Learning:
Wavelet-Dictionaries and Continuous Dictionaries

von Ludger Prünte

Dissertation

zur Erlangung des Grades eines Doktors der Naturwissenschaften

– Dr. rer. nat. –

Vorgelegt im Fachbereich 3 (Mathematik & Informatik)

der Universität Bremen

im Januar 2008

Datum des Promotionskolloquiums: 05.05.2008

Gutachter: Prof. Dr. Peter Maaß, Universität Bremen
Dr. Rémi Gribonval, CR1, INRIA-IRISA, Rennes, Frankreich

Zusammenfassung

Diese Arbeit ist ein Beitrag zum Thema „Lernen von Dictionaries“. In vielen Bereichen der Anwendung von dünn besetzten Kodierungen ist die Frage, welches Dictionary gewählt wird, als erste zu beantworten. Das Dictionary von den gegebenen Signalen quasi zu lernen, d.h. an diese anzupassen, ist hier eine Alternative zur Wahl eines Vordefinierten. Im Laufe des letzten Jahrzehnts wurden mehrere Algorithmen zur näherungsweisen Lösung dieses Problems vorgestellt, in letzter Zeit auch in Verbindung mit zusätzlichen Eigenschaften wie Verschiebungsinvarianz. Nichtsdestotrotz existierte bisher kein Algorithmus, der die wichtigen Eigenschaften Verschiebungs- und Skalierungsinvarianz mit einem schnellen Kodierungsalgorithmus verbindet. Auch Fragen zu stetigen Verallgemeinerungen des Lernen von Dictionaries sind bisher unbehandelt. Der Zweck dieser Arbeit ist es, Ergebnisse in diesen beiden Feldern zu erzielen.
Betreffend die erste Fragestellung, führen wir einen Algorithmus ein, der Wavelet-Dictionaries, d.h. Dictionaries, die eine Vereinigung einer festen Anzahl von Waveletbasen sind, ausgehend von gegebenen Signalen lernt. Dieser Algorithmus basiert auf der Minimierung eines Fehlermaßes, abhängig von Dictionary und dünn besetzten Koeffizienten, unter Nebenbedingungen, die aus dem Lifting-Schema folgen. Im Allgemeinen ist diese Minimierung ein nicht konvexes Problem. Wir entwickeln einen approximativen Lösungsalgorithmus, indem wir abwechselnd eine der beiden Variablen festhalten und so einfache Probleme aus den Bereichen dünn besetztes Kodieren und konvexe Minimierung erhalten. Später wenden wir diesen Algorithmus auf Probleme des Maschinenbaus, genauer die Analyse von Laufgeräuschen einer Linearführung, und Musik, genauer die Rekonstruktion der Partitur einer Klaviersonate, an.
Das Lernen einer stetige Verallgemeinerung eines Dictionarys ist Inhalt des zweiten Teils der Arbeit. Wir definieren zu einer beliebigen dünn-Besetztheits-Norm und einer gegebenen Signalfunktion ein Fehlerfunktional, äquivalent zum diskreten Fall. Dieses Funktional ist abhängig von zwei Variablen, den Analoga zu Dictionary und dünn besetzten Koeffizienten in Lebesgue- und Sobolev-Räumen. Wir untersuchen die Existenz eines Minimierers dieses Funktionals im Bezug auf diese beiden Variablen. Des weiteren zeigen wir einen Weg auf, wenigstens ein lokales Minimum mittels einer Verallgemeinerung des CG-Algorithmus zu berechnen.

Abstract

This thesis is a contribution to the field of "dictionary learning." In many fields of sparse coding applications, the question, which dictionary to choose, is the first one to answer. Learning the dictionary from the given signals, i.e. fitting it to them, is here an alternative to choosing a predefined one. During the past decade came up several algorithms for finding approximative solutions to this problem, lastly also in combination with additional properties like shift-invariance. Nevertheless, there is till now no algorithm, combining the important properties of shift- and scale-invariance with a fast algorithm for coding. Furthermore, also questions concerning a continuous generalization of the dictionary learning problem are not treated till now. The intention of this thesis is to obtain results in both fields.
For the first aim, we introduce an algorithm, learning a wavelet-dictionary, i.e. a dictionary being composed of a fixed number of wavelet bases, from a given data set. This algorithm is based on minimizing an error measure, depending on the dictionary and sparse coding coefficients, satisfying side conditions induced by the lifting scheme. Generally the minimization is a non-convex problem. We propose an approximative solution by iteratively fixing one of the two free variable sets and obtaining separated sparse coding and convex minimization problems. Later on we apply this algorithm to problems in the fields of mechanical engineering, more precisely analyzing the operation noise of a linear guideway, and musics, more precisely reconstructing the score of a piano sonata.
The learning of a continuous generalization of dictionaries is the topic of the second part of this thesis. We define for an arbitrary sparsity norm and a given signal function a corresponding error functional equivalent to the discrete case. This functional depends on two variables, the analogons to dictionary and sparse coding coefficients, in Lebesgue and Sobolev spaces. We investigate the existence of a minimizer of this non-convex functional according to this two variable functions. Furthermore we point out a practical way to obtain at least a local minimum using a generalization of the conditional gradient algorithm.

I would like to thank:

Prof. Dr. Peter Maaß for supervision, support and motivation in writing this thesis.

The second referee, Dr. Rémi Gribonval, CR1, for kindly undertaking the task of examining this thesis.

All the people from the "Zentrum für Technomathematik" for mathematical and technical support, especially the people of my group for a very pleasant working environment and the nice years I spent.

My fellow students at the graduate school "Scientific Computing in Engineering" (SCiE), especially my "Tandem" Wang, Yuhang, for the successful cooperation and the productive and amicable athmosphere.

My colleagues Zafer Akbay, Kristian Bredies, Lutz Justen, Dorota Kubalinska, Dirk Lorenz, Jonathan Montalvo, Manfred Nölte, Henning Thielemann and Yakub Tijani for endless fruitful discussions, advice and support – both mathematically and personally.

All the unmentioned people that supported and motivated me on my way to and through this thesis, to specify all of them would exceed the available space.

Ludger Prünte
Zentrum für Technomathematik
Universität Bremen

pruente@math.uni-bremen.de

Contents

Chapter 1

Introduction

During the past 15 years there had been a rising interest concerning themes called "sparse representation" and "overcomplete dictionaries." This progression started in the signal and image processing community, but spread out during the years to several fields of computer, natural and engineering sciences, as well as mathematics and beyond. Maybe a starting point was article [105] by Stéphane Mallat and Zhifeng Zhang, introducing the matching pursuit (cf. Section 2.2.1) and giving a first application on human voices.

The basic idea in this context is the following (for a more detailed introduction see Section 2.2.1): Usually a given signal y is uniquely described by use of a basis $\{e_j\}_{j\in\tilde{J}}$ of an Hilbert space H

$$y = \sum_{j\in\tilde{J}} x_j e_j \quad \text{where} \quad \forall j \in \tilde{J} : \; x_j \in \mathbb{K} \tag{1.1}$$

with $\mathbb{K} \in \{\mathbb{C}, \mathbb{R}\}$. Here we replace the basis by an overcomplete dictionary, i.e. a set $\mathcal{D} = \{d_j\}_{j\in J} \subset H$ of normed atoms $\|d_j\|_H = 1$, satisfying $H = V := \operatorname{span}\{d_j\}$. So we can choose the sparsest representation of y. Similar approaches are also valid for approximations of y instead of representations.

The main topic of this thesis is, as we will point out more detailed in Section 1.4, the learning of dictionaries $\mathcal{D}$, i.e. the construction in consideration of given signals, in two frameworks. First enforcing wavelet basis structures, second proving convergence of learning methods for a continuous gerneralization. Furthermore, we add our applications to the large field denoted in Section 1.3. For all this purposes we define exactly the concept of sparsity[1] in Section 1.2, but for motivation we want first to answer in Section 1.1 some natural questions arising in this context. In the meantime we propose the sparsest representation as that one, needing a minimum of non-zero information.

[1] In literature the words "sparsity" and "sparseness" are used indifferently. We use the first one (for the same reasons as in [61]), since it is somehow sparser.

1.1 Why using dictionaries and sparsity?

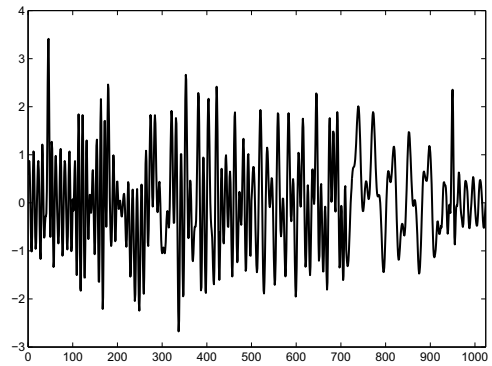
On the first sight using dictionaries instead of bases does not make sense. We add additional information to a system, that is already completely described. Nevertheless we see in the two following examples, that this is exactly the way how human brain works in several cases:

- Most probably every citizen of Bremen knows, where the ROLAND is. Somehow this is a necessary information, since legend says, Bremen would be independent, as long as the ROLAND stands at its place on market square. But if you ask a "Bremer" how to find the ROLAND, almost nobody will tell you to go to the point indicated by latitude $53°4'33.16''$ north and longitude $8°48'26.32''$ east to find it 11.5 meters above sea level (cf. [167]). This is not a question of approximation, the answer would also not be a rounded version of this data. More probable it would be like "*Leave the tram at Obernstraße, then go into direction of market place...*" or "*Cross Domshof and pass between cathedral and city hall...*". Even if someone asks this question in Antartica the answer would begin with "*First go to north western Germany...*". So one always has several different ways to describe a location, depending on situation, place and density of information. Exactly this is what we do by using a dictionary and choosing the sparsest representation.
- A second example for using dictionaries and sparse approximation is the human visual system. If we observe a landscape, we will not recognize every pixel on its own but we will recognize groups of pixels as, e.g., fields, rivers, trees or forests. Just if we concentrate on, e.g., one special tree, we will observe single leaves. This approach is typically for human behaviour also for other situations, as, e.g., the proverb "*can not see the forest for the trees*," that also exists in other languages (e.g. German: "*den Wald vor lauter Bäumen nicht sehen*"), shows. Furthermore, there is also an analogon to dictionary learning, since people adapt their dictionary to their familiar environment, e.g., Inuit have a large "dictionary" to distinguish different kinds of snow and ice, while our "standard dictionary" is more adapted to cities or fields, and this descriptions change if people move to a different environment. Naturally one of the first applications of dictionaries and sparse approximation was a simulation of the human visual cortex (cf. [121, 10, 47]) that also seems to have some physiological background (cf. [49]).

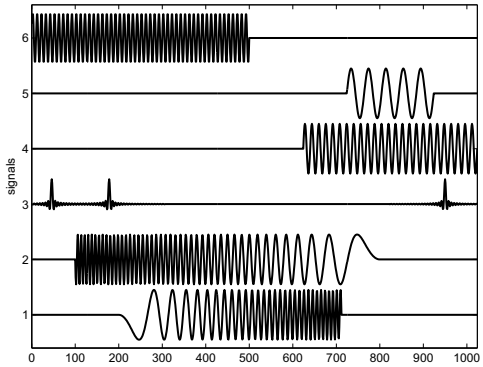
So dictionaries and sparsity are a typical way for humans to describe the world. Nevertheless, the question arises, why it should be also sensible for computer based image and signal processing. The first reason is based on the requirement of saving memory capacities. Even if our dictionary is redundant, the sparse structure of the dictionary coefficients x_j offers an efficient way to code them. In case sparsity and number of observed signals is high enough, this preface can balance or even outperform the amount of memory need for the higher number of atoms.

The second, more influencing reason is the structure of many signals and images. Human vision usually works well, the detected objects in images are reality and

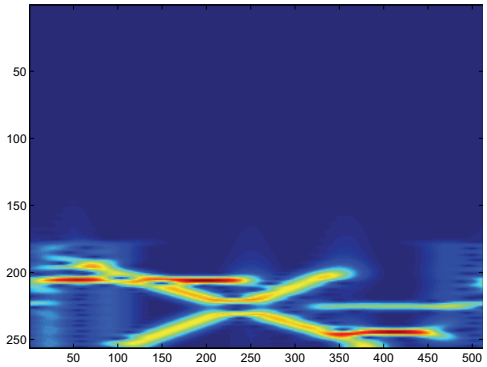
different from the surrounding ones[2]. Also acoustic or electric signals are often the superposition of different sources with different characteristics. One simple, simulated example is shown in Figure 1.1a, it is a linear combination of three sine waves with three extremely localized signals and two chirps as exemplified in Figure 1.1b. The GABOR transform in Figure 1.1c shows this structure clearly. But if we want to represent the signal in a easy way for recognizing or manipulating single parts of the signal, no basis will work well, in every case (see the FOURIER transform in Figure 1.1d) the non-zero coefficients spread out over the full x-axis. Nevertheless a sparse representation based on a dictionary, which contains sines, chirps and localized signals, does the job well.



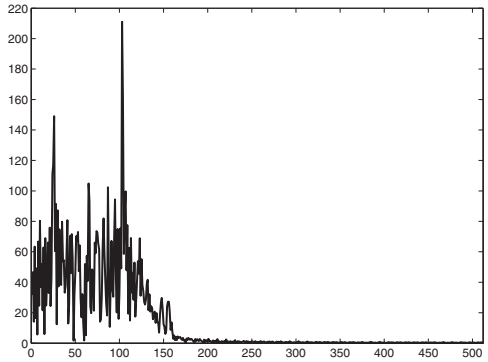
(a) Signal



(b) Summands for construction of Figure 1.1a



(c) Absolute values of GABOR transform of Figure 1.1a



(d) Absolute values of FOURIER transform of Figure 1.1a

Figure 1.1: Motivating example

The last and maybe most evident argument for using dictionaries and sparse coding is the wide field of successful applications obtained till know, as we present in Section 1.3.

[2]It would be interesting to know, if dictionary approaches are also vulnerable to the same optical illusions as human eye.

1.2 Definition of sparsity

Till now we did not define sparsity in a mathemetical exact way, we want to make up for this now. During the last years came up a several sparsity measures, everyone intending to hold in the representing sequence of coefficients, $x = (x_j)_{j\in J}$, defined analogous to Equation (1.1), as much $x_j = 0$ as possible. The maybe eldest and most basic one is probably the so called ℓ_0-"norm"[3] (cf., e.g., [69]).

Definition 1.1. *Let $x = (x_j)_{j\in J}$ for an index set J and $x_j \in \mathbb{K} \in \{\mathbb{C}, \mathbb{R}\}$ for all $j \in J$. Then the ℓ_0-"norm" of x is defined via*

$$\|x\|_0 = \sum_{j\in J} |\operatorname{sign} x_j| = \sum_{j\in J} |x_j|^0 \tag{1.2}$$

where we define $0^0 = 0$.

Clearly $\|x\|_0$ is a measure for the sparsity of x, but is has several disadvantages, especially it is non-continuous in zero, is not really a norm ($\|\alpha x\|_0 = \|x\|_0$ for every $\alpha \in \mathbb{K}\backslash\{0\}$), is non-convex and is not coercive (cf. Definitions 2.15 and 2.16). According to this there are several surrogates for $\|\cdot\|_0$, satisfying one or more of the above mentioned properties. Beside some other ones like log-based measures as used, e.g., in [12] there is a big family of measures of the form

$$\|x\|_f = \sum_J f(x_j)$$

For so called sparsity functions f according to the following definition, partly adopted from [69]:

Definition 1.2 (Sparsity functions)**.** *Let $\bar{f}$ be a non-decreasing function of the form $\bar{f} : [0,\infty) \mapsto [0,\infty)$, $\bar{f}(0) = 0$, $\bar{f} \not\equiv 0$. If additionally $t \mapsto \bar{f}(t)/t$ is non-increasing on $(0,\infty)$, then f defined as $f : \mathbb{R} \mapsto [0,\infty)$ with*

$$f(x) = \begin{cases} \bar{f}(x), & x \geq 0 \\ \bar{f}(-x), & x < 0 \end{cases}$$

is called a sparsity function of Type 1 on $\mathbb{R}$. The set of all this sparsity functions is called $\mathcal{F}_1$, the set of the $\bar{f}$ is called $\overline{\mathcal{F}}_1$.

Furthermore we define sparsity functions of Type 2 via

$$f \in \mathcal{F}_2 \Leftrightarrow \exists g \in \mathcal{F}_1,\ f(t) = \int_0^{|t|} g(\tau)\,\mathrm{d}\tau$$

We have in Subsection 5.1.1 a closer look on functions of this type and give also some examples. Here we just want to mention, that $|\cdot|^0$ is element of $\mathcal{F}_1$, while its most often used surrogate $|\cdot|^1$ is element of $\mathcal{F}_1$ and of $\mathcal{F}_2$.

Sparsity measures from $\mathcal{F}_1$ and $\mathcal{F}_2$ are used throughout this thesis and are also the basis for most of the results obtained by other authors, as for example stated in the Sections 1.3 and 2.2.2.

[3] Note the little misuse of notation that is quite common in literature for all sparsity "norms". We follow that notation throughout this thesis.

1.3 Results using dictionaries and sparse coding

Most of the results on sparse coding and approximation using dictionaries can be classified in three groups:

- Sparse coding algorithms and their properties.
- Choice and construction of dictionaries.
- Applications to real live examples.

We give in Subsection 2.2.1 a short introduction to the most important sparse coding algorithms – this are (orthogonal) matching pursuit, basis pursuit and the focal undertetermined system solver – as well as further references concerning them and their properties. Moreover there are several other algorithms like that ones given in [14, 169, 31, 157].

Since dictionaries are used, the question arises, which functions should be elements of $\mathcal{D}$. In Subsection 2.2.2 we introduce this problem in detail, but briefly there are two main approaches: Either to define $\mathcal{D}$ in advance, or to use a construction or rather learning algorithm. In the past years there was a development in direction of the second method and also we devote this thesis to the learning of dictionaries with respect to the given data.

Interestingly the field of application for sparse coding methods spread out over a wide field of subjects and grew through the past decade, for both approaches, with predefined dictionary as well as with constructed ones. We now want to summarize some applications briefly, giving a motivation of why we concern with dictionary learning.

First of all there are acoustical applications. Already in [105] the definition of matching pursuit was exemplified on recordings of human voice. Other examples are [59, 62, 129, 125, 65, 164, 14], working on themes like separation of audio sources till to separating and classifying different notes in a piano recording. We also propose a similar application in Section 6.3 of this thesis.

Another wide field of applications is image processing. One part of this is the classification of image contents, textures and 3D objects, as for example given in [146, 162, 111], or also recognition of faces and silhouettes (cf. [106, 104]) and inpainting as in [40]. Closely related to this are also methods of image coding like in [149, 48, 145], which are often compared to the JPEG2000 algorithm (cf. [156]), that also uses overcompleteness and sparsity, although in a slightly different context. Another very early application was the denoising of images, already proposed in [22] and later on also in [39, 45]. Of course also video signals can be coded using dictionary techniques (cf. [135, 119, 113, 131]).

Leaving the area of general signal and image processing and proceeding to other fields, we find, e.g., applications in mathematics, like the numerical solution of differential equations in [170]. But there are also applications in geosience, as in [128],

where meteorological data from northern pacific are analyzed, or in [77], about seismic data. Rarer implemetations are in subjects like astronomy, for denoising of images and detection of astronomical objects like in [147], or in finance (see [56]) trying to find structures in financial data for disproving the efficient market hypothesis.

Further applications can be found in medicine, where several articles (e.g., [144, 158, 33, 13]) address the analysis of ECG and EMG data, interestingly in this framework mostly learned dictionaries are used. Another medical utilization of learned dictionaries lies in the detection of masses in mammograms, as proposed in [75].

To complete this list of application we want to state also some from the area of engineering: In electrical engineering there are, e.g., compressions of impendance matrices (see [142]) or approximations of inner structures of objects by observing the conductivity (e.g., in [16]). In the case of mechanical or civil engineering the utilization are mostly to monitor different machineries like bearings (cf. [66]), drill failures (cf. [50]) or detecting the breaking sounds of concrete as exemplified in [37]. Also our own main application concerns an example in the engineering field, the condition monitoring of linear guideways by analysis of its operating noise.

1.4 Substance and structure of this thesis

As suggested above we propose as a first main topic of this thesis an algorithm, called MODW, for learning a dictionary $\mathcal{D} = \{d_l\}_{l=1}^L \subset \mathbb{R}^n$ from a given set of signals $\mathcal{Y} = \{y_i\}_{i=1}^N \subset \mathbb{R}^n$. The dictionary has the special structure of a union of several wavelet bases:

$$\begin{aligned}\mathcal{D} &= \{d_l\}_{l=1}^L = \bigcup_{j=1}^J \mathcal{D}_j \\ &= \bigcup_{j=1}^J \left\{ \psi_j\left(\frac{\cdot - r_1}{r_2}\right), \varphi_j\left(\frac{\cdot - r_1}{2^{j'}}\right) \middle| |r_1| \le R_\rho; r_2 = 2^\rho; \bar{j} \le \rho \le j'; r_1, \rho \in \mathbb{Z} \right\}.\end{aligned}$$

We point out this structure, as well as its prefaces, more detailed in the beginning of Chapter 4 and Section 4.1.

The second main topic of this thesis is the convergence analysis of dictionary learning in the continuous case. This results in the course of Chapter 5 in minimizing a functional E of the form

$$\begin{aligned}E(d,x) &= \int_I \left(\int_\Omega \left(y(\omega,l) - \int_{\Omega'} d(\omega,\omega') x(\omega',l) \,\mathrm{d}\mu'\omega' \right)^2 \mathrm{d}\mu\omega \right. \\ &\qquad \left. + \lambda_x \int_{\Omega'} f(x(\omega',l)) \,\mathrm{d}\mu'\omega' \right) \mathrm{d}\nu l \\ &\quad + \lambda_d \left(\int_{\Omega'} \left(\int_\Omega d(\omega,\omega')^2 \,\mathrm{d}\mu\omega \right)^{p'/2} \mathrm{d}\mu'\omega' - c \right)^2 \\ &\quad + (\lambda_1 g_1(\|x\|_p) + \lambda_2 g_2(\|\nabla x\|_p)) + \lambda_3 g_3(\|\nabla d\|_{(2,p)})\end{aligned}$$

according to the functions d and x.

For obtaining this results we cite some prelimiary knowledge in Chapter 2. First we state there definitions and conclusions on wavelets and their lifting scheme in Section 2.1, while Section 2.2 is devoted to a broad summary of the sparse coding and dictionary framework, especially dictionary learning algorithms in Subsection 2.2.2. Some results on function spaces and functional analysis in Section 2.3 complete that chapter.

Chapter 3 contains as a minor topic of this thesis, some preceding results on matching a wavelet to one given signal based on the lifting scheme[4]. The idea for this results has originally been developed in [160], while we added the less shift-variant version given in Section 3.3.

As already mentioned, Chapter 4 includes the first main topic of the thesis, the learning algorithm for wavelet dictionaries. We develop the theoretical fundamentals in Section 4.1 partly using the similar ideas as in Chapter 3, while Sections 4.2 and 4.3 contain several implementational details and the use of a priori knowledge.

The convergence analysis of continuous dictionaries is topic of Chapter 5. After deriving the functional $E(d, x)$ in Subsection 5.1.1 we prove in the remainder of Section 5.1 results on the existence of a minimum for fixing one variable, as well as for free d and x. Later on Section 5.2 contains methods for computing that minima based on conditional gradient methods. For the reason of readability we moved some additional information and a technical lemma of this chapter to Appendix B.

Before we summarize this thesis in Chapter 7 and give a short outlook to open questions, we want to apply in Chapter 6 the results of Chapters 3 and 4. Here Section 6.1 offers some examples on the dictionary learning algorithm using simulated data. Afterwards we state our main example, condition monitoring of linear guideways, in Section 6.2. In that context we use both, the wavelet matching from Chapter 3 and the results of Chapter 4. Please note, we added the short Appendix A for explaining linear guideways, especially our measurement device and the bases of condition monitoring. For the last application we refer to Section 6.3, where we apply the MODW algorithm to piano recordings, intending to reconstruct the original score.

[4]Please note, we use throughout this thesis lifting for construction of wavelets only, not for fast wavelet transform as suggested in [153, Section 6].

Chapter 2

Preliminaries

For preparation of the theoretical and applicational results, we give in Chapters 3 to 6, we need to introduce here some results from literature, used in the course of this thesis. Furthermore we use the chance to establish some notations for the other parts of the thesis.

Since the theoretical results in this thesis are based on three quite different areas of mathematics we preserve that partition also in this chapter. So the first section, 2.1, gives a short introduction to the well known framework of wavelets, focused on the theory of lifting. In Section 2.2 we give an overview about the theory of dictionaries, additionally some learning algorithms are presented in Section 2.2.2, as basis for classifying the results in Chapter 4. We finish this chapter giving some definitions and results from the framework of applied functional analysis in Section 2.3.

2.1 Wavelets

In this section we want to give a short overview about the basics of wavelet analysis. Furthermore we give in Subsection 2.1.2 an introduction to the framework of lifting, extensively used in the following Chapters 3 and 4. Lifting is there be the basis for construction of single wavelets or rather of whole dictionaries consisting of wavelets.

2.1.1 Wavelet analysis

This introduction is mostly given for the sake of completeness and for introduction of notation. For more detailed information on the theoretical background of wavelet methods see, e.g., [101, 28, 100].

Definition 2.1 (Wavelet basis)**.** *A dyadic wavelet basis is a* Riesz *basis (cf. [101, p. 595]) consisting for fixed $j' \in \mathbb{Z}$ of two types of functions: $\varphi_{j',k}$ and $\psi_{j,k}$, $j,k \in \mathbb{Z}$, $j \leq j'$. They are scaled and shifted versions of the scaling function φ and of the*

wavelet ψ:

$$\varphi_{j',k} = 2^{-j'/2}\varphi\left(\frac{\cdot - k}{2^{j'}}\right) \quad \text{and} \quad \psi_{j,k} = 2^{-j/2}\psi\left(\frac{\cdot - k}{2^{j}}\right), \quad j,k \in \mathbb{Z},\ j \le j'$$

Furthermore φ and ψ are usually normalized by $\|\varphi\|_2 = \|\psi\|_2 = 1$. The scaling function and the wavelet of a dyadic wavelet basis satisfy refinement equations, see [101, pp. 221 et sq.]:

$$\varphi = \sum_{k\in\mathbb{N}} h_k \varphi(2\cdot - k) \tag{2.1}$$

and

$$\psi = \sum_{k\in\mathbb{N}} g_k \varphi(2\cdot - k)\ , \tag{2.2}$$

(h_k) and (g_k) are called the filters of φ and ψ respectively. The filters and the refinement equations provide a fast method for calculating the wavelet transform ([101, p. 255]). Up to normalization, the scaling function, respectively the wavelet, is uniquely defined by the scaling coefficients (h_k), respectively by the wavelet coefficients (g_k). On the other side not every pair (h_k) and (g_k) defines a wavelet basis, some constraints for this as well as the fast algorithm are given specialized for biorthogonal wavelets below.

Even more, in most cases an analytically closed-form expression that defines the wavelet as a function of the filter coefficients can not be given. Moreover, for most filters, the corresponding wavelets do not exist in L^2 and in most other cases wavelet functions are visualized by pointwise values, which can be obtained, e.g., by cascade algorithms ([28, Chapter 6.5]). Due to our approach in Chapters 3 and 4 we do not need to care about the existence of the wavelets and scaling functions, we define them in Subsection 2.1.2 as finite sum of already existing functions.

For analytical purposes it is convenient to utilize the discrete Fourier transforms of the discrete filters

$$h(\omega) = \sum_{k\in\mathbb{Z}} h_k e^{-i\omega k} \quad \text{and} \quad g(\omega) = \sum_{k\in\mathbb{Z}} g_k e^{-i\omega k}\ .$$

Wavelets $\psi_{j,k}$ are often called discrete wavelets, or more accurate dyadic wavelets, in order to distinguish them from the less restrictive definition of a continuous wavelet $\psi_{a,b},\ a,b \in \mathbb{R}$. Additionally there are rarely used discrete, non-dyadic wavelets. Different techniques for adapting discrete or continuous wavelet concepts to bounded domains $\Omega \subset \mathbb{R}^d$ have been analyzed, see, e.g., [23].

Lemma 2.2 (Multiresolution analysis)**.** *A discrete wavelet basis introduces a hierarchy of nested function spaces, a so called multiresolution analysis, which are indexed by $j \in \mathbb{Z}$: $V_j := \operatorname{span}\{(\varphi_{j,k}) \mid k \in \mathbb{Z}\} \subset L_2(\mathbb{R})$. The wavelets $(\psi_{j,k})_{k\in\mathbb{Z}}$ then span a complement W_j of V_j in V_{j-1}. Accordingly, the wavelet decomposition of a function $f \in V_{j'}$ is given by*

$$f = \sum_{k\in\mathbb{Z}} \alpha_{j',k}\varphi_{j',k} + \sum_{j\le j'}\sum_{k\in\mathbb{Z}} f_{j,k}\psi_{j,k}. \tag{2.3}$$

Orthonormal wavelet bases, which satisfy $V_j \perp W_j$, $\langle \varphi_{j',k} \,|\, \varphi_{j',k'} \rangle = \delta_{k,k'}$ and $\langle \psi_{j,k} \,|\, \psi_{j,k'} \rangle = \delta_{k,k'}$, are particularly convenient. But below we want to stress the more flexible biorthogonal case. It is defined by an additional second wavelet basis, generated by functions $\{\tilde{\varphi}, \tilde{\psi}\}$ and filters $\{\tilde{h}, \tilde{g}\}$ respectively, satisfying $V_j \perp \tilde{W}_j$, $\tilde{V}_j \perp W_j$, $\langle \varphi_{j',k} \,|\, \tilde{\varphi}_{j',k'} \rangle = \delta_{k,k'}$ and $\left\langle \psi_{j,k} \,\middle|\, \tilde{\psi}_{j,k'} \right\rangle = \delta_{k,k'}$ [1]. Here the expansion coefficients and filters are defined via:

Lemma 2.3.

$$\alpha_{j',k} = \langle f \,|\, \tilde{\varphi}_{j',k} \rangle \quad \text{and} \quad f_{j,k} = \left\langle f \,\middle|\, \tilde{\psi}_{j,k} \right\rangle \ . \tag{2.4}$$

The constraints on the filters are given by $\tilde{g}_k = (-1)^k \overline{h_{1-k}}$, $g_k = (-1)^k \overline{\tilde{h}_{1-k}}$ *and*

$$\overline{h(\omega)}\tilde{h}(\omega) + \overline{h(\omega+\pi)}\tilde{h}(\omega+\pi) = 2 \ . \tag{2.5}$$

The fast decomposition algorithm runs as: (cf. [101, p. 268])

Algorithm 2.4 (Fast wavelet transform)**.** *Let* b_n *be a discrete input signal, sampled at intervals* $N^{-1} = 2^L$*, then there is a function* $f \in V_L$ *with* $\alpha_{L,k} = \langle f \,|\, \varphi_{L,k} \rangle = N^{-1/2} b_n$*. The wavelet coefficients are computed by successive (discrete) convolutions with* h *and* g *(in the complex case with* $\bar{h}$ *and* $\bar{g}$*). Then the wavelet analysis calculates as*

$$\alpha_{j+1,n} = \alpha_j \star h_{2n} \quad \textit{and} \quad f_{j+1,n} = \alpha_j \star g_{2n}$$

The reconstruction is given with the dual filters $\tilde{h}$ *and* $\tilde{g}$ *as*

$$\alpha_j = \check{\alpha}_{j+1} \star \tilde{h}_n + \check{f}_{j+1} \star \tilde{g}_n \ ,$$

where $\check{x}$ *is defined as*

$$\check{x}_n = \begin{cases} x_p & \text{if } n = 2p \\ 0 & \text{if } n = 2p+1. \end{cases}$$

In case the b_n includes N non-zero samples, the transform as well as the reconstruction is calculated within $O(N)$ operations.

Biorthogonal wavelets can be symmetric [28, pp. 258 et sq.], which is a major advantage in some applications. Moreover, biorthogonal wavelets appear naturally in the construction of wavelets via lifting schemes to be used in our algorithm for matching wavelets and dictionary learning.

2.1.2 Lifting

As mentioned before in Chapter 1.4, the purpose of Chapters 3 and 4 is to develop a procedure for constructing data dependent discrete wavelet bases or rather dictionaries, composed of them, and to apply this for tasks of pattern detection.

Our approach is based on lifting, so we shortly want to review the basics of wavelet lifting, which was introduced and analyzed in [153, 152, 154, 29]. The lifting

[1] especially $\left\langle \psi_{j,k} \,\middle|\, \tilde{\psi}_{j^*,k'} \right\rangle = \delta_{k,k'} \delta_{j,j^*}$

scheme always starts with a given set of biorthogonal wavelets and constructs a new quadruple of biorthogonal wavelets. The resulting quadruple is always denoted by $\{h, \tilde{h}, g, \tilde{g}\}$, the wavelets or scaling functions which are updated in the lifting scheme are indexed in their original form by an upper 0, e.g., g^0.

Lemma 2.5 (Lifting Scheme I)**.** *Let a set of finite biorthogonal filters $\{h, \tilde{h}^0, g^0, \tilde{g}\}$ and a trigonometric* Laurent *polynomial $s(\omega)$ be given.*

Then, the set of functions $\{h, \tilde{h}, g, \tilde{g}\}$ defined by

$$\begin{aligned} \tilde{h}(\omega) &= \tilde{h}^0(\omega) + \tilde{g}(\omega)\overline{s(2\omega)} \\ g(\omega) &= g^0(\omega) - h(\omega)s(2\omega) \ . \end{aligned}$$

also constitutes a set of finite biorthogonal filters.

This result can also be formulated in terms of wavelets and scaling functions:

Lemma 2.6 (Lifting Scheme II)**.** *Let a set of formally biorthogonal wavelets $\{\psi^0, \tilde{\psi}^0, \varphi, \tilde{\varphi}^0\}$ and a trigonometric* Laurent *polynomial $s(\omega)$ with coefficients s_k be given. Define $\{\psi, \tilde{\psi}, \varphi, \tilde{\varphi}\}$, by*

$$\begin{aligned} \psi(x) &= (\psi^0 + s \star \varphi)(x) := \psi^0(x) - \sum_k s_k \varphi(x-k) \qquad (2.6) \\ \tilde{\varphi}(x) &= 2\sum_k \tilde{h}^0_k \tilde{\varphi}(2x-k) + \sum_k s_{-k}\tilde{\psi}(x-k) \\ \tilde{\psi}(x) &= 2\sum_k \tilde{g}_k \tilde{\varphi}(2x-k) \ , \end{aligned}$$

Then $\{\psi, \tilde{\psi}, \varphi, \tilde{\varphi}\}$ constitutes a set of formally biorthogonal scaling and wavelet functions.

The term "formally" is chosen, as it is not clear, due to the remark following Definition 2.1, that the new dual wavelets and scaling functions $\tilde{\varphi}, \tilde{\psi}$ (and even the old one) are existing at all in L_2 and form a Riesz basis.

Of course for applicational frameworks it is necessary to obtain some smoothness properties of the lifted wavelet system: After exploiting the defining scaling equation (2.2):

$$\psi^0 = \sqrt{2} \sum_{n=-\infty}^{\infty} g_n \varphi(2 \cdot -n) \ ,$$

one immediately sees, a finite length of (g_n) implies that the order of differentiability of ψ^0 equals that of φ. By (2.6) also ψ has the same order of differentiability provided that s is not an infinite series. Furthermore its smoothness is depending on the number of vanishing moments $\tilde{p}$ of $\tilde{\psi}$ given by

Lemma 2.7. *Let*

$$\tilde{h}(\omega) = \left(\frac{1+e^{-i\omega}}{2}\right)^{\tilde{p}} \hat{l}(\omega) \quad p \in \mathbb{N} \qquad (2.7)$$

where $\hat{l}$ is a finite filter. Define even more $B = \sup_{\omega\in[-\pi,\pi]} \left|\hat{l}(\omega)\right|$. Then we have φ (and with it also ψ^0 and ψ) to be LIPSCHITZ *α for all*

$$\alpha < \tilde{p} - \log_2 B - 1.$$

See [101, pp. 268-269] for more details.

For the approximations following in next chapters simple and symmetric filters $\{h, \tilde{h}^0, g^0, \tilde{g}\}$ are preferable. Starting with Equation (2.7), using Equation (2.5) and following the course of [101, p. 270] we know:

Lemma 2.8. *Let h and $\tilde{h}$ be symmetric biorthogonal wavelet filters, then they are of the form*

$$\begin{aligned} h(\omega) &= \sqrt{2}e^{-i\epsilon\omega/2}\left(\cos\frac{\omega}{2}\right)^p L(\cos\omega) \\ \tilde{h}(\omega) &= \sqrt{2}e^{-i\epsilon\omega/2}\left(\cos\frac{\omega}{2}\right)^{\tilde{p}} \tilde{L}(\cos\omega). \end{aligned} \tag{2.8}$$

With $p+\tilde{p}$ even, $\epsilon = 0$ for $(p,\tilde{p})$ even and $\epsilon = 1$ otherwise and the following additional constraints on the product of the polynomials L and $\tilde{L}$:

$$L(\cos(\omega))\tilde{L}(\cos(\omega)) = P(\sin^2(\omega/2))$$

where the polynomial P satisfies

$$(1-y)^q P(y) + y^q P(1-y) = 1 \quad \textit{for} \quad q = (p+\tilde{p})/2 \quad \textit{and for all} \quad y \in [0,1]\ .$$

Remark 2.9.

- *For the simple case $L \equiv 1$ and $\tilde{L}$ chosen to satisfy all constraints we obtain the so called CDF wavelets (cf. [28, Chapter 8]) with B-splines φ of order p as scaling functions. Also the wavelet ψ is a spline of the same order with an additional knot. For an example see Figure 2.1*
- *Theoretically it is possible as stated in [29] to construct every wavelet via lifting starting with the lazy wavelet and iteratively lifting h and $\tilde{h}$. There also a method is presented, based on the* EUCLIDean *algorithm, to reconstruct this sequence of liftings. So the set of possible resulting wavelet functions after lifting depends only on the common divisors of g and h.*

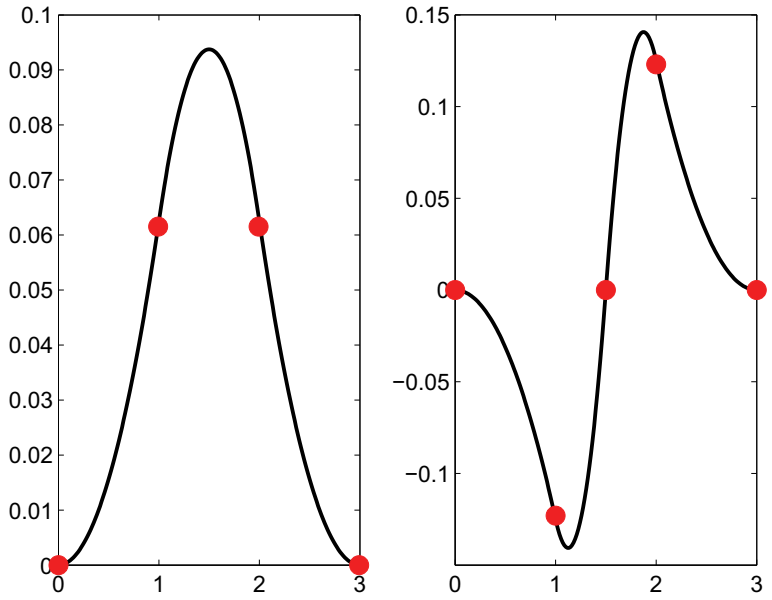


Figure 2.1: CDF scaling function (left) and wavelet(right) for parameters $p = 3$ and $\tilde{p} = 1$, the knots of the splines are marked.

2.2 Dictionaries

2.2.1 Sparse coding

In this subsection we want to give a detailed introduction to the subject of sparse coding, as briefly presented in Section 1.2.

In case there is a function y element of a HILBERT space H, it is a well known method, to describe y using a orthonormal basis $\{e_j\}_{j\in\tilde{J}}$:

$$y = \sum_{j\in\tilde{J}} x_j e_j \quad \text{where} \quad \forall j \in \tilde{J} : \; x_j \in \mathbb{K} \tag{2.9}$$

with $\mathbb{K} \in \{\mathbb{C}, \mathbb{R}\}$. This description is unique. If we use instead of the basis a so called (complete) dictionary (cf. [105, 22, 163]), the situation changes:

Definition 2.10 (Dictionary)**.** *Let H be a real or complex* HILBERT *space, Furthermore let J be an index set, then the set $\mathcal{D} = \{d_j\}_{j\in J} \subset H$ of normed atoms $\|d_j\|_H = 1$ is called a dictionary on H. If furthermore $H = V := \operatorname{span}\{d_j\}$, the dictionary is called complete.*

Now the description analog to Equation (2.9) is not unique anymore. So the question arises which choice of $x := (x_j)_{j\in J}$ is the most convenient one. In this case we want to have x as "sparse" as possible. Sparsity of x denotes, to have as much

$x_j = 0$ as possible. This can be achieved by minimizing the ℓ_0-"norm"[2] defined in Equation (1.2). Instead of minimizing $\|\cdot\|_0$ there is a wide field of other sparsity measures, e.g., $\|\cdot\|_1$. For a closed theory of sparsity measures see Subsection 5.1.1. Defining the linear mapping $D : \ell_\infty \mapsto H$

$$Dx = D(x_j)_{j\in J} = \sum_{j\in J} x_j d_j$$

we can formulate the problem in two differnet ways as given in [163]:

$$\min_{x\in\ell_\infty} \|x\|_0 \quad \text{while} \quad y = Dx \tag{2.10}$$

or

$$\min_{x\in\ell_\infty} \|e_x\|_H := \min_{x\in\ell_\infty} \|y - Dx\|_H \quad \text{while} \quad \|x\|_0 \leq m \tag{2.11}$$

While Equation (2.10) is the problem in context of exact solutions, the second one is especially applied to noisy functions y.

Solving both, Equations (2.10) and (2.11), is in generally an NP hard problem (cf. [102]), so there exist several methods for producing an approximative solution, as already mentioned in Section 1.3. The most important and best known ones are the orthogonal matching pursuit (OMP), the basis pursuit (BP) and the focal underdetermined system solver (FOCUSS), all of them having a number of variations, generalizations and antecessors, as, e.g., given in [105, 82, 46, 55, 44, 149, 98].

The eldest of the mentioned algorithms is the OMP (see also [103] or [126]), a greedy algorithm, choosing every step one atom $d_i \in \mathcal{D}$ of the dictionary defined by

$$\begin{aligned} d_i &= \operatorname*{argmax}_{d\in\mathcal{D}} |\langle R_i y \,|\, d\rangle| \\ R_i y &= y - P_{\operatorname{span}\{d_0,\ldots,d_{i-1}\}} y, \end{aligned} \tag{2.12}$$

here P denotes the orthogonal projection. Furthermore the coefficients x_l are fixed, after achieving m atoms or reaching $\|y - Dx\|_H = 0$, as the coefficients of this projection. In special cases there are methods to speed up the OMP, especially there is for a structured dictionary in general no necessity to compute all scalar products $\langle R_i y \,|\, d\rangle$ in Equation (2.12), instead they can be computed as weighted sum of the $\langle R_{i-1} y \,|\, d\rangle$. Interestingly there is also a version of this algorithm for non-discrete dictionaries, see [103] for details. A further variant is the otimized OMP (OOMP) (cf. [137]) changing the choice of the atom. Instead of the maximal scalar product between residual and atom it chooses the maximum of the absolute value of the projections onto the atoms, i.e.

$$\operatorname*{argmin}_{d_j} = \|y - P_{\operatorname{span}\{d_0,\ldots,d_j\}} y\|.$$

This approach needs generally more computation time than the original OMP, but in case the dictionary is composed from several biorthogonal wavelet bases like in Chapter 4, the dual transform gives directly this values (cf. also Page 50).

[2] Cf. Footnote 3 on Page 4

The two other mentioned algorithms optimize given starting coefficients parallely for all atoms of the dictionary. This makes them slower for choosing just a few representing atoms for one signal (e.g., as in Equation (2.11) with small m), but preferable for computing longer representations. The BP (see [22]) replaces the ℓ_0-norm in Equation (2.10) and (2.11) by the ℓ_1-norm, so the problem becomes convex and easy solvable, e.g., by simplex algorithm or interior point methods. Similar FOCUSS (cf. [136]) is using an ℓ_p-norm, $0 < p < 1$ and an iterative matrix multiplication. This l_p-norms are more similar to the original ℓ_0-norm than the l_1-norm, but give non-convex problems and by this the danger of local minima. [168] handles with the connections between BP, FOCUSS and some probabilistic methods.

Since this algorithms has been developed, additional results on their properties came up. For example there are for all this algorithms results concerning the convergence speed (e.g., in [70, 99]). Moreover there exist results (e.g., in [67, 36, 163, 35]), proving in dependency of $\mathcal{D}$ upper bounds for the value of $\|x\|_0$ up to that the results of the coding algorithms (BP, OMP) are equal to exact results got by searching all possibilities. Generally this bound is higher, if the coherence of the dictionary (the maximal absolute values of inner product of two atoms) is small. For checking the optimality of a given sparse representation x, there is the bound $\|x\|_0 < \text{spark}(\mathcal{D})/2$ up to that the x is the unique representation with minimal ℓ_0-norm ($\text{spark}(\mathcal{D})$ is the minimal cardinality of an linearly dependent subset of $\mathcal{D}$, cf. [67])[3]. Furthermore there are in [61] approximations of the the distances $\|x - x_B\|_2$, $\|x - x_B\|_\infty$ and $\|D(x - x_B)\|_H$, where x_B denotes the best possible solutions. Computing all this bounds and approximations needs in general solving a combinatorical setting, so they are poorly usable for the following learning algorithms in the case of applicational size problems.

2.2.2 Dictionary learning

In the last subsection we introduced some results and algorithms about finding a sparse representation x of one signal y or analogous for a whole family of signals $\mathcal{Y} = \{y_l\}_{l=1}^N \subset H$ in the case of given dictionary $\mathcal{D}$. But before doing so, the question arises, how to choose $\mathcal{D}$. Basically there are several criteria that should be considered:

- Special structure of the dictionary giving the opportunity to speed up the sparse coding algorithms.
- Mathematical models or prior results giving evidence for the structure of y or $\mathcal{Y}$.
- The quantity of the error estimates, e.g., $\text{spark}(\mathcal{D})$ and coherence as aforementioned.
- The current signals $\mathcal{Y}$ to achieve optimal sparsity of x.

[3] Also advanced properties, closely related to coherence and $\text{spark}(\mathcal{D})$ are used.

There is a large amount of predefined dictionaries in literature with special structure like wavelets, FOURIER or GABOR transforms (e.g., in [105, 58, 53]). Other examples are combinations of a fixed number of orthogonal bases (e.g., in [68, 74]), curvelets or contourlets (cf. [148, 109]). Dictionaries chosen like this can be used to satisfy the first two points of the above listing. But on the other side the quality of the sparsest solution is highly depending on the model or rather the inner structure of the dictionary.

To choose a dictionary just based on the error bounds is generally not advisable. The preface of optimal bounds is in general small in comparison with data or model based optimization (if this produces not extreme small bounds), since also sparsity measures greater than this bound have a high probability to be optimal (cf. [35, 38]). Nevertheless, one example is given in [151]. Furthermore the properties of some standard dictionaries had been analyzed, e.g., in [127, 30]. Up to our knowledge there is no attempt to combine this approach with the other ones, also due to the combinatorical character of the error bounds mentioned in Subsection 2.2.1.

The third possibility is to construct dictionaries that way, to fit the given data as well as possible; this way is known as dictionary learning. Generally this atoms have, due to their way of construction, no inner structure usable for fast computations, but the sparsity of representation is in general much better in comparison to standard dictionaries. In cases the number N of signals is high enough, it could be sensible to afford the additional calculations for a sparser representation. During the past years came up some approaches to import restrictions to the dictionary learning, especially focused on shift-invariance. But other restrictions are almost not considered till now.

Beside some variants there is approximately a handful of better known algorithms for learning dictionaries. We want to give a short overview about them, since we establish in Chapters 4 and 5 a new algorithm and convergence results. Since all the listed algorithms work just on $H = \ell_2(\{1, \dots n\})$, we identify the mapping D with the corresponding matrix $\mathbf{D} \in \mathbb{R}^{n \times K}$, $K < N$. Furthermore we denote by $\mathbf{Y} \in \mathbb{R}^{n \times N}$ the matrix whose i-th column equals $y_i \in \mathcal{Y}$, analogous $X \in \mathbb{R}^{K \times N}$ is the matrix of coefficients.

Fundamentally spoken there are two philosophies of dictionary learning: Either to maximize an expectation P or to minimize an error E. There are several connections between this two points of view, but throughout this thesis we concern with the later one. So just for the sake of completeness and as basis for comparisons we state first also some probabilistic approaches.

Maximum likelihood (ML) method

This method, introduced e.g., in [122, 96], is the eldest probabilistic algorithms of dictionary construction. Actually it is inspired by some earlier research in the neuronal network community, originally based on ideas in [6, 8] concerning redundancy reduction. In ML the signals of $\mathcal{Y}$ are modelled by

$$y_i = \mathbf{D}x_i + \epsilon,$$

where ϵ is a Gaussian noise with variance σ^2 and zero mean. For signals being independent of each other and defining the probability of a combination $(y_i, \mathbf{D})$ as integral over all x, we get

$$P(\mathbf{Y}|\mathbf{D}) = \prod_{i=1}^{N} P(y_i|\mathbf{D}) = \prod_{i=1}^{N} \int P(x)P(y_i|x,\mathbf{D})\,\mathrm{d}x\,.$$

The optimal $\mathbf{D}$ is that one, having the maximal probability. As there are calculative problems to solve this, usually approximative optimizations are considered. Typically they predefine a distribution of the elements of X as e.g., Gaussian as in [96], δ-approximation (cf. [120, 80]), evaluation of the maximum (e.g., in [121]) or Monte Carlo approximation (see [139, 123]), but also variational methods had been used (cf. [52, 81, 112]). So applications up to video coding (see [119]) were possible. Last year ML became basis of an advancement resulting in a shift-invariant dictionary (cf. [12], giving also an extensive composition of the mentioned approximations).

Maximum a-posteriori probability (MAP)

The MAP is a second representant for an stochastic dictionary learning algorithm, it is exemplified in [91, 116]. In contrast to the ML algorithm it uses the a posterior probability $P(\mathbf{D}, X|\mathbf{Y})$. With Bayes' rule this gives

$$P(\mathbf{D}, X|\mathbf{Y}) = cP(\mathbf{Y}|\mathbf{D}, X)P(\mathbf{D})P(X)/P(\mathbf{Y}).$$

The following calculation is similar to that in the case of ML, additionally a further assumption for $P(\mathbf{D})$ can be chosen. [91] gives two interesting possibilities, both giving an own iterative rule for computing the dictionary. In both cases updating of the dictionary alternates with a computation of X by FOCUSS. The first possibility is to fix $\|\mathbf{D}\|_F = 1$. If we want instead of this to have every d_j satisfying $\|d_j\|_2 = 1$, $\forall\ 1 \leq j \leq K$, every atom has to be updated individually via

$$d_j^{n+1} = d_j^n + \gamma(I - d_j^n d_j^{nT})(\mathbf{Y} - \mathbf{D}X)x_j^T,\ \gamma > 0,$$

with x_j denoting the j-th row of X, giving a double iterative algorithm.

Also for this algorithm there came up during the last years a bundle of applications (e.g., [13]) and shift-invariant versions (cf. [124, 139]). The last two papers propose a wide definition of wavelets, not satisfying the constraints from Section 2.1.1.

Methods using probabilistic approaches have in general the preface to introduce a lot of knowledge about the signal or a model into the assumptions on dictionary and error. On the other side they often result in complicate equations, connected to an highly approximative solving algorithms. As an alternative we introduce now some deterministic algorithms. Here introducing prior knowledge is in general just possible introducing it into the starting-values or by introducing weights in the sparse coding as in [34] or [55]. For a more detailed treatment of this options, see Chapter 4.

Method of optimal direction (MOD)

The MOD ([41, 43]) is the eldest non-stochastic algorithm, but it can be interpreted as a special approximation of the ML, fixing the number of non-zero coefficients. Aiming to find a best solution of Equation (2.11) for a set of signals $\mathcal{Y}$ MOD minimizes:

$$(\mathbf{D}^*, X^*) = \underset{\mathbf{D},X}{\operatorname{argmin}} \|\mathbf{Y} - \mathbf{D}X\|_F^2, \quad \text{while} \quad \max_{1\leq i\leq N} \|x_i\|_0 \leq T \ ,$$

$\|\cdot\|_F$ denotes the FROBENIUS norm. This algorithm is related to the K-mean algorithm (cf. [51], also known as generalized LLOYD algorithm), like that it is iterative and divides every iteration into two steps:

1. Fix $\mathbf{D}$ and find the best, sparse X
2. Fix X and find the best $\mathbf{D}$.

For the first step every known sparse coding method (see Section 2.2.1) is usable on every column of $\mathbf{Y}$, e.g., orthogonal matching pursuit or variants of the others, giving a fixed $\|x_i\|_0$. The second step is realized by finding the zeros of the derivative of $\|\mathbf{Y} - \mathbf{D}X\|_F^2$ in directions of all components of $\mathbf{D}$. This leads to

$$\mathbf{D} = (\mathbf{Y}X^T)(XX^T)^{-1}.$$

The inverse matrix exists, as the derivative has for reasonable data $\mathbf{Y}$ always exactly one zero indicating a minimum (see [41]).

There are several applications for the MOD and closely related algorithms of gradient descent like, e.g., in [76, 75, 144, 42]. Furthermore there are some results in [73, 166] about a shift-invariant dictionary, using the ℓ_1 norm for indicating sparsity.

K-SVD

Similar to MOD the K-SVD algorithm (stated in [3, 4]) tries to minimize the value of

$$\|E\|_F^2 = \|\mathbf{Y} - \mathbf{D}X\|_F^2$$

in dependency of $\mathbf{D}$ and X, while the values of all $\|x_i\|_0$ are bounded by T. The minimization follows the same steps as in MOD, the difference is just established in the second step: Here the coefficients of X are not necessarily constant. The atoms are updated sequentially, for every d_j the norm of E is calculated as

$$\|E\|_F^2 = \|\mathbf{Y} - \mathbf{D}X\|_F^2 = \left\| \mathbf{Y} - \sum_{\substack{i=1 \\ i\neq j}}^{K} d_i x_i^T - d_j x_j^T \right\|_F^2 = \|E_j - d_j x_j^T\|_F^2 \ .$$

Restricting x_j to the non-zero elements and E_j to the corresponding columns, we search for minimizing

$$\|E_{j,R} - d_j x_{j,R}\|_F^2.$$

This is possible by computing a singular value decomposition of $E_{j,R} = U\Delta V^T$ and choosing d_j as first column of U and $x_{j,R}$ as first column of V times $\Delta_{(1,1)}$. In the past two years there where several applications of K-SVD and a related algorithm (see [141]), especially in the area of image and video coding (cf. [39, 131]).

union of orthonormal bases (UOB)

In difference to all previous algorithms this one (exemplified in [94]) requires a special structure of the dictionary, it is composed from several orthonormal bases of the same space:

$$\mathcal{D} = \{\mathcal{D}_1 \dots \mathcal{D}_L\},\ \forall\ 1 \leq l' \leq L :\ \mathcal{D}_{l'} = \{d_{i,l'}\}_{i=1}^{n} \text{ is orthonormal basis}$$

The preface of this idea is the simple way to find the coefficients for sparse coding (see [94] for more details). The dictionary itself is computed iteratively, by alternately computing the coefficients and updating the dictionary. This update is itself sequentially for the $\mathcal{D}_{l'}$ and similar to K-SVD: For the current l' and fixed X the modified errormatrix $E_{l'} = \mathbf{Y} - \sum_{i \neq l'} \mathbf{D}_i X_i$ is computed, then the singular value decomposition of $E_{l'} X_{l'}^T = U\Delta V^T$ gives the update by $\mathbf{D}_{l'} = UV^T$.

There are also algorithms, that do not fit into the above given scheme of minimizing E or maximizing P. As an example we want to state the following:

Matching of time invariant filters (MoTIF)

In contrast to the previous algorithms, MoTIF (introduced first in [83], exemplified and applied in [63, 114, 113]) does not search directly for the dictionary $\mathcal{D}$. Instead it constructs a smaller set $\mathcal{G} = \{g_i\}_{i=1}^{K}$ of functions, the dictionary is defined afterwards as the set of all integer shifts of the elements of $\mathcal{G}$: $\mathcal{D} = \{T_p g_i | i = 1 \dots K, p \in \mathbb{Z}\}$. The function or rather vector set $\mathcal{G}$ is chosen via a sequentially procedure: Every g_i has to satisfy

$$g_i = \underset{\|g\|_2=1}{\operatorname{argmax}} \frac{\sum_{j=1}^{N} \max_{p\in\mathbb{Z}} |\langle y_j \,|\, T_p g \rangle|}{\sum_{l=1}^{i-1} \sum_{p\in\mathbb{Z}} |\langle g_l \,|\, T_p g \rangle|}$$

(for i=1 the denominator is defined to one). This maximization is done by iterating two steps. In the first for every y_j and fixed g the best translation p_j is found, this step is just maximizing the correlation between the translations $T_p g_{i,k}, p \in \mathbb{Z}$ and all y_j. In the second step the translation is fixed and the best g is found. This problem can be translated to a generalized eigenvalue problem.

We compare this algorithms in the introduction of Chapter 4 or rather Section 4.1 and hence obtain a list of desired properties for our learning algorithm, presented in Section 4.1.

2.3 Functional analysis

In this section we want to give a short introduction into the structures and methods we need especially in Chapter 5. We assume the reader is familiar with the basic structures of HILBERT and BANACH spaces, as well as with LEBESGUE measures and integrals. Otherwise we recommend [5] for an introduction.

The section is structured as follows: After an introduction to weak convergence and continuity in Subsection 2.3.1 we give in Subsection 2.3.2 a short composition on results about LEBESGUE and SOBOLEV spaces. All that results can be found in [5], this is also a basis for a more detailed introduction, as well as [26]. Afterwards we collect in Subsection 2.3.3 the ingredients for the direct method of the calculus of variations as given in [7], [171] or [27]. We complete this section in Subsection 2.3.4 with a short overview on the generalized conditional gradient algorithm as presented in [18].

2.3.1 Weak topologies

The well known concept of convergence, in the following denoted as strong convergence, is in many cases to restrictive in the framework of infinite dimensional vector spaces. Especially the famous result of BOLZANO-WEIERSTRASS, every bounded sequence has a strongly converging subsequence, is lost. In order to hold this result also for infinite dimensional spaces, we weaken the concept of convergence.

Below we denote with X a BANACH space and with X' its dual.

Definition 2.11 (Weak convergence)**.** *A sequence $(x_n) \subset X$ with $x \in X$ satisfying*

$$\langle x_n \,|\, x' \rangle \to \langle x \,|\, x' \rangle \quad \text{for all } x' \in X'$$

is called weakly convergent to x or $x_n \rightharpoonup x$. Analogous the sequence $(x'_n) \subset X'$ is called weakly convergent to $x' \in X'$, denoted by $x'_n \overset{*}{\rightharpoonup} x'$, if*

$$\langle x \,|\, x'_n \rangle \to \langle x \,|\, x' \rangle \quad \text{for all } x \in X$$

Weakly convergent sequences have the following important properties:

Theorem 2.12.

1. (a) *For a sequence $x_n \subset X$ converging weakly to $x \in X$, exists a constant $\mathbb{R} \ni K > 0$, such that*

$$\|x_n\| \leq K \qquad \text{and} \qquad \|x\| \leq \liminf_{n\to\infty} \|x_n\|$$

   (b) *For a sequence $x'_n \subset X'$ converging weakly* to $x' \in X'$, exists a constant $K > 0$, such that*

$$\|x'_n\| \leq K \qquad \text{and} \qquad \|x'\| \leq \liminf_{n\to\infty} \|x'_n\|$$

   (c) *Strong convergence* $x_n \to x$ *implies weak convergence* $x_n \rightharpoonup x$

   (d) *Strong convergence* $x'_n \to x'$ *implies weak* convergence* $x'_n \overset{*}{\rightharpoonup} x'$

2. *If* X *is a reflexive* BANACH *space, and there is a* $\mathbb{R} \ni K > 0$*, such that* $\|x_n\| \leq K$*. Then there exists a subsequence* (x_{n_j}) *of* (x_n) *and an* $x \in X$*, such that* $x_{n_j} \rightharpoonup x$*.*

3. *In the case of a separable* BANACH *space* X *let exist an* $0 < K \in \mathbb{R}$ *holding* $\|x'_n\| \leq K$*. Then there are a subsequence* (x'_{n_j}) *of* (x'_n) *and an* $x' \in X'$ *satisfying* $x'_{n_j} \overset{*}{\rightharpoonup} x'$*.*

The last two properties give a weak alternative of the BOLZANO-WEIERSTRASS theorem. This properties are also denoted by weak (sequential) compactness or rather weak* (sequential) compactness.

Analogous to the strong topology we can define a concept of (semi-)continuity using the weak topology.

Definition 2.13 (Weak lower semicontinuity)**.** *Let* f *be a functional* $f : X \mapsto \bar{\mathbb{R}} = \mathbb{R} \cup \{\infty\}$*.* f *is called weakly (sequential) lower semicontinuous in a point* $x \in X$*, if for every sequence* $x_n \rightharpoonup x$*:*

$$\liminf_{x_n \rightharpoonup x} f(x_n) \geq f(x) \ . \tag{2.13}$$

f *is called weak (sequential) lower semicontinuous, if Inequality* (2.13) *is valid in every point* $x \in X$*. Analogous we denote* f *as weak (sequential) continuous, if equality is satisfied in* (2.13) *for all* $x_n \rightharpoonup x$*.*

Analogous definitions are valid for the weak* case.
In the following we omit the term sequential, even as there is a not necessary equivalent setbased definition of all this terms.

Please observe, weak continuity is in contrast to its name a stronger concept than the standard strong continuity and implies the later one. But there is an important set of functionals giving weak lower semicontinuity from the strong one:

Theorem 2.14. *Let* $f : X \mapsto \mathbb{R}$ *be a convex functional. Then weak lower semicontinuity and strong lower semicontinuity are equivalent.*

The last concept we have to introduce is coercivity. There are several definitions of this term in literature, we use the following one:

Definition 2.15 (weak coercivity)**.** *A functional* $f : X \mapsto \bar{\mathbb{R}}$ *is called weak (sequential) coercive, if the sets* $\overline{\{f \leq t\}}$ *are weak (sequential) compact in* X *for every* $t \in \mathbb{R}$*.*

A comparison with the definition of weak compactness shows, that this definition is equivalent on reflexive BANACH spaces with

$$\lim_{\|x\| \to \infty} f(x) = \infty. \tag{2.14}$$

Unfortunately there are two concepts of coercivity in the strong topology, one for both of the stated equivalent weak formulations. If we need one of them in the following chapters, we use the term "sequential" for discrimination:

Definition 2.16 (strong coercivity)**.** *A functional $f : X \mapsto \bar{\mathbb{R}}$ is called strong sequential coercive, if the sets $\overline{\{f \leq t\}}$ are strong sequential compact in X for every $t \in \mathbb{R}$. It is called strong coercive, if*

$$\lim_{\|x\| \to \infty} f(x)/\|x\| = \infty.$$

2.3.2 Lebesgue and Sobolev spaces

The most important Banach spaces we use below are Lebesgue and Sobolev spaces. They are well known, but for reasons of completeness we want to give a short introduction here. Inside this subsection we denote $\Omega \subset \mathbb{R}^d$ as an open, μ-measurable set.

Lebesgue spaces

Definition 2.17 (Lebesgue space)**.** *Let $1 \leq p \leq \infty$. Then we define the space $L_p(\mu, \Omega, \mathbb{K})$ with $\mathbb{K} \in \{\mathbb{R}, \mathbb{C}\}$ as the set of functions*

$$L_p(\mu, \Omega, \mathbb{K}) = \{f : \Omega \mapsto \mathbb{K} \,|\, f \,\mu\text{-measurable}, \|f\|_p < \infty\}.$$

Here $\|\cdot\|_p$ defines via

$$\|f\|_p := \begin{cases} \left(\int\limits_\Omega |f(x)|^p \, \mathrm{d}\mu x \right)^{1/p} & \text{for } 1 \leq p < \infty \\ \operatorname*{esssup}\limits_\Omega |f| := \inf\limits_{\mu(N)=0} \sup\limits_{x \in \Omega \setminus N} |f(x)| & \text{for } p = \infty \end{cases} \tag{2.15}$$

Below we often suppress one ore more arguments of $L_p(\mu, \Omega, \mathbb{K})$ if the suggested argument is clear. The mapping $\|\cdot\|_p : L_p(\Omega) \mapsto \mathbb{R}$ defines a norm on the Banach space $L_p(\Omega)$ that is reflexive if $1 < p < \infty$. The dual space is given by $L_{p'}(\Omega)$, where $\frac{1}{p} + \frac{1}{p'} = 1$, thus the $L_2(\Omega)$ is even a Hilbert space. In the case $p = 1$ the dual is the $L_\infty(\Omega)$, but this spaces are not reflexive.

Additionally the Lebesgue spaces induce the important Hölder inequality for functions $f \in L_p$ and $g \in L_{p'}$, $p, p' \in [1, \infty]$, $\frac{1}{p} + \frac{1}{p'} = 1$:

$$g(f) := \int_\Omega fg \, \mathrm{d}\mu x \leq \|f\|_p \cdot \|g\|_{p'}$$

The definitions of Lebesgue norm and space are also sensible for values $0 < p < 1$ and are used already in Section 2.2 as well as intensely in the following chapters. But in this case we lose almost all interesting properties, beginning with the triangle inequality of the p-"norm".

Sobolev spaces

There is no regularity constraint in the definition of LEBESGUE spaces. In case we need a more smooth function, the natural alternative are so called SOBOLEV spaces. Among the wide field of closely related definition of SOBOLEV-type spaces (cf. [110]) we use the following in the course of this thesis:

Definition 2.18 (SOBOLEV space)**.** *Let $1 \leq p \leq \infty$ and $s \in \mathbb{N}$. Then the* SOBOLEV *space $W_{s,p}(\mu, \Omega, \mathbb{K})$ is defined as*

$$W_{s,p}(\Omega) = \{f \in L_p(\Omega) \,|\, \nabla^\alpha f \in L_p \,\forall\, |\alpha| \leq s\}.$$

Here α denotes a multi-index (cf. [5, Section 1.5]) and $\nabla^\alpha f$ denotes the weak derivative (also know as distributional derivative) of f defined as

$$\int_\Omega f(x)(\nabla^\alpha \zeta)(x)\,\mathrm{d}\mu x \;= (-1)^{|\alpha|} \int_\Omega (\nabla^\alpha f)(x)\zeta(x)\,\mathrm{d}\mu x \;\; \forall \zeta \in C_0^\infty(\Omega).$$

Furthermore we define a norm on this space by

$$\|f\|_{s,p} = \begin{cases} \left(\sum\limits_{|\alpha|=0}^{s} \|\nabla^\alpha u\|_p^p \right)^{1/p} & \text{for } 1 \leq p < \infty \\ \max\limits_{0 \leq |\alpha| \leq s} \{\|\nabla^\alpha f\|_\infty\} & \text{for } p = \infty \end{cases} \tag{2.16}$$

Similar to the case of LEBESGUE spaces, all SOBOLEV spaces are BANACH spaces and for $1 < p < \infty$ reflexive. In this case the dual is given by $W_{-s,p'}(\Omega)$ with $1/p + 1/p' = 1$. Analogous to the LEBESGUE case, spaces of the form $W_{s,2}(\Omega)$ are HILBERT spaces.

further properties

If we want to investigate the connections between different SOBOLEV and LEBESGUE spaces, It is clear from the definitions of the norms (Equations (2.15) and (2.16)) that $W_{s,p}(\Omega) \subset L_p(\Omega)$. For more advanced results we quote here the SOBOLEV embedding theorem. But before we need a short definition of compact operators. Below we denote for sake of simplicity $L_p = W_{0,p}$.

Definition 2.19. *A bounded linear mapping $A : X \to Y$ between two* BANACH *spaces X and Y is called compact if for every bounded sequence $(x_n) \subset X$ there is a (strongly) convergent subsequence of $(Ax_n) \subset Y$.*

Theorem 2.20 (SOBOLEV embedding theorem)**.** *Let $\Omega \in \mathbb{R}^n$ be bounded with* LIPSCHITZ *boundary and $1 \leq p_1, p_2 < \infty$, furthermore $m_1, m_2 \in \mathbb{N} \cup \{0\}$.*

1. *Let*
$$m_1 - \frac{n}{p_1} \geq m_2 - \frac{n}{p_2} \quad \text{and} \quad m_1 \geq m_2 \;, \tag{2.17}$$
then the embedding $Id : W_{m_1,p_1}(\Omega) \mapsto W_{m_2,p_2}(\Omega)$ is continuous, i.e. there is a constant $C = C(n, \Omega, m_1, p_1, m_2, p_2)$ with $\|f\|_{m_2,p_2} \leq C\|f\|_{m_1,p_1}$ for any $f \in W_{m_1,p_1}(\Omega)$.

2. *In case the inequalities in* (2.17) *are strict, the embedding Id is compact.*

We want to mention some special cases of the compact embeddings:

Lemma 2.21. *Let Ω be bounded with* Lipschitz *boundary and $f_n \rightharpoonup f$ in $W_{1,p}(\Omega)$, then the following convergences are valid:*

1. *If $1 \leq p < n$, then $f_n \to f$ in $L_q(\Omega)$, $1 \leq q < \frac{np}{n-p}$.*
2. *If $p = n$, then $f_n \to f$ in $L_q(\Omega)$, $1 \leq q < \infty$.*
3. *If $p > n$, then $f_n \to f$ in $L_\infty(\Omega)$.*

In the following we want to give results about weak lower semicontinuity of functions in Sobolev spaces. Here the measures μ and μ' denote measures, where the main theorems for the Lebesgue measure are valid, as, e.g., satified by a weighted version of this or a discrete measure. This result is adopted from [26, Chapter 3, Theorem 3.4], but as we use slightly different spaces we give also a modification of the proof simplified to our necessities. Furthermore we need an introducing definition, also taken from [26].

Definition 2.22. *For an open set $\Omega \subset \mathbb{R}^n$ a function $f : \Omega \times \mathbb{R}^m \times \mathbb{R}^N \mapsto \bar{\mathbb{R}} = \mathbb{R} \cup \{\infty\}$ is called a* Carathéodory *function if*

- *$f(x, \cdot, \cdot)$ is continuous for μ-almost every $x \in \Omega$,*
- *$f(\cdot, u, \xi)$ is measurable in x for every $(u, \xi) \in \mathbb{R}^m \times \mathbb{R}^N$.*

Theorem 2.23. *Let Ω and Ω' be two bounded, open subsets of $\mathbb{R}^l$ or rather $\mathbb{R}^{n-l}$. Furthermore, let $f\colon (\Omega \times \Omega') \times \mathbb{R}^m \times \mathbb{R}^N \mapsto \bar{\mathbb{R}}$ be a* Carathéodory *function, satisfying*

$$f((x, x'), u, \nabla u) \geq \langle a(x) \,|\, \nabla u \rangle + b(x)$$

for μ, μ'-almost every $(x, x') \in \Omega \times \Omega'$, every $(u, \nabla u) \in \mathbb{R}^m \times \mathbb{R}^N$ and for some $b \in L_1(\Omega \times \Omega')$ and $a \in L^q(L^{q'}(\Omega'), \Omega)^N$, where $1/q + 1/p = 1$, $1/q' + 1/p' = 1$. Define additionally a sequence

$$u_\nu \rightharpoonup \bar{u} \quad in \quad W_{1,p}(W_{1,p'}(\Omega'), \Omega)$$

with $p, p' \geq 1$. If additionally $f((x, x'), u, \cdot)$ is convex, then

$$J(u, \nabla u) = \int_\Omega \int_{\Omega'} f((x, x'), u(x, x'), \nabla u(x, x'))\, \mathrm{d}\mu' x'\, \mathrm{d}\mu x$$

is weak lower semicontinuous in $W_{1,p}(W_{1,p'}(\Omega'), \Omega)$.

Proof. The proof is decomposed in four steps:

1. If necessary we replace f by

$$\tilde{f}((x,x'),u,\nabla u) = f((x,x'),u,\nabla u) - \left\langle a(x,x') \,\middle|\, \nabla u \right\rangle + b(x)$$

so we can assume $f \geq 0$.

2. Due to the preceding step, we know for a sequence $u_\nu \rightharpoonup \bar{u} \in W_{1,p}(W_{1,p'}(\Omega'),\Omega)$ that

$$L = \liminf_{\nu\to\infty} J(u_\nu, \nabla u_\nu) > -\infty \ .$$

Additionally we assume $L < \infty$, else the result would be trivial. After choosing a subsequence of u_ν, we define $L = \lim_{\nu\to\infty} J(u_\nu, \nabla u_\nu)$.

3. For a fixed $\epsilon > 0$ we assume the existence of a measurable set $\Omega_\epsilon \subset \Omega \times \Omega'$, such that for a subsequence ν_j there exists an ν_ϵ satisfying for every $\nu_j \geq \nu_\epsilon$:

$$\begin{aligned} &\left|(\Omega\times\Omega')\backslash\Omega_\epsilon\right| < \epsilon \\ &\text{and} \\ &\iint_{\Omega_\epsilon} \left| f((x,x'),u_{\nu_j}(x,x'),\nabla u_{\nu_j}(x,x')) \right. \\ &\qquad \left. -f((x,x'),\bar{u}(x,x'),\nabla u_{\nu_j}(x,x'))\right| \,\mathrm{d}\mu x \,\mathrm{d}\mu' x' \ < \epsilon\,|\Omega| \end{aligned} \tag{2.18}$$

4. Let us define

$$\chi_\epsilon(x,x') = \begin{cases} 1 & \text{if } (x,x') \in \Omega_\epsilon \\ 0 & \text{else} \end{cases}$$

and

$$g((x,x'),\xi) = \chi_\epsilon(x,x') f((x,x'),\bar{u}(x,x'),\xi)$$

So $g : (\Omega\times\Omega')\times\mathbb{R}^N \to \bar{\mathbb{R}}$ is a Carathéodory function and the integral

$$G(\xi) \equiv \int_\Omega \int_{\Omega'} g((x,x'),\xi(x,x'))\,\mathrm{d}\mu' x' \,\mathrm{d}\mu x$$

is a convex function, as f is convex for μ,μ'-almost every (x,x'). Additionally G is lower semicontinuous over $L_p(L_{p'}(\Omega'),\Omega)^N$, according to the lemma of Fatou (cf. [84, 5.11]), as $g > 0$ μ,μ'-almost everywhere, together with Theorem 2.14.

The results of Steps 1 and 3 give for large enough ν_j:

$$\begin{aligned} &\int_\Omega\int_{\Omega'} f((x,x'),u_{\nu_j}(x,x'),\nabla u_{\nu_j}(x,x'))\,\mathrm{d}\mu' x' \,\mathrm{d}\mu x \\ \geq\ & \int_\Omega\int_{\Omega'} \chi_\epsilon(x,x') f((x,x'),u_{\nu_j}(x,x'),\nabla u_{\nu_j}(x,x'))\,\mathrm{d}\mu' x' \,\mathrm{d}\mu x \\ \geq\ & \int_\Omega\int_{\Omega'} \chi_\epsilon f((x,x'),\bar{u}(x,x'),\nabla u_{\nu_j}(x,x'))\,\mathrm{d}\mu' x' \,\mathrm{d}\mu x \\ & -\int_\Omega\int_{\Omega'} \chi_\epsilon(x,x') \left| f((x,x'),u_{\nu_j}(x,x'),\nabla u_{\nu_j}(x,x')) \right. \\ & \qquad \left. -f((x,x'),\bar{u}(x,x'),\nabla u_{\nu_j}(x,x'))\right| \,\mathrm{d}\mu' x' \,\mathrm{d}\mu x \\ \geq\ & \int_\Omega\int_{\Omega'} \chi_\epsilon(x,x') f((x,x'),\bar{u}(x,x'),\nabla u_{\nu_j}(x,x'))\,\mathrm{d}\mu' x' \,\mathrm{d}\mu x \ - \epsilon\left|\Omega\times\Omega'\right| \end{aligned}$$

According to the weak lower semicontinuity of G we get

$$\int_\Omega \int_{\Omega'} f((x,x'),u_{\nu_j}(x,x'),\nabla u_{\nu_j}(x,x'))\,\mathrm{d}\mu' x'\ \mathrm{d}\mu x$$
$$\geq \int_\Omega \int_{\Omega'} \chi_\epsilon(x,x') f((x,x'),\bar{u}(x,x'),\nabla\bar{u}(x,x')\,\mathrm{d}\mu' x'\ \mathrm{d}\mu x - \epsilon\left|\Omega\times\Omega'\right| .$$

So we get the result with $\epsilon\to 0$ via LEBESGUE's monotone convergence theorem (cf. [84]).

To prove the assumption of Number 3, we remember first, that $u_\nu \to \bar{u}$ and $\nabla u_\nu \rightharpoonup \nabla\bar{u}$ in $L_p(L_{p'}(\Omega'),\Omega)$. So there exists for every $\epsilon>0$ an $M_\epsilon>0$ independent of ν satifying for every ν:

$$\begin{array}{lll} K^1_{\epsilon,\nu}=\{(x,x')\in(\Omega\times\Omega')|\ |\bar{u}(x,x')|,|u_\nu(x,x')|\geq M_\epsilon\} & \Rightarrow & \left|K^1_{\epsilon,\nu}\right|<\epsilon/6 \\ K^2_{\epsilon,\nu}=\{(x,x')\in(\Omega\times\Omega')|\ |\nabla u_\nu(x,x')|\geq M_\epsilon\} & \Rightarrow & \left|K^2_{\epsilon,\nu}\right|<\epsilon/6 \end{array}$$

Therefore, by defining

$$\Omega^1_{\epsilon,\nu}=\Omega\backslash(K^1_{\epsilon,\nu}\cup K^2_{\epsilon,\nu})$$

we get

$$\left|\Omega\backslash\Omega^1_{\epsilon,\nu}\right|<\epsilon/3 .$$

According to the SCORZA-DRAGONI theorem (cf. [26, p. 74] or [20, p. 284]) there exists for the CARATHÉODORY function f a compact set $\Omega^2_{\epsilon,\nu}\subset\Omega^1_{\epsilon,\nu}$ such that for $(x,x')\in\Omega^2_{\epsilon,\nu}$: $f((x,x'),u,\nabla u)$ is continuous and $\left|\Omega^1_{\epsilon,\nu}\backslash\Omega^2_{\epsilon,\nu}\right|<\epsilon/3$. So there exists $\delta(\epsilon)>0$ giving for every $(x,x')\in\Omega^2_{\epsilon,\nu}$:

$$\left|u(x,x')-v(x,x')\right|<\delta(\epsilon)$$
$$\Rightarrow$$
$$\left|f((x,x'),u(x,x'),\nabla u(x,x'))-f((x,x'),v(x,x'),\nabla u(x,x'))\right|<\epsilon$$

Fixing $\delta(\epsilon)$ we can find by $u_\nu\to\bar{u}$ in $L_p(L_{p'}(\Omega'),\Omega)$ a $\nu_\epsilon=\nu_{\epsilon,\delta(\epsilon)}$ with

$$\Omega^3_{\epsilon,\nu}=\{(x,x')\in(\Omega\times\Omega')|\left|u_\nu(x,x')-\bar{u}(x,x')\right|<\delta(\epsilon)\}$$

and $\left|\Omega\backslash\Omega^3_{\epsilon,\nu}\right|<\epsilon/3$ for every $\nu\geq\nu_\epsilon$. So if we define $\Omega_{\epsilon,\nu}=\Omega^2_{\epsilon,\nu}\cap\Omega^3_{\epsilon,\nu}$, we get

$$\left|(\Omega\times\Omega')\backslash\Omega_{\epsilon,\nu}\right|<\epsilon,$$
and
$$\iint_{\Omega_{\epsilon,\nu}} |f((x,x'),u_\nu(x,x'),\nabla u_\nu(x,x'))$$
$$-f((x,x'),\bar{u}(x,x'),\nabla u_\nu(x,x'))|\,\mathrm{d}\mu' x'\ \mathrm{d}\mu x<\epsilon\,|\Omega|$$

for every $\nu\geq\nu_\epsilon$. In order to construct the subsequence mentioned in Step 3 we choose now $\epsilon_j=\epsilon/2^j$, $j=1,2,\ldots$. So we can define the ν_j by $\nu_j\geq\nu_{\epsilon_j}$ with $\lim\nu_j=\infty$ and furthermore

$$\Omega_\epsilon=\bigcap_{j=1}^{\infty}\Omega_{\epsilon_j,\nu_j} .$$

□

2.3.3 The direct method

Most of the 5th chapter concern with the problem of finding a minimum of a non-linear functional

$$\min_{x \in X} F(x) = \alpha.$$

For proving existence of a solution for such a problem usually a three step approach is used, the so called "direct method of the calculus of variations". This was in the given version originally proposed by TONELLI in [161] (for actual sources see, e.g., [7, 27]) and works as follows (below we denote by τ the strong, weak or weak* topology):

1. Constructing a minimizing sequence, i.e. a sequence (x_n) achieving $\lim_{n\to\infty} F(x_n) = \inf_{x\in X} F(x)$.
2. Finding a subsequence (x_{n_k}), τ-converging to $x \in X$
3. Proving that $F(x) \leq \tau\text{-}\liminf_{k\to\infty} F(x_{n_k})$

Generally, for the first step, we can just assume the existence of the sequence and have to solve the further steps for arbitrary sequences. We have just to observe that more complicate functionals F are vulnerable to producing several just local minima. Furthermore there has to be a minimum at all, so the functional has to be proper, i.e. $F \not\equiv \infty$. For the second step one has prove the existence of a τ-converging subsequence for every sequence. This means sequential τ-coercivity in the sense of Definition 2.15 or 2.16. In the standard case F operating on LEBESGUE or SOBOLEV spaces, this is according to Theorem 2.14:3-4 and Section 2.3.2 in general just possible in a weak or weak* topology. Finally sufficient for the third step is τ-lower semicontinuity as defined in Definition 2.13.

There are several reformulations and special cases of the direct method (cf. [171, Theorem 38.B, Proposition 38.12]), but all have this structure. So to prove existence of a minimum we have to prove in general:

1. Properness, i.e. $F \neq \infty$,
2. Weak coercivity on a reflexive BANACH space or rather weak* coercivity on a separable one,
3. Weak or rather weak* lower semicontinuity.

Here the last point is in general the most demanding one.

2.3.4 Generalized conditional gradient algorithm

For the constructive part of Chapter 5 we want to apply the following algorithm, realizing the generalized conditional gradient method as presented in [18] (for a introduction to FRÉCHET and GÂTEAUX derivatives see, e.g., [172]):

Algorithm 2.24 (Generalized conditional gradient algorithm)**.** *Let $E = F + \Phi : H \to]-\infty, \infty]$ for a* Hilbert *space H and let the* Gâteaux *derivative of F exist. Then the generalized conditional gradient algorithm is defined via:*

1. *Set $n = 0$ and choose $u_0 \in H$ with $\Phi(u_0) < \infty$, furthermore define a stopping criterium S.*
2. *Find a solution v_n of*
$$\operatorname*{argmin}_{v \in H} \left\langle F'(u_n) \,\middle|\, v \right\rangle + \Phi(v) \ .$$
3. *Determine s_n as*
$$s_n = \operatorname*{argmin}_{s \in [0,1]} F(u_n + s(v_n - u_n)) + \Phi(u_n + s(v_n - u_n)) \ .$$
4. *Set $u_{n+1} = u_n + s_n(v_n - u_n)$, $n = n + 1$ and return to Step 2, if S is not met.*

A basic property of the Φ in Algorithm 2.24 is

Condition 2.25. *Let H be a* Hilbert *space and $\Phi : H \to]-\infty, \infty]$ be a functional satisfying*

1. *Φ is proper, i.e. there exists a $u \in H$ with $\Phi(u) < \infty$*
2. *Φ is convex, so for all $u, v \in H$ and $s \in [0, 1]$ it satisfies*
$$\Phi(su + (1 - s)v) \leq s\Phi(u) + (1 - s)\Phi(v)$$
3. *Φ is lower semicontinuous, i.e. every $\lim_{n\to\infty} u_n = u$ in H satisfies $\Phi(u) \leq \liminf_{n\to\infty} \Phi(u_n)$*
4. *Φ is (strong) coercive, so $\Phi(u)/\|u\| \to \infty$ whenever $\|u\| \to \infty$.*

The authors of [18] proved the following two results:

Lemma 2.26 (Descent of the generalized conditional gradient algorithm)**.**
Let $F : H \to \mathbb{R}$ be a Gâteaux *differentiable functional and Φ satisfy Condition 2.25. Then a necessary condition of*
$$u = \operatorname*{argmin}_{v \in H} F(v) + \Phi(v)$$
is given by
$$\left\langle F'(u) \,\middle|\, u \right\rangle + \Phi(u) = \min_{v \in H} \left\langle F'(u) \,\middle|\, v \right\rangle + \Phi(v). \tag{2.19}$$
If this condition is not satisfied for some $u_n \in H$, the generalized Conditional Gradient Method as stated in Algorithm 2.24 gives a u_{n+1} with
$$F(u_{n+1}) + \Phi(u_{n+1}) < F(u_n) + \Phi(u_n) \ .$$

For the following theorem there are two versions, one concerning strong convergence, one weak, we state the later one in brackets.

Theorem 2.27 (Convergence of the generalized conditional gradient algorithm)**.** *Let Φ satisfy Condition 2.25 and F be (completely) continuously* Fréchet *differentiable, then*

$$\Psi(u) = \left\langle F'(u) \,\middle|\, u \right\rangle + \Phi(u) - \min_{v \in H}(\left\langle F'(u) \,\middle|\, v \right\rangle + \Phi(v) \tag{2.20}$$

is (weak) lower semicontinuous. If furthermore $F + \Phi$ is (weak) coercive and F' uniformly continuous on bounded sets, then a sequence $\{u_n\}$ generated by Algorithm 2.24 satisfies $\lim_{n\to\infty} \Psi(u_n) = 0$. In case $E_t = \{u \in H | \Phi(u) \leq t\}$ is compact for all $t \in \mathbb{R}$, (or rather F is weakly lower semicontinuous) there exists a convergent (or rather weakly convergent) subsequence of $\{u_n\}$ and every convergent subsequence of $\{u_n\}$ converges to a stationary point of Algorithm 2.24 according to $F + \Phi$.

Complete continuous functionals map weak convergent sequences to norm convergent ones, this property is in general extremely strong.

After collecting this well known results on wavelets, dictionaries and functional analysis, we turn in the following chapters to the real substance of this thesis.

Chapter 3

Matching Wavelets

As already mentioned in Section 1.4, a minor aspect of this thesis is the matching or approximation of wavelets to a given signal P. I.e. finding a wavelet ψ, satisfying $\psi = \operatorname{argmin} \|\psi - P\|$. During the past twelve years several ideas were published for solving this task, e.g., in [17, 57, 21, 134, 173]. Sweldens remarked 1996 in [153] the possibility of giving a desired shape to a wavelet via lifting (cf. Section 2.1.2). Later on Thielemann elaborated this in [159, 160] for the case of approximation to a given signal P according to $\|\cdot\|_2$, independently there where similar results in [79, 117], using different minimizing norms. This approach offers several advantages: all discrete wavelets can be constructed via lifting schemes (see Lemmata 2.5 and 2.6); additional desirable properties like vanishing moments and smoothness can be incorporated directly (see Section 3.2); computations for constructing matched wavelets via the lifting scheme are efficient ([153, Section 6]). Based on Thielemann's elaborations we gave in [133] in a joint work an advancement, offering approximative shift-invariance of the wavelet, that is the basis for this chapter. Furthermore we gave there some applications, as we demonstrate in Subsection 6.2.1.

Section 3.1 gives a short overview, why it is sensible at all, to use wavelets matched to a given signal. Afterwards we give in Section 3.2 our adaption of the wavelet matching method from [159, 160]. We finish this chapter in Section 3.3 with our own results concerning the shift-invariance properties of matched wavelets.

3.1 Why and how to use matched wavelets

The first question in every application of wavelets (cf. Section 2.1.1) is to choose one wavelet out of the big family of existing ones. The criteria are typically the smoothness of the wavelet and of its dual one, their localization in space (or time) and frequency. Other influencing properties are the symmetry of ψ and $\tilde{\psi}$, their orthogonality as well as the length of their filters. While the last properties influence directly the computation time of the wavelet transform, the preceding ones are chosen in order to optimize the transform to the given data and application.

In case we are searching for a special pattern P, $\|P\|_2 = 1$ in a given signal f or we want to adapt a shrinkage to preserve a predefined P, it is sensible to give the wavelet a special shape, optimized to this P. To illustrate this we assume $f \in L_2(\mathbb{R})$ to be of the form $f(t) = s(t) + P(t)$, where s is a noise type signal and $\operatorname{supp} P = [0, T]$. Then the wavelet coefficients $f_{j,k}$ (cf. Equation (2.4)) according to ψ are given by

$$f_{j,k} = \langle f \,|\, \psi_{j,k} \rangle = \langle s \,|\, \psi_{j,k} \rangle + \langle P \,|\, \psi_{j,k} \rangle .$$

The task is to distinguish the coefficients induced by s from that induced by P. Since s is a noise type signal, its coefficients spread over all j and k, even if their absolute value decrease with rising j. So the best distinction would be possible, if $\langle P \,|\, \psi_{j,k} \rangle$ would be sparse in $(j,k) = \mathbb{Z}^2$ with absolute values as big as possible. Since $\|P\|_2 = \|\psi_{j,k}\|_2 = 1$ the maximum of $|\langle P \,|\, \psi_{j,k} \rangle|$ equals 1. In case ψ and $\tilde{\psi}$ form a biorthogonal wavelet base, the best result can be achieved by a wavelet Ψ, whose dual $\tilde{\Psi}$ is of the form $\tilde{\Psi}_{j^*,k^*} = P$. In that case we have

$$\langle P \,|\, \Psi_{j,k} \rangle = \left\langle \tilde{\Psi}_{j^*,k^*} \,\middle|\, \Psi_{j,k} \right\rangle = \delta_{j^*,j}\delta_{k^*,k}. \tag{3.1}$$

Having a closer look on this equation we notice two problems:

- Unfortunately most P are not satisfying the conditions stated in Equation (2.5), necessary for forming a biorthogonal wavelet basis $\{\tilde{\psi}_{j,k} | (j,k) \in \mathbb{Z}^2, j \leq j'\}$.
- Furthermore even if the first point was satisfied, it would not be clear, whether the corresponding dual wavelet basis $\{\psi_{j,k} | (j,k) \in \mathbb{Z}^2, j \leq j'\}$ exists in $L_2(\mathbb{R})$. Otherwise ψ would have a fractal shape and every discretization would be numerical instable.

For solving the first problem we observe, an approximation $\tilde{\Psi}_{j^*,k^*} \approx P$ serves almost as good as the original P, due to the linearity of the transform. Furthermore P is in applicational framework just known from measurement or model. So it is even more sensible to use an approximation, since P is not an exact source.

One solution of the second problem would be to limit the approximation of P to orthogonal wavelets. Nevertheless this would increase the distance $\|P - \tilde{\Psi}_{j^*,k^*}\|_2$, since orthogonal wavelets are a subset of biorthogonal ones (cf. Page 11). Furthermore the construction of orthogonal wavelets would be a non-trivial and non-linear task, as the negated existence of symmetric orthogonal wavelets (beside the Haar wavelet) shows (see [28, Chapter 8]).

In the following sections we consider two other possible solutions: We offer briefly an approach to construct $\tilde{\Psi}$ connected to a smoother Ψ (cf. Page 35 et sq.). But more important is the alternative to approximate $\Psi_{j^*,k^*} \approx P$ instead of $\tilde{\Psi}_{j^*,k^*} \approx P$. In this case we do not get a sparsity as perfect as in (3.1), but we obtain in comparison to an standard wavelet ψ:

$$|\langle \psi_{j^*,k^*} \,|\, P \rangle| \leq \|\psi\|\|P\| = 1 = |\langle P \,|\, P \rangle| \approx |\langle \Psi_{j^*,k^*} \,|\, P \rangle| . \tag{3.2}$$

Here we assumed as above $\|P\|_2 = \|\Psi_{j,k}\|_2 = 1$. So the coefficient of the transform is for a matched wavelet Ψ in absolute values greater equal that one for an arbitrary ψ.

Concerning the tenor of this section it seems to be sensible to use matched wavelets, so we turn to their construction in the following section.

3.2 Matching wavelets via lifting

As mentioned in Section 3.1, our aim in this section is to find for a given pattern $P \in L_2(\mathbb{R})$ a wavelet ψ^*, such that for fixed $j^*, k^* \in \mathbb{Z}$ the condition

$$\psi^* = \operatorname{argmin} \|P - \psi_{j^*,k^*}\|_2 \tag{3.3}$$

is satisfied. Without loss of generality we can assume P to be rescaled and shifted that way to set $j^* = k^* = 0$, so ψ_{j^*,k^*} equals the mother wavelet ψ.

According to the properties that a wavelet, or more exact its filter, has to satisfy (see, e.g., Equation (2.5) for the biorthogonal case), this problem is highly non-linear. For obtaining an approximative solution we restrict ourselves to a linear subproblem: Based on Lemma 2.6 we know about linear combinations of wavelets that are wavelets again. So we can formulate the following subproblem of Equation (3.3):

$$\{c^*, s^*\} = \operatorname{argmin}\Big\{ \|c\psi^0 + \varphi \star s - P\| \;\Big|\; c \in \mathbb{R}\backslash\{0\}, s \in \mathbb{R}^{\mathbb{Z}} \Big\} , \tag{3.4}$$

$$\psi^*_\sigma(x) = c^*\psi^0(x) - \sum_k s^*_k \varphi(x-k) , \tag{3.5}$$

where σ denotes a special choice of starting wavelets, $\sigma = \{\psi^0, \tilde{\psi}^0, \varphi, \tilde{\varphi}^0\}$. Here the coefficient c is necessary to hold the norm $\|\psi^*_\sigma\|_2$ in the size of $\|P\|_2$ and obtain a good approximation. We refuse to use several lifting steps as metioned in Remark 2.9, since in the case of dual lifting the connection between the matched wavelet ψ and the Lifting filter s is highly non-linear.

So we separated the original problem given in Equation (3.3) into two steps:

- Fixing an appropriate set σ of starting wavelet functions and n, m denoting the position and length of $s = \{s_m, \dots s_n\}$.
- Optimizing the lifting coefficients c^* and s^*.

We see later, the later one is a simple linear problem.

The choice of σ is not based on computation but on the following considerations:

- There should be an a priori estimation of the matching error $E = \|P - \psi^*_2\|$.
- The wavelet should be as smooth as the pattern.

Thus the so called CDF wavelets (cf. [28, Chapter 8]) are a good choice, resulting from fixing $L \equiv 1$ in Lemma 2.8. In that case the φ and ψ as well as ψ^* are splines of order p that can be chosen according to the patterns smoothness. Here the wavelet matching reduces to a spline approximation with well known error estimates as stated, e.g., in [32, pp. 223–229]. Notice here the spline has equidistant knots, beside an additional one introduced by ψ^0 (cf. Figure 2.1). Below we restrict ourselves to the CDF wavelets and we give also some evidence about the choice of n, m and p.

Let us now turn to the optimization of s and c in an applicational framework. There the pattern P is in general not available as an L_2-function, but P is given at a finite set of points. For simplicity we assume equidistributed data, i.e.:

Definition 3.1. *Let $S = \{x_1, \ldots, x_l\}$, satisfying $\forall r \in \{0, \ldots, l-1\} : x_{r+1} - x_r = \tau, \tau \in \mathbb{R}^+$. Then a pattern P is defined as the finite sequence*

$$P = P_S = (P_{x_r})_{x_r \in S} \in \ell_2(S) \ .$$

Let us assume that the values of φ and ψ^0 (at every $x_r \in S$) are given. For example, these values can be computed from the related filters h and $\tilde{h}$ by the cascade algorithm (cf. [28, Chapter 6.5]). Then we get the optimal s^* and c^* by

Proposition 3.2. *Let a set of CDF scaling functions and wavelets $\sigma = \{\psi^0, \tilde{\psi}^0, \varphi, \tilde{\varphi}^0\}$ and a pattern $P = (P_{x_r})_{x_r \in S} \in \ell_2(S)$ be given. For a pair of fixed indices (n, m) we define the optimally matched wavelet – according to Equation* (3.4) *– via a lifting filter $s(\omega) = \sum_{j=m}^{n} s_j e^{-ij\omega}$ by*

$$\underset{c, s_m, \ldots, s_n}{\operatorname{argmin}} \left\| \left(\begin{array}{ccc|c} \varphi(x_1 - m) & \ldots & \varphi(x_1 - n) & \psi^0(x_1) \\ \vdots & \ddots & \vdots & \vdots \\ \varphi(x_l - m) & \ldots & \varphi(x_l - n) & \psi^0(x_l) \end{array} \right) \cdot \left(\begin{array}{c} s_m \\ \vdots \\ s_n \\ \hline c \end{array} \right) - \left(\begin{array}{c} P_{x_1} \\ \vdots \\ P_{x_l} \end{array} \right) \right\|_2 . \tag{3.6}$$

For the sake of simplicity we define the following abbreviations:

Definition 3.3. *Let $m, n \in \mathbb{Z}$ be given, S as defined in Definition 3.1 and $\sigma = \{\psi^0, \tilde{\psi}^0, \varphi, \tilde{\varphi}^0\}$ be a set of biorthogonal wavelets and scaling functions. Then we define*

$$\tilde{x} = \begin{pmatrix} x_1 \\ \vdots \\ x_l \end{pmatrix}, \qquad \tilde{s}_{m,n} = \begin{pmatrix} s_m \\ \vdots \\ s_n \\ \hline c \end{pmatrix}$$

and

$$\Phi_{m,n,r,\sigma}(\tilde{x}) = \left(\begin{array}{ccc|c} \varphi(x_1 - m) & \ldots & \varphi(x_1 - n) & \psi^0(x_1 - r) \\ \vdots & \ddots & \vdots & \vdots \\ \varphi(x_l - m) & \ldots & \varphi(x_l - n) & \psi^0(x_l - r) \end{array} \right) \tag{3.7}$$

Using this, Equation (3.4) can be written as

$$\tilde{s}^*_{m,n} = \underset{\tilde{s}_{m,n}}{\operatorname{argmin}} \left\| \Phi_{m,n,0,\sigma}(\tilde{x}) \cdot \tilde{s}_{m,n} - P_S \right\| .$$

The minimization is achievable using the MOORE-PENROSE inverse of $\Phi_{m,n,0,\sigma}(\tilde{x})$, see [150, Theorem 4.8.5.4] for the properties of this inverse, especially in connection with minimum norm solutions. So we obtain directly:

Corollary 3.4. *Under the assumptions of Proposition 3.2 the solution of Equation* (3.6) *is given by*

$$\tilde{s}^*_{m,n} = \Phi_{m,n,0,\sigma}(\tilde{x})^+ P_S$$

and we gain the optimal wavelet according to Equation (3.4) *via*

$$\psi^*_\sigma = (\tilde{s}^*_{m,n})_{n-m+2}\psi^0 - \sum_{k=m}^{n} (\tilde{s}^*_{m,n})_{k-m+1}\varphi(\cdot - k) \ .$$

We already mentioned in Section 3.1, that the duals of the fitted wavelets are in general of fractal nature. Unfortunately this is also true in the case of CDF wavelets, except for the case of HAAR wavelet, see, e.g., Figure 3.1. Furthermore there is no difference in most applications, since also for σ with smoother dual wavelets the resulting $\tilde{\psi}^*_\sigma$ can also be non-integrable, as shown in Figure 3.2.

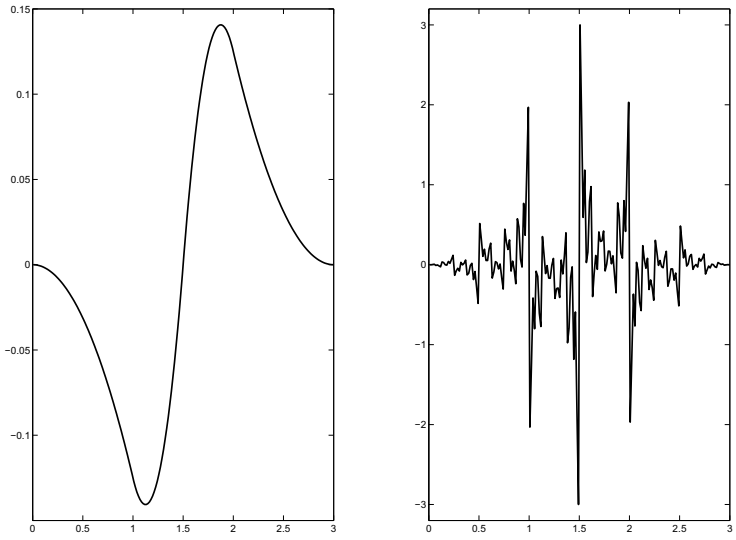


Figure 3.1: CDF wavelet (left) and dual wavelet(right) for parameters $p = 3$ and $\tilde{p} = 1$

We have now a look on two practical ways to go around this problem: First to use Equation (3.2) for application instead of Equation (3.1), not using the dual wavelets. We use this in the application in Section 6.2. The second possibility is to use the

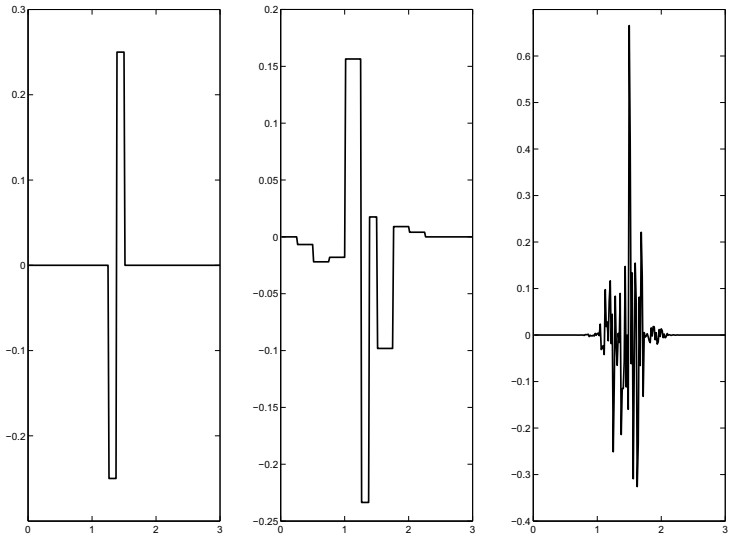
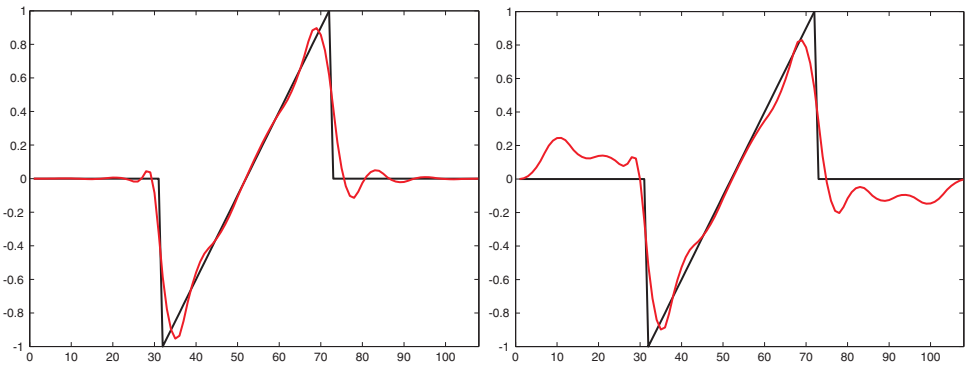


Figure 3.2: Original primal and dual wavelet (left), lifted primal wavelet (middle) and lifted dual wavelet(right). A fractal dual wavelet can also result, if the original one is in L_2.

result of Lemma 2.7 and minimize Equation (3.6) with a constraint forcing $\tilde{h}$ to have several vanishing moments. We do not account this method for application, since Figure 3.3 shows that the performance of matching decreases to fast, if the difference between the numbers of vanishing moments in P and ψ_σ^* is to high. For a deeper consideration of influencing the smoothness of $\tilde{\psi}_\sigma^*$ including some estimates of Lipschitz and Sobolev smoothness see [160].



(a) 0 vanishing moments (b) 3 vanishing moments

Figure 3.3: Matching (red) of the same pattern (black, 0 Moments) with different number of vanishing moments

We want to finish this section by mentioning some practical considerations:

In most applications it is sensible not to approximate just the pattern P_S, but an extended version $\bar{P}_T$ to lessen the matching error. Here T satisfies $T = \{x_{\tilde{l}}, \dots x_{\hat{l}}\}$,

$\forall r \in \{x_{\tilde{l}}, \ldots, x_{\hat{l}}\} : \; x_{r+1} - x_r = \tau$ with the same τ as in Definition 3.1 and $S \subset T$. Furthermore $P_{x_r} = \bar{P}_{x_r}$ for all $x_r \in S$. In the case of acoustic data, that are mostly near zero at endpoints, we used in Subsection 6.2.1 an extension by zeros. For simplicity we denote below the extended pattern $\bar{P}$ by P.

A further option for improving the approximation properties of the matched wavelet is to vary the parameters m and n. Changing their distance also influences the length of g^*, i.e. the length of the filter connected to ϕ_σ^*, and thus the computation time. Furthermore also the accuracy of the approximation depends on the length of s as can be seen from Equation (3.6). But an integer shift $(m, n) \to (m + r, n + r)$ is an additional free parameter. This can easily be incorporated by a variation of Φ:

Remark 3.5. *Let P_S and σ be defined as in Proposition 3.2. Furthermore a fixed number $n \in \mathbb{N}$ is given. Then the wavelet ψ_σ^* matching best P_S with a lifting filter s of length n is given via*

$$\begin{aligned}
\tilde{s}^*_{1+r^*,n+r^*} &= \operatorname*{argmin}_{\tilde{s}_{1,n}} \min_{r\in\mathbb{Z}} \|\Phi_{1,n,r,\sigma}(\tilde{x}) \cdot \tilde{s}_{1,n} - P_S\| \\
&= \operatorname*{argmin}_{\tilde{s}_{1+r,n+r}} \min_{r\in\mathbb{Z}} \left\| \Phi_{1-r,n-r,0,\sigma}(\tilde{x}) \cdot \tilde{s}_{1-r,n-r} - \begin{pmatrix} P_{x_1+r} \\ \vdots \\ P_{x_l+r} \end{pmatrix} \right\| \\
&= \left(\Phi_{1-r^*,n-r^*,0,\sigma}(\tilde{x})^+ \cdot \begin{pmatrix} P_{x_1+r^*} \\ \vdots \\ P_{x_l+r^*} \end{pmatrix} \right) ,
\end{aligned}$$

here r^* is defined via

$$r^* = \operatorname*{argmin}_{r\in\mathbb{Z}} \left\| (\Phi_{1-r,n-r,0,\sigma}(\tilde{x}) \cdot \Phi_{1-r^*,n-r^*,0,\sigma}(\tilde{x})^+ - E) \cdot \begin{pmatrix} P_{x_1+r} \\ \vdots \\ P_{x_l+r} \end{pmatrix} \right\| ,$$

where E denotes the unit matrix.

The last minimum can easily be obtained, since for finite P there is just a finite number of notable values.

We will use the result of this remark also in Section 6.2 where we apply matched wavelets for detection of errors in linear guideways. Since errors of that type can occure at every time, also the results of the following section are valuable in this context.

3.3 Shift-invariance

Discrete wavelet transform has many applications in signal processing and has many advantages like flexibility or efficient implementation and data storage. But there is one disadvantage: Its lack of shift-invariance. According to the definition and

notations given in Definition 2.1 and Lemmata 2.2 and 2.3 it is clear, that for a signal f, a shift parameter $T_\tau f = f(\cdot - \tau)$, arbitrary numbers $j, k \in \mathbb{Z}$ and the dual wavelet $\tilde{\psi}$ holds

$$\begin{aligned} \left\langle T_\tau f \,\middle|\, \tilde{\psi}_{j,k} \right\rangle &= \left\langle f(\cdot - \tau) \,\middle|\, 2^{-j/2}\tilde{\psi}\left(\frac{\cdot - k}{2^j}\right) \right\rangle \\ &= \left\langle f \,\middle|\, 2^{-j/2}\tilde{\psi}\left(\frac{\cdot - k}{2^j} + \tau\right) \right\rangle \neq \left\langle f \,\middle|\, \tilde{\psi}_{\tilde{j},\tilde{k}} \right\rangle \end{aligned}$$

for every $\tilde{j}, \tilde{k} \in \mathbb{Z}$, if $\tau \neq \tau'/2^j$ for any $\tau' \in \mathbb{Z}$.

Clearly the impact of the shift-variance would be especially high, if the set of all $(f_{j,k})_{k\in\mathbb{Z}, j\leq j'}$ would be sparse – or have just a few values exceeding an ϵ – for one fixed shift. This is exactly that case we proposed in Section 3.1 for the application of matched wavelets.

In this section we discuss options for filling this gap, especially we present one that is easily adaptable to our matched wavelet approach. With this results starts that part of the thesis on results developed by ourselves.

But before presenting the result, we need to introduce a natural measure, also used in [54], for quantifying the shift dependence of a given wavelet:

Definition 3.6. *Let T_τ denote the shift operator $T_\tau f = f(\cdot - \tau)$ and define $P_{j,\psi}$ as the projection operator onto the space spanned by $\psi_{j,k} = 2^{-j/2}\psi(\frac{x-k}{2^j})$:*

$$P_{j,\psi} f = \sum_{k\in\mathbb{Z}} f_{j,k}\psi_{j,k} = \sum_{k\in\mathbb{Z}} \left\langle f \,\middle|\, \tilde{\psi}_{j,k} \right\rangle \psi_{j,k} \ ,$$

for biorthogonal wavelets ψ and $\tilde{\psi}$.

Then the shift dependence of ψ for analyzing a function f is defined by

$$e_{j,\psi}(\tau, f) = \frac{\|P_{j,\psi} T_\tau f - T_\tau P_{j,\psi} f\|^2}{\|P_{j,\psi} f\|^2} \ . \tag{3.8}$$

$e_{j,\psi}(\tau, f)$ equals zero for any τ for the continuous wavelet transform and is larger for the discrete one. For our application we need to minimize its value, especially $\max_\tau e_{j,\psi}(\tau, P)$.

Several different ideas have been proposed for minimizing $e_{j,\psi}(\tau, f)$ see, e.g., [87, 88, 143, 54]. Some of these papers use the natural idea of minimizing the measure $e_{j,\psi}(\tau, f)$ over a certain given set of admissible wavelets. However, such an approach reduces the options to incorporate the construction of wavelets that are matched to a given pattern P. We did not consider the second natural idea to add an additional summand for punishing high values of $e_{j,\psi}(\tau, f)$ to Equation (3.6). In that case especially wavelets ψ with jumps or steep slopes would be punished, giving bad approximation results for patterns like presented in Section 6.2 or approximations with less smoothness. Additionally we would lose the linear structure of the problem.

But maybe this method can be used to increase the efficiency of the below one, in case it is necessary.

Some other methods are more flexible but produce redundant data and, hence, enlarge the amount of data to be processed: Best known in this context is the algorithm à trous (cf. [101, p. 154]), a variation of Algorithm 2.4 producing additional data that give shift-invariance of the transform. All in all there are (for an f discretized to N samples) $\log_2 N$ times more data than for a usual biorthogonal transform and the computational complexity increases by the same factor.

Another less known but more flexible attempt of this type is the phaselet concept as proposed in [54]. The basic idea is to find a family of wavelets $\{\psi^\lambda | \lambda \in \Lambda = \{0 \dots L-1\}\}$ with finite wavelet filters $\{g^\lambda\}$, such that ($\tau_\lambda \in [0,1]$):

$$\begin{aligned} \psi^\lambda(x) &\approx \psi(x - \tau_\lambda) \ , \\ \hat{\psi}^\lambda(\omega) &\approx \hat{\psi}(\omega) e^{-i\pi\tau_\lambda(\omega)} \end{aligned}$$

Gopinath proved in [54], that a perfect shift i.e. $\psi^\lambda(x) = \psi(x - \tau_\lambda)$ can not be achieved with finite filters. But this is not necessary in the case of wavelets from Section 3.2, since they are already just approximations to the pattern P.

We define the shift-variance measure of the whole set $\{\psi^\lambda\}$ analog to Equation (3.8) as

$$e_{j,\psi^\Lambda}(\tau, f) = \min_{\lambda \in \Lambda} \frac{\|P_{j,\psi^0} T_\tau f - T_\tau P_{j,\psi^\lambda} f\|^2}{\|P_{j,\psi} f\|^2} \ . \tag{3.9}$$

The additional minimum results in a reduction of the measure for every τ, especially we obtain

$$e_{j,\psi^\Lambda}(\tau_\lambda, f) \approx 0 \ .$$

We can adapt this approach easily and optimally using the lifting approach for constructing matched phaselets (or phase-selflets): For every $\lambda \in \{0 \dots L-1\}$ we define $\tau_\lambda := \lambda / L$. Then the matching wavelets ψ^λ are computed by fitting it to a shifted version of the pattern:

$$\{c^{*,\lambda}, s^{*,\lambda}\} = \operatorname*{argmin}_{c,s} \|c\psi^0 + \varphi \star s - P(\cdot - \lambda/L)\|, \quad c \in \mathbb{R}\backslash\{0\}, s \in \mathbb{R}^{\mathbb{Z}} \tag{3.10}$$

$$\psi^\lambda(x) = c^{*,\lambda} \psi^0(x) - \sum_k s_k^{*,\lambda} \varphi(x - k) \tag{3.11}$$

Note, that shifting P is possible, because of the extension given on Page 36. We can choose the degree of redundancy L such that the measure $e_{j,\psi^\Lambda}(\tau, f)$ stays below an acceptable tolerance ϵ for all $\tau \in [-1/2L, 1/2L]$. This make this approach more flexible than the algorithm à trous, if total shift-invariance is not necessary. Furthermore we can restrict the complexity and amount of data to LN, which is in most cases less than $N \log_2 N$ as for the algorithm à trous.

Of course there is also a variant to increase the invariance of the wavelet transform according to the scaling j, but we will not treat it here in detail. Basically the

same procedure works also in this case, matching m wavelets to m different scalings of P. Up to our knowledge there is no alternative method giving (approximative) shift-invariance in scale.

We apply the results of this chapter in Subsection 6.2.1 as an approach to detect the characteristic signal of an error in a linear guideway. There we see also the necessity to use the shift-invariance results from this section.

Chapter 4

Learning Wavelet-Dictionaries

In the preceding chapter we learned how to match one wavelet ψ to a given pattern P via lifting. Furthermore we generalized this problem to the matching of several wavelet bases to the shifts of one pattern. Considering all this, the question arises, whether also a set of several wavelet bases can be learned, based on a set $\mathcal{Y} = \{y_i\}_{i=1}^N$ of patterns. That case guides us directly into the framework of dictionary learning as presented in Subsection 2.2.2. A dictionary consisting from learned wavelet bases, as defined later on in Equation (4.1), has several advantages in comparison with other learned ones:

- Limited shift-invariance: Many signals being analyzed with a dictionary are time series. Thus the atoms of the dictionary can occure at any position in the signal. According to this it is sensible to use a shift-invariant dictionary for obtaining a sparse decomposition. As pointed out in Section 3.3, a discrete wavelet transform is shift-invariant in a limited sense. Additional desired shifts can be obtained either by increasing the resolution via increasing the number of lifting coefficients in s (i.e. the difference $n - m$ in Equation (3.6)) or by applying the methods we proposed in Section 3.3. Thus a combination of the dictionary from several wavelet bases offers the possibility to use just a necessary density of shifts.

  Till now there exist about a handful of other algorithms producing more or less shift-invariant dictionaries. We already presented MoTIF on Page 20, producing dictionaries which are invariant according to a shift of $n \in \mathbb{Z}$. Other types of shift-invariant learning algorithms are variations of methods already known from Section 2.2.2 like ML (cf. [118, 12]), MAP (see [95]) or gradient based methods (cf. [15, 166]). The last article proposes a general invariance according to a fixed number of linear transforms, but applies (as all the other algorithms) shift-invariance just for all shifts in steps of the discretization distance.

  The idea of learning a wavelet-dictionary had already been proposed in [139, 124] using a MAP approach. Nevertheless, the wavelets there are no wavelets in the sense of Subsection 2.1.1, especially they do not satisfy Constraint (2.5).

- A second preface of wavelet-dictionaries is some scale invariance. It is (cf. Definition 2.1) just valid for scale shifts of 2^j, but there are up to our knowledge no other dictionary learning algorithms (beside the mentioned [139, 124, 166]) offering properties similar to this in combination with shift-invariance in time or rather space domain. Furthermore we mentioned already in Section 3.3 ways to increase this invariance.

- Lastly we want to mention the potential of using the fast wavelet transform from Algorithm 2.4 for obtaining the coefficients, e.g., for the OMP (cf. Page 15).

So we can combine three of the four criteria for choosing a dictionary mentioned in Subsection 2.2.2. I.e. computational speed, consideration of the signals $\mathcal{Y}$ and invariance properties maybe arisen from a model.

We also want to mention the disadvantage of this method of dictionary learning: We are fixed to the framework of wavelets! If it is, due to its structure, not sensible to express a certain family of signals $\mathcal{Y}$ as linear combinations of wavelets, this method will fail. Nevertheless, for a wide field of $\mathcal{Y}$ it would be desirable to have an algorithm producing wavelet-dictionaries.

In Section 4.1 we expose the basic theory of an algorithm fulfilling this task. Meanwhile Section 4.2 contains several details, which occure during implementation of the algorithm. In the last section, 4.3, there is a short discussion about adequate starting values for the algorithm and additional weights, both have a big influence on the quality of result. For better readability of the thesis we postponed applications of the dictionary construction algorithm to Chapter 6. We already composed parts of this chapter, as well as of the belonging applications in [132].

4.1 Derivation of the algorithm

In this section we want to present a modification of the MOD algorithm (cf. Page 19) called MODW (method of optimal direction, wavelets). The advantage of using learned wavelets in a dictionary had already been exposed above. The decision to modify the MOD is based on the following observations on the dictionary learning algorithms from Section 2.2.2 and the desired properties:

- Fast convergence and computation: Every algorithm used till now has a computational bottleneck like inverting a big matrix (MOD), singular value decomposition (K-SVD, UOB), generalized eigenvalue problems (MoTIF) or double iterations. Also the convergence speed of some of the algorithms (especially MAP and ML based ones) is slow. Using lifting we do not have to optimize the atoms d_j at every point, but just the few lifting coefficients in s_j analog to the $\tilde{s}_{m,n}$ in Equation (3.7). This is especially benefical for the MOD algorithm, where the inverted matrix is no longer of size "length of d_j", but of size "length of s".

- Having a well defined quality measure to extremize: The existing algorithms use a wide field of different quality measures, including probabilities (ML, MAP), approximation errors (MOD, K-SVD) and inner products (MoTIF). It is advantageous to use a measure, concerning practicable and applicational reasonable values, and use this measure in every stage of the algorithm. Since we obtain in the case of a wavelet-dictionary a linear connection between the error at each point and the coefficients of s, a method like MOD or K-SVD fits well to this framework.
- Be able to work with any algorithm for sparse coding: As explained in Section 2.2.1, depending on the exact structure of the problem, different sparse coding algorithms are favorably. So a construction algorithm should offer a way to adapt the result to any of this sparse coding algorithms. One step into this direction is the decoupling into two stages like in MOD or K-SVD.

So we decided to develop in the remainder of this section the theory of MODW, an algorithm for learning wavelet-dictionaries (as defined below in Equation (4.1)), closely related to MOD and constructed to satisfy the above mentioned properties. First we derive all components of the algorithm, before we give a closed description in Algorithm 4.1. Below we are using for $\mathcal{Y} = \{y_i\}_{i=1}^N \subset \mathbb{R}^n$, $\mathbf{Y}$, $\mathcal{D} = \{d_l\}_{l=1}^L$, $\mathbf{D}$ and X the definitions given on Pages 14, 16 and 17 with $n = 2^{-\bar{j}-1}$. Furthermore $\mathcal{D}$ is the union of several wavelet bases, i.e.

$$\begin{aligned} \mathcal{D} &= \{d_l\}_{l=1}^L = \bigcup_{j=1}^J \mathcal{D}_j \qquad (4.1) \\ &= \bigcup_{j=1}^J \left\{ \psi_j \left(\frac{\cdot - r_1}{r_2} \right), \varphi_j \left(\frac{\cdot - r_1}{2^{j'}} \right) \middle| |r_1| \leq R_\rho; r_2 = 2^\rho; \bar{j} \leq \rho \leq j'; r_1, \rho \in \mathbb{Z} \right\}. \end{aligned}$$

Theoretically we can choose j' independent for every $1 \leq j \leq J$, nevertheless, we treat them to be equal throughout all applications in this thesis.

Analog to the MOD the MODW is an iterative learning algorithm, it uses a given start dictionary $\mathcal{D}_0$ to find for the given set of signals $\mathcal{Y}$ an optimal dictionary $\mathcal{D}^*$ and the optimal coefficients X by

$$(\mathbf{D}^*, X^*) = \underset{\mathbf{D},X}{\operatorname{argmin}} \|\mathbf{Y} - \mathbf{D}X\|_F, \text{ while } \forall i: \ \|x_i\|_0 \leq T \qquad (4.2)$$

Here Restriction (4.1) on $\mathcal{D}$ is obtained by already starting the algorithm with a union of wavelet bases as in (4.1) and restricting the dictionary update to lifting steps (cf. Section 2.1.2), resulting every time in a new union of wavelet bases.

Since it is in general complicate to optimize $\mathbf{D}$ and X together (cf. Section 5.1 for this point), every iteration of MODW is split into two substeps:

1. Fix $\mathbf{D}$ and compute the best coefficients X.
2. Fix X and compute the best dictionary $\mathbf{D}$.

The first step is solved for a given dictionary by applying one of the sparse coding algorithms from Pages 15–16 (or rather variants with limited $\|x_i\|_0$) to every y_j. This works, since the problem is decoupled for the different $i = 1, \ldots, N$. We prefer this in comparison with updating algorithms like gradient methods, since first this gives an (approximative) exact solution in the bounds of the coding algorithm. Second, the non-continuous structure of ℓ_0-norm is not treatable for most updating algorithms (especially gradient methods) and we prefer to construct a method usable in this case as well as in the ℓ_1 case.

The second step is performed by minimizing Equation (4.2) in direction of $\mathcal{D}$, similar to the MOD stated on Page 19 (for reasons of simplicity we assume all data to be continued by zero). The learning of the wavelet bases is performed by using the lifting scheme basing on CDF wavelets (cf. Subsection 2.1.2), thus the φ_j remain unchanged and we treat them as part of the y_i:

$$\begin{aligned} E^2 &= \|\mathbf{Y} - \mathbf{D}X\|_F^2 \qquad (4.3)\\ &= \sum_{i=1}^{N} \left\| y_i - \sum_{l=1}^{L} x_{i,l} d_l \right\|^2 \\ &= \sum_{i=1}^{N} \left\| y_i - \sum_{j=1}^{J} \sum_{r\in\mathcal{R}} x_{i,j,r} \psi_j \left(\frac{\cdot - r_1}{r_2} \right) - \sum_{j=1}^{J} \sum_{r_1=-R}^{R} x_{i,j,r_1} \varphi_j \left(\frac{\cdot - r_1}{2^{j'}} \right) \right\|^2 \\ &= \sum_{i=1}^{N} \left\| \bar{y}_i - \sum_{j=1}^{J} \sum_{r\in\mathcal{R}} x_{i,j,r} \psi_j \left(\frac{\cdot - r_1}{r_2} \right) \right\|^2 \end{aligned}$$

Here r denotes a double-index $r = (r_1, r_2)$ and

$$\mathcal{R} = \{-R_\rho, \ldots, R_\rho\} \times \{2^\rho | \rho \in \{\bar{j}, \ldots, j'\}\} \qquad (4.4)$$

a set of adequate shifts and scalings[1].

Remembering every used wavelet to result from a spline wavelet by lifting (see Subsection 2.1.2) we obtain

$$E^2 = \sum_{i=1}^{N} \left\| \bar{y}_i - \sum_{j=1}^{J} \sum_{r\in\mathcal{R}} x_{i,j,r} \left(\sum_{k\in\mathcal{K}} s_{j,k} \varphi_j \left(\frac{\cdot - (k + r_1)}{r_2} \right) + c_j \psi_j^0 \left(\frac{\cdot - r_1}{r_2} \right) \right) \right\|^2, \qquad (4.5)$$

where $\mathcal{K}$ is a finite (cardinal number K) subset of $\mathbb{Z}$. If we define for reasons of simplicity $s_{j,\infty} := c_j$, in the same way the infinite shift of φ_j as ψ_j^0 and $\hat{\mathcal{K}} := \mathcal{K} \cup \infty$,

[1] If $j' = j'_j$, depending on j, we will choose $j' = \max\{j'_j\}$ and set the coefficients $x_{i,j,r}$ of the additional wavelets equaling zero.

we get

$$\begin{aligned} E^2 &= \sum_{i=1}^{N} \left\| \bar{y}_i - \sum_{j=1}^{J} \sum_{r\in\mathcal{R}} x_{i,j,r} \sum_{k\in\hat{\mathcal{K}}} s_{j,k} \varphi_j \left(\frac{\cdot - (k+r_1)}{r_2} \right) \right\|^2 \\ &= \sum_{i=1}^{N} \|\bar{y}_i\|^2 - 2 \sum_{i=1}^{N} \sum_{j=1}^{J} \sum_{r\in\mathcal{R}} x_{i,j,r} \sum_{k\in\hat{\mathcal{K}}} s_{j,k} \left\langle \varphi_j \left(\frac{\cdot - (k+r_1)}{r_2} \right) \middle| \bar{y}_i \right\rangle \\ &\quad + \sum_{i=1}^{N} \left\| \sum_{j=1}^{J} \sum_{r\in\mathcal{R}} x_{i,j,r} \sum_{k\in\hat{\mathcal{K}}} s_{j,k} \varphi_j \left(\frac{\cdot - (k+r_1)}{r_2} \right) \right\|^2 . \end{aligned}$$

Using the abbreviations

$$b_{j,r} := \sum_{i=1}^{N} x_{i,j,r} \bar{y}_i \in \mathbb{R}^n \quad \text{and} \quad a_{j_1,r_a,j_2,r_b} := \sum_{i=1}^{N} x_{i,j_1,r_a} x_{i,j_2,r_b} \in \mathbb{R}$$

E^2 shortens to

$$\begin{aligned} E^2 &= \sum_{i=1}^{N} \|\bar{y}_i\|^2 - 2 \sum_{j=1}^{J} \sum_{r\in\mathcal{R}} \sum_{k\in\hat{\mathcal{K}}} s_{k,j} \left\langle \varphi_j \left(\frac{\cdot - (k+r_1)}{r_2} \right) \middle| b_{j,r} \right\rangle \\ &\quad + \sum_{j_1=1}^{J} \sum_{j_2=1}^{J} \sum_{r_a\in\mathcal{R}} \sum_{r_b\in\mathcal{R}} \sum_{k_1\in\hat{\mathcal{K}}} \sum_{k_2\in\hat{\mathcal{K}}} s_{k_1,j_1} s_{k_2,j_2} a_{j_1,r_a,j_2,r_b} \\ &\quad \left\langle \varphi_{j_1} \left(\frac{\cdot - (k_1+r_{a,1})}{r_{a,2}} \right) \middle| \varphi_{j_2} \left(\frac{\cdot - (k_2+r_{b,1})}{r_{b,2}} \right) \right\rangle . \end{aligned}$$

This gives E as a function just of the lifting coefficients $s_{k,j}$. For minimizing E, the zero of the derivative of E^2 in direction of all $s_{p,q},\ 1 \le q \le J,\ p \in \hat{\mathcal{K}}$ has to be found:

$$\begin{aligned} 0 &\stackrel{!}{=} \frac{\mathrm{d}(E^2((s_{k,j})_{k\in\hat{\mathcal{K}},1\le j\le J}))}{\mathrm{d}s_{p,q}} \\ &= -2 \sum_{r\in\mathcal{R}} \left\langle \varphi_q \left(\frac{\cdot - (p+r_1)}{r_2} \right) \middle| b_{q,r} \right\rangle \\ &\quad + 2 \sum_{j=1}^{J} \sum_{k\in\hat{\mathcal{K}}} s_{k,j} \sum_{r_a\in\mathcal{R}} \sum_{r_b\in\mathcal{R}} a_{q,r_a,j,r_b} \left\langle \varphi_q \left(\frac{\cdot - (p+r_{a,1})}{r_{a,2}} \right) \middle| \varphi_j \left(\frac{\cdot - (k+r_{b,1})}{r_{b,2}} \right) \right\rangle \end{aligned}$$

After defining

$$B_{j,k} := \sum_{r\in\mathcal{R}} \left\langle \varphi_k \left(\frac{\cdot - (j+r_1)}{r_2} \right) \middle| b_{k,r} \right\rangle \in \mathbb{R}$$

and

$$A_{k_1,j_1,k_2,j_2} := \sum_{r_a\in\mathcal{R}} \sum_{r_b\in\mathcal{R}} a_{j_1,r_a,j_2,r_b} \left\langle \varphi_{j_1} \left(\frac{\cdot - (k_1+r_{a,1})}{r_{a,2}} \right) \middle| \varphi_{j_2} \left(\frac{\cdot - (k_2+r_{b,1})}{r_{b,2}} \right) \right\rangle \in \mathbb{R}$$

we get the conditions

$$0 = -B_{p,q} + \sum_{j=1}^{J} \sum_{k \in \hat{\mathcal{K}}} s_{k,j} A_{k,j,p,q} \quad \forall\, p \in \hat{\mathcal{K}},\ 1 \leq q \leq J. \tag{4.6}$$

This are $(K+1) \cdot J$ linear conditions for the same number of unknowns, so there exists generally just one solution. This zero of derivative is a minimum, as E^2 is always greater or equal zero and infinite if any $|s_{p,q}|$ goes to infinity.

As a summary we give now a closed description of the MODW algorithm:

Algorithm 4.1 (MODW). *Let $\mathcal{Y} = \{y_i\}_{i=1}^{N} \subset \mathbb{R}^n$ be a given set of signals of length $n = 2^{-\bar{j}-1}$. Furthermore let $\mathcal{D}_0 = \{d_l\}_{l=1}^{L}$ be a given dictionary of the form of Equation* (4.1) *and apply the definitions of $\mathbf{Y}, \mathbf{D}$ and X from Page 17. Then an approximative best dictionary $\mathcal{D}^*$ and coefficients X^* are given by:*

1. *Set m=0, define stopping criterium S.*
2. *Define X_m via a sparse coding algorithm (cf. Pages 15–16):*

$$y_i = \sum_{l=1}^{L} x_{i,l,m} d_l + e_i, \ \text{where } \forall i : \|x_i\|_0 \leq T \tag{4.7}$$

3. *Define $\mathcal{D}_{m+1}$ via*

$$\psi_{j,m+1} = \sum_{k \in \hat{\mathcal{K}}} s_{j,k} \varphi_j(\cdot - k)\ ,$$

   where $s_{j,k}$ is computed using Equation (4.6) *with $X = X_m$.*

4. *If S is satisfied set $X^* = X_m$ and $\mathcal{D}^* = \mathcal{D}_{m+1}$, otherwise set $m = m+1$ and go back to Step 2.*

Every iteration of the dictionary update substep decreases the value of E. If the sparse coding substep would be perfectly and gives always the best approximation of all signals y_i, also this steps would decrease E. In that case E would be monotone decreasing, bounded by zero and by this converging at least to a local minimum. As the algorithms used for sparse coding are just approximative, as mentioned in Subsection 2.2.1, it is not guaranteed to have decreasing E in every iteration of the sparse coding substep. A closer look on this point, including a converging variant, is a main topic of the following Section 4.2, caring about several details of implementing Algorithm 4.1.

But before we give a short remark about another variation of the MODW:

Remark 4.2. *Instead of minimizing Equation* (4.2)*, the – from the analytical point of view – more natural way to compute*

$$(\mathbf{D}^{**}, X^{**}) = \underset{\mathbf{D},X}{\operatorname{argmin}} \|\mathbf{Y} - \mathbf{D}X\|_F + \lambda \sum_{i=1}^{N} \|x_i\|_0 \tag{4.8}$$

is an interesting alternative. A modification like this does not change Step 3 of MODW, but Step 2. Here one simple usage of a sparse coding algorithm for every $i = 1, \ldots, N$ does not work anymore. But since the problem is still decoupled, the minima can still be computed separately for every i:

$$e_i = \min_x \|y_i - \mathbf{D}x\|_2 + \lambda\|x\|_0 \tag{4.9}$$

Methods for achieving an approximation to this are, e.g.:

- *Using OMP and comparing the e_i for every step.*
- *Applying the seldom used backward elimination algorithm (BELA) (cf. [25]), which also changes $\|x\|_0$ stepwise.*
- *Replacing the ℓ_0-norm by a smoother one and using a gradient method, analog to Section 5.2.*

The most efficient method has to be chosen according to the parameters, since the multiple evaluation of the several algorithms can be coded in different optimized ways.

In Subsection 5.1.2 we prove the existence of a minimizer for a more general case and a big class of sparsity measures. Since $\mathcal{D}$ is fixed there and here, that proof is still valid also in this case, despite the wavelet structure of the dictionary here.

4.2 Implementation

In the last section we gave the main theory of the MODW algorithm. Now we want to care about several details, generalizations and simplifications, occuring in the line of its implementation and application.

- Let every signal y_i be composed from atoms or rather wavelets on just one scale ρ_i. Then it would be an interesting modification, to rescale all y_i to a fixed scale ρ_0 and restrict the MODW algorithm to this scale instead of using all scales between $\bar{j}$ and a reasonable chosen j'. Such a restriction is not necessary, but its value for reducing the computational effort and simplifying Equations (4.1) is extremely high (cf. also Section 6.2). Especially there will be no necessity anymore to compute the scalar products of wavelets on different scales. Furthermore using this a priori knowledge would reduce the approximation error, since the (approximative) sparse coding algorithms (cf. Pages 15–16) would produce more probable exact results. Unfortunately there is for an arbitrary application no clue, if the y_i satisfy this constraint, so this simplification can generally be used just in a second computation, after obtaining the scale distribution from the transform.

  We deliver in Subsection 6.2.2 an insight to the differences between application of MODW with one and several scales on the same example and obtain still usable results for the first case. In Section 6.1 we give also some evidence on the usability of the intermediate strategy to use just one scale for generating the dictionary but several ones for sparse coding of further data.

- Sparse coding algorithm: As explained on Pages 19 and 43, theoretically every sparse coding algorithm can be used in MODW, according to the desired accuracy, computation time and application. We know from Pages 15–16, all the sparse coding algorithms are just approximative, beside some cases of high sparsity (i.e. $\|x_i\|_0$ is small enough for every i) or special sparsity measures (cf. Definition 1.2).

  Especially there is always the possibility that the result worsens in some steps of Algorithm 4.1, i.e.

  $$\|y_i - \mathbf{D}_n x_{i,n}\|_2 > \|y_i - \mathbf{D}_n x_{i,n-1}\|_2.$$

  In that case it would not be clear, that Algorithm 4.1 lessens the error $E = \|\mathbf{Y} - \mathbf{D}X\|_2$ at all and converges. To assure this we can force the algorithm to reduce E every iterational half step by comparing for every i separately the approximation errors and choosing the better one:

  $$\tilde{x}_{i,n} = \begin{cases} x_{i,n} & \text{if} \quad \|y_i - \mathbf{D}x_{i,n}\|_2 \leq \|y_i - \mathbf{D}x_{i,n-1}\|_2 \\ x_{i,n-1} & \text{else} \end{cases} \tag{4.10}$$

  Using this we can enforce convergence of the algorithm at least to a stationary point whose error E^* is smaller than the starting error E_0. On the other side the question arises, if some increase of error in the course of iterations would not be positive to avoid unwanted stationary points. For answer we prepared a comparison of results applying Equation (4.10) and ignoring it in [132], a brief summary is given in Section 6.1.

- One detail that arises in every application of MODW is, how to treat the boundaries. The sparse coding uses shifted versions of the wavelets and furthermore the construction of the wavelets via lifting uses shifts of the scaling functions (cf. Equation (2.6)). So if R_ρ (cf. Equation (4.4)) and the length of s in Equation (4.5) are not chosen extremely small in comparison to the length of y_i, there will be some shifts that are at least partly outside the support of the y_i. Nevertheless, since the length of s affects the value of E in Equation (4.3) (see also Equation (3.6)) we can not reduce it to much. Moreover, reducing R_ρ for holding all shifts small would result in an $e_i = y_i - \sum x_{i,l} d_l$ (as defined in 4.7) that generally increases towards the boundaries of y_i, since the mapping assigning every point $\iota \in \{1, \ldots, n\}$ the number of atoms d_j with $\iota \in \operatorname{supp} d_j$ is decreasing towards the boundaries of y_i. Furthermore the approximation would lose the slowly changing scales of the wavelets which have a greater support than the fast changing ones.

  But there are several alternative ways to deal with this problem:

  1. Continue the y_i by zeros and fit the dictionary to this longer signals $\bar{y}_i$.
  2. Cutting of the wavelets by using $\breve{\psi}_{j,r_1,r_2} := \psi_{j,r_1,r_2} \cdot \chi_{\operatorname{supp} y_i}$ instead of ψ_{j,r_1,r_2}.
  3. Using periodized wavelets.

From the point of view of computational efficiency there is almost no difference between the first two options: On the one side zero continuation is more easy for the dictionary update step, it simplifies the computation of A and B, which has here more inner structure than in the case of cutting wavelets. On the other side the added zeros produce larger amounts of data. More important than the computational points are that ones concerning the structure of the data: The approximation errors $\bar{e}_i$ of the $\bar{y}_i$ include an approximation of the additional zeros. So this approach is just sensible, in case a prolongation of the signal would be almost zero too. This is especially not the case, if the y_i are cut-outs of a time series. Similar arguments are valid against the usage of periodized wavelets. Nevertheless we use them several times in Chapter 6, since most wavelet decomposition algorithms are adjusted to periodized wavelets. But we give also some results on the second case there, especially if we restrict the algorithm to wavelets of one scale, as stated on Page 47.

- One problem arising in every dictionary learning algorithm is, that the set of its stationary points contains not necessary only the best combinations of dictionary and coefficients. There are also additional stationary points, which are at most local extrema of the error or rather probability function. In the case of error minimization this is clear, due to the non-convex structure of the ℓ_0-norm and also most alternative sparsity norms (cf. Definition 1.2). Similary the probabilistic complement is not directly treatable. In the case of algorithms based on K-mean (cf. [51]), like MOD, K-SVD (see Pages 19–20) or also MODW, the algorithm produces additional stationary points. This is a standard problem of splitting the optimization of the multidimensional variable (d, x) into two variables d, x and occures also in the framework of probabilistic learning. Basically there are two well known methods to increase the probability of obtaining that dictionary with minimal error:
  - Using a priori knowledge about the dictionary.
  - A posteriori identification of stationary points that are no global optima.

  We already mentioned the use of a priori knowledge in the paragraph on restricting the number of used scales on Page 47 and we do again in Section 4.3 on starting values and weights. According to the second point, we do not concern with the often used method to let the error increase for a few steps and accepts the minimum to be "strong" enough, if we fall into the same minimum again. Here we want to propose the method, already used in [3], that matches with our data structure: Every 10th iteration, the actual dictionary is tested, if there are
  - Wavelet bases $\mathcal{D}_j$ which are used rarely, i.e.

    $$\sum_{i=1}^{N} \|\overline{x_j}_i\|_0 < C_1 \ ,$$

    here $\overline{X}_j = X * \chi_j$ denotes the modified coefficient matrix equaling

    $$\bar{x}_{i,l} = \begin{cases} x_{i,l}, & \text{if } d_l = \psi_{j,r_1,r_2} \ , \\ 0, & \text{otherwise.} \end{cases}$$

– Wavelet bases $\mathcal{D}_j$, $\mathcal{D}_{\breve{j}}$ which are to similar, i.e. $\left\langle \psi_j \,\middle|\, \psi_{\breve{j}} \right\rangle > C_2$.

In that cases this $\mathcal{D}_j$ are replaced by a wavelet basis, approximizing best the least represented training signal,

$$\psi_j = \operatorname*{argmin}_{\psi} \|y_I - \psi\|_2, \quad I = \operatorname*{argmax}_{i=1}^{N} \|y_i - \mathbf{D}(x_i - \overline{x_{j_i}})\|_2 \ .$$

Of course, there are several other criteria the absolute minimum has to satisfy. For Example, the distribution of the errors e_i (defined in Equation (4.7)) is often known a priori. We restricted ourselves to the above mentioned two cases, as this are expressible in terms of the wavelet bases and practicable in the cases of most applications.

- An unwanted way to increase the sparsity of the x_i is to increase $\|\psi_j\|_2$ for all j reducing in return $\|x_i\|_2$. Even if $\|\cdot\|_0$ is (unlike most other sparsity measures, see Definition 1.2) not affected by this, there is the danger to obtain an $x_{i,l}$ below the computational accuracy of the machine.

  The best way to prevent procedures like this is to normalize $\|\psi_j\|_2 = 1$ for all j after every update step. In cases we have no access to the wavelets, e.g., using the fast wavelet transform from Algorithm 2.4 there is the alternative to normalize the lifting coefficients $s_{q,p}$. While for obtaining good numerical properties a normalization $\sum_{p\in\hat{K}} s_{q,p}^2 = 1$ is favorably, Algorithm 2.4 needs $\sum_{p\in\hat{K}} s_{q,p} = 1$ for correct norming of the coefficients, as in that case the filters h and $\tilde{h}$ satisfy Equation (2.5). Holding this property is especially necessary if a variant of OOMP is used as sparse coding algorithm (cf. Page 15), using the fast transform with dual wavelets. In that case we obtain often numerical problems, while in the case of another sparse coding algorithm the transform still works if we mind the norming. Nevertheless, OOMP can be advantageous in the case of wavelet-dictionaries, if no numerical problems occure from Equation (2.5), since the coefficients result computationally advantageous from the dual wavelet transform.

- Absurd lifting: All results about lifting (e.g., [153, 29]) are referring to a fixed coefficient $c_j = s_{j,\infty} = 1$ for the wavelet ψ_j for holding Constraint (2.5). The different amplitudes of the y_i force us, also to optimize this value and renormalize all lifting filters later. So it is crucial, to hold c_j different from zero. Furthermore c_j near zero introduces numerical problems in connection with the normalization. To obtain this purpose we tested two strategies:

  1. Holding the value of c_j constant and changing it only by the normalization of the dictionary. So the $|c_j|$ change more slowly than in the original case.
  2. Treating wavelets with $|c_j| < \epsilon$ as a local minimum and replace them like above.

  The results in Chapter 6 are usually computed using the second alternative or by ignoring this problem at all, if numerical problems do not arise.

- Stopping criteria: There is a wide field of possible stopping criteria for MODW, e.g.: number of iterations, the error $\|\mathbf{Y} - \mathbf{D}X\|_2$ and its variation, sparsity of X or even the sum from Equation (4.8). While the sparsity is fixed in case the coding is made via OMP, there is almost no restriction to the other ones, their usability bases on the application. Most computations in Chapter 6 are made by using a connection of relative change of error within a maximal number of iterations to control computation time.

- For the choice of j' – the maximal scale of the wavelet transform – there are two criteria to consider: First the scales that are existent in the signals y_i. It is clearly sensible, to use wavelets on all scales specified there. On the other side, choosing j' to high increases the computation time of the fast wavelet transform. Furthermore, in the case of periodized wavelets (cf. Page 48), wavelets on the slow changing scales could overlap themselves and by this increase the coherence (cf. Subsection 2.2.1) of the dictionary, and therewith decrease the coding quality of standard sparse coding algorithms like OMP or BP.
  For the computations in Sections 6.2 and 6.3 we used no a priori information on the dominating scales in the signals, so we deciced for a first computation to choose $j' = 0$. Since the wavelets on the scales $j = \{0, -1, -2\}$ have been chosen in almost no case (see Figure 6.21 and 6.27), we did not need to consider the influence of this elements on the coherence of the dictionary, unlike that one on computation time.

4.3 Starting values and weights

The most effective way to influence the result of MODW is via its starting values $\mathcal{D}_0$. In Chapter 6 we mention one example, that for wrong chosen $\mathcal{D}_0$ the resulting stationary point $\mathcal{D}^*$ can produce a higher error E^* than the E'_0 at start for a different dictionary $\mathcal{D}'_0$, while the differences $E_0 - E^*$ and $E'_0 - E'^*$ between start and final error are similar.

For correct choice of the starting dictionary a priori knowledge can be used. Of special value are already approximatively known wavelets like from modelling the signal generating process. Also atoms already separated from the signals are a point for generating the starting dictionary. A method for this separation is presented in Section 6.2 not in the framework of dictionary learning, but as basis for a wavelet matching from Chapter 3. In the cases where no information on the expected dictionary are available, other algorithms (e.g., K-SVD in [3]) use often elements of $\mathcal{Y}$ as starting dictionary. Since we have to take into account also the wavelet constraints, we use the following auxiliary algorithm for obtaining an acceptable $\mathcal{D}_0$:

Algorithm 4.3. *Let $\mathcal{Y}$ be given as in Algorithm 4.1 as well as $\mathbf{Y}$, $\mathbf{D}$ and X. Furthermore take an arbitrary wavelet-dictionary $\tilde{\mathcal{D}}_{0,1}$ as defined in* (4.1).

1. *Fix an $\tilde{N} \in \mathbb{N}$, $\tilde{N}$ divides N and define*

$$\mathcal{Y}_{m_2} = \{y_i | i = \tilde{N}i' + m_2, \tilde{N}i' < N\} \, \forall m_2 \in \mathbb{N}, 1 \leq m_2 \leq \tilde{N}.$$

Furthermore set $m_1 = 0$ *and* $m_2 = 1$.

2. *Compute* X_{m_1,m_2}, *the sparse coding of* $\mathcal{Y}_{m_2}$ *using* $\tilde{\mathcal{D}}_{m_1,m_2}$.

3. *Define the subdictionaries*

$$\mathcal{D}_{m_1,m_2,j} = \{d_{m_1,m_2,l(j)}\} = \left\{ \psi_{m_1,m_2,j}\left(\frac{\cdot - r_1}{r_2}\right) \middle| (r_1, r_2) \in \mathcal{R} \right\} .$$

Find the wavelet basis $\tilde{\mathcal{D}}_{m_1,m_2,\breve{j}}$ *used most rarely, i.e.*

$$\breve{j} = \underset{j=1}{\overset{J}{\operatorname{argmin}}} \sum_{i=1}^{\tilde{N}} \|x_{i,m_1,m_2}\|_0 .$$

4. *Define* $\mathcal{D}^H$ *by* $\mathcal{D}^H_j = \mathcal{D}_{l,m_2,j}$ *for* $j \neq \breve{j}$ *and* $\mathcal{D}^H_{\breve{j}}$ *equals the basis induced by*

$$\psi^H_{\breve{j}} = \underset{\psi}{\operatorname{argmin}} \min_{r_1,r_2} \left\| y_I - \psi\left(\frac{\cdot - r_1}{r_2}\right) \right\|_2, \quad I = \underset{i=1}{\overset{N}{\operatorname{argmax}}} \|y_i - \mathbf{D}(x_i - \overline{x_{\breve{j}i}})\|_2.$$

Furthermore compute the associated sparse coding X^H *of* $\mathcal{Y}_{m_2}$.

5. *Compute the errors*

$$E_{m_1,m_2} = \|\mathbf{Y}_{m_2} - \mathbf{D}_{m_1,m_2} X_{m_1,m_2}\|_F \text{ and } E^H = \|\mathbf{Y}_{m_2} - \mathbf{D}^H X^H\|_F .$$

6. *If* $E^H > E_{m_1,m_2}$

   - *If* $m_2 < \tilde{N}$, *set* $m_1 = 0$, $m_2 = m_2 + 1$, *go back to Step 2.*
   - *If* $m_2 = \tilde{N}$ *set* $\mathcal{D}_0 = \tilde{\mathcal{D}}_{m_1,m_2}$.

   If $E^H \leq E_{m_1,m_2}$ *set* $\tilde{\mathcal{D}}_{m_1+1,m_2} = \mathcal{D}^H$, $X_{m_1+1,m_2} = X^H$ *and* $m_1 = m_1 + 1$, *go back to Step 3.*

It is clear that this algorithm lessens the error E_{m_1,m_2} with every m_1 for fixed m_2. Furthermore the number of steps in direction of m_2 is finite, so Algorithm 4.3 terminates. The sense behind it is, to replace rarely used bases $\mathcal{D}_j$ from the starting dictionary $\mathcal{D}_0$ before starting Algorithm 4.1. This approach is analog to the approach of removing local minima during the course of Algorithm 4.1 as explained in Section 4.2. Please note, this algorithm is an advanced version of the mentioned case of using signals y_i as starting dictionary, specific to MODW. Due to the separation of the signal set $\mathcal{Y}$ into $\tilde{N} > 1$ subsets, the number of tested dictionaries increases (at almost constant calculation time) compared with treating the full $\mathcal{Y}$ at once. Furthermore a second run of the algorithm would be able to improve the dictionary even more. On the other side the danger arises, that optimizing the error in relation to $\mathcal{Y}_{m_2}$ can increase the errors related to the other subsets of $\mathcal{Y}$. In application this problem was almost negligible, so we obtain practical starting dictionaries $\mathcal{D}_0$ for Algorithm 4.1 as presented in Chapter 6. Nevertheless, this algorithm can still be improved, since the current version is mainly searching for an extreme sparse coding

of a few elements of $\mathcal{Y}$, based on the assumption, the other elements of $\mathcal{Y}$ are similar. In case this assumption is not valid, the algorithm must fail.

A second method to influence the course of Algorithm 4.1 for avoiding local minima and other stationary points is to modify the sparse coding algorithms (cf. Pages 15–2.2.1) for increasing the probability of an exact solution. For all these sparse coding algorithms there are variants (cf. [34, 55, 169]) introducing weights. They are influencing the choice of atoms, e.g., in the case of weighted OMP the scalar products of residual and atoms (cf. Equation (2.12)) are multiplied by the atom's weight. Since in the case of dictionary learning there are $N > 1$ signals y_i, we can use the codings of all y_i to generate weights. This weights can later on be used to minimize the approximative errors of the coding algorithm, what increases the chance of convergence to a minimizing combination $\mathcal{D}^*, X^*$. For obtaining this aim we propose the following modification of Algorithm 4.1:

Algorithm 4.4. *Let the assumptions of Algorithm 4.1 be valid. Define furthermore a starting weight $W_0 \in \mathbb{R}^L$, $w_l \in [0,1]$ and replace the sparse coding algorithm by its weighted variant. Then the approximative best weighted dictionary $\mathcal{D}_w^*$, the associated weights W^* and the coefficients X_w^* are given by*

- *Execute Step 1-3 of Algorithm 4.1.*
- *Define for every subdictionary*

  $$\mathcal{D}_{j,r_2} = \{d_{l(j,r_2)}\} = \left\{\psi_j\left(\frac{\cdot - r_1}{r_2}\right)\right\}$$

  according to (4.1) *a ratio*

  $$\alpha_{j,r_2} = \frac{\sum_{i=1}^N \|x_{i,j,r_2}\|_0}{\sum_{i=1}^N \|x_i\|_0},$$

  where x_{i,j,r_2} denotes the elements in the ith line of X connected with the atoms in $\mathcal{D}_{j,r_2}$.
- *Set for every atom $d_l = \psi_j\left(\frac{\cdot - r_1}{r_2}\right)$ the weight*

  $$w_l = c + (1-c)\alpha_{j,r_2},$$

  here c is a fixed constant $0 \leq c < 1$.
- *Execute Step 4 of Algorithm 4.1.*

The idea behind this approach is, that all signals y_i are similar, i.e. they are more or less composed from the same atoms in different shifts. Applying a more general approach, also a $\mathcal{Y} = \mathcal{Y}_1 \dots \mathcal{Y}_z$ composed from several subsets of similar signals work. We resigned to use this modificated algorithm in Subsection 6.2.2, since one of the aims there is the detection of an extremely small group of signals, indicating special statuses of the machinery, thus the indicating atoms of the dictionary have maybe a too small weight. On the other hand, this algorithm could be advantageous for the musical application in Section 6.3, since the atoms there are much more simliar.

Chapter 5

Convergence of Dictionary Learning

All the dictionary learning methods, presented in Subsection 2.2.2, emanate from given discrete signals. This results in dictionaries and coefficients that are discrete, too. From the functional analysis point of view this is not really satisfactory: First since there is no reason, why dictionary methods are not working in the continuous case; second the discretization of a continuous problem could reveal further properties and improved algorithms. Thus we want to analyze a continuous generalization of the dictionary learning problem in this chapter.

There are four main problems arising in the continuous framework: First we have to define the continuous analogons of dictionaries, signals and coefficients. Second we need to find a functional, whose extrema identify optimal dictionaries. Afterwards the existence of this extrema has to be proven, this contains often a prove of weak lower semicontinuity. In a last step we need to find a learning method to obtain the optimal dictionary. Here it is not clear, if the iterative, discrete methods presented in Subsection 2.2.2 can serve as a basis for this task, since they change in general in every step the whole dictionary[1]. Thus the question arises, if convergence to an extremum of E respectively P coincides with convergence of the dictionary.

We found no results concerning this continuous problems in literature, so we propose some definitions and obtain some first results in the following two sections, 5.1 and 5.2. In Subsection 5.1.1 we define the continuous dictionary and deduce an error functional E to be minimized. The remains of Section 5.1 comprise existence proves for an optimal dictionary, minimizing E, and the related coefficients, while the second section, 5.2, shows the convergence of an generalized conditional gradient method for finding a locally optimal dictionary.

Throughout this chapter the measures μ, μ', $\tilde{\mu}$, $\bar{\mu}$ and ν denote measures, where the main theorems for the Lebesgue measure are valid, those are, e.g., weighted Lebesgue or discrete measures.

[1] with the exception of the MoTIF algorithm (see Page 20)

5.1 Existence of an optimal continuous dictionary

This section contains in Subsection 5.1.1 the definition of continuous dictionaries and coefficients. Furthermore, we define there an error functional E, for that we prove the existence of a minimizer, i.e. an optimal dictionary, in Subsections 5.1.2– 5.1.5.

5.1.1 Derivation of the error functional

In the beginning we want to remind briefly of the discrete framework of dictionary learning, serving as basis for the continuous generalization in Definition 5.4. Furthermore, we define for the sake of generality a set of general sparsity functions, applicable in the discrete, as well as in the continuous case:

Let $y_l \in \mathbb{R}^d$ and $x_l \in \mathbb{R}^k$ for all $l \in I$, I being a discrete set[2], $|I| < \infty$. Additionally D is a linear, continuous operator between $\mathbb{R}^d$ and $\mathbb{R}^k$, giving $\|De_j\| = 1$ for all basis-vectors e_j of $\mathbb{R}^k$. Based on Section 2.2.2, especially the MOD algorithm, and Chapter 4, we want to express the problem of learning a dictionary as minimization of an error[3]

$$E_d = \sum_l (\|y_l - Dx_l\|^2 + \lambda_x \|x_l\|_f), \tag{5.1}$$

where λ_x is a positive constant, x_l is a bounded sequence and $\|x\|_f$ its sparsity "norm" as defined in Section 1.2. For reasons of simplicity we recapitulate here the definition of $\|\cdot\|_f$:

$$\|x\|_f = \sum_{i=1}^{k} f(x_i), \tag{5.2}$$

for a function f being a sparsity function in $\mathcal{F}_1$ according to the following definition that is in large part adopted from [69]:

Definition 5.1 (Sparsity functions)**.** *Let $\bar{f}$ be a non-decreasing function of the form $\bar{f} : [0,\infty) \mapsto [0,\infty)$, $\bar{f}(0) = 0$, $\bar{f} \not\equiv 0$. If additionally $t \mapsto \bar{f}(t)/t$ is non-increasing on $(0,\infty)$, then $f : \mathbb{R} \mapsto [0,\infty)$ defined as*

$$f(x) = \begin{cases} \bar{f}(x), \ x \geq 0 \\ \bar{f}(-x), \ x < 0 \end{cases} \tag{5.3}$$

is called a sparsity function of Type 1 on $\mathbb{R}$. The set of all this sparsity functions is called $\mathcal{F}_1$, the set of the $\bar{f}$ is called $\overline{\mathcal{F}}_1$.

Furthermore we define sparsity functions of Type 2 via

$$f \in \mathcal{F}_2 \Leftrightarrow \exists g \in \mathcal{F}_1, \ f(t) = \int_0^{|t|} g(\tau)\,\mathrm{d}\tau$$

[2]Please observe the slightly changes in naming the variables in comparison to Chapters 4 and 6; according to the translation to a continuous problem, we prefer i not to denote an integration variable

[3]We use here a formulation analog to Equation (4.8), since we treat below the functional analysis framework.

For a better understanding of the classes $\mathcal{F}_1$ and $\mathcal{F}_2$ of sparsity functions we give some examples and properties (cf. also [69]):

Example 5.2. *1. For $f, g \in \mathcal{F}_1$ also $\max(f,g)$ and $\min(f,g)$ are elements of $\mathcal{F}_1$.*

2. If $f, g \in \overline{\mathcal{F}}_1$, then $f \circ g$ is in $\overline{\mathcal{F}}_1$.

3. $f(\tau) = |\tau|^\gamma \in \mathcal{F}_1$ for $0 \leq \gamma \leq 1$ and element of $\mathcal{F}_2$ for $1 \leq \gamma \leq 2$. In this case we call $\|\cdot\|_f = \|\cdot\|_\gamma$. In case $\gamma = 0$ we slightly misuse the notation and set f to

$$f(\tau) = |\mathrm{sign}(\tau)| = \begin{cases} 0, & \tau = 0 \ , \\ 1, & \textit{otherwise.} \end{cases}$$

4. $f(\tau) = \arctan(|\tau|) \in \mathcal{F}_1$, as well as $f(\tau) = \tanh(|\tau|) \in \mathcal{F}_1$ as applied in [85].

5. Non-constant, on $[0,\infty)$ monotonically increasing and concave symmetric functions f, satisfying $f(0) = 0$, are elements of $\mathcal{F}_1$, since according to concavity and Definition 5.1:

$$\begin{array}{ccc} \dfrac{f(tx + (1-t)y)}{tx + (1-t)y} & > & \dfrac{tf(x) + (1-t)f(y)}{tx + (1-t)y} \\ & \overset{y=0,x,t>0}{\Longrightarrow} & \\ \dfrac{f(tx)}{tx} & > & \dfrac{f(x)}{x} \ , \end{array}$$

giving the missing condition for a function $f \in \mathcal{F}_1$.

6. There are also non-concave functions in $\overline{\mathcal{F}}_1$, as for example some of the $\bar{f}$ that are positively constant on an interval and (affine) linear increasing everywhere else.

7. Functions $f \in \mathcal{F}_2$ are convex, since their derivatives $g \in \mathcal{F}_1$ are monotonically increasing.

Since $|\cdot|^0$ and $|\cdot|^1$ are elements of $\mathcal{F}_1$, it covers the two most important cases of sparsity constraints (cf. 2.2.1). Gribonval and Nielsen proved in [69], that functions in $\mathcal{F}_1$ satisfy analogous optimality results as $|\cdot|^0$ and $|\cdot|^1$, based on coherence, spark and related properties of the dictionary that are mentioned in Subsection 2.2.1. Also the error estimates touched there hold according to [61]. So they are a natural set for sparsity considerations. Unfortunately functions $f \in \mathcal{F}_1$ are in general non-convex, so there are in general problems with the standard minimization results. Since the elements of $\mathcal{F}_2$ are near to the original $\mathcal{F}_1$ (cf. Example 5.2,3), we use partly the former ones as a convex substitute of the later ones.

A further interesting point concerning $f \in \mathcal{F}_1$ is the following connection to elements of L_p:

Lemma 5.3. *Let $x \in L_p(U)$, for a fixed $p \in (1,\infty]$. Let furthermore $\int_u f(x(\zeta))\,\mathrm{d}\bar{\mu}\zeta = C < \infty$ for an $f \in \mathcal{F}_1$. Then $x \in L_{p'}(U)$ for all $1 \leq p' \leq p$.*

Proof. Let us first assume $p < \infty$: We define $1/c_1 = f(1)$ and furthermore the set A that way, that $|x(\zeta)| < 1$ on A $\bar{\mu}$-almost everywhere and $|x(\zeta)| \geq 1$ on A^C $\bar{\mu}$-almost everywhere. Then we get from Definition 5.1:

$$\begin{aligned}\int_U |x(\zeta)| \,\mathrm{d}\bar{\mu}\zeta &\leq c_1 \int_A f(x(\zeta)) \,\mathrm{d}\bar{\mu}\zeta + \int_{A^C} |x(\zeta)|^p \,\mathrm{d}\bar{\mu}\zeta \\ &\leq c_1 \int_U f(x(\zeta)) \,\mathrm{d}\bar{\mu}\zeta + \int_U |x(\zeta)|^p \,\mathrm{d}\bar{\mu}\zeta = c_1 C + \|x\|_p^p < \infty\end{aligned}$$

So $x \in L_1(U)$ and by this element of all spaces between $L_1(U)$ and $L_p(U)$.

If $p = \infty$, we know:

$$\frac{f(x(\zeta))}{x(\zeta)} \geq \frac{f(\|x\|_\infty)}{\|x\|_\infty} = c_2$$

$\bar{\mu}$-almost everywhere. So we get

$$\int_U |x(\zeta)| \,\mathrm{d}\bar{\mu}\zeta \leq 1/c_2 \int_U f(x(\zeta)) \,\mathrm{d}\bar{\mu}\zeta = C/c_2 \,,$$

which proves the lemma. □

The Definition of E_d in Equation (5.1) can be extended to $y_l \in H$, where H is an arbitrary real HILBERT space. Here I does not need to be a discrete set and furthermore x be just element of some adequate BANACH space. Also $x \in L_\infty$ is in the continuous case not necessary. One general example for this is to define a continuous $\tilde{E}$ by

Definition 5.4 (Continuous dictionary and error measure)**.** *Given a function $y \in L^2((\Omega, I), \mathbb{R})$ and $f \in \mathcal{F}_i$, $i \in \{1, 2\}$ being a sparsity measure according to Definition 5.1. Let x be a function $x : (\Omega', I) \mapsto \mathbb{R}$ and D an integral operator defined by*

$$Dx(\omega, l) = \int_{\Omega'} d(\omega, \omega') x(\omega', l) \,\mathrm{d}\mu'\omega' \,, \tag{5.4}$$

where d is a real function satisfying

$$\int_\Omega d(\omega, \omega')^2 \,\mathrm{d}\mu\omega \equiv \tilde{c}. \tag{5.5}$$

Then $\tilde{E}$ is given by

$$\begin{aligned}\tilde{E}(d, x) &= E_1(d, x) + E_2(x) \\ &= \int_I \left(\int_\Omega (y(\omega, l) - (Dx)(\omega, l))^2 \,\mathrm{d}\mu\omega + \lambda_x \int_{\Omega'} f(x(\omega', l)) \,\mathrm{d}\mu'\omega' \right) \mathrm{d}\nu l \,.\end{aligned} \tag{5.6}$$

with $1 \leq p \leq \infty$.

We see, D or rather d is the continuous equivalent to the discrete dictionary $\mathcal{D}$, especially $d_{\omega'} : \omega \mapsto d(\omega, \omega')$ represents the different atoms of the dictionary. Furthermore, x represents the coefficients and y the given set of signals $\mathcal{Y}$.

We want to choose x as freely as possible, so we postpone the question of the function space defining x for a moment.

The constraint in Equation (5.5) is introduced in analogy to the norming in the discrete case, to prevent d to increase boundless, this would reduce $\tilde{E}$, as:

Remark 5.5. *$\tilde{E}(\lambda d, x/\lambda) \leq \tilde{E}(d,x)$ for $\lambda \geq 1$, because $(\lambda D)(x/\lambda) = Dx$, so*

$$\tilde{E}(\lambda d, x/\lambda) - \tilde{E}(d,x) = \int_I \int_{\Omega'} (f(x(\omega', l)/\lambda) - f(x(\omega', l))) \leq 0 \ ,$$

since f is non-decreasing.

For a better handling of Equation (5.5) we introduce it to $\tilde{E}$ and redefine this as

$$\begin{aligned} E(d,x) &= \tilde{E}(d,x) + E_3(d) \\ &= \tilde{E}(d,x) + \lambda_d \left(\int_{\Omega'} \left(\int_{\Omega} d(\omega,\omega')^2 \, \mathrm{d}\mu\omega \right)^{p'/2} \mathrm{d}\mu'\omega' \ - c \right)^2 \end{aligned} \tag{5.7}$$

with a positive constant λ_d. The different norms in the two directions Ω and Ω' are chosen to hold $Dx \in L^2$. We emphasize that the additional summand in Equation (5.7) does not force the integral of d over Ω to be constant, but to be an integrable function. This condition can in the discrete case be interpreted as a weighted dictionary (cf. [34], [55], or Section 4.3), where the probability of choosing an atom is different, depending on the weight, i.e. here the norm $\|d(\cdot,\omega')\|_{\Omega,2}$. An alternative additional summand like

$$\tilde{E}_3 = \lambda_d \left(\int_{\Omega'} \left| \left(\int_{\Omega} d(\omega,\omega')^2 \, \mathrm{d}\mu\omega \right)^{p'/2} - \tilde{c} \right| \mathrm{d}\mu'\omega' \right)^2 \tag{5.8}$$

would be usable in the case of finite Ω' but we would lose coercivity, as necessary in Theorem 5.13 in most cases of infinite Ω' as pointed out in Example 5.26.

In the remainder of this section we try to find some conclusions about the existence of a minimum of E. According to this, we fix first d – this works as sparse coding in the continuous environment – later fix x and afterwards bring both cases together.

5.1.2 Fixed d

According to Subsection 2.3.3 the existence of a minimum of E can be proven using the so called direct method in the calculus of variations if E satisfies three properties, this are, as brief reminder (cf. also Page 28):

Condition 5.6. *There exists a minimum of E in X, if*

1. *E is proper,*
2. *E is τ-coercive in X,*

3. E is τ-lower semicontinuous in X.

Here τ denotes a fixed weak, weak or strong topology.*

As mentioned in Subsection 2.3.3 the only practical way in continuous environment is to choose the weak topology in some reflexive L_p, $W_{k,p}$ or more complicate space (or rather the weak* in the separable L_∞ or $W_{k,\infty}$), offering an accessible definition of coercivity (cf. 2.15), but paying this with the necessity of proving weak(*)[4] lower semicontinuity, especially complicate in a case like this, where $f \in \mathcal{F}_1$ is not a convex function.

In this subsection we prove the three requirements of Condition 5.6 for a fixed d, satisfying

$$E_3 = \lambda_d \left(\int_{\Omega'} \left(\int_\Omega d(\omega, \omega')^2 \, \mathrm{d}\mu\omega \right)^{p'/2} \mathrm{d}\mu'\omega' \; - c \right)^2 < \infty \tag{5.9}$$

or rather $\tilde{E}_3 < \infty$. Proving the existence of this minimum serves not only as first step of the minimization of the whole E, but the minimum can also be interpreted as the optimal continuous sparse coding for given $d(\omega, \omega')$.

Lemma 5.7 (Properness of E)**.** *For fixed d satisfying* (5.9) *and given $y \in L^2((\Omega, I), \mathbb{R})$, $E(x)$ is a proper functional*

Proof. Fix $x \equiv 0$, then we get

$$\begin{aligned} E(d,0) &= \int_I \left(\int_\Omega (y(\omega,l) - (D0))^2 \, \mathrm{d}\mu\omega \; + \lambda_x \int_{\Omega'} f(0) \, \mathrm{d}\mu'\omega' \right) \mathrm{d}\nu l \; + E_3 \\ &= \int_I \int_\Omega y(\omega,l)^2 \, \mathrm{d}\mu\omega \; \mathrm{d}\nu l \; + E_3 \\ &< \infty \end{aligned}$$

as $y \in L^2((\Omega, I_l), \mathbb{R})$ and $f(0) = 0$. □

For proving the coercivity of E we need to make some additional assumptions on f, as we can see via the following remark:

Remark 5.8. *A freely chosen sparsity measure $f \in \mathcal{F}_1$ (cf. Definition 5.1) can result in a non-coercive E. Examples are:*

- *Let $\|\cdot\|_f = \|\cdot\|_0$ (cf. Example 5.2,2) and $x_\alpha(\omega', l) = \alpha \cdot \chi_{[0,1]^2}(\omega', l)$, then is $\|x_\alpha\|_f = 1$ independent of $\alpha > 0$, while $\|x\|_p = \alpha \to \infty$, for $\alpha \to \infty$, $\forall\, 1 \le p \le \infty$. If we have additionally d with $\int dx_1 \, \mathrm{d}\mu\omega = 0$, then $E = \|y\|_2^2 + \lambda_x$.*

[4] weak(*) denotes weak or rather weak* for the remainder of this chapter.

- *Let* $f(\tau) < t^\gamma$, *for some* $\gamma < 1$ *and* $x_\alpha(\omega', l) = \chi_{[0,1]^2}(\omega', l) \cdot \omega'^{1/\alpha - 1/p} \in L_p$, *then* $\|x_\alpha\|_p \to \infty$ *for* $\alpha \to \infty$, *while*

$$\|x_\alpha\|_f < \int_0^1 \int_0^1 \omega'^{\gamma(1/\alpha - 1/p)} \underset{\alpha\to\infty}{\longrightarrow} \int_0^1 \int_0^1 \omega'^{-\gamma/p} < \infty$$

  as $\gamma < 1$ *and* $p \geq 1$.

This examples show, that $f \in \mathcal{F}_1$ has to increase faster than every $\tau^\gamma, \gamma < 1$ if we want to get the weak coercivity of E out of E_2. Unfortunately this condition is not satisfiable for most $f \in \mathcal{F}_1$. We will prove in Lemma 5.10 that we need to satisfy one of the following conditions for gaining weak coercivity of E:

Condition 5.9. *Let* f *or rather* E *or both satisfy one of the following conditions*

1. $f \in \mathcal{F}_i$, $i \in \{1, 2\}$ *and* $f(\tau)/\tau^p > \gamma$ *with* $\gamma > 0$ *for all* $\tau > 0$, *and an adequate* $p > 1$.
2. $E(d, x) = (E_1 + E_2 + E_3)(d, x) + \tilde{E_4}(x) = E_1 + E_2 + E_3 + \lambda_1 g_1(\|x\|_p)$, *where* $g_1 : \mathbb{R} \mapsto \mathbb{R}$ *is a lower semicontinuous, strong coercive*[5], *convex function with* $g_1(0) = 0$ *and* $1/p + 1/p' = 1$.

For the sake of completeness we define $g_1 \equiv 0$ in the first case. p should hold $p > 1$ to obtain a reflexive space L_p. So we choose $x \in L_p(\Omega', I)$, nevertheless, the integral in E_1 is just finite, if also $x_{\omega'} : l \mapsto x(\omega', l)$ is element of $L_2(I)$ for μ'-almost every ω', thus

$$x \in \{x \in L_p(\Omega, I) | x_{\omega'} \in L_2(I)\} \ .$$

Lemma 5.10. *Let* $y \in L^2((\Omega, I), \mathbb{R})$ *and* f *or rather* E *satisfy Condition 5.9, then* E *is coercive in the weak(*) L_p-topology in the sense of* (2.14).

Proof. We have to prove that both conditions of 5.9 give coercivity of E. If we use the first one we get:

$$\begin{aligned} E(d, x) &\geq \lambda_x \int_I \int_{\Omega'} f(x(\omega', l)) \, \mathrm{d}\nu l \, \mathrm{d}\mu'\omega' \\ &= \lambda_x \iint_{x(\omega', l) \neq 0} \frac{f(x(\omega', l))}{x(\omega', l)^p} x(\omega', l)^p \, \mathrm{d}\nu l \, \mathrm{d}\mu'\omega' \\ &\geq \lambda_x \int_I \int_{\Omega'} \gamma x(\omega', l)^p \, \mathrm{d}\nu l \, \mathrm{d}\mu'\omega' \\ &= \lambda_x \gamma \|x\|_p \end{aligned}$$

In the second line we use $f(0) = 0$ as defined in 5.1.

In case the second condition is valid we have the most direct proof, as here $E(d, x) \geq \lambda_1 c \|x\|_p$ for $\|x\|_p$ according to the coercivity of g_1 in Condition 5.9,2. □

[5] We could choose here weaker condition than strong coercivity, but we will need the strong one in Section 5.2.

If we have a closer look to the conditions of 5.9, we see, that the first one is that one, which is most natural, as it just exceeds for $p = 1$ the disproved cases of the counter-examples. Practical examples are $f \in \mathcal{F}_2$, that are of the form $f(t) = t^p$ for $|t| \geq \rho$ and $p \geq 1$. Unfortunately for $f \in \mathcal{F}_1$ this condition works just for f approximating $|\cdot|_1$ for $p = 1$, and $L_1(\Omega \times I)$ is not reflexive. Thus, throughout this section, we always assume $p > 1$ for achieving a reflexive BANACH space and in the case $p = \infty$ we realize the weak topology as weak* topology. The second alternative in Condition 5.9 is maybe more restrictive on x than the first one, since Lemma 5.3 proved, that in this case x is element of L_1 and even more coercive in L_1. But we recive in return more freedom for the choice of d, since in the first case of Condition 5.9, the summand E_1 forces d to be element of $L_2(L_\infty(\Omega), \Omega')$. Furthermore, we can choose $1 < p < \infty$ giving a reflexive BANACH space, necessary for applying Theorem 2.12,2. In case $p = \infty$ the space is separable and we can apply Theorem 2.12,3 for weak* coercivity.

We now turn to prove weak lower semicontinuity of E. In general this is the hardest to prove one, out of the three parts of Condition 5.6. For simplicity we look at the three summands of E individually. Especially problematic is the second summand, which is not convex in x. For preparation we want to mention, that in cases of non-lower semicontinuous f also weak lower semicontinuity in L^p is not given, as the following remark shows:

Remark 5.11. *A non-(strong)-lower semicontinuous function f is not weak lower semicontinuous, too. This is easily to prove, as the condition for both properties is the same (cf. [171, Def 38.4]), but in the strong case just for a subset of sequences for which it has to be valid for the weak case.*

Analogous, for most f weak continuity of f and E_2 is not given (cf. Lemma 5.29), but here weak lower semicontinuity is sufficient. In the case of convex functionals this is equivalent to (strong) lower semicontinuity (cf. Theorem 2.14). The point is, sparsity measures in $\mathcal{F}_1$ are in general non-convex and then a proof of weak lower semicontinuity is in general a hard task. This is especially valid, since the class of sparsity functions $\mathcal{F}_1$ is quite wide. Nevertheless, there are some elements of $\mathcal{F}_1$, where the weak lower semicontinuity can be proven directly and a for greater subset of $\mathcal{F}_1$ it can be disproven. The trivial positive example is $f(\tau) = \sigma \cdot |\tau|$. In this case we have $\|\cdot\|_f = \sigma\|\cdot\|_1$, being convex, so the weak lower semicontinuity follows from the strong one according to Theorem 2.14. The less trivial but important negative case is the following one:[6]

Lemma 5.12. *If $f \in \mathcal{F}_1$ and $f(\tau)/\tau$ is strictly monotonically decreasing on $(0, \infty]$, then $E_2 = \int_I \int_{\Omega'} f(\cdot)\, \mathrm{d}\mu' \omega'\, \mathrm{d}\nu l$ is not weak lower semicontinuous.*

Proof. For proving the Lemma we fix $z \in \mathbb{R}^+$, define $x_n(\omega', l)$ as

$$x_n(\omega', l) = \begin{cases} z & l \in [0,1], \omega' \in [0, 2^{-n}] + k2^{-(n-1)}, k = 0, \ldots, 2^{n-1} - 1 \\ 0 & \text{otherwise} \end{cases} \tag{5.10}$$

[6]

and prove that $x_n \rightharpoonup x = z/2\chi_{[0,1]^2}$, while $\lim_{n\to\infty} E_2(x_n) < E_2(x) = M > 0$. This is done by computing $\int_{[0,1]}\int_{[0,1]} x_n\phi \,\mathrm{d}\mu'\omega' \,\mathrm{d}\nu l$ for a continuous function ϕ (continuous functions on compact sets are regulated functions and C_c^∞ is dense in L_p, $p \in [1,\infty)$):

$$\begin{aligned}\int_{[0,1]}\int_{[0,1]} x_n\phi \,\mathrm{d}\mu'\omega' \,\mathrm{d}\nu l &= z\sum_{k=0}^{2^{n-1}-1}\int_{[0,1]} \phi\chi_{(k2^{-(n-1)}+[0,2^{-n}])}\,\mathrm{d}\mu'\omega' \\ &= z\sum_{k=0}^{2^{n-1}-1} 2^{-n}\phi(\xi_k) = z/2\sum_{k=0}^{2^{n-1}-1} 2^{-(n-1)}\phi(\xi_k)\ .\end{aligned}$$

The prelast step is, with $\xi_k \in k2^{-(n-1)} + [0, 2^{-n}]$, valid according to the mean value theorem. Since last term can be interpreted as integral over a step function converging for $n \to \infty$ in supremum norm towards ϕ, the whole integral converges to $\int_{[0,1]}\int_{[0,1]} z/2\phi \,\mathrm{d}\mu'\omega' \,\mathrm{d}\nu l$, so we proved the weak convergence of $x_n \rightharpoonup x$.

Computing E_2 for x_n as in Equation (5.10) and for x we obtain:

$$E_2(x_n) = \int_{\Omega'}\sum_{k=0}^{2^{n-1}-1} f(z)\chi_{k2^{-(n-1)}+[0,2^{-n}]}\,\mathrm{d}\mu'\omega' = 1/2f(z) \tag{5.11}$$

and

$$E_2(x) = \int_I\int_{\Omega'} f(x(\omega', l))\,\mathrm{d}\mu'\omega' \,\mathrm{d}\nu l = f(z/2) \tag{5.12}$$

Since $f(\tau)/\tau$ is strictly monotonically decreasing on the positive axis we know $\frac{f(z/2)}{z/2} > \frac{f(z)}{z}$, thus E_2 is not weak lower semicontinuous according to Equations 5.11 and 5.12. □

Even if we know the the weak lower semicontinuity (in L^p, $p \geq 1$) of the very important case $f = |\cdot|^1$, in general weak lower semicontinuity of E_2 is not valid. So we see three alternatives:

1. Using only elements of the subclass $\mathcal{F}_{wlsc} \subset \mathcal{F}_1 \cup \mathcal{F}_2$ where weak lower semicontinuity is given.
2. Using relaxation (cf. [7] or [27]).
3. Try to find the weak lower semicontinuity in a different space.

It is clear, that $\mathcal{F}_{wlsc}$ is a superset of $\mathcal{F}_2$ since the elements of the later one are convex and differentiable.

In the following we focus on the first and third alternative. For a short adaption of the second one see the appended Section B.1.

We handle this two interesting options – to use an $f \in \mathcal{F}_{wlsc}$ or to change the space we search weak lower semicontinuity in – at once. In the second case it is additionally necessary also to change the space in which E is weak coercive. One possible solution is to define

$$\hat{E} = E + E_4 = E + (\lambda_1 g_1(\|x\|_p) + \lambda_2 g_2(\|\nabla x\|_p)), \tag{5.13}$$

Here we have $g_2 : \mathbb{R} \mapsto \mathbb{R}$ to be a convex, lower semicontinuous, strong coercive[7] function with $g_2(0) = 0$ in case E_2 is not weak lower semicontinuous in L_p. Otherwise we choose $g_2 \equiv 0$. g_1 is defined according to Condition 5.9. This gives (together with Lemma 5.10) in Theorem 5.13 (weak) coercivity of $\hat{E}$ in $W_{1,p}(\Omega', I)$ or, if $f \in \mathcal{F}_{wlsc}$, in $L_p(\Omega', I)$ and we have to search for weak lower semicontinuity in the same space. We chose here a SOBOLEV space, since the Embedding Theorem 2.20 can serve as a basis for further analysis.

Using this $\hat{E}$ we get a result for the existence of a minimum of $\hat{E}$ in both cases. For simplicity we give first a special case:

Theorem 5.13 (Minimum of E according to x)**.** *Let $y \in L^2(\Omega, I)$ with bounded Ω, additionally let f or rather E satisfy the conditions according to Condition 5.9, f be strong continuous and $g_2 = |\cdot|^p$. Then $\hat{E}$ has a minimum in $W_{1,p}(\Omega, I)$, $p > 1$.*

Proof. We have to prove that $\hat{E}$ is proper, weak coercive and weak lower semicontinuous. For the properness the proof for proper E in 5.7 is still valid, as for $x \equiv 0$ also $\|\nabla x\|_2 = 0$. Weak coercivity is now given in $W_{1,2}(\Omega, I)$, as according to Lemma 5.10: $E \geq c\|x\|_p$ for great enough $\|x\|_p$, this gives

$$\hat{E} \geq c\|x\|_p + \lambda_2\|\nabla x\|_p \geq c\|x\|_{1,p} \text{ for } \|x\|_{1,p} \to \infty$$

For proving the weak lower semicontinuity of $\hat{E}$ we separate $\hat{E}$ again into its summands, as the sum of several weak lower semicontinuous functions is, due to [27, Proposition 1.9], weak lower semicontinuous again. E_1 and E_3 are weak lower semicontinuous as shown in the proof of Theorem B.1, also the first summand of E_4 is clear as to its definition. For the weak lower semicontinuity of E_2 and the second summand of E_4 we use the result of Theorem 2.23: This two summands together are given by

$$E_2 + E_4 = \iint f(x(\omega', l)) + \lambda_2 \left|\nabla x(\omega', l)\right|^p \, \mathrm{d}\mu'\omega' \, \mathrm{d}\nu l \ .$$

According to Theorem 2.23 we have to prove, that $g((\omega', l), x, \nabla x) = f(x) + |\nabla x|^p$ is a CARATHÉODORY function, $g((\omega', l), x, \cdot)$ is convex and $g((\omega', l), x, \nabla x) \geq \langle a(\omega', l) \,|\, \nabla x \rangle + b(\omega', l)$ for some L^p function a and L^1 function b.

1. g is a CARATHÉODORY function according to Definition 2.22, as
   - $g((\omega', l), \cdot, \cdot) = f(x) + |\nabla x|^p$ is continuous (f is continuous according to the assumptions of the theorem) and
   - $g(\cdot, x, \nabla x) = \text{const.}$ is measurable,
2. $g \geq 0$ for all $x, \nabla x$, so defining $a, b \equiv 0$ holds,
3. as $|\cdot|^p$ is convex, g is also convex in ∇x.

[7] This constraint can be weakened here, but we will need it in Section 5.2.

So we can apply Theorem 2.23 and get weak lower semicontinuity of $\hat{E}$ in every $W_{1,p}(\Omega, I)$, $p \geq 1$. Altogether we get the existence of a minimum of $\hat{E}$ via Condition 5.6 in $W_{1,p}(\Omega, I),\ p > 1$. □

Corollary 5.14. *The arguments for weak lower semicontinuity in the proof above are also valid for each summand E_2 and E_4 separately. So the results are especially valid if E_2 or E_4 is weak lower semicontinuous by definition. So we get the result for all definitions of g_1 and g_2 according to Condition 5.9 or rather Equation* (5.13)*. Especially in the case $f \in \mathcal{F}_{wlsc}$ with $g_2 \equiv 0$, the result is valid in $L_2(\Omega', I)$.*

In that cases, where E_2 is not weak lower semicontinuous in $L_p(\Omega, I)$, and we obtain the existence of a minimum just in $W_{1,p}(\Omega, I)$, it would be interesting, especially according to the applicational effects, if this condition can be reduced, e.g., to a space X, $W_{1,p}(\Omega, I) \subset X \subset L_p(\Omega, I)$. In general, we have no evidences for results like this. Nevertheless, in the most interesting cases $f(\tau) = \tau^0$ and $f(\tau) = |\tau|$ this is not necessary and we can treat now the minimization according to d.

5.1.3 Fixed x

In difference to the previous subsection we want to fix x and minimize $\hat{E}$ according to d now. In this case there are again just two summands of $\hat{E}$ being not constant, namely E_1 and E_3. Again we have to prove properness, weak coercivity and weak lower semicontinuity for $\hat{E}$.

Lemma 5.15. *For $y \in L_2$ and x chosen to hold E_2 and E_4 finite, $\hat{E}$ is proper.*

Proof. Choose $d \equiv 0$, then

$$\begin{aligned} \hat{E} &= \iint \left(y - \int 0 \cdot x \right)^2 + E_2 + \lambda_d \left(\left(\iint 0^2 \right) - c \right)^2 + E_4 \\ &= \|y\|_2 + E_2 + \lambda_d c + E_4 < \infty \ . \end{aligned}$$

□

To prove the weak coercivity of $\hat{E}$, we use a similar approach as in Lemma 5.10: We prove weak coercivity of E_3 which is enough, as E_1 is positive. E_1 can not be used to show coercivity in every case, as the following example shows:

Example 5.16. *Let $y \in L_2$ and choose $d \neq 0$ orthogonal to x, i.e. $\int x d \, \mathrm{d}\mu' \omega' \equiv 0$. In this case*

$$E_1(d) = \|y\|_2^2 = E_1(\sigma d)$$

for every $\sigma \in \mathbb{R}^+$.

In the following $\| \cdot \|_{(2,p)}$ denotes the norm in $L_{(2,p)} = L_2(L_p(\Omega'), \Omega)$.

Lemma 5.17. *Under the assumptions of Lemma 5.15 E_3 is weak coercive in $L_2(L_p(\Omega'), \Omega)$.*

Proof.

$$E_3 = \lambda_d(\|d\|_{(2,p)}^p - c)^2 \geq \lambda_d/2\|d\|_{(2,p)}^{2p}$$

for $\|d\|_{(2,p)}^p > (2+\sqrt{2})c$. □

To prove the weak lower semicontinuity of E for fixed x is again the most complicate part. Especially in this case we can disprove it for d in $L_2(\Omega, \Omega')$, as the following result shows:

Example 5.18. *Let (d_n) be a orthonormal sequence in $L_2(\Omega, \Omega')$ and $c = 1$. Then $d_n \rightharpoonup 0 = d$ and $E_3(d_n) = \lambda_d(\|d_n\|_2^2 - 1)^2 = 0$ for all n, but $E_3(d) = 1$.*

Similar results are valid for $p \neq 2$. For obtaining a positive result, we would need a weak continuous surrogate function g for $\|\cdot\|_{(2,p)}$. In that case the surrogate $E_{3,g} = \lambda_d(g(d) - c)^2$ for E_3 would be weak lower semicontinuous. Furthermore the surrogate function needs to have similar properties like the norm, to fit into the context of dictionary learning. As as there is, up to our knowledge, no concrete g like this, we have again to find the weak lower semicontinuity (and also coercivity) in another space than $L_{(2,p)}$.

Again a SOBOLEV space, here $W_{1,(2,p)}$, is a good choice, similar to the previous subsection. Especially, as in this case the existence of a derivative of d is more easy to justify as above. In direction of ω this is equivalent to forcing the atoms to be derivable. In the other direction it holds neighboring atoms similar. So let us define

$$\tilde{E} = \hat{E} + E_5 = \hat{E} + \lambda_3 g_3(\|\nabla d\|_{(2,p)}) \tag{5.14}$$

here $g_3 : \mathbb{R} \mapsto \mathbb{R}$ is a lower semicontinuous, convex, strong coercive[8] function of, satisfying $g_3(0) = 0$. As an example we give $g_3(\|\nabla d\|_{(2,p)}) = \|\nabla d\|_{(2,p)}^{1+\epsilon}$.

Lemma 5.19. *Given finite sets Ω, Ω', I, $y \in L_2(\Omega, I)$, a sparsity measure (cf. Definition 5.1) f and a fixed x assuring E_2, E_4 to be finite. Then $\hat{E}$ is weak lower semicontinuous according to $d \in W_{1,2}(W_{1,p'}(\Omega'), \Omega)$.*

Proof. We prove weak lower semicontinuity for every summand of $\hat{E}$ separately. Later we use [27, Proposition 1.9] to get the weak lower semicontinuity of the sum. As E_2 and E_4 are constant for fixed x, and E_5 is already weak lower semicontinuous by definition, we just have to concern with E_1 and E_3:

- In the proof of Theorem B.1 we observed, that E_1 is continuous and convex in x for fixed d. Since E_1 is symmetric related to the exchange of d and x, we have also for fixed x continuity and lower semicontinuity of E_1 in d and therewith we get weak lower semicontinuity by Theorem 2.14.

[8] strong coercivity is not necessary here, but in Section 5.2.

- For proving the weak lower semicontinuity of E_3 we see first, that $\|\cdot\|^2_{(2,p)}$ is weak continuous in $W_{1,(2,p)}$ according to Lemma 2.21, based on SOBOLEV's Embedding Theorem 2.20. Using this we have also

$$E_3 = \lambda_d(\|d\|^{2p}_{(2,p)} - 2\|d\|^p_{(2,p)} + 1)$$

  to be weak lower semicontinuous in $W_{1,(2,p)}$.

□

Putting the results of this subsection together, we get easily the following theorem:

Theorem 5.20 (Minimum of $\tilde{E}$ according to d)**.** *Let Ω, Ω', I be finite and $y \in L_2(\Omega, I)$. Furthermore let $f \in \mathcal{F}_i$, $i \in \{1, 2\}$ be a sparsity measure according to Definition 5.1 and x be fixed to hold E_2 and E_4 finite. Then there exists a minimum of $\tilde{E}(d, x)$ in $W_{1,(2,p)}$ according to d.*

Proof. According to Condition 5.6 we need to proof properness, weak coercivity and weak lower semicontinuity of $\tilde{E}$. The three preceding lemmata give most of this properties, we just have to prove the coercivity in $W_{1,(2,p)}$. This is given immediately, using Lemma 5.17, as

$$\tilde{E} = \hat{E} + \lambda_3 g_3(\|\nabla d\|_{(2,p)}) \geq c\|d\|_{(2,p)} + \lambda_3\|\nabla d\|_{(2,p)} \geq c\|d\|_{1,(2,p)}$$

For $\|d\|_{1,(2,p)} \to \infty$. □

5.1.4 Free x and d

In the preceding two subsections we tried to find a minimum for E with one fixed argument. Alternating fixing one argument and optimizing the other one is a principle, often used in dictionary learning algorithms in Subsection 2.2.2. But already in two dimensions such strategies often fail to find the absolute minimum as, e.g., minimizing

$$g(x, y) = |x - y| + a(x + y)$$

on $[0, 1]^2$ shows. So we want to optimize both parameters simultaneous now. Doing that, we use some of the foregoing results. First we want to show, just combining this results is not enough. Especially E_1 is not convex anymore in the two variables d and x and even more, not weak lower semicontinuous in $(L_2(\Omega, \Omega'), L_2(\Omega, I))$:

Example 5.21. *Let $y, d_1, x_2 \equiv 0$, $d_2 \equiv c_1$ and $x_1 \equiv c_2$. Additionally I, Ω and Ω' are finite. In this case we have $E_1(d_1, x_1) = 0 = E_1(d_2, x_2)$, while*

$$E_1(tx_1 + (1 - t)d_2) = (t(1 - t)c_1c_2)^2 \, |(\Omega \times I)| > 0 \ .$$

Example 5.22. *For $p = 2$ define*

$$d_n(\omega, \omega') = \sin(n \cdot \omega') \cdot \chi_{[0,2\pi]^2}(\omega', \omega)$$

and

$$x_n(\omega', l) = \sin(n \cdot \omega') \cdot \chi_{[0,2\pi]\times[0,1]}(\omega', l) \ .$$

We get

$$d_n \rightharpoonup d \equiv 0 \quad \textit{and} \quad x_n \rightharpoonup x \equiv 0 \ ,$$

since both are orthogonal sequences. So we have for an arbitrary y:

$$\int_I \int_\Omega (y - Dx)^2 \, \mathrm{d}\nu l \ \mathrm{d}\mu\omega = \int_I \int_\Omega \left(y - \int_0^{2\pi} dx \, \mathrm{d}\mu'\omega' \right)^2 \mathrm{d}\nu l \ \mathrm{d}\mu\omega = \|y\|_2^2 \ ,$$

while

$$\begin{aligned} \int_I \int_\Omega (y - D_n x_n)^2 \, \mathrm{d}\nu l \ \mathrm{d}\mu\omega &= \int_I \int_\Omega \left(y - \int d_n x_n \, \mathrm{d}\mu'\omega' \right)^2 \mathrm{d}\nu l \ \mathrm{d}\mu\omega \\ &= \int_I \int_\Omega \left(y - \int_0^{2\pi} \sin^2(n\omega') \, \mathrm{d}\mu'\omega' \right)^2 \mathrm{d}\nu l \ \mathrm{d}\mu\omega \\ &= \int_I \int_\Omega (y - \pi)^2 \, \mathrm{d}\nu l \ \mathrm{d}\mu\omega \ . \end{aligned}$$

Thus we have no weak lower semicontinuity for $\|y\|_2 > \|y - \pi\|_2$.

If we change p, similar examples will work. Thus we have to find the weak lower semicontinuity of $E_1(d, x)$ in another space than $(L_{(2,p')}(\Omega, \Omega'), L_p(\Omega', I))$. Since we already need to introduce SOBOLEV spaces in the preceding two subsections, we try to prove the weak lower semicontinuity in some similar space also in this case. As we already proved the weak lower semicontinuity of E according to x in $W_{1,p}$ (or rather L_p) and of d in $W_{1,p'}$ as well as the weak coercivity, we try to prove weak lower semicontinuity in $(W_{1,(2,p')}(\Omega, \Omega'), W_{1,p}(\Omega', I))$ or in case it is possible (cf. Theorem 5.13) in $(W_{1,(2,p')}(\Omega, \Omega'), L_p(\Omega', I))$. For simplicity we denote the norm in this spaces with $\|\cdot\|_{1,*}$ or rather $\|\cdot\|_* = \|\cdot\|_{0,*}$. In the following we denote both together by using $\|\cdot\|_{i,*}$, $i \in \{0, 1\}$.

For the proof of the existence of the minimum of $\tilde{E}$, the properness and weak coercivity of E can almost be copied from the previous sections, just for completeness:

Remark 5.23. *For a given $y \in L_2$ and a sparsity norm f according to Condition 5.9, $\tilde{E}$ is weak coercive in $(W_{1,(2,p')}, W_{1,p})$ if g_2 in Equation* (5.13) *is not constant zero, or rather $(W_{1,(2,p')}, L_p)$ otherwise.*

Proof. According to the proof of Theorem 5.13, there is a constant c, giving

$$\tilde{E} \geq C\|x\|_{i,p} \text{ for } \|x\| > 1$$

with $i \in \{0, 1\}$ depending on g_2. Additionally we have by Lemma 5.17 that

$$\tilde{E} \geq C\|d\|_{1,(2,p')}$$

for $\|d\|_{1,(2,p')} \geq \bar{C}$. Altogether we obtain the weak coercivity of $\tilde{E}$ in x and d

$$\tilde{E} \geq C(\|x\|_{i,p} + \|d\|_{1,(2,p')}) = C\|(d,x)\|_{i,*}$$

for $\|(d,x)\|_{i,*}$ greater than a constant. □

Remark 5.24. *For a given $y \in L_2$, $\tilde{E}$ is proper in $W_{i,*}$. We can choose, e.g., $x \equiv 0$ and $d = c\chi_{[0,1]^2}$ for getting $\tilde{E}(d,x) = \|y\|_2^2$.*

Now we have all the ingredients for proving the existence of a minimum of E in the two variables d and x:

Theorem 5.25 (Minimum of $\tilde{E}(d,x)$)**.** *Let $y \in L_2(\Omega, I)$ and f or rather E satisfy Condition 5.9. Furthermore let Ω, Ω' and I be bounded regions or rather intervals. Then there exists a minimum of $\tilde{E}$ in $W_{i,*}$ with $i = 0$ if f is weak lower semicontinuous and $i = 1$ otherwise.*

Proof. According to Condition 5.6 we have to prove properness, weak coercivity and weak lower semicontinuity of $\tilde{E}$. The first two properties are already proven by the foregoing remarks. For the weak lower semicontinuity we consider each summand of $\tilde{E}$ for its own.

- $E_5 + E_4$ is weak lower semicontinuous in $W_{i,*}$, according to Theorem 2.14, as the sumands are defined as convex and strong lower semicontinuous functions in Condition 5.9 and Equations (5.13) and (5.14).
- E_3 and E_2 are weak lower semicontinuous in (d,x), as they are by Lemma 5.19 or rather Theorem 5.13 weak lower semicontinous in d and constant in x or rather vice versa.
- For proving the weak lower semicontinuity of E_1 first we prove weak continuity of $Dx(\omega,l) = \int_{\Omega'} x(\omega',l)d(\omega,\omega')\,\mathrm{d}\mu'\omega'$ in $(W_{1,p'}, W_{i,p})$ for fixed (ω,l): We can do this directly! If there is a sequence $d_n(\omega,\cdot) = d_{n,\omega} \rightharpoonup d_\omega = d(\omega,\cdot)$ in $W_{1,p'}$ and a sequence $x_n(\cdot,l) = x_{n,l} \rightharpoonup x_l = x(\cdot,l)$ in $W_{i,p}$ we know:

$$\begin{aligned}
&\left|\int_{\Omega'} (d_{n,\omega}(\omega')x_{n,l}(\omega') - d_\omega(\omega')x_l(\omega'))\,\mathrm{d}\mu'\omega'\right| \\
= &\left|\int_{\Omega'} ((d_{n,\omega}(\omega')x_{n,l}(\omega') - d_\omega(\omega')x_{n,l}(\omega')) \right. \\
&\qquad \left. +(d_\omega(\omega')x_{n,l}(\omega') - d_\omega(\omega')x_l(\omega')))\,\mathrm{d}\mu'\omega'\right| \\
\leq &\left|\int_{\Omega'} x_{n,l}(\omega')(d_{n,\omega} - d_\omega)(\omega')\,\mathrm{d}\mu'\omega'\right| + \left|\int_{\Omega'} d_\omega(\omega')(x_{n,l} - x_l)(\omega')\,\mathrm{d}\mu'\omega'\right|
\end{aligned}$$

It is clear, that the second integral converges towards zero, since $d_\omega \in W_{1,p'}$ and $x_{n,l}$ converges weakly. For the first summand we see, that

$$\left|\int_{\Omega'} x_{n,l}(\omega')(d_{n,\omega} - d_\omega)(\omega')\,\mathrm{d}\mu'\omega'\right| \leq \|x_{n,l}\|_p \|d_{n,\omega} - d_\omega\|_{p'}.$$

According to a special case of the SOBOLEV embedding theorem (cf. Lemma 2.21, Ω' is bounded according to the assumptions of this theorem) we know, that from the weak convergence of $d_{n,\omega}$ in $W_{1,p'}$ follows strong convergence in $L_{p'}$. This holds independently of p'. Furthermore we know from Theorem 2.12,1 that $\|x_{n,l}\|_p \leq C$ for a positve constant C. All in all we know about the convergence of the first summand towards zero and so about the weak continuity of $\int_{\Omega'} d_\omega(\omega')x_l(\omega') \,\mathrm{d}\mu'\omega'$.

In a second step we prove now weak lower semicontinuity of $\iint (y-u)^2 \,\mathrm{d}\mu\omega \,\mathrm{d}\nu l$ in $W_{i,(2,p)}$. For this we define

$$h((\omega, l), u, \xi) = h(u) = (y-u)^2$$

It is easy to see, that $h(u)$ is a CARATHÉODORY function as defined in Definition 2.22[9], additionally $h(u) \geq 0$ for all u and h is constant in ξ, so especially convex. So, according to Theorem 2.23, h is weak lower semicontinuous in $W_{1,2}(\Omega, I)$.

In combination this two results give weak lower semicontinuity of E_1. This result raises as $d_n \rightharpoonup d$ in $\Omega \times \Omega'$ implies weak convergence in Ω' for μ-almost every fixed ω. Analogous results hold for x. So according to the weak continuity of Dx we get $\int d_n(\omega, \omega')x_n(\omega', l) \,\mathrm{d}\mu'\omega' \to \int d(\omega, \omega')x(\omega', l) \,\mathrm{d}\mu'\omega'$ for μ, ν-almost every ω and l. This results in the weak lower semicontinuity of E_1 according to the second step.

□

So we know about the existence of a minimum of $\tilde{E}(d, x)$ in $W_{i,*}$. Proving uniqueness of the minimum as given for the discrete case (modulo resorting and norming) of K-SVD in [4] would be an interesting point, but we prefer to go into the direction of constructing a minimum as stated in Section 5.2. But before we want to improve the preceding results in the next subsection.

5.1.5 Additional results

We saw in Example 5.18, there is no weak lower semicontinuity of $\tilde{E}$ in $(L_2(L_{p'}), L_p)$, so we can not expect the existence of a minimum in the case of minimizing $\tilde{E}$ in $(L_2(L_{p'}), L_p)$. But there are several other conditions we maybe can reduce, for these we want to have a look in this subsection. Especially it is interesting, if there is the possibility to use an unbounded Ω and a non-continuous sparsity measure f.

unbounded domains

The proofs in the preceding three subsections are given mostly just in cases of bounded Ω, Ω' and I. An interesting question is, if the results stay true in the case

[9] Alternatively, this proof can be made completely direct without using Theorem 2.23, we prefered this one, since it is more easy to adapt if the problem changes slightly.

of unbounded domains. For this reason we go over the proofs again and see, where we use the restriction of a bounded domain. All in all we find the following points:

1. In Equation (5.8) is given an alternative for bounded domains, does it work also for unbounded?
2. In Theorem 5.13 as well as in Lemma 5.19 and Theorem 5.25 weak lower semicontinuity of functionals had been proven just on bounded domains.

For the alternative given in Number 1, there is a easy counter-example disproving the weak coercivity of this term in unbounded domains:

Example 5.26. *Let* $\Omega' = \mathbb{R}$, $\Omega = [0, 2\pi]$ *and* $d(\omega, \omega') = \tilde{c}^{1/p'} \sin\omega$. *Then we have* $\int_\Omega d^2(\omega, \omega') \,\mathrm{d}\mu\omega \equiv \tilde{c}^{2/p'}$, *giving*

$$\int_{\Omega'} \left| \left(\int_\Omega d^2(\omega, \omega') \,\mathrm{d}\mu\omega \right)^{p'/2} - \tilde{c} \right| \mathrm{d}\mu'\omega' = 0 \ ,$$

while

$$\|d\|_{2,p'} = \left(\int_{\mathbb{R}} \tilde{c} \,\mathrm{d}\mu'\omega' \right)^{1/p'} = \infty \quad \textit{for} \quad 1 \le p' < \infty \ .$$

As there is no other summand of $\tilde{E}$ giving coercivity in d, we can not use Equation (5.8) for unbounded domains and $p \neq \infty$. Also a combination with a second summand limiting $\|d\|_{p'}$ is not possible, as Equation (5.8) forces $\|d\|_{p'}$ to be infinite on unbounded domains Ω'. So the only possibility to use $\tilde{E}_3$ on unbounded sets in question is to choose $p' = \infty$ and perform the minimization in $(W_{1,(2,\infty)}, W_{i,1})$.

The much more important point is to extend the proofs given in Number 2. There are two different problems: In most cases we have to prove the weak lower semicontinuousity of the integral over a positiv function g, this is rather simple:

Lemma 5.27 (weak lower semicontinuity on unbounded sets I)**.** *Let* g *be a positive function and* $I(u) = \int_{\tilde{\Omega}} g(u(\zeta)) \,\mathrm{d}\tilde{\mu}\zeta$ *be weakly lower semicontinuous for all bounded sets* $\tilde{\Omega} \subset \mathbb{R}^n$. *Then is* $I(u)$ *also weakly lower semicontinuous for unbounded* $\bar{\Omega} \subset \mathbb{R}^n$.

Proof. Let $B_s(0)$ be the ball of radius s around 0 and let us define

$$I_s(u) = \int_{\bar{\Omega} \cap B_s(0)} g(u(\zeta)) \,\mathrm{d}\tilde{\mu}\zeta \ ,$$

then $\bar{\Omega} \cap B_s(0)$ is finite and we have

$$\liminf_{\nu\to\infty} I_s(u_\nu) \ge I_s(u) \ .$$

Additionally, as g is positive everywhere, we have for $s_1 < s_2$ also $I_{s_1}(u) \le I_{s_2}(u)$ and

$$\liminf_{\nu\to\infty} I_{s_1}(u_\nu) \le \liminf_{\nu\to\infty} I_{s_2}(u_\nu) \ .$$

If we define $I = I_\infty$ we get

$$I(u) = \lim_{s\to\infty} I_s(u) \leq \lim_{s\to\infty} \liminf_{\nu\to\infty} I_s(u_\nu) \leq \lim_{s\to\infty} \liminf_{\nu\to\infty} I(u_\nu) = \liminf_{\nu\to\infty} I(u_\nu) \ .$$

□

The question is, what happens in the case of Theorem 5.25, where we proved weak continuity of $\int_{\Omega'} x_l d_\omega \, \mathrm{d}\mu'\omega'$ in $(W_{1,p'}, W_{i,p})$ on bounded intervals Ω'. Also here weak continuity is given:

Lemma 5.28 (weak lower semicontinuity on unbounded sets II)**.** *For $x \in W_{i,p}(\Omega')$ and $d \in W_{1,(2,p')}(\Omega')$ we have $\int_{\Omega'} dx \, \mathrm{d}\mu'\omega'$ weak continuous.*

Proof. We know from Theorem 5.25 the result in the case of finite Ω_F, so we have $\lim \int_{\Omega_F} x_n(\omega')d_n(\omega') \, \mathrm{d}\mu'\omega' = \int_{\Omega_F} x(\omega')d(\omega') \, \mathrm{d}\mu'\omega'$. If we define $R_m = (B_m(0)\backslash B_{m-1}(0)) \cap \Omega'$ we get:

$$\sum_{m=1}^{\infty} \lim_{n\to\infty} \int_{R_m} x_n(\omega')d_n(\omega') \, \mathrm{d}\mu'\omega' = \sum_{m=1}^{\infty} \int_{R_m} x(\omega')d(\omega') \, \mathrm{d}\mu'\omega'$$

The result is valid, if we can exchange the limit and the sum in the preceding equation. This holds according to Lebesgue's dominated convergence theorem ((cf. [84, Theorem 5.13])) applied on a discrete measure on $\mathbb{N}$. It is applicable, as $\int_{R_m} x_n(\omega')d_n(\omega') \, \mathrm{d}\mu'\omega' \leq \|d_n\|^2_{p',R_m} \|x_n\|^2_{p,R_m}$ is a summable function, as shown by

$$\begin{aligned} \sum_{m=1}^{\infty} \|d_n\|^2_{p',R_m} \|x_n\|^2_{p,R_m} &\leq \sum_{m=1}^{\infty} \|d_n\|^2_{p',R_m} \left(\sum_{r=1}^{\infty} \|x_n\|^2_{p,R_r} \right) \\ &= \sum_{m=1}^{\infty} \|d_n\|^2_{p',R_m} \|x_n\|^2_{p,\Omega'} = \|d_n\|^2_{p',\Omega'} \|x_n\|^2_{p,\Omega'} \leq C \ . \end{aligned}$$

The last step holds according to Theorem 2.12,1.a. □

So we obtain altogether the applicability of the results from Sections 5.1.2– 5.21 to the case of infinite I, Ω and Ω'.

non-continuous f

Till now we had to assume, that the sparsity function f is continuous, due to the requirement in Theorem 2.23. We want to show in this subsection that this condition is not necessary. The conditions of Definition 5.1 on $\mathcal{F}_1$ are already enough[10], to reach weak lower semicontinuity of E_2 in $W_{i,*}$. First we want to state a result on the continuity properties a sparsity function in $\mathcal{F}_1$ already has, as given in [69], later on we prove the weak lower semicontinuity of E_2 implying this properties.

[10] Elements of $\mathcal{F}_2$ are continuous in any case.

Lemma 5.29 (Continuity of $f \in \mathcal{F}_1$)**.** *A function $f \in \mathcal{F}_1$ according to Definition 5.1 is (strong) continuous for all $\tau \in \mathbb{R}\backslash\{0\}$ and lower semicontinuous on $\mathbb{R}$.*

Proof. In order to prove continuity of f in τ we need

$$\forall\, \epsilon > 0\ \exists\, \delta > 0, \forall x \in \mathbb{R} : |x - \tau| < \delta \Rightarrow |f(x) - f(\tau)| < \epsilon\ .$$

Let us first assume $\tau > 0$. We know from Definition 5.1:

$$x_1 < x_2 \Rightarrow f(x_1) \leq f(x_2) \tag{5.15}$$

and

$$x_1 < x_2 \Rightarrow f(x_1)/x_1 \geq f(x_2)/x_2 \tag{5.16}$$

and we get

$$\begin{array}{rcl} \dfrac{f(\tau+\delta)}{\tau+\delta} & \leq & \dfrac{f(\tau)}{\tau} \\ & \Leftrightarrow & \\ f(\tau+\delta) - f(\tau) & \leq & \delta\dfrac{f(\tau)}{\tau}\ . \end{array}$$

Furthermore $0 \leq f(\tau+\delta) - f(\tau)$ by Equation (5.15), so we get for $x > \tau$:

$$0 \leq f(x) - f(\tau) \leq \epsilon \quad \text{for} \quad \delta \leq \epsilon\frac{\tau}{f(\tau)}\ .$$

By an analogous calculation using $f(\tau-\delta)/(\tau-\delta) \geq f(\tau)/\tau$ we get the analog result for $0 < x < \tau$ and by symmetry of f according to 0 we have continuity of f in $\mathbb{R}\backslash\{0\}$. Even more, according to the symmetry of f we have

$$a = \lim_{x\to 0,\ x>0} f(x) = \lim_{x\to 0,\ x<0} f(x)$$

and lower semicontinuity of f on $\mathbb{R}$ as $0 = f(0) \leq a \leq f(\tau)$ for every τ due to Equation (5.15). □

To generalize the results from the former subsections, we have to prove now the sufficience of an f which is non-continuous in one point. There is just one point, we have to modify: The validity of Theorem 2.23 in the proof of Theorem 5.13. Since the above results on infinite I, Ω and Ω' make no assumptions on the continuity of f, we can restrict the proof to finite sets.

Lemma 5.30 (weak lower semicontinuity for general $f \in \mathcal{F}_1$)**.** *Let $f(x,u,\xi) = f(u)$ be a sparsity function $f \in \mathcal{F}_1$, potentially non-continuous in 0. Furthermore let f satisfy the remaining assumptions of Theorem 2.23 with $l = n$. Then (using the notations of Theorem 2.23)*

$$J(u, \nabla u) = \int_\Omega f(x, u(x), \nabla u(x))\, \mathrm{d}\mu x$$

is weak lower semicontinuous in $W_{1,p}(\Omega)$.

Proof. According to the proof of Lemma 5.29 we can define the continuous function

$$\tilde{f}(\tau) = \begin{cases} a & \tau = 0 \\ f(\tau) & \text{else} \end{cases}$$

$\tilde{f}$ satisfies all requirements of Theorem 2.23, so we know from there:

$$\liminf_{\nu_j \to \infty} \int_\Omega \tilde{f}(x, u_{\nu_j}(x), \nabla u_{\nu_j}(x)) \,\mathrm{d}\mu x \geq \int_\Omega \tilde{f}(x, u(x), \nabla u(x)) \,\mathrm{d}\mu x$$

Additionally we have:

$$\int_\Omega f(x, u_{\nu_j}(x), \nabla u_{\nu_j}(x)) \,\mathrm{d}\mu x = \int_\Omega \tilde{f}(x, u_{\nu_j}(x), \nabla u_{\nu_j}(x)) \,\mathrm{d}\mu x - a \int_\Omega \chi_{\{x|u_{\nu_j}(x)=0\}}(x) \,\mathrm{d}\mu x$$

The characteristic function is measurable according to [84, Satz 6.18]. As Ω is bounded and $u_{\nu_j} \to u$ in L_2 we know that

$$\limsup_{\nu_j \to \infty} \int_\Omega \chi_{\{x|u_{\nu_j}(x)=0\}}(x) \,\mathrm{d}\mu x = \int_\Omega \chi_A(x) \,\mathrm{d}\mu x$$

with $A \subset \{x|u(x) = 0\} + \mathcal{N}$, $\mathcal{N}$ of measure 0. Otherwise there would exist sets $B_i \subset \{x|u(x) \neq 0\}$ with $u_{\nu_{j_i}} = 0$ for μ-almost every $x \in B_i$. Furthermore we have $\int_{B_i} |u(x)| \,\mathrm{d}\mu x \geq b > 0$, this results in $\int_\Omega \left| u_{\nu_{j_i}} - u \right| (x) \,\mathrm{d}\mu x \geq b$ in contradiction to the convergence of u_{ν_j}.

Using this we get:

$$\begin{aligned}
&\liminf_{\nu_j \to \infty} \int_\Omega f(x, u_{\nu_j}(x), \nabla u_{\nu_j}(x)) \,\mathrm{d}\mu x \\
\geq\ & \int_\Omega \tilde{f}(x, u(x), \nabla u(x)) \,\mathrm{d}\mu x - \int_\Omega \chi_A(x) \,\mathrm{d}\mu x \\
=\ & \int_\Omega f(x, u(x), \nabla u(x)) \,\mathrm{d}\mu x \\
& \quad + \int_\Omega \chi_{\{x|u(x)=0\}}(x) \,\mathrm{d}\mu x - \int_\Omega \chi_A(x) \,\mathrm{d}\mu x \\
\geq\ & \int_\Omega f(x, u(x), \nabla u(x)) \,\mathrm{d}\mu x
\end{aligned}$$

and we proved the lemma. □

Altogether we proved the validity of Theorem 5.25 also in the case of infinite I, Ω, Ω' and non-continuous $f \in \mathcal{F}_1$. Thus, we proved the existence of a minimum of the continuous dictionary learning $\tilde{E}(d, x)$ in all interesting frameworks, minimizing both variables simultaneous.

5.2 Gradient methods for learning continuous dictionaries

Since, according to the last section, the existence of at least one minimum of $\tilde{E}$ is clear, the question arises, how to find this minimum or at least a combination of d and x, being locally optimal.

In this section we want to offer a method, constructing a minimizing continuous sequence for

$$E = E_1 + E_2 + E_3 + E_4 + E_5 \tag{5.17}$$

under special circumstances, or at least a decreasing sequence for a wide class of sparsity measures f. The summands of E are defined according to Equations (5.6), (5.7), (5.13) and (5.14).

The algorithm is based on a generalized conditional gradient method, presented in [18]. The conclusions of that article are summarized in 2.3, especially in Lemma 2.26 and Theorem 2.27. As there the results are proven for HILBERT spaces only, we restrict ourselves to the case of $p = p' = 2$.

We resign to generalize the algorithms offered in Subsection 2.2.2 instead of the conditional gradient approach, since these are just approximative, optimized with regard to computing time, but not necessarily optimal in finding the best solution, i.e. here that one with minimal E. The non-optimality of the algorithms can easily be recognized, as they all do not optimize all elements of the dictionary together with the coefficients (cf. the easy two dimensional example, stated in the beginning of Subsection 5.1.4). Additionally, several of this algorithms, like K-SVD or MAP (cf. Subsection 2.2.2), use, as there is no choice due to computation time, a standard sparse coding algorithm (cf. Subsection 2.2.1), which is not optimal, beside in the case of basis pursuit, if we choose in $\tilde{E}$ an f of the form $f = |\cdot|$, or in case of $f \in \mathcal{F}_2$.

Our method is far from being practicable for applicational scale problems, according to its high computation time, due to the slow convergence of gradient algorithms. But in contrast to the mentioned other methods (see Subsection 2.2.2 and above) we obtain in this section conditions for convergence to a local minimum in the continuous case, which are not clear elsewise. Additionally, for most methods in Subsection 2.2.2 we have no results about the characterization of stationary points. So one step of the following algorithm can help to identify their locally non-optimal stationary points and to obtain better results.

5.2.1 Conditional gradient in the case of $f \in \mathcal{F}_2$

For applying the generalized conditional gradient method we have to separate E according to Algorithm 2.24 into two summands F and Φ, F being differentiable, Φ being convex. Using the notations from Section 5.1, we know that E_1 is not convex in $(L_2(\Omega, \Omega'), L_2(\Omega, I))$ (cf. Example 5.21), as well as E_3 in $L_2(\Omega, \Omega')$ (via Example 5.18 we see that E_3 is not weak lower semicontinuous, but strong). Nevertheless, both are FRÉCHET differentiable, as the two following lemmata show:

Lemma 5.31 (FRÉCHET differentiability of E_1)**.** *Let* $E_1 : (L_2(\Omega\times\Omega')\times L_2(I\times\Omega')) \to \mathbb{R}$ *with*

$$E_1(d,x) = \int_I \int_\Omega \left(y(\omega,l) - \int_{\Omega'} d(\omega,\omega')x(l,\omega')\,\mathrm{d}\mu'\omega' \right)^2 \mathrm{d}\mu\omega\ \mathrm{d}\nu l\ .$$

Then E_1 *is* FRÉCHET *differentiable and the derivative equals*

$$\begin{aligned} E_1'(d,x)(h,k) &= 2\int_I \int_\Omega \left(\int_{\Omega'} d(\omega,\omega')x(l,\omega')\,\mathrm{d}\mu'\omega' - y(\omega,l) \right) \\ &\quad \cdot \left(\int_{\Omega'} (d(\omega,\omega')k(l,\omega') + h(\omega,\omega')x(l,\omega'))\,\mathrm{d}\mu'\omega' \right) \mathrm{d}\mu\omega\ \mathrm{d}\nu l\ . \end{aligned} \tag{5.18}$$

Furthermore, $E_1'(d,x)$ *is uniformly continuous on bounded sets.*

Proof. For the sake of readability we omit the integration variables.

We have to prove

$$\lim_{h\to 0} \frac{|E_1(d+h,x+k) - E_1(d,x) - E_1'(d,x)(h,k)|}{\sqrt{\|h\|_2^2 + \|k\|_2^2}} = 0$$

According to the definition of E_1 we have

$$\begin{aligned} E_1(d+h,x+k) - E_1(d,x) &= \iint \Bigg(-2y \left(\int (dk + hx + hk)\,\mathrm{d}\mu'\omega' \right) \\ &\quad + \left(2\int dx\,\mathrm{d}\mu'\omega' \right) \left(\int (hx + dk + hk)\,\mathrm{d}\mu'\omega' \right) \\ &\quad + \left(\int (hx + dk + hk)\,\mathrm{d}\mu'\omega' \right)^2 \Bigg) \mathrm{d}\mu\omega\ \mathrm{d}\nu l\ . \end{aligned}$$

Taking into account the given assumption of E_1' in Equation (5.18), it remains to compute

$$\begin{aligned} a &= \lim_{(h,k)\to 0} \frac{1}{\sqrt{\|h\|_2^2 + \|k\|_2^2}} \Bigg| \iint \Bigg(\left(-2y + 2\int dx\,\mathrm{d}\mu'\omega' \right) \left(\int hk\,\mathrm{d}\mu'\omega' \right) \\ &\qquad + \left(\int (hx + dk + hk)\,\mathrm{d}\mu'\omega' \right)^2 \Bigg) \mathrm{d}\mu\omega\ \mathrm{d}\nu l \Bigg| \\ &\leq \lim_{(h,k)\to 0} \Bigg(\frac{2\| \int hk\,\mathrm{d}\mu'\omega' \|_2 (\| \int dx\,\mathrm{d}\mu'\omega' - y\|_2 + \| \int (hx+dk)\,\mathrm{d}\mu'\omega' \|)}{\sqrt{\|h\|_2^2 + \|k\|_2^2}} \\ &\qquad + \frac{\| \int hk\,\mathrm{d}\mu'\omega' \|_2^2 + \| \int (hx+dk)\,\mathrm{d}\mu'\omega' \|_2^2}{\sqrt{\|h\|_2^2 + \|k\|_2^2}} \Bigg)\ . \end{aligned}$$

We know, the second factor of the first summand is finite. Since $\| \int hk\,\mathrm{d}\mu'\omega' \|_2 \leq \max\{\|h\|_2, \|k\|_2\}^2$ and $\sqrt{\|h\|_2^2 + \|k\|_2^2} \geq \max\{\|h\|_2, \|k\|_2\}$ we know, that the first summand of the above inequality converges to zero:

$$a \leq \lim_{(h,k)\to 0} \frac{\| \int (hx+dk)\,\mathrm{d}\mu'\omega' \|_2^2}{\sqrt{\|h\|_2^2 + \|k\|_2^2}}$$

Using HÖLDER's inequality we can transfer the numerator into a sum of norms and obtain

$$a \leq \lim_{(h,k)\to 0} \left(\frac{\|h\|_2^2 \cdot \|x\|_2^2 + 2\|h\|_2 \cdot \|x\|_2 \cdot \|d\|_2 \cdot \|k\|_2}{\|h\|_2} + \frac{\|d\|_2^2 \cdot \|k\|_2^2}{\|k\|_2} \right) = 0 \ .$$

For being a FRÉCHET derivative, $E_1'(d,x)$ additionally has to be linear in (h,k), this is given by the linearity of the integral and as the first factor in 5.18 is independent of (h,k).

To prove the uniform continuity of $E_1'(d,x)$ on bounded sets, we separate it into the two summands,

$$\begin{aligned} A_1(d,x)(h,k) &= \iint \left(\int dx \, \mathrm{d}\mu'\omega' \right) \left(\int dk + hx \, \mathrm{d}\mu'\omega' \right) \mathrm{d}\mu\omega \ \mathrm{d}\nu l \\ A_2(d,x)(h,k) &= \iint y \left(\int dk + hx \, \mathrm{d}\mu'\omega' \right) \mathrm{d}\mu\omega \ \mathrm{d}\nu l \end{aligned}$$

and prove the assumption separately for both summands. Let us fix an $\epsilon > 0$, so we have with $\delta = \sqrt{\|d_1 - d_2\|_2^2 + \|x_1 - x_2\|_2^2}$ via HÖLDER's inequality

$$\begin{aligned} &\|A_2(d_1,x_1) - A_2(d_2,x_2)\| \\ &= \sup_{\|(h,k)\|_2=1} \left| \iint y \left(\int (d_1 k + hx_1 - d_2 k - hx_2) \, \mathrm{d}\mu'\omega' \right) \mathrm{d}\mu\omega \ \mathrm{d}\nu l \right| \\ &\leq \sup_{\|(h,k)\|_2=1} \|y\|_2 \left(\iint \left| \int ((d_1 - d_2)k + (x_1 - x_2)h) \, \mathrm{d}\mu'\omega' \right|^2 \mathrm{d}\mu\omega \ \mathrm{d}\nu l \right)^{1/2} \\ &\leq \sup_{\|(h,k)\|_2=1} \|y\|_2 \cdot (\|d_1 - d_2\|_2^2 \cdot \|k\|_2^2 + \|x_1 - x_2\|_2^2 \cdot \|h\|_2^2 \\ &\qquad\qquad +2\|d_1 - d_2\|_2 \cdot \|x_1 - x_2\|_2 \cdot \|h\|_2 \cdot \|k\|_2)^{1/2} \\ &\leq \sqrt{3}\|y\|_2 \delta \overset{!}{\leq} \epsilon \ . \end{aligned}$$

The last step uses $\|h\|_2, \|k\|_2 \leq 1$ and $\|d_1 - d_2\|_2 \|x_1 - x_2\|_2 \leq \delta^2$. So we have the uniform continuity of A_2.
In the proof for A_1 we do a further separation of the difference:

$$\begin{aligned} &\|A_1(d_1,x_1) - A_1(d_2,x_2)\| \\ &= \sup_{\|(h,k)\|_2=1} \left| \iint \left(\left(\int d_1 x_1 \, \mathrm{d}\mu'\omega' \right) \left(\int (d_1 k + hx_1) \, \mathrm{d}\mu'\omega' \right) \right. \right. \\ &\qquad \left. \left. - \left(\int d_2 x_2 \, \mathrm{d}\mu'\omega' \right) \left(\int (d_2 k + hx_2) \, \mathrm{d}\mu'\omega' \right) \right) \mathrm{d}\mu\omega \ \mathrm{d}\nu l \right| \\ &\leq \sup_{\|(h,k)\|_2=1} \left| \iint \left(\int d_1 x_1 \, \mathrm{d}\mu'\omega' \right) \left(\int (d_1 k + hx_1 - d_2 k - hx_2) \, \mathrm{d}\mu'\omega' \right) \mathrm{d}\mu\omega \ \mathrm{d}\nu l \right| \\ &\qquad + \sup_{\|(h,k)\|_2=1} \left| \iint \left(\int (d_1 x_1 - d_2 x_2) \, \mathrm{d}\mu'\omega' \right) \left(\int (d_2 k + hx_2) \, \mathrm{d}\mu'\omega' \right) \mathrm{d}\mu\omega \ \mathrm{d}\nu l \right| \\ &=: \sup_{\|(h,k)\|_2=1} A_{11} + \sup_{\|(h,k)\|_2=1} A_{12} \end{aligned}$$

For A_{11} we use the same estimates as in the case of A_2, so we get directly

$$\sup_{\|(h,k)\|_2=1} A_{11} \leq \sqrt{3}\|d_1\|_2\|x_1\|_2\delta \ .$$

In the case of A_{12} the situation is different, but not more complicate, we have

$$\begin{aligned}
\sup_{\|(h,k)\|_2=1} A_{12} &\leq \left\| \int (d_1x_1 - d_2x_2)\,\mathrm{d}\mu'\omega' \right\|_2 \cdot \sup_{\|(h,k)\|_2=1} \left\| \int (d_2k + hx_2)\,\mathrm{d}\mu'\omega' \right\|_2 \\
&\leq \sup_{\|(h,k)\|_2=1} \left\| \int (d_1(x_1 - x_2) + x_2(d_1 - d_2))\,\mathrm{d}\mu'\omega' \right\|_2 \\
&\qquad \cdot \left(\left\| \int d_2k\,\mathrm{d}\mu'\omega' \right\|_2 + \left\| \int hx_2\,\mathrm{d}\mu'\omega' \right\|_2 \right) \\
&\leq \sup_{\|(h,k)\|_2=1} (\|d_1\|_2^2\|x_1 - x_2\|_2^2 + \|x_2\|_2^2\|d_1 - d_2\|_2^2 \\
&\qquad +2(\|d_1\|_2\|x_2\|_2\|x_1 - x_2\|_2\|d_1 - d_2\|_2))^{1/2} \cdot (\|d_2\|_2\|k\|_2 + \|h\|_2\|x_2\|_2) \\
&\leq (\|d_1\|_2^2 + \|x_2\|_2^2 + 2\|d_1\|_2\|x_2\|_2)^{1/2}(\|x_1 - x_2\|_2^2 + \|d_1 - d_2\|_2^2)^{1/2}(\|d_2\|_2 + \|x_2\|_2) \\
&\leq (\|d_2\|_2 + \|x_2\|_2)(\|d_1\|_2 + \|x_2\|) \cdot \delta
\end{aligned}$$

So altogether we have with $d_1, x_1, d_2, x_2 \in B_R(0)$

$$\begin{aligned}
\|A_1(d_1, x_1) - A_1(d_2, x_2)\| &\leq (\sqrt{3}\|d_1\|_2 \cdot \|x_1\|_2 + (\|d_2\|_2 + \|x_2\|_2)(\|d_1\|_2 + \|x_2\|_2)) \cdot \delta \\
&\leq (4 + \sqrt{3})R^2\delta \overset{!}{\leq} \epsilon
\end{aligned}$$

and so the uniform continuity on bounded sets. □

Lemma 5.32 (FRÉCHET differentiability of E_3)**.** *Given $E_3 : L_2(\Omega \times \Omega') \to \mathbb{R}$ with*

$$E_3(d) = \left(\left(\int_\Omega \int_{\Omega'} d(\omega, \omega')^2\,\mathrm{d}\mu\omega\ \mathrm{d}\mu'\omega' \right) - c \right)^2$$

then we know, E_3 is on bounded sets uniform continuous FRÉCHET *differentiable with*

$$\begin{aligned}
E_3'(d)(h) &= 4 \left(\left(\int_\Omega \int_{\Omega'} d(\omega, \omega')^2\,\mathrm{d}\mu\omega\ \mathrm{d}\mu'\omega' \right) - c \right) \\
&\qquad \cdot \left(\int_\Omega \int_{\Omega'} h(\omega, \omega')d(\omega, \omega')\,\mathrm{d}\mu\omega\ \mathrm{d}\mu'\omega' \right) \ .
\end{aligned} \tag{5.19}$$

Proof. Again we omit the integration variables during the proof.

The FRÉCHET derivative of $E_3(d)$ in (5.19) can simply be computed by using the chain rule: $f : \mathbb{R} \to \mathbb{R}, f(x) = (x-1)^2$ is simply derivable, as well as $A : L_2(\Omega \times \Omega') \to \mathbb{R}$, $A(d) = \iint d^2\,\mathrm{d}\mu\omega\ \mathrm{d}\mu'\omega'$. It follows

$$E_3'(d)(h) = f'(A(d)) \cdot A'(d)(h) = 2 \left(\iint d^2\,\mathrm{d}\mu\omega\ \mathrm{d}\mu'\omega' - c \right) \cdot \left(2 \iint hd\,\mathrm{d}\mu\omega\ \mathrm{d}\mu'\omega' \right) \ .$$

For proving the required continuity property of $E_3'(d)$, we separate it like in the proof above for simplicity into two parts:

$$E_3'(d)(h) = 4\left(\|d\|_2^2 \iint hd \,\mathrm{d}\mu\omega \;\mathrm{d}\mu'\omega' - c \iint hd \,\mathrm{d}\mu\omega \;\mathrm{d}\mu'\omega'\right) = 4(A_1 - A_2)$$

The uniform continuity of A_2 is directly given by fixing $\epsilon > 0$ and

$$\|A_2(d_1) - A_2(d_2)\| \leq \sup_{\|h\|_2=1} \|h\|_2 c \|d_1 - d_2\|_2 \leq c\delta$$

for $\delta = \|d_1 - d_2\|_2$. In the case of A_1 we obtain:

$$\begin{aligned}
\|A_1(d_1) - A_1(d_2)\| &= \sup_{\|h\|_2=1} \left| \|d_1\|_2^2 \iint hd_1 \,\mathrm{d}\mu\omega \;\mathrm{d}\mu'\omega' - \|d_2\| \iint hd_2 \,\mathrm{d}\mu\omega \;\mathrm{d}\mu'\omega' \right| \\
&\leq \sup_{\|h\|_2=1} \left| \|d_1\|_2^2 \iint (hd_1 - hd_2) \,\mathrm{d}\mu\omega \;\mathrm{d}\mu'\omega' \right| \\
&\qquad + \left| (\|d_1\|_2 - \|d_2\|_2) \iint hd_2 \,\mathrm{d}\mu\omega \;\mathrm{d}\mu'\omega' \right| \\
&\leq \sup_{\|h\|_2=1} \|d_1\|_2^2 \cdot \|h\|_2 \delta + \|h\|_2 \cdot \|d_2\|_2 \left| \iint (d_1^2 - d_2^2) \,\mathrm{d}\mu\omega \;\mathrm{d}\mu'\omega' \right| \\
&\leq \|d_1\|_2^2 \delta + \|d_2\|_2 \cdot \|d_1 - d_2\|_2 \cdot \|d_1 + d_2\|_2 \\
&\leq \delta(\|d_1\|_2^2 + \|d_2\|_2^2 + \|d_1\|_2 \cdot \|d_2\|_2)
\end{aligned}$$

So we have also A_1 to be uniformly continuous on bounded sets. □

Thus for allying Algorithm 2.24 we have E_1 and E_3 as summands of F[11]. The several alternative forms of of E_4 and E_5 according to Equation (5.13) or rather (5.14) are not all FRÉCHET differentiable (e.g., an E_4 defined in Equation (5.13) with g_2 being zero on a finite, convex set and infinite outside), but they are all convex, cf. the proofs of Theorem 5.13 and Lemma 5.19. So for reasons of simplicity we let them in any case to be summands of Φ. Problematic is the classification of the sparsity measure E_2. In the case of a sparsity function $f \in \mathcal{F}_2$ (cf. Definition 5.1) we have an E_2 which is convex per definition and FRÉCHET differentiable, as

$$E_2'(x)(k) = \left(\int_I \int_\Omega f(x) \,\mathrm{d}\nu l \;\mathrm{d}\mu\omega\right)'(k) = \int_I \int_\Omega f'(x)k \,\mathrm{d}\nu l \;\mathrm{d}\mu\omega \tag{5.20}$$

and f' exists in the case of $f \in \mathcal{F}_2$. But the following example shows, E_2' is not necessarily uniform continuous:

Example 5.33. *Let $f(x) = 2/3x^{3/2} \in \mathcal{F}_2$. Then $E_2'(x)(k) = \iint \sqrt{x}k \,\mathrm{d}\nu l \;\mathrm{d}\mu\omega$ is not uniform continuous, as for*

$$x_1(\omega, l) = \begin{cases} 1 & (\omega, l) \in (-1, 1)^2 \\ \omega^{-0.6} & \omega \notin (-1, 1), l \in (-1, 1) \\ 0 & \textit{otherwise} \end{cases}$$

[11] This would also be valid using $\tilde{E}_3$ from Equation (5.8) in the case of finite Ω, Ω', I.

and

$$x_2(\omega) = \begin{cases} x_1(\omega, l) & \omega \in (-a, a) \\ 0 & \textit{otherwise} \end{cases}$$

we have $\|x_1 - x_2\|_2 = \sqrt{2\int_a^\infty x^{-1.2}} \leq \delta_a$*, for an appropriate* δ_a*, but defining* $b = 1/\|x_1\|_2$ *we obtain:*

$$\begin{aligned} & \left\| \iint \sqrt{x_1} k \, \mathrm{d}\nu l \, \mathrm{d}\mu\omega - \iint \sqrt{x_2} k \, \mathrm{d}\nu l \, \mathrm{d}\mu\omega \right\| \\ = & \sup_{\|k\|_2=1} \left| \int \left(\sqrt{x_1(\omega)} - \sqrt{x_2(\omega)} \right) k \, \mathrm{d}\mu\omega \right| \\ \geq & \; b \int \left(\sqrt{x_1(\omega)} - \sqrt{x_2(\omega)} \right) x_1(\omega) \, \mathrm{d}\mu\omega \\ = & \; 2b \int_a^\infty \omega^{-0.9} \, \mathrm{d}\mu\omega = \infty \quad \forall a \in \mathbb{R}^+ \end{aligned}$$

So for full usage of Theorem 2.27 E_2 has to be a summand of Φ, but according to Lemma 2.26 there are also some positive results for E_2 being summand of F. All in all we have the following alternatives:

Theorem 5.34 (Generalized conditional gradient method for $f \in \mathcal{F}_2$)**.** *Let* $E(d, x)$ *be defined according to Equation* (5.14) *and* $f \in \mathcal{F}_2$ *be a sparsity function of Type 2. Then exists a indicator function*

$$I_U(d, x) = \begin{cases} 0, & x \in U \\ \infty, & \textit{otherwise} \end{cases} \tag{5.21}$$

with

$$U = \{(d, x) |\ \|(d, x)\|_2 \leq C\} \tag{5.22}$$

and a constant $\hat{C}$*, such that with* $\|(d_0, x_0)\|_2 < \hat{C}$*:*

1. *If* $F = E_1 + E_3$ *and* $\Phi = E_2 + E_4 + E_5 + I_U$*, the sequence* (d_n, x_n) *generated by the generalized conditional gradient algorithm satisfies* $E(d_n, x_n) > E(d_{n+1}, x_{n+1})$ *for all* $n \in \mathbb{N}$*. If there is a convergent subsequence of* (d_n, x_n) *it converges to a stationary point* $(d, x) \in (W_{1,2}, W_{i,2})$ *of Algorithm 2.24.*
2. *If* $F = E_1 + E_2 + E_3$ *and* $\Phi = E_4 + E_5 + I_U$*, then the generalized conditional gradient algorithm generates a sequence* $(d_n, x_n) \in (W_{1,2}, W_{i,2})$ *with* $E(d_n, x_n) > E(d_{n+1}, x_{n+1})$ *for all* $n \in \mathbb{N}$*.*

Proof. According to Lemmata 5.31 and 5.32 we have E_1 and E_3 to be FRÉCHET differentiable and consequential also GÂTEAUX differentiable. The later is also true for E_2 according to Equation (5.20). So we have for both cases of the theorem a GÂTEAUX differentiable F.

Additionally Φ satisfies the conditions of 2.25, since U is chosen appropriately: Clearly $(0, 0) \in U$, thus Φ is zeros for $(d, x) \equiv (0, 0)$ (see Definition 5.1 and Equations (5.13), (5.14)). Also the convexity of E_4 and E_5 is already given in the proofs

of Theorem 5.13 and Lemma 5.19 and the convexity of E_2 is clear, as $f \in \mathcal{F}_2$ is a convex function by definition and the integral is linear. Thus the convex definition of U results in convexity of Φ in both cases.
For proving the strong coercivity of Φ in $(W_{1,2}, W_{i,2})$ we know, that $\Phi = \infty$ for $\|(d,x)\|_2 > C$. Thus, according to the definition of of E_4 and E_5, based on strong coercive g_2 and g_3, we obtain analog to 5.23 strong coercivity of Φ in $(W_{1,2}, W_{i,2})$. As to the lower semicontinuity of Φ we already proved weak lower semicontinuity of E_4 and E_5 in the proofs of Theorem 5.13 and Lemma 5.19, this implies (strong) lower semicontinuity. Since U is defined as closed set in Equation (5.22), I_U and following Φ is strong lower semicontinuous.

So applying Lemma 2.26 we have with $E = F + \Phi$:

$$E(d_{n+1}, x_{n+1}) + I_U((d_{n+1}, x_{n+1}) \leq E(d_n, x_n) + I_U((d_n, x_n)$$

Adding I_U to E has no influence on the resulting sequence (d_n, x_n), if we choose C in Equation (5.22) high enough: According to Remark 5.23 we know, that $E(d,x) \geq \tilde{C}\|(d,x)\|_{i,*}$ for $\|(d,x)\|_{i,*} > \bar{C}$, furthermore, $E(0,0) = \hat{C} = \|y\|_2^2 + c$. Thus $(d_n, x_n) \in U$ for all $n \in \mathbb{N}$, if $E(d_0, x_0) \leq \hat{C}$ and $C \geq \max\{\bar{C}, \hat{C}\tilde{C}\}$.

For proving the additional result of Number 1, we need to apply Theorem 2.27. Additionally to the properties used above, we know from Lemmata 5.31 and 5.32 about the uniformly continuous FRÉCHET differentiability of F on bounded sets. Furthermore $E = F + \Phi$ is weak coercive, as proven in Remark 5.23. So by applying Theorem 2.27 we have with Ψ as defined in Equation (2.20): $\lim_{n\to\infty} \Psi(d_n, x_n) = 0$. As Ψ is lower semicontinuous we have for a convergent subsequence $(d_{n_k}, x_{n_k}) \to (d,x)$ also $\Psi(d,x) = 0$, thus (d,x) is a stationary point of the algorithm. □

Remark 5.35.

- *We can not force the existence of the strong convergent subsequence in Theorem 5.34 in the general case, since there is just weak coercivity of $F + \Phi$ given. This heals generally if*
  - *Φ satisfies a condition similar to the strong sequential coercivity (cf. Definition 2.16 i.e.*

    $$E_t = \{(d,x) | \Phi(d,x) \leq t\} \text{ is compact } \forall t \in \mathbb{R} \ .$$

  - *Ψ (cf. Equation (2.20)) is weak lower semicontinuous, then the existing weak limit of a subsequence $(d_{n_k}, x_{n_k}) \to (d,x)$ satisfies $\Psi(d,x) = 0$.*

  Unfortunately both is not given here. Function spaces are in general just connected to weak compactness, and for the weak lower semicontinuity of Ψ complete continuity of F' is required (cf. Theorem 2.27). Observing $E_1'(d_n, x_n)(h,k)$ (cf. Equation (5.18)), where the sequences are orthogonal bases, shows, this is not achievable. Thus, there is just a weakly convergent subsequence $(d_{n_k}, x_{n_k}) \rightharpoonup (d,x)$, but the limit is not necessarily a stationary point of Algorithm 2.24.

- *Interestingly the second result of Theorem 5.34 is also valid, if we choose Φ as a function of $\|x\|_2$ and $\|d\|_2$ satisfying Condition 2.25 without any use of the norms of the derivatives $\|\nabla x\|_2$ and $\|\nabla d\|_2$. Then we obtain a sequence*

$(d_n, x_n)_{n\in\mathbb{N}} \subset (L_2, L_2)$ with $E(d_{n+1}, x_{n+1}) \leq E(d_n, x_n)$ for all n. But we do not get any information about the convergence of the sequence. Even if there exists a clusterpoint (d, x), there is no need to have $E(d, x) \leq E(d_n, x_n)$ for any n. This holds, since there is no weak lower semicontinuity of $E = F + \Phi$ in this case.

- *In case, E_2 is summand of Φ or $g_1 \not\equiv 0$, we can weaken the constraints on U and just need an indicator function in direction of d.*
- *In the case of an application with a discretized version there would not exist a difference between weak and strong lower semicontinuity. So the first point of this remark is always satisfied and there exists a convergent subsequence of (d_n, x_n). Furthermore we can apply the second remark, since weak lower semicontinuity is satisfied.*

5.2.2 Conditional gradient in the case of $f \in \mathcal{F}_1$

If there is just a sparsity measure $f \in \mathcal{F}_1$ (cf. Definition 5.1), the situation for applying Algorithm 2.24 becomes more complicate. Now E_2 is no longer a convex function, and even more the derivative of f is not existing in 0, as we see in the following lemma and remark.

Lemma 5.36. *Let $\bar{f} \in \overline{\mathcal{F}}_1$ be differentiable on $\mathbb{R}^+$. Then $\bar{f}'$ is bounded by*

$$0 \leq \bar{f}'(t) \leq \frac{\bar{f}(t)}{t} \leq b := \lim_{\tau\to 0} \frac{\bar{f}(\tau)}{\tau} \ .$$

Proof. $\bar{f}$ is non-decreasing, so $\bar{f}' > 0$. According to Definition 5.1 we know, the function $t \mapsto \frac{\bar{f}(t)}{t}$ is non-increasing on $(0, \infty)$. So we know:

$$\begin{aligned} \frac{\bar{f}(t+s)}{t+s} - \frac{\bar{f}(t)}{t} &\leq 0 \\ \Leftrightarrow \\ t(\bar{f}(t+s) - \bar{f}(t)) - s\bar{f}(t) &\leq 0 \\ \Leftrightarrow \\ \frac{\bar{f}(t+s) - \bar{f}(t)}{s} &\leq \frac{\bar{f}(t)}{t} \end{aligned}$$

Thus the lemma follows by applying the limit $s \to 0$. □

Remark 5.37.

- *As $t \mapsto \frac{\bar{f}(t)}{t}$ is non-increasing on $(0, \infty)$, the limit b exists in $(0, \infty]$. It equals the right side derivative of $\bar{f}$ in 0.*

  According to the symmetry of $f \in \mathcal{F}_1$ defined via Equation (5.3), the left side derivative of f in 0 equals $-b$, so a derivative in zero does not exist, since $b \neq 0$.

- *According to [84, Theorem 14.5] we know, a monotone function (like $\bar{f}$) is almost everywhere differentiable, so claiming differentiability for $\bar{f}$ is not as*

big a constraint as for many other functions, but still we do not find in this context results about a big class of sparsity functions like that one in Example 5.2,6.

Based on this result it seems to be clear, that also the FRÉCHET derivative of E_2 does not exist, as depicted by the following counter-example:

Example 5.38. *Let $x, h \in L_2(\Omega' \times I)$. Additionally let C be a open finite subset of $\Omega' \times I$, $x \equiv 0$ on C and $\operatorname{supp} h \subset C$. So we have*

$$E_2(x+h) - E_2(x) = \int_{\Omega'} \int_I f((x+h)(\omega', l)) - f(x(\omega', l)) \, \mathrm{d}\mu' \omega' \, \mathrm{d}\nu l \ .$$

While for elements of C we have $x + h = h$, there is on $(\Omega' \times I) \backslash C$: $x + h = x$. So

$$E_2(x+h) - E_2(x) = \int_{\Omega'} \int_I f(h) \, \mathrm{d}\mu' \omega' \, \mathrm{d}\nu l \ = E_2(h) \ .$$

If we propose the existence of a FRÉCHET *derivative of $E_2(x)$ denoted by $A(x)(h)$, then:*

$$\frac{\|E_2(x+h) - E_2(x) - A(x)(h)\|}{\|h\|} = \frac{\|E_2(h) - A(x)(h)\|}{\|h\|} \xrightarrow{!} 0$$

In the case $\bar{h} = -h$ we get analogous $E_2(\bar{h}) = E_2(h)$ and $A(x)(\bar{h}) = -A(x)(h)$ resulting in

$$\frac{\|E_2(h) + A(x)(h)\|}{\|h\|} \xrightarrow{!} 0 \ .$$

So we have in sum:

$$\begin{aligned} &\frac{\|E_2(h) - A(x)(h)\| + \|E_2(h) + A(x)(h)\|}{\|h\|} \\ \geq \ &\frac{\|E_2(h) - A(x)(h) + E_2(h) + A(x)(h)\|}{\|h\|} = \frac{2\|E_2(h)\|}{\|h\|} \xrightarrow{!} 0 \end{aligned}$$

This is impossible for some special h, choose, e.g., $h_n \equiv 1/n$ on C, giving

$$\frac{f(1/n)}{1/n} \xrightarrow{!} 0,$$

while $t \mapsto f(t)/t$ is maximal in 0 according to Definition 5.1.

Taking this results into acount, there are two alternatives:

1. Changing E_2 or rather f in a way, that it satisfies the conditions of differentiability.
2. Proving a generalization of the generalized conditional gradient algorithm, applicable for E_2 or rather the whole E.

In the following we analyze both possibilities.

Surrogate for $f \in \mathcal{F}_1$

We saw in the previous lemma and example, a sparsity function $f \in \mathcal{F}_1$ does not result in a differentiable E_2. Since there are good reasons, as described beneath Example 5.2, to define a sparsity measure like this, the question arises, how to justify a change of the function in context of application: Since a sparsity function in $\mathcal{F}_1$ is continuous outside 0, it punishes small differences of values, based on rounding errors, just with a small difference in E_2. Unless there is a small difference of 0, here rounding errors would result in great differences of E_2. Even if f is continuous in 0, the absolute value of the derivative $|f'(t)|$ (if existing) will increase for $t \to 0$. As E_2 should increase the sparsity of the coefficient function x, the measure of $\{x(\omega', l) = 0 \mid (\omega', l) \in \Omega' \times \ I\}$ or rather (due to rounding errors) $\{|x(\omega', l)| \leq \epsilon \mid (\omega', l) \in \Omega' \times I\}$ are a big percentage of the measure of $\Omega' \times I$. Furthermore, there is no reason, why rounding errors should not appear in the neighborhood of 0, unless we are using (O)MP in a discrete framework (cf. Subsection 2.2.1). So it seems sensible, to change f in a neighborhood of 0, accoring to the following definition.

Definition 5.39 (Surrogate for $f \in \mathcal{F}_1$)**.** *Let $\tilde{\mathcal{F}}_1 \subset \mathcal{F}_1$ be the subset of sparsity measures according to Definition 5.1 that are continuously differentiable outside 0, let $f \in \tilde{\mathcal{F}}_1$ and let $\delta > 0$. Then we define the set of functions $g \in \mathcal{G}_{f,\delta}$ having the following properties:*

- *g is symmetric,*
- *g is continuously differentiable,*
- *g is non-decreasing on $[0, \infty)$,*
- *$g(0) = g'(0) = 0$,*
- *$g(x) = f(x) \ \forall \ |x| \geq \delta$.*

We denote $\mathcal{G}_\delta = \bigcup_{f \in \tilde{\mathcal{F}}_1} \mathcal{G}_{f,\delta}$

Example 5.40. *As an example for a $g \in \mathcal{G}_{f,\delta}$ let us define (the other parts of g are given by Definition 5.39)*

$$g_{|[0,\delta]}(t) = \int_0^t h_1(\tau)\,\mathrm{d}\tau \ + \alpha h_2(t)$$

with

$$\begin{aligned} h_1(0) = 0, \ h_1(\delta) = f'(\delta) \\ h_2'(0) = h_2(0) = 0, \ h_2'(\delta) = 0, \ h_2(\delta) = 1 \\ \alpha = f(\delta) - \int_0^\delta h_1(\tau)\,\mathrm{d}\tau \ . \end{aligned}$$

We choose h_1 as linear function and h_2 trigonometric:

$$h_1(t) = t\frac{f'(\delta)}{\delta} \ , \quad h_2(t) = \frac{1}{2}\left(1 - \cos\left(\frac{\pi t}{\delta}\right)\right)$$

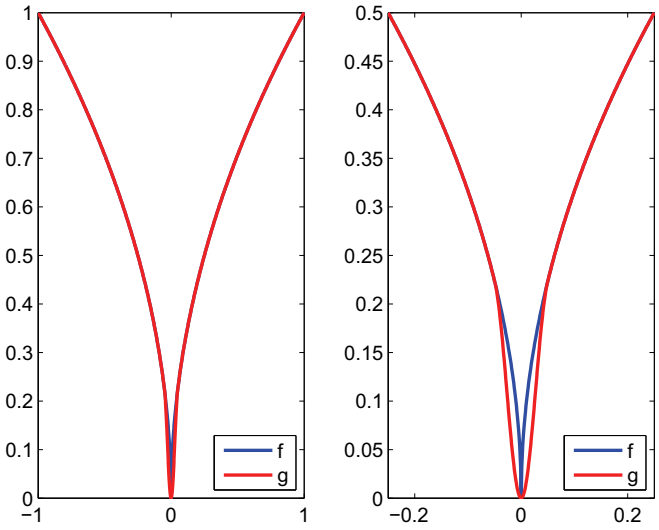


Figure 5.1: Example of f and g according to Example 5.40 in two different magnifications

In Figure 5.1 we gave an example for $f(t) = \sqrt{t}$ *and* $\delta = 0.05$*, resulting in*

$$g_{|[0,\delta]}(t) = 100\sqrt{0.05}\,t^2 + \frac{3\sqrt{0.05}}{8}(1 - \cos(20\pi t)) \ .$$

Using a $g \in \mathcal{G}_\delta$ instead of $f \in \mathcal{F}$ gives an E_2 being uniform continuous Fréchet differentiable:

Lemma 5.41. *Let* $\delta > 0$ *and* $g \in \mathcal{G}_\delta$ *be a function with* Lipschitz *continuous derivative* g'*. Then*

$$E_{2,g}(x) = \int_{\Omega'} \int_I g(x(\omega', l)) \,\mathrm{d}\mu'\omega' \,\mathrm{d}\nu l$$

is uniform continuously Fréchet *differentiable.*

Proof. We know (cf. [171, Page 192]) $E_{2,g}$ is Fréchet derivable, if it is Gâteaux derivable and the derivative is continuous in x. Furthermore we know, the Gâteaux derivative equals

$$\begin{aligned} E'_{2,g}(x)(h) &= \lim_{\tau \to 0} \frac{1}{\tau} \int_{\Omega'} \int_I (g(x(\omega', l) - \tau h(\omega', l)) - g(x(\omega', l))) \,\mathrm{d}\mu'\omega' \,\mathrm{d}\nu l \\ &= \lim_{\tau \to 0} \frac{1}{\tau} \int_{\Omega'} \int_I \int_0^1 \frac{\mathrm{d}}{\mathrm{d}s} g(x(\omega', l) - s\tau h(\omega', l)) \,\mathrm{d}s \,\mathrm{d}\mu'\omega' \,\mathrm{d}\nu l \\ &= \int_{\Omega'} \int_I g'(x(\omega', l)) h(\omega', l) \,\mathrm{d}\mu'\omega' \,\mathrm{d}\nu l \ . \end{aligned}$$

If we now assume $x_1, x_2 \in L_2(\Omega' \times I)$, $\|x_1 - x_2\|_2 < \delta$, we have

$$\begin{aligned}
\|E'(x_1) - E'(x_2)\| &= \sup_{\|h\|_2=1} \Big| \int_{\Omega'} \int_I h(\omega', l)(g'(x_1(\omega', l)) \\
&\qquad\qquad -g'(x_2(\omega', l))) \,\mathrm{d}\mu'\omega' \,\mathrm{d}\nu l \Big| \\
&\leq \int_{\Omega'} \int_I \left| g'(x_1(\omega', l)) - g'(x_2(\omega', l)) \right|^2 \mathrm{d}\mu'\omega' \,\mathrm{d}\nu l \\
&\leq \left(\int_{\Omega'} \int_I M^2 |x_1(\omega', l) - x_2(\omega', l)|^2 \,\mathrm{d}\mu'\omega' \,\mathrm{d}\nu l \right)^{1/2} \\
&\leq M\delta \,,
\end{aligned}$$

this proves the FRÉCHET differentiability. □

Remark 5.42. *In the preceding lemma we proved not just uniform continuity but* LIPSCHITZ *continuity. So the assumptions on g can probably be weakened. Nevertheless, since f' is monotonical falling on $(0, \infty)$ the constraint is satisfied, if g' is* LIPSCHITZ *continuous on $[-\delta, \delta]$ and continuous outside.*

So using $E_{2,g}$ instead of E_2 in Equation (5.14) results in an alternative convergence result to Theorem 5.34 on a surrogate functional:

Corollary 5.43 (Generalized conditional gradient for $g \in \mathcal{G}_\delta$)**.** *Let $E_g(d, x)$ be defined as*

$$E_g = E_1 + E_{2,g} + E_3 + E_4 + E_5,$$

according to Equation (5.14) *and $g \in \mathcal{G}_\delta$ be a modified sparsity function as in Definition 5.39. Furthermore let I_U be an indicator function (cf. Equation* (5.21)*) with U defined as in Equation* (5.22) *and (d_0, x_0) satisfying $\|(d_0, x_0)\|_2 < \hat{C}$. If we define $F = E_1 + E_3 + E_{2,g}$ and $\Phi = E_4 + E_5 + I_U$, then the sequence $(d_n, x_n) \subset (W_{1,2}, W_{i,2})$ generated by the generalized conditional gradient algorithm satisfies $E_g(d_n, x_n) > E_g(d_{n+1}, x_{n+1})$ for all $n \in \mathbb{N}$, converging weakly to a $(d, x) \in (W_{1,2}, W_{i,2})$. If the convergence is strong, (d, x) is a stationary point of the algorithm.*

The proof would be the same as for the first case of Theorem 5.34, as now $E_{2,g}$ is a uniform continuous FRÉCHET differentiable functional.

Modification of generalized conditional gradient algorithm

Of course it would be preferable, to have results using $f \in \mathcal{F}_1$ instead of a surrogate functions in $\mathcal{G}_\delta$. As, due to the exploited properties of E_2, the results of [18] do not hold in this case, we want to analyze a generalization in the following paragraphes.

First the question arises, if we want to generalize in direction of using non-convex functionals Φ, or of using non-differentiable functionals F. As concave functions are a basic subset of $\mathcal{F}_1$, there is not much hope for finding a way to handle E_2 in Φ. On the other hand we can choose f to be differentiable everywhere outside 0, so we try to handle E_2 as summand of F at least for a subset of $f \in \mathcal{F}_1$.

We follow the structure of [18, Section 3], but change every proof. First of all we have to find a practical alternative for the derivative of E_2.

Definition 5.44 (H-derivative)**.** *Let $f \in \mathcal{F}_1$ be derivable in $(0,\infty)$ and $b = \lim_{t\to 0^+} \frac{f(t)}{t}$ be finite. Then we define for $x \in L_\infty(\Omega,\Omega')$ the H-derivative*[12] *of E_2 as*

$$E_2^H(u)(h) = \int_A f'(x(\omega',l))h(\omega',l)\,\mathrm{d}\mu'\omega'\,\mathrm{d}\nu l + \int_{A^C} b\left|h(\omega',l)\right|\,\mathrm{d}\mu'\omega'\,\mathrm{d}\nu l \ . \quad (5.23)$$

Here $A \subset \Omega' \times I$ denotes a set[13] *satisfying $x(\omega',l) \neq 0$ μ',ν-almost everywhere on A and $x(\omega',l) = 0$ μ',ν-almost everwhere on A^C.*

The definition in Equation (5.23) exists, as A is a measurable set according to [84, Satz 6.18]. Furthermore it is sensible, as b is the right side derivative of f in 0 and $-b$ the left side derivative. Please note, $E^H(u)$ is not linear in h, as the example below shows. Please remember for the next steps moreover the result of Lemma 5.3: $x \in L_\infty(\Omega',I)$ and $\|x\|_f < \infty$ result in $x \in L_p(\Omega',I)$ for all $p \in [1,\infty]$.

Example 5.45. *Let $u \equiv 0$ in $\Omega' \times I$ and let $h \in L_1(\Omega' \times I)$. Then we have*

$$E_2^H(u)(h) = b\|h\|_1 \ ,$$

this is a sublinear functional, but not a linear one.

But some remainder of linearity is still existent, namely in the multiplication of constants and in adding $E_2^H(x)(x)$:

Lemma 5.46. *Let f be chosen as in Definition 5.44. Then we have for $\alpha,\beta \in \mathbb{R}$:*

$$E_2^H(x)(\alpha h + \beta x) = \alpha E_2^H(x)(h) + \beta E_2^H(x)(x) \ .$$

Proof. The proof follows simply the definition. The first summand of $E_2^H(x)$ is clearly linear in h. Concerning the second summand, x equals 0 on A^C, so we have

$$\begin{aligned}\int_{A^C} b\left|\alpha h(\omega',l) + \beta x(\omega',l)\right|\,\mathrm{d}\mu'\omega'\,\mathrm{d}\nu l &= \alpha\int_{A^C} b\left|h(\omega',l)\right|\,\mathrm{d}\mu'\omega'\,\mathrm{d}\nu l \\ = \quad \alpha\int_{A^C} b\left|h(\omega',l)\right|\,\mathrm{d}\mu'\omega'\,\mathrm{d}\nu l &+ \beta\int_{A^C} b\left|x(\omega',l)\right|\,\mathrm{d}\mu'\omega'\,\mathrm{d}\nu l \ ,\end{aligned}$$

giving directly the assumption. □

In the remainder of this subsection we want to use this E_2^H as a surrogate for a derivative in the generalized conditional gradient algorithm and see what result we can prove. But first we want to set down the course of the algorithm. For reasons of generality we concern directly a function on an HILBERT space $H = H' \times L_2(U)$ and $P_{L_2(U)}$ denotes the projection on $L_2(U)$:

[12] H stands for "help"

[13] If A is not an open set, there is for every $\epsilon > 0$ an open set $\tilde{A} \subset A$ satisfying $\int_{A\setminus\tilde{A}} \mathrm{d}\mu'\omega'\,\mathrm{d}\nu l < \epsilon$. We identify A and $\tilde{A}$.

Algorithm 5.47. *Let $H = H' \times L_2(U)$ be a* Hilbert *space, $E_* = E_2 \circ P_{L_2(U)} : H \mapsto \mathbb{R}$ be defined analogous to Definition 5.4 and $\Phi : H \to (-\infty, \infty]$ be a functional satisfying Condition 2.25. Furthermore let $F : H \to \mathbb{R}$* Gâteaux *differentiable, $u = (d, x)$ and $v = (d_v, x_v)$.*

1. *Set $n = 0$ and choose $u_0 \in H$ with $\Phi(u_0) < \infty$.*
2. *Find a solution v_n of*
$$\operatorname*{argmin}_{v \in H} \left(\left\langle F'(u_n) \,\middle|\, v \right\rangle + E_*^H(u_n)(v) + \Phi(v) \right) .$$
3. *Determine s_n as*
$$s_n = \operatorname*{argmin}_{s \in [0,1]} F(u_n + s(v_n - u_n)) + E_*(u_n + s(v_n - u_n)) + \Phi(u_n + s(v_n - u_n)) .$$
4. *Set $u_{n+1} = u_n + s_n(v_n - u_n)$, stop if $u_{n+1} = u_n$, otherwise set $n = n + 1$ and return to Step 2.*

The next step is, to prove a first order necessary condition for a minimizer of E. The following conclusion is well known in the case of Gâteaux differentiable $F + E_2$, see, e.g., [18, 97]. For reasons of simplicity we denote in the following $\Omega' \times I = U$, $(\omega', l) = \zeta$ and $\mathrm{d}\mu'\omega' \, \mathrm{d}\nu l = \mathrm{d}\tilde{\mu}\zeta$.

Lemma 5.48. *Under the assumptions of Algorithm 5.47 the first order necessary condition for a minimum of $E(u) = F(u) + E_*(u) + \Phi(u)$ is given by*

$$u \in H : \quad \left\langle F'(u) \,\middle|\, v - u \right\rangle + E_*^H(u)(v - u) \geq \Phi(u) - \Phi(v), \quad \forall\, v \in H , \tag{5.24}$$

with E_*^H following the definition in Equation (5.23)*. This is equivalent to*

$$\left\langle F'(u) \,\middle|\, u \right\rangle + E_*^H(u)(u) + \Phi(u) = \min_{v \in H} \left(\left\langle F'(u) \,\middle|\, v \right\rangle + E_*^H(u)(v) + \Phi(v) \right) .$$

Proof. Assume $u = (d, x) \in H$ be a (global) minimum of E. Furthermore let $v = (d_v, x_v) \in H$ and $s \in (0, 1]$. Then the minimization property gives

$$F(u) + E_*(u) + \Phi(u) \leq F(u + s(v - u)) + E_*(u + s(v - u)) + \Phi(u + s(v - u)) .$$

According to the convexity of Φ we know $\Phi(u + s(v - u)) \leq (1 - s)\Phi(u) + s\Phi(v)$. So the inequality transforms to

$$\Phi(u) - \Phi(v) \leq \frac{F(u + s(v - u)) - F(u)}{s} + \frac{E_*(u + s(v - u)) - E_*(u)}{s} .$$

The limit $s \to 0$ gives for the first summand on the right side the derivative $\langle F'(u) \,|\, v - u \rangle$, while the second summand gives (with E_2 and A defined as in Defi-

nition 5.4 or rather Definition 5.44):

$$\begin{aligned}
&\lim_{s\to 0} \frac{E_*(u + s(v-u)) - E_*(u)}{s} \\
&= \lim_{s\to 0} \frac{\int_U f(x(\zeta) - s(x_v(\zeta) - x(\zeta))) - f(x(\zeta))\,\mathrm{d}\tilde{\mu}\zeta}{s} \\
&= \lim_{s\to 0} \frac{\int_A f(x(\zeta) - s(x_v(\zeta) - x(\zeta))) - f(x(\zeta))\,\mathrm{d}\tilde{\mu}\zeta}{s} \\
&\qquad + \frac{\int_{A^C} f(x(\zeta) - s(x_v(\zeta) - x(\zeta))) - f(x(\zeta))\,\mathrm{d}\tilde{\mu}\zeta}{s}
\end{aligned}$$

f is differentiable on A, so the first summand converges to the integral over the GÂTEAUX derivative on A. This can easily be proven as we can exchange the limit and the integral by LEBESGUE's dominated convergence theorem. This is applicable, since the fraction converges pointwise $\tilde{\mu}$-almost everywhere to $E_2'(x(\zeta))(x_v(\zeta))$ and this is always smaller than $|bx_v(\zeta)|$ and this is sufficient, according to Lemma 5.3.

On A^C we have $x(\zeta) = 0$, so

$$\lim_{s\to 0} \frac{E_*(u + s(v-u)) - E_*(u)}{s} = \int_A f'(x(\zeta))(x_v - x)\,\mathrm{d}\tilde{\mu}\zeta + \lim_{s\to 0} \int_{A^C} \frac{f(sx_v(\zeta))}{s}\,\mathrm{d}\tilde{\mu}\zeta \ .$$

In case the last integral is not finite, the conclusion holds immediately, as f is positive everywhere. Otherwise for $x_v(\zeta) \neq 0$, we have

$$\frac{f(sx_v(\zeta))}{s} = \frac{\int_0^1 \frac{\mathrm{d}}{\mathrm{d}t} f(stx_v(\zeta))\,\mathrm{d}t}{s} = \int_0^1 f'(stx_v(\zeta))x_v(\zeta)\,\mathrm{d}t \ .$$

So we obtain:

$$\lim_{s\to 0} \int_{A^C} \frac{f(sx_v(\zeta))}{s}\,\mathrm{d}\tilde{\mu}\zeta = \lim_{s\to 0} \int_{A^C} \int_0^1 \left|f'(stx_v(\zeta))\right| |x_v(\zeta)|\ \mathrm{d}t\ \mathrm{d}\tilde{\mu}\zeta$$

This exists also for $x_v(\zeta) = 0$, as $t \mapsto |f'(t)|$ is continuous. Additionally using the b from Definition 5.44, the derivative is limited by $0 \leq |f'(t)| \leq b < \infty$. As we know from Lemma 5.3: $x_v \in L_1(U)$, the limit exists due to LEBESGUE's dominated convergence theorem and equals

$$\lim_{s\to 0} \int_{A^C} \frac{f(sx_v(\zeta))}{s}\,\mathrm{d}\tilde{\mu}\zeta = \lim_{s\to 0} \int_{A^C} b\,|x_v(\zeta)|\ \mathrm{d}\tilde{\mu}\zeta \ .$$

Using Definition 5.44 of $E^H(u)$ we obtain then the assumption by Lemma 5.46. □

A comparison of the first order necessary condition for a minimum of $E(u)$ with Algorithm 5.47 shows, that u satisfying Condition (5.24) is equivalent to u being a stationary point of the algorithm. In the case of an u not satisfying Inequality (5.24), the following lemma is valid:

Lemma 5.49 (Descent of Algorithm 5.47)**.** *Let $E(u)$ be defined as in Equation* (5.14) *with an f satisfying the additional assumptions from Definition 5.44. Furthermore $u_n \in H$ does not satisfy Inequality* (5.24)*. Then one step of Algorithm 5.47 results in a $u_{n+1} \in H$ satisfying*

$$E(u_{n+1}) \leq E(u_n) \ .$$

Proof. Let $u_{n+1} = u_n + s_n(v_n - u_n)$ as defined in Step 4 of Algorithm 5.47. As Inequality (5.24) is not satisfied, there exists a $c_n > 0$ balancing the equation:

$$-\left\langle F'(u_n) \,\middle|\, v_n - u_n \right\rangle - E_*^H(u_n)(v_n - u_n) - c_n = \Phi(v_n) - \Phi(u_n) \tag{5.25}$$

Due to the property of E_*^H as limit, given in the proof of Lemma 5.48, we have for $0 \leq s \leq \delta \leq 1$ an $\epsilon > 0$ satisfying

$$\begin{aligned} E_*^H(u_n)(v_n - u_n) &= \lim_{s \to 0} \frac{E_*(u_n + s(v_n - u_n)) - E_*(u_n)}{s} \\ &\leq \frac{E_*(u_n + s(v_n - u_n)) - E_*(u_n)}{s} + \epsilon \ . \end{aligned} \tag{5.26}$$

An analog result is valid for F due to the Gâteaux differentiability:

$$F(u_n + s(v_n - u_n)) \leq F(u_n) + s\left\langle F'(u_n) \,\middle|\, v_n - u_n \right\rangle + s\epsilon \ . \tag{5.27}$$

Combining Equation (5.25) with the convexity of Φ we obtain

$$\begin{aligned} \Phi(u_n - s(v_n - u_n)) &\leq \Phi(u_n) + s(\Phi(v_n) - \Phi(u_n)) \\ &= \Phi(u_n) - s\left\langle F'(u_n) \,\middle|\, v_n - u_n \right\rangle - sE_*^H(u_n)(v_n - u_n) - sc_n \ . \end{aligned}$$

Replacing the derivatives, using Inequalities (5.26) and (5.27) and reordering the inequality results in

$$(F + E_* + \Phi)(u_n + s(v_n - u_n)) \leq (F + E_* + \Phi)(u_n) - (c_n - 2\epsilon)s.$$

So the assumption is valid for $2\epsilon \leq c_n$, guaranteed for sufficient small s. The lemma holds, since s_n minimizes $(F + E_* + \Phi)(u_n + s(v_n - u_n))$ over $s \in [0, 1]$. □

Using this result we know, the $E(u_n)$ are monotone falling and positive. So the question arises, if the convergence $E(u_n) \to E_0$ implies also u_n converging to a minimal function u; even if this would be just a local minimum, as in general there will be several stationary points of Algorithm 5.47.

For analyzing this convergence properties of the algorithm, we define an auxiliary functional

$$\Psi(u) = \left\langle F'(u) \,\middle|\, u \right\rangle + E_*^H(u)(u) + \Phi(u) - \min_{v \in H}(\left\langle F'(u) \,\middle|\, v \right\rangle + E_*^H(u)(v) + \Phi(v)) \tag{5.28}$$

According to the structure of the algorithm, Ψ equals zero exactly at the stationary points of Algorithm 5.47. Furthermore, $\Psi(u)$ is positive for all u, since $u \in H$. All in all we have to answer three questions:

1. Is there for every sequence u_n a converging subsequence $u_{n_j} \to u$?
2. Is u a stationary point of Algorithm 5.47?
3. Is $(F + E_* + \Phi)(u) \leq (F + E_* + \Phi)(u_n)$ valid for all $n \in \mathbb{N}$?

The third questions answers positively from Lemma 5.49 and the proof of Theorem 5.25, due to the weak lower semicontinuity of E in $(W_{1,2}, W_{i,2})$. There we also proved results about the existence of converging subsequences of minimizing sequences to answer the first question. For the second question we need to prove, that Ψ is *tau*-lower semicontinuous and $\Psi(u_n)$ converges towards 0 in the τ-topology. In the following we handle this two points:

Lemma 5.50. *Let F, E_* and Φ be defined as in Lemma 5.48. Furthermore let f have a* Lipschitz *continuous derivative outside* 0 *and F be continuously* Fréchet *differentiable. Then Ψ, as defined in Equation* (5.28)*, is lower semicontinuous.*

Proof. The F and Φ parts of the proof are the same as in [18, Lemma 4], while the E_* part uses directly the definitions:

Let $u_n = (d_n, x_n)$ be a given sequence in H, $u_n \to u = (d, x)$. Furthermore $w = (d_w, x_w)$ is an arbitrary element of H, $\Phi(w) < \infty$. Then we know:

$$\begin{aligned} \Psi(u_n) &= \left\langle F'(u_n) \,\middle|\, u_n \right\rangle + E_*^H(u_n)(u_n) + \Phi(u_n) \\ &\quad - \min_{v \in H} \left(\left\langle F'(u_n) \,\middle|\, v \right\rangle + E_*^H(u_n)(v) + \Phi(v) \right) \\ &\geq \underbrace{\left\langle F'(u_n) \,\middle|\, u_n - w \right\rangle + \Phi(u_n) - \Phi(w)}_{\Psi_1(u_n, w)} + \underbrace{E_*^H(u_n)(u_n) - E_*^H(u_n)(w)}_{\Psi_2(u_n, w)} \end{aligned} \tag{5.29}$$

We now prove the lower semicontinuity of Ψ_1 and Ψ_2 separately. In the case of Ψ_1 this is easily done by the continuity of F' and the lower semicontinuity of Φ, given by Condition 2.25:

$$\begin{aligned} \liminf_{n \to \infty} \Psi_1(u_n, w) &= \left\langle F'(u_n) - F'(u) \,\middle|\, u_n - w \right\rangle + \left\langle F'(u) \,\middle|\, u_n - w \right\rangle + \Phi(u_n) - \Phi(w) \\ &\geq \left\langle F'(u) \,\middle|\, u - w \right\rangle + \Phi(u) - \Phi(w) \end{aligned}$$

For getting the limit of Ψ_2 we define $A_n \subset \Omega' \times I$ as a set, where $x_n \neq 0$ $\tilde{\mu}$-almost everywhere on A_n and $x_n = 0$ $\tilde{\mu}$-almost everywhere on A_n^C. Furthermore A and A^C satisfy the same conditions for x. Then we know by definition of E_* in Algorithm 5.47 (Below we suppress the integration variable for better readability):

$$\begin{aligned} \Psi_2(u_n, w) &= E_*^H(u_n)(u_n) - E_*^H(u_n)(w) \\ &= \underbrace{\int_{A_n} f'(x_n) x_n \, \mathrm{d}\tilde{\mu}\zeta}_{\Psi_{2,1}} - \underbrace{\int_{A_n} f'(x_n) x_w \, \mathrm{d}\tilde{\mu}\zeta}_{\Psi_{2,2}} - \underbrace{\int_{A_n^C} b \, |w| \, \mathrm{d}\tilde{\mu}\zeta}_{\Psi_{2,3}} \end{aligned}$$

The integral of $b\,|x_n|$ over A_n^C eliminates, as x_n equals zero $\tilde{\mu}$-almost everywhere on A_n^C.

For the analysis of $\Psi_{2,1}$ we want to mention first, that

$$f'(t)t = \left|f'(t)\right| |t| \geq 0 \ ,$$

since f is monotonically increasing on $[0, \infty)$ and monotonically decreasing on $(-\infty, 0]$. Now we want to prove $\Psi_{2,1}(x_n) \to \Psi_{2,1}(x)$:

$$\begin{aligned}
|\Psi_{2,1}(x_n) - \Psi_{2,1}(x)| &= \left| \int_{A_n} f'(x_n)x_n - f'(x)x \, \mathrm{d}\tilde{\mu}\zeta \right| \\
&= \left| \int_{A_n} \left(\left|f'(x_n)\right| - \left|f'(x)\right|\right) |x_n| + \left|f'(x)\right| \left(|x_n| - |x|\right) \, \mathrm{d}\tilde{\mu}\zeta \right| \\
&\leq \int_{A_n} \left|\left|f'(x_n)\right| - \left|f'(x)\right|\right| |x_n| \, \mathrm{d}\tilde{\mu}\zeta + \int_{A_n} \left|f'(x)\right| ||x_n| - |x|| \, \mathrm{d}\tilde{\mu}\zeta
\end{aligned}$$

Since f' is Lipschitz continuous on $(0, \infty)$, $\lim_{t \to 0^+} f' = b > 0$ and $f(-t) = f(t)$, the mapping

$$t \mapsto \begin{cases} |f'(t)| & t \neq 0 \\ b & t = 0 \end{cases}$$

is Lipschitz continuous on $\mathbb{R}$. Let M be the Lipschitz constant and remember $|f'(t)| \leq b \ \forall t \in \mathbb{R}$, we obtain

$$\begin{aligned}
|\Psi_{2,1}(x_n) - \Psi_{2,1}(x)| &\leq M \int_{A_n} |x_n - x| \, |x_n| \, \mathrm{d}\tilde{\mu}\zeta + \int_{A_n} \left|f'(x)\right| |x_n - x| \, \mathrm{d}\tilde{\mu}\zeta \\
&\leq M \|x_n - x\|_2 \|x_n\|_2 + b\|x_n - x\|_1 \\
&\to 0 \ .
\end{aligned}$$

The last step is based on the result of Lemma 5.3: Since x and x_n are elements of L_2 and $\|x\|_f, \|x_n\|_f < \infty$, they are also elements of L_1.

For proving the lower semicontinuity of $\Psi_{2,2}$ we introduce the sets A_n^+ and A_n^-, with $x(\zeta)x_n(\zeta) > 0$ for $\tilde{\mu}$-almost every $\zeta \in A_n^+$, $x(\zeta)x_n(\zeta) < 0$ for $\tilde{\mu}$-almost every $\zeta \in A_n^-$ and $x(\zeta) = 0$ for $\tilde{\mu}$-almost every $\zeta \in \{A_n^+ \cup A_n^-\}^C$. Then we obtain

$$\begin{aligned}
\Psi_{2,2}(u_n, w) &= \int_{A_n} f'(x_n)x_w \, \mathrm{d}\tilde{\mu}\zeta \\
&= \underbrace{\int_{A_n^+} f'(x_n)x_w \, \mathrm{d}\tilde{\mu}\zeta}_{\Psi_{2,2,1}} + \underbrace{\int_{A_n^-} f'(x_n)x_w \, \mathrm{d}\tilde{\mu}\zeta}_{\Psi_{2,2,2}} \\
&\quad + \underbrace{\int_{A_n \cap (A_n^+ \cup A_n^-)^C} f'(x_n)x_w \, \mathrm{d}\tilde{\mu}\zeta}_{\Psi_{2,2,3}} \ .
\end{aligned}$$

The continuity of $\Psi_{2,2,1}$ arises directly from

$$\begin{aligned}
|\Psi_{2,2,1}(x_n, x_w) - \Psi_{2,2,1}(x, x_w)| &= \left| \int_{A_n^+} f'(x_n)x_w - f'(x)x_w \, \mathrm{d}\tilde{\mu}\zeta \right| \\
&\leq \int_{A_n^+} |f'(x_n) - f'(x)||x_w| \, \mathrm{d}\tilde{\mu}\zeta \\
&\leq M \int_U |x_n - x||x_w| \, \mathrm{d}\tilde{\mu}\zeta \ \to 0 \ .
\end{aligned}$$

The last inequality is given by the Lipschitz continuity of $|f'|$ and $|f'(x_n) - f'(x)| = ||f'(x_n)| - |f'(x)||$ on A_n^+.

The remaining two parts $\Psi_{2,2,2}$ and $\Psi_{2,2,3}$ are combined with $\Psi_{2,3}$, so we have

$$\begin{aligned}
&(\Psi_{2,2,2} + \Psi_{2,2,3} + \Psi_{2,3})(u_n, w) \\
&= \underbrace{\int_{A_n^C \cap (A_n^+ \cup A_n^-)^C} b\,|x_w|\,\mathrm{d}\tilde{\mu}\zeta + \int_{A_n \cap (A_n^+ \cup A_n^-)^C} f'(x_n) x_w\,\mathrm{d}\tilde{\mu}\zeta}_{\Psi_A} \\
&\quad + \underbrace{\int_{A_n^C \cap A} b\,|x_w|\,\mathrm{d}\tilde{\mu}\zeta + \int_{A_n^-} f'(x_n) x_w\,\mathrm{d}\tilde{\mu}\zeta}_{\Psi_B} \ .
\end{aligned}$$

For Ψ_A we have

$$\begin{aligned}
\limsup_{n\to\infty} \Psi_A &\leq \limsup_{n\to\infty} \left(\int_{A_n^C \cap (A_n^+ \cup A_n^-)^C} b\,|x_w|\,\mathrm{d}\tilde{\mu}\zeta + \int_{A_n \cap (A_n^+ \cup A_n^-)^C} b\,|x_w|\,\mathrm{d}\tilde{\mu}\zeta \right) \\
&= \int_{(A_n^+ \cup A_n^-)^C} b\,|x_w|\,\mathrm{d}\tilde{\mu}\zeta \to \int_{A^C} b\,|x_w|\,\mathrm{d}\tilde{\mu}\zeta \ ,
\end{aligned}$$

while Ψ_B is smaller than ϵ, as

$$\begin{aligned}
|\Psi_B| &\leq \left| \int_{A_n^C \cap A} b\,|x_w|\,\mathrm{d}\tilde{\mu}\zeta + \int_{A_n^-} b\,|x_w|\,\mathrm{d}\tilde{\mu}\zeta \right| \\
&= \int_{A_{0,n}} b\,|x_w|\,\mathrm{d}\tilde{\mu}\zeta
\end{aligned}$$

and the Technical Lemma B.3 show. Here $A_{0,n}$ denotes a set, where $x(\zeta) \neq 0$ and $x(\zeta) \cdot x_n(\zeta) \leq 0$ $\tilde{\mu}$-almost everywhere.

Combining all the above results we obtain:

$$\begin{aligned}
\liminf_{n\to\infty} \Psi_2(u_n, w) &= \liminf_{n\to\infty} \Psi_{2,1}(x_n) - \limsup_{n\to\infty} (\Psi_{2,2,1}(x_n, x_w) \\
&\qquad + \Psi_A(x_n, x_w) + \Psi_B(x_n, x_w)) \\
&\geq \Psi_{2,1}(x) - \Psi_{2,2,1}(x, x_w) - \int_{A^C} b\,|x_w|\,\mathrm{d}\tilde{\mu}\zeta - 0 \\
&= \int_A f'(x)(x - x_w)\,\mathrm{d}\tilde{\mu}\zeta - \int_{A^C} b x_w\,\mathrm{d}\tilde{\mu}\zeta \\
&= \Psi_2(u, w) \ ,
\end{aligned}$$

which is the lower semicontinuity of Ψ_2 and by this we obtain by Equation (5.29) the lower semicontinuity of the whole Ψ. □

After proving Lemma 5.50, we know that a sequence established by Algorithm 5.47 converges towards a stationary point of the algorithm, if it converges towards a zero of Ψ and has a convergent subsequence. So to go on we have to answer the question, if every sequence generated by the algorithm converges towards a zero of Ψ. Unfortunately this is not the case, as the following example shows:

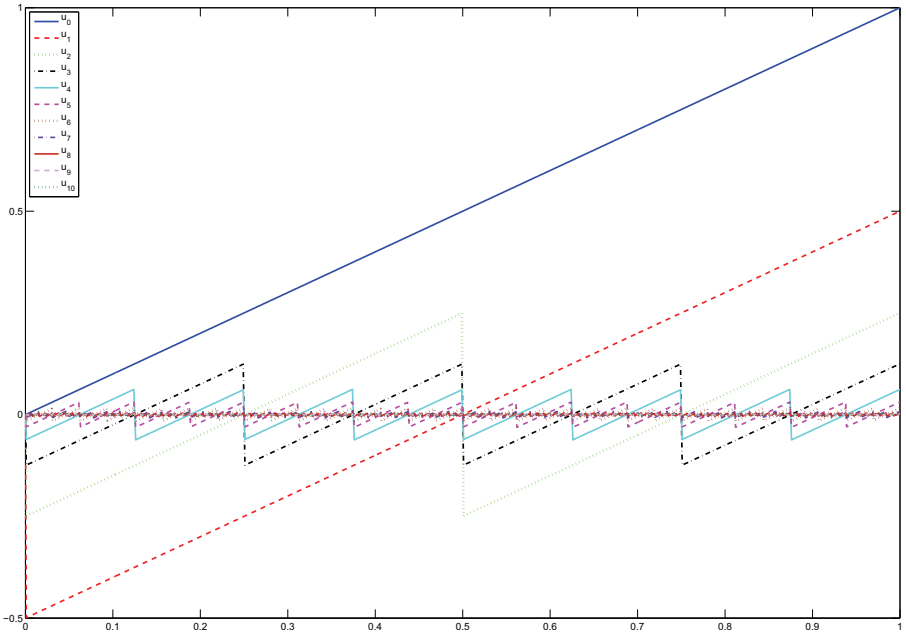


Figure 5.2: u_n according to Example 5.51

Example 5.51. *Let*

$$u_0(\zeta) = \begin{cases} \zeta & x \in [0,1] \\ 0 & \textit{otherwise} \end{cases}$$

furthermore let $f(t) = |t|$*, and, for reasons of simplicity,* $F \equiv 0$ *and* $\Phi(u) = I_{\|u\|_2 \leq 1}(u)$*. In that case the algorithm gives a* v_n *of the form* $v_n(\zeta) = -\operatorname{sign}(u_n)$*, while the* u_n *look like sawtooth-functions as given in Figure 5.2 and in magnification in Figure 5.3, which converge to* 0 *pointwise and in* L_2*. So we obtain for the* $\Psi(u_n)$*:*

$$\Psi(u_n) = E_*^H(u_n)(u_n) - E_*^H(u_n)(v_n) = \int_0^1 1 + |u_n(\zeta)| \, \mathrm{d}\tilde{\mu}\zeta$$

converging towards 1*, while the* $s_n = 2^{-n}$ *converge towards* 0*.*

Since, according to the proof of Lemma 5.49, the s never equal zero, we have in general u not to be a stationary point of Algorithm 5.47. For example, we can extend the previous example that way, to choose

$$u_0(\zeta) = \begin{cases} \zeta & x \in [0,1] \\ 3 & x \in (1,2] \\ 0 & \text{otherwise} \end{cases}$$

and

$$f(t) = \begin{cases} |t| & -1.8 \leq t \leq 1.8 \\ 5.05\,|t| - 1.125t^2 - 3.645 & 1.8 < |t| \leq 2.2 \\ 1.8 + |t|\,/10 & \text{otherwise.} \end{cases} \tag{5.30}$$

Then $f \in \mathcal{F}_1$ (cf. Figure 5.4) is continuous differentiable outside 0 and u_n converge to $u = 3\chi_{(1,2]}$ being not a stationary point of the algorithm.

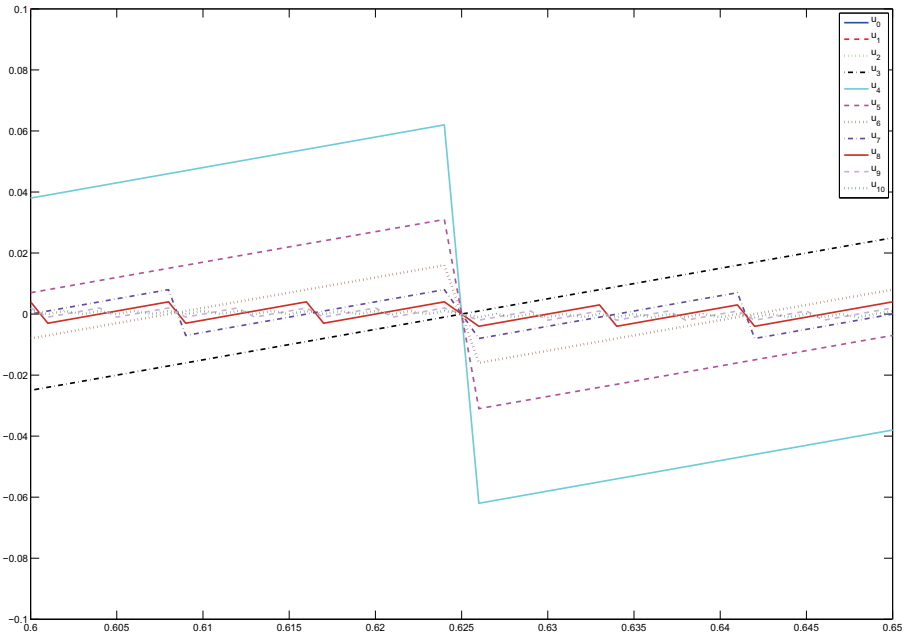


Figure 5.3: Detail of u_n according to Example 5.51

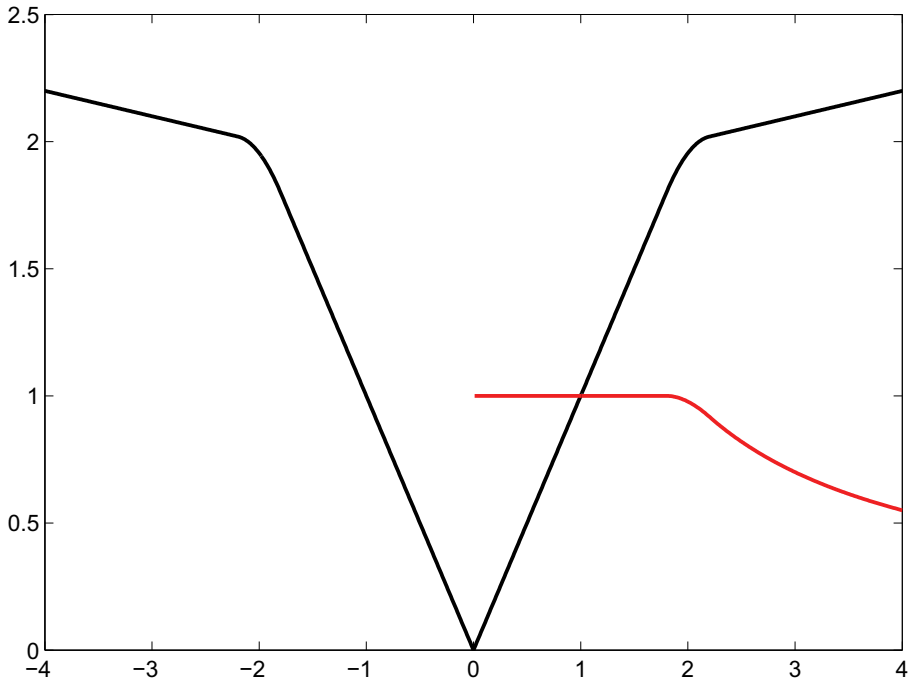


Figure 5.4: Sparsity measure f according to Equation (5.30) (black) and $f(t)/t$ (red) for the positive part

Remark 5.52. *If we examine the proof of the coresponding result in the case of the original generalized conditional gradient algorithm as stated in [18, Lemma 5], we see, the missing property of E_*^H, in comparison with a standard* FRÉCHET *derivative, is an equivalent to the intermediate value theorem. But that can not be valid in this framework, as there is no continuity of f' in 0.*

Needless to say, it is a big disadvantage of Algorithm 5.47 not to guarantee convergence to a stationary point. Nevertheless we can state the following result in the case of the matching dictionary problem based on Definition 5.4.

Theorem 5.53 (Convergence of Algorithm 5.47 in the case $f \in \mathcal{F}_1$)**.** *Let E be defined as in Equation (5.17). Furthermore let $f \in \mathcal{F}_1$ be a sparsity measure, differentiable on $(0,\infty)$ with* LIPSCHITZ *continuous derivative f'. Additionally let $\lim_{t\to 0} f'(t) = b < \infty$, $E(d_0, x_0) \neq \infty$ and I_U be an indicator function according to Equation (5.21).*
Then Algorithm 5.47 applied on $E + I_U$ with starting values $(d_0, x_0) \in (W_{1,2}, W_{i,2})$ and $\|(d_0, x_0)\|_2 < \hat{C}$ results in a sequence $(d_n, x_n)_{n\in\mathbb{N}} \subset (W_{1,2}, W_{i,2})$ with $E(d_n, x_n) \leq E(d_{n-1}, x_{n-1})$ for all $n \in \mathbb{N}$ and there is a subsequence $(d_{n_j}, x_{n_j}) \rightharpoonup (d, x)$ with $E(d, x) \leq E(d_n, x_n)$ for all $n \in \mathbb{N}$.
If furthermore the convergence is strong and if $\Psi(u_{n_j}) \to 0$, then (d, x) is a stationary point of Algorithm 5.47 according to E.

Proof. To prove this theorem we basically need to confirm, that E satisfies the properties demanded in Lemmata 5.49 and 5.50. As we proved in Lemmata 5.31 and 5.32 the continuous FRÉCHET differentiability of $F(d, x) = (E_1 + E_3)(d, x)$, we have all the properties on F. Furthermore $\Phi(d, x) = (E_4 + E_5 + I_U)(d, x)$ is satisfying the claims of Condition 2.25 according to the definitions of g_1, g_2 and g_3 in Condition 5.9 and Equations (5.13) and (5.14) analog to the proof of Theorem 5.34. Also like in Theorem 5.34, I_U has no influence on the values $E(d_n, x_n)$.
Since all necessary constraints on E_2 or rather f are given in the assumptions of the theorem, the Lemmata 5.49 and 5.50 are valid.

Furthermore we need the existence of a weak converging subsequence of (d_n, x_n). This is also clear, due to the weak coercivity of E, proven in Remark 5.23. The minimizing property arises then from the weak lower semicontinuity of E, according to Theorem 5.25, while the stationarity in case of the last paragraphe arises directly from the definition of Ψ in Equation (5.28). □

5.2.3 Summary

We obtain in this section several results about the usability of the generalized conditional gradient algorithm and a variant for finding a minimizing sequence (d_n, x_n) of three different cases of E. Every time $E(d_n, x_n)$ was monotonical falling (cf. Lemma 2.26 or rather Lemma 5.49) and a subsequence (d_{n_j}, x_{n_j}) converged weakly to (d, x). Furthermore this limit satisfies $E(d, x) \leq E(d_n, x_n)$ for all $n \in \mathbb{N}$. Since the functional is in general just weak coercive, we can not prove the strong convergence $(d_{n_j}, x_{n_j}) \to (d, x)$. If this exists, we could prove in the case $f \in \mathcal{F}_2$ and for

LIPSCHITZ continuous $f \in \mathcal{G}_\delta$, as given in Theorem 5.34 and Corollary 5.43, that (d, x) is a stationary point of the algorithm. This is not possible in the case of

$$f \in \{\mathcal{F}_1 | f' < b < \infty, f' \text{ LIPSCHITZ on } (0, \infty)\} ,$$

where the non-differentiability of f in 0 prevents better results.

In applicational frameworks, using discrete data, here strong and weak topology equal, we obtain also positive results about strong convergence, combined with this convergence of the algorithm to stationary points. Thus numerical applications should generally serve well.

Chapter 6

Applications

In the last chapters we developed, beside theoretical results, several algorithmic devices for matching wavelets and learning dictionaries. Now we want to test this algorithms in several frameworks. The focus of this applications is mostly directed at the algorithms of Chapter 4, since it is in comparison to that one from Chapter 3 fully developed by ourselves and has a wider field of application. Furthermore it fits better in the framework of the theoretical chapter. We avoid to adapt the conditional gradient method from Section 5.2 to application, due to the structure of our data, demanding for a shift-invariant devices. This constraint is satisfied by the MODW, but not by the gradient algorithm. Nevertheless the later method is also interesting, especially since it is one of the rare examples that combine the optimization of dictionary $\mathcal{D}$ and coefficients X.

As we already mentioned in Section 1.3, arised during the past years a wide field of applications for sparse coding and dictionary learning approaches. Analog to that we want to exemplify here the MODW algorithm on data, based on two origins: In Section 6.2 we separate different overlapping sounds in machine noise, exemplified at a linear guideway. The aim is to recognize the status of the machinery from the combination of atoms. Especially we try to identify the position of one predefined machine error. Furthermore, around this example also some applications of the algorithm of Chapter 3 are given, demonstrating the necessity to use the shift-invariant methods stated in Section 3.3.
The second real-life example deals in Section 6.3 with the recognition of notes in the record of some piano performance. The idea is to reconstruct the score from the record. Additionally this problem gives us the chance to compare the performance of MODW with some alternative algorithms proposed by Blumensath for learning shift-invariant dictionaries.
Supplementary to this two examples we present first in Section 6.1 some results on simulated data, for testing the performance of MODW in several well defined frameworks.

All the computations in this chapter (beside Subsection 6.3) had been made on a 3 GHz Intel Pentium 4 PC using the Linux KDE version of MatLab 7.3.0.298. For the basic wavelet and sparse coding algorithms we used “WaveLab 802” (cf.

`http://www-stat.stanford.edu/~wavelab/`) or rather "Atomizer 802" (a newer version, called "Sparselab," is available under `http://sparselab.stanford.edu/`).

6.1 Examples with simulated data

The main preface of using a set of simulated data is the potential to analyze the influence of different parameters directly. So we try in this section two things: First a comparison of a bundle of dictionary reconstruction problems, changed in just one parameter. Second we execute the option explained on Page 47 to use just one scale for the construction of a dictionary, but apply this dictionary later by using several scales.

6.1.1 Testing parameters of MODW

The basic idea of this simulation is to construct randomly a dictionary $\mathcal{D}$ being a union of several wavelet bases as defined in Equation (4.1). Afterwards we construct from this dictionary a set $\mathcal{Y}$ (cf. Page 16) of signals y_i and try to reconstruct $\mathcal{D}$ from $\mathcal{Y}$ using MODW.

For this purpose we execute the following steps:

1. Define J vectors s_j of length $K+1$, $\|s_j\|_2 = 1$, $|s_{j,K+1}| > 0.1$.
2. Use the s_j as lifting filters for obtaining wavelets analog to Equation (3.5). Here $s = (s_{j,1}, \ldots, s_{j,K})$ and $c = s_{j,K+1}$. The dictionary $\mathcal{D}$ is then defined as in Equation (4.1).
3. Define a set $\mathcal{Y} = \{y_i\}_{i=1}^N \subset \mathbb{R}^n$, of signals, via

$$y_i = \sum_{l=1}^{L} f_{i,l} d_l.$$

   l, L are defined as in Equation (4.1), $\|(f_{i,1}, \ldots, f_{i,L})\|_0 = \Delta$ and $|f_{i,l}| \leq 1$ for all $1 \leq i \leq N$ and $1 \leq l \leq L$.
4. Use MODW and the $\mathcal{Y} = \{y_i\}$ to compute $\mathcal{D}^*$ as approximation[1] to $\mathcal{D}$.
5. Compare $\mathcal{D}$ and $\mathcal{D}^*$.

We repeated this example 40 times for obtaining average values for computation time and approximation error. For the basis of comparison we defined $J = 2$, $K = 4$, $N = 30$, $n = 16$ and $\Delta = 3$, in that case wavelets look (depending on the smoothness parameter, below we use the piecewise constant ones), e.g., like in Figure

[1] The upper index * denotes in any case throughout this chapter the results of MODW or its properties.

6.1. Furthermore, we initialized the MODW for using OMP (cf. Page 15), ignoring absurd lifting (cf. Page 50), restricting the wavelets to the support of the y_i and using a starting dictionary[2] defined by $s^0_{j,k} = s_{j,k}+\delta_{j,k}$ for all $1 \leq j \leq J$, $1 \leq k \leq K+1$ and uniformly distributed random values $-0.01 \leq \delta_{j,k} \leq 0.01$. The algorithm stopped after 10 iterations or if the relative correction of the approximation error falls below 10^{-5}. On the first sight the starting dictionary is relatively near to the original one, nevertheless, for the above mentioned parameters, this is equivalent to an average relative distance of 10.7% for every $s_{j,k}$. Furthermore the relative approximation error (d^0 denotes the atoms deduced from $s^0_{j,k}$), using the original coefficients $f_{i,l}$,

$$e_0^2 = \frac{\sum_{i=1}^{N} \|y_i - \sum_{l=1}^{L} f_{i,l} d_l^0\|^2}{\sum_{i=1}^{N} \|y_i\|^2} \ ,$$

equals in average $e_0 = 0.694$. Interestingly the approximation errors of OMP (cf. Page 16) has also some major influence. For example, if we apply the MODW as given above using the generating $\mathcal{D}$ as starting dictionary, we still result in a $\mathcal{D}^*$ connected to a average relative error $e^* = 0.149$. In the following tabulars we vary just one of the values J, K, N, n and Δ and observe maximum, minimum, average and standard deviation of e^* and e^0, as well as the average number of iterations and the computation time.

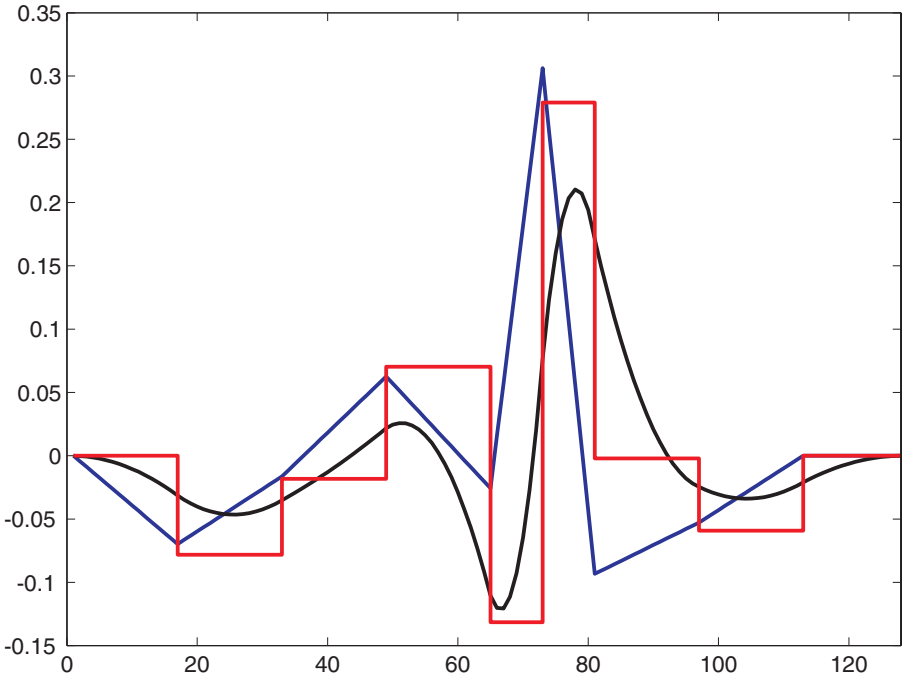


Figure 6.1: Examples of generating wavelets used during construction of $\mathcal{Y}$

[2]The upper index ^{0} denotes in any case throughout this chapter the starting-values of MODW or its properties.

K	error type	max	min	mean	std	time [s]	iterations
K=2	e^0	0.166	0.0631	0.121	0.0412	16.195	8
	e^*	0.157	0.0515	0.108	0.0407		
K=4	e^0	0.226	0.184	0.213	0.0171	20.446	10
	e^*	0.189	0.128	0.159	0.0254		
K=8	e^0	0.447	0.190	0.281	0.104	16.039	8
	e^*	0.174	0.118	0.152	0.0224		

N	error type	max	min	mean	std	time [s]	iterations
N=15	e^0	0.208	0.104	0.169	0.0429	9.645	9.2
	e^*	0.167	0.0722	0.138	0.0408		
N=30	e^0	0.226	0.184	0.213	0.0171	20.446	10
	e^*	0.189	0.128	0.159	0.0254		
N=60	e^0	0.267	0.172	0.203	0.0420	40.703	10
	e^*	0.161	0.118	0.144	0.0160		

n	error type	max	min	mean	std	time [s]	iterations
n=16	e^0	0.226	0.184	0.213	0.0171	20.446	10
	e^*	0.189	0.128	0.159	0.0254		
n=32	e^0	0.205	0.0691	0.141	0.0614	21.154	9
	e^*	0.202	0.0664	0.126	0.0537		
n=64	e^0	0.143	0.0535	0.0869	0.0355	24.332	8.6
	e^*	0.142	0.0490	0.0801	0.0393		

J	error type	max	min	mean	std	time [s]	iterations
J=2	e^0	0.226	0.184	0.213	0.0171	20.446	10
	e^*	0.189	0.128	0.159	0.0254		
J=3	e^0	0.249	0.197	0.229	0.0242	26.222	8.4
	e^*	0.181	0.148	0.169	0.0133		
J=4	e^0	0.236	0.165	0.200	0.0274	40.593	9.8
	e^*	0.191	0.126	0.149	0.0252		

Δ	error type	max	min	mean	std	time [s]	iterations
Δ=2	e^0	0.268	0.112	0.175	0.0665	16.122	9.4
	e^*	0.150	0.0551	0.113	0.0382		
Δ=3	e^0	0.226	0.184	0.213	0.0171	20.446	10
	e^*	0.189	0.128	0.159	0.0254		
Δ=4	e^0	0.368	0.133	0.225	0.0922	22.875	9.2
	e^*	0.173	0.117	0.143	0.0230		

Interestingly increasing the length of the lifting filters K does not change computation time and error (of course K=2 is a more easy subcase with smaller error). Also the influence of Δ, the sparsity, and N, the number of signals, is small in this range ($\Delta = 2$ is again a special case), just the standard deviation lessens in the later case while the computation time is increasing. The later one can be expected, since the costs for the OMP increase linearly with N, while it also increases the number of local minima. Also a change of the number of wavelet bases J has big influence on computation time, as it increases the number of coefficients to be calculated. A positive result is that the influence of J on the error, as well as that one of Δ, is

sublinear or almost inexistent. Thus the error increases much slower than the complexity of the problem. Lastly the falling error with increasing length of signals n is predictable, as the number of overlapping atom decreases, giving a better recognition by OMP. Another effect can be observed in the relationship between e^0 and e^*: While increasing n usually decreases the improvement obtained by MODW, it increases with K and also less with N and Δ. This shows, that the improvement of $e^0 - e^*$ is better if the original problem features more overlapping atoms, as arising in applications in general, or if the number of information increases.

A last aspect affects the choice of the sparse coding algorithm. In the preceding tabulars we used just OMP. A similar situation is given for most applications, since other algorithms are usually to slow. Surprisingly, for this special example the OMP is also the most effective algorithm, basis pursuit produces in average 4% more error.

Of course there is also the alternative to restrict the MODW to just one scale as proposed on Page 47. We do not present this case in detail here, as it is more or less a parallel to the above one with one missing feature. We included an application of that kind already in [132]. There we presented additionally to the relative reconstruction error also the number of refound wavelet bases, i.e. that one, where

$$\max_{j_1=1}^{J} \left| \langle d_{j_1} \mid d^*_{j_2} \rangle \right| > 1 - \epsilon$$

for $\epsilon \in \{0.01, 0.05\}$ and we obtained rates up to 83% or rather 86% (using a full random initialized start dictionary). That computations highlight also the big positive influence of suppressing absurd lifting (cf. Page 50), changing the relative error by 7.56%. On the other side, there is also shown, that the convergence force on the OMP has just minor influence, the relative error decreases in average by just 1.15%.

For concerning one scale applications with simulated data at all in this thesis, we give in the following subsection a short recapitulation of an in-between example that is also a part of [132].

6.1.2 One and several scales

In this second example we demonstrate the possibilities of a restricted version of the MODW for signal analysis. For this purpose we use two set of signals: A first one $\mathcal{Y}_1$ for the learning of the dictionary and a second one $\mathcal{Y}_2$ to be analyzed. Both are created using the same wavelet-dictionary $\mathcal{D}$, defined similar as in Subsection 6.1.1, especially Figure 6.1, but here we set $J = 3$. For construction of $\mathcal{Y}_1 = \{y_i\}_{i=1}^{200} \subset \mathbb{R}^{128}$ we restrict the wavelets to one scale ($\bar{j} = j'$ in Equation (4.1)) and obtained a subdictionary $\mathcal{D}'$ containing 17 shifts at the same scale for each of the three wavelets (51 atoms). Each of the 200 signals in $\mathcal{Y}_1$ is a linear combination of three atoms. The 10 signals of the second set $\mathcal{Y}_2$ consist of 12 atoms out of 4 scales using 128 shifts for the lowest scale. Examples of the signals of both sets are given in Figure 6.2 or rather 6.3. The computation has been repeated 50 times, initialized with different random values, in the following are given mean values.

Similar to Section 6.1.1 the dictionary $\mathcal{D}^*$ has been learned by MODW, based on $\mathbf{Y}_1$. But here we changed the following parameters: We restricted the MODW to use

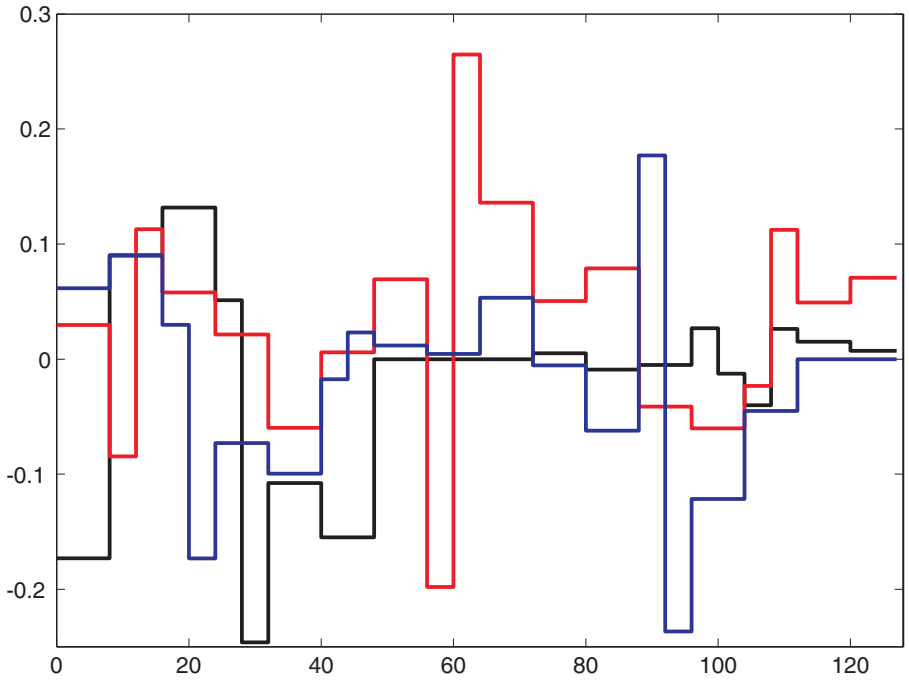


Figure 6.2: Examples of signals from $\mathcal{Y}_1$

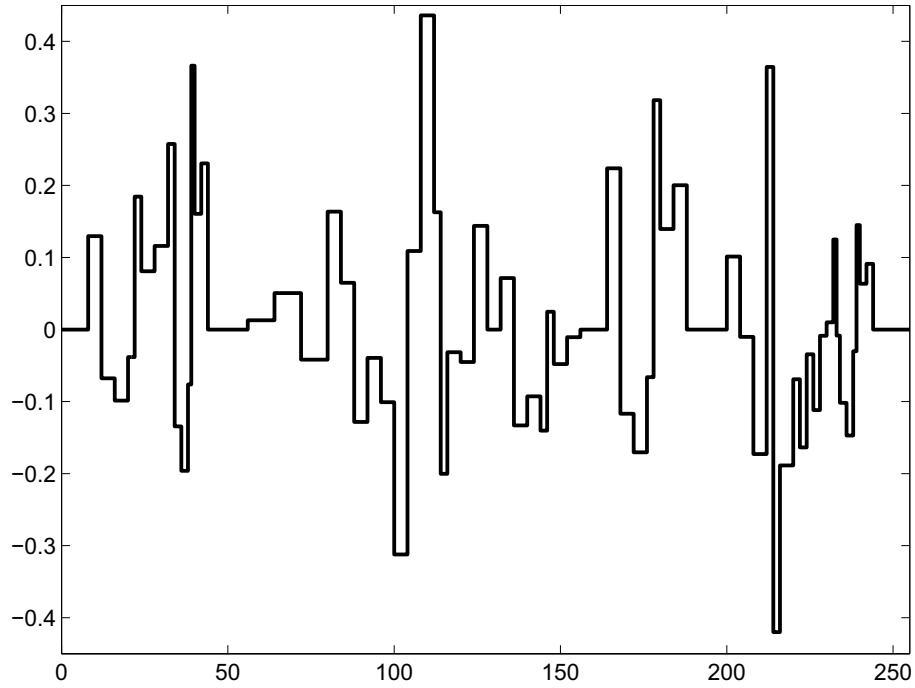


Figure 6.3: Example of a signal from $\mathcal{Y}_2$

just one scale, but defined the starting dictionary (or more exactly the underlying $s_{k,k}$ randomly and allowed up to 80 iterations (generally the algorithm stops before). Furthermore, we treat absurd lifting as a local minimum. All in all the mean relative error of the dictionary reconstruction lessens to 5.63%. This result is achieved in 2:35 minutes, since (additionally to the number of iterations) the number and length of signals in $\mathcal{Y}_1$ is higher than in the preceding example.

Afterwards $\mathcal{Y}_2$ has been analyzed by using OMP or rather BP (with interior point method, see Page 16) based on the dictionary $\mathcal{D}^*$. Both sparse coding algorithms were modified to use fast wavelet transform for decomposition, as to use, for reasons of computation time, in most applicational frameworks. As quality measure for the signal analysis had been used the relative error $E(\mathbf{Y}_2, \tilde{\mathbf{D}}_R, X)$ as defined in Equation (4.3). The results of both algorithms had been limited to 12 atoms (in the case of BP that ones whose coefficients have maximal absolute values) as much as are used for construction of the signals. So we obtained a relative error for the analysis of 5.57% in the OMP case and 1.19% in the BP case. The results show also the well known preface of BP against OMP, even in this case, where the dictionary is constructed by help of OMP. We set aside the possibility to modify the procedure and apply also BP in MODW, due to the expected increase of computation time.

6.2 Condition monitoring of linear guideways

In this section we want to exemplify the usage of the methods from Chapter 3 and 4 for the detection of features in the acoustic emission (AE) of a machinery and the decomposition of this signals. As machinery we chose a linear guideway as presented in Appendix A. The running noise is recorded by use of an encapsulated piezo-ultrasonic-microphone fixed either at the front end of the lateral surface of the carriage (see Figure A.5) or at the front face of the guideway. Depending on the position of the microphone the noise is extremely different and even small displacements have big effects, as Figure 6.4 show.

In the following two subsections we illustrate first the wavelet matching method from Chapter 3 on signals like in Figures 6.4a–6.4d afterwards we exemplify MODW (cf. Chapter 4) on signals like in Figure 6.4e or 6.4f.

6.2.1 Linear guideways and wavelet matching

For obtaining the signals in Figures 6.4a–6.4d the acoustic emission had been measured directly at the carriage. Our special intention is to detect two given pittings (cf. Appendix A.1) in the guideway, produced around 1/4 and 1/2 of the feed way. In signals measured at slow velocities, like in Figure 6.4a they are visible at the first sight without any further data processing. Meanwhile they are mostly masked by the background signal of the machinery for faster velocities.

The first task in the course of applying the results of Chapter 3 is how to obtain a pattern P for the matching from a set of signals $\mathcal{Y} = \{y_i\}_{i=1}^N$, measured at several

velocities. Therefore we chose the following sequence of well known filtering methods, resulting in an average P:

1. Most of the unwanted parts of the signal are arising from the sounds made by the ball bearings of the carriage. This have a well defined frequency just depending on the velocity of the carriage and the radius of the balls. In Figure 6.5, representing the GABOR transform of the signal from Figure 6.4b, it is well visible together with its harmonics. They are not at a constant scale, since the carriage has to accelerate and decelerate. So the first natural step is to compute $\{\tilde{y}_i\}$ by filtering this rolling frequencies from the y_i using FOURIER or GABOR techniques.

2. For minimizing the stochastic influence on the $\tilde{y}_i$ we took the average over all recordings with the same velocity and obtain some $\bar{\mathcal{Y}} = \{\bar{y}_i\}_{i=1}^{\bar{N}}$. Furthermore we computed y as average over rescaled versions of all $\bar{y}_i$ for having y independent of the feed velocity. The time-consuming point concerning this computations is to find coinciding points in the signal, for adjusting shifts of the signals, since the pattern P has to be at the same place in every $\tilde{y}_i$ and there is no marker of the position of the carriage. For this example, this had been done manually.

3. For filtering the the pattern from remaining noise we used a soft wavelet shrinkage (cf. [155]). For improving the results cycle spinning has been incorporated (cf. [24, 11]), offering a shift-invariant variant of the shrinkage. Additionally we repeated this procedure three times with three different wavelets (Coiflet 4, DAUBECHIES 8, HAAR wavelet) and took the average of the results $\bar{y} = (y_C + y_D + y_H)/3$. So we suppressed influences of the shape of the wavelet to the result.

4. Define P as a suitable time interval of $\bar{y}$, containing the critical pattern.

Some examples of resulting patterns P, indicating the responce of a small local pitting in an AE-signal, are depicted in Figure 6.6. Their difference is based on the different positions of the measuring device. A problem occuring in this special case is, the resulting patterns are more or less sine-shaped. So it is difficult to distinguish these patterns from periodic perturbations like the rolling of the balls in the bearing. Interestingly this patterns are extremely localized, apparently the microphone registrates just the interaction of the ball next to its position with the pitting. Just in a few cases an additional pattern of the same shape and with a trickle of the amplitude has been found preceding the main pattern.

We now compute the matching wavelet according to the P given in Figure 6.6a. Using Remark 3.5 and Corollary 3.4 we obtain, for different choices of the length S of the lifting filter s (8 or rather 16 elements) and of the starting quadruple (wavelets and scaling functions) σ, approximations like, e.g., the ψ in Figure 6.7a or 6.7b. Here σ is chosen as CDF wavelet with p equals $0, 1$ and 2 respectively (cf. Lemma 2.7). If we match a full set of phaselets instead (cf. Equation (3.10)), we obtain for $n = 3$ results like in Figure 6.7c or 6.7d (here $S = 8$ or rater $S = 10$).

Before we state some detection results by applying this wavelets, we need first to introduce a visual representation of the transform. In the case of just one wavelet ψ^λ there is a well known two dimensional representation of the coefficients

$$f_{k,j}^\lambda = 2^{-j/2} \int f(x)\psi^\lambda \left(\frac{x-k}{2^j}\right) \mathrm{d}x$$

(cf. also Equations (3.11) and (2.3)) according to a signal f of length $2^{-\bar{j}-1}$. We give here directly a version for the coefficients of a full phaselet family $\{\psi^\lambda\}_{\lambda=0}^{n-1}$: Every coefficient $\left|f_{k,j}^\lambda\right|$ is given as a color scaled value in a rectangle of size $1/(2^j\lambda) \times 1$ around the point (x, y) given via

$$x = \frac{k+1/\lambda}{2^j} \quad \text{and} \quad y = -j, \qquad j \in \{\bar{j}, \ldots, j'\} \quad \text{and} \quad k \in \{0, \ldots, 2^{-j}-1\},$$

So x is denoting the position and y the scale. Figure 6.8a shows an example of this type of representation. Since color scaled illustrations are often bad readable, we propose as an alternative second representation a measure function similar to a weighted local energy sum:

$$m\left(k/2^{1-\bar{j}}\right) = \sum_{j=\bar{j}}^{j'} 2^{-j/2} \left(\sum_{\lambda=0}^{n-1} f_{\lfloor 2^{\bar{j}-j-1}k \rfloor, j}^\lambda\right)^2, \tag{6.1}$$

here $\lfloor\cdot\rfloor$ denotes the GAUSS brackets. An example for m is given in Figure 6.8b, it illustrates perfectly the positions of the two artificial pittings in the guideway, as suggested but Equation (3.2)[3]. The additional horizontal line in the plot marks the value of $\theta = d_m + 7\sigma_m$ with d_m denoting the average of m and σ_m its standard deviation. We will see in the following examples, that θ is a good threshold for indicating errors. This as well as the following computations are made using a signal of 8192 points (4096 points in cases with higher velocity of the carriage) sampled at 2000 Hz. For the wavelet transform we defined $j' = 0$ and $\bar{j} = -12$ or rather $\bar{j} = -11$.

Of course the question arises, if the usage of this complicate framework of matched phaselets is really necessary for obtaining a valuable classification method. So we give in Figure 6.9 a comparison of the wavelet coefficients computed from the signal in figure 6.4b by the standard HAAR wavelet, the Symmlet 8, one matched wavelet (Figure 6.7c, blue line) and a family of three matched phaselets (Figure 6.7c). We observe clearly that just in the last case both pittings are separated significantly from the background. Please observe, there is no difference concerning the choice of θ, also other choices would not result in exact positions for the two pittings in the first three cases.

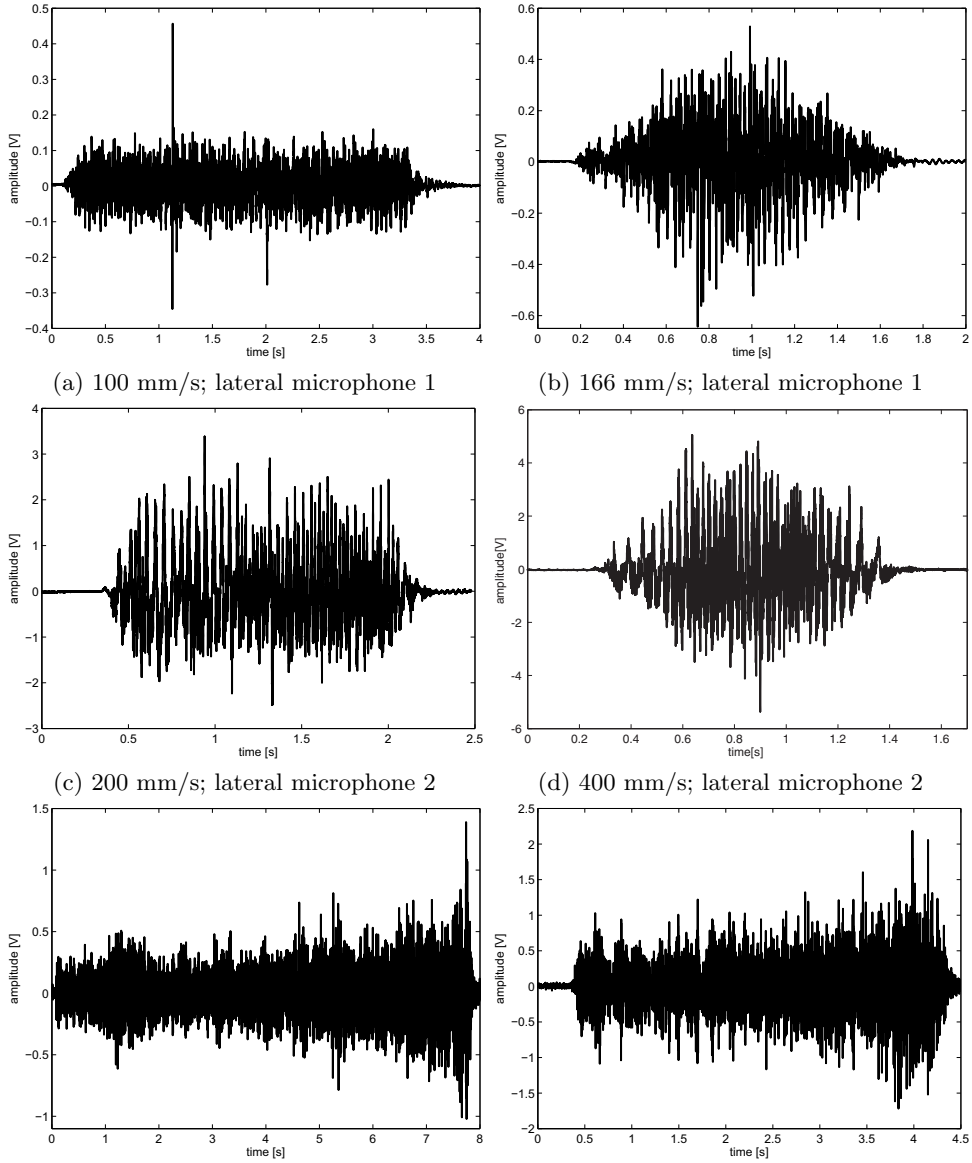
Another question is, if it is really necessary to apply the matched wavelets using the numerical robuster Equation (3.2), or if also the application of Equation (3.1) can be considered (cf. Section 3.1 for the differences). For Figure 6.10b we computed the

[3]We do not state exact positions throughout this analysis, since the starting point of the carriage is not fixed in the signal.

later one according to the signal in Figure 6.4a and applying the phaselet family in Figure 6.7c. For comparison we give on the left the original m function as already given in Figure 6.8b. We see, that there is almost no evidence concerning the right pitting in the right side figure and the left pitting is also not significantly detectable. This result also holds under changes in the computation of m (cf. Equation (6.1)), especially changing the weights or the power of the $f_{j,k}^{\lambda}$. So choosing the method based on Equation (3.2) is the correct decision.

Therewith we know about the usability of the wavelet matching method from Chapter 3. The last question arising is, how to choose the parameters given in Chapter 3 (see especially Proposition 3.2 and Equation (3.10)): n denoting the number of phaselets, p affecting their smoothness and the length S of s. From Equations (3.9) and (3.6) it is clear, that greater n and S give better detection results, while smaller values increase the computation time. The above experiments, as well as the following ones show, that $n = 3$ and $S = 16$ perform good results and can serve as a compromise between this two objectives. More interesting are the results concerning the choice of p. Till now we usually used the picewise constant wavelets resulting from $p = 0$. Interestingly they are as good as the smoother wavelets arising from using $p \in \{1, 2\}$ or even slightly better, as Figure 6.11 shows. There we plotted the m functions of the transforms using a phaselet family analog to that from Figure 6.7d with $S = 16$. The resulting figures can directly be compared to Figures 6.8b and 6.9h being similar or even slightly better.

We want to conclude this subsection by presenting four further examples with higher velocity of the carriage and the usual phaselet family from Figure 6.7c in Figures 6.12 and 6.13. While the first two examples are going analog to the previous ones, the later two ones are recorded at a very bad microphone position. So the amplitude of the rolling frequencies is extremely high. Since the P is looking similar to a sine, as mentioned above, arise problems in distinction and we have to filter the rolling frequency before wavelet transform, this filtered versions are given in Figures 6.13c and 6.13d. Nevertheless this is barely successful in the left side example (note the three peaks around $x = 0.8$), but in the right example there is one additional signal component, in the middle of the two pitting signals, whose signal exceeds by far the pittings. A closer analysis of the coefficients of the transform shows a disturbance of unknown source on the finest frequency at this point giving the only false alarm in all our experiments.



(a) 100 mm/s; lateral microphone 1

(b) 166 mm/s; lateral microphone 1

(c) 200 mm/s; lateral microphone 2

(d) 400 mm/s; lateral microphone 2

(e) 40 mm/s; face microphone

(f) 80 mm/s; face microphone

Figure 6.4: AE-Signals for different velocities and microphone positions

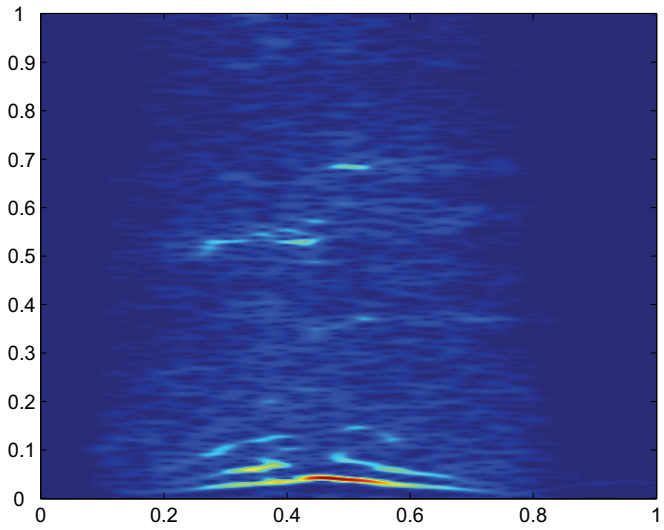


Figure 6.5: Discrete GABOR transform of the signal in Figure 6.4b

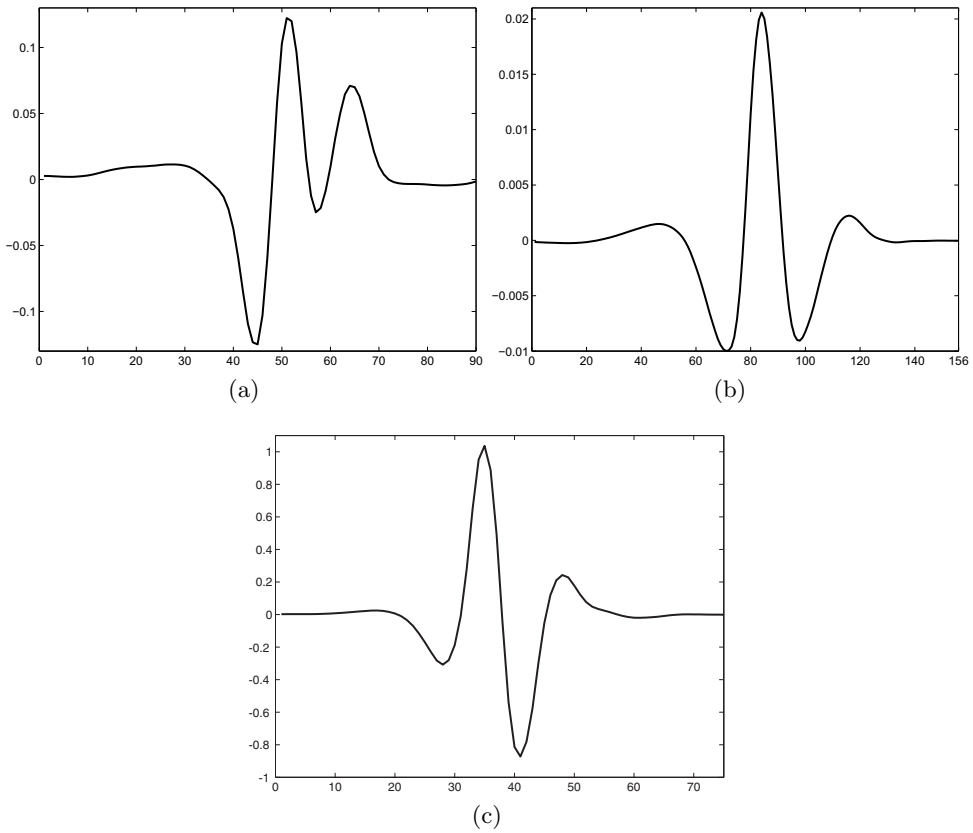


Figure 6.6: Pattern of a local pitting for different positions of the measurement device

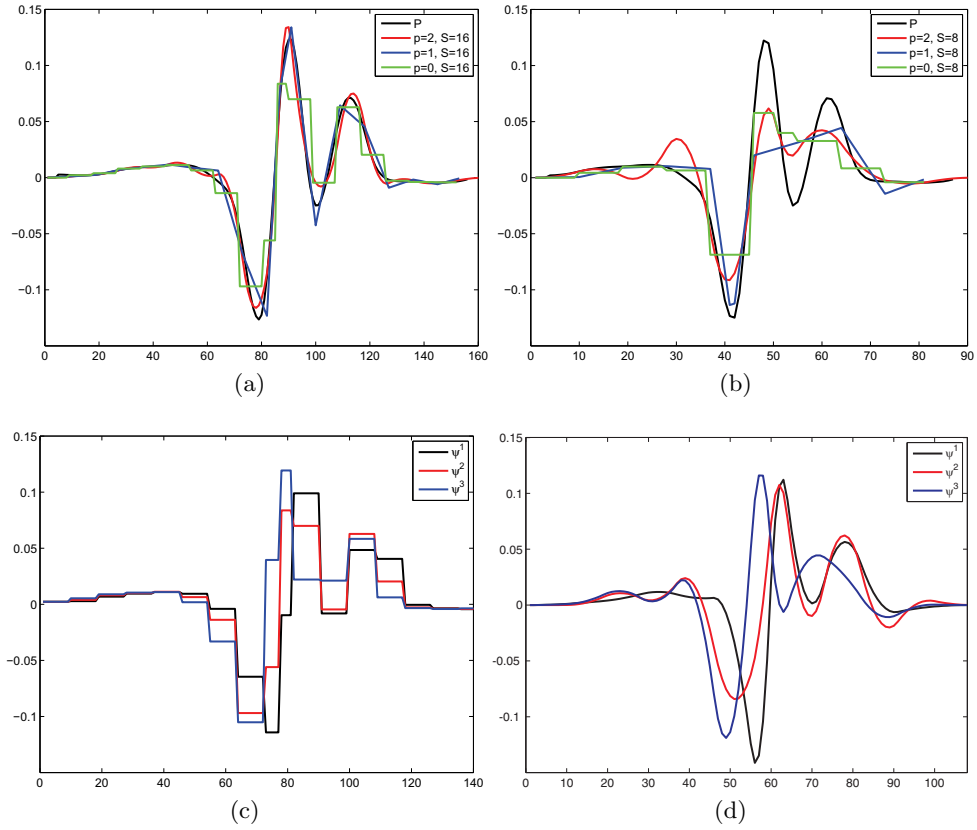


Figure 6.7: Single matched wavelets (above) and phaselet family (below)

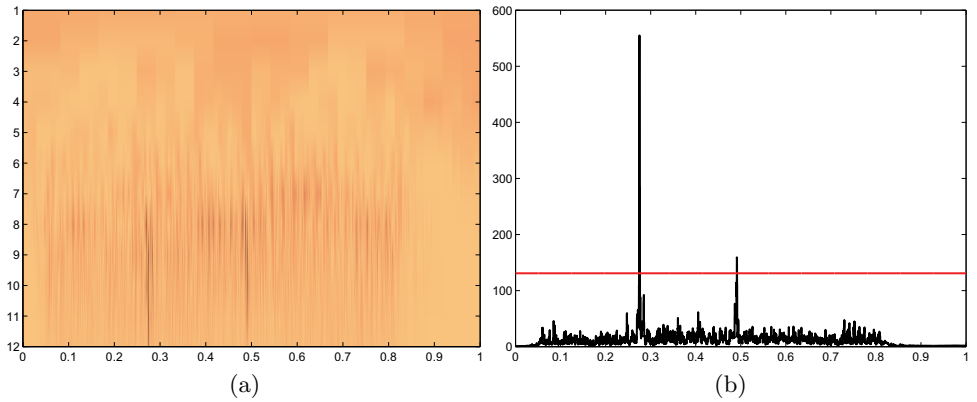
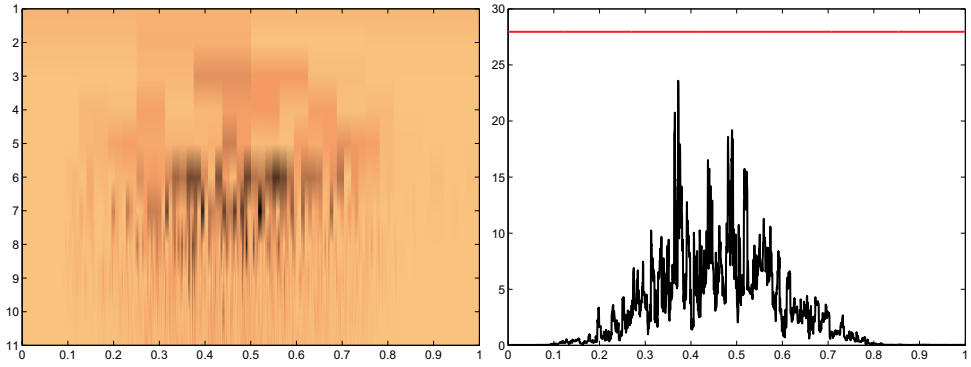
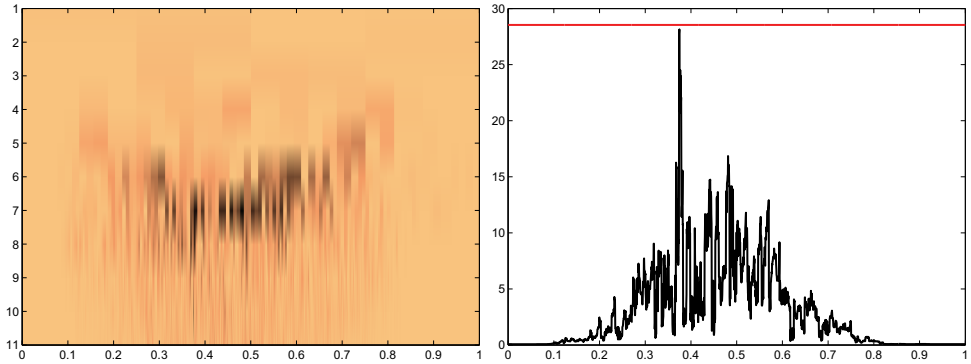


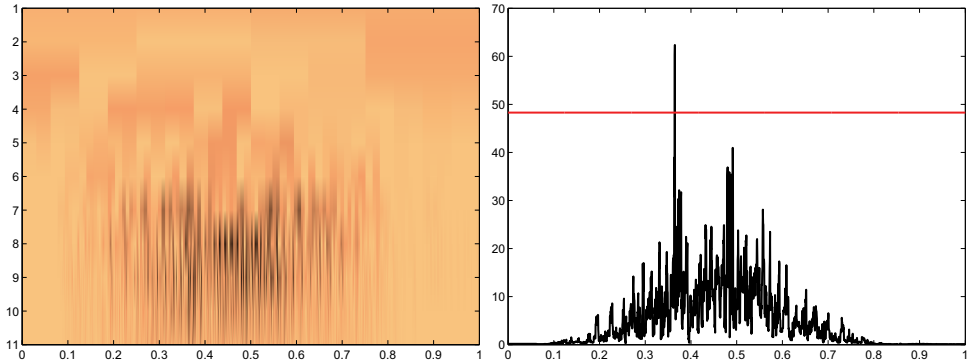
Figure 6.8: Matched phaselet transform of the signal from Figure 6.4a



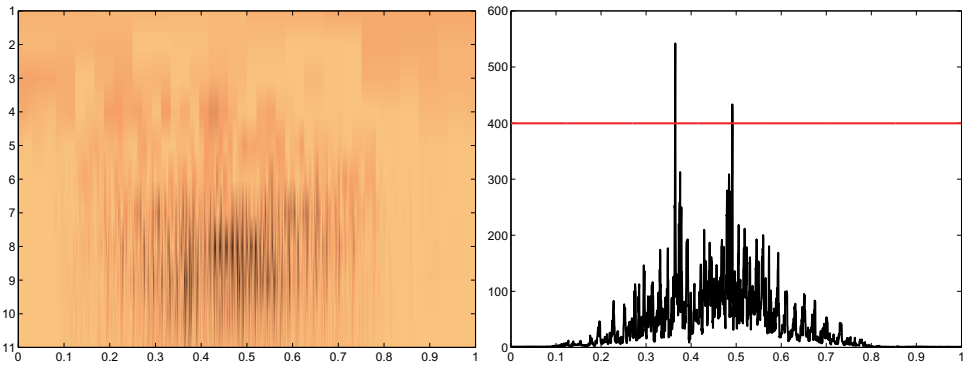
(a) Haar wavelet (b) Haar wavelet



(c) Symmlet 8 (d) Symmlet 8

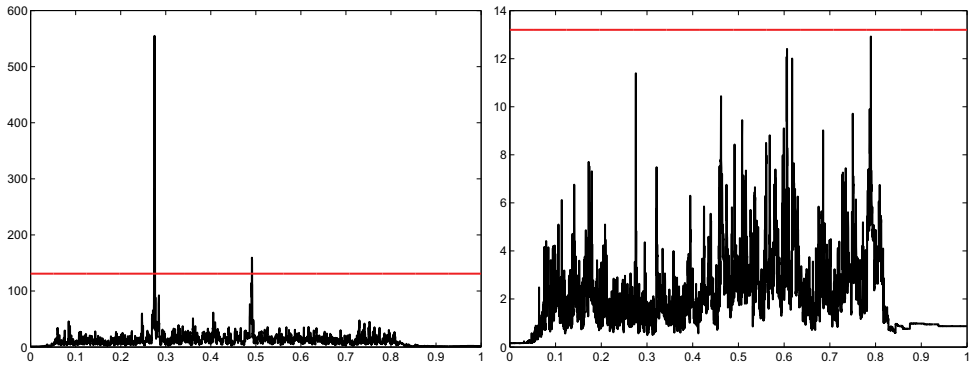


(e) Selflet (f) Selflet



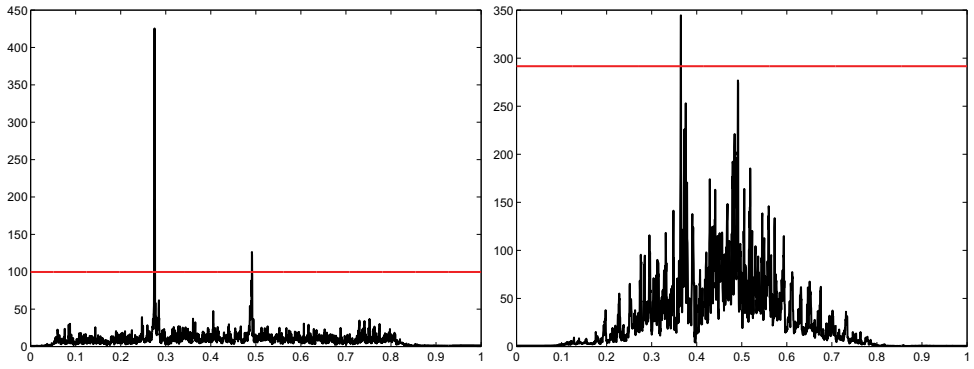
(g) Phase-Selflet (h) Phase-Selflet

Figure 6.9: Comparison of the pitting detection qualities for several wavelet and phaselet transforms



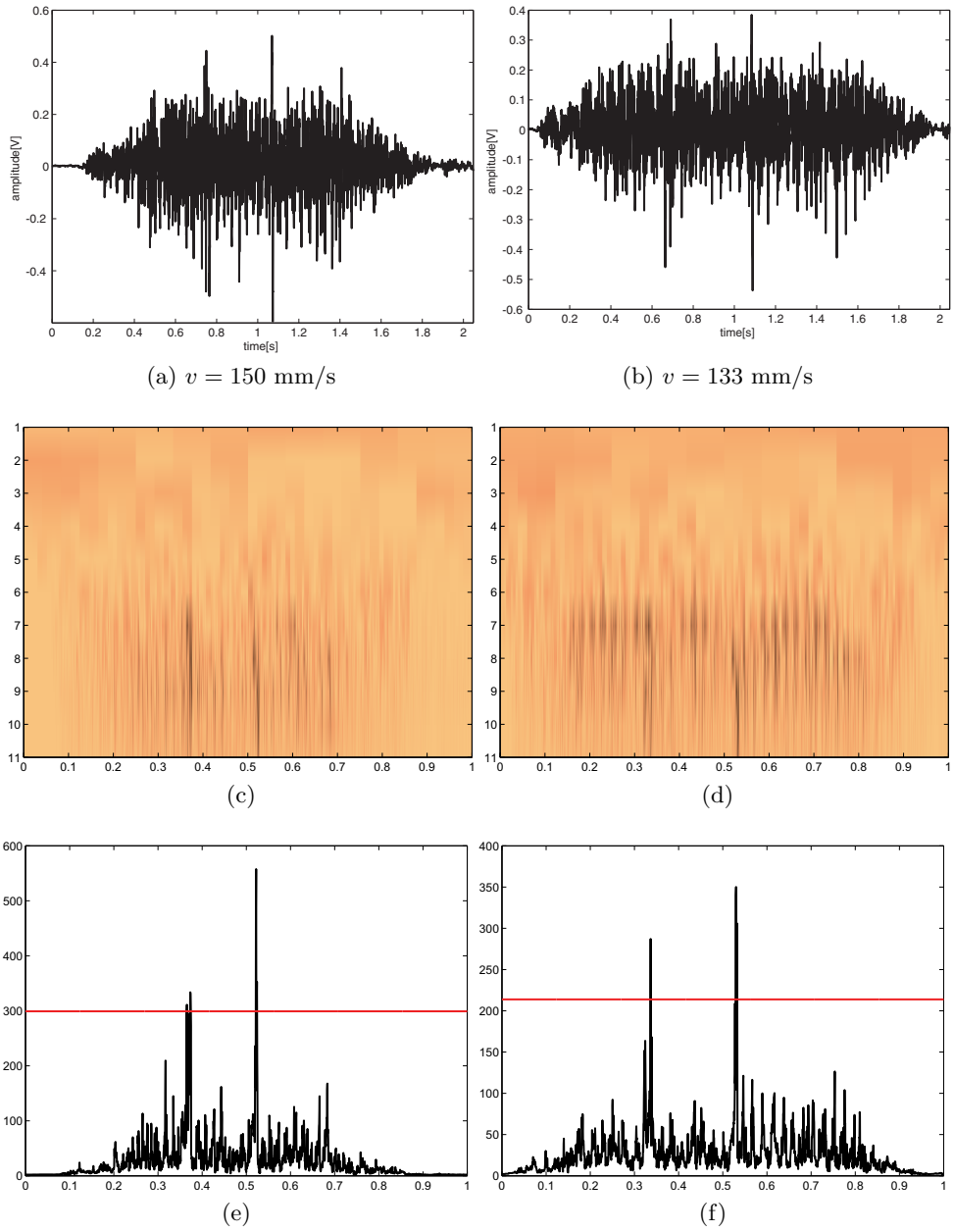
(a) Using Equation (3.2) (b) Using Equation (3.1)

Figure 6.10: Comparison of the pitting detection qualities for the different approaches of Section 3.1



(a) (b)

Figure 6.11: Transform of the signal from Figure 6.4a and 6.4b using the matched phaselets from Figure 6.7d

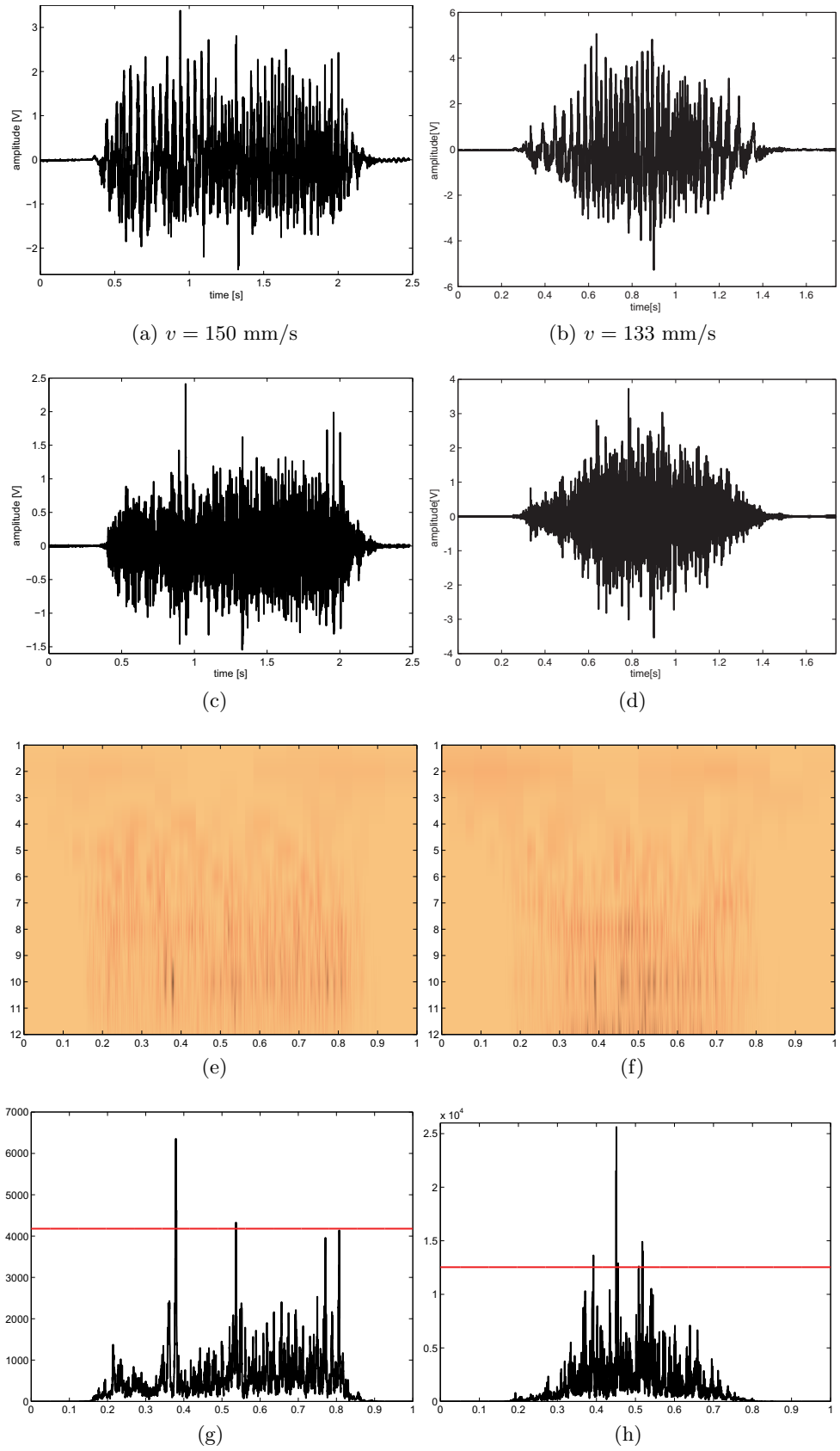


(a) $v = 150$ mm/s (b) $v = 133$ mm/s

(c) (d)

(e) (f)

Figure 6.12: Pitting detection in two further examples, using the matched phaselets from Figure 6.7c



(a) $v = 150$ mm/s (b) $v = 133$ mm/s

(c) (d)

(e) (f)

(g) (h)

Figure 6.13: Pitting detection in two filtered examples, using the matched phaselets from Figure 6.7c

6.2.2 Linear guideways and MODW

After finding a way to extract one special error out of measured signals, it is an interesting task, to separate the whole signal into its components. Doing this, we would be able to monitor the condition of guideway and carriage, what results in a method to find different types of errors in one analysis. The natural method to choose is, according to several of the references in Sections 1.3 and 2.2, a dictionary approach. Since we have already good results with matched wavelets, it is a natural idea, to use a dictionary consisting of wavelet bases like in Equation (4.1). Furthermore, there is no natural dictionary available, so this example is a predestinated for application of the MODW algorithm. In this subsection we want to present the procedure of applying MODW to this problem and present the results. First we use a version of the algorithm optimizing just one, fixed scale, afterwards we use all reasonable scales.

Fixed scale

For application of the MODW to linear guideways, we used acoustic measurements at the front face of a 30 cm long linear guideway with an artificial pitting almost in the center. This microphone position implies, pittings are no longer exactly localized, instead every ball of the bearing produces an own signal. Furthermore the balls are connected by a chain, reducing their degrees of freedom. Examples for the resulting signals are presented in Figures 6.4e and 6.4f, measured with a sampling rate of 22 kHz. Please notice the the amplitude of the signal rises, when the carriage approaches the microphone. The database consists of the signals from two runs of the carriage divided in non-overlapping pieces of 100 points. For holding the dictionary and especially the computation time small we restrict the database $\mathcal{Y} = \{y_i\}_{i=1}^N$ to one velocity and one driving direction of the carriage, this results in still $N = 3600$ signals. If we give up this restrictions, the total number of signals increases and computation time exceed a sensible range, since we can not restrict the number of signal per condition to strictly, according to the higher standard deviation of the resulting error for small N (cf. Subsection 6.1.1).

The MODW was initialized with the following parameters: Stopping at a relative error of 1% or after maximal 6 iterations. Using OMP (cf. Page 15), which chooses maximal A atoms, if a relative error of 1% is not reached before. Absurd lifting had been treated as local minima (cf. Page 50), at the boundaries the signals and wavelets had been truncated (cf. Page 48 et sq.) and the OMP had been forced to converge (cf. Equation (4.10)).

We computed dictionaries $\mathcal{D}^*$ in the cases of piecewise constant, piecewise affine linear and picewise quadratic polynomial wavelets ($p \in \{0, 1, 2\}$ in Equation (2.7)) for all the $A \in \{10, 20, 35, 50, 75\}$ and different velocities of the carriage. Out of this examples we present below results using $p = 2$ and $A = 10$ for signals measured at 40 mm/s and the carriage departing from the microphone.

As starting dictionary $\mathcal{D}_0$ we used a collection of the shifts of 25 wavelets on one, a priori fixed scale, all in all 1025 atoms. All this wavelets are computed using the

the method from Chapter 3 satisfying

$$\|\psi - \sin(k_1(\cdot - k_2\pi/3))\|_2 = \min$$

for arbitrary $k_1 \in \mathbb{N}$ and $k_2 \in \{0, 1, 2\}$. Naturally the smoothness of the starting wavelets equals that one used during MODW.

We used $\mathcal{D}^*$ for analyzing four signals, which are obtained the same way as the training signals and cut again into subsignals of 100 points. The signals themselves are stringed to one signal, presented in Figure 6.14, while M, the ℓ_2-norms of the subsignals, is given in Figure 6.15. In the following we want to highlight two results from this analysis:

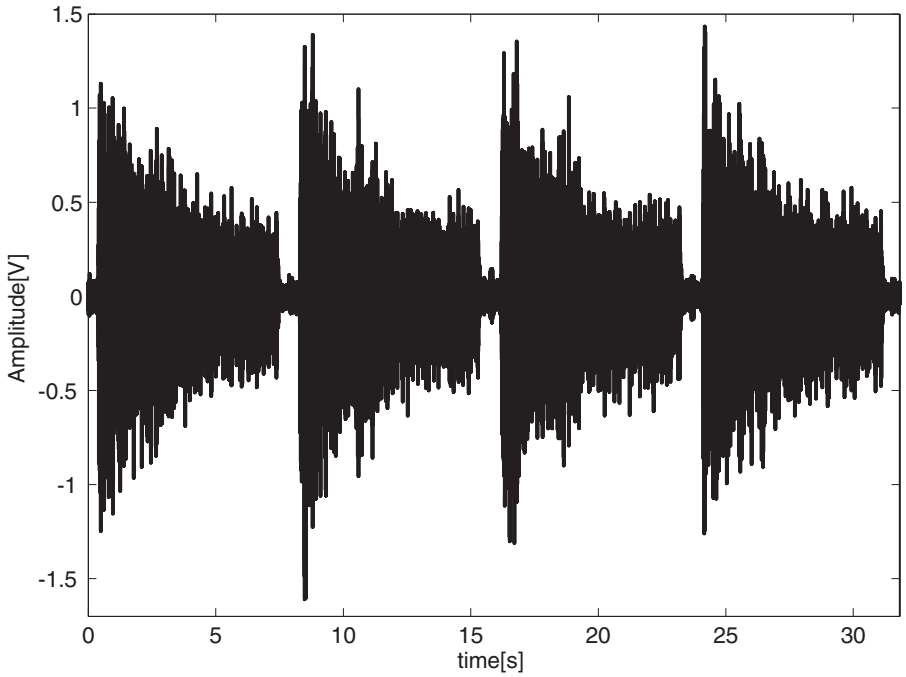


Figure 6.14: Example signals for analysis using $\mathcal{D}^*$

- The behaviour of the approximation error.
- The correlation of the different wavelets to the different parts of the signal.

Figure 6.16 is given for explaining the first mentioned point. It shows the ℓ_2-norm of the approximation error for the 100 point subsignals, the average relative error is 7%. Clearly visible is a high peak after approximately one third of every signal. The position of this peak is almost constant, it varies between the 504th and 508th subsignal following the estimated start of the signal of the run [4]. The same effect is

[4]Since we have no trigger, indicating the start of the carriage, we fixed a starting point of the signal manually, comparing the subsignals in the beginning of the four runs, but we have no connection of this points with the start of the carriage and its position on the profile rail.

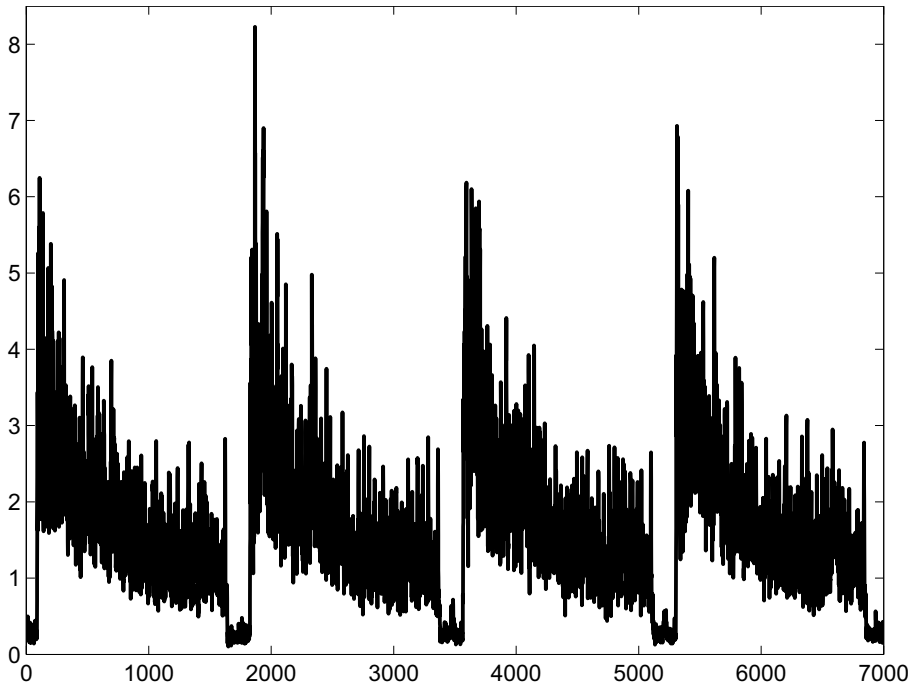


Figure 6.15: Norms M of the 100 point subsignals of the signals from Figure 6.14

in weaker form also visible in the case the carriage approaches the microphone, but in that case the position is between 358 and 359 (cf. Figure 6.17). The difference equals to approximately 0.68 s or (at 40 mm/s) 2.73 cm. This are 8.5% (in time) of the total signal length and around the length of the stopping time between two runs of the carriage. So there seem to be no connection between the peaks at different driving directions, as they also mark total different positions on the guideway. Hence their mechanical background has to stay open.

For observing the distribution of the wavelets we define the measure

$$m_{j,i} = \sum_{r_1} |f_{j,r_1,i}| \,, \text{ where } y_i = \sum_{r_1,j} f_{j,r_1,i}\psi_j(\cdot - r_1) + e_i \,, \tag{6.2}$$

as the sum over the absolute values of the dictionary coefficients. For better recognizability we will give in the following figure the renormed average defined as

$$\tilde{m}_j = \frac{m_j * \mathbb{1}}{M * \mathbb{1}},$$

where $\mathbb{1}$ denotes a 100 dimensional vector, where every element equals one[5]. So we have on the one side a signal that is smoother and independent from the changing amplitudes of the original signal. On the other side the positions of peaks blur out.

Observing $\tilde{m}_j$ for all $j \in \{1, \ldots, 25\}$ we can catergorize the associated wavelets to different classes: Some have great coefficients if the carriage accelerate, some at deceleration, other ones at constant speed. All in all we found eight classes of

[5]We prefered $\tilde{m}_j$ over $(m_j/M) * \mathbb{1}$, since it is more stable.

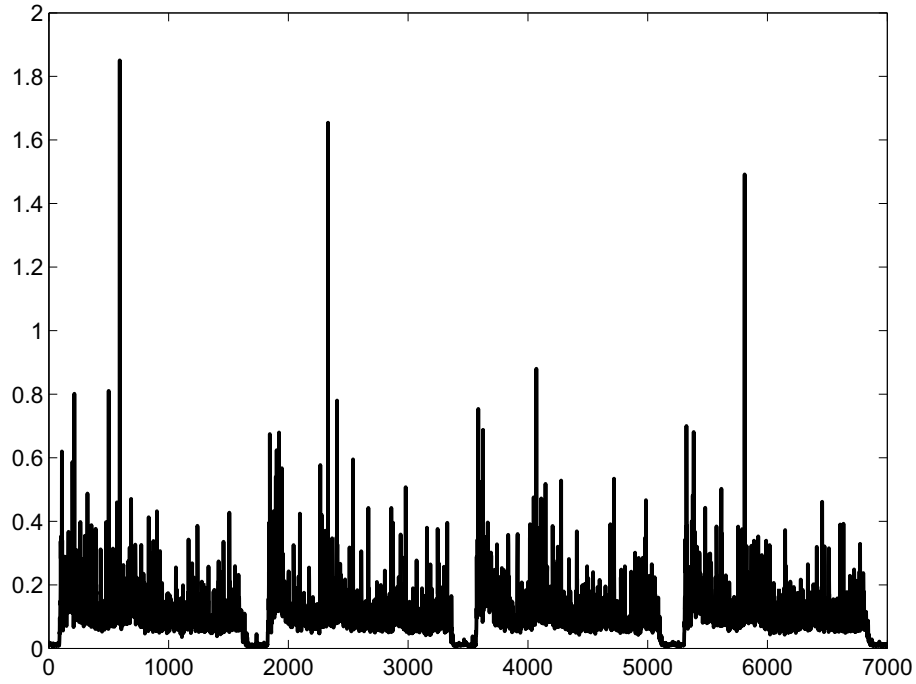


Figure 6.16: Approximation error of the signals from Figure 6.14 displayed as norms of the 100 point subsignals

wavelets, some with just two elements, one with six ones. It is not expectable to obtain just one element in every class, according to the not given shift-invariance of the wavelets (cf. Section 3.3) and since our dictionary $\mathcal{D}^*$ is in general just locally optimal. The $\tilde{m}_j$ are given in Figure 6.18 sorted by classes and labeled with the number of the wavelet. Here denote the vertical red lines the stopping times of the carriage. The separation is not perfect, as for example the elements of the class in Figure 6.18d show, which have peaks in the accelerating and the decelerating phase. Maybe the signals are to similar for distinction, but more probable, this is a further evidence for the local optimal nature of our dictionary.

The existence of starting and stopping signals is almost clear (cf. Figure 6.18a or rather 6.18g), maybe the Wavelet 23 in the first one can in a further development also be used for recognition of sealing problems at the carriage, as it coefficients become great already at start of the carriage. Also the existence of acceleration and deceleration classes (see Figure 6.18d) is clear. Nevertheless, this both classes are not separated from each other and are in the deceleration part also connected to the underlying signal class (Figure 6.18c). Furthermore this class also arises, with oscilations, in the deceleration part and, limited, also in the acceleration part. So its elements seem to form at least partly the background noise of the machinery. The reason for the oscillations in the last third of every signal of this class is not quite clear, maybe they are additional artifacts of a non-perfect distinction from the deceleration class.

Most interesting are the two middle classes (Figures 6.18e and 6.18f): The peak of the first one coincides almost with the peak in the approximation error mentioned

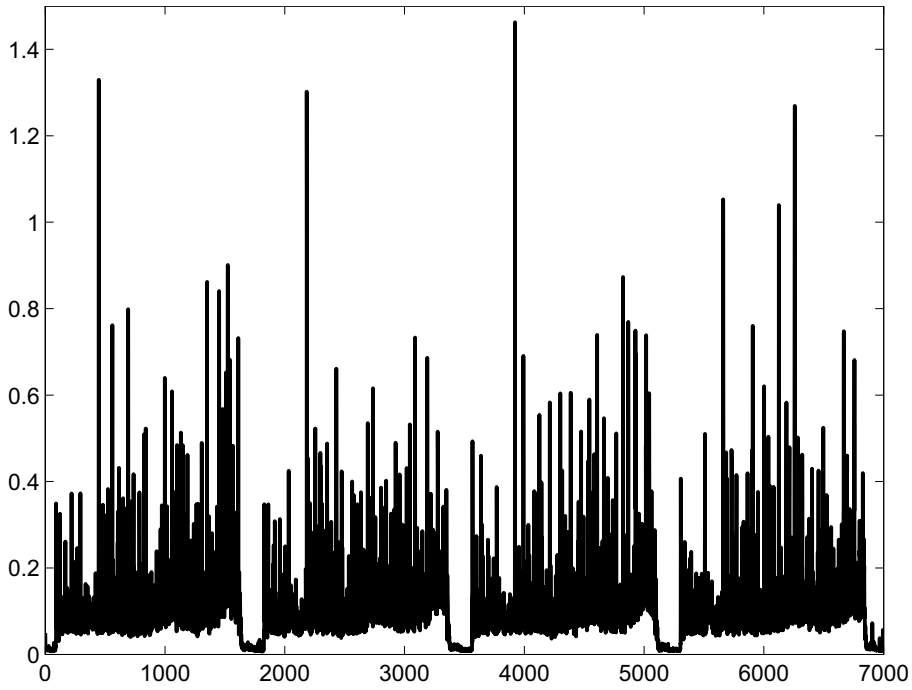


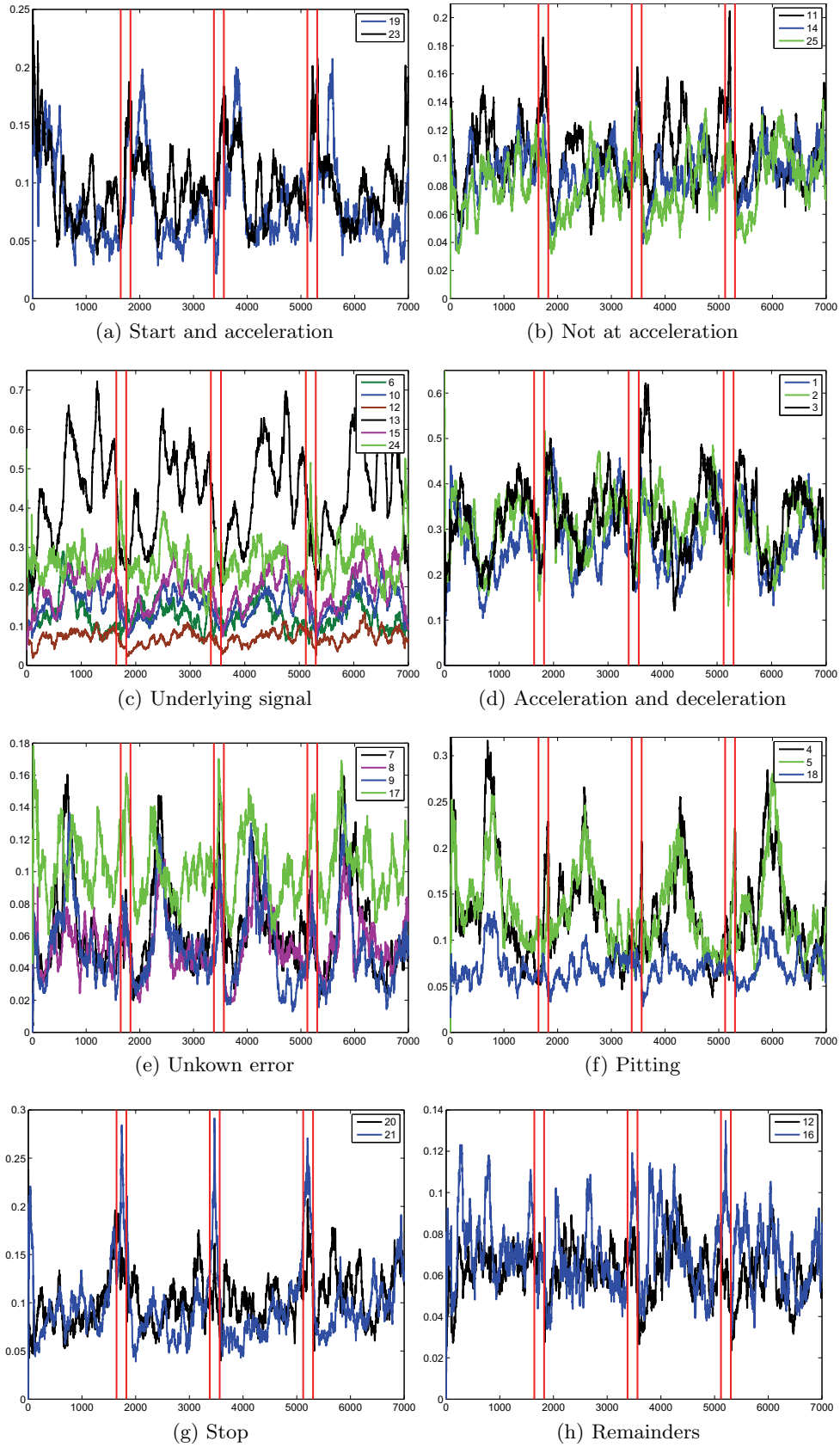
Figure 6.17: Approximation error of analog signals to Figure 6.14 for reverse direction of the carriage, displayed as norms of the 100 point subsignals

above, it follows it directly or rather meets it in the case of Wavelet 17. Probably it arises from the same source. Lastly the second class (cf. Figure 6.18f) has it peaks at the position of the artificial pitting in the guideway mentioned in the beginning. Especially in the case of Wavelet 5 the peak is also thin enough to serve as a basis for pitting detection. Nevertheless, for a quantitave analysis of the exact positions of the error, we need to adapt a method for fixing the center of the peak, and furthermore, consider the convolution and the not exactly known connection between time and the position of the carriage. Thus we give here just qualitative results.

So the restricted version of MODW is able to obtain similar results like the wavelet matching in Subsection 6.2.1. Even more, this task is more comlicate, due to the non-constant amplitude of the signal, the higher distance between pitting and microphone, the greater part of the signal covered by the pitting and the chain controlling the ball bearing. On the other side, the carriage was much slower than in Subsection 6.2.1. Furthermore, we obtain a complete decomposition of the signal, not only restricted to the detection of one feature. Nevertheless, there is still potential for improvement, as the imperfect discrimination of acceleration and deceleration shows. Here arises the necessity to find a method for avoiding local minima.

Free scale

In this subsection we want to use the full potential of MODW and release the restriction on the wavelet scales. So the OMP is free to choose the wavelets on every scale as denoted in Equation (4.1) and the MODW connects the coefficients



(a) Start and acceleration (b) Not at acceleration

(c) Underlying signal (d) Acceleration and deceleration

(e) Unkown error (f) Pitting

(g) Stop (h) Remainders

Figure 6.18: Absolute values of dictionary coefficients ordered by classes of the dictionary, displayed as norms of the 100 point subsignals

of the scales following fully the results given in Section 4.1. We want to underline, that there is no necessity for obtaining better results than on Pages 116–120. Since we work with one velocity of the carriage (see Page 116), there is no evidence of signals arising on different scales. So the above example maybe uses, beside the simplification, a priori knowledge as mentioned in Subsection 4.3. Nevertheless we add this example for reason of completeness and comparison.

For obtaining the dictionary as well as for the following analysis we use the same signals as in the above example (cf. also Figure 6.14). But in difference to that, the OMP is based on the fast wavelet decomposition. So we need to separate the signals in 1406 subsignals $\tilde{y}_i$ of length 2^J, here we chose 256. Furthermore, since every wavelet provides several scales, we decrease the number of wavelets to 12, resulting in 3072 atoms per subsignal (cf. for one scale we had 1025 atoms for a subsignal of length 100). Additionally we chose the starting dictionary $\mathcal{D}_0$ using the start-value-finder proposed in Algorithm 4.3, giving us the opportunity to avoid the local minima connected to the largest errors. This is absolutely necessary, since initialization with wavelets which are approximating sine similar shapes, as in the preceding example, results after MODW in a dictionary connected to an approximation error that is 2% higher than the starting error for this approximation.

We present here results analog to that one on the preceding pages. Figure 6.19 represents parallel to Figure 6.16 the approximation error for the signals given in Figure 6.14. Analog to Figure 6.16 we plot the ℓ_2-norms over the subsignal of 256 samples. For balancing the difference in length of the subsignals, we divided it by 2.56. Again we recognize the peaks of the unknown error in constant distance from the start of the signal. But in comparison the error is not as great as above, the main part part of it seems to be coded in the approximation. However, the approximation of the whole signal is worse than before, the relative error is 23% compared to 7% with one fixed scale. The major reason for this difference is the approximation of indifferent long signals with the same number of atoms that have furthermore a small support. One additional explanation for the higher error is the already mentioned interpretation of fixed scale applications as use of priori knowledge. Lastly also the question arises, if the basic assumption of the starting dictionary finder (Algorithm 4.3) is valid. Since we have in Figure 6.19 several outliers, the approximation of that signals would (even if it lessens the error) downgrade the performance of the dictionary. Nevertheless, this approximation is still practicable, since the error appears mostly on high frequencies, as exemplified in Figure 6.20.

Nevertheless, this definition of the starting dictionary $\mathcal{D}_0$ results later in excelent classification properties of the wavelets of the resulting dictionary $\mathcal{D}^*$. Analog to Equation (6.2) we chose the measure

$$m^{\xi}_{j,r_2,i} = \sum_{r_1} |f_{j,r_1,r_2,i}|^{\xi} \ , \text{ where } \tilde{y}_i = \sum_{r_1,j} f_{j,r_1,i}\psi_j\left(\frac{\cdot - r_1}{r_2}\right) + e_i \tag{6.3}$$

using the notations of Equation (4.1). Again we display below (here m_{j,r_2} denotes the sequence $(m^1_{j,r_2,i})_{i=1}^N$) the renormed average

$$\tilde{m}_{j,r_2} = \frac{m_{j,r_2} * \mathbb{1}'}{M * \mathbb{1}'},$$

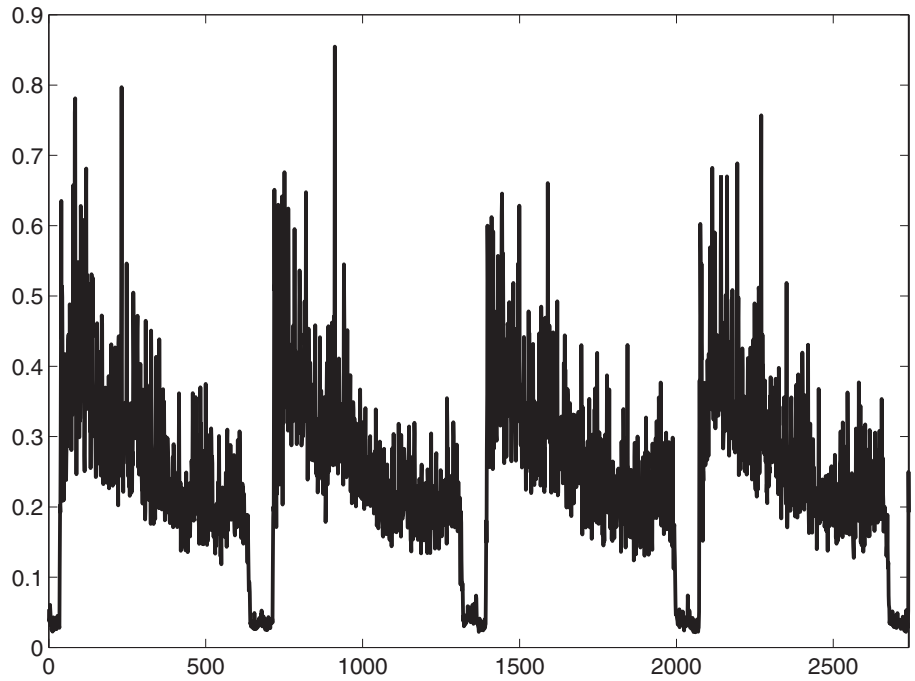


Figure 6.19: Approximation error of the signals from Figure 6.14 displayed as norms over subsignals of length 256, divided by 2.56

here $\mathbb{1}'$ denotes a 50 dimensional vector, where every element equals one. We established the different length of $\mathbb{1}'$, according to the different length of $\tilde{y}_i$. Since there are now $J = 12$ wavelets on 8 scales and additionally the scaling function Φ (equal for all 12 wavelets), there are 97 different $\tilde{m}_{j,r_2}$. Figure 6.21 shows the distribution of $\mu = \sum_{i=1}^{N} m^0_{j,r_2,i}$, denoting the use of the wavelets on different scales. The last (empty) column indicates the not used scaling function φ. Since there are large variations in μ, we just classified the wavelets and scales, indicating the largest 20 values in μ, that one exceeding the red line. The associated $\tilde{m}_{j,r_2}$ are presented in Figure 6.22.

In parts this classification is better than that one in Figure 6.18: Especially the distinction between the acceleration- (cf. Figures 6.22a and 6.22b) and deceleration-phase (cf. Figure 6.22c) is better recognizable. The deceleration-phase can additionally splitted in two parts: The actual deceleration and the stopping, denoted by the Wavelets 19 and 17. Interestingly we found three classes of wavelets having peaks between acceleration and deceleration. The first one (cf. Figure 6.22e) is localized in the same position as the first middle class in the preceding example (cf. Figure 6.18e) and the peaks in the approximation error (Figures 6.19 and 6.16) notifying the mentioned unknown error. Furthermore, the second class (cf. Figure 6.22f) indicates, with relatively narrow peaks, the same position of the pitting as the class in Figure 6.18f. Lastly, we obtain a third class in Figure 6.22g, which denotes with small amplitude a position on the right side of the pitting. Since it is arising at every passage of the carriage, there is a strong evidence for some mechanical reason, even if its nature is not clear.

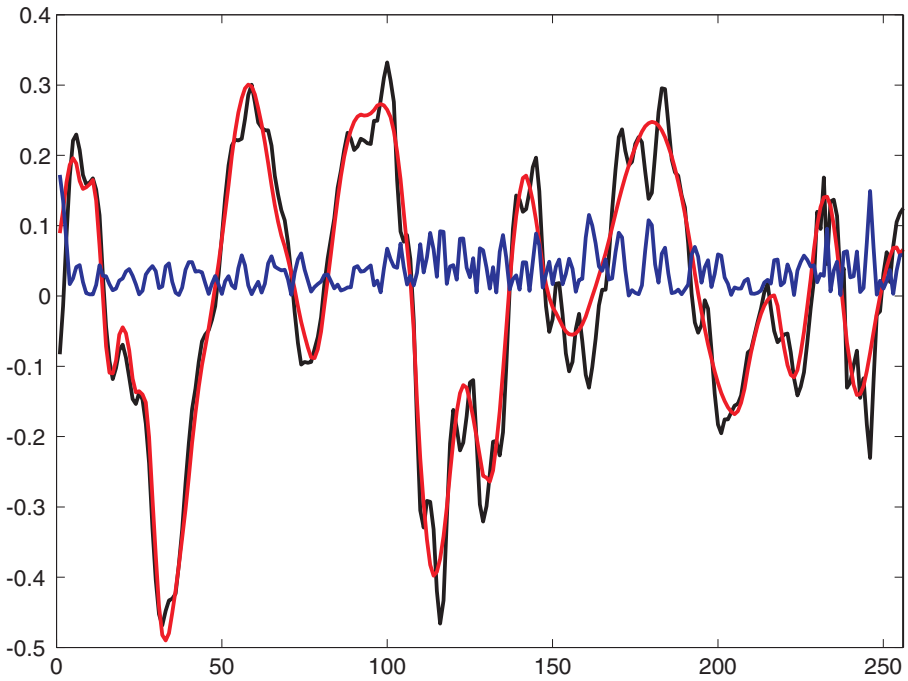


Figure 6.20: Approximation of signals during dictionary learning, exemplified on $\tilde{y}_{1000}$ (relative approximation error 24.2%). The black line indicates the signal, the red one the approximation and the blue one their absolute distance.

Altogether, a dictionary constructed by MODW, especially in case we use all scales, offers a good method for monitoring the condition of the linear guideway. Beside acceleration and deceleration phases also pittings in the guideway are detected. Furthermore, it gives evidence for two other mechanical properties, whose nature is not clear till now. In an additional step it would be indicated, to compute a dictionary optimized to varying velocities of the carriage and several predefined conditions of the linear guideway (as, e.g., different states of lubrication). In the later case, it would maybe be advisable, to compute dictionaries for every condition first and combine them later, for avoiding local minima in MODW.
Furthermore, MODW or rather the following analysis can be adapted twice to this special problem or close relative ones: First, cutting the signals to $\mathcal{Y}$ has the disadvantage, that short signals of interest are maybe not only to rare, but also cut into two pieces, distributed to $y_{\tilde{i}}$ and $y_{\tilde{i}+1}$. In the analysis of the signal will both signals give some detection result, but both are not necessary meeting a predefined threshold, what an unseparated signal would do. To reduce this problem, producing overlapping y_i would be necessary, but this increases the amount of data and we found no evidence that this problem occurs in our analysis. A second way for achieving better results could be non-negative coding, i.e. to redefine the coding coefficients that $\tilde{f}_{l,i} = \max\{0, f_{l,i}\}$. Applications for this approach are, e.g., given in [73, 2, 60, 78]. It is motivated in the fact that the machinery behaves always equal and the resulting noise has also the same algebraic sign. Nevertheless, we resigned here to do so, since there is no necessity for it, and several wavelets are oscillations, where the negative versions approximates a shifted one.

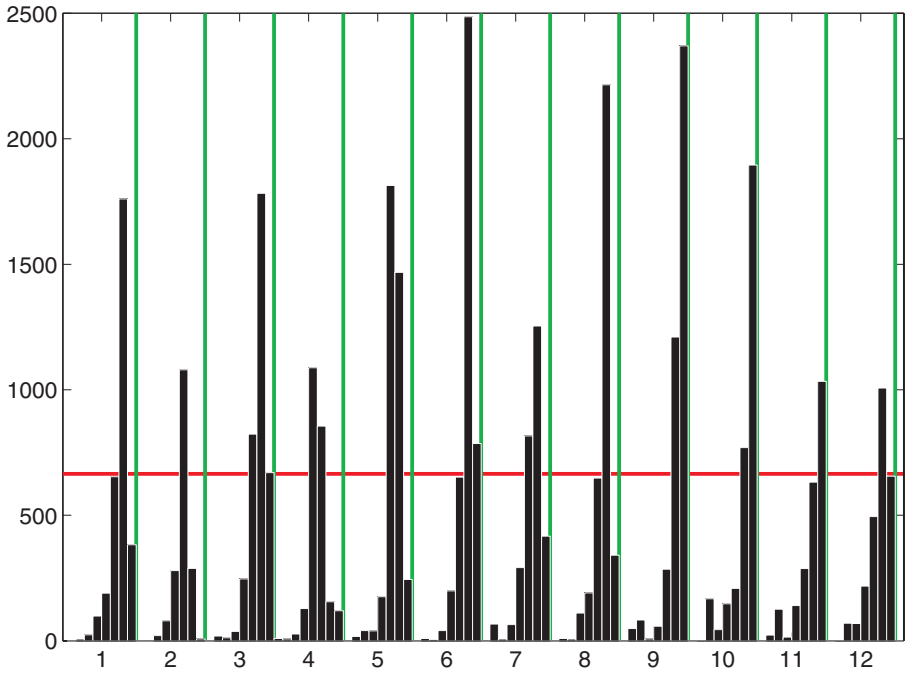
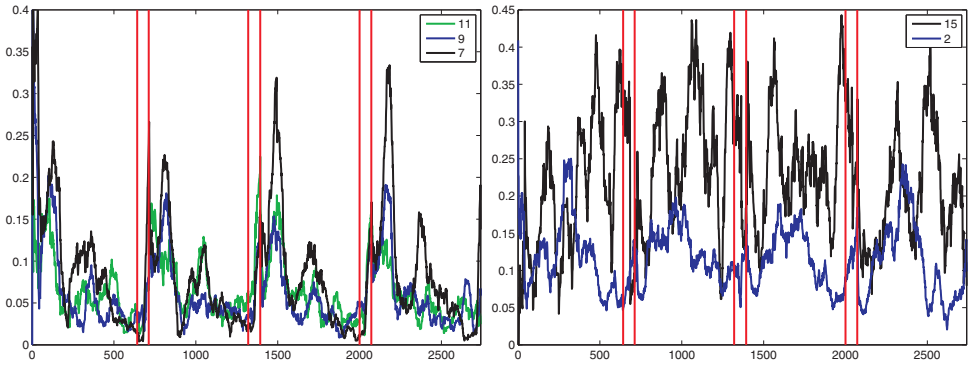
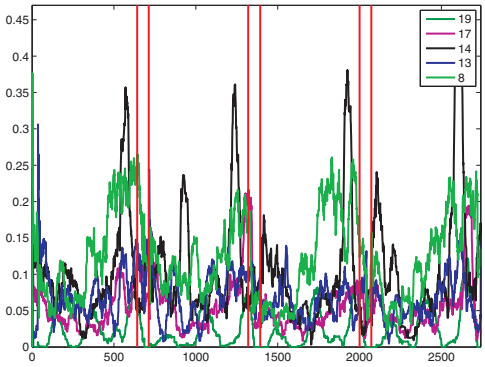


Figure 6.21: μ, describing the usage of wavelets on different scales for sparse coding. The vertical lines separate the groups of different scales of one wavelet.

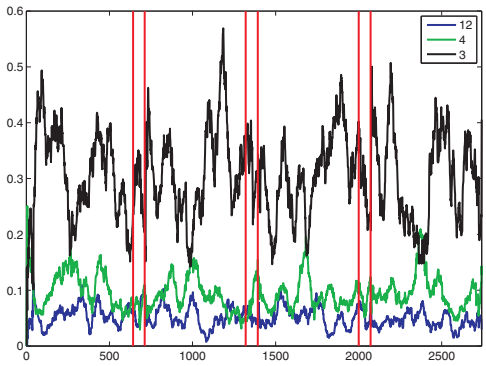


(a) Start and acceleration

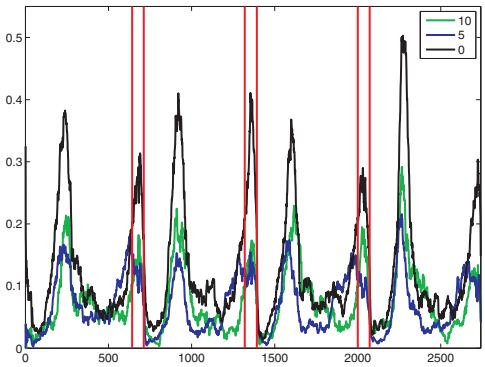
(b) Not at acceleration



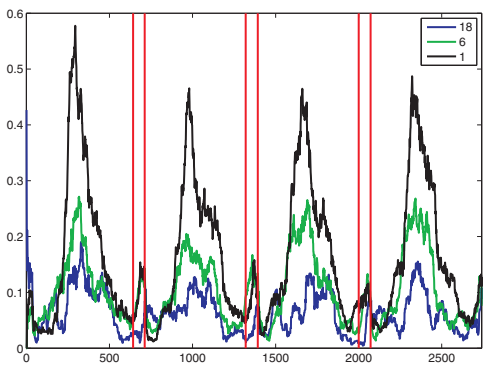
(c) Deceleration and stopping



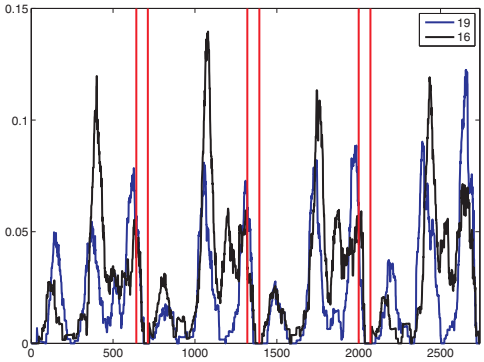
(d) Underlying signal



(e) Unkown error



(f) Pitting



(g) Second unknown source

Figure 6.22: Absolute values of dictionary coefficients sorted by classes of the dictionary, displayed as norms of the subsignals $\tilde{y}_i$

6.3 Transcription of piano music

6.3.1 Introduction and description of the problem

As a second application of MODW we propose here the reconstruction of the score of music based on a recording, this problem is also known as transcription of the recording. Piano recordings became a standard example for this task, due to its popularity, the wide field of existing compositions, the covered frequency domain and the easy availability of polyphonic music. But also the clean tuned notes are an advantage in comparison with, e.g., stringed bowed instruments. For solving this problem, there had been several approaches during the past years. While for monophonic music (just one note played at one time) there are successful solutions as, e.g., in [19], the situation becomes more complicate in the case of polyphonic music. Several techniques had already been proposed, maybe beginning with [115] in 1975, newer ones are, e.g., adaptive oscillators (cf. [107]), blackboard systems (see [108]), spectral smoothing (e.g., in [90]) or Bayesian inference (cf. [86]). Lately also approaches using support vector machines came up (cf. [174, 130]). See also [12, 129, 89] for further introduction to this problem and its history.

To use sparse coding techniques for music transcription seems to be a natural approach, since each key of the keyboard gives a well defined sound and the number of simultaneous played keys is in general limited to the pianist's number of fingers[6]. Furthermore, the different sounds superpose in first approximation linearly. Some approaches using sparsity or related concepts in musics, as, e.g., [1, 93], passed to the spectral domain, since for an application in time domain shift-invariance is necessary. Nevertheless, those approaches are less sensitive to the exact position of a note.

In [12], as well as in [129] Blumensath or rather Blumensath et al. firstly proposed shift-invariant sparse coding approaches to the music transcription problem in time domain. Their dictionary learning algorithms are maximum-likelihood-based (cf. Page 17) and propose for comparison to [1, 93] also a version, shift-invariant in frequency domain.

Wavelets are maybe not the first choice for the decomposition of music, often, e.g., in [64, 59], Gabor atoms are chosen, but there are also applications of wavelet packets, as, e.g., in [125]. In applying MODW we obtain here one main advantage: In addition to the fast wavelet transform, the scale-invariance of the wavelets corresponds to the different octaves of the musical scale. So we expect basically just $J = 12$ different wavelets ψ_j in a dictionary as defined in Equation (4.1), corresponding to the 12 notes from c to b. Of course, there occure additional variants, depending on the specific piano and pianist or the use of pedals. Below we see also, the tone at action differs from that one during fading phase and furthermore the noise of the recording is added. For minimizing the number of parameters, we restrict our analysis to one recording of one composition, here we used Ludwig van Beethoven's Sonata

[6] There is a legend, that Mozart won a bet against Haydn by playing an additional key with his nose, but that does not change the situation in general

for piano, Number 12, in a♭ major, Opus 26, second movement (Scherzo and Trio). This is one of the composition also used in [12, Section 8.2]. In our case we chose a 1962 recording by Claudio Arrau with a total length of 155 s, as plotted in Figure 6.23. Furthermore, we restrict ourselves to the learning of 12 wavelets, not taking into account the mentioned possible variations of the tones.

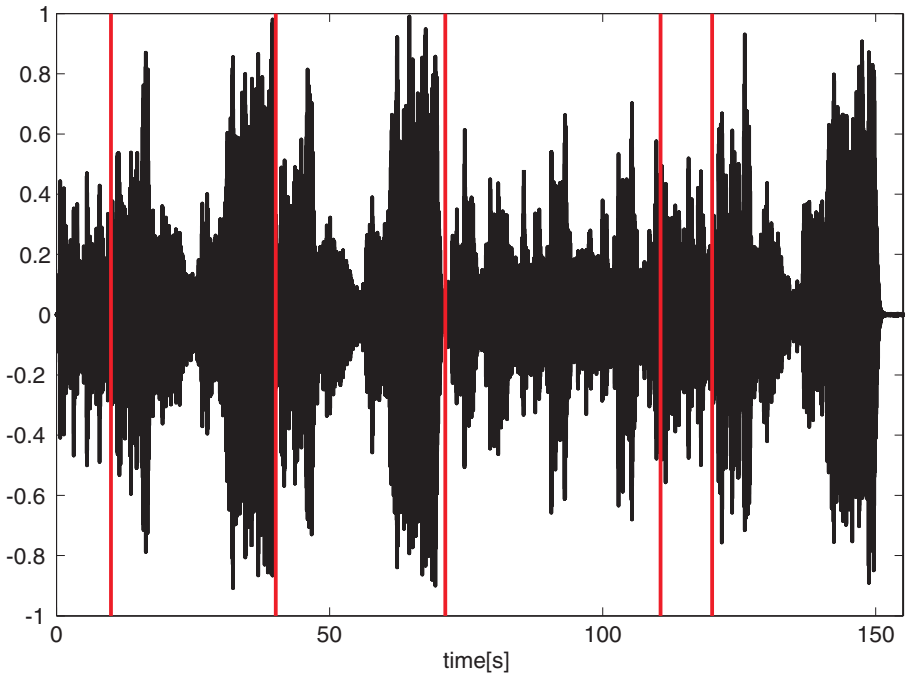


Figure 6.23: Recording of Beethoven's Sonata 12, Opus 26, sampling rate: 8000 Hz. The vertical, red lines separate the parts of the composition: First part of the Scherzo, two repetitions of the Scherzo's second part, the Trio and again the two parts of the Scherzo without repetition.

As signal we define one of the two stereo channels and resample it (for reasons of comparison with [12]) to 8000 Hz. Similar to Subsection 6.2.2 we divide this signal into 4844 subsignals y_i of length 256. The further parameters of the MODW algorithm (cf. Algorithm 4.1) are defined as: Generating a dictionary with 12 piecewise quadratic spline wavelets (cf. Equation (2.7)), based on a starting dictionary $\mathcal{D}_0$ with 20 lifting coefficients, obtained by the start-value finder described in Algorithm 4.3. The sparse coding is, for reasons of computation time, obtained by OMP, choosing for every subsignal 10 atoms. The MODW stops after 12 iterations or if the relative change of error is less than 10^{-5}. Furthermore, we do not use methods for prevention of absurd lifting, since no numerical problem occured, but force the algorithm to converge (cf. Section 4.2).

In comparison with the starting values used in [12], that ones here are possibly disadvantageous: Since the mostly used atoms are especially chosen from the finer scales of the wavelets, and the atom's support does not cover the whole length of y_i, more than the 10 atoms chosen by OMP would maybe be necessary for a good approximation. In addition, also the number of 12 wavelet bases for the dictionary

is maybe to small, as mentioned above. Nevertheless, both reduces the computation time to 28.5 hours[7](compared with 6 days in [12]) and we obtain, as we see below, still solid results.

6.3.2 Results of MODW on piano music

The resulting wavelets

The analysis of the results divides into two parts: Properties of the resulting wavelets and comparing the sparse coding with the original score. We want to start with the first one: In Figure 6.24 we plotted representatives of the resulting 12 wavelets on the second-finest scale. Almost all show a harmonic structure[8], maybe fewest distinctive in the case of Wavelet 6. If we look at the fast FOURIER transforms of this functions in Figure 6.25, we see, that most of them have a clear, single maximum at one frequency. Exceptions are the Wavelets 5 and 6, which both have two peaks of almost equal heigth. As this wavelets seem to represent a chord instead of a single note, we separate them for the further analysis from the other 10 ones and have later on a closer look at them. Also other wavelets, especially Number 1, 3 and 7 have additional peaks in their transform, or a wide support like Number 9, but we will see below, that their results in reconstructing the score are still reliable. Nevertheless, such shapes indicate, that we obtained, as in general, just a locally optimal dictionary.

As an additional effect we observe, there are in general just negligible influences of higher frequency harmonics. This is clear, since the construction and tuning of a piano does not fix a single phase shift between the harmonics. This effect is based on a necessity of piano tuning called “stretching”, becoming as more influencing as smaller the piano’s dimensions are. Furthermore, this effect becomes stronger, since our wavelets represent the same note on different octaves connected to slightly different harmonics. We have no experience concerning other intruments, but on the first sight, there should be a higher impact of the harmonics. As an effect of this data structure, the number of notes with reconstruction in a wrong octave should decrease, but on the other side there should be more additional notes detected, as harmonics of an also detected existing note.

In Figure 6.26 we superimposed magnifications of the 12 plots of Figure 6.25, here we used the original norming. We set the transforms of Wavelets 5 and 6 to the same color (cyan), for laying the attention to the other ones. The different positions of the peaks on the frequency axis are clearly visible. By comparing their frequencies with the 12 notes from the chromatic scale, we obtain the following correspondings[9], sorted from c to b without concerning the octave[10]:

[7]The machine was in this case an AMD Athlon X2 64, with 2 GHz clock rate and 2 GB memory.

[8]Thus, we resigned to use non-negative coding in the analysis as introduced on Page 124.

[9]Please note, the frequencies on the x-axis of Figure 6.26 equal one plus the number of oscillations per 256 points.

[10]We label the octaves following the HELMHOLTZ pitch notation as standard in European music.

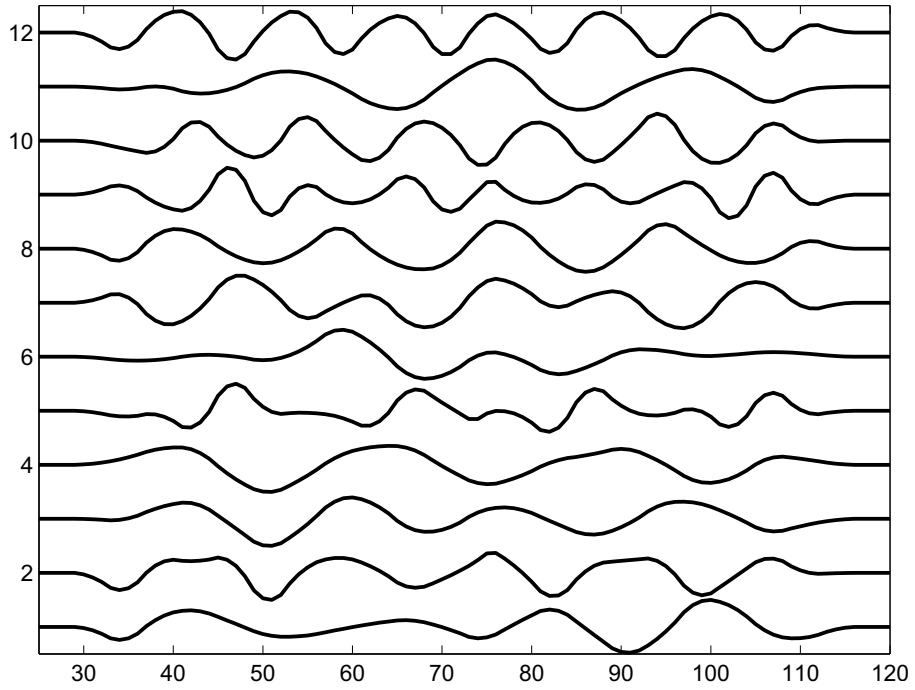


Figure 6.24: Representants of the wavelets in $\mathcal{D}^*$, according to Equation (4.1), $r_1 = 18$, $r_2 = 2^6$ for $j = 1 \dots 12$, restricted to the support and normalized to a maximal absolute value of 1/2.

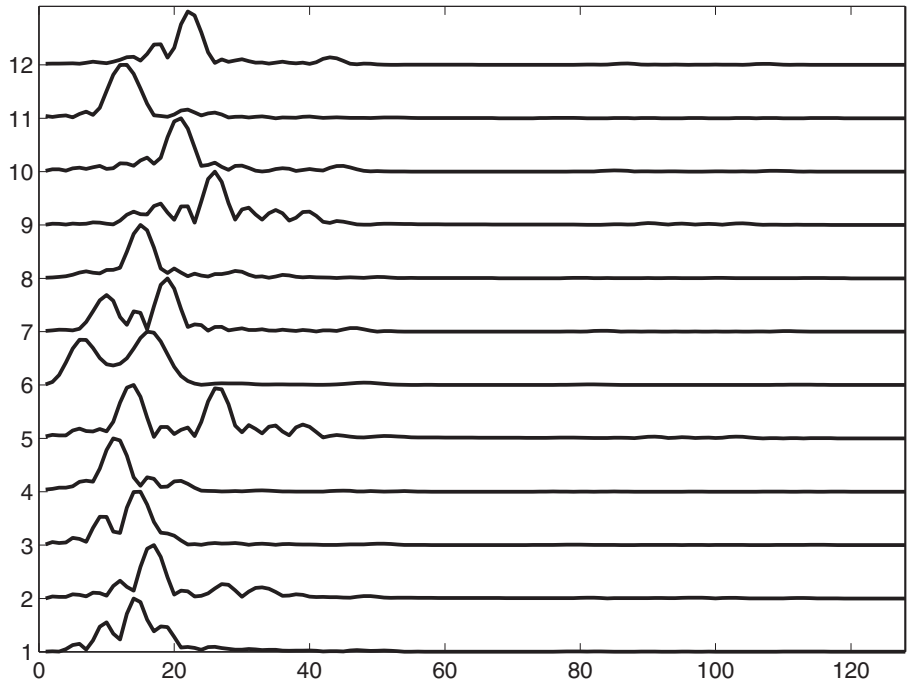


Figure 6.25: Absolute values of the fast FOURIER transform of the functions from Figure 6.24, normalized to maximal amplitude 1.

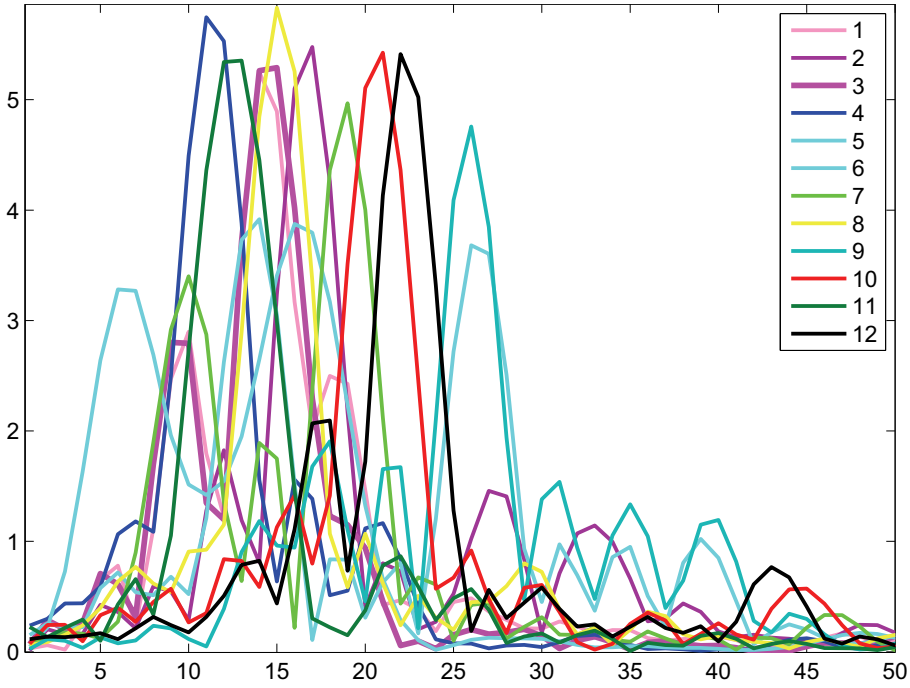


Figure 6.26: Superposition of the transforms from Figure 6.25 with original norming.

note	Frequency [Hz]	osc./256 pt		corresponding wavelet
c″	523.3	16.74	x	2
c♯″/d♭″	554.4	17.74	x	7
d″	587.3	18.8		
d♯″/e♭″	622.3	19.91	x	10
e′	329.7	10.6		4
f″	698.5	22.35	x	12
f♯″/g♭″	740	23.68		
g′	392	12.54	x	11
g″	784	25.08	x	9
g♯′/a♭′	415.3	13.29	x	1 and 3
a′	440	14.08		
a♯′/b♭′	466.2	14.92	x	8
b′	493.9	15.8		

Please note, that a♭ and g are both represented by two wavelets (in the case of g in two different octaves) and Figure 6.24 shows, they are not shifted versions of each other. We will later have a closer look on them, but this effect also occurs using other approaches as [12] shows. Nevertheless, the most interesting point of this tabular is the fact, that every element of the diatonic a♭ major scale (a♭, b♭, c, d♭, e♭, f, g) is represented by at least one wavelet, corresponding to the key of the analyzed sonata. The additional occurence of natural e is also clear, since there are several natural e in the three times repeated second part of the Scherzo.

Reconstructing the score

As there is now a correlation between the wavelets and the different notes, the next step is to compare the score with the coding signal. For this purpose we compute first the sums $m^{\xi}_{j,r_2,i}$ of the coefficients as already proposed in Equation (6.3) in context of linear guideways. Figure 6.27 displays $\mu = \sum_{i=1}^{N} m^{0}_{j,r_2,i}$, revealing analog to Figure 6.21, which wavelet (i.e. which note) has been used how often on which scale (i.e. which octave). For detecting the notes, we analyze the sequences $m_{j,r_2} = (m^{1}_{j,r_2,i})_{i=1}^{N}$ or rather $\tilde{m}_{j,r_2} = (m^{1}_{j,r_2,i})_{i=1}^{N} * \tilde{\mathbb{1}}$, in slight contrast to the approach of Section 6.2.2. Here $\tilde{\mathbb{1}}$ is a sequence containing 5 elements equaling one. We had to choose this, in comparison with Subsection 6.2.2 short, length of $\tilde{\mathbb{1}}$, since a quarter-note has an average length of 0.218 s or 1695.8 data points, spreading out on usually 7 subsignals. So this necessary smoothing of m_{j,r_2} already smears it into almost the whole following quarter-note, as we have to keep in mind during the reconstruction. Nevertheless, we prefer this instead of using overlapping subsignals y_i, since 4484 subsignals are already given.

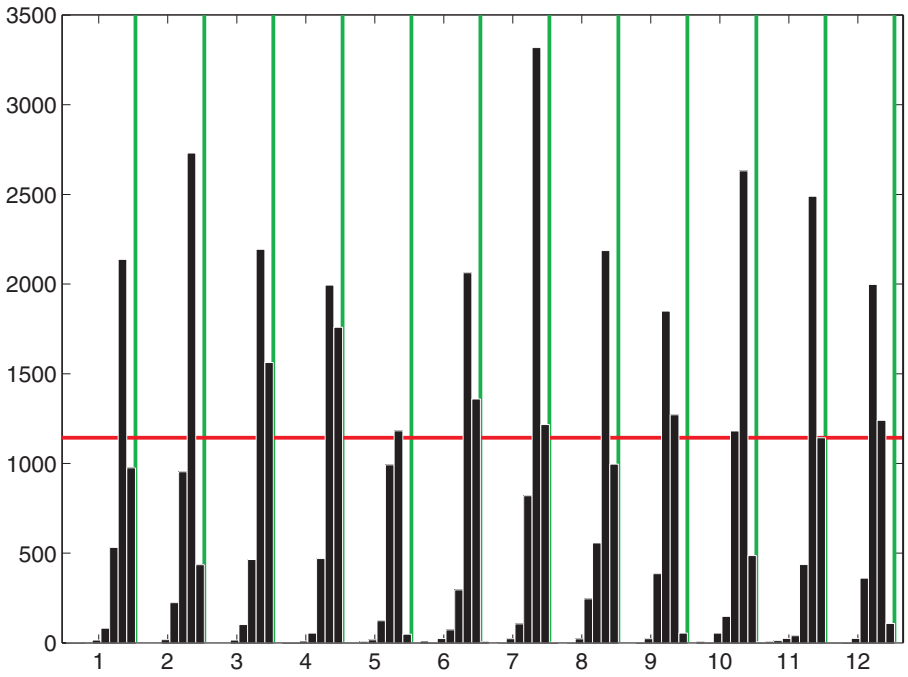


Figure 6.27: μ, usages of the numbered wavelets on different scales for coding the Scherzo. The green lines separate the wavelets, the red line indicates the 20 largest values.

The authors of [129] and [12] made no transcription of this composition, but instead exemplified their algorithm on Ludwig van Beethoven's Bagatelle Opus 33,1 and analyzed their analogons to m_{j,r_2} by thresholding. We prefer to use a different method for the following reasons: First the dynamics of the example composition is varying between pianissimo and fortissimo, additionally there are sforzatos, accents of the notes and rests. All this is not only varying according to the score, but also

accoring to the interpretation of the pianist. Thus a pure thresholding would weight the loud and accented parts higher against the other ones. Second the volume of the music is also depending on the number of notes played at the same time and to be detected, and also the decay of notes during fading phase has to be taken into account. So also a renormalization of the $\tilde{m}_{j,r_2}$ by the amplitudes of the y_i, as used on Pages 120–124 for machine noise is not reasonable, since this can not handle especially rests and the unknown number of notes. That can also not be handled by strategies like choosing the n notes connected to the largest n values of $(\tilde{m}_{j,r_2,r_1;i})_{r_1}$ or choosing that values above fixed fraction of $\max_{r_1}\{\tilde{m}_{j,r_2,r_1;i}\}$ for every subsignal y_i. A combination of several of this methods, maybe with additional musical knowledge, would probably offer reliable results, but for adjusting the parameters we have to consider a wide variance of compositions.

So we let the point of automatic detection as a task for further improvement and detect here the active notes by establishing manually for every quarter note a local threshold from the set $\{0.9, 1.2, 1.8, 2.1\}$ and accepted every note as detected, that exceeded this threshold. Here we treat the two representations of g and a$^\flat$ separately, since, according to the results we establish on Page 137 (concerning notes without action), the should rarley arise at the same time. Due to the slowness of this transcription method, we analyze just the first part of the Scherzo, beginning with the pickup up to Measure 16 (Figure 6.29), and search just for that notes connected to μ increasing the red line in Figure 6.27, indicating the 20 largest values. Not taking into account the Wavelets 5 and 6 (see above), this are 17 ones representing, since a$^\flat{}'$, g$'$ and g$''$ are double represented, 14 notes. This notes are illustrated in Figure 6.28. It is clear that this notes can not cover the whole range of the notes in the score, especially most of that notes on bass cleve are not considered. Thus the reconstruction, established in Figure 6.30, contains many rests in the left hands stave.

Figure 6.31 shows the $\tilde{m}_{j,r_2}$ according to the first 4 measures, incuding pickup, parallel to the original and reconstructed score, including the local threshold. Figure 6.32 shows additionally the first measure in detail.

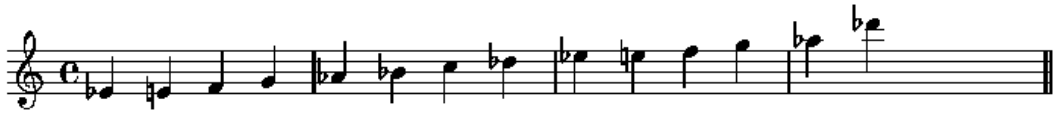


Figure 6.28: The 14 notes, considered for reconstruction of the score

All in all the first 16 measures contain 98 notes. If we count every half and dotted half double or rather tripple, there are $n = 124$ notes, where eight of them are eighths, the remainings quarters. Out of this we did not search for 17, not listed in Figure 6.28. All in all we found 118 notes, so we have to distinguish (we use the same notation as in [12] and [129]):

- True positive (tp) detected: The note in the score and the detected one correlate.

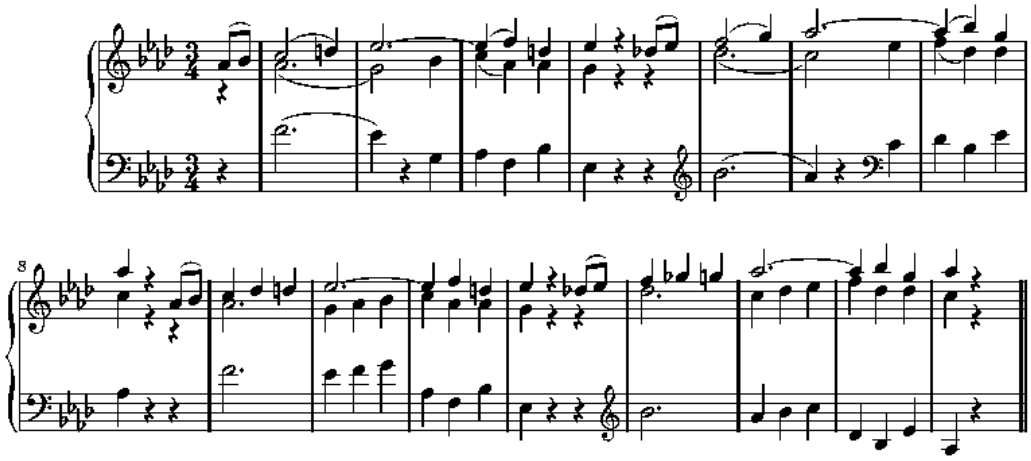


Figure 6.29: Score of the first part of the analyzed Scherzo. Dynamics and accents had been suppressed for better readability.

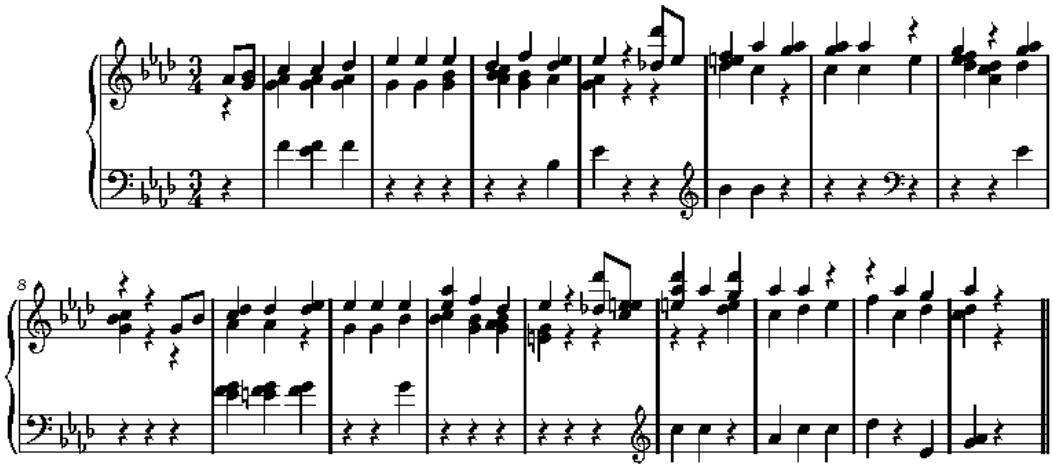
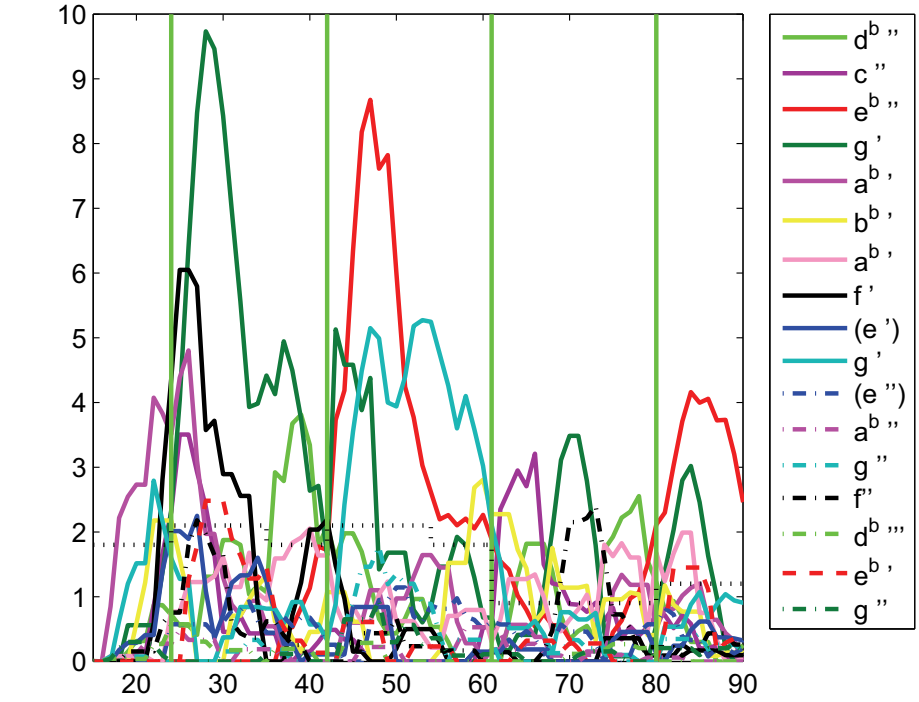
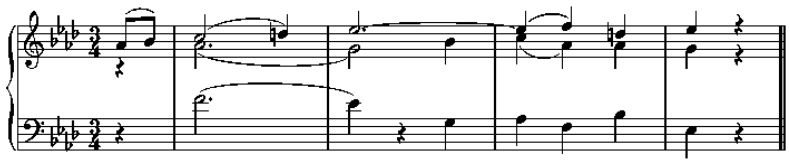
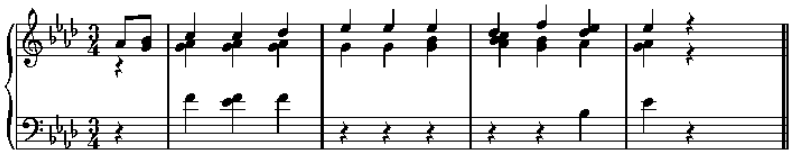


Figure 6.30: Reconstructed score of the first part of the analyzed Scherzo. Halfs and longer notes are separated into quartes. Please observe, we just transcripted the notes listed in Figure 6.28.





(b) Original Score



(c) Reconstructed Score

Figure 6.31: First 4 measures of the Scherzo, decomposition and score. The local threshold is indicated by the horizontal, black, dotted lines, the measures are separated by the vertical, green lines

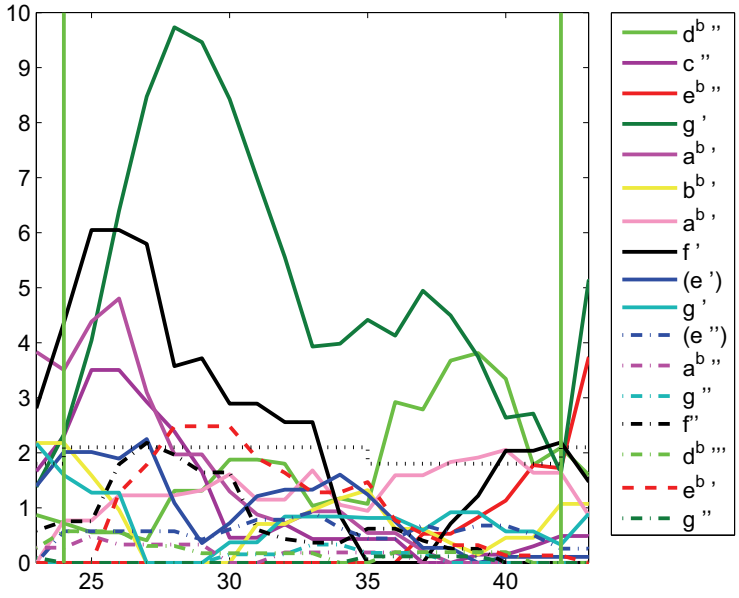


Figure 6.32: Decomposition of the first measure of the Scherzo. Observe the magenta $a^{\flat\prime}$ declines with rising length of the note, while the pink one arises.

- False positive (fp) detected: There is no correspondent in the score to a detected note.
- False negative (fn) detected: The note in the score is not detected.

Counting the different cases we obtain:

type	among this	No	e_1	e_2	e_3	e_4
tp		76	61.3%	71.0%	73.8%	84.4%
fn		46				
	not searched (ns)	17	13.7%			
	semitone found, no description of original (sno)	4	3.2%	3.7%		
	without action (wa)	13	10.5%	12.2%	12.7%	
	semitone found, existing description of original (seo)	6	4.8%	5.6%	5.8%	6.7%
	octave found (of)	1	0.8%	0.9%	1.0%	1.1%
	nothing found (no)	7	5.6%	6.5%	6.8%	7.8%
			$\frac{\cdot}{tp+fp}$			
fp		42	35.6%			
	inside a third (it)	12	10.2%			
	additional semitone (as)	10	8.5%			
	additional octave (ao)	5	4.2%			
	others	15	12.7%			

Here e_1 denotes the relative error $\frac{\cdot}{tp+fn}$, while $e_2 = \frac{\cdot}{tp+fn-ns}$, $e_3 = \frac{\cdot}{tp+fn-ns-sno}$ and $e_4 = \frac{\cdot}{tp+fn-ns-sno-wa}$.

In the tabular we stated several rates for true positive and false positive detection, since we have to consider the following distinctions:

- It is clear not to take into account the notes, not considered during transcription (ns), as introduced above. We set aside to observe this notes additionally, before developing an automatic detection system, since they represent just 12% of the notes but would induce high additional complexity in analysis. Furthermore, introducing this notes into Figures 6.31–6.36 would make them unreadable.

- Notes without any corresponding wavelet, as, e.g., the d″ in Measure 1. They occure in the score to rarely for learning a representing wavelet. In such a case we found generally the next semitone (sno), which is the best possible approximation. For obtaining also approximations of this tones, we need to enlarge either the database by more recordings of this note, or to consider a greater dictionary. The first case would be a passable way, if we need to transcript several compositions in different keys. In the second case, it is not clear, what the additional wavelets will aproximate, see below.

- Notes without action (wa). In the case of to long notes, as, e.g., the dotted half e♭″ with bounded quarter in Measure 2 and 3 (Figure 6.33). We see, the amplitude of the note fades fast and in many cases it is missing at all at least during the last quarter. Clearly the actions of the notes are louder, but additionally the fading sound has a different structure of harmonics, as recognizable for the trained human ear, so it maybe needs to be specified by a different wavelet. To prove this, we have a look on the g′ in Measure 2 (see Figure 6.33) and the dotted half a♭′ in Measure 1 (see Figure 6.32): In both cases we see a high value for the coresponding $\tilde{m}_{j,r_2}$ of one representing wavelets ($j = 3$, $r_2 = 7$, magenta or rather $j = 11$, $r_2 = 7$, dark green) in the action phase. Later on the other representing wavelet ($j = 1$, $r_2 = 7$, pink or rather $j = 9$, $r_2 = 6$, turquoise) arises in the fading phase. So we see, most of our wavelets are just optimized to the action sequence of the note, since it has a higher amplitude than the fading sound. Consequently we have problems to detect the last phase of long notes. Thus we need for a better detection of long notes a higher number of wavelets to fit also all fading sounds.

The other errors (seo, of, no) has to be classified as real missdetections, maybe avoidable with a better fitted dictionary (i.e. a better local optimum). We want to emphasize, that true detection rates of 71.0% and 73.8% in combination with false positive 35.6% are close to that one achieved in [129], while 84.4% is much better.

In the case of false positive detected notes, there are several classes where the reasons for their detection are more or less clear:

- Additional octaves (ao): As already mentioned on Page 129 we expect to find additional harmonics, especially octaves (the strongest harmonic), since

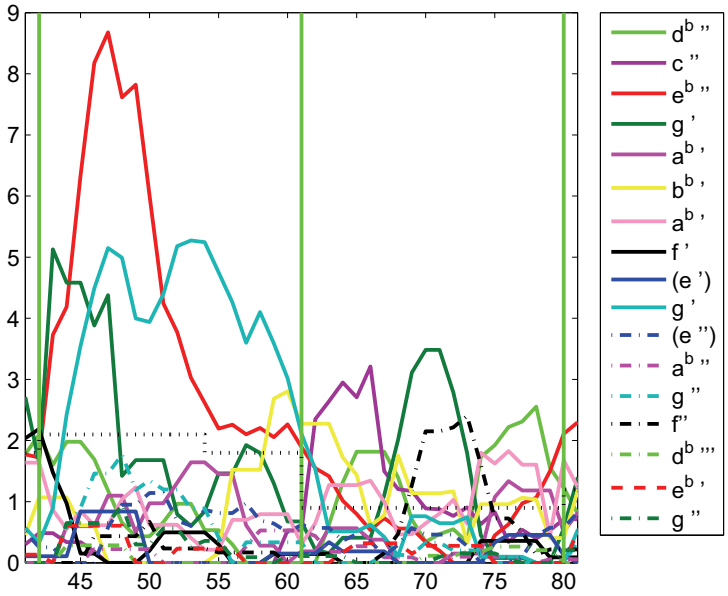


Figure 6.33: Decomposition of the 2nd and 3rd measure of the Scherzo, indicating the change of the note between the action phase and the fading phase – The dark green representative of g' is strong in the beginning of the measure and fades later, while the turquoise one acts reversely.

harmonic structure of the wavelets represents probably not the full harmonic structure of the notes. Here we registered 5 octaves, as, e.g., in the end of Measure 12 (Figure 6.34), where $d^{\flat'''}$ is added to $d^{\flat''}$. Furthermore, we registered 8 octaves at points we did not search for the original note (above denoted with *ns*) that probably mask their octaves in a full representation. We forego observing other harmonics like fifth, since their number should be even smaller.

- Additional notes inside a third (*it*): If in the original score two note form a third, we often obtain the detection of a additional note at the minor or major second. Its amplitude is sometimes already much higher than that of the true positive detected ones, as, e.g., exemplified in Measure 5, Figure 6.35. This is a clear consequence of the application of the OMP, which chooses first a middle frequency if two neighboring ones superpose. Choosing another sparse coding algorithm like BP or FOCUSS maybe could help to minimize the number of this false positive detections, but need a multiple of computation time.

- The same could be an explanation for detected additional semitones (*as*) to notes existing in the score, possibly induced by the superposition of this note with some other ones.

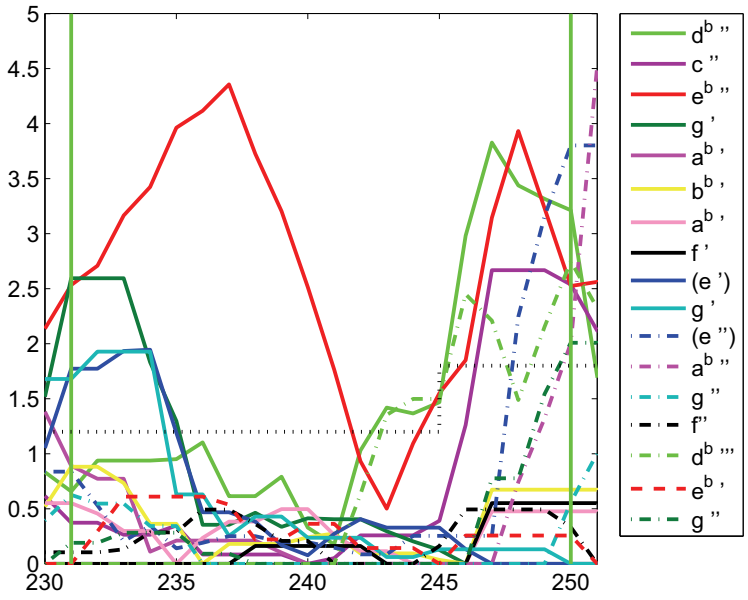


Figure 6.34: Decomposition of the 12th measure of the Scherzo, observe the additional $d^{\flat'''}$ (light green, dash-dotted).

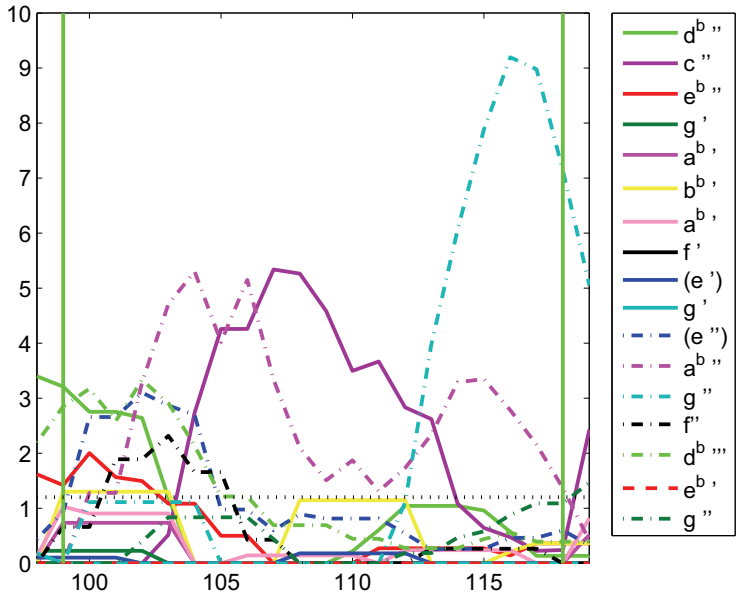


Figure 6.35: Decomposition of the 5th measure of the Scherzo, indicating seconds (here c'', violett) in thirds (here $b^{\flat\prime}$, yellow and $d^{\flat\prime\prime}$, light green).

Other results

Not in relation to the score, but considering its interpretation by the pianist, we can observer several features in the course of the different m_{j,r_2}: Most noticeable is the shortening of Measure 13 and, to a lesser extent, 14 (see Figure 6.36), having a support of just 15 or rather 17 subsignals, instead of 19 or rarely 20 as the other measures of the Scherzo's first part. Another element of the pianist's interpretation are notes played up to an eighth earlier or later than denoted in the score, as for example pointed out in measure 14 (see Figure 6.36), where the $d^{\flat\prime\prime}$ arises late.

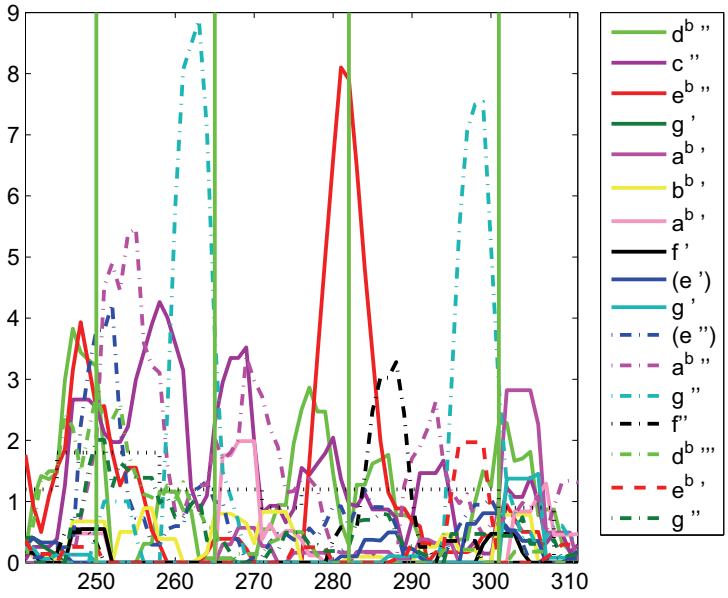


Figure 6.36: Decomposition of the 13th to 16th measure of the Scherzo, including pickup, pointing at the shortenings of Measure 13 and 14.

Before we conclude this section, we want to have an additional look to the double represented notes and the not considered wavelets. We mentioned already above, that the two wavelets represent different structures of the harmonics and probably indicate by this different phases of the note. But there are more music-theoretical sources for changing the harmonics. All of them could be reasons for obtaining notes represented by several wavelets:

- Usage of the pedals: For a trained human ear it is clear recognizable, that the use of the pedals (the right one changing the damping of the piano, the left one the volume, the middle one has no fixed properties) changes also the harmonics of the notes. Thus pedals are probable one reason, why the ture-positve detection rate decreases at the sforzatos in measures 1, 5, 9 and 13.
- A similar effect can occure through the coaction of two parallelly played notes, increasing maybe the resonance oscillation or higher harmonic of one string.

Although this is in general a less influential point, according to the in general not fixed phase between the notes, this can introduce some non-linearities to the whole signal. Depending on this, different wavelets can be approximated, corresponding to the notes played at the same time.

Similar considerations are true for the Wavelets 5 and 6, representing the superposition of two notes. According to the variable phase between the different sounds there should not exist representatives for chords. It is also not possible to interpret this as harmonics as introduced on Page 129, since a harmonic has not an amplitude as high as the original frequency. Most probable the existence of this wavelets shows that our dictionary is just locally optimal (as already assumed on Page 129) and there are alternative dictionaries, equal to this one beside Wavelets 5 and 6, offering a better performance.

Conclusion

All in all we see, there is still a wide field for improvement of this example: First we need to find an automatic detection device. Additionally the number of representing wavelets could be enlarged, giving the chance to detect also different phases of the notes, the usage of pedals and rarely used notes. According to computation time and local minima, such an enlargement should probably be introduced stepwise, first learning a smaller dictionary, later adding extra wavelets, for separating the details of tones. Almost the same is valid for the number of atoms chosen by OMP, as considered on Page 128. Furthermore, the standard structure of notes with a main frequency and some harmonics offers also chances to adapt the local minima finder. A wavelet not representing this structure can be replaced by one, more adapted to the data. Lastly the usage of BP instead of OMP could increase the detection rate, but this is, for reasons of computation time, not usable throughout the learning phase.

Nevertheless, the results of this sections are encouraging: We obtained, without optimizing the parameters, true-positive and false-positive detection rates in the range of the results of [12] and [129], although wavelets are maybe not the first choice in the field of music encoding. However this results have to hold in the case of using an automatic detection device and different same composition (e.g., Beethoven's Bagatelle Opus 33,1 as in [12]). Nevertheless we were able to extract plenty of information on, e.g., the score, the waveform of the tones or even some intepretatory elements of the pianist, although we are still working with a linear model assuming the undisturbed superposition of the notes and including almost no additional musical knowledge. This knowledge could be included in weights for the atoms, e.g., an active note is probable active also in the next subsignal. Thus considering deeper musical knowledge, as, e.g., done in [174, 140, 108, 90] can maybe result in better detection rates.

Chapter 7

Summary and Outlook

Main topic of this thesis is the learning of dictionaries $\mathcal{D} = \{d_l\}_{l=1}^L \subset \mathbb{R}^n$ or rather $d \in L_{(2,p')}(\Omega, \Omega')$ for sparse coding in the case of given signals $\mathcal{Y} = \{y_i\}_{i=1}^N \subset \mathbb{R}^n$ or rather $y \in L_2(I, \Omega)$. Basically the body of the thesis, so the Chapters 3 to 6, is divided into two parts: First an algorithmic part, containing Chapters 3, 4 and the applications in Chapter 6, and second the functional analytical chapter, 5.

The later one contains the derivative and a convergence analysis for a continuous complement of dictionary learning, minimizing the functional

$$\begin{aligned} E(d,x) \;=\; & \int_I \left(\int_\Omega \left(y(\omega,l) - \int_{\Omega'} d(\omega,\omega') x(\omega',l) \,\mathrm{d}\mu'\omega' \right)^2 \mathrm{d}\mu\omega \right. \\ & \qquad \left. + \lambda_x \int_{\Omega'} f(x(\omega',l)) \,\mathrm{d}\mu'\omega' \right) \mathrm{d}\nu l \\ & + \lambda_d \left(\int_{\Omega'} \left(\int_\Omega d(\omega,\omega')^2 \,\mathrm{d}\mu\omega \right)^{p'/2} \mathrm{d}\mu'\omega' - c \right)^2 \\ & + \left(\lambda_1 g_1 \left(\|x\|_p \right) + \lambda_2 g_2 \left(\|\nabla x\|_p \right) \right) + \lambda_3 g_3 \left(\|\nabla d\|_{(2,p)} \right) \end{aligned}$$

for a wide field of sparsity functions $f \in \mathcal{F}_1 \cup \mathcal{F}_2$ as defined in Definition 5.1. In Theorem 5.25 we proved[1], in combination with the results of Subsection 5.1.5 the existence of a minimum of E for $y \in L_2(\Omega, I)$ in $(W_{1,(2,p')}, W_{1,p})$ or rather, depending on the properties of f, in $(W_{1,(2,p')}, L_p)$.

The second main result of Chapter 5, as stated in Theorems 5.34, 5.53 and Corollary 5.43, concerns how to find the existing minimum. We proved, that a generalized conditional gradient algorithm constructs a sequence $(d_n, x_n) \rightharpoonup (d, x)$ with $E(d,x) \leq E(d_n, x_n)$. The question, if (d,x) has to be a stationary point of the algorithm is answered negative in Example 5.51 in case

$$f \in \{\mathcal{F}_1 | f' < b < \infty, f' \text{ Lipschitz on } (0,\infty)\} \,,$$

[1] This proof includes also the existence of an optimal continuous sparse coding for a given dictionary d

while in case $f \in \mathcal{F}_2$, or if f is an LIPSCHITZ continuous element of $\mathcal{G}_\delta$ (an surrogate for $\mathcal{F}_1$), this problem is in general still open (cf. Remark 5.35), but positively answered, if $(d_n, x_n) \to (d, x)$.

Additionally to the mentioned open questions there are in this context some topics for further research:

- Does a minimum also exist in the case of an alternative sparsity measure $f \notin (\mathcal{F}_1 \cup \mathcal{F}_2)$, e.g., in case f has an logarithmic behaviour? If yes, can the minimum also be constructed?
- Are there analog results if the optimal dictionary is defined via another error functional or even via maximization of a probability functional, as introduced, e.g., on Page 17?
- Can the dictionary operator

$$Dx = \int_{\Omega'} d(\omega, \omega,)x(\omega', l)\, \mathrm{d}\mu'\omega'$$

  be replaced by some operator $K = AD$? If yes, what kind of constraints has A to satisfy?
- Can any uniqueness result, according to the minimization of E, be obtained?

Turning to the algorithmic part of this thesis, we state in Chapter 3 first a method to match wavelets ψ to given signals y by use of the lifting scheme (see especially Corollary 3.4 and Remark 3.5), minimizing

$$\|y - \psi_{j^*,k^*}\|_2.$$

Later on in Section 3.3 we enhanced this method to a more shift-invariant version (cf. Equations (3.10) and (3.11)). Only with this version we obtain in the application in Subsection 6.2.1 significant results, which maybe could even improved, if we extend the theory by a non-linear matching via a dual lifting step.

The main results of the algorithmic part are developed in Chapter 4. The MODW Algorithm 4.1 offers the potential to learn dictionaries of the form

$$\begin{aligned} \mathcal{D} &= \{d_l\}_{l=1}^{L} = \bigcup_{j=1}^{J} \mathcal{D}_j \\ &= \bigcup_{j=1}^{J} \left\{ \psi_j \left(\frac{\cdot - r_1}{r_2} \right), \varphi_j \left(\frac{\cdot - r_1}{2^{j'}} \right) \middle| |r_1| \le R_\rho; r_2 = 2^\rho; \bar{j} \le \rho \le j'; r_1, \rho \in \mathbb{Z} \right\} \end{aligned}$$

using a combination of the well known lifting scheme, as also used in Chapter 3, and the MOD algorithm. This satisfies the main intention to obtain a dictionary with a fast coding algorithm and (limited) shift- and scaling-invariance. The applications in Chapter 6 showed later the usability of MODW in several frameworks, where the main problem is still the vulnerability of MODW to find just a locally optimal

dictionary. This problem all known dictionary learning algorithms have in common (see Subsection 2.2.2), but it would be a topic for further research, to reduce this vulnerability. One additional idea could be the combination of lifting with dictionary learning algorithms more advanced than MOD, as, e.g., the K-SVD. Furthermore, a second direction of research would be the question, if learning algorithms like this can also be constructed for other classes of functions, which are maybe more favorably for special sets of data $\mathcal{Y}$ or have better coherence and spark (cf. 2.2.1) than wavelet dictionaries. Examples for this could be α-modulation classes, wavelet packets or curvelets.

In Chapter 6 we collect applications of the algorithms from Chapters 3 and 4. Our main topic there is the condition monitoring of linear guideways in Section 6.2. We obtain reliable results with both methods, according to detect pittings in the guideway. While the transform using phase-selflets denotes clearly the position of simulated pittings (cf., e.g., Figures 6.8 or 6.9), the results from the MODW based sparse coding are, due to the measurement device, less exact in position as visible in Figures 6.18f and 6.22f but with extreme good detection-rate. Moreover, this measurement device is less interfering for the running machinery and offers, due to the full decomposition of the signal, a natural basis for the detection of further conditions of the machinery. This guides also to the natural point for advancements of this example, to widen the signal basis $\mathcal{Y}$ and learn also wavelets indicating problems of, e.g., lubrication, the sealings and the bearings.

The second important example in Section 6.3 concerns the reconstruction of the score from a piano recording. Although the results are not as good as in the foregoing example, we were able to assign every note of the a$^\flat$ major scale a wavelet, having approximately the same pricipal frequency (cf. Page 129). Furthermore, also the reconstruction exemplified on the first 16 measures (cf. Pages 132–138) were relatively successful and produces results in the range of other algorithm proposed during the past two years. Nevertheless, in particular additional atoms where chosen without any counterpart in the score. Moreover, the detection of notes could be improved by investing a higher computational effort and more musical knowledge, as pointed out on Page 141.

All in all, the theme of dictionary construction is, especially compared with fixed dictionaries, still a field in fast development, and with several issues to solve. Additionally to the mentioned one we did not concern at all, e.g., with problems like convergence speed and approximation qualities of the result. Also in the field of continuous dictionaries there are more areas for further research than we could concern. Since this thesis gives some new evidence on dictionary learning, both on the general side and on a more specific problem, it can help to understand the structure of this non-linear problem and offers additional methods for technical applications.

Appendix A

Linear Guideways

In this short appendix we want to introduce the mechanical device used for obtaining the datasets in Section 6.2 and their applications. Furthermore, we provide some evidence about the background of why our data processing has some applicational impact, according to the terms of mechanical engineering.

A.1 Experimental setup

Linear guideways (cf. also [138, 72]) are important components of every machine tool. Several of them are used for positioning of the workpiece as well as of the milling head, drill or another tool. But linear guideways exist also in a wide variance of other mechanical devices, e.g., they are installed in push loading drawers.

Basically there are two types of linear guideways: linear sliding guides and linear rolling guides. Here we just concern with the linear rolling guides with revolving ball bearings, as usually installed in machine tools. It usually consist of a profile rail and a carriage (sometimes also called runner block), which serves as a carrier for workpiece or tool. The contact between rail and carriage is realized by usually four lines of ball bearings, running through the carriage from front to the end, see Figure A.1 for a cross-section of this device. Every bearing rolls between the profile rail and a second rail inside the carriage and revolves in a channel through the interior of the carriage (cf. Figure A.2).

For our experiments we used two different setups: The data in Subsection 6.2.1 represent acoustic emmisions of a Rexroth 1623-812-10 runner block on a 1605-.0 profile rail. Figure A.3 shows this machinery. In Figure A.4 we see additionally the drive mechanism, a single axis NC feed equipment, connected with the carriage using a screw fixed plate and a connection rod. At the way of one bearing two artificial pittings were produced, one approximately at a quarter of the total feed way of 330 mm, the other, less deep one, approximately at a half. For the acoustic emission (AE) measurement, we fixed an encapsulated piezo-ultrasonic-microphone at the front end of the lateral surface of the carriage using double-sided adhesive

tape, see Figure A.5 for details. Using this measurement device, we obtained in general one single signal , if the carriage passes a pitting, probable generated when the ball next to the the microphone passes the pitting as exemplified in Figure A.6.

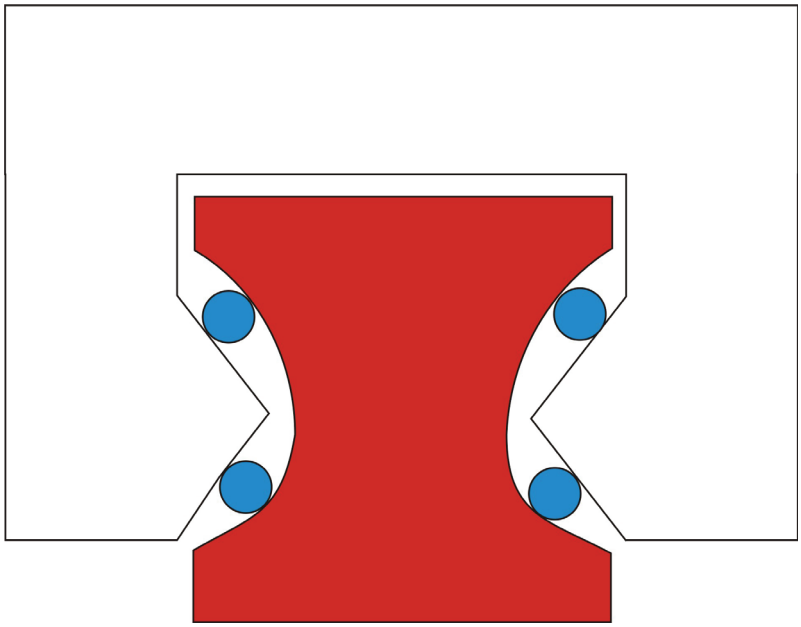


Figure A.1: Cross section through a linear guideway, between the carriage (white) and the profile rail (red) are four ball bearings (blue).

The second measurement device for the data of Subsection 6.2.2 is slightly modified: First, we have here just one pitting, that locates almost at the midpoint of the feed way on a 300 mm long profile rail. Second, the balls in the bearings of the carriage were connected by a ball-chain, reducing the vibrations and collisions of the balls and improving their lubrication. This also reduces the amplitude of pitting signal of interest. Furthermore, this time we placed the microphone, mechanically advantageous but metrologically disadvantageous, at the face end of the profile rail. Thus we obtain a more indirect measurement, which additionally is also varying in amplitude, depending on the distance to the carriage. This positioning results in a less localized signal induced by the pitting, since every ball passing it generates the same small subsignal.

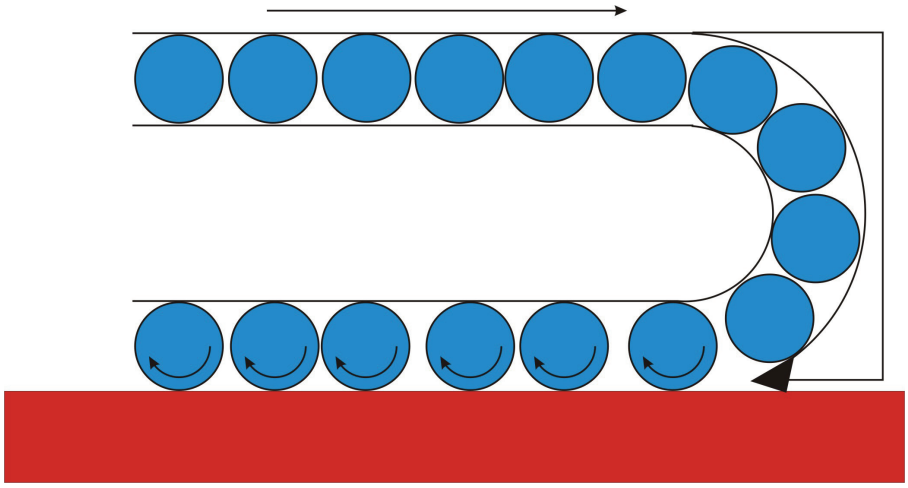


Figure A.2: Ways of the balls (blue) inside the carriage (white), below (red) the profile rail is visible. Observe the sealing of the carriage (black). The arrows indicate the moving or rather rolling direction.

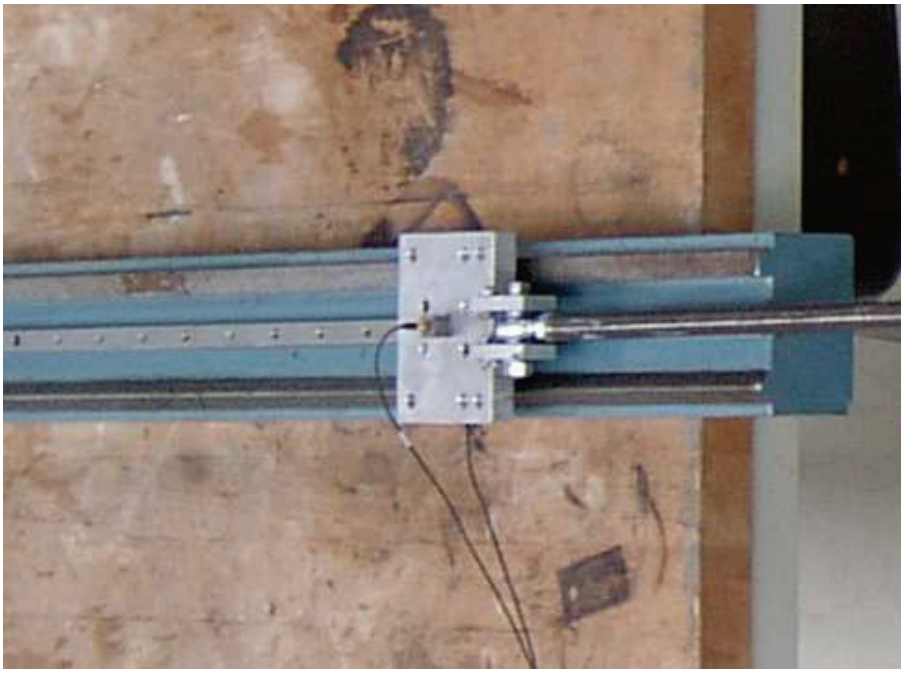


Figure A.3: Linear guideway with carriage

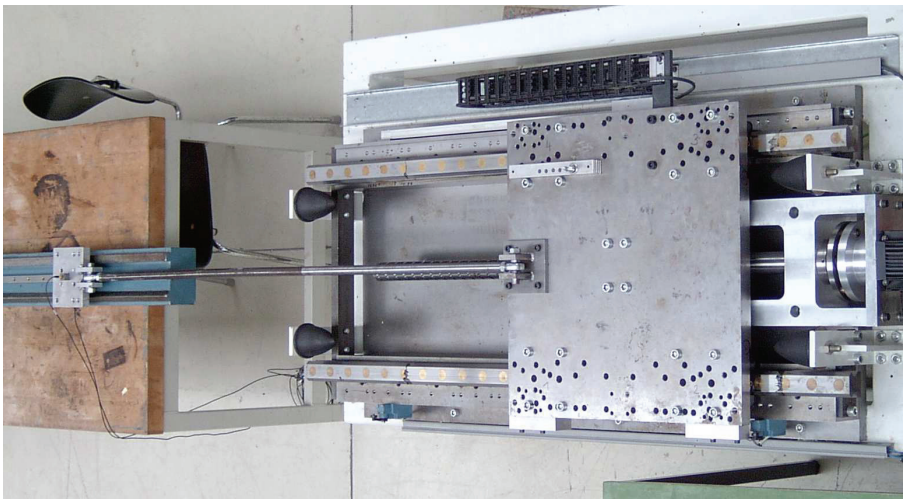


Figure A.4: Linear guideway with carriage and drive equipment

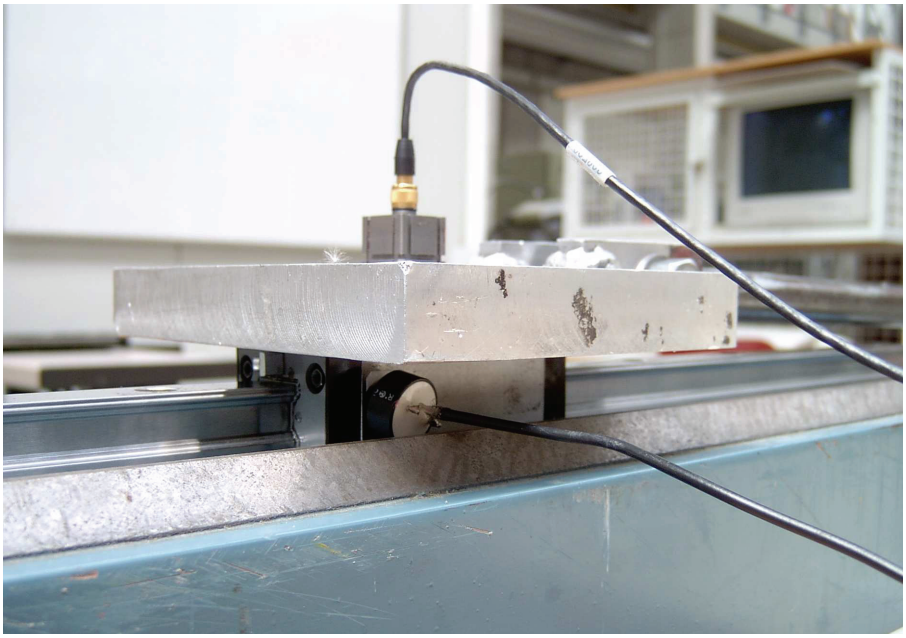


Figure A.5: Magnification of the carriage. On the lateral surface the microphone for acoustic emission measurement is placed. The accelerations sensor on top of the carriage is not used.

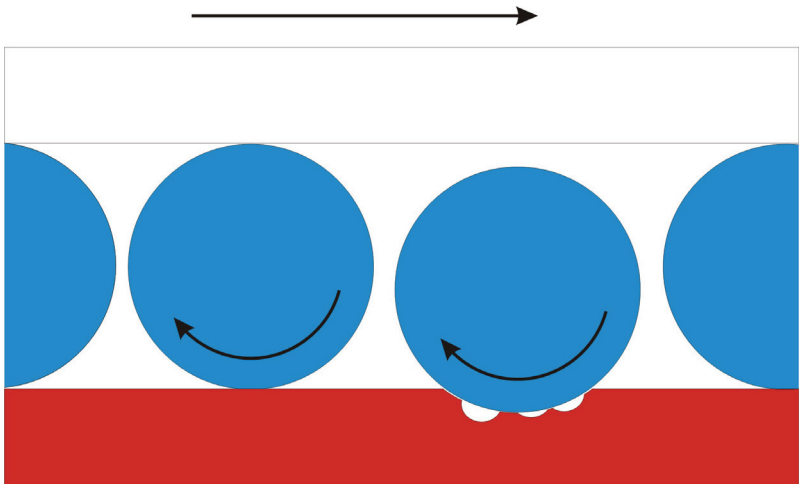


Figure A.6: A ball of the bearing falls into a pitting in the profile rail (red). Above (white) the corresponding rail inside the carriage. The arrows indicate the moving or rather rolling direction.

A.2 Condition monitoring

Condition monitoring (cf. [165, Chapter 6]) of any type of machinery is necessary for maintaining its efficiency and accuracy (cf. [71, 9]). Monitoring implies the detection of changes in the state of the machinery by measuring critical parameters and comparing them with reference values. Condition monitoring also includes an automatic classification of potential defects. This classification should allow to assess, e.g., which components are defective and whether, or to which degree, the subsequent production process is affected by the defect.

Typically, several sets of different data (e.g., acceleration or rather vibration, velocity or position of components, acoustic emission, forces, pressures, received electric power) are measured and analyzed for satisfying this task. Such data sets can either be obtained by external sensors or can directly been read off given control data of the machinery. Classical monitoring algorithms define thresholds either for the data themselves or for derived quantities like their mean values, standard deviation, moments et cetera. Alternatively, thresholds can be defined after performing a transformation of the data (Fourier transform, cepstrum, wavelet transform). A defect would be detected if one of the thresholds is not met.

In the case of linear guideways (a short description of the used device is given in Section A.1) interesting defects are, e.g. (see also Figure A.2):

- Abrasion of balls or guideways,
- Impurities inside the carriage,
- Blocking of the balls,
- Sealing wear,

- Local defects (pittings) of the guideways.

These defects affect the positioning accuracy, velocity and current drain of the machinery.

In this special environment, we try to monitor the condition of one linear guideway based on just one measurement of acoustic emission. Local defects can arise at every point of the profile rail at a scale depending on the size of the defect and the velocity of the carriage. So wavelets are a natural choice for analyzing this data. In order to obtain an optimal signal-to-noise ratio, the wavelets should be matched to the characteristic pattern of the actual defect, as pointed out in Section 3.1. KUHFUSS and SCHÄDLICH obtained in [92] reliable detection results using continuous wavelets. However, the computing time for a continuous framework exceeds by far the capacity of available online systems and produces redundant, unused data. So we chose for our application in Subsection 6.2.1 a discrete transform with matched phaselets and, as natural development, in Subsection 6.2.2 the dictionary contruction restricted to wavelets.

Appendix B

Additional Material to Chapter 5

In this appendix we collected for the sake of readability additional information, alternative approaches and a technical lemma to Chapter 5.

B.1 Using relaxation in Section 5.1.2

As mentioned on Page 63 we can achieve a weak lower semicontinous surrogate for E_2 as defined in Definition 5.4 by using relaxation.

The approach of relaxation, as presented for example in [27, Chapter 3] or shorter in [7, Section 2.1.3], is based on replacing a functional E by its weak lower semicontinuous envelope $\operatorname{sc}^- E$. The minimum of the later one is then also infimum of the first one. Using this approach we just have to change the 2nd summand and get the following

Theorem B.1 (Minimum of $\operatorname{sc}^- E$)**.** *Let $y \in L^p((\Omega, I), \mathbb{R})$ and let f or rather E satisfy Condition 5.9. Then the weak lower semicontinuous envelope of E (defined by Equation (5.7)) is given by*

$$\begin{aligned}\operatorname{sc}^- E \;\geq \int_I \int_\Omega (y(\omega,l) - (Dx)(\omega,l))^2 \,\mathrm{d}\mu\omega \;\mathrm{d}\nu l \;+ \lambda_x \operatorname{sc}^- \left(\int_I \int_{\Omega'} f(x(\omega',l)) \,\mathrm{d}\mu'\omega' \;\mathrm{d}\nu l \right) \\ + \lambda_d \left(\int_{\Omega'} \left(\int_\Omega d(\omega,\omega')^2 \,\mathrm{d}\mu\omega \right)^{p'/2} \mathrm{d}\mu'\omega' \;- 1 \right)^2 + \lambda_1 g_1(\|x\|_p). \qquad \text{(B.1)}\end{aligned}$$

$\operatorname{sc}^- E$ has a minimum in $L^p(\mathbb{R})$ that is also an infinmum of E.

Proof. According to [27, Proposition 3.7] $\operatorname{sc}^- E$ is given by

$$\operatorname{sc}^- E \geq \operatorname{sc}^- E_1 + \operatorname{sc}^- E_2 + \operatorname{sc}^- E_3 + \operatorname{sc}^- \tilde{E}_4.$$

As E_3 is constant for fixed d, $\mathrm{sc}^- E_3 = E_3$. Furthermore since g_1 is weak lower semicontinuous according to its definition in Condition 5.9, we have $\mathrm{sc}^- \tilde{E}_4 = \tilde{E}_4$.

For proving weak lower semicontinuity of E_1, we prove strong continuity and convexity. The first one holds on L_2, as for $x_1, x_2 \in L_2$

$$\begin{aligned} |E_1(x_1) - E_1(x_2)| &= \left| \iint (y - Dx_1)^2 - (y - Dx_2)^2 \, \mathrm{d}\mu\omega \, \mathrm{d}\nu l \right| \\ &\leq \|D(x_1 - x_2)\|_2 \cdot \| - 2y + Dx_1 + Dx_2\|_2 \end{aligned}$$

converges to zero (the second factor is finite), if $\|Dx\|_2$ tends to zero for $\|x\|_2 \to 0$. This convergence can be deduced, using

$$\|Dx\|_2 = \left\| \int_{\Omega'} d(\omega, \omega') x(l, \omega') \, \mathrm{d}\mu'\omega' \right\| \leq \|d\|_2^2 \cdot \|x\|_2^2 \, ,$$

(holding, as d is independent of l, and x independent is of ω), from the finiteness of d obtained from $E_3 < \infty$.

E_1 is also convex, as

$$\begin{aligned} \lambda E_1(x_1) + (1-\lambda) E_1(x_2) &\geq E_1(\lambda x_1 - (1-\lambda) x_2) \\ &\Leftrightarrow \\ \iint \lambda (Dx_1)^2 + (1-\lambda)(Dx_2)^2 &\geq \iint \lambda^2 (Dx_1)^2 + (1-\lambda)^2 (Dx_2)^2 \\ &\qquad + 2\lambda(1-\lambda) Dx_1 Dx_2 \\ &\Leftrightarrow \\ \iint (Dx_1 - Dx_2)^2 &\geq 0 \, . \end{aligned}$$

As a continuous and convex function E_1 is also weak lower semicontinuous (cf. Theorem 2.14), thus we have $\mathrm{sc}^- E_1 = E_1$. Using the sum rule for weak lower semicontinuous envelopes as given in [27, Proposition 3.7] we have

$$\mathrm{sc}^-(E_1 + E_2 + E_3 + \tilde{E}_4) \geq \mathrm{sc}^- E_1 + \mathrm{sc}^- E_2 + \mathrm{sc}^- E_3 + \mathrm{sc}^- \tilde{E}_4 = E_1 + \mathrm{sc}^- E_2 + E_3 + \tilde{E}_4$$

and all the results of the theorem we get by applying [27, Theorem 3.8], since E is coercive according to Lemma 5.10. □

Equality in (B.1) can just be achieved, if E_1, E_3 and E_4 are weak continuous, which is in general not satisfied.

This result gives us a lower bound for $\mathrm{sc}^- E$. Even more, if we can achieve the weak lower semicontinuous envelope, we get the following corollary:

Corollary B.2. *Under the assumptions of Theorem B.1 let x_n be a minimizing sequence of $\mathrm{sc}^- E$. Then there is a subsequence x_{n_j} with $E(x_{n_{j+1}}) \leq E(x_{n_j})$. Furthermore $\lim_{n\to\infty} E(x_n)$ is an infimum of E.*

Since the existence of an infimum is not a satisfactory result, and additionally $\mathrm{sc}^- E$ and $\mathrm{sc}^- E_2$ are not easy to compute, we focus in the course of the thesis on finding a minimum using other tools.

B.2 Technical lemma

For the proof of Lemma 5.50 the following technical lemma is necessary. We placed it here for the sake of readability of the thesis.

Lemma B.3. *Let U be a measurable set, $m_U = \int \chi_U(\zeta)\,\mathrm{d}\tilde{\mu}\zeta > 0$ and $x_w \in L_2(U)$. Furthermore let $x_n \to x$ in $L_2(U)$ and $x(\zeta) > 0$ for $\tilde{\mu}$-almost every $\zeta \in U$. If B_n is a set with $x_n(\zeta) \leq 0$ for $\tilde{\mu}$-almost every $\zeta \in B_n$, then*

$$\int_{B_n} x_w(\zeta)^2\,\mathrm{d}\tilde{\mu}\zeta \overset{n\to\infty}{\longrightarrow} 0\ .$$

Proof. Let us assume there is an $\epsilon > 0$ and a subsequence $n_j \subset \mathbb{N}$,

$$\int_{B_{n_j}} x_w(\zeta)^2\,\mathrm{d}\tilde{\mu}\zeta = \int_U \chi_{B_{n_j}}(\zeta) x_w(\zeta)^2\,\mathrm{d}\tilde{\mu}\zeta > \epsilon$$

for every $n_j > n_0$, then we have

$$\int_U \chi_{B_{n_j}}(\zeta)\,\mathrm{d}\tilde{\mu}\zeta > \frac{\epsilon}{\|x_w\|_2^2} \quad \forall n_j > n_0. \tag{B.2}$$

On the other hand, from the convergence of x_n to x, we get:

$$\begin{aligned} \int_U \chi_{B_n}(\zeta) x(\zeta)^2\,\mathrm{d}\tilde{\mu}\zeta &\leq \int_U \chi_{B_n}(\zeta)(x(\zeta) - x_n(\zeta))^2\,\mathrm{d}\tilde{\mu}\zeta \\ &\leq \int_U (x(\zeta) - x_n(\zeta))^2\,\mathrm{d}\tilde{\mu}\zeta \leq \delta \end{aligned} \tag{B.3}$$

for all $n > n_0$. In the following we show, that this two results contradict, this gives $\int_{B_n} x(\zeta)^2\,\mathrm{d}\tilde{\mu}\zeta < \epsilon$ for all $\epsilon > 0$.

First we assume, U is a finite set. We know from Equation (B.3),

$$\int_U \chi_{B_n}(\zeta) x(\zeta)^2\,\mathrm{d}\tilde{\mu}\zeta \to 0\ . \tag{B.4}$$

Assume, this sequence converges monotonically (otherwise we choose a subsequence), then according to the monotone convergence theorem, there is an $x_B \in L_2(U)$,

$$\int_U \chi_{B_n}(\zeta) x(\zeta)^2\,\mathrm{d}\tilde{\mu}\zeta \to \int_U x_B(\zeta)^2\,\mathrm{d}\tilde{\mu}\zeta$$

and $\chi_{B_n}(\zeta)x(\zeta)^2 \to x_B(\zeta)^2$ $\tilde{\mu}$-almost everywhere. Since

$$\chi_{B_n}(\zeta)x(\zeta)^2 = \begin{cases} x(\zeta)^2, & \zeta \in B_n \\ 0, & \text{otherwise}\ , \end{cases}$$

also x_B has to be of the form $x_B = \chi_B x$. So the rules of convergence give ($x(\zeta) > 0$ $\tilde{\mu}$-almost everywhere) $\chi_{B_n}(\zeta) \to \chi_B(\zeta)$ $\tilde{\mu}$-almost everywhere. For satisfying Equation (B.4) B has to be (without loss of generality) an empty set, since in the case of a null set $B \neq \emptyset$ the whole argumentation is valid for $U \backslash B$.

Since $\chi_{B_n}(\zeta) \to \chi_B(\zeta)$ $\tilde{\mu}$-almost everywhere, also the integral has to converge:

$$\int_U \chi_{B_n}(\zeta)\,\mathrm{d}\tilde{\mu}\zeta \;\to \int_u \chi_B(\zeta)\,\mathrm{d}\tilde{\mu}\zeta \;= 0$$

This contradicts Inequality (B.2), defining a positive lower bound for $\int_U \chi_{B_n}(\zeta)\,\mathrm{d}\tilde{\mu}\zeta$. Thus, the assumption is disproved for finite U.

In the case of an infinite U, there is a ball B_R , such that

$$\int_{U\cap B_R} x_w(\zeta)^2\,\mathrm{d}\tilde{\mu}\zeta \;\geq \|x_w\|^2 - \epsilon/2\ .$$

Furthermore, according to the finite part of the proof, there is an n_0, such that

$$\int_{B_n\cap B_R} x_w(\zeta)^2\,\mathrm{d}\tilde{\mu}\zeta \;\leq \|x_w\|^2 \int_{B_n\cap B_R} \mathrm{d}\tilde{\mu}\zeta \;\leq \epsilon/2\ .$$

So all in all we have

$$\int_{B_n} x_w(\zeta)^2\,\mathrm{d}\tilde{\mu}\zeta \;\leq \int_{B_n\cap B_R} x_w(\zeta)^2\,\mathrm{d}\tilde{\mu}\zeta \;+ \int_{U\cap B_R^C} x_w(\zeta)^2\,\mathrm{d}\tilde{\mu}\zeta \;\leq \epsilon\ ,$$

what proves the lemma. □

Bibliography

[1] Samer A. Abdallah. *Towards music perception by redundancy reduction and unsupervised learning in probabilistic models*. PhD thesis, King's College, London, February 2003.

[2] Michal Aharon, Michael Elad, and Alfred M. Bruckstein. K-SVD and its non-negative variant for dictionary design. In Manos Papadakis, Andrew F. Laine, and Michael A. Unser, editors, *Proceedings of SPIE*, volume 5914, September 2005.

[3] Michal Aharon, Michael Elad, and Alfred M. Bruckstein. The K-SVD: An algorithm for designing of overcomplete dictionaries for sparse representation. *IEEE Transactions on Signal Processing*, 54(11):4311–4322, November 2006.

[4] Michal Aharon, Michael Elad, and Alfred M. Bruckstein. On the uniqueness of overcomplete dictionaries, and a practical way to retrieve them. *Journal of Linear Algebra and Applications*, 416:48–67, July 2006.

[5] Hans Wilhelm Alt. *Lineare Funktionalanalysis: Eine anwendungsorientierte Einführung*. Springer, Berlin, Heidelberg, 3rd edition, 1999.

[6] Fred Attneave. Some informational aspects of visual perception. *Psychological Review*, 61(3):183–193, 1954.

[7] Gilles Aubert and Pierre Kornprobst. *Mathematical Problems in Image Processing: Partial Differential Equations and the Calculus of Variations*. Number 147 in Applied Mathematical Sciences. Springer-Verlag, New York, 1st edition, 2002.

[8] Horace Basil Barlow. Possible principles underlying the transformation of sensory messages. In Walter A. Rosenblith, editor, *Sensory Communication*, pages 217–234. MIT Press, Cambridge, MA, 1961.

[9] Wilfried J. Bartz, L. Engel, and Ullrich Engel. *Schäden an geschmierten Maschinenelementen. Gleitlager, Wälzlager, Zahnräder*. Number 28 in Kontakt und Studium. Expert-Verlag, third edition, January 1999.

[10] Anthony J. Bell and Terence J. Sejnowski. The 'independent components' of natural scenes are edge filters. *Vision Research*, 37(23):3327–3338, 1997.

[11] Kathrin Berkner and Raymond O. Wells, Jr. Smoothness estimates for soft-threshold denoising via translation invariant wavelet transforms. *Applied and Computational Harmonic Analysis*, 12(1):1–24, 2002.

[12] Thomas Blumensath. *Bayesian Modelling of Music: Algorithmic Advances and Experimental Studies of Shift-Invariant Sparse Coding*. PhD thesis, Queen Mary, University of London, January 2006.

[13] Thomas Blumensath and Mike E. Davies. Blind separation of maternal and fetal ECG recordings using adaptive sparse representations. In *Proceedings of ICA Research Network International Workshop*, Liverpool, UK, September 2006.

[14] Thomas Blumensath, M. Yaghoobi, and Mike E. Davies. Iterative hard thresholding and L_0 regularisation. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, Honolulu, USA, April 2007.

[15] Rafael Bogacz, Malcom W. Brown, and Christophe Giraud-Carrier. Emergence of movement-sensitive neurons' properties by learning a sparse code of natural moving images. *Advances in neuronal information processing systems (NIPS)*, pages 838–844, 1999.

[16] Liliana Borcea, James G. Berryman, and George C. Papanicolaou. Matching pursuit for imaging high-contrast conductivity. *Inverse Problems*, 15(4):811–849, 1999.

[17] Andrew P. Bradley. Shift-invariance in the discrete wavelet transform. In Changming Sun, Hugues Talbot, Sébastien Ourselin, and Tony Adriaansen, editors, *Proc. VIIth Digital Image Computing: Techniques and Applications*, Sydney, December 2003.

[18] Kristian Bredies, Dirk A. Lorenz, and Peter Maass. A generalized conditional gradient method and its connection to an iterative shrinkage method. Accepted for publication in *Computational Optimization and Applications*, 2005.

[19] Judith C. Brown and Bin Zhang. Musical frequency tracking using the methods of conventional and 'narrowed' autocorrelation. *Journal of the Acoustic Society of America*, 89:2346–2354, 1991.

[20] Lamberto Cesari. *Optimization – theory and applications*. Number 17 in Applications of mathematics. Springer, New York, 1983.

[21] Joseph O. Chapa and M. Rao Raghuveer. Optimal matched wavelet construction and its application to image pattern recognition. In *Proceedings of the SPIE Conference on Wavelet Applications for Dual Use*, Symposium on OE/Aerospace Sensing and Dual Use Photonics, 1995.

[22] Scott Shaobing Chen, David L. Donoho, and Michael A. Saunders. Atomic decomposition by basis pursuit. *SIAM Journal on Scientific Computing*, 20(1):33–61, 1998.

[23] Albert Cohen. *Numerical Analysis of Wavelet Methods*. Elsevier Science B. V., 2003.

[24] Ronald R. Coifman and David L. Donoho. Translation-invariant de-noising. In Anestis Antoniadis and Georges Oppenheim, editors, *Wavelets and Statistics*, number 103 in Lecture Notes in Statistics, pages 125–150. Springer-Verlag, New York, 1995.

[25] Shane F. Cotter, Kenneth Kreutz-Delgado, and Bhaskar D. Rao. Efficient backward elimination algorithm for sparse signal representation using overcomplete dictionaries. *IEEE Signal Processing Letters*, 9(5):145–147, May 2002.

[26] Bernard Dacorogna. *Direct Methods in the Calculus of Variations*. Number 78 in Applied mathematical sciences. Springer, Berlin, 1st edition, 1989.

[27] Gianni Dal Maso. *An Introduction to* Γ*-Convergence*. Birkhäuser, Boston, 1992.

[28] Ingrid Daubechies. *Ten Lectures on Wavelets*. Number 61 in CBMS-NSF Regional Conference Series in Applied Mathematics. SIAM (Society for Industrial and Applied Mathematics), 1992.

[29] Ingrid Daubechies and Wim Sweldens. Factoring wavelet transforms into lifting steps. *Journal of Fourier Analysis and Applications*, 4(3):245–267, 1998.

[30] Laurent Daudet. Sparse and structured decompositions of audio signals in overcomplete spaces. In *Proceedings of the Digital Audio Effects Workshop (DAFx2004)*, Naples, 2004.

[31] Mike E. Davies and Laurent Daudet. Fast sparse subband decomposition using FIRSP. In *EUSIPCO'04*, Vienna, 2004.

[32] Ronald A. DeVore and George G. Lorentz. *Constructive Approximation*. Number 303 in Grundlehren der mathematischen Wissenschaften. Springer Verlag, Heidelberg, 1993.

[33] Oscar Divorra Escoda, Lorenzo Granai, Mathieu Lemay, Javier Molinero Hernandez, Pierre Vandergheynst, and Jean-Marc Vesin. Ventricular and atrial activity estimation through sparse ECG signal decompositions. In *Proc. of IEEE ICASSP'06*, May 2006.

[34] Oscar Divorra Escoda, Lorenzo Granai, and Pierre Vandergheynst. On the use of a priori information for sparse signal approximations. *IEEE Transactions on Signal Processing*, 54(9):3468–3482, 2006.

[35] David L. Donoho. For most large underdetermined systems of linear equations, the minimal ℓ^1-norm near-solution approximates the sparsest near-solution. *Communications on Pure and Applied Mathematics*, 59(7):907–934, July 2006.

[36] David L. Donoho and Michael Elad. Optimally-sparse representation in general (non-orthogonal) dictionaries via ℓ^1 minimization. *Proceedings of the National Academy of Sciences*, 100:2197–2202, 2003.

[37] Samuel P. Ebenezer, Antonia Papandreou-Suppappola, and Seth B. Suppappola. Classification of acoustic emissions using modified matching pursuit. *EURASIP Journal on Applied Signal Processing*, 3:347–357, 2004.

[38] Michael Elad. Sparse representations are most likely to be the sparsest possible. *EURASIP Journal on Applied Signal Processing*, 2006, 2006.

[39] Michael Elad and Michal Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Transactions on Image Processing*, 15(12):3736–3745, December 2006.

[40] Michael Elad, Jean-Luc Starck, P. Querre, and David L. Donoho. Simultaneous cartoon and texture image inpainting using morphological component analysis (MCA). *Journal on Applied and Computational Harmonic Analysis*, 19:340–358, November 2005.

[41] Kjersti Engan, Sven Ole Aase, and John Håkon Husøy. Multi-frame compression: Theory and design. *EURASIP Signal Processing*, 80(10):2121–2140, 2000.

[42] Kjersti Engan, John Håkon Husøy, and Sven Ole Aase. Optimized frame design for a matching pursuit based compression scheme. In *Proceedings EUSIPCO'98*, pages 153–156, Rhodos, Greece, September 1998.

[43] Kjersti Engan, Bhaskar D. Rao, and Kenneth Kreutz-Delgado. Frame design using FOCUSS with method of optimal directions (MOD). In *Norwegian Signal Processing Symposium*, pages 31–33, Norway, September 1999.

[44] Kjersti Engan, Bhaskar D. Rao, and Kenneth Kreutz-Delgado. Regularized FOCUSS for subset selection in noise. In *Nordic Signal Processing Symposium*, pages 247–250, Linkoping, Sweden, June 2000.

[45] Kjersti Engan and Karl Skretting. A novel image denoising technique using overlapping frames. In *Proceedings of 2nd IASTED International Conference on Visualization, Imaging, and Image Processing*, pages 265–270, Malaga, Spain, September 2002.

[46] Sebastian E. Ferrando, Edward J. Doolittle, Ariel J. Bernal, and Luis J. Bernal. Probabilistic matching pursuit with gabor dictionaries. *Signal Processing archive*, 80(10):2099–2120, October 2000.

[47] Rosa M. Figueras i Ventura, Pierre Vandergheynst, and Pascal Frossard. Evolutionary multiresolution matching pursuit and its relations with the human visual system. In *Proceedings of the EUSIPCO 2002 Conference*, volume 2, pages 395–398, September 2002.

[48] Rosa M. Figueras i Ventura, Pierre Vandergheynst, and Pascal Frossard. Low rate and flexible image coding with redundant representations. *IEEE Transactions on Image Processing*, 15(3):726–739, March 2006.

[49] Peter Földiák and Malcom P. Young. Sparse coding in the primate cortex. In Michael A. Arbib, editor, *The Handbook of Brain Theory and Neural Networks*, pages 895–898, 1995.

[50] J. C. Fu, C. A. Troy, and P. Jonathan Phillips. A matching pursuit approach to small drill bit breakage prediction. *International Journal of Production Research*, 17(14):3247–3261, September 1999.

[51] Allen Gersho and Robert M. Gray. *Vector Quantization and Signal Compression*. The International Series in Engineering and Computer Science. Kluwer Academic Publishers, Norwell, MA, USA, 1991.

[52] Mark Girolami. A variational method for learning sparse and overcomplete representations. *Neural Computation*, 13:2517–2532, 2001.

[53] Michael M. Goodwin and Martin Vetterli. Matching pursuit and atomic signal models based on recursive filter banks. *IEEE Transactions on Signal Processing*, 47(7):1890–1902, July 1999.

[54] Ramesh A. Gopinath. The phaselet transform – an integral redundancy nearly shift-invariant wavelet transform. *IEEE Transactions on Signal Processing*, July 2003.

[55] Lorenzo Granai and Pierre Vandergheynst. Sparse decomposition over multi-component redundant dictionaries. In *Workshop on Multimedia Signal Processing (MMSP'04)*, pages 494–497, September 2004.

[56] Seth A. Greenblatt. Atomic decomposition of financial data. *Computational Economics*, 12(3):275–293, 1998.

[57] Thomas Greiner. Orthogonal and biorthogonal texture matched wavelet filterbanks for hierarchical texture analysis. *Signal Processing*, 54:1–22, April 1996.

[58] Rémi Gribonval. Fast matching pursuit with a multiscale dictionary of gaussian chirps. *IEEE Transactions on Signal Processing*, 49(5):994–1001, 2001.

[59] Rémi Gribonval. Sparse decomposition of stereo signals with matching pursuit and application to blind separation of more than two sources from a stereo mixture. In *Proc. Int. Conf. Acoust. Speech Signal Process. (ICASSP'02)*, Orlando, FL, USA, May 2002.

[60] Rémi Gribonval, Laurent Benaroya, Lorcan Mc Donag, and Frédéric Bimbot. Non negative sparse representation for Wiener based source separation with a single sensor. In *Proc. IEEE Intl. Conf. Acoust. Speech Signal Process (ICASSP'03)*, pages 613–616, 2003.

[61] Rémi Gribonval, Rosa M. Figueras i Ventura, and Pierre Vandergheynst. A simple test to check the optimality of a sparse signal approximation. *EURASIP Signal Processing Journal*, 86(3):496–510, March 2006.

[62] Rémi Gribonval, Guillaume Gravier, Laurent Benaroya, Alexey Ozerov, and Frédéric Bimbot. Séparation de sources à partir d'un seul capteur pour la reconnaissance robuste de la parole. In *Proceedings Journées d'Etude sur la Parole (JEP 2004)*, February 2004.

[63] Rémi Gribonval, Philippe Jost, Pierre Vandergheynst, and Sylvain Lesage. MoTIF: An efficient algorithm for learning translation invariant dictionaries. In *Proceedings of the ICASSP'06*, Toulouse, France, May 2006.

[64] Rémi Gribonval, Sacha Krstulovic, Pierre Leveau, and Laurent Daudet. A comparison of two extensions of the matching pursuit algorithm for the harmonic decomposition of sounds. In *Proceedings of the Workshop on Applications of Signal Processing to Audio and Acoustics*, pages 259–262. IEEE, October 16–19 2005.

[65] Rémi Gribonval, Sylvain Lesage, and Sacha Krstulovic. Under-determined source separation: Comparison of two approaches based on sparse decompositions. In *Proc. ICA'06*, Springer-Verlag LNCS series, pages 633–640, March 2006.

[66] Rémi Gribonval, Bao Liu, and Shih-Fu Ling. Bearing failure detection using matching pursuit. *NDT & E International*, 35(4):255–262, 2002.

[67] Rémi Gribonval and Morten Nielsen. Approximation with highly redundant dictionaries. In *Wavelets: Applications in Signal and Image Processing*, volume 5207 of *Proceedings of the SPIE*, pages 216–227, San Diego, USA, August 2003.

[68] Rémi Gribonval and Morten Nielsen. Sparse representation in unions of bases. *IEEE Transactions of Information Theory*, 49(12):3320–3325, December 2003.

[69] Rémi Gribonval and Morten Nielsen. Highly sparse representations from dictionaries are unique and independent of the sparseness measure. *Applied and Computational Harmonic Analysis*, 22(3):335–355, May 2007.

[70] Rémi Gribonval and Pierre Vandergheynst. On the exponential convergence of matching pursuits in quasi-incoherent dictionaries. *IEEE Transactions on Information Theory*, 52(1):255–261, January 2006.

[71] Johann Grosch, Anton Bäumel, Jürgen Föhl, Thomas Hirsch, Lothar Issler, Wolfgang Janzen, Axel Roßmann, Rolf Sinz, Bernd Thoden, and André Werner. *Schadenskunde im Maschinenbau, Charakteristische Schadensursachen – Analyse und Aussagen von Schadensfällen.* Number 308 in Kontakt und Studium. Expert-Verlag, fourth edition, 2004.

[72] Knut Großmann, editor. *Stand und Entwicklung von Profilschienenführungen an Werkzeugmaschinen.* TU Dresden, December 2004.

[73] Stephan Hasler, Heiko Wersing, and Edgar Körner. Combining reconstruction and discrimination with class-specific sparse coding. *Neural Computation*, 19(7):1897–1918, 2007.

[74] Robert W. Heath, Jr., Thomas Strohmer, and Arogyaswami J. Paulraj. On quasi-orthogonal sequences for CDMA. *IEEE Transactions on Information Theory*, 52(3):1217–1226, 2006.

[75] Jostein Herredsvela, Kjersti Engan, Thor Ole Gulsrud, and Karl Skretting. Detection of masses in mammograms by watershed segmentation and sparse representation using learned dictionaries. In *Proceedings NORSIG 2005*, pages 35–40, Stavanger, Norway, September 2005. Tapir Akademiske Forlag.

[76] Jostein Herredsvela, Kjersti Engan, Thor Ole Gulsrud, and Karl Skretting. Texture classification using sparse representations by learned compound dictionaries. In *SPARSE'05*, Rennes, France, November 2005.

[77] Felix Herrmann. Fractional spline matching pursuit: A quantitative tool for seismic stratigraphy. Expanded Abstract in the *Proceedings of the 71th Annual meeting of the Society of Exploration Geophysicists*, 2001.

[78] Patrik O. Hoyer. Non-negative sparse coding. In *Proceedings of the IEEE Workshop on Neural Networks for Signal Processing*, number XII in Neural Networks for Signal Processing, pages 557–565, Martigny, Switzerland, 2002.

[79] Niall Hurley, Scott Rickard, and Paul Curran. Parameterized lifting for sparse signal representations using the gini index. In *SPARSE'05*, Rennes, France, November 2005.

[80] Aapo Hyvarinen. Independent component analysis in the presence of gaussian noise by maximizing joint likelyhood. *Neurocomuting*, 22:49–67, 1998.

[81] Tommi S. Jaakkola. *Variational methods for inference and estimation in graphical models*. PhD thesis, MIT, Cambridge, 1997.

[82] Philippe Jost, Pierre Vandergheynst, and Pascal Frossard. Tree-based pursuit: Algorithm and properties. *IEEE Transactions on Signal Processing*, 54(12):4685–4697, 2006.

[83] Philippe Jost, Pierre Vandergheynst, Sylvain Lesage, and Rémi Gribonval. Learning redundant dictionaries with translation invariance property: The MoTIF algorithm. In *SPARSE'05*, Rennes, France, November 2005.

[84] Winfried Kaballo. *Einführung in die Analysis, III*. Hochschultaschenbuch. Spektrum, Akademischer Verlag, Heidelberg, Berlin, 1999.

[85] Juha Karvanen and Andrzej Cichocki. Measuring sparseness of noisy signals. In *Proceedings of the Fourth International Symposium on Independent Component Analysis and Blind Signal Separation*, pages 125–130, Nara, Japan, April 1–4 2003.

[86] Kunio Kashino, Kazuhiro Nakadai, Tomoyoshi Kinoshita, and Hidehiko Tanaka. Application of bayesian probability network to music scene analysis. In *Working Notes of IJCAI Workshop of Computational Auditory Scene Analysis (IJCAI-CASA)*, pages 52–59, August 1995.

[87] Nick Kingsbury. Complex wavelets and shift invariance. In *Proceedings of the IEE Colloquium on Time-Scale and Time-Frequency Analysis and Applications*, London, February 2000.

[88] Nick Kingsbury. Complex wavelets for shift invariant analysis and filtering of signals. *Applied and Computational Harmonic Analysis*, 10(3):234–253, 2001.

[89] Anssi Klapuri. *Signal processing methods for the automatic transcription of music*. PhD thesis, Tampere University of Technology, Finland, April 2004.

[90] Anssi Klapuri, Tuomas Virtanen, and Jan-Markus Holm. Robust multipitch estimation for the analysis and manipulation of polyphonic musical signals. In *Proceedings of the COST-G6 Conference on Digital Audio Effects*, pages 141–146, Verona, Italy, 2000.

[91] Kenneth Kreutz-Delgado, Joseph F. Murray, Bhaskar D. Rao, Kjersti Engan, Te-Won Lee, and Terence J. Sejnowski. Dictionary learning algorithms for sparse representation. *Neural Computation*, 15:349–396, February 2003.

[92] Bernd Kuhfuss and Stephan Schädlich. Condition monitoring of linear guideways with process adapted wavelets (selflets). In *First International Conference on Computing and Solutions in Manufacturing Engineering - CoSME' 04*, Brasov, Romania, September 16–18 2004.

[93] Konrad P. Körding, Peter König, and David J. Klein. Learning of sparse auditory receptive fields. In *Proceedings of the International joint Confonference of Neural Networks*, volume 2, pages 1103–1108, 2002.

[94] Sylvain Lesage, Rémi Gribonval, Frédéric Bimbot, and Laurent Benaroya. Learning unions of orthonormal bases with thresholded singular value decomposition. In *Proceedings of the IEEE Conference on Acoustics, Speech and Signal Processing*, volume V, pages 293–296, 2005.

[95] Michael S. Lewicki and Terence J. Sejnowski. Coding time-varying signals using sparse, shift-invariant representations. *Advances in neuronal information processing systems (NIPS)*, 11:730–736, 1999.

[96] Michael S. Lewicki and Terence J. Sejnowski. Learning overcomplete representations. *Neural Computation*, 12:337–365, 2000.

[97] Jacques-Louis Lions. *Optimal Control of Systems Governed by Partial Differential Equations*. Springer, 1971.

[98] E. D. Livshits. Convergence of greedy algorithms in banach spaces. *Math. Notes*, 73(3):342–358, 2003. translation from Mat. Zametki 73, No.3, 371–389.

[99] E. D. Livshits. On the rate of convergence of a pure greedy algorithm. *Math. Notes*, 76(4):497–510, 2004. translation from Mat. Zametki 76, No.4, 539–552.

[100] Alfred Karl Louis, Peter Maass, and Andreas Rieder. *Wavelets: Theory and Application*. Wiley, Chichester, 1997.

[101] Stéphane Mallat. *A Wavelet Tour of Signal Processing*. Academic Press, San Diego, CA, USA, 1999.

[102] Stéphane Mallat, Geoff Davis, and Marco Avelaneda. Adaptive greedy approximations. *Journal of Constructive Approximation*, 13(1):57–98, 1997.

[103] Stéphane Mallat, Geoff Davis, and Zhifeng Zhang. Adaptive time-frequency decompositions. *SPIE Journal of Optical Engineering*, 33(7):2183–2191, July 1994.

[104] Stéphane Mallat, Seema Jaggi, William C. Karl, and Alan S. Willsky. Silhouette recognition using high resolution pursuit. *Pattern Recognition*, 23(5):753–771, 1999.

[105] Stéphane Mallat and Zhifeng Zhang. Matching pursuits with time-frequency dictionaries. *IEEE Transactions on Signal Processing*, 41(12):3397–3415, December 1993.

[106] Sebastien Marcel, Philippe Jost, Pierre Vandergheynst, and Jean-Philippe Thiran. Face authentication using client-specific matching pursuit. IDIAP-RR 78, IDIAP, 2004.

[107] Matija Marolt. Networks of adaptive oscillators for partial tracking and transcription of music recordings. *Journal of New Music Research*, 33(1):49–59, 2004.

[108] Keith D. Martin. A blackboard system for automatic transcription of simple polyphonic music. Technical Report 385, M.I.T. Media Lab Perceptual Computing, July 1996.

[109] Boaz Matalon, Michael Zibulevsky, and Michael Elad. Improved denoising of images using modeling of the redundant contourlet transform. In *Proceedings of the SPIE conference wavelets*, volume 5914, July 2005.

[110] Vladimir G. Maz'ja. *Sobolev spaces*. Springer, Berlin etc., 1985. Translation from the Russian by T. O. Shaposhnikova.

[111] François Mendels, Jean-Philippe Thiran, and Pierre Vandergheynst. Matching pursuit-based shape representation and recognition using scale-space. *International Journal of Imaging Systems and Technology*, 6(15):162–180, 2006.

[112] James W. Miskin. *Ensemble learning for independent component analysis*. PhD thesis, Cambridge, Selwyn College, GB, 2000.

[113] Gianluca Monaci, Philippe Jost, Pierre Vandergheynst, Boris Mailhe, Sylvain Lesage, and Rémi Gribonval. Learning multi-modal dictionaries: Application to audiovisual data. In *Proceedings of the International Workshop on Multimedia Content Representation, Classification and Security*, Springer-Verlag LNCS series, September 2006.

[114] Gianluca Monaci, Philippe Jost, Pierre Vandergheynst, Boris Mailhe, Sylvain Lesage, and Rémi Gribonval. Learning multi-modal dictionaries. *IEEE Transactions on Image Processing*, 16(9):2272–2283, September 2007.

[115] James A. Moorer. *On the segmentation and analysis of continuous musical sounds by digital computer*. PhD thesis, Departement of Music, Stanford University, May 1975.

[116] Joseph F. Murray and Kenneth Kreutz-Delgado. An improved FOCUSS-based learning algorithm for solving sparse linear inverse problems. In *Conference Record of the 35th Asilomar Conference on Signals, Systems and Computers*. IEEE, November 2001.

[117] Koichi Niijima. Fast objects detection by variance-maximization learning of lifting wavelet filters. In *SPARSE'05*, Rennes, France, November 2005.

[118] Bruno A. Olshausen. Sparse coding of time-varying natural images. In *Proceedings of the International Conference on Independent Component Analysis and Blind Source Separation (ICA)*, pages 603–608, 2000.

[119] Bruno A. Olshausen. Learning sparse, overcomplete representations of time-varying natural images. In *IEEE International Conference on Image Processing*, Barcelona, Spain, September 2003.

[120] Bruno A. Olshausen and David J. Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, (381):607–609, 1996.

[121] Bruno A. Olshausen and David J. Field. Sparse coding with an overcomplete basis set: A strategy employed by V1? *Vision Research*, 37:3311–3325, 1997.

[122] Bruno A. Olshausen and David J. Field. Natural image statistics and efficient coding. *Comput. Neural Syst.*, 7:333–339, 1996.

[123] Bruno A. Olshausen and K. Jarrod Millman. Learning sparse codes with a mixture-of-gaussians prior. *Advances in Neural Information Processing Systems*, 12:841–847, 2000.

[124] Bruno A. Olshausen, Phil Sallee, and Michael S. Lewicki. Learning sparse image codes using a wavelet pyramid architecture. *Advances in Neural Information Processing Systems*, 13:887–893, 2001.

[125] Line Ørtoft Endelt and Anders la Cour-Harbo. Time-frequency distributions of music based on sparse wavelet packet representations. Submitted to: EURASIP Journal on Applied Signal Processing, Special Issue on Music Information Retrieval Based on Signal Processing, 2006.

[126] Y. C. Pati, R. Rezaiifar, and P.S. Krishnaprasad. Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition. In *Signals, Systems and Computers*, volume 1, pages 40–44, Pacific Grove, CA, USA, November 1993. Conference Record of The Twenty-Seventh Asilomar Conference.

[127] Lorenzo Peotta, Philippe Jost, Pierre Vandergheynst, and Pascal Frossard. Sparse approximation with block incoherent dictionaries. Technical Report TR-ITS-2003.007, EPFL Signal Processing Institute, Lausanne, December 2003.

[128] Donald B. Percival, James E. Overland, and Harold O. Mofjeld. Using matching pursuit to assess atmospheric circulation changes over the north pacific. submitted, May 2002.

[129] Mark D. Plumbley, Samer A. Abdallah, Thomas Blumensath, and Mike Davies. Sparse representations of polyphonic music. *EURASIP Signal Processing Journal*, 86(3):417–431, March 2006.

[130] Graham E. Poliner and Daniel P. W. Ellis. A discriminative model for polyphonic piano transcription. *EURASIP Journal on Advances in Signal Processing*, 2007, 2007.

[131] Matan Protter and Michael Elad. Image sequence denoising via sparse and redundant representations. Submitted to IEEE Transactions on Image Processing, January 2007.

[132] Ludger Prünte. Construction of wavelet-dictionaries. in preparation for submission, September 2007.

[133] Ludger Prünte, Peter Maass, and Henning Thielemann. Condition monitoring on linear guideways by discrete selflet transformation. Accepted for publication in *Signal Processing*, December 2007.

[134] M. Rao Raghuveer and Joseph O. Chapa. Algorithms for designing wavelets to match a specified signal. *IEEE Transactions on Signal Processing*, 48(12):3395–3406, 2000.

[135] Adel Rahmoune, Pierre Vandergheynst, and Pascal Frossard. Flexible motion-adaptive video coding with redundant expansions. *IEEE Transactions on Circuits and Systems for Video Technology*, 16(2):178–190, February 2006.

[136] Bhaskar D. Rao and Kenneth Kreutz-Delgado. Deriving algorithms for computing sparse solutions to linear inverse problems. In *Asilomar Conference on Signals, Systems and Computers*, volume 1, pages 955–959, Monterey, CA, USA, November 1997.

[137] Laura Rebollo-Neira and David Lowe. Optimised orthogonal matching pursuit approach. *IEEE Signal Processing Letters*, 9(4):137–140, 2002.

[138] August G. Ruß, K. Baalmann, M. Beier, Herbert Bretscher, Jochen Bretschneider, J. Haselhoff, Wolfgang Heuser, J. Miecke, E. Piwowarsky, István Sebestyén, Cornelia Tetzlaff, and Henryk Velde. *Linearlager und Linearführungssysteme. Einsatzmöglichkeiten – Berechnung – Auslegung*. Number 337 in Kontakt und Studium. Expert-Verlag, second edition, 2000.

[139] Phil Sallee and Bruno A. Olshausen. Learning sparse multiscale image representations. *Advances in Neural Information Processing Systems*, 15:1327–1334, 2003.

[140] Mark Sandler and Xue Wen. Transcribing piano recordings using signal novelty. In *Proceedings of 118th AES Convention*, Barcelona, May 2005.

[141] Phillipe Schmid-Saugeon and Avideh Zakhor. Dictionary design for matching pursuit and application to motion compensated video coding. *IEEE Transactions on Circuits and Systems for Video Technology*, 14(6):880–886, June 2004.

[142] Yair Shifman and Yehuda Leviatan. Iterative selection of expansion functions from an overcomplete dictionary of wavelet packets for impedance matrix compression. *Journal of Electromagnetic Waves and Applications*, 12(11):1403–1421, 1998.

[143] Eero P. Simoncelli, William T. Freeman, Edward H. Adelson, and David J. Heeger. Shiftable multi-scale transforms. *IEEE Transactions on Information Theory*, 38(2):587–607, March 1992.

[144] Karl Skretting, Kjersti Engan, and John Håkon Husøy. ECG compression using signal dependent frames and matching pursuit. In *Proceedings ICASSP 2005*, volume IV, pages 585–588, Philadelphia, PA, USA, March 2005.

[145] Karl Skretting, Kjersti Engan, John Håkon Husøy, and Sven Ole Aase. Sparse representation of images using overlapping frames. In *SCIA 2001*, Bergen, Norway, June 11–14 2001.

[146] Karl Skretting and John Håkon Husøy. Texture classification using sparse frame-based representations. *EURASIP Journal on Applied Signal Processing, special issue on "Frames and Overcomplete Representations in Signal Processing, Communications, and Information Theory"*, 2006.

[147] Jean-Luc Starck, Emmanuel J. Candès, and David L. Donoho. Astronomical image representation by the curvelet transform. *Astronomy and Astrophysics*, 398(2):785–800, February 2003.

[148] Jean-Luc Starck, Michael Elad, and David L. Donoho. Redundant multiscale transforms and their application for morphological component analysis. *The Journal of Advances in Imaging and Electron Physics*, 132:287–348, 2004.

[149] Jean-Luc Starck, Michael Elad, and David L. Donoho. Image decomposition via the combination of sparse representations and a variational approach. *IEEE Transactions on Image Processing*, 14(10):1570–1582, October 2005.

[150] Josef Stoer and Roland Burlisch. *Introduction to Numerical Analysis*. Number 12 in Texts in Applied Mathematics. Springer, New York, second edition, 1992.

[151] Thomas Strohmer and Robert W. Heath, Jr. Grassmannian frames with applications to coding and communcations. *Applied and Computational Harmonic Analysis*, 14(3):257–275, 2003.

[152] Wim Sweldens. The lifting scheme: A new philosophy in biorthogonal wavelet construction. In Andrew F. Laine and Michael A. Unser, editors, *Wavelet Applications in Signal and Image Processing III*, volume 2569 of *Proceedings of the SPIE Conference on Mathematical Imaging*, pages 68–79, 1995.

[153] Wim Sweldens. The lifting scheme: A custom-design construction of biorthogonal wavelets. *Applied and Computational Harmonic Analysis*, 3(2):186–200, 1996.

[154] Wim Sweldens. The lifting scheme: A construction of second generation wavelets. *SIAM J. Math. Anal.*, 29(2):511–546, 1998.

[155] Carl Taswell. The what, how, and why of wavelet shrinkage denoising. *Computing in Science and Engineering*, 2(3):12–19, 2000.

[156] David S. Taubman and Michael W. Marcellin. *JPEG 2000: Image Compression Fundamentals, Standards, and Practice*. Kluwer Academic Publishers, January 2002.

[157] Ahmed H. Tewfik and Massoud Alghoniemy. Pruned enumeration approach to bounded error subset selection. In *SPARSE'05*, Rennes, France, November 2005.

[158] Fabian J. Theis and Gonzalo A. García. On the use of sparse signal decomposition in the analysis of multi-channel surface electromyograms. *EURASIP Signal Processing Journal*, 86(3):603–623, March 2006.

[159] Henning Thielemann. Optimally matched wavelets. In *PAMM: Proceedings of Applied Mathematics and Mechanics*, volume 4, pages 586–587. Wiley VCH, 2004.

[160] Henning Thielemann. *Optimally matched wavelets*. PhD thesis, Universität Bremen, March 2006.

[161] Leonida Tonelli. *Fondamenti di calcolo delle variazioni*, volume 1–2. Nicola Zanichelli, Bologna, 1921–1923.

[162] Ivana Tosic, Pascal Frossard, and Pierre Vandergheynst. Progressive coding of 3D objects based on overcomplete decompositions. *IEEE Transactions on Circuits and Systems for Video Technology*, 16(11):1338–1349, August 2006.

[163] Joel A. Tropp. Greed is good: Algorithmic results for sparse approximation. *IEEE Transactions on Information Theory*, 50(10):2231–2242, October 2004.

[164] Pedro Vera-Candeas, Nicolás Ruiz-Reyes, Manuel Rosa-Zurera, Juan C. Cuevas-Martínez, and Francisco López-Ferreras. Fast implementation of an improved parametric audio coder based on a mixed dictionary. *EURASIP Signal Processing Journal*, 86(3):432–443, March 2006.

[165] Manfred Weck and Christian Brecher. *Werkzeugmaschinen*, volume 3 (Mechatronische Systeme, Vorschubantriebe, Prozessdiagnose). Springer VDI Verlag, Berlin, sixth edition, 2006.

[166] Heiko Wersing, Julian Eggert, and Edgar Körner. Sparse coding with invariance constraints. In *Proc. Int. Conf. on Art. Neur. Netw. ICANN 2003*, pages 385–392, Istanbul, 2003.

[167] Wikipedia. Bremen — Wikipedia, Die freie Enzyklopädie, 2008. `http://de.wikipedia.org/w/index.php?title=Bremen&oldid=40796697`, [Online; Stand 5. Januar 2008].

[168] David Wipf, Jason Palmer, Bhaskar D. Rao, and Kenneth Kreutz-Delgado. Performance evaluation of latent variable models with sparse priors. Hawaii, USA, April 2007. ICASSP.

[169] David Paul Wipf and Bhaskar D. Rao. Comparing the effects of different weight distributions on finding sparse representations. *Advances in Neural Information Processing Systems*, 18, 2006.

[170] Pei Yu and Yuan Yuan. A matching pursuit technique for computing the simplest normal forms of vector fields. *Journal of Symbolic Computation*, 35(5):591–615, 2003.

[171] Eberhard Zeidler. *Nonlinear Functional Analysis and its Applications*, volume III. Springer-Verlag, New York, 1984.

[172] Eberhard Zeidler. *Nonlinear Functional Analysis and its Applications*, volume I. Springer-Verlag, New York, 1985.

[173] Jian-Kang Zhang, Timothy N. Davidson, and K. Max Wong. Efficient design of orthonormal wavelet bases for signal representation. *IEEE Transactions on Signal Processing*, 52(7):1983–1996, July 2004.

[174] Ruohua Zhou. *Feature extraction of musical content for automatic music transcription.* PhD thesis, EPFL, Lausanne, Switzerland, 2006.