geq 0\}}$ ($\mathbb{1}$ is the indicator function).

The Log-posterior of this model writes down:

$$\begin{aligned}
\mathcal{L}(\mathbf{S},\mathbf{A}\,|X) &= \log p(X|\,\mathbf{S}\,\mathbf{A}) + \log p(\mathbf{S}) + \log p(\mathbf{A}) - \log p(X) && (7.25)\\
&= -\frac{1}{2}\|X - \mathbf{S}\,\mathbf{A}\,\|_{Fro}^2 - \lambda\mathcal{G}(\mathbf{A}) - \log(\mathbb{1}_{\{\mathbf{A}\geq 0\}})\\
&\quad - \sum_k \log(\mathbb{1}_{\{\mathbf{s}_k\in\mathcal{C}_S\}}) - cst
\end{aligned}$$

The MAP estimators are then given by:

$$\hat{\mathbf{S}}_{map}, \hat{\mathbf{A}}_{map} = \underset{\mathbf{S},\mathbf{A}}{\text{argmax}}\ \mathcal{L}(\mathbf{S},\mathbf{A}\,|X) \tag{7.26}$$

which is an equivalent problem than the one expressed in Equation (7.6).

7.5 Experimentations

The goal of this section is to investigate the practical behavior of our algorithm. We are interested in testing the ability to recover the true factors from an observed matrix and also in analyzing the rate of convergence of the algorithm. Our algorithm includes one parameter: the regularization coefficient λ. We will test the influence of this parameter on the results of the algorithm. In this section we do not address the linear constraint coefficient α_0 as a parameter.

In this section, we simulate data that follows the factorization model: $X = \mathbf{S}\,\mathbf{A} + \Gamma$, where $\mathbf{A} \in \mathbb{R}_+^{K \times T}$ is the matrix of positive activations, $\mathbf{S} \in \mathbb{R}^{N \times K}$ is the real mixing matrix named *signature* and $\Gamma \in \mathbb{R}^{N \times T}$ is an additive white noise (each entry of Γ is independent and identically distributed). The simulation of $\mathbf{A}$ is as follows: (i) draw independent and identically distributed Laplace variable corresponding to the each entry of the variations of $\mathbf{A}$, (ii) [optional] set a certain percentage of these values to zero to add true sparsity, (iii) cumulate along the time dimension to calculate $\mathbf{A}$, (iv) add to all elements of $\mathbf{A}$ its minimum value along each row to ensure positivity. The entries of $\mathbf{S}$ are simulated independently from a normal distribution and then adjusted so that the ℓ_2-norm of each column equals 1. Γ is also simulated from a normal distribution. Finally, X is deterministically computed from $\mathbf{S}$, $\mathbf{A}$ and Γ.

In a first experiment, we fix the noise variance to 0, set the sparsity to 0.8 (80% of the variations of $\mathbf{A}$ are zero), simulate 100 datasets and then run ICA, SNMF and IVMF. We repeat this procedure with a sparsity of 0 (meaning that no variation is forced to be 0). Finally, we set the noise variance to 1. The size of the simulation is $K = N = 4$ and $T = 50$.

For IVMF, we use the whitening preprocessing described in Section 7.3 and we vary the λ parameter from 0 to 0.1. The ϵ parameter used in the regularizer $G_{smooth}(x)$ is set to 10^{-7}. For ICA we use FastICA implementation found in the Python package Scikit-Learn [Pedregosa et al., 2011]. We used the standard *logcosh* cost function. We apply it to a transformation of the data that computes the time variations: $\tilde{X}(n,t) = X(n,t) - X(n,t-1)$. The optimal $\mathbf{S}_{ica}$ is then used to recover the optimal $\mathbf{A}_{ica}$ from the original data X with $\mathbf{A}_{ica} = \mathbf{S}_{ica}^{-1} X$. For SNMF, we used our own Python implementation that uses a non-negative least square solver to update the activation matrix $\mathbf{A}$. This approach appeared to be faster in practice than the original implementation of [Ding et al., 2010].

To estimate the performance of the algorithms we use the estimation error on $\mathbf{A}$ defined as: $E_{\mathbf{A}} = \| \mathbf{A} - \hat{\mathbf{A}} \|_{Fro}^2 / \| \mathbf{A} \|_{Fro}^2$, where $\hat{\mathbf{A}}$ is the estimated matrix. To fix the multiplicative ambiguity we normalize the output of the different algorithms so that

the ℓ_2-norm of the columns of $\mathbf{S}$ equals 1, especially for SNMF and ICA as IVMF already requires this feature. To fix the permutation ambiguity, we reorder the rows of $\hat{\mathbf{A}}$ by using a greedy algorithm to match each row of $\mathbf{A}$ with its closest row in $\hat{\mathbf{A}}$. We present this simple greedy algorithm in the Chapter 8.

7.5.1 A first decomposition example

Let us start with a visual inspection of one run on a sparse and noiseless factorization. For IVMF, we have used the smooth regularizer (Equation (7.8)), the whitening preprocessing and a regularization parameter $\lambda = 0.01$. Figure 7.2 illustrates that IVMF succeed in estimating the true factors, whereas ICA is a bit worse and SNMF fails completely. To be more specific, it illustrates the identified drawbacks of ICA and SNMF. We can see for instance that ICA activations may be negative (on the third one) and that SNMF activations are too much dependent (it is especially visible during the first 5 points of the activations: although only ground truth present a decreasing shape, all the activation estimated by SNMF present a decreasing shape). In the following sections we continue this exploration in a more quantitative way.

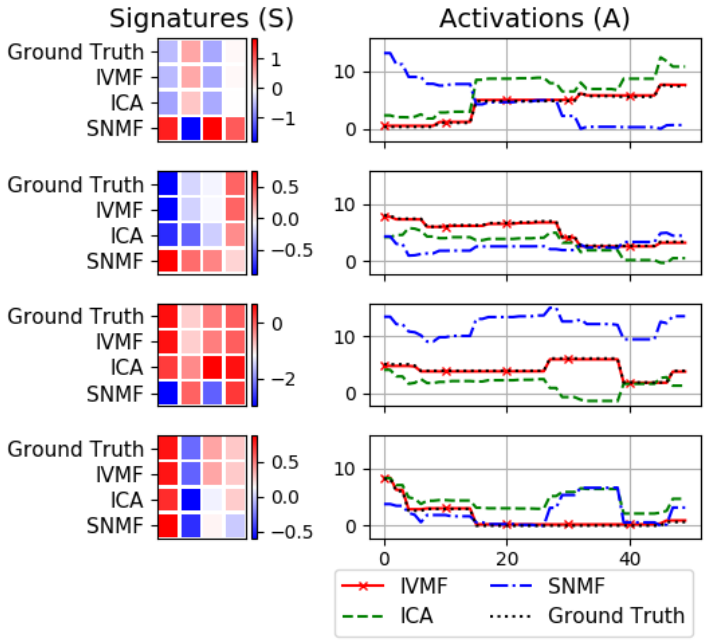
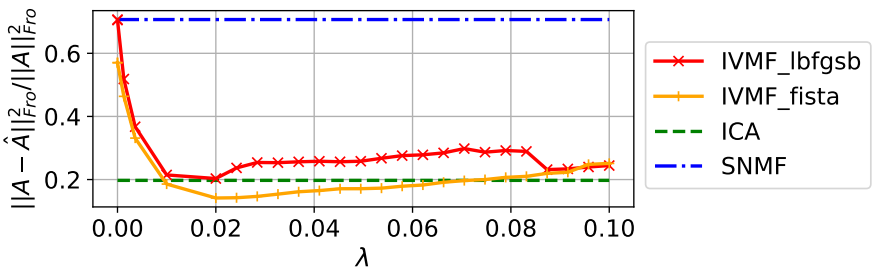


Figure 7.2: Example of a *sparse* simulation: the result of IVMF, ICA and SNMF are presented component by component: each one of the 4 rows of plot in the figure corresponds to one columns of the signature matrix **S** and the corresponding row in the activation matrix **A**. Each row of the figure is then composed of one heat-map plot (on the left) for the signature and a classic line chart (on the right). The heat-map makes the visual comparison between algorithm easy as each row inside the plot corresponds to a different algorithm.

7.5.2 Identifiability

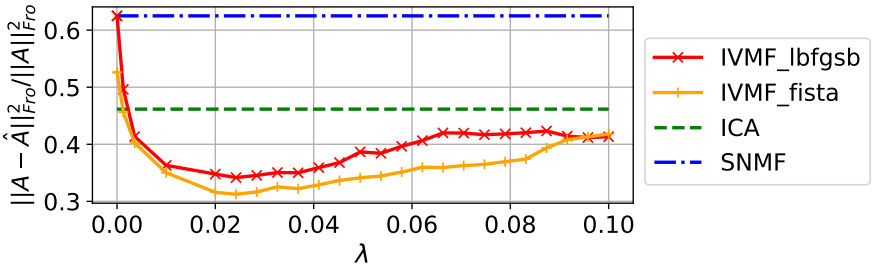
In this section, we want to quantify the ability of IVMF to recover the true factors, this concept is called identifiability. For IVMF, we compare here the two different regularization proposed: the smooth one denoted *IVMF lbfgsb* and the non smooth one denoted *IVMF fista*. Figures 7.3 and 7.4 present the average error over the 100 simulations for the 3 generative procedures described previously (not sparse, sparse, not sparse and noisy). This particular example first shows that IVMF consistently outperforms SNMF. IVMF's performance also improve as λ decreases until an optimal value. For lower values the performance deteriorates until reaching those of SNMF when $\lambda \approx 0$. Figure 7.3(a) and 7.4(a) present the result of simulations without noise and IVMF outperforms ICA for λ values in the range $[0.01, 0.05]$. We may explain this by the fact that ICA does not take advantage of the positivity property

of the signal and also that ICA enforces perfect decorrelation of the sources which in practice is not the case. We can notice that IVMF performs better on the sparse example than on the not sparse one. Figure 7.4(b) shows that IVMF is more robust to additive noise than ICA. For the noisy setting IVMF's performance seems to be less sensitive to the choice of the regularization parameter λ. To compare the two IVMF's implementation, we can say the global behavior is similar for the two of them. In the noiseless scenario, the non smooth regularizer seems to have a small advantage.

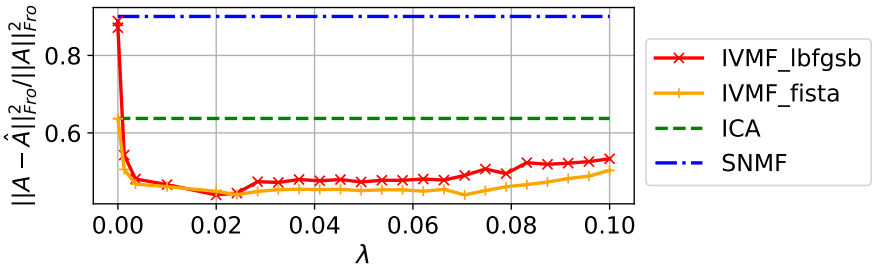


(a) sparsity = 80 %

Figure 7.3: Influence of sparsity level, noise and regularization parameters (1/2): comparison of IVMF, ICA and SNMF. Each plot presents the average error over 100 different scenario as a function of the regularization parameter λ. For ICA and SNMF, its represented as a constant line because there is no parameter.



(a) sparsity = 0%



(b) noise = 1 & sparsity = 0%

Figure 7.4: Influence of sparsity level, noise and regularization parameters (2/2): comparison of IVMF, ICA and SNMF. Each plot presents the average error over 100 different scenario as a function of the regularization parameter λ. For ICA and SNMF, it is represented as a constant line because there is no parameter.

7.5.3 Correlation

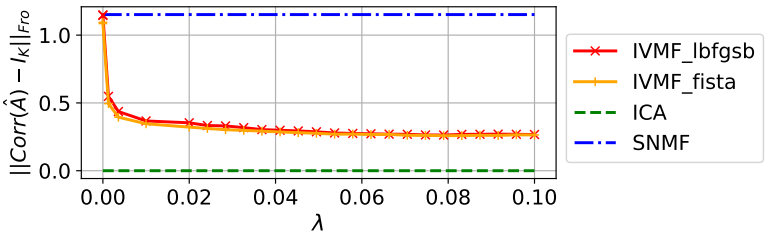
To understand why IVMF outperforms ICA and SNMF, we propose here to explore the influence of each method on the correlation of the variations of the estimated activations, i.e. the correlation between the rows of $\hat{\mathbf{A}}$. The metric used in the following is the the sum of squared non diagonal elements of the correlation matrix of the variations of activations, formally defined as: $\sum_{k,k'} \left(\text{Corr} \left[\Delta\hat{\mathbf{A}}_k, \Delta\hat{\mathbf{A}}_{k'} \right] - \delta_{k,k'} \right)^2$.

Recall that we use the FastICA implementation of ICA [Hyvarinen, 1999]. This algorithm when used with a whitening preprocessing ensures the estimated sources to be decorrelated. As we have applied ICA to the variations of X, the correlation of the resulting activations shall be zero. Figure 7.5 shows exactly this property.

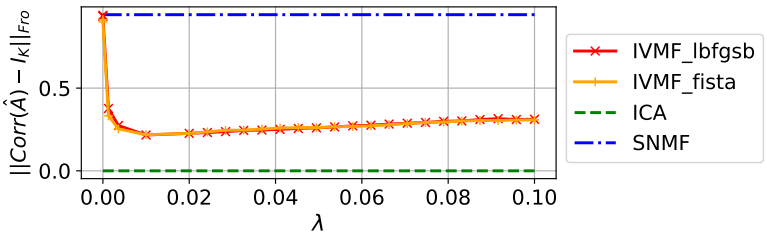
On contrary, SNMF impose no particular structure on the correlation of the activations. As mentioned earlier, SNMF seems to imply a relatively high value of correlation regarding other methods.

As for IVMF, Figures 7.5(a) and 7.5 show the regularization has the effect of

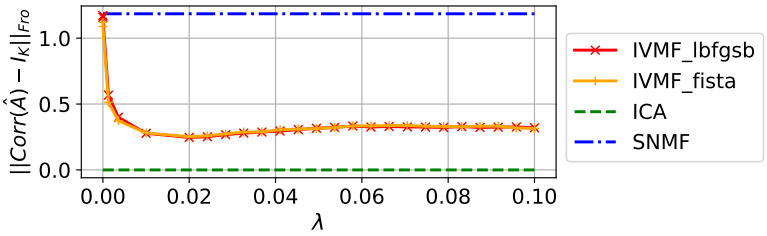
reducing the correlation. For sub-figures 7.5(b) and 7.5(c), from $\lambda = 0.1$ down to ≈ 0.01, decreasing λ resulting in more decorrelated estimations. This argument is in favor of IVMF to induce independence. Note however that decorrelation is not a sufficient condition for independence. [Feng and Kowalski, 2018] showed a theoretical result reinforcing this idea. Indeed, they showed that under the whitening preprocessing of data, a square signature matrix **S**, Sparse Coding solution (with a specific regularization) tends to FastICA's solution when the regularization parameter tends to 0. If we discard the positivity constraint and the linear inequality constraint, the result also applies to IVMF. This is another argument suggesting that IVMF can help identifying independent components which are positive. This can also be linked to the probability interpretation of IVMF as a MAP inference (Section 7.4).



(a) sparsity = 80 %



(b) sparsity = 0%



(c) noise = 1 & sparsity = 0%

Figure 7.5: On the correlation of the variations of activations. The metric used is the the sum of squared non diagonal elements of the correlation matrix of the variations of activations.

7.5.4 Convergence rate

In this section we study the convergence rate in practice of IVMF. Figure 7.6 shows the evolution of the estimates along the algorithm iterations, where one iteration consists of one update of **S** and one update of **A**. Figure 7.6, shows the evolution of the cost function (minimized by our algorithm) and the evolution of the estimation error on the activation matrix **A**. The cost function starts by decreasing sharply on the first iterations and then the rate slows down until iteration ≈ 13000. During this period, the error on **A** however has sharply decreased. After iteration 13000, both the cost function and the estimation error decrease very slowly.

Figure 7.7 shows the evolution of the solutions at different point of the convergence. After *only* 6000 iterations (sub-figure 7.7(a)) the solution is already acceptable (approximately as good as ICA in sub-figure 7.7(d)). After 13000 iterations (sub-figure 7.7(b)) the solution is almost perfect (better than ICA). Until iteration 50000, the solution does not change much to finally achieve perfect separation (sub-figure 7.7(c)). This slow convergence rate near the optimum is a drawback of alternating minimization.

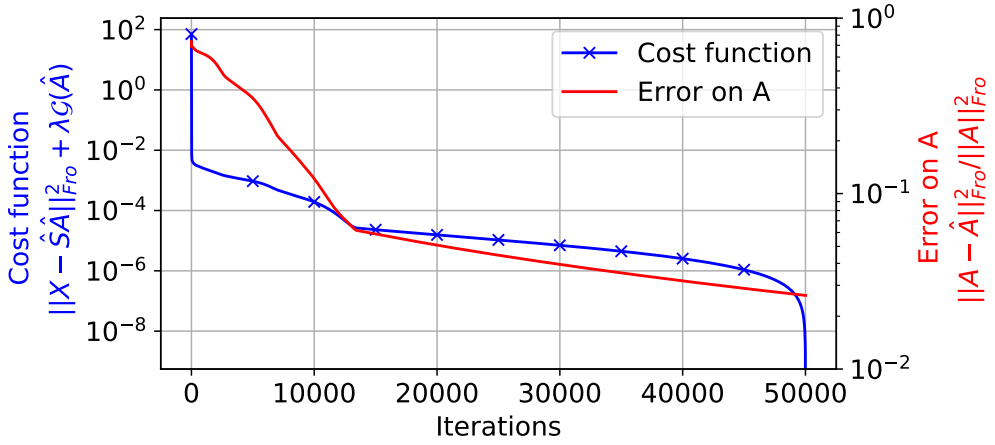
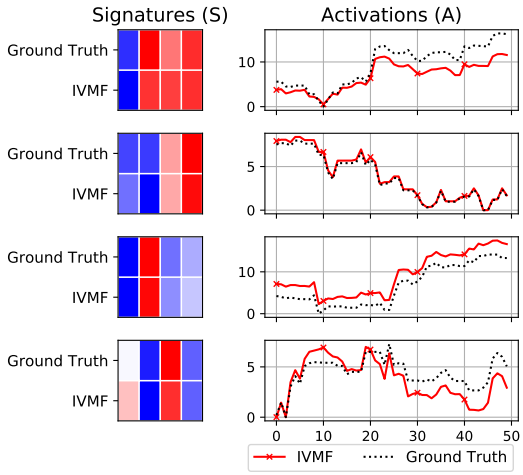
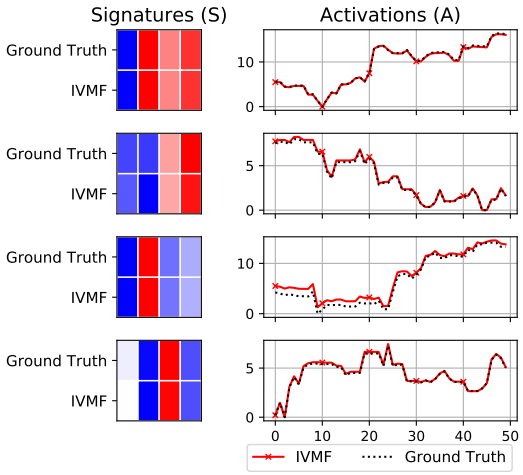


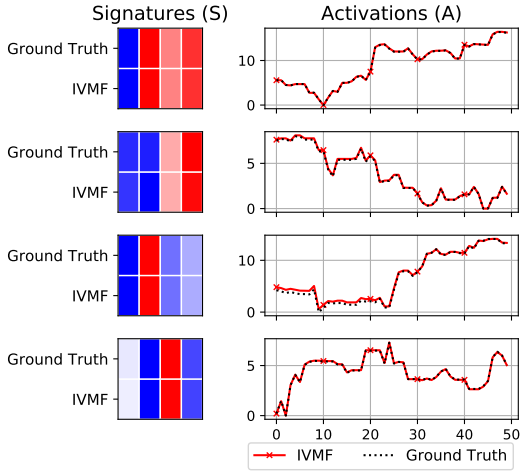
Figure 7.6: Convergence of IVMF on simulated data (sparsity=30%, noiseless). The cost function is on the left axis and error on activation **A** is on the right axis, both on log scale.



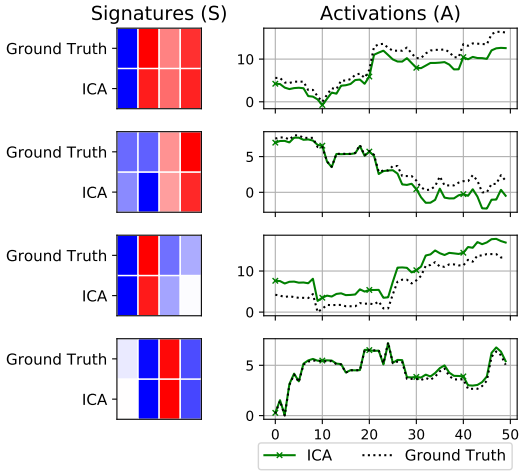
(a) IVMF after 6000 iterations



(b) IVMF after 13000 iterations



(c) IVMF after 50000 iterations.



(d) ICA

Figure 7.7: Evolution of solutions: IVMF on simulated data.

7.6 Conclusion

To conclude this chapter, we have have proposed a novel matrix factorization technique called IVMF. It is formulated as an optimization problem and we have designed an associated optimization algorithm for the separation of time dependent sources whose variations exhibit independence or sparsity. The proposed approach extends SNMF by introducing a physically-inspired regularizer over the variation of the sources and linear and quadratic constraints on the signature matrix. Note that we have proposed two different regularizers: one smooth and thus differentiable everywhere and one non smooth not differentiable on 0. We have also shown that our approach can achieve independent source separation and outperforms its natural competitors on toy examples.

In Chapter 8, we will use IVMF to solve NILM problems on three publicly available datasets. The first one, the main interest of this dissertation, concerns big systems such as commercial buildings. The last two datasets consist of residential houses.

Chapter 8

Disaggregation Results

Contents

We have just developed a new method called IVMF for addressing the NILM software problem using high frequency current measurements. IVMF has been designed for the special case of NILM in big systems such as commercial buildings. We will therefore analyze the performance of IVMF on a synthetic commercial buildings dataset called SHED. The lack of real and high frequency data for commercial buildings forces us to only use synthetic dataset for the evaluation of our method. However, real and high frequency data are available for residential buildings. We will then test the performance of IVMF on two other real datasets (REDD, BLUED).

As explained before our unsupervised approach address the only the disaggregation part of the NILM software problem and not the identification (or classification) of the electric device. We will then present the procedure to calculate the IVMF performance in this setting.

8.1 Evaluation Metrics

Evaluating the performance of NILM algorithm is a complicated task due to a lack of standardization [Pereira and Nunes, 2018]. In this section we present the metric used to evaluate the performance of our NILM algorithm. We will use the *disaggregation error* or power error, defined as:

$$E_{\text{power}} = \sum_c \| \mathbf{P}_c - \hat{\mathbf{P}}_c \|_2^2 / \sum_c \| \mathbf{P}_c \|_2^2, \tag{8.1}$$

where $\mathbf{P}_c \in \mathbb{R}_+^T$ is the ground truth (given by the dataset) and $\hat{\mathbf{P}}_c \in \mathbb{R}_+^T$ is the corresponding estimation.

As explained before, we want to evaluate algorithms that can disaggregate the total consumption into sub-component representative of individual devices. Thus, the studied algorithm do not directly identify the *label* of the subcomponent. To be able to estimate the disaggregation error we need an identification step before computing it. In the following we use a greedy identification algorithm. It starts by finding the couple $(\mathbf{P}_i, \hat{\mathbf{P}}_j)$ that has the lowest distance ($\| \mathbf{P}_i - \hat{\mathbf{P}}_j \|_2^2$). Indexes i and j are then discarded from the ground truth indexes and respectively the estimations indexes and a new couple is identified. We repeat this procedure until all the ground truth component have been affected to an estimated sub-component. Note that the association between the ground truth and the estimations components are *one to one*, meaning that we do not allow for re-aggregation of several sub-component (by summing them) to best match a ground truth component. This kind of re-aggregation have been proposed in [Lange and Bergés, 2016] but from our point of view may bias too much the result. Indeed, with such a multiple re-aggregation procedure, algorithm that tends to *over disaggregate* the consumptions may be privileged.

8.2 Results on public datasets

8.2.1 SHED

In this section, we consider a NILM dataset called SHED introduced in Chapter 4. This dataset contains simulated current measurements for 8 commercial buildings and the corresponding individual power consumptions. These simulations have shown to be realistic for the NILM task (Section 4.3.3). We can also notice that the simulation process involves a factorization structure of the current data, which is

more complex. We use a sample of this dataset corresponding to 1 day of current waveforms averaged at 5 minutes. The number of individual devices or categories ranges from 5 to 10 depending on the building. In this dataset, the voltage is periodic: $\forall\, t,\ \mathbf{U}(n,t) = \mathbf{u}(n)$.

The experiment consists in running the ICA, SNMF and IVMF algorithms with 50 different initializations. The number of component is set to $K = 10$, to make sure we have more sub-components than the true number of categories in the building. The dimensions of the problem are thus $N = 100$, $K = 15$ and $T = 288$. For IVMF, we use the smooth regularization and chose to fix manually the λ parameter to a low value of 0.01 and α_0 to 0. For the number of iterations, we use 5000 global iterations, 200 iterations maximum per signature update, and 10 maximum iterations for the activation matrix update. ICA is run on the derivatives of the data $(X_t - X_{t-1})$ as it was done in Section 7.5. SNMF is used in the same conditions than IVMF in terms of number of iteration.

The power error results given in Table 8.1 show that IVMF presents the best performance on all buildings but one (building 3) where no method seems to perform well (due to confusions between 2 categories and the presence of important constant power consumptions, see Appendix A). We can note that, the result for ICA are, on average, comparable to IVMF for the disaggregation error. This can be interpreted by the fact that the result of ICA seems to be less sensitive to initialization than IVMF.

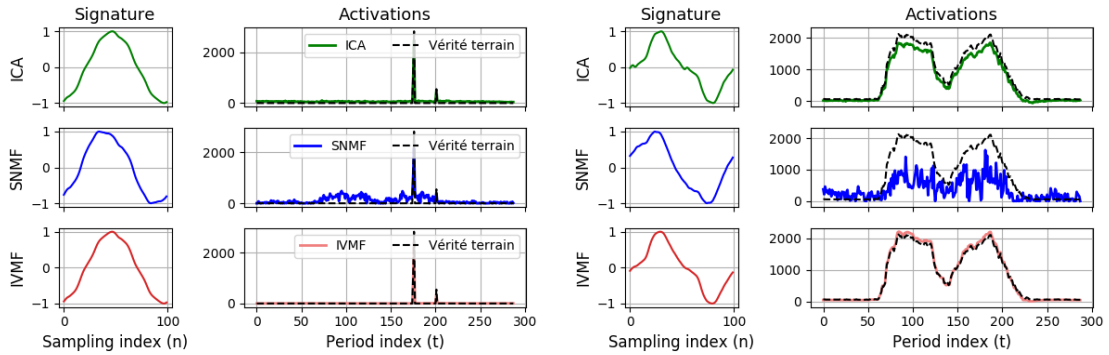
Table 8.1: Performance comparison between ICA, SNMF and IVMF on the NILM task using the SHED dataset.

Building	ICA		SNMF		IVMF	
	Avg.	Best	Avg.	Best	Avg.	Best
1	0.39	0.19	0.50	0.26	0.26	**0.12**
2	0.53	0.48	0.67	0.46	0.50	**0.39**
3	0.52	0.47	0.78	**0.41**	0.76	0.57
4	0.65	0.48	0.81	0.52	0.70	**0.31**
5	0.53	0.44	0.77	0.45	0.70	**0.43**
6	0.68	0.53	0.65	0.46	0.60	**0.45**
7	0.24	0.20	0.46	0.26	0.38	**0.19**
8	0.54	0.45	0.67	0.37	0.61	**0.32**

Avg (resp. Best) refers to the average (resp. minimal) error over 50 different runs of the algorithm.

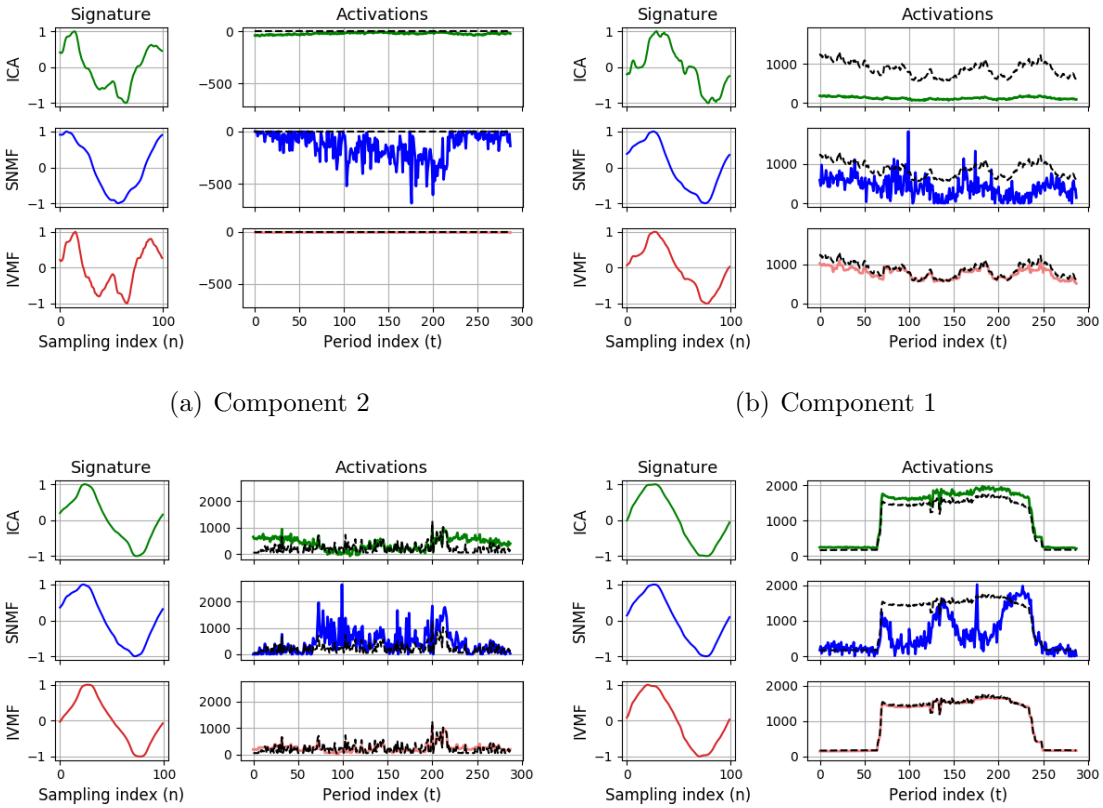
Although the disaggregation error is interesting to quantify an overall performance, it lacks of information concerning the desired property of the power estimations, namely, the positivity, the interpretability of the signature or the ability to separate small sources. To compare the performance of algorithms on these fundamental qualitative property, a closer look at the results plots is needed. To do so, we provide in Figures 8.1 and 8.2 some decomposition examples that illustrate the nice properties of the power consumptions estimated by IVMF. Figure 8.1(a) illustrates that IVMF is better to recover activations with sparse variations: SNMF and ICA present small variations when IVMF present no variation. Figures 8.1(b) and 8.2(a) present an activation with negative values for ICA whereas IVMF estimates them perfectly. In Figure 8.2(a), we observe completely negative power consumptions for SNMF. It is due to the fact that even if the activation is positive the sign of the estimated power consumption is given by the scalar product of the signature and the voltage waveform. The positivity of estimated power consumption is ensured in IVMF by an inequality constraint on the signature as explained in Chapter 7. Figure 8.2(b) shows that even though ICA and IVMF estimates similar activations the associated signature can be different. Finally, Figure 8.2(c) shows one category that is not recovered by ICA.

Further results (Buildings 2 and 6) on the SHED dataset are given in Appendix C.



(a) Component 4 (b) Component 3

Figure 8.1: IVMF, SNMF and ICA disaggregation results on SHED building 1 (1/2). Each sub-figure presents the estimation of one component (signature and calculated power consumption) and the corresponding ground truth.



(a) Component 2

(b) Component 1

(c) Component 5

(d) Component 6

Figure 8.2: IVMF, SNMF and ICA disaggregation results on SHED building 1 (2/2). Each sub-figure presents the estimation of one component (signature and calculated power consumption) and the corresponding ground truth.

8.2.2 REDD

The REDD dataset [Kolter and Johnson, 2011] is made of High Frequency (HF) mains current measurement and Low Frequency (LF) measurement of individual equipment or circuit for 2 out of 5 houses (house 3 and house 5). The HF data are compressed and consists of waveforms representing the current during one voltage period on 275 points for the 2 current phase lines. Each waveform is considered to repeat for a certain variable number of period given in the dataset. The LF data are power readings measured every ≈ 3 seconds.

We have selected one day of data of house 3: 23$^{\text{rd}}$ May 2011. The HF data has been down sampled to 5 minutes, meaning that we now have one waveform of 275 point per 5 minutes. The LF data has also been down sampled by averaging to fit

to the HF data. We can note that the sum of the LF measurements of individual equipment almost equal the total power consumption given by the HF data. Figure 8.3 present the small difference. The biggest difference resides in the recurrent spikes present in the HF data and smoothed in the LF data. Our supposition is that these spikes come from the refrigerator and are at a too high frequency for being measured in the LF data.

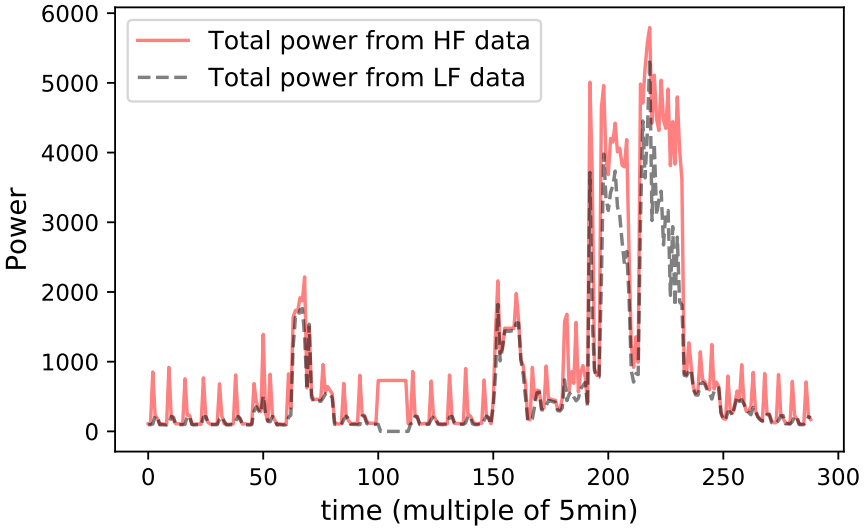


Figure 8.3: Total power consumption of the REDD house 3, resampled at 5 minutes.

We have run IVMF, ICA and SNMF have all been run using 10 components ($K = 10$). For IVMF we have used the non-smooth regularizer with 5000 global iterations, 200 iterations maximum per signature update, and 10 maximum iterations for the activation matrix update. The regularization parameter has been set to $\lambda = 0.001$. Note that we have used the non-smooth regularizer because in practice it seems to reach sparser results than the smooth regularizer. Residential buildings consumptions are also known to have sparser variations that commercial buildings one.

The following figures present the best results over 50 runs for the 3 algorithms. We present estimated power consumption with their respective signatures and compare it to the true power consumptions given by the dataset. Figure 8.4 shows the 6 most consuming equipments and the residual (not affected) power consumptions. The most interesting thing is that on top of having *good* estimation of the power consumption, the *signature* of each component is highly interpretable. For instance, the estimated signature of the microwave is very characteristics of such equipment.

Indeed, in Chapter 3 we have showed a very similar microwave waveforms which have been measured. The two first components have a *resistive* signature. Each of them have been estimated on a different phase line (A and B) and corresponds to the group of washer dryer/dishwasher/bathroom circuit(GFI). Our interpretation is that the washer dryer is a bi-phased equipment (an equipment plugged on both phases that *consume* power on each phase line). The component corresponding to the refrigerator present a signature characteristics of motor. The last two components have similar signatures that correspond to electronic power supplied devices.

ICA and SNMF results are given in Figure 8.5 and 8.6 for comparison. Even if ICA manage to recover two *resistive* signatures, the associated power estimation poorly corresponds to the ground truth and can be negative.

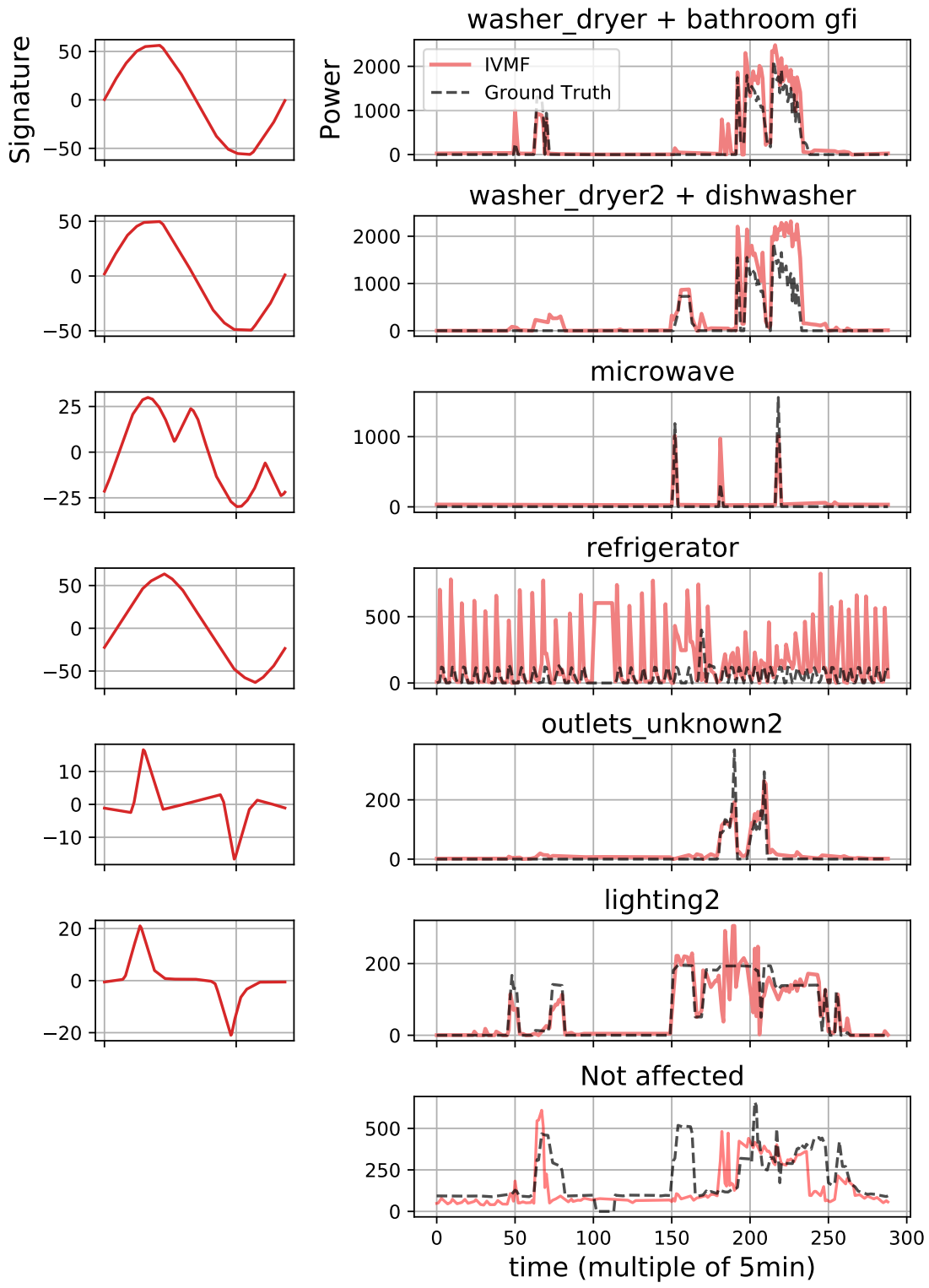


Figure 8.4: IVMF disaggregation of REDD house 3.

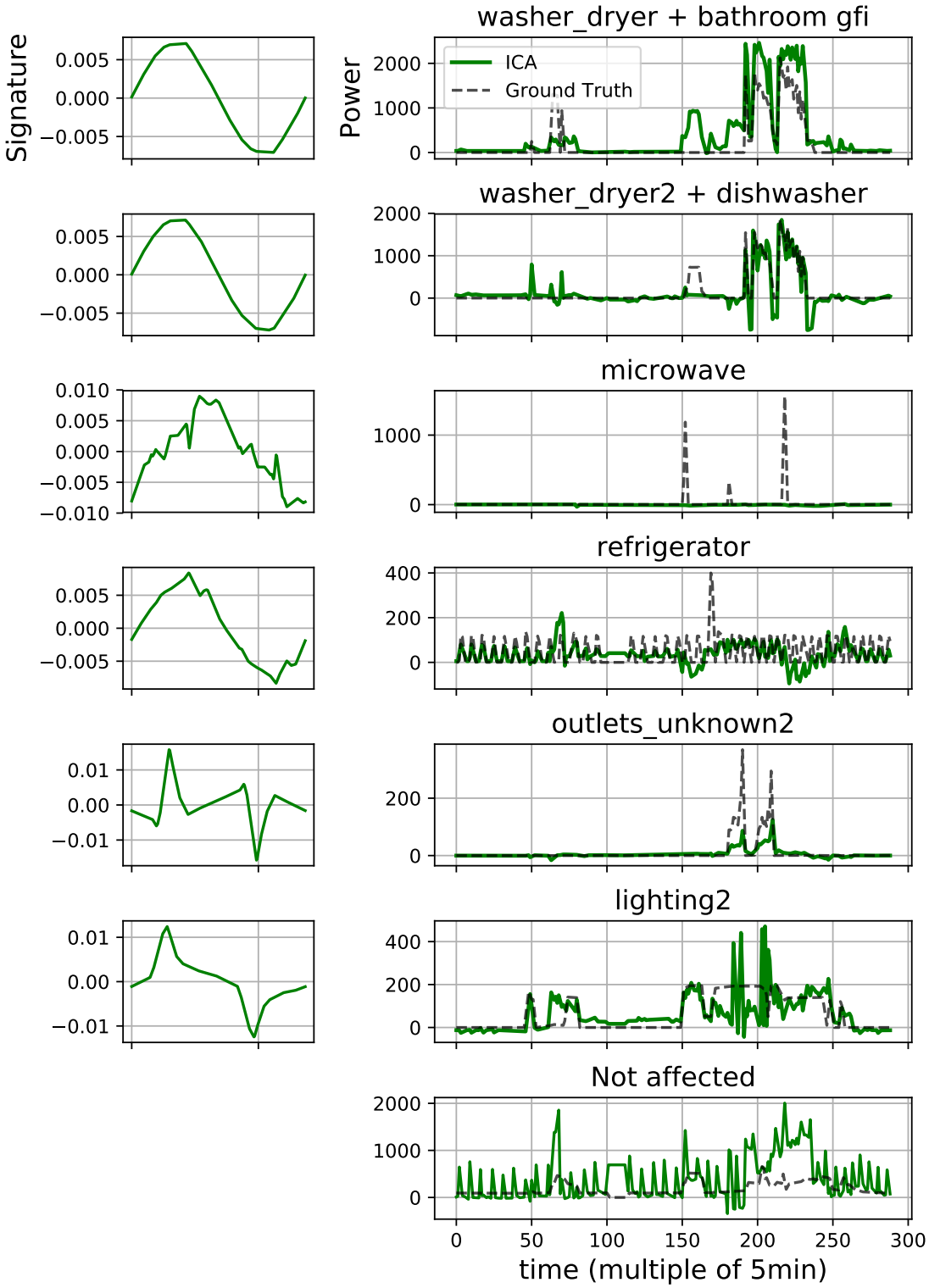


Figure 8.5: ICA disaggregation of REDD house 3.

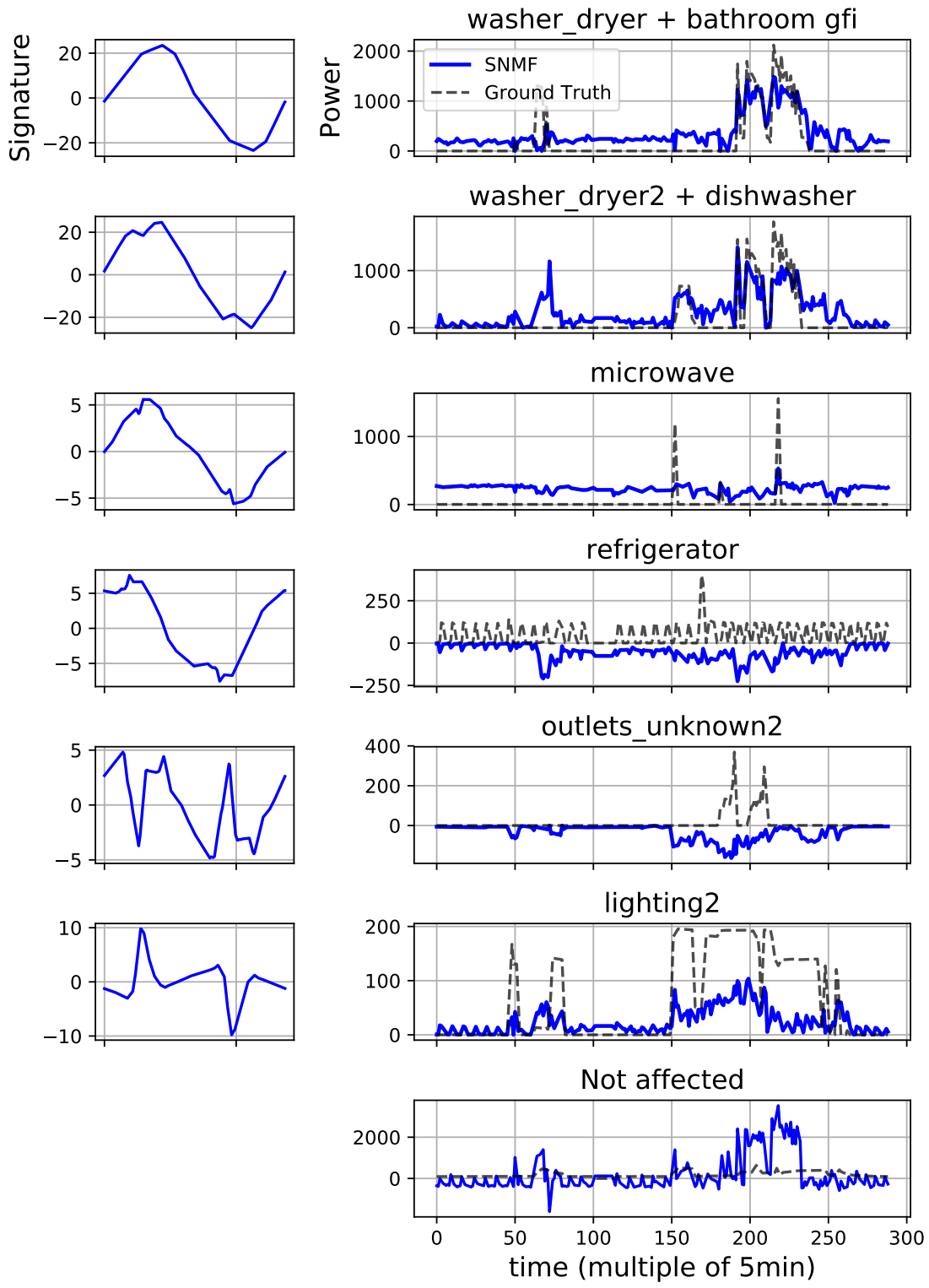


Figure 8.6: SNMF disaggregation of REDD house 3.

8.2.3 BLUED

In this final experiment we apply on IVMF on another high frequency current dataset called BLUED [Filip, 2011]. The data was collected at a residential building in Pittsburgh, USA during October 2011 with a sampling frequency of 12kHz. We used the same data processing as explained in the previous Section 8.2.2. Unfortunately the BLUED dataset do not include as ground truth the power consumption per electric device (see Figure 8.7). This is the reason why we just present in Figure 8.8 raw result from IVMF without ground truth comparison. To differentiate from the previous section, we have run IVMF over one week of data instead one only one day.

It is still interesting to see that the same kind of results are obtained. We can for instance distinguish a resistive signature, a microwave, electronic fed devices.

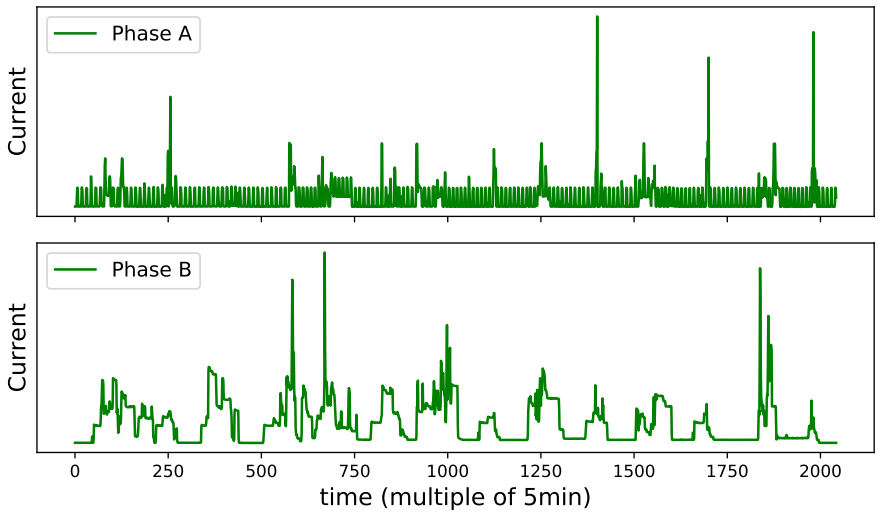


Figure 8.7: Total power consumption of the BLUED, resampled at 5 minutes.

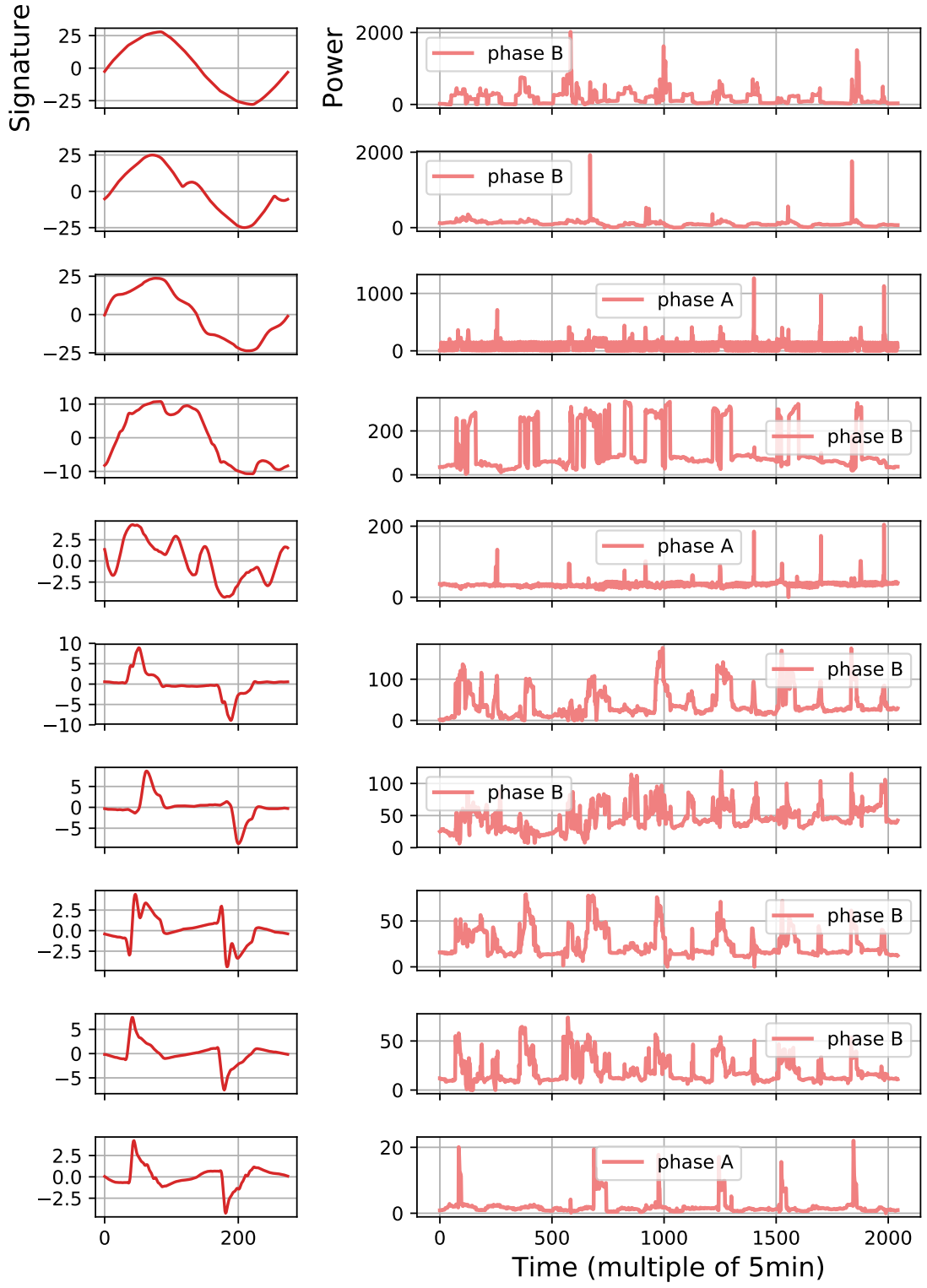


Figure 8.8: IVMF disaggregation of BLUED.

8.3 Conclusion

We have just seen that our proposed IVMF approach outperforms ICA and SNMF on the SHED dataset which is a synthetic dataset reproducing the behavior of big systems such as commercial buildings. The performance has been evaluated using a global metric (the disaggregation error) and using plots of the estimated power consumptions. The global metric showed that IVMF achieve better results than SNMF and ICA. However, the average result of ICA and IVMF are comparable. A closer look at the results plots, revealed that IVMF present fundamental properties: (i) it estimates always non-negative power consumptions and (ii) it can estimate power consumption with sparse variations.

To the best of our knowledge, this is the first time a NILM algorithm achieve such results on a disaggregation NILM software problem for commercial buildings.

Concerning the observed drawbacks of IVMF, it appears that the result is more sensitive to the initialization than ICA. TWe could think of using ICA to initialize IVMF. We can also note that in terms of processing time, ICA is faster than IVMF. This is mainly due to the fact that ICA ensures perfect reconstruction of the data. Contrary, we think that the perfect reconstruction constraint in ICA can also be seen as a major drawback especially when the observed data do not follow exactly the ICA model.

Finally, we recall that IVMF has been developed to address the limitations of NILM algorithms focusing on smaller systems such as residential buildings, mainly due to the assumption of constant power consumption of electric devices. Preliminary results on residential datasets indicate that IVMF can also be very efficient in this scenario. Its ability to estimate consumptions with sparse variations seems to be very useful for residential buildings.

Conclusion and Perspectives

In this thesis we have addressed the Non Intrusive Load Monitoring problem with a particular attention to big systems such as commercial buildings using High Frequency current and voltage measurements. We have defined the NILM problem as a *source separation* problem.

In order to address the lack of knowledge, we have proposed an extensive data analysis on public and private datasets. The first result of our analysis is the **low rank assumption**. Indeed we showed that the current matrix of any electric devices admits a *good* approximation whose rank is lesser or equal to 5. We then showed how to construct such approximations using Matrix Factorization techniques. This factorization analysis conducted to the definition of a new device taxonomy reflecting the complexity of the source/device.

The second result of our analysis is the design of metrics that can discriminate between residential and commercial buildings data. We carefully studied the variations of the power consumption which has been an important quantity in the rest of our work. Their time correlation and distribution appeared to be very different and accredited two hypotheses: (i) consumptions in commercial buildings have a stronger seasonal effect than the one in residential houses and (ii) there is much more activity in the power variations in commercial buildings than in residential houses.

Motivated by the lack of data for commercial buildings, we have developed a generative model for synthesizing high frequency current waveforms. Our model is based on a matrix factorization approach and the *low rank assumption*. The model efficiency has been validated with real data using previously mentioned metrics. Finally, we have proposed a simulation procedure that enables us to learn parameters on real data and then simulate new synthetic data. To enable algorithms testing and comparison, a simulated dataset called **SHED** is released.

In the second part of this thesis, we have developed an unsupervised learning framework for solving the NILM software problem with the specificity of using high frequency current and voltage measurements. One of the main difficulties of

this particular problem is the absence of result in the literature. We first showed how to transform the NILM single-channel source separation problem into a matrix factorization problem. To do so, we used the *rank one hypothesis* which states that any current matrix measured from an electric device has a *good enough* approximation of rank one. To leverage our knowledge on the electric sources (positivity, independence of power variations between devices and a relative sparsity of power variations) we have proposed **IVMF: the Independent-Variations Matrix Factorization**.

Finally we have applied IVMF to three different NILM datasets. On the SHED dataset a synthetic dataset for commercial building containing 8 buildings, IVMF outperformed 2 competitors (Independent Component Analysis (ICA) and Semi Non-negative Matrix Factorization (SNMF)). On the REDD dataset, a real dataset consisting of the total consumption of a full house and of the consumption of individual equipments, IVMF showed its ability to solve the NILM problem. Without any adaptation step, IVMF has also been applied on the BLUED dataset showing similar results. Eventhough IVMF has been developed to operate on commercial buildings, it shows consistent results on residential buildings. Along with a good estimation of power consumptions, IVMF has produced easily interpretable results.

We have already shown important advances in the knowledge of the NILM problem and proposed an efficient algorithm to solve it using high frequency data. However many challenges are still to be solved. We review here some of the future work that could be conducted.

Concerning the knowledge on the current sources it would be interesting to study more precisely the structure of the activations when the approximate rank is greater than one. In a first attempt we used a Dirichlet distribution to model the instantaneous share of each activation but this could be extended. Studying the activations' correlation structure of devices whose rank is greater than one could help designing a proper disaggregation method. Another lead to improve knowledge on devices would be to study at the same time the current and voltage measurements. Indeed, having a good understanding of the relationship between current and voltage for each device could help a lot the disaggregation.

One major limitation of IVMF is its ability to estimate only rank one sources. A first work around would be to add a *re-aggregation* step after the estimation of sub-components. This approach consisting of summing sub-components together has already been investigated in [Lange and Bergés, 2016] but a fully unsupervised adaptation to IVMF is not straightforward. We have already seen that the correlation matrix of the estimated activations (and thus power estimates) is almost the identify

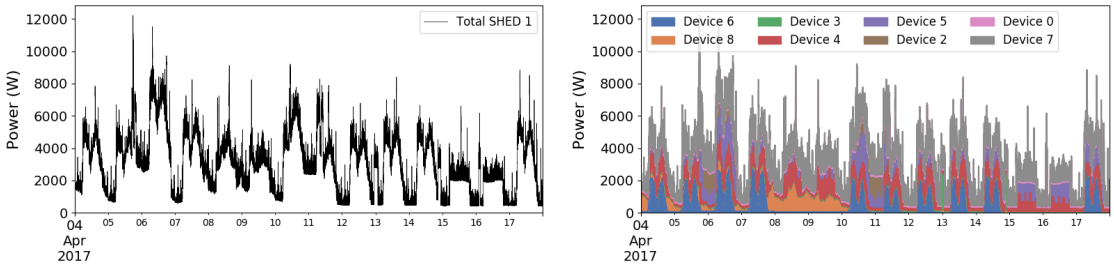
matrix signifying that activations are pairwise decorrelated. However, the activations can still present pairwise dependence. One way to group the activations could be by a greedy method based on an estimation of the measure of independence. Another lead could be to directly incorporate inside IVMF the capability of estimating higher rank signals. One could for instance use group regularization instead of row wise regularization [Friedman et al., 2010] based on the study of the correlation structure of activations for complex devices.

Another limitations of IVMF is that it treats the disaggregation problem as a batch problem. A major improvement would be to develop an online version of IVMF to be able to address larger datasets and also to adapt to potential changes in the behavior of the building. Addressing larger datasets could be done by only changing the optimization procedure: replacing the activation matrix update with a stochastic gradient descent approach for instance. However, adapting to changes of behavior in the building needs to change the model by using group regularization as expressed before, incorporating voltage measurements or making the signature matrix depend upon the time.

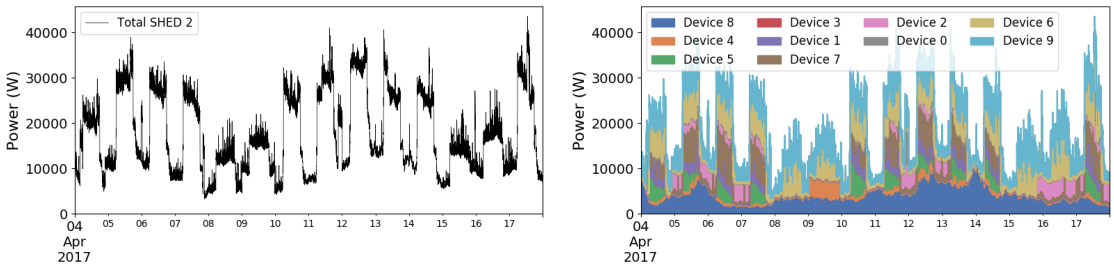
Finally, a major improvement would be to take advantage of the multiple buildings training. For now, IVMF is a completely blind method which focuses only on one building but Deep learning methods in the context of unsupervised learning have recently made progress. Generative Adversarial Network based methods have been applied to inverse problems [Lunz et al., 2018] or to audio source separation [Stoller et al., 2018]. Although this approach needs a lot of data, it only requires aggregated data.

Appendices

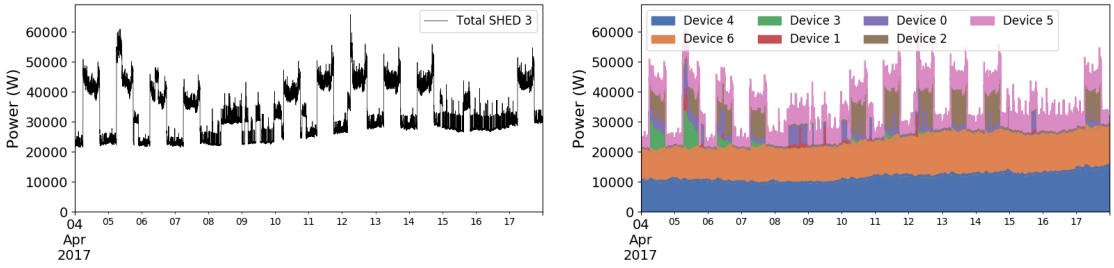
A The SHED dataset



(a) Building 1

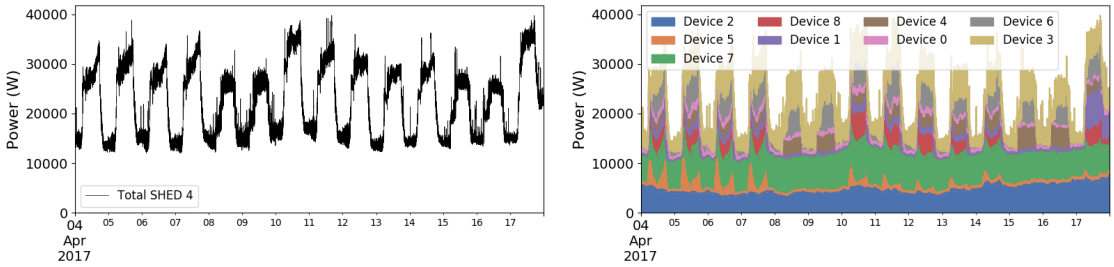


(b) Building 2

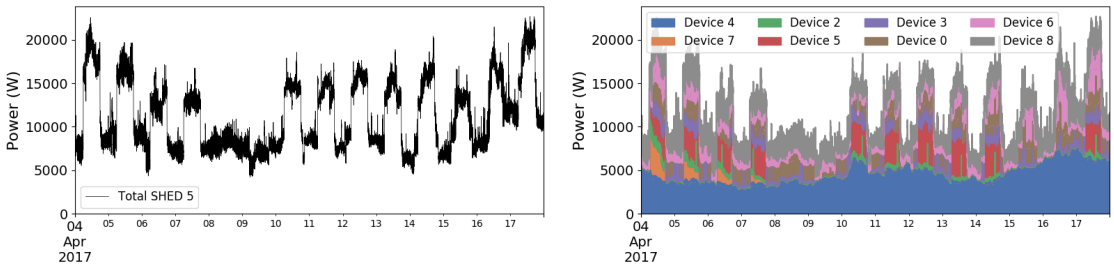


(c) Building 3

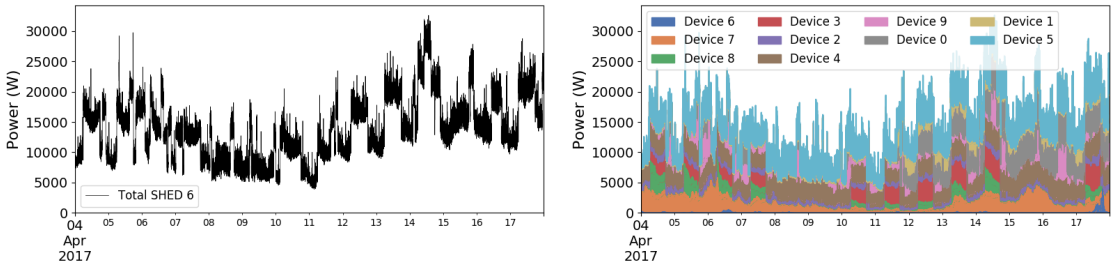
Figure A.1: Total and disaggregated power consumptions of buildings 1 to 3 of the SHED dataset.



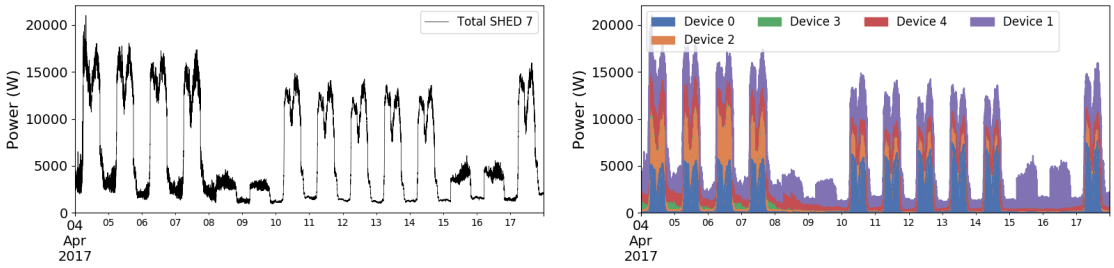
(a) Building 4



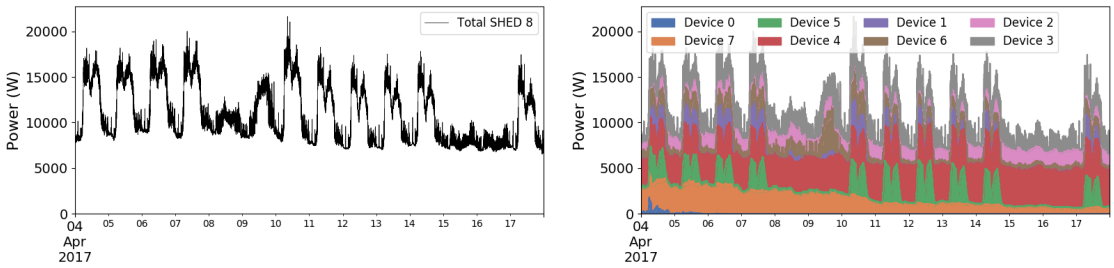
(b) Building 5



(c) Building 6

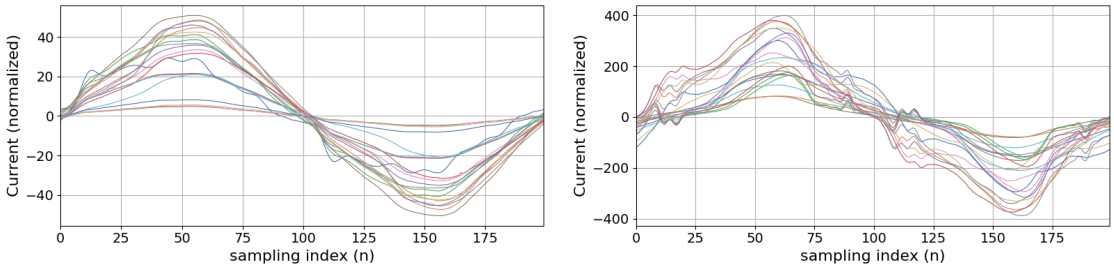


(d) Building 7



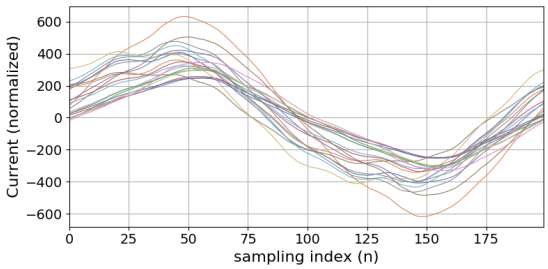
(e) Building 8

Figure A.2: Total and disaggregated power consumptions of buildings 4 to 8 of the SHED dataset.

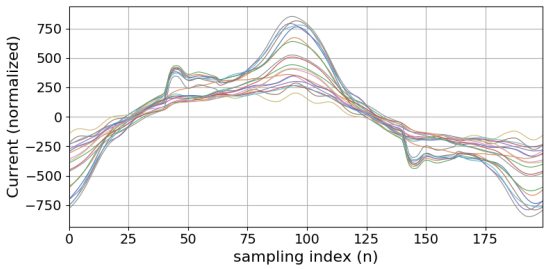


(a) Building 1

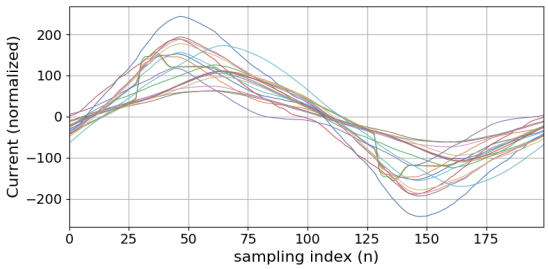
(b) Building 2



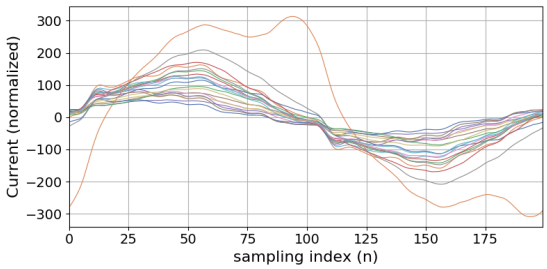
(c) Building 3



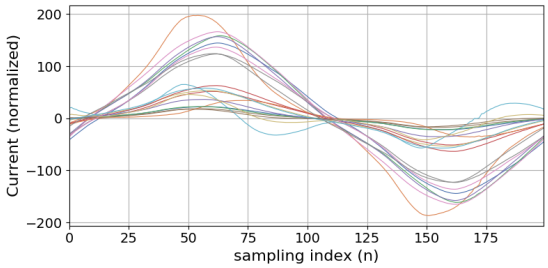
(d) Building 4



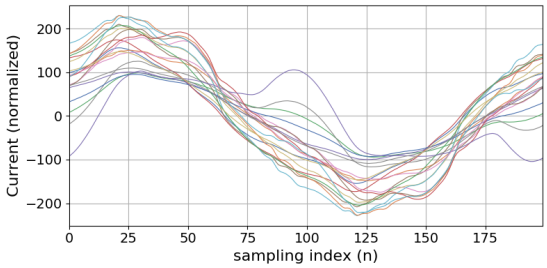
(e) Building 5



(f) Building 6



(g) Building 7



(h) Building 8

Figure A.3: Randomly chosen current waveforms for buildings 1 to 8 of the SHED dataset.

B IVMF: updating the activation matrix with a non-smooth regularizer.

The main problem with using non-smooth regularization is the fact that the non differentiability of the function to optimize makes it impossible to use efficient smooth convex optimization tool such as quasi Newton methods. Instead we use an accelerated proximal gradient methods. The optimization problem of interest is the following:

$$\begin{aligned} \underset{\mathbf{A}}{\text{minimize}} \quad & \frac{1}{2}\|X - \mathbf{S}\,\mathbf{A}\,\|_{Fro}^2 + \lambda \mathcal{G}_{abs}(\mathbf{A}) \\ \text{subject to} \quad & \mathbf{A} \geq 0 \end{aligned} \tag{2}$$

with $\mathcal{G}_{abs}(\mathbf{A}) = \sum_{k,t} |\,\mathbf{A}(k, t+1) - \mathbf{A}(k,t)|$.

In the image processing community, this subproblem is called deblurring with anisotropic total variation. One minor difference is that our total variation regularization is only on one direction (the time t) of the activation matrix $\mathbf{A}$, whereas in image processing it is calculated on both directions. Eventhough the total variation regularization is separable in the rows of $\mathbf{A}$, the euclidean data fit term makes the overall problem not separable in the general case. We will later discuss the case where $\mathbf{S}$ is an orthogonal matrix.

Our algorithm to solve this problem is greatly inspired by [Beck and Teboulle, 2009a]. It uses the FISTA algorithm which is a gradient based algorithm that can achieve fast convergence rate where each iterate depends on the previous 2 iterates. Its principle is to separate the function to minimize into two functions: one differentiable and one not differentiable. One iteration of FISTA consists in computing the gradient of the differentiable function and the proximal operator of the non-differentiable one.

The gradient is simply given by:

$$\nabla_{\mathbf{A}} f(\mathbf{A}) = -\,\mathbf{S}^\top (X - \mathbf{S}\,\mathbf{A}) \tag{3}$$

The Lipschitz constant of the gradient, denoted $L(\nabla_{\mathbf{A}} f)$ and used in FISTA, is bounded by:

$$L(\nabla_{\mathbf{A}} f) \leq \rho_{max}(\mathbf{S}^\top\,\mathbf{S}) \tag{4}$$

where $\rho_{max}(M)$ is the maximum eigen value of a matrix M.

Proof.

$$\begin{aligned}\|\nabla_{\mathbf{A}} f(\mathbf{A}_1) - \nabla_{\mathbf{A}} f(\mathbf{A}_2)\|_{Fro} =& \| -\mathbf{S}^\top (X - \mathbf{S}\,\mathbf{A}_1) + \mathbf{S}^\top (X - \mathbf{S}\,\mathbf{A}_2)\|_{Fro} && (5)\\ =& \|\,\mathbf{S}^\top\,\mathbf{S}\,\|_{Fro} \|\,\mathbf{A}_1 - \mathbf{A}_2\,\|_{Fro} && (6)\\ \leq& \rho_{max}(\mathbf{S}^\top\,\mathbf{S}) \|\,\mathbf{A}_1 - \mathbf{A}_2\,\|_{Fro} && (7)\end{aligned}$$

■

The proximal operator is defined itself as an optimization problem and unfortunately, in our case, has no closed form solution:

$$\text{prox}_{\lambda \mathcal{G}_{abs}}(Y) = \underset{A \in \mathbb{R}^{K \times T}}{\text{argmin}} \left[\frac{1}{2} \|\,\mathbf{A} - Y\,\|_{Fro}^2 + \lambda \mathcal{G}_{abs}(\mathbf{A}) + \mathbb{1}_{\{\mathbf{A} \geq 0\}} \right] \tag{8}$$

Since it has no closed form solution, an iterative procedure is needed. This new problem is separable in the rows of $\mathbf{A}$ ($\mathbf{a}_{k.}$), due to the fact that our total variation is only along the rows of $\mathbf{A}$. Using a matrix notation for the regularization, the problem reads:

$$\underset{\mathbf{a}_{k.} \in \mathbb{R}^T}{\text{minimize}} \quad \frac{1}{2} \|\,\mathbf{a}_{k.} - y_{k.}\|_2^2 + \lambda \|D\,\mathbf{a}_{k.}\,\|_1 + \mathbb{1}_{\{\mathbf{a}_{k.} \geq 0\}} \tag{9}$$

where D is a sparse $T - 1 \times T$ real matrix with non zero values defined by: $\forall t \in [1, T-1]$, $D_{t,t} = -1$ and $D_{t,t+1} = 1$. This vectorial optimization problem is similar to the fused lasso approximation and can also be viewed as the proximal operator of the fused lasso regression. An efficient way to calculate it is by solving its dual problem. The dual formulation enables us to get rid off the absolute value of the regularization and *replace* it with box constrained easier to handle. Let us write down the dual problem. For ease of notation we replace the $\mathbf{a}_{k.}$ vector by $\mathbf{a}$ and $y_{k.}$ by y.

We first start by introducing a new variable and its corresponding constraint:

$$\begin{aligned}\underset{\mathbf{a} \in \mathbb{R}^T,\, b \in \mathbb{R}^{T-1}}{\text{minimize}} \quad & \frac{1}{2} \|\,\mathbf{a} - y\|_2^2 + \lambda \|b\|_1 + \mathbb{1}_{\{\mathbf{a} \geq 0\}} && (10)\\ \text{such that} \quad & D\,\mathbf{a} = b\end{aligned}$$

The Lagrangian reads:

$$\mathcal{D}(\mu, \mathbf{a}, b) = \frac{1}{2}\| \mathbf{a} - y\|_2^2 + \lambda\|b\|_1 + \mathbb{1}_{\{\mathbf{a}\geq 0\}} + \mu^\top(D\,\mathbf{a} - b) \tag{11}$$

where $\mu \in \mathcal{R}^{T-1}$ is the dual variable.

The dual function is then defined as follows:

$$\begin{aligned}
\mathcal{D}(\mu) &= \inf_{\mathbf{a}\in\mathbb{R}^T,\, b\in\mathbb{R}^{T-1}} \mathcal{D}(\mu, \mathbf{a}, b) && (12)\\
&= \inf_{\mathbf{a}\in\mathbb{R}^T} \left[\frac{1}{2}\| \mathbf{a} - y\|_2^2 + \mu^\top D\,\mathbf{a} + \mathbb{1}_{\{\mathbf{a}\geq 0\}}\right] + \inf_{b\in\mathbb{R}^{T-1}} \left[\lambda\|b\|_1 - \mu^\top b\right] && (13)\\
& && (14)
\end{aligned}$$

Two minimization problems are needed to evaluate $\mathcal{D}$. In the optimization literature, they are called the *Fenchel* conjugate functions. The first one, on the variable $\mathbf{a}$ can be factorized to give:

$$\operatorname*{argmin}_{\mathbf{a}\in\mathbb{R}^T} \frac{1}{2}\| \mathbf{a} - y\|_2^2 + \mu^\top D\,\mathbf{a} + \mathbb{1}_{\{\mathbf{a}\geq 0\}} = (y - D^\top\mu)_+ \tag{15}$$

where $(x)_+ = \max(x, 0)$ represents the projection onto the positive orthant.

Proof.

$$\begin{aligned}
&\operatorname*{argmin}_{\mathbf{a}\in\mathbb{R}^T} \frac{1}{2}\| \mathbf{a} - y\|_2^2 + \mu^\top D\,\mathbf{a} + \mathbb{1}_{\{\mathbf{a}\geq 0\}} && (16)\\
=&\operatorname*{argmin}_{\mathbf{a}\in\mathbb{R}_+^T} \mathbf{a}^\top\mathbf{a} - 2y^\top\mathbf{a} + 2\mu^\top D\,\mathbf{a} + y^\top y && (17)\\
=&\operatorname*{argmin}_{\mathbf{a}\in\mathbb{R}_+^T} \| \mathbf{a} - (y - D^\top\mu)\|_2^2 + \|y\|_2^2 - \|y - D^\top\mu\|_2^2 && (18)\\
=&\operatorname*{argmin}_{\mathbf{a}\in\mathbb{R}_+^T} \| \mathbf{a} - (y - D^\top\mu)\|_2^2 && (19)\\
=&(y - D^\top\mu)_+ && (20)
\end{aligned}$$

■

The second problem, on the variable b is given by:

$$\inf_{b\in\mathbb{R}^{T-1}} \lambda\|b\|_1 - \mu^\top b = \mathbb{1}_{\{\|\mu\|_\infty\leq\lambda\}} \tag{21}$$

Combining these two results, the dual problem now reads:

$$\begin{aligned} \underset{\mu}{\text{maximize}} \quad & \|(y - D^\top \mu)_+ - (y - D^\top \mu)\|_2^2 - \|y - D^\top \mu\|_2^2 \\ \text{such that} \quad & \|\mu\|_\infty \leq \lambda \end{aligned} \tag{22}$$

It can be written as the following minimization problem:

$$\begin{aligned} \underset{\mu}{\text{minimize}} \quad & \|(y - D^\top \mu)_+\|_2^2 \overset{def}{=} f(\mu) \\ \text{such that} \quad & \|\mu\|_\infty \leq \lambda \end{aligned} \tag{23}$$

Problem (23) is a convex differentiable problem with box constraints. It has no closed form solutions and is not twice differentiable. We use FISTA to numerically optimize this problem. As defined previously, FISTA needs to compute the gradient with respect to μ, its Lipschitz constant and the proximal operator of the box constraints.

The gradient is given by:

$$\nabla_\mu f(\mu) = -2D(y - D^\top \mu)_+ \tag{24}$$

The proximal operator of $\mathbb{1}_{\{\|\|_1 \leq \lambda\}}$ is simply the orthogonal projection onto the convex set defined by the infinity norm: the interval $[-\lambda, \lambda]^{T-1}$.

Finally, the Lipschitz constant of the gradient is bounded by:

$$L(\nabla_\mu f) \leq 8 \tag{25}$$

Proof. Using the fact that, for $z \in \mathbb{R}^T$:

$$\|Dz\|_2^2 = \sum_{t=1}^{T-1} (z_t - z_{t-1})^2 \tag{26}$$

$$\leq 2 \sum_{t=1}^{T-1} (z_t^2 - z_{t-1}^2) \tag{27}$$

$$\leq 4 \|z_t\|_2^2 \tag{28}$$

$$\Rightarrow \|D\|_2 = \|D^\top\|_2 \leq \sqrt{4} \tag{29}$$

Then:

$$\begin{aligned}
\|\nabla_\mu f(\mu_1) - \nabla_\mu f(\mu_2)\|_2 =& \|2D(y - D^\top \mu_1)_+ - 2D(y - D^\top \mu_2)_+\|_2 && (30)\\
\leq& 2\|D\|_2 \|(y - D^\top \mu_1)_+ - (y - D^\top \mu_2)_+\|_2 && (31)\\
\leq& 2\|D\|_2 \|D^\top(\mu_1 - \mu_2)\|_2 && (32)\\
\leq& 2\|D\|_2 \|D^\top\|_2 \|\mu_1 - \mu_2\|_2 && (33)\\
\leq& 8\|\mu_1 - \mu_2\|_2 && (34)
\end{aligned}$$

■

The full iterations are given by Algorithm 2.

Algorithm 2: Updating activations $\mathbf{A}$ in the non smooth case

input : X, $\mathbf{S}$, $\mathbf{A}^{(0)}$, λ, I_{max} and J_{max}, the maximal number of iterations

Initialize $B^{(0)} = \mathbf{A}^{(0)}$, $t_{(0)} = 1$, $\tilde{\lambda} = \lambda/(4\rho_{max}(\mathbf{S}^\top \mathbf{S})(T-1))$

for $i = 1$ *to* I_{max} **do**

/* Dual Proximal Gradient step, see Equation (24) */

$Y = \mathbf{A}^{(i-1)} - \frac{1}{\rho_{max}(\mathbf{S}^\top \mathbf{S})} \nabla_{\mathbf{A}} f(\mathbf{A}^{(i-1)})$

for $k = 1$ *to* K **do**

/* Can be *parallelized* using matrix-matrix operations */

Initialize $\mu^{(0)} = \nu^{(0)}$, $s_{(0)} = 1$, $y = y_{.k}$

/* Solve K 1D-Prox TV, see (10) */

for $j = 1$ *to* J_{max} **do**

/* Dual Proximal Gradient step, see Equation (24) */

$\nu^{(j)} = \text{prox}_{\tilde{\lambda}}(\mu^{(j-1)} - 1/8\, \nabla_\mu \mathcal{D}(\mu^{(j-1)}))$

$s_{(j)} = \frac{1+\sqrt{1+4s^2_{(j-1)}}}{2}$

/* FISTA update */

$\mu^{(j)} = \nu^{(j-1)} + \left(\frac{s_{(j-1)}-1}{s_{(j)}}\right)\left(\nu^{(j)} - \nu^{(j-1)}\right)$

/* Update one row of B, Compute Proximal operator from dual optimal, see (15) */

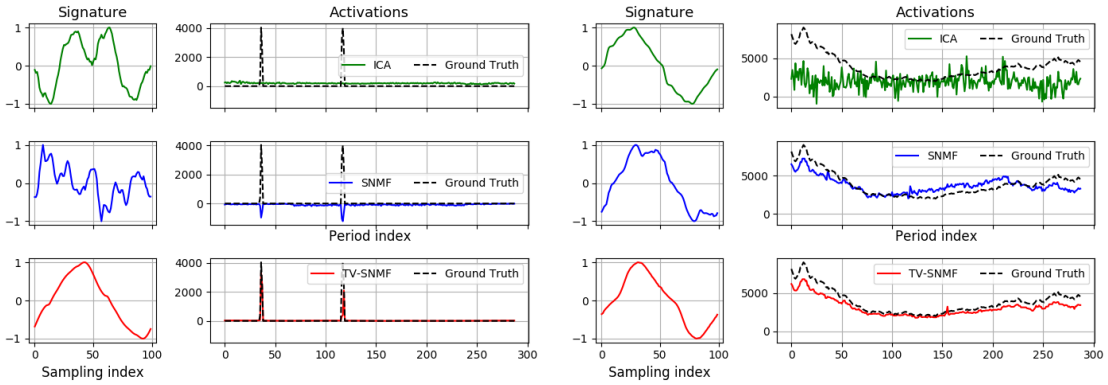
$b^{(i)}_{k.} = \left(y - \mu^{(J_{max})\top} D\right)_+$

$t_{(i)} = \frac{1+\sqrt{1+4t^2_{(i-1)}}}{2}$

/* FISTA update */

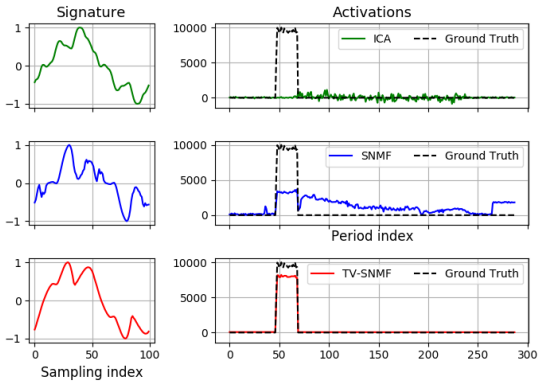
$\mathbf{A}^{(i)} = B^{(i-1)} + \left(\frac{t_{(i-1)}-1}{t_{(i)}}\right)\left(B^{(i)} - B^{(i-1)}\right)$

C Disaggregation results on the SHED dataset

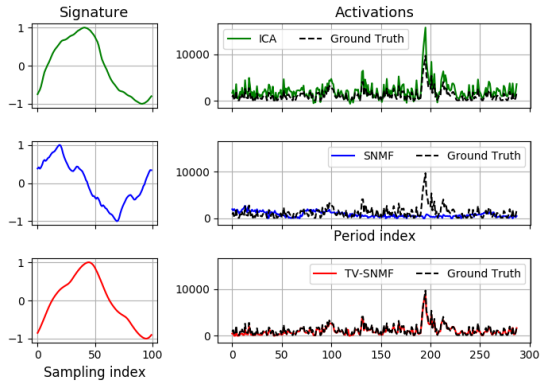


(a) Component 1

(b) Component 2

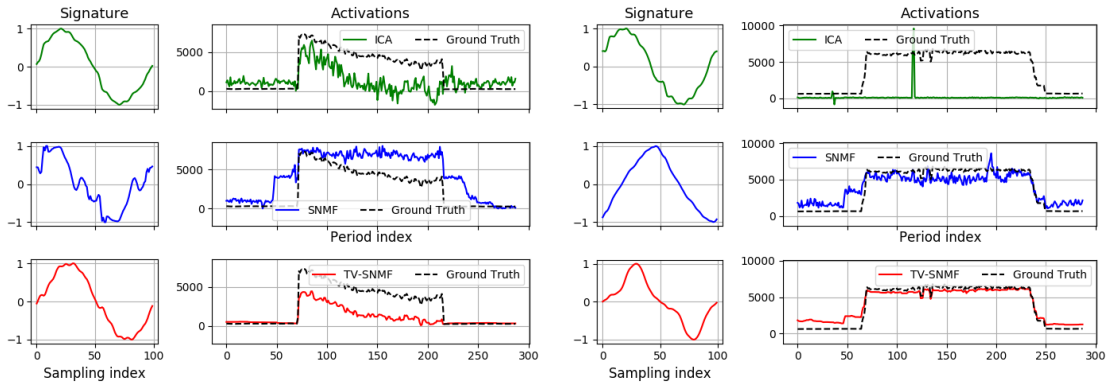


(c) Component 3



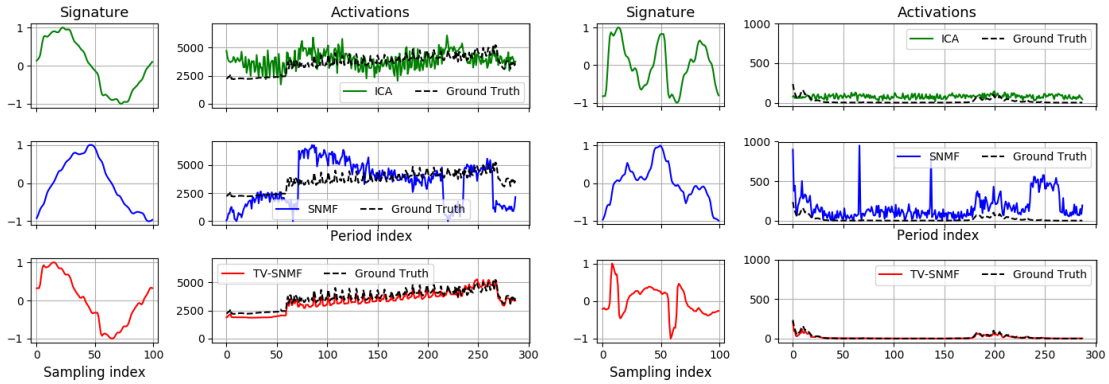
(d) Component 4

Figure C.1: IVMF, SNMF and ICA disaggregation results on SHED building 2 (1/2). Each sub-figure presents the estimation of one component (signature and calculated power consumption) and the corresponding ground truth.



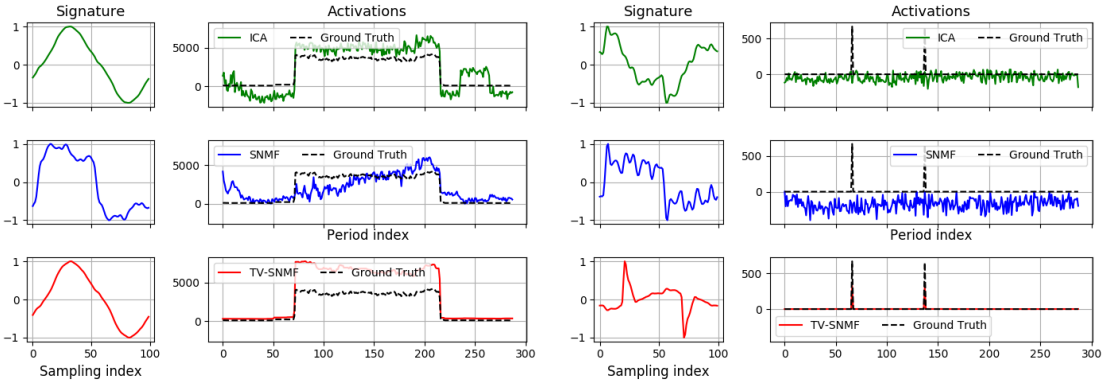
(a) Component 5 (b) Component 6

Figure C.2: IVMF, SNMF and ICA disaggregation results on SHED building 2 (2/2). Each sub-figure presents the estimation of one component (signature and calculated power consumption) and the corresponding ground truth.



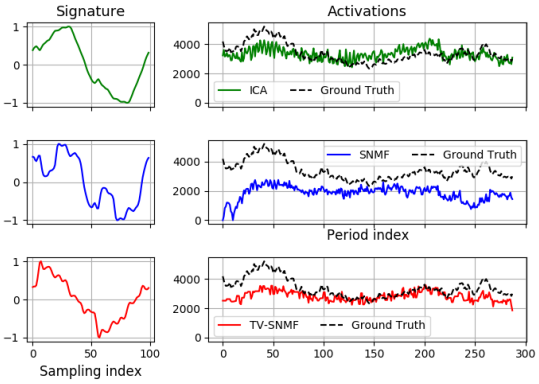
(a) Component 1 (b) Component 2

Figure C.3: IVMF, SNMF and ICA disaggregation results on SHED building 6 (1/2). Each sub-figure presents the estimation of one component (signature and calculated power consumption) and the corresponding ground truth.

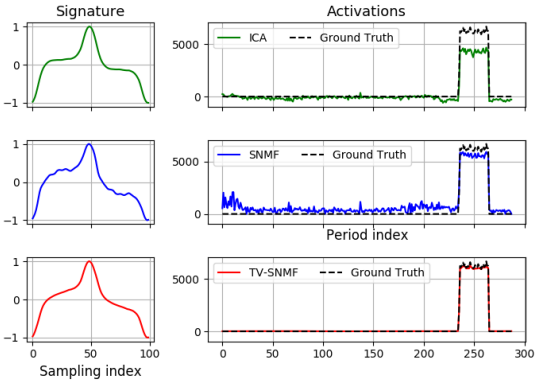


(a) Component 3

(b) Component 4



(c) Component 5



(d) Component 6

Figure C.4: IVMF, SNMF and ICA disaggregation results on SHED building 6 (2/2). Each sub-figure presents the estimation of one component (signature and calculated power consumption) and the corresponding ground truth.

Bibliography

[Ablin et al., 2018] Ablin, P., Cardoso, J.-F., and Gramfort, A. (2018). Faster ica under orthogonal constraint. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4464–4468. IEEE. Page 118.

[Aharon et al., 2006] Aharon, M., Elad, M., and Bruckstein, A. (2006). The k-svd: An algorithm for designing of overcomplete dictionaries for sparse representation. to appear in the ieee trans. *On Signal Processing*. Pages 99, 113.

[Alfares and Nazeeruddin, 2002] Alfares, H. K. and Nazeeruddin, M. (2002). Electric load forecasting: literature survey and classification of methods. *International journal of systems science*, 33(1):23–34. Page 20.

[Baranski and Voss, 2004] Baranski, M. and Voss, J. (2004). Genetic algorithm for pattern detection in nialm systems. In *2004 IEEE International Conference on Systems, Man and Cybernetics (IEEE Cat. No. 04CH37583)*, volume 4, pages 3462–3468. IEEE. Page 94.

[Barker et al., 2013] Barker, S., Kalra, S., Irwin, D., and Shenoy, P. (2013). Empirical characterization and modeling of electrical loads in smart homes. In *Green Computing Conference (IGCC), 2013 International*, pages 1–10. IEEE. Page 57.

[Batra et al., 2013] Batra, N., Gulati, M., Singh, A., and Srivastava, M. B. (2013). It's different: Insights into home energy consumption in india. In *Proceedings of the 5th ACM Workshop on Embedded Systems For Energy-Efficient Buildings*, pages 1–8. ACM. Page 52.

[Batra et al., 2014] Batra, N., Parson, O., Berges, M., Singh, A., and Rogers, A. (2014). A comparison of non-intrusive load monitoring methods for commercial and residential buildings. *arXiv preprint arXiv:1408.6595*. Pages 52, 57.

[Beck and Teboulle, 2009a] Beck, A. and Teboulle, M. (2009a). Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems. *IEEE transactions on image processing*, 18(11):2419–2434. Pages 123, 126, 127, 163.

[Beck and Teboulle, 2009b] Beck, A. and Teboulle, M. (2009b). A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences*, 2(1):183–202. Page 127.

[Beckel et al., 2014] Beckel, C., Kleiminger, W., Cicchetti, R., Staake, T., and Santini, S. (2014). 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*, pages 80–89. ACM. Page 52.

[Bell and Sejnowski, 1995] Bell, A. J. and Sejnowski, T. J. (1995). An information-maximization approach to blind separation and blind deconvolution. *Neural computation*, 7(6):1129–1159. Page 118.

[Bertsekas, 1997] Bertsekas, D. P. (1997). Nonlinear programming. *Journal of the Operational Research Society*, 48(3):334–334. Pages 116, 124, 127.

[Bonfigli et al., 2018] Bonfigli, R., Felicetti, A., Principi, E., Fagiani, M., Squartini, S., and Piazza, F. (2018). Denoising autoencoders for non-intrusive load monitoring: improvements and comparative evaluation. *Energy and Buildings*, 158:1461–1474. Page 103.

[Box et al., 2015] Box, G. E., Jenkins, G. M., Reinsel, G. C., and Ljung, G. M. (2015). *Time series analysis: forecasting and control*. John Wiley & Sons. Page 80.

[Boyd and Vandenberghe, 2004] Boyd, S. and Vandenberghe, L. (2004). *Convex optimization*. Cambridge university press. Page 124.

[Buneeva and Reinhardt, 2017] Buneeva, N. and Reinhardt, A. (2017). Ambal: Realistic load signature generation for load disaggregation performance evaluation. In *Smart Grid Communications (SmartGridComm), 2017 IEEE International Conference on. IEEE*, pages 1–9. Page 57.

[Byrd et al., 1995] Byrd, R. H., Lu, P., Nocedal, J., and Zhu, C. (1995). A limited memory algorithm for bound constrained optimization. *SIAM Journal on Scientific Computing*, 16(5):1190–1208. Pages 126, 127.

[Chan et al., 2000] Chan, W., So, A. T., and Lai, L. (2000). Harmonics load signature recognition by wavelets transforms. In *DRPT2000. International Conference on Electric Utility Deregulation and Restructuring and Power Technologies. Proceedings (Cat. No. 00EX382)*, pages 666–671. IEEE. Page 56.

[Chua et al., 1987] Chua, L. O., Desoer, C. A., and Kuh, E. S. (1987). *Linear and nonlinear circuits*. Page 46.

[Cole and Albicki, 1998] Cole, A. I. and Albicki, A. (1998). Algorithm for nonintrusive identification of residential appliances. In *ISCAS'98. Proceedings of the 1998 IEEE International Symposium on Circuits and Systems (Cat. No. 98CH36187)*, volume 3, pages 338–341. IEEE. Page 56.

[Comon, 1994] Comon, P. (1994). Independent component analysis, a new concept? *Signal processing*, 36(3):287–314. Page 117.

[Comon and Jutten, 2010] Comon, P. and Jutten, C. (2010). *Handbook of Blind Source Separation: Independent component analysis and applications*. Academic press. Page 117.

[De Baets et al., 2018] De Baets, L., Ruyssinck, J., Develder, C., Dhaene, T., and Deschrijver, D. (2018). Appliance classification using vi trajectories and convolutional neural networks. *Energy and Buildings*, 158:32–36. Page 57.

[Dinesh et al., 2017] Dinesh, C., Makanin, S., and Bajic, I. V. (2017). Incorporating time-of-day usage patterns into non-intrusive load monitoring. Page 77.

[Ding et al., 2010] Ding, C. H., Li, T., and Jordan, M. I. (2010). Convex and semi-nonnegative matrix factorizations. *IEEE transactions on pattern analysis and machine intelligence*, 32(1):45–55. Pages 67, 99, 113, 115, 116, 130.

[Dong et al., 2013] Dong, M., Meira, P. C., Xu, W., and Chung, C. (2013). Non-intrusive signature extraction for major residential loads. *IEEE Transactions on Smart Grid*, 4(3):1421–1430. Page 56.

[Ehrhardt-Martinez et al., 2010] Ehrhardt-Martinez, K., Donnelly, K. A., Laitner, S., et al. (2010). Advanced metering initiatives and residential feedback programs: a meta-review for household electricity-saving opportunities. American Council for an Energy-Efficient Economy Washington, DC. Page 20.

[Elhamifar and Sastry, 2015] Elhamifar, E. and Sastry, S. (2015). Energy disaggregation via learning powerlets and sparse coding. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*. Page 100.

[Farinaccio and Zmeureanu, 1999] Farinaccio, L. and Zmeureanu, R. (1999). Using a pattern recognition approach to disaggregate the total electricity consumption in a house into the major end-uses. *Energy and Buildings*, 30(3):245–259. Page 56.

[Feng and Kowalski, 2018] Feng, F. and Kowalski, M. (2018). Revisiting sparse ica from a synthesis point of view: Blind source separation for over and underdetermined mixtures. *Signal Processing*, 152:165–177. Pages 119, 135.

[Filip, 2011] Filip, A. (2011). Blued: A fully labeled public dataset for event-based non-intrusive load monitoring research. In *2nd Workshop on Data Mining Applications in Sustainability (SustKDD)*. Pages 52, 149.

[Fischer et al., 2015] Fischer, D., Härtl, A., and Wille-Haussmann, B. (2015). Model for electric load profiles with high time resolution for german households. *Energy and Buildings*, 92:170–179. Page 57.

[Friedman et al., 2010] Friedman, J., Hastie, T., and Tibshirani, R. (2010). A note on the group lasso and a sparse group lasso. *arXiv preprint arXiv:1001.0736*. Page 155.

[Gao et al., 2014] Gao, J., Giri, S., Kara, E. C., and Bergés, M. (2014). Plaid: a public dataset of high-resoultion electrical appliance measurements for load identification research: demo abstract. In *Proceedings of the 1st ACM Conference on Embedded Systems for Energy-Efficient Buildings*, pages 198–199. ACM. Pages 45, 52, 58, 65, 77, 81.

[Gao et al., 2015] Gao, J., Kara, E. C., Giri, S., and Bergés, M. (2015). A feasibility study of automated plug-load identification from high-frequency measurements. In *2015 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, pages 220–224. IEEE. Page 57.

[García et al., 2017] García, D., Díaz Blanco, I., Pérez García, D., Vega, C., Alberto, A., and Domínguez, M. (2017). Latent variable analysis in hospital electric power demand using non-negative matrix factorization. In *ESANN 2017 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*. i6doc. com publication. Page 101.

[Ghahramani and Jordan, 1996] Ghahramani, Z. and Jordan, M. I. (1996). Factorial hidden markov models. In *Advances in Neural Information Processing Systems*, pages 472–478. Page 95.

[Gillis and Kumar, 2015] Gillis, N. and Kumar, A. (2015). Exact and heuristic algorithms for semi-nonnegative matrix factorization. *SIAM Journal on Matrix Analysis and Applications*, 36(4):1404–1424. Page 116.

[Grainger et al., 2003] Grainger, J. J., Stevenson, W. D., Stevenson, W. D., et al. (2003). *Power system analysis*. Page 24.

[Harell et al., 2019] Harell, A., Makonin, S., and Bajić, I. V. (2019). Wavenilm: A causal neural network for power disaggregation from the complex power signal. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8335–8339. IEEE. Page 103.

[Hart, 1992] Hart, G. W. (1992). Nonintrusive appliance load monitoring. In *Proceedings of the IEEE*. Pages 29, 54, 57, 69, 93, 94.

[Hart et al., 1989] Hart, G. W., Kern Jr, E. C., and Schweppe, F. C. (1989). Non-intrusive appliance monitor apparatus. US Patent 4,858,141. Pages 21, 27.

[Hattori and Shinohara, 2017] Hattori, S. and Shinohara, Y. (2017). Actual consumption estimation algorithm for occupancy detection using low resolution smart meter data. In *SENSORNETS*, pages 39–48. Page 20.

[He et al., 2012] He, D., Du, L., Yang, Y., Harley, R., and Habetler, T. (2012). Front-end electronic circuit topology analysis for model-driven classification and monitoring of appliance loads in smart buildings. *IEEE Transactions on Smart Grid*, 3(4):2286–2293. Pages 49, 50, 51.

[Hippert et al., 2001] Hippert, H. S., Pedreira, C. E., and Souza, R. C. (2001). Neural networks for short-term load forecasting: A review and evaluation. *IEEE Transactions on power systems*, 16(1):44–55. Page 20.

[Hubert, 1990] Hubert, C. I. (1990). *Electric machines*. Prentice Hall. Page 47.

[Hyvarinen, 1999] Hyvarinen, A. (1999). Fast and robust fixed-point algorithms for independent component analysis. *IEEE transactions on Neural Networks*, 10(3):626–634. Pages 118, 127, 134.

[Hyvärinen and Oja, 2000] Hyvärinen, A. and Oja, E. (2000). Independent component analysis: algorithms and applications. *Neural networks*, 13(4):411–430. Pages 99, 113, 117.

[Jiang et al., 2019] Jiang, J., Kong, Q., Plumbley, M., and Gilbert, N. (2019). Deep learning based energy disaggregation and on/off detection of household appliances. *arXiv preprint arXiv:1908.00941*. Page 103.

[Johnson and Willsky, 2013] Johnson, M. J. and Willsky, A. S. (2013). Bayesian nonparametric hidden semi-markov models. *Journal of Machine Learning Research*, 14(Feb):673–701. Page 97.

[Jones et al., 2001] Jones, E., Oliphant, T., Peterson, P., et al. (2001). SciPy: Open source scientific tools for Python. Pages 62, 63, 127.

[Jutten and Herault, 1991] Jutten, C. and Herault, J. (1991). Blind separation of sources, part i: An adaptive algorithm based on neuromimetic architecture. *Signal processing*, 24(1):1–10. Pages 99, 113, 117.

[Kahl et al., 2016] Kahl, M., Haq, A. U., Kriechbaumer, T., and Jacobsen, H.-A. (2016). Whited-a worldwide household and industry transient energy data set. In *Workshop on Non-Intrusive Load Monitoring (NILM), 2016 Proceedings of the 3rd International*. Page 52.

[Kaselimi et al., 2019] Kaselimi, M., Doulamis, N., Doulamis, A., Voulodimos, A., and Protopapadakis, E. (2019). Bayesian-optimized bidirectional lstm regression model for non-intrusive load monitoring. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2747–2751. IEEE. Page 103.

[Kelly and Knottenbelt, 2015a] Kelly, J. and Knottenbelt, W. (2015a). Neural nilm: Deep neural networks applied to energy disaggregation. In *Proceedings of the 2nd ACM International Conference on Embedded Systems for Energy-Efficient Built Environments*, pages 55–64. ACM. Pages 31, 103.

[Kelly and Knottenbelt, 2015b] Kelly, J. and Knottenbelt, W. (2015b). The UK-DALE dataset, domestic appliance-level electricity demand and whole-house demand from five UK homes. *Scientific Data*, 2(150007). Page 52.

[Kim et al., 2011] Kim, H., Marwah, M., Arlitt, M., Lyon, G., and Han, J. (2011). Unsupervised disaggregation of low frequency power measurements. In *2011 SIAM international conference on data mining*, pages 39–48. SIAM. Pages 95, 96.

[Kim et al., 2017] Kim, J., Le, T.-T.-H., and Kim, H. (2017). Nonintrusive load monitoring based on advanced deep learning and novel signature. *Computational intelligence and neuroscience*, 2017. Page 104.

[Klemenjak and Goldsborough, 2016] Klemenjak, C. and Goldsborough, P. (2016). Non-intrusive load monitoring: A review and outlook. *arXiv preprint arXiv:1610.01191*. Page 69.

[Kolter et al., 2010] Kolter, J. Z., Batra, S., and Ng, A. Y. (2010). Energy disaggregation via discriminative sparse coding. In *Advances in Neural Information Processing Systems*, pages 1153–1161. Page 99.

[Kolter and Jaakkola, 2012] Kolter, J. Z. and Jaakkola, T. (2012). Approximate inference in additive factorial hmms with application to energy disaggregation. In *Artificial Intelligence and Statistics*, pages 1472–1482. Pages 57, 96, 97, 98.

[Kolter and Johnson, 2011] Kolter, J. Z. and Johnson, M. J. (2011). Redd: A public data set for energy disaggregation research. In *Workshop on Data Mining Applications in Sustainability (SIGKDD), San Diego, CA*, volume 25, pages 59–62. Pages 30, 33, 52, 143.

[Kotz et al., 2012] Kotz, S., Kozubowski, T., and Podgorski, K. (2012). *The Laplace distribution and generalizations: a revisit with applications to communications, economics, engineering, and finance*. Springer Science & Business Media. Page 63.

[Kriechbaumer et al., 2019] Kriechbaumer, T., Jorde, D., and Jacobsen, H.-A. (2019). Waveform signal entropy and compression study of whole-building energy datasets. In *Proceedings of the Tenth ACM International Conference on Future Energy Systems*, pages 58–67. ACM. Page 32.

[Lam et al., 2007] Lam, H. Y., Fung, G. S. K., and Lee, W. K. (2007). A novel method to construct taxonomy electrical appliances based on load signatures of electrical appliances based on load signatures. *IEEE Transactions on Consumer Electronics*, 53(2):653–660. Pages 58, 94.

[Lange and Bergés, 2016] Lange, H. and Bergés, M. (2016). Bolt: Energy disaggregation by online binary matrix factorization of current waveforms. In *Proceedings*

of the 3rd ACM International Conference on Systems for Energy-Efficient Built Environments, pages 11–20. ACM. Pages 31, 102, 105, 140, 154.

[Lange and Bergés, 2018] Lange, H. and Bergés, M. (2018). Variational bolt: Approximate learning in factorial hidden markov models with application to energy disaggregation. In *Thirty-Second AAAI Conference on Artificial Intelligence*. Page 97.

[LeCun et al., 1998] LeCun, Y., Bottou, L., Bengio, Y., Haffner, P., et al. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324. Page 33.

[Lee and Seung, 2001] Lee, D. D. and Seung, H. S. (2001). Algorithms for non-negative matrix factorization. In *Advances in neural information processing systems*, pages 556–562. Pages 99, 113.

[Lee et al., 2007] Lee, H., Battle, A., Raina, R., and Ng, A. Y. (2007). Efficient sparse coding algorithms. In *Advances in neural information processing systems*, pages 801–808. Pages 117, 125.

[Lee et al., 2005] Lee, K. D., Leeb, S. B., Norford, L. K., Armstrong, P. R., Holloway, J., and Shaw, S. R. (2005). Estimation of variable-speed-drive power consumption from harmonic content. *IEEE Transactions on Energy Conversion*, 20(3):566–574. Pages 29, 95.

[Lee et al., 2004] Lee, W., Fung, G., Lam, H., Chan, F., and Lucente, M. (2004). Exploration on load signatures. In *International conference on Electrical Engineering (ICEE)*, volume 152. Page 58.

[Leeb et al., 1995] Leeb, S. B., Shaw, S. R., and Kirtley, J. L. (1995). Transient event detection in spectral envelope estimates for nonintrusive load monitoring. *IEEE Transactions on Power Delivery*, 10(3):1200–1210. Page 56.

[Liang et al., 2010] Liang, J., Ng, S. K., Kendall, G., and Cheng, J. W. (2010). Load signature study—part i: Basic concept, structure, and methodology. *IEEE Transactions on Power Delivery*, 25(2):551–560. Pages 58, 94.

[Liang et al., 2008] Liang, Z., Wei, J., Zhao, J., Liu, H., Li, B., Shen, J., and Zheng, C. (2008). The statistical meaning of kurtosis and its new application to identification of persons based on seismic signals. *Sensors*, 8(8):5106–5119. Page 62.

[Lunz et al., 2018] Lunz, S., Öktem, O., and Schönlieb, C.-B. (2018). Adversarial regularizers in inverse problems. In *Advances in Neural Information Processing Systems*, pages 8507–8516. Page 155.

[Mairal et al., 2010] Mairal, J., Bach, F., Ponce, J., and Sapiro, G. (2010). Online learning for matrix factorization and sparse coding. *Journal of Machine Learning Research*, 11(Jan):19–60. Pages 99, 113, 117.

[Makonin et al., 2017] Makonin, S., Wang, Z. J., and Tumpach, C. (2017). Rae: The rainforest automation energy dataset for smart grid meter data analysis. *arXiv preprint arXiv:1705.05767*. Page 52.

[Marceau and Zmeureanu, 2000] Marceau, M. L. and Zmeureanu, R. (2000). Nonintrusive load disaggregation computer program to estimate the energy consumption of major end uses in residential buildings. *Energy conversion and management*, 41(13):1389–1403. Page 56.

[Marcus et al., 1993] Marcus, M., Santorini, B., and Marcinkiewicz, M. A. (1993). Building a large annotated corpus of english: The penn treebank. Page 33.

[Martins et al., 2018] Martins, P. B., Gomes, J. G., Nascimento, V. B., and de Freitas, A. R. (2018). Application of a deep learning generative model to load disaggregation for industrial machinery power consumption monitoring. In *2018 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm)*, pages 1–6. IEEE. Pages 103, 104.

[Mei et al., 2017] Mei, J., De Castro, Y., Goude, Y., and Hébrail, G. (2017). Nonnegative matrix factorization for time series recovery from a few temporal aggregates. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 2382–2390. JMLR. org. Page 57.

[Meziane et al., 2017] Meziane, M. N., Hacine-Gharbi, A., Ravier, P., Lamarque, G., Le Bunetel, J.-C., and Raingeaud, Y. (2017). Electrical appliances identification and clustering using novel turn-on transient features. In *ICPRAM*, pages 647–654. Page 56.

[Muir and Lopatto, 2004] Muir, A. and Lopatto, J. (2004). Final report on the august 14, 2003 blackout in the united states and canada: causes and recommendations. Page 22.

[Murray et al., 2017] Murray, D., Stankovic, L., and Stankovic, V. (2017). An electrical load measurements dataset of united kingdom households from a two-year longitudinal study. *Scientific data*, 4:160122. Page 52.

[Murray et al., 2019] Murray, D., Stankovic, L., Stankovic, V., Lulic, S., and Sladojevic, S. (2019). Transferability of neural network approaches for low-rate energy disaggregation. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 8330–8334. IEEE. Page 103.

[Nait-Meziane et al., 2016] Nait-Meziane, M., Hacine-Gharbi, A., Ravier, P., Lamarque, G., Le Bunetel, J.-C., and Raingeaud, Y. (2016). Hmm-based transient and steady-state current signals modeling for electrical appliances identification. In *Proceedings of the 5th International Conference on Pattern Recognition Applications and Methods*, pages 670–677. SCITEPRESS-Science and Technology Publications, Lda. Page 56.

[Nocedal and Wright, 2006] Nocedal, J. and Wright, S. J. (2006). *Numerical optimization 2nd*. Springer. Page 124.

[Norford and Mabey, 1992] Norford, L. and Mabey, N. (1992). Non-inlrusive eleclric load moniloring in commercial buildings leslie k. *Norford and Nicholas Mabey Massachusells Institute of Technology Cambridge, MA*. Page 94.

[Norford and Leeb, 1996] Norford, L. K. and Leeb, S. B. (1996). Non-intrusive electrical load monitoring in commercial buildings based on steady-state and transient load-detection algorithms. *Energy and Buildings*, 24(1):51–64. Page 94.

[Ohm, 1827] Ohm, G. S. (1827). *Die galvanische Kette, mathematisch bearbeitet*. TH Riemann. Page 44.

[Olshausen and Field, 1997] Olshausen, B. A. and Field, D. J. (1997). Sparse coding with an overcomplete basis set: A strategy employed by v1? *Vision research*, 37(23):3311–3325. Pages 99, 113, 116.

[Oord et al., 2016] Oord, A. v. d., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., Kalchbrenner, N., Senior, A., and Kavukcuoglu, K. (2016). Wavenet: A generative model for raw audio. *arXiv preprint arXiv:1609.03499*. Page 103.

[Oukrich et al., 2017] Oukrich, N., El Moumni, S., Maach, A., et al. (2017). Discrete wavelet transform and classifiers for appliances recognition. In *Proceedings of the*

Mediterranean Symposium on Smart City Applications, pages 223–232. Springer. Page 56.

[Parikh and Boyd, 2014] Parikh, N. and Boyd, S. (2014). Proximal algorithms. *Foundations and Trends in Optimization*, 1(3):127–239. Pages 124, 126.

[Parson et al., 2012] Parson, O., Ghosh, S., Weal, M., and Rogers, A. (2012). Non-intrusive load monitoring using prior models of general appliance types. In *Twenty-Sixth AAAI Conference on Artificial Intelligence*. Page 97.

[Pedregosa et al., 2011] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830. Page 130.

[Pereira and Nunes, 2018] Pereira, L. and Nunes, N. (2018). Performance evaluation in non-intrusive load monitoring: Datasets, metrics, and tools—a review. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 8(6):e1265. Page 140.

[Picon et al., 2016] Picon, T., Meziane, M. N., Ravier, P., Lamarque, G., Novello, C., Bunetel, J.-C. L., and Raingeaud, Y. (2016). Cooll: Controlled on/off loads library, a public dataset of high-sampled electrical signals for appliance identification. *arXiv preprint arXiv:1611.05803*. Pages 52, 58, 65, 77, 81.

[Plumbley, 2003] Plumbley, M. D. (2003). Algorithms for nonnegative independent component analysis. *IEEE Transactions on Neural Networks*, 14(3):534–543. Page 119.

[Powers et al., 1991] Powers, J., Margossian, B., and Smith, B. (1991). Using a rule-based algorithm to disaggregate end-use load profiles from premise-level data. *IEEE Computer Applications in Power*, 4(2):42–47. Page 56.

[Reinhardt et al., 2012] Reinhardt, A., Baumann, P., Burgstahler, D., Hollick, M., Chonov, H., Werner, M., and Steinmetz, R. (2012). On the accuracy of appliance identification based on distributed load metering data. In *Sustainable Internet and ICT for Sustainability (SustainIT), 2012*, pages 1–9. IEEE. Pages 52, 78, 81.

[Rudin et al., 1992] Rudin, L. I., Osher, S., and Fatemi, E. (1992). Nonlinear total variation based noise removal algorithms. *Physica D: nonlinear phenomena*, 60(1-4):259–268. Page 123.

[Sadeghianpourhamami et al., 2017] Sadeghianpourhamami, N., Ruyssinck, J., Deschrijver, D., Dhaene, T., and Develder, C. (2017). Comprehensive feature selection for appliance classification in nilm. *Energy and Buildings*, 151:98–106. Page 55.

[Seichepine et al., 2014] Seichepine, N., Essid, S., Févotte, C., and Cappe, O. (2014). Piecewise constant nonnegative matrix factorization. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6721–6725. IEEE. Page 123.

[Shaloudegi et al., 2016] Shaloudegi, K., György, A., Szepesvári, C., and Xu, W. (2016). Sdp relaxation with randomized rounding for energy disaggregation. In *Advances in Neural Information Processing Systems*, pages 4978–4986. Page 97.

[Shin et al., 2019] Shin, C., Joo, S., Yim, J., Lee, H., Moon, T., and Rhee, W. (2019). Subtask gated networks for non-intrusive load monitoring. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 1150–1157. Page 104.

[Sivakumar et al., 2016] Sivakumar, D., Srividhya, J., and Shanmathi, T. (2016). A review on power quality monitoring and its controlling techniques. In *8th International Conference on Latest Trends in Engineering and Technology (ICLTET'2016) May*, pages 5–6. Page 20.

[Srinivasan et al., 2005] Srinivasan, D., Ng, W., and Liew, A. (2005). Neural-network-based signature recognition for harmonic source identification. *IEEE Transactions on Power Delivery*, 21(1):398–405. Page 56.

[Stoller et al., 2018] Stoller, D., Ewert, S., and Dixon, S. (2018). Adversarial semi-supervised audio source separation applied to singing voice extraction. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2391–2395. IEEE. Page 155.

[Sultanem, 1991] Sultanem, F. (1991). Using appliance signatures for monitoring residential loads at meter panel level. *IEEE Transactions on Power Delivery*, 6(4):1380–1385. Pages 28, 54, 55, 56, 58.

[Tabatabaei et al., 2016] Tabatabaei, S. M., Dick, S., and Xu, W. (2016). Toward non-intrusive load monitoring via multi-label classification. *IEEE Transactions on Smart Grid*, 8(1):26–40. Page 31.

[Tavner et al., 2008] Tavner, P., Ran, L., Penman, J., and Sedding, H. (2008). Condition monitoring of rotating electrical machines. *IET Electric Power Applications*, 56(4):215–247. Page 20.

[Tibshirani, 1996] Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288. Page 116.

[Yu, 2010] Yu, S.-Z. (2010). Hidden semi-markov models. *Artificial intelligence*, 174(2):215–243. Page 96.

[Zhang et al., 2018] Zhang, C., Zhong, M., Wang, Z., Goddard, N., Sutton, C., et al. (2018). Sequence-to-point learning with neural networks for nonintrusive load monitoring. In *Thirty-Second AAAI Conference on Artificial Intelligence*. Page 103.

[Zhong et al., 2014] Zhong, M., Goddard, N., and Sutton, C. (2014). Signal aggregate constraints in additive factorial hmms, with application to energy disaggregation. In *Advances in Neural Information Processing Systems*, pages 3590–3598. Page 97.

ECOLE
DOCTORALE

Titre : Sur la Solution au Problème de Décomposition des Consommations Electriques dans les Grands Batiments: Analyses, Simulations et Apprentissage Non-supervisé à Base de Factorisation de Matrices.

Mots clés : Décomposition des Consommations Electriques, Apprentissage Non-supervisé, Factorisation de Matrices.

Résumé : La prise de conscience des conséquences du réchauffement climatique a permis de lancer un mouvement de réduction de l'utilisation d'énergie. L'électricité utilisée dans les bâtiments représente une part importante de la consommation d'énergie et doit donc être utilisée de manière efficace. Pour cela, il est nécessaire de pouvoir mesurer et suivre la consommation électrique de chaque appareil au sein d'un bâtiment. Depuis 30 ans, une méthode de suivi des consommations électriques, *Non Intrusive Load Monitoring* (NILM), propose, à partir d'un unique compteur mesurant la consommation totale du bâtiment, de déterminer la contribution de chaque appareil électrique. Cette méthode est basée sur un algorithme de désagrégation des consommations électriques et permet de s'affranchir de l'utilisation d'un compteur de mesure pour chaque appareil électrique du bâtiment.
Cette thèse aborde les problèmes algorithmiques que présente le NILM. De manière générale, la problématique est celle de la séparation de sources. Les différentes sources à estimer correspondent ici à la consommation électrique des différents appareils branchés sur un même réseau. La mesure réalisée, aussi appelée observation mélangée, correspond à la somme de toutes les consommations. Ainsi, les principales difficultés du NILM sont : (i) la standardisation de la formulation, (ii) le caractère mal-posé du problème (perte d'information), (iii) les connaissances insuffisantes sur les signaux et (iv) l'implémentation d'un algorithme d'apprentissage. L'objectif principale de cette thèse est de traiter le NILM dans le cadre des grands bâtiments (commerciaux, bureaux, industriels) en utilisant des mesures hautes fréquences du courant et de la tension. Cependant les maisons individuelles et leurs propres types d'appareils électriques ne sont pas exclus de cette étude.
Dans une première partie nous abordons le problème du manque de connaissance des signaux de consommation électriques, à la fois ceux des grands bâtiments et ceux des différents appareils utilisés. Dans une seconde partie, nous abordons le problème de la séparation de source. Grâce à nos résultats d'analyse et par manque de données, nous traitons ce problème à l'aide de techniques d'apprentissage non-supervisées.

Title : On Solving the Non Intrusive Load Monitoring Problem in Large Buildings: Analyses, Simulations and Factorization Based Unsupervised Learning.

Keywords : Energy Disaggregation, Unsupervised Learning, Matrix Factorization.

Abstract : With the increasing awareness about the problem of climate change and the high level of energy consumption, a need for energy efficiency has emerged especially for electric power consumptions in buildings. To spur energy savings, a method called Non Intrusive Load Monitoring (NILM) has been introduced thirty years ago. It consists of estimating individual appliance energy consumptions from the measurement of the total consumption of the building. Its main advantage over traditional sub-metering methods is to use a single electric power meter at the main breaker of the building and then use a disaggregation algorithm to separate the contributions of each appliance.
The goal of this thesis is to address the algorithmic challenge offered by NILM. The NILM problem can be formulated as a source separation problem, where the sources are the individual electric consumptions and the mixed observation is simply the sum of individual consumptions. Its main difficulties are: (i) the standardization of the formulation, (ii) the ill-posedness of the problem, (iii) the lack of knowledge and (iv) the machine learning algorithm design. All our contributions follow from the principal objective that is to solve the NILM problem for huge systems such as commercial or industrial buildings using high frequency current and voltage measurements. However, houses and the specific equipment found inside these buildings are not excluded of the study. This thesis is split into two parts.
In the first part, we tackle the lack of knowledge and datasets for NILM in commercial buildings. In the second part, we deal with the NILM software challenges by exploring unsupervised source separation techniques.

Institut Polytechnique de Paris
91120 Palaiseau, France