Regular And Irregular Gabor Multipliers With Application To Psychoacoustic Masking

Dissertation zur Erlangung des akademischen Grades eines Doktors der Naturwissenschaften

an der Fakultät für Mathematik

der Universität Wien

Verfasst von

Mag. Peter Balazs

Begutachtet von

Prof. Dr. H.G. Feichtinger

und

Prof. Dr. B. Torrésani

Wien, June 8, 2005 (printed)
July 18, 2005(revised)

Preliminaries

You read the revised version, as of of this PhD thesis. Experience (and Murphy's law) tells us that as soon as it is printed, the first errors will appear. If you find an error or have any comment, please contact the author, `peter.balazs@oeaw.ac.at`. For this work it was only !26 hours! until the first error was found. The author will try to keep an updated and corrected version available on the internet. For the time being this work can be found at

`http://www.kfs.oeaw.ac.at/xxl/dissertation/dissertation.pdf`

Motivation

The relevance of signal processing in today's life is clearly evident. Without exaggeration it can be said, that any advance in signal processing sciences directly lead to an application in technology and information processing. Without modern signal processing methods several modern technologies would not be possible, like the mobile phone, UMTS, xDSL or digital television.

In many scientific fields, for example in statistics and theoretical physics, it could be seen in the past, that scientists from different fields develop parallel and incoherent theories. This is highly inefficient from the research community's point of view. On the other hand *wavelet theory* had shown that, if theory and application, respectively mathematicians and engineers, work together, coherent results can be reached and through concentrated work a high synergy effect can be obtained.

Although the Fourier transformation and the short time Fourier transformation have been used for quite some time, only in the last couple of years a concentrated field, namely *"Mathematical Signal Processing"*, was defined and pursued. The connection between application and theory in the so called Gabor theory has lead to many interesting results. This work should be seen right at this connection.

Gabor analysis is the mathematical name for a sampled version of the so called *Short-Time Fourier Transformation*, which is a time-frequency analysis method. This mathematical subfield allows the answer of many questions, which are relevant for applications, e.g. how the parameters of an analysis-synthesis system can be chosen, such that perfect reconstruction can be achieved. As another example, Gabor theory is currently used to develop the background for a standardized duplex scheme for vDSL (very high bit rate digital subscriber lines).

Many applications use a modification on the coefficients obtained from the analysis operation. An example of this is an equalizer, which uses a transformation into the frequency domain, modifies the obtained coefficients and then a synthesizer transforms the result back into a time domain signal.

If the modification of the time-domain coefficients is done by multiplying them with a function in the frequency domain, the whole process is called *time-invariant filtering*. This technique have been used for many years and finds a wide range of applications, e.g., to improve the sound of telephone communications. A generalization of this technique is the so called *time-variant filtering*, which has got more and more attention in the last couple of years. The so called Gabor multipliers are particular cases of time-variant filters. In this case, the signal to be processed is transformed into the time-frequency domain and the resulting coefficients are multiplied by a function on the same domain.

A frequently used and publicly well-known technology is the *MP3*-format for audio file. This is an encoding / decoding scheme in the MPEG1/MPEG2 (Layer 3) standard. This technique is used to reduce the digital size of a sound signal. It is based on a coder, which uses a model of the human audio perception. It is known in psychoacoustics that not all the information contained in an audio signal can be perceived by the human ear. More precisely, if the audio signal is transformed into the time-frequency domain , it turns out that some time-frequency components mask (i.e., hide) other components which are close in time or frequency. Clearly filtering out this components will result in reducing the memory space required to store the signal, without any subjective quality loss. An idea of how to extend known masking algorithms to a time-frequency model is given at the end of this work. This is done by using Gabor multipliers. As the linear frequency scale *Hz* is not very well-fitted to the auditory perception, another frequency sampling, following the *Bark* scale, is chosen. This leads to irregular Gabor multipliers. As many powerful mathematical tools are lost by giving up the group structure of regular sampling, the author decided to first investigate an even more general case, namely, *frame multipliers*.

Approach

The goal of this work is to span the whole arc from mathematical theory to application. We will start with the theory of frame multipliers, proceeding with Gabor multipliers (especially for irregular systems), investigating the numerics of the discrete Gabor analysis and applying the theory to give an idea for a time-frequency masking algorithm. A big goal of this work is to develop the connection between theory and applications and therefore the implementation of algorithms is a natural goal.

We will investigate the mathematical background for a possible implementation of a time-frequency masking filter. As the human perception is not very well fitted to a regular frequency sampling, we will consider irregular Gabor multipliers. We will introduce a generalization of Gabor multipliers, namely frame multipliers, intended to provide the background for other possible representation for the auditory system. With the popularity of wavelets, irregular Gabor frames, multi Gabor frames or other analysis systems like Gammatone filters, it is worthwhile to investigate the most general class of these operators.

As mentioned above, this work deals with theoretical results as well as computational issues. Nearly all results in this work are linked to some algorithm, and therefore, we concentrate on the analysis of finite-dimensional spaces. In Chapter 4, we will focus on a potential application, namely, psychoacoustical masking.

Historical Remarks

The name "Gabor analysis" is a rather recent one, but the idea goes back quite some while. In engineering the Fourier transformation was used extensively, especially after the development of the very efficient FFT-algorithm [27]. For application in music or speech processing it is necessary to get a joint time-frequency representation. For example the phase vocoder [56] has been used as early as in the 60ties.

Dennis Gabor investigated in [60] the representation of a one dimensional signal in two dimensions, time and frequency. He suggested to represent a function by a linear combination of translated and modulated Gaussians. Interestingly there is a tight connection of this approach to quantum mechanics, c.f. e.g. [57]. The most prominent connection is the uncertainty principle, which is very important in both fields.

The concept of a time frequency representation based on the FFT was made more concrete and the *Short Time Fourier Transformation (STFT)* was introduced, c.f. e.g. [3]. On the mathematical side the representation of functions by other functions was further investigated and leaded to the theory of *atomic decomposition*, for example by Feichtinger and Gröchenig, refer e.g. to [40]. With time the STFT became a widely used tools. Apart from the uncertainty principle another disadvantage of this technique is the high redundancy. Instead of using the whole STFT a sampled version is used for resynthesis and this is what today is understood as *Gabor analysis*. With the advent of wavelet theory, cf. [29], and the general interest to investigate the theory of signals due to the new telecommunication applications Gabor theory and applications have become an important field of applied mathematics. For example Wexler and Raz investigated in [131] how to use Gabor analysis in applications and algorithms. A fundamental property was shown there, the duality principle, which reduces the question of perfect reconstruction to a simple set of equations. Today Gabor analysis and the closely related wavelet theory are one of the mathematical fields, where theory and application, mathematicians and engineers work closely together. For example the equivalence between Gabor analysis and filter-bank approaches was shown in [14].

From the applications of Gabor theory it soon become apparent that the notion of an orthonormal basis is not always useful. Sometimes it is more important for a decomposing set to have special properties, like good time frequency localization, than to have unique coefficients. This leaded to the concept of frames, which was introduced by Duffin and Schaefer in [36]. It was made popular by Daubechies, c.f. [29], and today is one of the most important foundations of Gabor theory. In application frames became more and more attention, in the form of *oversampled filter banks*, c.f. e.g. [14]. With this theory many questions can be formulated in a very clear and precise way. For example the question whether a filter bank yields perfect reconstructions can be translated to the search for a dual frame.

Filters are a common tool in signal processing. They correspond to time-variant operators. Clearly there are also time-variant systems, refer for example to [70]. Gabor multipliers are special cases for such operators. They are a natural extension of filters, which are operators where the spectrum is multiplied with fixed coefficients. Gabor multipliers are operators, where the time frequency coefficients are multiplied by a fixed time-frequency pattern. They have been investigated most prominently by Feichtinger for example in [47]. These operators have been used in engineering implicitly for quite some time, for a recent application in seismic imaging see for example [93].

The effect of psychoacoustical masking is well-known cf. [137]. A lot of publications in the last 25 years dealt with this effect, refer for example to [39] or [109]. One of its most important application is the psychoacoustical model of the MP3 coding scheme see e.g. [79]. In this work we will investigate an extension of the irrelevance filter found in [37] and implemented in ST^X [96], a signal processing software system programmed at the Acoustics Research Institute Vienna, to a time-variant filter, that models both frequency and temporal masking.

Main results

As usual all results in this work with given proofs respectively without a citation are original work. A few of them are well-known, but had to be proved under different assumptions. We will summarize the most important results in the following sections:

Mathematical Theory:

Interesting new results from a theoretical mathematical point of view are the following:

We are going to introduce the concept of frame multipliers, a generalization of the idea of Gabor multipliers. This idea will be formulated for Bessel sequences, frame sequences and Riesz sequences. Two main theorems will be proved. One is dealing with the connection of the symbol to the operator. Most notably if the symbol is in the sequence space l^∞, l^2 or l^1 respectively, then the multiplier is a bounded, trace class or Hilbert-Schmidt operator respectively. The other main result is the continuous dependency of the operators on symbol and frames, where the measurement of the similarities of frames has to be chosen in the right way. We are going to investigate other connections of frames and operators, for example how an operator can be described by a matrix using frames. We will investigate multipliers for Riesz sequences and we will see, that in this case these operator can be described uniquely be their symbols.

As mentioned before, we nearly always intend to implement an algorithm connected to the results. Therefore the investigation of frames in connection with finite dimensional spaces is investigated. In particular we will show that it is possible to classify finite-dimensional spaces and, connected to that, Hilbert-Schmidt operators by frames.

We are going to investigate Gabor multipliers for irregular Gabor systems. We will investigate the irregular Gabor systems and, for example, will show directly that for relatively separated irregular lattices, the Gabor system with a window in S_0 (i.e., the Feichtinger's Algebra) forms a Bessel sequence.

We will use the developed theory for frame multipliers for irregular Gabor multipliers. Moreover we will use the special coherent structure of these frames for other results. Most importantly we will show that under the right conditions the continuous dependency of the multipliers can be extended to the connection to the symbol, the atoms and the lattice. And for these results a 'Jitter-like' norm suffices.

We will investigate the Gabor analysis for the finite-dimensional case very thoroughly. We will look into the theory of block matrix important for Gabor analysis. We will show that they form matrix algebras and are connected to each other using the Matrix Fourier Transformation. We will show a tight connection between the well-known representations of a Gabor frame matrix by the so-called 'non-zero' block matrix and the Janssen matrix. Based on that we will introduce two new matrix norms and investigate the equivalences between them.

We will dedicate a full section to an article by Thomas Strohmer [122], which is a perfect starting point for the investigation of Gabor algorithms. We have found a few small errors in this article, which we will correct here in this work.

Computational Aspects:

From a more computational point of view the following original statements should be highlighted:

For the general frame case we are comparing different ways to calculate the inner product of a matrix with the Kronecker product matrix of two vectors. This is important for the approximation of any matrix by a frame multiplier for a fixed frame. An algorithm for this will be presented in this work. It will be programmed in MATLAB and can be found in the appendix.

Several MATLAB algorithms will also be implemented for Gabor systems, e.g. the calculation of an irregular Gabor family. Also an algorithm for the approximation of an arbitrary matrix by the Gabor multiplier of two irregular Gabor families is going to be presented. It will be compared to existing algorithms for regular lattices.

For the inversion of Gabor frame matrices we will present a method using the special sparse structure of it to find a fast algorithm. This program uses double preconditioning and numerical experiments will be done to investigate the efficiency of this method. In particular it will be compared to the methods using only single preconditioning by projection on circular or respectively diagonal matrices. We will especially examine how well and in which case the preconditioning matrix itself is already a good approximation of the inverse matrix.

Application In Psychoacoustics:

We will introduce a concept of how to extend an existing masking algorithm, which only incorporates simultaneous frequency masking, to a time-frequency model. This will be an irregular Gabor multiplier with coefficients 1 or 0.

Organization

This work is organized as follows.

- In Chapter 1 we will investigate the general theory of frames, which will be used in our development of the Gabor theory. After a thorough introduction to frame theory, special emphasis will be given to the investigation of the connections between frames and operators, as well as the relationship between finite-dimensional spaces and frames. The new concept of frame multipliers, a generalization of Gabor multipliers, will be also introduced and investigated. We will address basic questions like the dependency of the operator on the symbol.

- In Chapter 2 the Gabor theory is investigated, especially the irregular case. Special attention will be given to Gabor multipliers, again with special emphasis on irregular sampling. The problems we will investigate include the continuity of the dependency of this operators on the windows, symbol and lattice. We will also introduce an algorithm for the approximation of any matrix by irregular Gabor multipliers.

- In Chapter 3 will introduce the discrete final-dimensional Gabor analysis. We will investigate special types of matrices important for Gabor analysis. In connection with the well-known special structure of the Gabor frame matrix, we will introduce two new norms, which are upper bounds for the operator norm and which can be calculated in a numerically very efficient way. For the regular case, we will introduce a new method to approximately invert the Gabor matrix by using Double Preconditioning.

- In the last chapter, i.e., Chapter 4, we will introduce the basic ideas of human auditory perception and masking filters. Although this is a mathematical work, we will introduce a concept for a method on how to extend the masking filter of the program ST^X to a time-frequency filter incorporating simultaneous and temporal masking. This concept was developed with the help of a psychoacoustican.

- In the appendix we provide the required mathematical background, with the aim of making this work "more" self-contained. The appendix also contains the algorithms developed in Chapters 1 – 3.

Thanks

There are many groups, I'd like to thank.

I have to thank my teachers, who have developed my interest in maths during school and studies. As an example I mention Gerhard Kowol, the adviser of my master thesis.

I have been privileged by being a *NuHAG* member. I have to thank all members for a lot of fruitful discussions, mentioning as examples Mario Hampejs, Ole Christensen, Peter Soendergaard and Monika Dörfler. I have to single out Tobias Werther, who was always ready and eager to help.

With the *HASSIP* EU network I could again enjoy a big privilege, to be able to go to another country and getting to know the people and their way of scientific work. I have to thank many HASSIP members for interesting discussion on many very interesting workshops and conferences, mentioning as examples Hartmut Führ, Remi Gribonval and Jean-Pierre Antoine.

I have enjoyed my stay in Marseille from November 2003 to May 2004 with HASSIP tremendously. For that, the willingness to discuss, answer questions, but also solve non-scientific problems (like administration and the like) as well as his readiness to be a reviewer of my PhD I have to thank Bruno Torrésani a lot. At the *LATP, CMI Marseille* everybody gave me a warm welcome, I still miss them all, but have to specially mention Florent Jaillet and Damian Marelli with whom I had a lot of discussions. At the *LMA, CNRS* apart from a good time, I could work further on the cooperation between mathematical theory and application. I have to especially mention Richard Kronland-Martinet and Solvi Ystad with the hope that in the future we can intensify the exchange between our institutes.

At the Acoustics Research Institute I was given the opportunity to merge my mathematical interests with the application in acoustics. I had interesting discussions with everybody. I have to mention Piotr Majdak for a lot of scientific discussions, Berhard Laback for helping me understand psychoacoustics and especially Werner A. Deutsch and the Academy of Sciences, who gave me the chance to work on mathematical theory in the context of acoustics research. In my opinion this is a perfect connection of theory and application.

Although this not directly connected to this work, I have to thank all my friends, who have helped me enjoy the life apart from mathematics, including playing baseball with the Tulln Ravens, enjoying many evenings roleplaying with friends, playing music and enjoying life with 'Oberst' and band, and last but not least distracting me from work in Marseille by the "gang of Marseille". Let me single out Thomas Krutzler, who has proof-read this work.

I have to thank my family, especially my mother, for supporting me all my life.

I have to specially thank Hans G. Feichtinger for everything. He gave me the chance to choose a PhD topic, which fits my work at the ARI, he was always open for questions and problems, he gave me the chance to participate in the HASSIP network and tried to support me in any way possible.

Finally, this work is dedicated to my wife, *Claudia*. I can't express, how much I want to thank her. Let me try it like that: If I were a Cauchy sequence, I would certainly converge with her, as she makes my life *complete*.

Contents

Chapter 1

Frame Multiplier

" [A company] designed this brilliant open frame multiplier to meet the exacting criteria of the demanding tournament casting circuit and the discerning UK beach angler. [...]"

(Taken from `http://www.fishingmegastore.com`)

Figure 1.1: A Frame Multiplier

The application of signal processing algorithms are numerous, many of them adaptive or time variant filters, for example the implementation of a psychoacoustic masking filter, as in Section 4.1.2. If the STFT, the Short Time Fourier Transformation, refer to Section 2.1.1, is used in its sampled version, the Gabor transform, one possibility for time variant filter is the usage of *Gabor multipliers*, see Section 2.3. Gabor multipliers are a current topic of research (cf. e.g. [47] and [34]). For these operators the Gabor transform is used to calculate time frequency coefficients, they are multiplied with a fixed time-frequency mask and then the result is synthesized, see chapter 2.3. These operators have been used for quite some time implicitly. Recent applications are for example in system identification, see [85].

If another way of calculating these coefficients is chosen or if another synthesis is used, many modifications can still be seen and implemented as multipliers. So for example it seems quite natural to define the equivalent for wavelet frames, something like a *wavelet multiplier*.

Also as irregular Gabor frames get more and more attention (see e.g. [82]), Gabor multiplier on irregular lattices can be investigated, refer to Section 2.5. There the group structure of the lattice is non-existent and so cannot be exploited. So it is quite natural to look on frames without any further structure first.

The formulation of a concept of a multiplier for other analysis / synthesis systems like e.g. Gammatone filter banks (e.g. refer to [67]), which are mainly used for analysis based on the auditory system, is possible and useful. In [100] a Gammatone filter bank was used for analysis and synthesis, for the sound separation part a neuronal network creates a frame multiplier for these coefficients.

To have perfect reconstruction / synthesis seems in all theses cases to be valuable and a frame would give this possibility. The added restriction, needed for frames, that the l^2-Norm of the coefficients should be an equivalent norm, seems very natural.

So it seems useful to group all these operators in a more general concept than the Gabor multiplier: the *frame multiplier*, which we will investigate in section 1.3.

We start this chapter with a general introduction to the theory of frames.

1.1 Frames

1.1.1 Introduction

Definition 1.1.1 *The sequence $\mathcal{G} = (g_k | k \in K)$ is called a* **frame** *for the (separable) Hilbert space $\mathcal{H}$, if constants $A, B > 0$ exist, such that*

$$A \cdot \|f\|_{\mathcal{H}}^2 \leq \sum_k |\langle f, g_k \rangle|^2 \leq B \cdot \|f\|_{\mathcal{H}}^2 \ \ \forall \ f \in \mathcal{H} \tag{1.1}$$

Definition 1.1.2 *In the above definition A is called the* **lower***, B the* **upper frame bound***. If the bounds can be chosen such that $A = B$ the frame is called* **tight***. If $\mathcal{G}$ ceases to be a frame, if any one element is removed, then the frame is called* **exact***. If the frame is not a basis, it is called* **overcomplete***. If $\|g_k\|_{\mathcal{H}} = 1$ for all k, then the frame is called* **normalized***. basis, it is called* **overcomplete***. If $\|g_k\|_{\mathcal{H}} = 1$ for all k, then the frame is called* **normalized***.*

The index set will be omitted in the following, if no distinction is necessary. In 1.1.32 we will see that the properties exact and overcomplete are mutually exclusive. Lemma [23] 5.1.7. shows that it is sufficient for a sequence to be a frame to fulfill the frame condition on a dense subspace.

Definition 1.1.3 *If a sequence (g_k) fulfills the "upper frame condition"*

$$\sum_k |\langle f, g_k \rangle|^2 \leq B \cdot \|f\|_{\mathcal{H}}^2 \ \forall \ f \in \mathcal{H}$$

it is called **Bessel sequence***.*

If a sequence (g_k) fulfills the frame condition for its closed linear span, then it is called a **frame sequence***, i.e.*

$$A \cdot \|f\|_{\mathcal{H}}^2 \leq \sum_k |\langle f, g_k \rangle|^2 \leq B \cdot \|f\|_{\mathcal{H}}^2 \ \forall \ f \in \overline{span\{g_k\}}$$

In a finite-dimensional space, see Section 1.2, clearly every subset of a frame is a frame sequence. But in general Hilbert spaces, this is not true anymore, see [23] Section 6.2. Also in the general, infinite-dimensional case not every frame can be thinned out to a basis, as there are frames, which do not contain a basis, see [23] Section 6.4. This shows that some expectations, which arise from experience with finite dimensional bases, have to be dropped.

As a direct consequence of the definition of Bessel sequence we can show

Lemma 1.1.1 *Let (f_k) be a Bessel sequence for $\mathcal{H}$. Then*

$$\|f_k\|_{\mathcal{H}} \leq \sqrt{B}.$$

Proof: For f_{k_0} use the inequality

$$\sum_k |\langle f_{k_0}, f_k \rangle|^2 \leq B \cdot \|f_{k_0}\|_{\mathcal{H}}^2$$

$$\Longrightarrow \|f_{k_0}\|_{\mathcal{H}}^4 + \sum_{k \neq k_0} |\langle f_{k_0}, f_k \rangle|^2 \leq B \cdot \|f_{k_0}\|_{\mathcal{H}}^2$$

$$\|f_{k_0}\|_{\mathcal{H}}^4 \leq \|f_{k_0}\|_{\mathcal{H}}^4 + \sum_{k \neq k_0} |\langle f_{k_0}, f_k \rangle|^2 \leq B \cdot \|f_{k_0}\|_{\mathcal{H}}^2$$

and so (wlog $f_{k_0} \neq 0$)

$$\|f_{k_0}\|_{\mathcal{H}}^2 \leq B$$

□

This result for frames is also part of Proposition 1.1.16.

It is an interesting property of frames, that by removing an element you cannot keep completeness while losing the frame property:

Proposition 1.1.2 ([23] 5.4.7) *If you take out one element of a frame, the reduced sequence will either form a frame again or be incomplete.*

Let us look at a simple example:

Example 1.1.1 :

Let $\{e_i\}$ and $\{e'_j\}$ be two disjoint ONBs for the Hilbert space $\mathcal{H}$. Then $\{g_k\} = \{e_i\} \cup \{e'_i\}$ is a tight frame with the frame bound $A = 2$.

$$\sum_k |\langle f, g_k \rangle|^2 = \sum_i |\langle f, e_i \rangle|^2 + \sum_j \left|\langle f, e'_j \rangle\right|^2 = \|f\| + \|f\| = 2 \cdot \|f\|$$

1.1.2 The Frame Operator

Definition 1.1.4 *Let $\mathcal{G} = \{g_k\}$ be a frame in $\mathcal{H}$. Then let $C_{\mathcal{G}} : \mathcal{H} \to l^2(K)$ be the* **analysis operator**

$$C_{\mathcal{G}}(f) = \{\langle f, g_k \rangle\}.$$

Let $D_{\mathcal{G}} : c_c(K) \to \mathcal{H}$ be the **synthesis operator**

$$D_{\mathcal{G}}(\{c_k\}) = \sum_k c_k \cdot g_k.$$

Let $S_{\mathcal{G}} : \mathcal{H} \to \mathcal{H}$ be the **frame operator**

$$S_{\mathcal{G}}(f) = \sum_k \langle f, g_k \rangle \cdot g_k.$$

If it is not necessary to distinguish different frames, and it is clear, which frame is used, we will just write S for $S_{\mathcal{G}}$, C for $C_{\mathcal{G}}$ and D for $D_{\mathcal{G}}$. We will also use the indexing C_{g_k} for $C_{\mathcal{G}}$ and also the other operators.

For a given frame C and D are clearly linear and on c_c, the sequence space of finite sequences, the equality $C^* = D$ is true. It can be easily shown that the operator C is bounded and injective. (See also Section 1.1.6). This is just a rewriting of the frame property from Definition 1.1 as

$$A \cdot \|f\|_{\mathcal{H}}^2 \leq \|C(f)\|_2^2 \leq B \cdot \|f\|_{\mathcal{H}}^2$$

which is equivalent, see appendix A.4.3.3, to C being bounded and having a bounded inverse (on $ran(C)$).

So D can be extended to a function $D_{\mathcal{G}} : l^2(K) \to \mathcal{H}$ with $C^* = D$ on l^2. Even more:

Theorem 1.1.3 ([63] 5.1.1) *Let $\mathcal{G} = \{g_k\}$ be a frame for $\mathcal{H}$.*

1. *C is a bounded, injective operator with closed range with $\|C\|_{op} \leq \sqrt{B}$*
2. *C and D are adjoint to each other, $D = C^*$ and so $\|D\|_{op} = \|C\|_{op} \leq \sqrt{B}$. The series $\sum\limits_k c_k \cdot g_k$ converges unconditionally.*
3. *$S = C^*C = DD^*$ is a positive invertible operator satisfying $AI_{\mathcal{H}} \leq S \leq BI_{\mathcal{H}}$ and $B^{-1}I_{\mathcal{H}} \leq S^{-1} \leq A^{-1}I_{\mathcal{H}}$.*
4. *The optimal frame bounds are $B_{opt} = \|S\|_{Op}$ and $A_{opt} = \|S^{-1}\|_{Op}^{-1}$.*

From 1.1.3 we know that $AI_{\mathcal{H}} \leq S \leq BI_{\mathcal{H}}$ and therefore $A \leq \|S\|_{Op} \leq B$ as $\|T\|_{Op} = \sup\limits_{\|f\|_{\mathcal{H}} \leq 1} \{\langle Tf, f\rangle\}$ for positive operators T.

$$A \leq \|S\|_{\mathcal{H}} = \|C^* \circ C\|_{\mathcal{H}} \leq \|C\|_{\mathcal{H}}^2$$

So $\sqrt{A} \leq \|C\|_{\mathcal{H}}$.

Corollary 1.1.4 *Let $\mathcal{G} = \{g_k\}$ be a frame for $\mathcal{H}$. Then the operator norm of the analysis operator C is bounded by roots of the frame bounds A, B:*

$$\sqrt{A} \leq \|C\|_{Op} \leq \sqrt{B}$$

Ignoring the fact that we don't have a frame, we define

Definition 1.1.5 *Let $\{g_k\}$ and $\{\gamma_l\}$ be two sequences, then we call*

$$C_{g_k}(f) = (\langle f, g_k \rangle)$$

the **associated analysis operator***,*

$$D_{g_k}(c) = \sum_k c_k \cdot g_k$$

the **associated synthesis operator***, and*

$$S_{g_k,\gamma_k} f := \sum_k \langle f, g_k \rangle \gamma_k$$

the **associated frame operator***.*

These definitions are possibly not well-defined. In Section 1.1.6 it is shown that they certainly are for Bessel sequences. We will omit the word 'associated' if there is no confusion possible.

If the sequences are not frames or they are different from each other, the operator S is clearly not a true frame operator, but it shares a lot of properties, like this simple one:

Lemma 1.1.5 *Let $\{g_k\}$ be a Bessel sequences, then S_{g_k,g_k} is self-adjoint and positive semi-definite and positive definite on $\overline{span\{g_k\}}$.*

Proof: S is clearly well-defined as C and D are.

$$\langle Sf, g \rangle = \sum_k \langle f, g_k \rangle \langle g_k, g \rangle = \langle f, Sg \rangle$$

$$\langle Sf, f \rangle = \left\langle \sum_k \langle f, g_k \rangle g_k, f \right\rangle = \sum_k \langle f, g_k \rangle \langle g_k, f \rangle = \sum_k |\langle f, g_k \rangle|^2 \geq 0$$

If $\sum\limits_k |\langle f, g_k \rangle|^2 = 0 \Longrightarrow \langle f, g_k \rangle = 0 \ \forall g_k \Longrightarrow f \in span\{g_k\}^\perp$. $\square$

This stays true, if we do not know anything about the sequence except that the associated frame operator is well-defined.

1.1.3 Union Of Frames

The union of two frames is clearly a frame again. Let (h_k) and (g_i) be the two frames with A_h, A_g as lower and B_h, B_g as upper frame bounds. Then

$$A_h \|f\|_{\mathcal{H}}^2 \leq \sum_k |\langle f, h_k \rangle|^2 \leq \sum_l |\langle f, h_l \rangle|^2 + \sum_k |\langle f, g_k \rangle|^2$$

So A_h is a lower bound for the union. If a 'more tight' bound is desirable other options are $\max\{A_h, A_g\}$ or $A = A_h + A_g$.

On the other hand

$$\sum_k |\langle f, h_k \rangle|^2 + \sum_k |\langle f, g_k \rangle|^2 \leq B_h \|f\|_{\mathcal{H}}^2 + B_g \|f\|_{\mathcal{H}}^2 = (B_h + B_g) \|f\|_{\mathcal{H}}^2$$

For countable many frames a sufficient condition for the union to be a frame again is $\sum_i B_i < \infty$, if the sums of the upper frame bounds are summable . The lower bound is fulfilled by any A_i. This means that even a union of a frame with countable many Bessel sequences is a frame again.

A much more interesting question is, when is a union of parts of frames a frame again, see e.g. *quilted Gabor frames* [35] or *time-frequency jigsaw puzzle* [72]. In these studies a frame decomposition is searched, where different (Gabor) frames are used on a local level, so intuitively on certain parts of the time-frequency plans one frame is used and the others are disregarded.

But clearly the upper frame conditions is not the problem in these cases.

Lemma 1.1.6 *Let $(f_k | k \in K)$ and $(g_i | i \in I)$ be two Bessel sequences for $\mathcal{H}$ with bounds B_1 and B_2, let $K_1 \subseteq K$, $I_1 \subseteq I$. Then $\{h_j\} = \{f_k | k \in K_1\} \cup \{g_i | i \in I_1\}$ is a Bessel sequence with bound $B_1 + B_2$.*

Proof:

$$\sum_j |\langle f, h_j \rangle|^2 = \sum_{k \in K_1} |\langle f, f_k \rangle|^2 + \sum_{i \in I_1} |\langle f, g_i \rangle|^2 \leq$$

$$\leq \sum_{k \in K} |\langle f, f_k \rangle|^2 + \sum_{i \in I} |\langle f, g_i \rangle|^2 \leq B_1 \|f\|_{\mathcal{H}}^2 + B_2 \|f\|_{\mathcal{H}}^2$$

□

Clearly this can be extended to any finite number of frames and the result stays valid for a countable number of frames, where $\sum_i B_i < \infty$.

So the problem that remains (for the union of parts of frames) is the lower frame bound. See Section 1.1.6 for classifications, a Bessel sequence fails to be a frame if the synthesis (respectively analysis) operator does not have closed span or it is not injective (respectively surjective).

In the finite dimensional case this just means that the kernel of the synthesis operator is not zero, that means there is a "hole" where the coefficients are zero, i.e. there are functions "living in this hole", non-zero functions whose coefficients are zero.

For an example look at the above example 1.1.1, where $e_i' = \gamma_i \cdot e_i$. Then it is clear if only parts of the two ONBs are used, we have to either take e_i or e_i' for every i. So this gives rise to the naive, intuitive notion that we have to pick the parts carefully, such that information is not lost. But in the cases of over-complete frames this decision is hard to make, for more information see [35] and [72].

1.1.4 Dual Frames

If we have a frame in $\mathcal{H}$, we can find an expansion of every member of $\mathcal{H}$ with this frame:

Theorem 1.1.7 ([63] 5.1.3) *Let $\mathcal{G} = (g_k)$ be a frame for $\mathcal{H}$ with frame bounds A, $B > 0$. Then $\tilde{\mathcal{G}} = (\tilde{g}_k) = (S^{-1} g_k)$ is a frame with frame bounds B^{-1}, $A^{-1} > 0$, the so called* **canonical dual frame**. *Every $f \in \mathcal{H}$ has a (possibly non-orthogonal) expansions*

$$f = \sum_{k \in K} \langle f, S^{-1} g_k \rangle \, g_k$$

and

$$f = \sum_{k \in K} \langle f, g_k \rangle \, S^{-1} g_k$$

where both sums converge unconditionally in $\mathcal{H}$.

Any sequence of elements for which synthesis works are called *dual*, i.e. (γ_k) is dual to (g_k) if and only if for every $f \in \mathcal{H}$ we get

$$f = \sum_{k \in K} \langle f, g_k \rangle \, \gamma_k = \sum_{k \in K} \langle f, \gamma_k \rangle \, g_k$$

For the frame $\{g_k\}$ the inverse frame operator is just the frame operator of the dual frame:

$$S_{\{g_k\}}^{-1} f = S_{\{g_k\}}^{-1} \left(\sum_k \langle f, S^{-1} g_k \rangle \, g_k \right) = \sum_k \langle f, S^{-1} g_k \rangle \, S^{-1} g_k = S_{\{S^{-1} g_k\}} f$$

So every member of the Hilbert space $\mathcal{H}$ with a frame can be written as countable linear combination of the (countable) frame. So

Corollary 1.1.8 *Let $\{g_k\}$ be a frame for the Hilbert space $\mathcal{H}$ then $\mathcal{H}$ is separable.*

In contrast to orthonormal bases, these expansions are not unique, but canonical in the following sense:

Proposition 1.1.9 ([63] 5.1.4) *Let $\mathcal{G} = \{g_k\}$ be a frame for $\mathcal{H}$ and $f = \sum\limits_{k\in K} c_k g_k$ for some $\{c_k\} \in l^2$, then*

$$\sum_{k\in K} |c_k|^2 \geq \sum_{k\in K} \left|\left\langle f, S^{-1} g_k \right\rangle\right|^2$$

with equality only if $c_k = \langle f, S^{-1} g_k\rangle$ for all $k \in K$.

More precise ([23] 5.4.2)

$$\sum_{k\in K} |c_k|^2 = \sum_{k\in K} \left|\left\langle f, S^{-1} g_k \right\rangle\right|^2 + \sum_{k\in K} \left|c_k - \left\langle f, S^{-1} g_k \right\rangle\right|^2$$

Duals can also be used to see a property similar to properties of ONBs regarding the inner product:

Lemma 1.1.10 *Let (g_k) be a frame for $\mathcal{H}$ with a dual (γ_k). Then for all $f, g \in \mathcal{H}$*

$$\langle f, g\rangle = \sum_k \langle f, \gamma_k\rangle \langle g_k, g\rangle$$

Proof: As $f = \sum\limits_k \langle f, \tilde{g}_k\rangle g_k$

$$\langle f, g\rangle = \sum_k \langle f, \tilde{g}_k\rangle \langle g_k, g\rangle$$

□

Of course the roles of the dual and the original frame can be switched. This is also point (iii) in [23] Lemma 5.6.2.

1.1.5 Why Are Frames Useful?

Why should we use frames in the first place? Why not use ONBs? (Because we now know that the Hilbert spaces are separable, so ONBs exist.)

A short answer: Error-robustness and flexibility.

Error-robustness or redundancy: Take the example of signal transmissions. Analysis with ONBs gives non-redundant, independent data. If this data is distorted, for example by noise or transmission errors, there is no way that the original signal can be reconstructed (without a priori knowledge). In redundant systems, like frames that are not bases, there may be a chance to reconstruct the signal. Redundant systems may be error resistant, bases cannot. (Take as a practical example the internet, which is a highly redundant system. Signals have many different possible routes to be exchanged. This also means that it is very error resistant, as the loss of one or a few of these routes does not result in a disturbance of the data.)

Flexibility or degrees of freedom: Sometimes you would like to have certain properties, while you don't mind losing others. For the case of frames for some application the uniqueness of the coefficients is not important, but some other properties are, like a good time frequency behavior in the Gabor frame case. With bases you don't have a lot of freedom, with frames, which are a bigger class, you have more options.

For a longer answer have a look at chapter 4 of O. Christensen's book [23].

1.1.6 Classification

As mentioned in Section 1.1.2 the frame property is equivalent to C being injective and bounded. But also Bessel and frame sequences as well as frames can be classified by using the synthesis or analysis operator. We collect the results from [23] chapter 3 and chapter 5 as well as in [19] and [21] into a compilation and extend them in a natural way to all (in this context) possible combination of operators and sequences:

Theorem 1.1.11

1. *A sequence (g_k) is a Bessel sequence with bound B if and only if the synthesis operator operator*

$$D : l^2 \to \mathcal{H} \text{ with } D(c_k) = \sum_k c_k g_k$$

is well defined and bounded from l^2 in $\mathcal{H}$ with $\|D\|_{Op} \leq \sqrt{B}$.

2. *A sequence (g_k) is a frame sequence if and only if it is a Bessel sequence and D has closed range.*

3. *A sequence (g_k) is a frame if and only if it is a frame sequence and D is surjective.*

4. *A sequence (g_k) is a frame if and only if it is a Bessel sequence and D is surjective.*

Proof: 1.) [23] 3.2.3 states, that the sequence is a Bessel sequence if and only if D_{g_k} is bounded.

2.) See [21] 4.4.

3.) [23] 5.2.1 states, that a frame sequence is a frame if and only if C_{g_k} is injective. This is equivalent to $D_{g_k} = C_{g_k}^*$ having a dense range. If D_{g_k} is surjective, this is certainly true.

4.) If D is surjective, it has closed range. So the last item is clearly equivalent to the third one. □

We can now state the same result for the adjoint operator:

Theorem 1.1.12 1. *A sequence (g_k) is a Bessel sequence with bound B if and only if the analysis operator*

$$C : \mathcal{H} \to l^2 \text{ with } C(f) = (\langle f, f_k \rangle)_k$$

is well defined and bounded from $\mathcal{H}$ in l^2 with $\|D\|_{Op} \leq \sqrt{B}$.

2. *A sequence (g_k) is a frame sequence if and only if it is a Bessel sequence and C has closed range.*

3. *A sequence (g_k) is a frame if and only if it is a frame sequence and C is injective.*

4. *A sequence (g_k) is a frame if and only if it is a Bessel sequence and C is injective.*

We can also do the same compilation with the associated frame operator and extend Theorem 2.5. from [22] to:

Theorem 1.1.13 1. *A sequence (g_k) is a Bessel sequence with bound B if and only if the associated frame operator*

$$S : \mathcal{H} \to \mathcal{H} \text{ with } S(f) = \sum_k \langle f, g_k \rangle \gamma_k$$

is well defined and bounded from $\mathcal{H}$ in l^2 with $\|S\|_{Op} \leq B$.

2. *A sequence (g_k) is a frame sequence if and only if it is a Bessel sequence and S has closed range.*

3. *A sequence* (g_k) *is a frame if and only if it is a frame sequence and* S *is injective.*

4. *A sequence* (g_k) *is a frame if and only if it is a Bessel sequence and* S *is injective.*

Proof: From Lemma 1.1.5 we know that, if S is well-defined, then S is self-adjoint and positive semi-definite.

1.) We know from Proposition A.4.15 that

$$\|S\|_{Op} = \sup_{\|f\|_{\mathcal{H}} \leq 1} |\langle Sf, f \rangle|$$

and because

$$\left\langle S \frac{f}{\|f\|_{\mathcal{H}}}, \frac{f}{\|f\|_{\mathcal{H}}} \right\rangle = \frac{1}{\|f\|_{\mathcal{H}}^2} \langle Sf, f \rangle$$

we know that

$$\|S\|_{Op} \cdot \|f\|_{\mathcal{H}}^2 \geq \langle Sf, f \rangle \text{ for all} f \neq 0.$$

$$\langle Sf, f \rangle = \sum_k |\langle f, g_k \rangle|^2$$

and so we know

$$\sum_k |\langle f, g_k \rangle|^2 \leq \|S\|_{Op} \cdot \|f\|_{\mathcal{H}}^2 \forall f$$

For the other direction let $\{g_k\}$ be a Bessel sequence, then C and D are well-defined and bounded. The frame operator $S = D \circ C$ and therefore it is also bounded.

2.) As S is positive on $span\{g_h\}$, this means that S is injective on $\overline{ran(D)} = ker(C)^{\perp}$, therefore $ker(S) \subseteq ker(C)$. But as $S = D \circ C$, $ker(C) \subseteq ker(S) \implies \overline{ran(S)} = ker(S)^{\perp} = ker(C)^{\perp} = \overline{ran(D)}$, therefore D is closed and $\{g_k\}$ is a frame sequence.

If $\{g_k\}$ is a frame sequence, we know from Proposition 1.1.14, that S is an orthogonal projection and therefore closed.

3.) & 4.) If $S = D \circ C$ is injective, C is injective.

□

The proof also tells us, that for frame sequences $ran(S) = ran(D)$ and $ker(S) = ker(C)$.

1.1.7 Frames And Operators

In this section we will look at the connection of operators and frames respectively other sequences. We will investigate the connection of the frame bounds and certain operators, apply operators on frames, describe operators with frames and describe frames as images of ONBs.

But first let us state the very important result:

Proposition 1.1.14 ([23] 5.3.5) *Let (g_k) be a frame sequence. Then the orthogonal projection P on the space $V = \overline{span}\{g_k\}$ is just the frame operator extended to the whole space $\mathcal{H}$, so*

$$P_V(f) = \sum_k \langle f, \tilde{g}_k \rangle g_k$$

where $(\tilde{g}_k)$ is the dual frame in V.

The space V is closed, see Section 1.1.6, so the projection on this space is an orthogonal projection.

1.1.7.1 Frame Bounds

Apart from the equalities in Proposition 1.1.3 the optimal frame bounds can also be given by the operator norm of the analysis or synthesis operators:

Corollary 1.1.15 ([23] Proposition 5.4.4)*Let $\mathcal{G} = \{g_k\}$ be a frame for $\mathcal{H}$. Then the optimal frame bounds A_{opt}, B_{opt} are*

$$B_{opt} = \left\| C_{\{g_k\}} \right\|_{Op}^2 = \left\| S_{\{g_k\}} \right\|_{Op}$$

$$A_{opt} = \left\| C_{\{S^{-1}g_k\}} \right\|_{Op}^2 = \left\| S_{\{g_k\}}^{-1} \right\|_{Op}$$

Some other statements regarding the properties of the bounds:

Proposition 1.1.16 *Let $\{g_k\}$ be a frame with the lower frame bound A and the upper frame bound B. Then $A \leq \sum \|g_l\|^2$ and $B \geq \|g_l\|^2$. If $\|g_l\|^2 < A$, then $g_n \in \overline{span}_{l \neq n}(g_l)$, so $\{g_k\}$ is not minimal. $B = \|g_l\|^2$, if $b_l \perp span\{g_k\}_{k \neq l}$.*

That $A \cdot dim\mathcal{H} \leq \sum\limits_k \|g_k\|^2$ can be found in Corollary 1.2.16. The rest can be found in [16] 4.6.

1.1.7.2 Operators Applied On Frames

A natural question arises, when we ask if frames keep there frame property if an operator is applied to its elements. If the operator is surjective, this is true. Note that $U^\dagger$ signifies the pseudo-inverse of the operator U, cf. appendix A.4.6.

Proposition 1.1.17 ([23] 5.3.2) *Let (g_k) be a frame with bounds A, B and $U : \mathcal{H} \to \mathcal{H}$ a surjective bounded operator. Then (Ug_k) is a frame with the frame bounds $A \cdot \left\| U^\dagger \right\|_{\mathcal{H}}^{-2}$ and $B \cdot \|U\|_{\mathcal{H}}^2$.*

Again this can be easily adapted to Bessel sequences and *any* operators

Proposition 1.1.18 *Let (g_k) be a Bessel sequence with bound B and $U : \mathcal{H} \to \mathcal{H}$ a bounded operator. Then (Ug_k) is a Bessel sequence with the Bessel bound $B \cdot \|U\|_{\mathcal{H}}^2$.*

Proof:

$$\sum_k |\langle f, Ug_k \rangle|^2 \leq \sum_k |\langle U^* f, g_k \rangle|^2 \leq B \cdot \|U^* f\|_{\mathcal{H}} \leq B \cdot \|U\|_{Op}^2 \|f\|_{\mathcal{H}}^2$$

□

We can state a similar result for frames:

Proposition 1.1.19 ([23] 5.3.1) *Let (g_k) be a frame with bounds A and B and $U : \mathcal{H} \to \mathcal{H}$ a bounded operator with closed range. Then (Ug_k) is a frame sequence with the bounds $A \cdot \left\| U^\dagger \right\|_{\mathcal{H}}^{-2}$ and $B \cdot \|U\|_{\mathcal{H}}^2$.*

If $\{g_k\}$ is only a frame sequence the proposition stays not true, which might be surprising. To get this result also for frame sequences, a sufficient condition would be, that U is a closed function, i.e. it maps closed sets onto closed sets. (Note: This is not equivalent to being an operator with closed graph, which is sometimes also called a closed operator). This becomes clear with the following result:

Corollary 1.1.20 *Let (g_k) be a Bessel sequence and $U : \mathcal{H} \to \mathcal{H}$ an operator. Then*

$$S_{Ug_k} = U \circ S_{g_k} \circ U^*$$

Proof: We know that (Ug_k) is a frame. So

$$S_{Ug_k} f = \sum \langle f, Ug_k \rangle Ug_k = U \left(\sum \langle U^* f, g_k \rangle g_k \right)$$

□

For tight frames (f_k) with the frame bound A the last corollary gives

$$S_{Uf_k} = A \cdot UU^*$$

and therefore only unitary operators map tight frames on tight frames with the same bound .

Lemma 1.1.21 *Let (g_k) be a frame and let (γ_k) be the canonical dual frame, then let U_g and U_γ be the operators with $U_g(e_k) = g_k$ and $U_\gamma(e_k) = \gamma_k$, then*

$$U_\gamma = U_g^\dagger$$

Proof: $U_g = D_g \circ C_e$ is surjective, as (g_k) is frame. So

$$U_g^\dagger = (D_g \circ C_e)^* \circ \left[D_g \circ C_e (D_g \circ C_e)^*\right]^{-1} =$$

$$= D_e \circ C_g \left[D_g \circ C_e \circ D_e \circ C_g\right]^{-1} = D_e \circ C_g \left[D_g \circ C_g\right]^{-1} = D_e \circ C_g \circ S_g^{-1} =$$

$$= D_e \circ C_g \circ S_\gamma = D_e \circ C_g \circ D_\gamma \circ C_\gamma = D_e \circ C_\gamma = U_\gamma$$

□

1.1.7.3 Matrix Representation With Frames

An operator U can be described by the image of the elements of the frame. For a linear operator $U(f) = U(\sum_k \langle f, \tilde{g}_k \rangle g_k) = \sum_k \langle f, \tilde{g}_k \rangle U g_k$. The right hand side is well-defined, because the Ug_k form a Bessel sequence. It is clearly linear, and it is bounded, again because the Ug_k form a Bessel sequence. The opposite direction, which is often used with ONBs, to define an operator by the images of the frame $U(g_k) := h_k$ is in general not well-defined. It is well-defined if for $\sum\limits_k c_k g_k = \sum\limits_k d_k g_k \Longrightarrow \sum\limits_k c_k h_k = \sum\limits_k d_k h_k$, so if $ker\left(D_{g_k}\right) \subseteq ker\left(D_{h_k}\right)$. If D_{g_k} is injective, then this is certainly true. We will look at sequences with that property in 1.1.8.

For ONBs it is well known, that operators can be uniquely described by the image of this basis, but the same is true for frames. Any operator can be defined by the images of the elements of a frame. (But contrary to a basis this definition is not unique any more.) Recall A.4.3.4 the definition of the operator defined by a (possibly infinite) matrix : $(Mc)_j = \sum\limits_k M_{j,k} c_k$.

We will start with the more general case of Bessel sequences. Note that we will use the notation $\|.\|_{Hil_2 \to \mathcal{H}_2}$ for the operator norm to be able to distinguish between different operator norms.

Theorem 1.1.22 *Let (g_k) be a Bessel sequence in $\mathcal{H}_1$ with bound B, (f_k) in $\mathcal{H}_2$ with B'.*

1. *Let $O : \mathcal{H}_1 \to \mathcal{H}_2$ be a bounded, linear operator. Then the infinite matrix*
$$\mathcal{M}^{(f_k,g_j)}_{k,j} = \langle Og_j, f_k \rangle$$
defines a bounded operator from l^2 to l^2 with $\|\mathcal{M}\|_{l^2\to l^2} \leq \sqrt{B \cdot B'} \cdot \|O\|_{\mathcal{H}\to\mathcal{H}}$.

2. *On the other hand let M be a infinite matrix for which the operator $(Mc)_i = \sum\limits_k M_{i,k} c_k$ defines a bounded operator from l^2 to l^2, then the operator defined by*
$$\left(\mathcal{O}^{(f_k,g_j)}(M)\right) f = \sum_k \left(\sum_j M_{k,j} \langle f, g_j \rangle \right) f_k$$
is a bounded operator from $\mathcal{H}_1$ to $\mathcal{H}_2$ with $\|O_M\|_{\mathcal{H}_1\to\mathcal{H}_2} \leq \sqrt{B \cdot B'} \, \|M\|_{l^2\to l^2}$.
$$O^{(f_k,g_j)} = D_{f_k} \cdot M \circ C_{g_k} = \sum_k \sum_j M_{k,j} \cdot f_k \otimes \overline{g}_j$$

Proof: Let $\mathcal{M} = \mathcal{M}^{(f_k,g_j)}$ and $\mathcal{O} = \mathcal{O}^{(f_k,g_j)}$

1.)
$$\left(\mathcal{M}(O)\, c\right)_j = \sum_k \left(\mathcal{M}(O)\right)_{j,k} c_k = \sum_k \langle Og_k, f_j \rangle c_k =$$
$$= \left\langle \sum_k c_k O g_k, f_j \right\rangle = \left\langle O \sum_k c_k g_k, f_j \right\rangle = \langle O D_{g_k} c, f_j \rangle$$
$$\Longrightarrow \|\mathcal{M}c\|_2^2 = \sum_j |\langle O D_{g_k} c, f_j \rangle|^2 \leq$$
$$\leq B' \cdot \|O D_{g_k} c\|_{\mathcal{H}}^2 \leq B' \cdot \|O\|_{Op}^2 B \|c\|_2^2$$

2.)
$$\mathcal{O}(M) = D_{f_k} \circ M \circ C_{g_k}$$
$$\Longrightarrow \|\mathcal{O}(M)\|_{\mathcal{H}_1\to\mathcal{H}_2} \leq \|D_{f_k}\|_{l^2\to\mathcal{H}_2} \cdot \|M\|_{l^2\to l^2} \cdot \|C_{g_k}\|_{\mathcal{H}_1\to l^2} \leq$$
$$\leq \sqrt{B'} \cdot \|M\|_{l^2\to l^2} \sqrt{B}$$

□

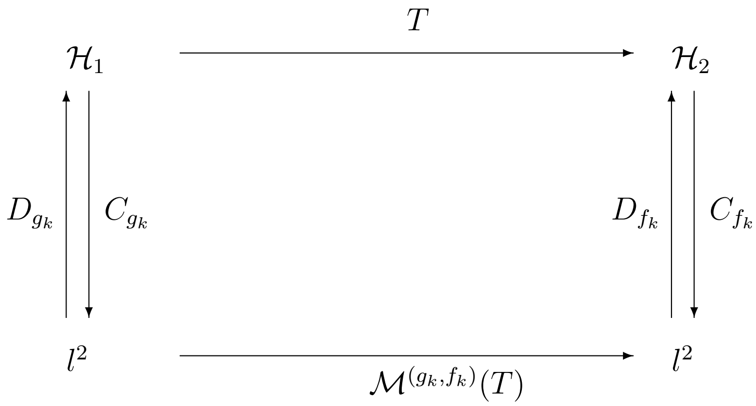

Figure 1.2: Matrix and operator induced by each other

Definition 1.1.6 *For an operator O and a matrix M like in theorem 1.1.22, we call $\mathcal{M}^{(g_k,f_k)}(O)$ the* **matrix induced by the operator** *O with respect to the frames (g_k) and (f_k) and $\mathcal{O}^{(g_k,f_k)}(M)$ the* **operator induced by the matrix** *M with respect to the frames (g_k) and (f_k).*

If we do not want to stress the dependency on the frames and there is no change of confusion, the notation $\mathcal{M}(O)$ and $\mathcal{O}(M)$ will be used.

For frames we get

Proposition 1.1.23 *Let (g_k) be a frame in $\mathcal{H}_1$ with bounds A, B, (f_k) in $\mathcal{H}_2$ with A', B'. Then*

1. $$\left(\mathcal{O}^{(f_k,g_j)} \circ M^{(\tilde{f}_k,\tilde{g}_j)}\right)(O) = Id = \left(\mathcal{O}^{(\tilde{f}_k,\tilde{g}_j)} \circ M^{(f_k,g_j)}\right)(O)$$
   And so
   $$O = \sum_{k,i} \left\langle O\tilde{g}_j, \tilde{f}_k \right\rangle f_k \otimes \overline{g}_j$$

2. *$\mathcal{M}^{(f_k,g_j)}$ is injective and $\mathcal{O}^{(f_k,g_j)}$ is surjective.*

3. *Let $\mathcal{H}_1 = \mathcal{H}_2$ and $(g_k) = (f_k)$, then $\mathcal{O}^{(g_k,\tilde{g}_j)}(Id_{l^2}) = Id_{\mathcal{H}_1}$*

4. *Let (h_k) be any frame in $\mathcal{H}_3$, and $O : \mathcal{H}_3 \to \mathcal{H}_2$ and $P : \mathcal{H}_1 \to \mathcal{H}_3$. Then*
   $$\mathcal{M}^{(f_q,g_p)}(O \circ P) = \left(\mathcal{M}^{(f_p,h_k)}(O) \cdot \mathcal{M}^{(\tilde{h}_k,g_q)}(P)\right)$$

Proof: 1.)

$$\left(\mathcal{O} \circ \mathcal{M}\right)(O)(f) = \sum_k \left(\sum_j \left\langle O\tilde{g}_j, \tilde{f}_k \right\rangle \langle f, g_j \rangle \right) f_k =$$

$$= \sum_j \left(\sum_k \left\langle O\tilde{g}_j, \tilde{f}_k \right\rangle f_k \right) \langle f, g_j \rangle = \sum_j O\tilde{g}_j \langle f, g_j \rangle = Of$$

For the other equality the roles of the frame and the dual are just switched.

2.) From $\mathcal{O}\mathcal{M} = Id$ we know that $\mathcal{M}$ is injective and $\mathcal{O}$ is surjective.

3.)

$$\mathcal{O}(Id)f = \sum_k \left(\sum_j \delta_{k,j} \langle f, \tilde{g}_j \rangle \right) g_k = \sum_k \langle f, \tilde{g}_k \rangle g_k = f$$

4.)

$$\mathcal{M}^{(f_q, g_p)} (O \circ P)_{p,q} = \langle O \circ P g_q, f_p \rangle = \left\langle P g_q, O^* \tilde{f}_p \right\rangle$$

On the other hand

$$\left(\mathcal{M}^{(f_p, h_k)} (O) \cdot \mathcal{M}^{(\tilde{h}_k, g_q)} (P) \right)_{p,q} = \sum_k \mathcal{M}^{(f_p, h_k)} (O)_{p,k} \cdot \mathcal{M}^{(\tilde{h}_k, g_q)} (P)_{k,q} =$$

$$= \sum_k \langle O h_k, f_p \rangle \left\langle P g_q, \tilde{h}_k \right\rangle = \sum_k \left\langle h_k, O^* \tilde{f}_p \right\rangle \left\langle P g_q, \tilde{h}_k \right\rangle =$$

$$= \left\langle \sum_k \left\langle P g_q, \tilde{h}_k \right\rangle h_k, O^* \tilde{f}_p \right\rangle = \left\langle P g_p, O^* \tilde{f}_p \right\rangle$$

□

As a direct consequence we get the following corollary.

Corollary 1.1.24 *$\mathcal{M}^{(f_k, \tilde{f}_k)}$ is a Banach-algebra monomorphism between the algebra of bounded operators from $\mathcal{H}$ to $\mathcal{H}$ with $\circ$ and the (infinite) matrices with the normal matrix-multiplication.*

The other function $\mathcal{O}$ is in general not so "well-behaved". Again if the dual frames are biorthogonal this is true, refer to the Section 1.1.8.

For the description of the Gram Matrix (cf. Section 1.1.9) and its behavior (cf. [58]) it would be very interesting to look more closely at the class of infinite matrix defining bounded operators. This is important to get sufficient conditions for Bessel sequences. One well-known condition is *Schur's lemma*, refer to Lemma A.4.19. We will state another result in Section 1.2.3.1, where we will look especially at Hilbert-Schmidt operators.

Let us give another look to the matrix induced by an operator $O : \mathcal{H}_1 \rightarrow \mathcal{H}_2$,

Lemma 1.1.25 *Let $O : \mathcal{H}_1 \rightarrow \mathcal{H}_2$ be a linear and bounded operator, $(g_k) \subseteq \mathcal{H}_1$ and $(f_k) \subseteq \mathcal{H}_2$ frames. Then $\mathcal{M}^{(f_k,\tilde{g}_j)}(O)$ maps $ran\left(C_{g_k}\right)$ into $ran\left(C_{f_k}\right)$ with*

$$\langle Of, g_j \rangle \mapsto \langle f, f_k \rangle .$$

If O is surjective respectively injective, then $\mathcal{M}^{(f_k,\tilde{g}_j)}(O)$ is, too.

Proof: Let $c \in ran(C_{g_k})$, then there exists $f \in \mathcal{H}_1$ such that $c_k = \langle f, g_k \rangle$.

$$\left(\mathcal{M}^{(f_k,\tilde{g}_j)}(O)(c)\right)_i = \sum_k \langle O\tilde{g}_k, f_i \rangle \langle f, g_k \rangle = \left\langle \sum_k \langle f, \tilde{g}_k \rangle Og_k, f_i \right\rangle = \langle Of, f_i \rangle$$

So $\left(\langle f, g_k \rangle\right)_k \mapsto \left(\langle Of, f_i \rangle\right)_i$.

If O is surjective, then for every f there exists a g such that $Og = f$, and therefore $\langle g, g_k \rangle \mapsto \langle f, f_i \rangle$.

If O is injective, then let's suppose that $\langle Of, f_i \rangle = \langle Og, f_i \rangle$. Because (f_i) is a frame $\Longrightarrow Of = Og \Longrightarrow f = g \Longrightarrow \langle f, g_k \rangle = \langle g, g_k \rangle$. □

Particularly for $O = Id$ the matrix $G_{f_i,\tilde{g}_k} = \left(\langle \tilde{g}_k, f_i \rangle\right)_{k,i}$ maps $ran\left(C_{g_k}\right)$ bijectively on $ran\left(C_{f_k}\right)$. So we get a way to a way to "switch" between frames. For more on this kind of matrix we refer to Section 1.1.9.

Let us finish with some interesting examples:

Example 1.1.2 :

Let (g_k) and (f_k) be frames in $\mathcal{H}$ and δ_j the canonical basis of l^2. Then

1. For $S : \mathcal{H} \rightarrow \mathcal{H}$ we have $\mathcal{M}(S) = G_{g_j}$.
2. For $S^{-1} : \mathcal{H} \rightarrow \mathcal{H}$ we have $\mathcal{M}(S^{-1}) = G_{\tilde{g}_j}$.
3. For $C_{f_l} : \mathcal{H} \rightarrow l^2$ we have
$$\mathcal{M}(O)_{i,k} = \langle C_{f_l} g_k, \delta_i \rangle = \sum_l \langle g_k, f_l \rangle \delta_{i,l} = \langle g_k, f_i \rangle = \left(G_{f_i,g_k}\right)_{i,k}$$
4. For $Id : \mathcal{H} \rightarrow \mathcal{H}$ we have $\mathcal{M}(Id) = G_{\tilde{f}_i,g_k}$.
5. For $Id : l^2 \rightarrow l^2$ we have $\mathcal{O}(Id) = S_{g_k,\tilde{f}_i} = D_{f_k} \circ C_{\tilde{g}_k}$.

1.1.7.4 Classification With ONBs

Frames can be described as images of an orthonormal basis by bounded linear operators in an infinite dimensional Hilbert space. They can even be classified by this result:

Proposition 1.1.26 ([23] 5.5.5) *Let $\{e_k\}_{k=0}^{\infty}$ be an arbitrary infinite ONB for $\mathcal{H}$. The frames for $\mathcal{H}$ are precisely the families $\{Ue_k\}$, where $U : \mathcal{H} \to \mathcal{H}$ is a bounded and surjective operator.*

This operator is just the composition of an analysis and a synthesis operator. $U = D_{f_k} C_{e_k}$. From the application and finite dimensional space viewpoint this proposition seems to be very strange and we will revisit this statement in these circumstances in Proposition 1.2.5.

With the knowledge we have gained from Section 1.1.7.2 we can again restate this result for Bessel and frame sequences:.

Corollary 1.1.27 *Let $\{e_k\}_{k=0}^{\infty}$ be an arbitrary infinite ONB for $\mathcal{H}$.*

1. *The Bessel sequences for $\mathcal{H}$ are precisely the families $\{Ue_k\}$, where $U : \mathcal{H} \to \mathcal{H}$ is a bounded operator.*

2. *The frame sequences for $\mathcal{H}$ are precisely the families $\{Ue_k\}$, where $U : \mathcal{H} \to \mathcal{H}$ is a bounded operator with closed range.*

Proof: 1.) From Proposition 1.1.18 we know that $U(e_k)$ is a Bessel sequence, if U is a bounded operator. For the opposite direction let (f_k) be the Bessel sequence. Use $U = D_{f_k} \circ C_{e_k}$. This operator fulfills the condition.

2.) From Proposition 1.1.19 we know that $U(e_k)$ is a frame sequence, if U is a bounded operator with closed range. For the opposite direction let (f_k) be the frame sequence. Use $U = D_{f_k} \circ C_{e_k}$. C_{e_k} is a bijection and D_{f_k} has close range 1.1.11, so U has closed range. □

With Riesz bases, a class of sequences defined in the next section, this classification can be extended, see 1.1.30.

From 1.1.26 we know now that every frame (f_k) can be described as the image of an surjective operator U of an arbitrary ONB (e_n), $f_k = Ue_k$. So as ONBs are tight frames with $A = 1$, we now know with Corollary 1.1.20:

Corollary 1.1.28 *Let $\{e_k\}_{k=0}^{\infty}$ be an arbitrary infinite ONB for $\mathcal{H}$. Let $\{f_k\}$ be a Bessel sequence and let U be the bounded operator with $U(e_k) = f_k$, then*

$$S_{f_k} = UU^*$$

1.1.8 Riesz Bases

Recall the following definitions from Section A.4.2.2:

Definition 1.1.7 *1. A sequence $\{e_k\}$ is called a* **basis** *for $\mathcal{H}$, if for all $f \in \mathcal{H}$ there are unique c_k such that*

$$f = \sum_k c_k f_k$$

2. Two sequences (g_k), (f_k) are called **biorthogonal** *if*

$$\langle g_k, h_j \rangle = \delta_{kj}$$

For any basis there is a unique biorthogonal sequence, which also is a basis [23], and so e.g. the following is true:

Lemma 1.1.29 *Let (g_k) be a basis of $\mathcal{H}$ and let $(\tilde{g}_k)$ be its unique biorthogonal sequence, then for all $f, g \in \mathcal{H}$*

$$\langle f, g \rangle = \sum_k \langle f, g_k \rangle \langle \tilde{g}_k, g \rangle$$

Proof: Use for f and g the expansion to the bases (g_k) and $(\tilde{g}_k)$ respectively. □

Compare to the equivalent property for frames in Proposition 1.1.10.

Proposition 1.1.30 ([23] 3.6.6.) *Let (g_k) be a sequence in $\mathcal{H}$. Then the following conditions are equivalent:*

1. *(g_k) is an image of an ONB (e_k) under an invertible bounded operator $T \in \mathcal{B}(\mathcal{H})$.*

2. *(g_k) is complete in $\mathcal{H}$ and there exist constants A, $B > 0$ such that the inequalities*

$$A \|c\|_2^2 \le \left\| \sum_{k \in K} c_k g_k \right\|_{\mathcal{H}}^2 \le B \|c\|_2^2$$

hold for all finite sequences $c = \{c_k\}$.

3. *(g_k) is complete in $\mathcal{H}$ and the* **Gram Matrix** *G, given by $G_{jm} = \langle g_m, g_j \rangle$ $j, m \in K$ (cf. Section 1.1.9) defines a bounded invertible operator on $l^2(K)$. (It is, even more, a positive operator.)*

4. (g_k) *is a complete Bessel sequence in* $\mathcal{H}$ *and it has a complete biorthogonal sequence* (f_k)*, which is also a Bessel sequence.*

Definition 1.1.8 *If a sequence fulfills the conditions in 1.1.30 it is called a* **Riesz bases***. A sequence* (g_k) *that is a Riesz basis only for* $\overline{span}(g_k)$ *is called a* **Riesz sequence***.*

Proposition 1.1.30 Point 2.) leads directly to

Corollary 1.1.31 *Every subfamily of a Riesz basis is a Riesz sequence.*

Clearly Riesz bases are bases, and from property 1 above and 1.1.26 it is evident that Riesz basis are frames. In this case the Riesz bounds coincide with the frame bounds. But when are frames Riesz Bases? We can state the following equivalent conditions found in [23] (6.1.1) and [63] (5.1.5).

Theorem 1.1.32 *Let* $\{g_k\}$ *be a frame for* $\mathcal{H}$*. Then the following conditions are equivalent:*

1. (g_k) *is a Riesz basis for* $\mathcal{H}$*.*
2. *The coefficients* $(c_k) \in l^2$ *for the series expansion with* (g_k) *are unique. So the synthesis operator* D *is injective.*
3. *The analysis operator* C *is surjective.*
4. (g_k) *is an exact frame.*
5. (g_k) *is* **minimal** *(meaning* $g_j \notin \overline{span}(g_k)_{k \neq j}$ *for all* j*) (cf. Definition A.4.7).*
6. (g_k) *has a biorthogonal sequence.*
7. (g_k) *and* $(S^{-1}g_k)$ *are biorthogonal.*
8. (g_k) *is a basis.*

This clearly means, that if the frame is a Riesz Basis, then the analysis and synthesis operators are bijections. As the coefficients are unique, clearly $0 \notin \{g_k\}$. As any Riesz basis is minimal, we see from 1.1.16 that for Riesz bases $A \leq \|g_l\|^2$. That's another reason why 0 cannot be an element of a Riesz basis.

There is an equivalence condition, when a sequence is a Riesz basis. For that we need the following definition

Definition 1.1.9 *A sequence* (f_k) *is called* **semi-normalized** *if it fulfills*

$$0 < \inf_k \|f_k\|_{\mathcal{H}} \leq \sup_k \|f_k\|_{\mathcal{H}} < \infty$$

Lemma 1.1.33 ([23] 3.6.2) *A sequence* (f_k) *is a Riesz basis for* $\mathcal{H}$ *if and only if it is a semi-normalized, unconditional basis.*

Bounds for the norm of the elements of Riesz bases are exactly the frame bounds.

Corollary 1.1.34 *Let* (g_k) *be a Riesz basis with bounds* A *and* B*. Then for all* k

$$\sqrt{A} \leq \|g_k\|_{\mathcal{H}} \leq \sqrt{B}$$

Proof: The upper bound follows from Lemma 1.1.1.

For the lower bound we know for the dual frame that

$$\sum_k \left| \left\langle f, \tilde{f}_k \right\rangle \right|^2 \leq \frac{1}{A} \cdot \|f\|_{\mathcal{H}}^2$$

Therefore for a fixed i

$$1 = \sum_k \left| \left\langle f_i, \tilde{f}_k \right\rangle \right|^2 \leq \frac{1}{A} \cdot \|f_i\|_{\mathcal{H}}^2$$

□

The coefficients using a Riesz Basis are unique, so 1.1.22 can be extended to:

Theorem 1.1.35 *Let* (f_k) *be a Riesz basis for* $\mathcal{H}_1$*,* (g_k) *for* $\mathcal{H}_2$*. The functions* $\mathcal{M}^{(f_k,g_k)}$ *and* $\mathcal{O}^{(\tilde{f}_k,\tilde{g}_k)}$ *between the set of bounded linear operator from* $\mathcal{H}_1$ *to* $\mathcal{H}_2$*,* $\mathcal{B}(\mathcal{H}1,\mathcal{H}_2)$ *and the infinite matrices, which induce bounded operators from* l^2 *to* l^2*, defined in 1.1.22, are bijective.* $\mathcal{M}^{(f_k,g_k)}$ *and* $\mathcal{O}^{(f_k,g_k)}$ *are inverse functions. The identity is mapped on the identity by* $\mathcal{M}^{(f_k,g_k)}$ *and* $\mathcal{O}^{(f_k,g_k)}$*.*

$\mathcal{M}^{(f_k,\tilde{f}_k)}$ *and* $\mathcal{O}^{(f_k,\tilde{f}_k)}$ *are Banach algebra isomorphisms between the algebra of bounded operators* $\mathcal{H}_1 \to \mathcal{H}_1$ *and the infinite matrices, which induce bounded operators from* l^2 *to* l^2*.*

Proof: We know that $\mathcal{O} \circ M = Id$. Let's look at

$$(\mathcal{M} \circ O)(M)_{p,q} = \mathcal{M}\left(\sum_k \sum_j M_{k,j} \left\langle \cdot, \tilde{g}_j \right\rangle f_k \right)_{p,q} =$$

$$= \left\langle \sum_k \sum_j M_{k,j} \langle g_q, \tilde{g}_j \rangle f_k, \tilde{f}_p \right\rangle = \sum_k \sum_j M_{k,j} \underbrace{\langle g_q, \tilde{g}_j \rangle}_{\delta_{k,p}} \underbrace{\left\langle f_k, \tilde{f}_p \right\rangle}_{\delta_{k,p}} = M_{p,q}$$

So these functions are inverse to each other and therefore bijective.

$$\mathcal{M}(Id_{\mathcal{H}\to\mathcal{H}})_{p,q} = \langle Idg_q, \tilde{g}_p \rangle = \langle g_q, \tilde{g}_p \rangle = \delta_{q,p} = Id_{l^2\to l^2}$$

We know that $\mathcal{M}^{(f_k,\tilde{f}_k)}$ is a Banach algebra homomorphism and so its inverse is, too.

□

If the frame is tight, we clearly see from Theorem 1.1.22 that these functions are isometric. To get isometric isomorphism we have to use tight Riesz bases. We will see in 1.1.11.1 that such frames are just rescaled orthonormal bases (with fixed scale).

The proposition 1.1.26 can be very easily be extended to Riesz Bases.

Proposition 1.1.36 *Let (e_n) with $n \in I$ be an arbitrary Riesz basis for the infinite dimensional $\mathcal{H}$. The frames for $\mathcal{H}$ are precisely the families $\{Ue_k\}$, where $U : \mathcal{H} \to \mathcal{H}$ is a bounded and surjective operator.*

Proof:

Let (e_n), $n \in I$ be a (countable) Riesz basis for $\mathcal{H}$, and let (δ_n) be the canonical basis for $l^2(I)$. If we look at $C_{(e_n)}$ the analysis operator of the frame (e_n), then clearly the operator defined by $U := D_{(f_k)} \circ C_{(e_n)}$ is a surjective bounded operator. And from 1.1.32 we know that $C(e_n) = \delta_n$, so $U(e_n) = f_n$.

On the other side $U(e_n)$ is clearly a frame, see 1.1.17. □

If we want to have a classification of frames by operators, it's more useful to do that with a smaller class. So it is preferable to stick to orthonormal bases here.

1.1.9 Gram Matrix

We have mentioned the Gram matrix before. Let us repeat

Definition 1.1.10 *Let $\{g_k\}$ and $\{g'_k\}$ be two sequences in $\mathcal{H}$. The* **cross-Gram matrix** *G_{g_k,g'_k} for these sequences is given by $\left(G_{g_k,g'_k}\right)_{jm} = \langle g'_m, g_j \rangle$, $j, m \in K$.*

If $(g_k) = (g'_k)$ we call this matrix the **Gram matrix** *G_{g_k}.*

We can look at the operator induced by the Gram matrix, defined for $c \in l^2$ formally as

$$(G_{g_k,g'_k} c)_j = \sum_k c_k \langle g'_k, g_j \rangle$$

Clearly for two Bessel sequences it is well defined, as

$$\left(\left(C_{g_k} \circ D_{g'_k} \right) c \right)_j = \left\langle \sum_k c_k g'_k, g_j \right\rangle = \sum_k c_k \langle g'_k, g_j \rangle = (G_{g_k,g'_k} c)_j$$

and therefore

$$\left\| G_{g_k,g'_k} \right\|_{Op} \leq \left\| C_{g_k} \right\|_{Op} \left\| D_{g'_k} \right\|_{Op} \leq B$$

1.1.9.1 Classification With The Gram Matrix

Let us state the connection between the kind of sequence and the Gram matrix:

Theorem 1.1.37 *Let (g_k) be a sequence in $\mathcal{H}$ and let G be its Gram matrix. Then*

1. *The Gram Matrix defines G a bounded function from l^2 into l^2 if and only if the sequence (g_k) is a Bessel sequence. In this case the Gram matrix defines an injective operator from R_C to R_C. The range of G is dense in R_C. The operator norm of G is the optimal Bessel bound.*

2. *The Gram Matrix defines a bounded operator from R_C onto R_C with bounded inverse if and only if the sequence (g_k) is a frame sequence.*

3. *The Gram Matrix G defines an bounded, invertible operator on l^2 if and only if the sequence (g_k) is a Riesz sequence.*

Proof: 1.) See [23] 3.5.1. and 3.5.2.
2.) For one direction, (g_k) is a frame sequence, see [23] 5.2.2.

For the other direction, suppose that G is bounded invertible on R_C. With A.4.6 it is enough to show that $C^{-1} : ran(C) \to \mathcal{H}$ is bounded. $C^{-1} = C^\dagger \left|_{ran(C)} \right.$. So

$$C^{-1} = C^\dagger \left|_{ran(C)} \right. \overset{A.4.45}{=} C^* \left(C C^* \right)^{-1} =$$
$$= D \left(C D \right)^{-1} = D \left(C D \right)^{-1} = D G^{-1}$$

Therefore

$$\left\| C^{-1} \right\|_{ran(C) \to \mathcal{H}} = \left\| D G^{-1} \right\|_{ran(C) \to \mathcal{H}} \leq \left\| D \right\|_{l^2 \to \mathcal{H}} \left\| G^{-1} \right\|_{ran(C) \to ran(C)}$$

3.) See Proposition 1.1.30. □

In [58] decay properties of the Gram matrix are investigated. As can be seen from the above theorem this can be useful for sufficient conditions for sequences being a Bessel sequence.

Connected to the last property, we can prove a result stated in [58] and extend it to Bessel and frame sequences

Lemma 1.1.38 *1. Let (g_k) be a Bessel sequence, then $ker(G) = ker(D) = ran(C)^{\perp}$.*

2. Let (g_k) be a frame sequence, then $ran(C) = ran(G) = ker(D)^{\perp}$

Proof: 1.) From 1.1.37 we know $ker(D) \subseteq ker(G)$ as $G = C \circ D$. G is injective on $ran(C)$, this means that $ran(C) \subseteq ker(G)^{\perp}$. So

$$ran(C)^{\perp} \supseteq ker(G) \Longrightarrow ker(D) \supseteq ker(G) \Longrightarrow ker(D) = ker(G)$$

2.) Follows directly from 1.1.37 as G is surjective on $ran(C)$ and $ran(G) \subseteq ran(C)$.

□

1.1.9.2 Properties Of The Cross-Gram-Matrix

Let us look at a figure, Figure 1.3, where we see the connection of the different operators and the cross-Gram matrix in a commutative function diagram. There a lot of properties can be seen, like stated in the following lemma.

Lemma 1.1.39 *Let (g_k) and (g'_k) be Bessel sequences. Then*

$$S_{g'_k} \circ D_{g_k} = D_{g'_k} \circ G_{g'_k g_k}$$

$$S_{g_k} \circ S_{g'_k} = D_{g_k} \circ G_{g_k, g'_k} \circ C_{g'_k}$$

We can link the bound of the Gram matrix with the frame bounds:

Proposition 1.1.40 *Let (e_k) be an ONB for $\mathcal{H}$, $(f_k) = (Ue_k)$ be a Bessel sequence, and let $G : l^2 \to l^2$ be the Gram matrix for (f_k). The optimal Bessel bound for this sequence is*

$$B_{opt} = \|U\|^2 = \|G\|$$

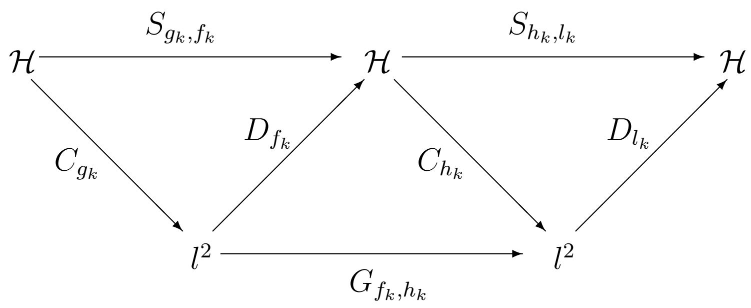

Figure 1.3: Gram matrix for Bessel sequences

Furthermore if (f_k) is a frame, then the optimal lower frame bound is

$$A_{opt} = \left\| U^\dagger \right\|^{-2} = \frac{1}{\| G^\dagger \|}$$

Proof: Look at Figure 1.3, set g_k and l_k to e_k, as well as f_k and h_k in the picture to the f_k here. Then we know that

$$D_{e_k} \circ G_{f_k} \circ C_{e_k} = S_{f_k,e_k} \circ S_{e_k,f_k} = U^* U$$

But as (e_k) is an ONB and so D_{e_k} and C_{e_k} are isometries (and D_{e_k} is surjective) (see Lemma A.3.6 and Proposition A.4.14)

$$\| G_{f_k} \| = \| D_{e_k} \circ G_{f_k} \circ C_{e_k} \| = \| U^* U \| = \| U \|^2$$

Finally we know from Corollary 1.1.28 that $UU^* = S_{f_k}$ and from Theorem 1.1.13 that $B_{opt} = \| S_{f_k} \|$ and so

$$B_{opt} = \| UU^* \| = \| U^* U \| = \| U \|^2 = \| G \|$$

For the optimal lower frame bound use the dual frame with Theorem 1.1.7 and Lemma 1.1.47. □

Compare to [23] Prop. 3.6.8., where this is stated for Riesz bases and an estimation for the lower Riesz bound.

Let us finish this section with a connection between the inner products of $\mathcal{H}$ and l^2

Lemma 1.1.41

$$\langle f, g \rangle_{\mathcal{H}} = \langle G_{\tilde{g}_k} C_{g_k}(f), C_{g_k}(g) \rangle_{l^2}$$

Proof: From figure 1.3 it is clear that $G_{\tilde{g}} C_g = C_{\tilde{g}}$ and so

$$\langle f, g \rangle_{\mathcal{H}} \overset{1.1.10}{=} \sum_k \langle f, \tilde{g}_k \rangle_{\mathcal{H}} \overline{\langle g, g_k \rangle}_{\mathcal{H}} = \langle C_{\tilde{g}_k}(f), C_{g_k}(g) \rangle_{l^2} = \langle G_{\tilde{g}_k} C_{g_k}(f), C_{g_k}(g) \rangle_{l^2}$$

□

Lemma 1.1.42 *Let $O : \mathcal{H}_1 \to \mathcal{H}_2$ be a linear and bounded operator, $(g_k) \subseteq \mathcal{H}_1$ and $(f_k) \subseteq \mathcal{H}_2$ frames. Then $\mathcal{M}^{(f_k, \tilde{g}_j)}(O)$ maps $ran\,(C_{g_k})$ into $ran\,(C_{f_k})$ with*

$$\langle f, g_j \rangle \mapsto \langle Of, f_k \rangle .$$

If O is surjective respectively injective, then $\mathcal{M}^{(f_k, \tilde{g}_j)}(O)$ is, too.

Proof: Let $c \in ran(C_{g_k})$, there exists $f \in \mathcal{H}_1$ such that $c_k = \langle f, g_k \rangle$.

$$\left(\mathcal{M}^{(f_k, \tilde{g}_j)}(O)(c) \right)_i = \sum_k \langle O\tilde{g}_k, f_i \rangle \langle f, g_k \rangle = \left\langle \sum_k \langle f, \tilde{g}_k \rangle O g_k, f_i \right\rangle = \langle Of, f_i \rangle$$

So $(\langle f, g_k \rangle)_k \mapsto (\langle Of, f_i \rangle)_i$.

If O is surjective, then for every f there exists a g such that $Og = f$, and therefore $\langle g, g_k \rangle \mapsto \langle f, f_i \rangle$.

If O is injective, suppose that $\langle Of, f_i \rangle = \langle Og, f_i \rangle$. Because (f_i) is a frame $\Longrightarrow Of = Og \Longrightarrow f = g \Longrightarrow \langle f, g_k \rangle = \langle g, g_k \rangle$. □

Particularly for $O = Id$ the matrix $G_{f_i, \tilde{g}_k} = (\langle \tilde{g}_k, f_i \rangle)_{k,i}$ maps $ran\,(C_{g_k})$ bijectively on $ran\,(C_{f_k})$. This can be seen as a way to "‘switch"’ between frames. For more information on this kind of matrix, see the Section 1.2.1.4.

1.1.9.3 The Cross Gram Matrix Of A Frame And Its Dual

From the Examples 1.1.2 it is to be expected that the matrix $G_{g_k, \tilde{g}_i}$ has very special properties. For example the following.

Lemma 1.1.43 *Let (g_k) be a frame in $\mathcal{H}$. Then*

1. *$G_{g_k, \tilde{g}_k}$ is self-adjoint.*
2. *$G_{g_k, \tilde{g}_k} = G_{\tilde{g}_k, g_k}$*

3. $G_{g_k,\tilde{g}_k} = G_{g_k} \circ G_{\tilde{g}_k}$

Proof:
1.) & 2.)

$$(G_{g_k,\tilde{g}_k})_{j,i} = \left(\left\langle S_g^{-1} g_i, g_j \right\rangle\right)_{j,i} = \left(\left\langle g_i, S_g^{-1} g_j \right\rangle\right)_{j,i} = \left(\overline{\left\langle S_g^{-1} g_j, g_i \right\rangle}\right)_{j,i}$$

As $S_{g_k}^{-1} = S_{\tilde{g}_k}$, this operator is self-adjoint and so $(G_{g_k,\tilde{g}_k})_{j,i} = (G_{\tilde{g}_k,g_k})_{j,i}$ and both these matrices are self-adjoint

3.)

$$G_{g_k,\tilde{g}_k} = C_{g_k} \circ D_{\tilde{g}_k} = C_{g_k} \circ D_{g_k} \circ C_{\tilde{g}_k} \circ D_{\tilde{g}_k} = G_{g_k} \circ G_{\tilde{g}_k}$$

□

From Lemma 1.1.42 we know that this matrix maps onto $ran(C_{g_k})$. Even more it is the orthogonal projection from l^2 to $ran(C_{g_k})$ as stated in

Lemma 1.1.44 ([23] 5.3.6) *Let (f_k) be a frame sequence, then the orthogonal projection P from l^2 onto $ran(C)$ is given by*

$$Q(c_k) = \left(\left\langle \sum_l c_l S^{-1} f_l, f_j \right\rangle \right)_j = G_{\tilde{f}_l, f_l} c$$

1.1.10 Frames And The Pseudoinverse

We can collect a lot of possible descriptions of the pseudoinverse of the synthesis operator or the Gram matrix of a frame, refer to [21].

From Proposition 1.1.9 and Proposition A.4.46 we can deduce the following property as both operators are the minimal norm solutions of $D_{f_k} h = c$.

Proposition 1.1.45 *Let $\{f_k\}$ be a frame for $\mathcal{H}$, then*

$$D_{f_k}^{\dagger} = C_{\tilde{f}_k}$$

This result can also be found in [21]. A direct consequence from this result and Lemma A.4.45 is

Corollary 1.1.46 *Let $\{f_k\}$ be a frame for $\mathcal{H}$, then*

$$C_{f_k}^{\dagger} = D_{\tilde{f}_k}$$

Let us state some more results from [21]:

Lemma 1.1.47 ([21]) *Let (f_k) be a frame for $\mathcal{H}$.*

1. *Let* $\mathcal{M}_{i,j} = \left\langle D^\dagger e_j, a_i \right\rangle$ *be the matrix for* $D^\dagger$ *for the ONBs* $(e_i) \subseteq \mathcal{H}$ *and* $(a_j) \subseteq l^2$*. Then*
$$S^{-1} f_k = \sum_i \overline{\mathcal{M}_{i,j}} e_i$$

2. $$G_{f_k}^\dagger = G_{\tilde{f}_k}$$

From 2.) we can directly deduce a property similar to 1.) :

Corollary 1.1.48 *Let* (g_k) *be a frame for* $\mathcal{H}$*. Then the coefficients of the dual frame in the frame expansion are the entries of the pseudo-inverse of the Gram matrix.*
$$\tilde{g}_j = \sum_k \left(G^\dagger\right)_{k,i} g_k$$

Proof: Clearly

$$\tilde{g}_j = \sum_k \langle \tilde{g}_j, \tilde{g}_k \rangle g_k = \sum_k \left(G_{\tilde{g}_k}\right)_{k,i} g_k = \sum_k \left(G_{g_k}^\dagger\right)_{k,i} g_k$$

□

In applications this relationship can be used to calculate the dual frame.

Corollary 1.1.49
$$G_{g_k,\tilde{g}_k} = G_{g_k}^\dagger G_{g_k}$$

Proof: This is just a combination of the second point of Lemma 1.1.47 and Lemma 1.1.43. □

In [58] this property is extended to

Lemma 1.1.50 ([58] Lemma 3.1)

$$G_{\tilde{g}_k} = \left(G_{g_k}^\dagger\right)^2 G_{g_k}$$

We can now find several different possibilities to describe the pseudoinverse of the synthesis operator of a frame:

Proposition 1.1.51 *Let* (g_k) *be a frame on* $\mathcal{H}$*.*

$$D_{g_k}^\dagger = C_{g_k} S_{g_k}^{-1} = G_{\tilde{g}_k,g_k} C_{\tilde{g}_k} = G_{g_k}^\dagger C_{g_k}$$

Proof:

$$D_{g_k}^{\dagger} = C_{\tilde{g}_k} = C_{\tilde{g}_k} D_{\tilde{g}_k} C_{g_k} = G_{\tilde{g}_k} C_{g_k} = G_{g_k}^{\dagger} C_{g_k}$$

D is surjective, so

$$D_{g_k}^{\dagger} = D_{g_k}^{*} \left(D_{g_k} D_{g_k}^{*} \right)^{-1} = C_{g_k} \left(D_{g_k} C_{g_k} \right)^{-1} = C_{g_k} S_{g_k}^{-1} =$$

$$= C_{g_k} S_{\tilde{g}_k} = C_{g_k} D_{\tilde{g}_k} C_{\tilde{g}_k} = G_{g_k, \tilde{g}_k} C_{\tilde{g}_k}$$

□

1.1.10.1 Best Approximation By Frame Sequences

In applications we very often have the problem to find an approximation of a certain object, for example finding the Gabor multiplier which approximates a given matrix, see Section 2.7.3. In a Hilbert space setting, where the interesting objects are in a space spanned by a frame sequence, Proposition 1.1.14 gives the right tool for this problem, because then we know that the best approximation is just the orthogonal projection on this space, which is given by

$$P(f) = \sum_k \langle f, \tilde{g}_k \rangle g_k = (*)$$

A disadvantage of this formula for practical solutions is that the dual frame has to be calculated. This can be time-consuming and is not needed per-se. But we can use the formulas we have established previously to get

$$(*) = D_{g_k} C_{\tilde{g}_k} = D_{g_k} D_{\tilde{g}_k}^{\dagger} = D_{g_k} G_{g_k}^{\dagger} C_{g_k}$$

Here we can avoid calculating the dual frame directly, instead using the existing algorithm to calculate the pseudoinverse.

Theorem 1.1.52 *Let (g_k) be a frame sequence in $\mathcal{H}$. Let $V = \overline{span(g_k)}$. The best approximation of an arbitrary element $f \in \mathcal{H}$ is*

$$P(f) = D_{g_k} G_{g_k}^{\dagger} C_{g_k} f$$

See Section 1.3.9 for an application with frame multiplier.

1.1.11 Tight Frames

Tight frames are very attractive, as the dual frame can be easily calculated via $\tilde{g}_k = \frac{1}{A} \cdot g_k$. This is because we know that $AI \leq S \leq BI$ and so for a tight frame we have $S = AI$.

We are using the term *normalized* for a sequence (f_k) where $\|f_k\|_{\mathcal{H}} = 1$, in accordance with the definition semi-normalized, see Definition 1.1.9. This is sometimes used in the literature in connection with tight frames for the case where $A = 1$. Another, more precise, name for a tight frame with bound $A = 1$ is *Parseval frame*, cf. [6], as this kind of frame fulfills the Parseval equation, cf. Theorem A.4.11.

Lemma 1.1.53 ([63], Lemma 5.1.6.(a)) *Let $\{g_k\}$ be a tight normalized frame with the frame bound 1. Then $\{g_k\}$ is an ONB.*

We'll repeat the proof for insight:
Proof:

$$\|f\|^2 = \sum_k |\langle f, g_k \rangle|^2 \ \ \forall f \Longrightarrow$$

For any l

$$\|g_l\|^2 = \sum_k |\langle g_l, g_k \rangle|^2 = |\langle g_l, g_l \rangle|^2 + \sum_{k \neq l} |\langle g_l, g_k \rangle|^2 = \|g_l\|^4 + \sum_{k \neq l} |\langle g_l, g_k \rangle|^2$$

$$1 = 1 + \sum_{k \neq l} |\langle g_l, g_k \rangle|^2$$

$$\Longrightarrow 0 = \sum_{k \neq l} |\langle g_l, g_k \rangle|^2 \Longrightarrow \langle g_l, g_k \rangle = 0 \ \forall k \neq l$$

□

If we have a frame for $\mathcal{H}$, we can find a tight frame by:

Lemma 1.1.54 ([63], Lemma 5.1.6.(b)) *Let $\{g_k\}$ be a frame. Then $\{S^{-1/2} g_k\}$ is a tight frame with $A = 1$.*

$S^{-1/2}$ can be defined as S is positive.

As mentioned above a tight frame with the bound $A = 1$ is called Parseval frame. It is also called a *self-dual frame* as this is equivalent to $S = Id$ and $g = \tilde{g}$.

We can show:

Lemma 1.1.55 *Let $\{g_k\}$ be a tight frame with the frame bound A. Then $\left\{ \frac{g_k}{\sqrt{A}} \right\}$ is a tight frame with frame bound 1.*

Proof: The frame bound $A > 0$.

$$A \cdot \|f\|^2 = \sum_k |\langle f, g_k \rangle|^2 \Longrightarrow$$

$$\sum_k \left| \left\langle f, \frac{g_k}{\sqrt{A}} \right\rangle \right|^2 = \sum_k \left| \frac{1}{\sqrt{A}} \right|^2 |\langle f, g_k \rangle|^2 = \frac{1}{A} \sum_k |\langle f, g_k \rangle|^2 = \frac{1}{A} \cdot A \cdot \|f\|^2 = \|f\|^2$$

□

Another fact, which is different to what might be expected from experience with orthonormal bases and finite dimensional spaces is that we know that in every separable $\mathcal{H}$ there exists a tight frame, which is norm bounded below and does not contain a basis. This was shown in [23] 6.4.2. basis. This was shown in [23] 6.4.2.

1.1.11.1 Exact Tight Frames

Let us now analyze exact and tight frames. This should be very near to an ONB. Every ONB is clearly both exact and tight. If an ONB is part of the sequence, then due to the minimality this has to be already the whole sequence. We know that the exact frames are the Riesz bases. ONBs with constant factors, $a \cdot ONB$, are clearly tight and exact. But this already includes all possible cases.

Corollary 1.1.56 *The exact, tight frames are exactly the ONBs scaled by a fixed scalar $a \neq 0$. $(g_k) = (a \cdot e_k)$ where (e_k) is an ONB.*

Proof: From 1.1.16 we get that $A = B \geq \|g_l\|^2$ and because the frame is exact and minimal $A \leq \|g_l\|^2$. Therefore $\|g_l\|^2 = A$ for all l.

Following 1.1.55 $\tilde{g}_l = \frac{g_l}{\sqrt{A}}$ form a tight frame with frame bounds 1. $\|\tilde{g}_l\|^2 = 1$. So by 1.1.53 $\tilde{g}_l$ form an ONB. □

This result can also be deduced from 1.1.53.

Let us now look at an ONB multiplied not by one constant but by a semi-normalized sequence $0 < a < |\lambda_k| < b$. Then the sequence $(\lambda_k e_k)$ is clearly not tight in general as $\sum_k |\langle e_i, \lambda_k e_k \rangle|^2 = |\lambda_i|^2 \|e_i\|_{\mathcal{H}}$. Clearly this sequence is a frame, but does it also stay a Riesz basis? This leads us directly to the questions in the next section.

1.1.12 Perturbation Of Frames

Lemma 1.1.57 *Let $\{\lambda_k\}$ be a semi-normalized sequence, Then if $\{g_k\}$ is a Bessel sequence, frame or Riesz basis with bounds A and B, $\{\lambda_k \cdot g_k\}$ is also one with bounds $a^2 \cdot A$ and $b^2 \cdot B$.*

Proof:

$$|\langle f, c_k g_k\rangle|^2 = |c_k|^2 |\langle f, g_k\rangle|^2$$

$$\Longrightarrow \sum_k |\langle f, c_k g_k\rangle|^2 = \sum_k |c_k|^2 |\langle f, g_k\rangle|^2 = (*)$$

$$(*) \leq b^2 \sum_k |\langle f, g_k\rangle|^2 \leq b^2 B \|f\|^2$$

$$(*) \geq a^2 \sum_k |\langle f, g_k\rangle|^2 \geq a^2 A \|f\|^2$$

If $\{g_k\}$ is a Riesz basis, then we know now that $\{\lambda_k \cdot g_k\}$ is a Bessel sequence. So we only have to show that $D_{\lambda_k g_k}$ is injective. Let

$$D_{\lambda_k g_k}(c) = D_{\lambda_k g_k}(d) \Longrightarrow \sum_k c_k \lambda_k \cdot g_k = \sum_k d_k \lambda_k \cdot g_k$$

$$\Longrightarrow c_k \lambda_k = d_k \lambda_k \Longrightarrow c_k = d_k$$

□

The standard question of perturbation theory is whether the frame related properties of a sequence is shared with 'similar' sequences.

Theorem 1.1.58 ([23] Theorem 15.1.1.) *Let $(f_k)_{k=1}^{\infty}$ be a frame for $\mathcal{H}$. Let $(g_k)_{k=1}^{\infty}$ be a sequence in $\mathcal{H}$. If there exist $\lambda, \mu \geq 0$ such that $\lambda + \frac{\mu}{\sqrt{A}} < 1$ and*

$$\left\| \sum_k c_k (f_k - g_k) \right\|_{\mathcal{H}} \leq \lambda \left\| \sum_k c_k f_k \right\|_{\mathcal{H}} + \mu \|c\|_{l^2}$$

for all finite scalar sequences c, then (g_k) is a frame with bounds

$$A\left(1 - \left(\lambda + \frac{\mu}{\sqrt{A}}\right)\right)^2, B\left(1 + \lambda + \frac{\mu}{\sqrt{B}}\right)^2$$

Moreover if (f_k) is a Riesz bases, (g_k) is, too.

This can easily formulated for Bessel sequences using parts of the proofs in [23]:

Lemma 1.1.59 *Let $(f_k)_{k=1}^{\infty}$ be a Bessel sequence for $\mathcal{H}$. Let $(g_k)_{k=1}^{\infty}$ be a sequence in $\mathcal{H}$. If there exist $\lambda, \mu \geq 0$ such that*

$$\left\| \sum_k c_k (f_k - g_k) \right\|_{\mathcal{H}} \leq \lambda \left\| \sum_k c_k f_k \right\|_{\mathcal{H}} + \mu \left\| c \right\|_{l^2}$$

for all finite scalar sequences c, then (g_k) is a Bessel sequence with bound

$$B \left(1 + \lambda + \frac{\mu}{\sqrt{B}} \right)^2$$

Proof: Let c be a finite sequence, then

$$\left\| \sum_k c_k g_k \right\|_{\mathcal{H}} \leq \left\| \sum_k c_k f_k + \sum_k c_k (g_k - f_k) \right\|_{\mathcal{H}} \leq$$

$$\leq \left\| \sum_k c_k f_k \right\|_{\mathcal{H}} + \left\| \sum_k c_k (g_k - f_k) \right\|_{\mathcal{H}} \leq \left\| \sum_k c_k f_k \right\|_{\mathcal{H}} + \lambda \left\| \sum_k c_k f_k \right\|_{\mathcal{H}} + \mu \left\| c \right\|_{l^2} =$$

$$\overset{Th. 1.1.3}{=} \left(\sqrt{B} \left\| c \right\|_2 \right) \cdot (1 + \lambda) + \mu \left\| c \right\|_2$$

For infinite sequences we know that for $m > n$

$$\left\| \sum_{k=1}^{m} c_k g_k - \sum_{k=1}^{n} c_k g_k \right\|_{\mathcal{H}} = \left\| \sum_{k=n+1}^{m} c_k g_k \right\|_{\mathcal{H}} \leq$$

$$\leq (1 + \lambda) \left\| \sum_{k=n+1}^{m} c_k f_k \right\|_{\mathcal{H}} + \mu \sqrt{\sum_{k=n+1}^{m} |c_k|^2}$$

The first term is convergent because (f_k) is a Bessel sequence, the second because $c \in l^2$. Therefore $\sum\limits_{k=1}^{n} c_k g_k$ is a Cauchy sequence and therefore convergent. So (g_k) is a Bessel sequence.
With Theorem 1.1.11, we know that

$$\left(\sqrt{B} \cdot (1 + \lambda) + \mu \right)^2 = B \left(1 + \lambda + \frac{\mu}{\sqrt{B}} \right)^2$$

is a Bessel bound. □

For frame sequence an equivalent statement is not possible as can be seen in [23] Example 15.3.1. But for Riesz sequence it is possible to formulate

Theorem 1.1.60 ([23] Theorem 15.3.2.) *Let $(f_k)_{k=1}^{\infty}$ be a Riesz sequence in $\mathcal{H}$. Let $(g_k)_{k=1}^{\infty}$ be a sequence in $\mathcal{H}$. If there exist $\lambda, \mu \geq 0$ such that $\lambda + \frac{\mu}{\sqrt{A}} < 1$ and*

$$\left\| \sum_k c_k \left(f_k - g_k \right) \right\|_{\mathcal{H}} \leq \lambda \left\| \sum_k c_k f_k \right\|_{\mathcal{H}} + \mu \left\| c \right\|_{l^2}$$

for all finite scalar sequences c, then (g_k) is a Riesz sequence with bounds

$$A \left(1 - \left(\lambda + \frac{\mu}{\sqrt{A}} \right) \right)^2, B \left(1 + \lambda + \frac{\mu}{\sqrt{B}} \right)^2$$

We can specialize and rephrase Theorem 1.1.58, Lemma 1.1.59 and Theorem 1.1.60. For that let us denote the normed vector space of finite sequences in l^2 by $c_c^2 = (c_c, \|.\|_2)$.

Proposition 1.1.61 *Let (f_k) be a Bessel sequence, frame, Riesz sequence or Riesz basis for $\mathcal{H}$. Let (g_k) be a sequence in $\mathcal{H}$. If there exists μ such that*

$$\|D_{f_k} - D_{g_k}\|_{c_c^2 \to \mathcal{H}} \leq \mu < \sqrt{A}$$

then g_k is a Bessel sequence with bound

$$B \left(1 + \frac{\mu}{\sqrt{B}} \right)^2$$

respectively a frame or Riesz basis with bounds

$$A \left(1 - \frac{\mu}{\sqrt{A}} \right)^2, B \left(1 + \frac{\mu}{\sqrt{B}} \right)^2$$

and

$$\|D_{f_k} - D_{g_k}\|_{l^2 \to \mathcal{H}} \leq \mu$$

If (f_k) is a Riesz basis, (g_k) is, too.

Proof: For every $c \in c_c$

$$\left\| \left(D_{f_k} - D_{g_k} \right) c \right\|_{\mathcal{H}} \leq \left\| D_{f_k} - D_{g_k} \right\|_{Op} \|c\|_2 \leq \mu \|c\|_2$$

This is just the condition in Theorem 1.1.58 respectively Lemma 1.1.59 with $\lambda = 0$ and $\mu < \sqrt(A)$, so that $\lambda + \frac{\mu}{\sqrt{A}} < 1$.

Because g_k is a Bessel sequence, we know that $D_{g_k} : l^2 \to \mathcal{H}$ is well defined. Because c_c^2 is dense in l^2, therefore

$$\|D_{f_k} - D_{g_k}\|_{l^2 \to \mathcal{H}} = \|D_{f_k} - D_{g_k}\|_{c_c^2 \to \mathcal{H}} \leq \mu$$

□

This also means

Corollary 1.1.62 *Let (f_k) be a Bessel sequence, frame, Riesz sequence respectively Riesz basis and $(g_k^{(n)})$ sequences with*

$$\left\| D_{g_k^{(n)}} - D_{f_k} \right\|_{c_c^2 \to \mathcal{H}} \to 0$$

for $n \to \infty$. Then there exists an N such that $(g_k^{(n)})$ are Bessel sequences, frames, Riesz sequences respectively Riesz bases for $n \geq N$. For the optimal upper frame bounds $B_{opt}^{(n)} \to B_{opt}$. And

$$\left\| D_{g_k^{(n)}} - D_{f_k} \right\|_{l^2 \to \mathcal{H}} \to 0$$

for $n \to \infty$.

Proof: The first property is a direct consequence from Proposition 1.1.61.

For all $\epsilon > 0$ there is an N such that for all $n \geq \max\{N(\epsilon), N(A)\}$

$$\left\| D_{g_k^{(n)}} \right\|_{Op} \leq \|D_{f_k}\|_{Op} + \left\| D_{f_k} - D_{g_k^{(n)}} \right\|_{Op} \leq B + \epsilon$$

□

We will use this result in Section 1.3 and Section 2.3.

A simple way to measure the similarity of two frames would be in a uniform sense, using the supremum of $\|g_k - f_k\|_{\mathcal{H}}$, but this is not a good measure in general for frames and related sequences.

Example 1.1.3 :

Let $(\delta_i)_j = \delta_{i,j}$ be the standard basis of l^2 and consider the sequence

$$\left(e_i^{(\delta)}\right)_j = \begin{cases} \delta & j < i \\ 1 & j = i \\ 0 & otherwise \end{cases}$$

Clearly $\left\|\delta_k - e_k^{(\delta)}\right\|_\infty = \delta$. But this sequence cannot be a frame because $\left\|e_i^{(\delta)}\right\|_2 = (i-1)\,\delta + 1 \to \infty$ and therefore it cannot have an upper bound.

But it is rather easy to show that if another similarity measure is used for the frame elements the frame property is kept:

Corollary 1.1.63 *Let (g_k) be a Bessel sequences, frame, Riesz sequence respectively a Riesz basis. Let (f_k) be a sequence with*

$$\sum_k \|g_k - f_k\|_{\mathcal{H}}^2 < A$$

then (f_k) is a Bessel sequences, frame, Riesz sequence or Riesz basis.

If $(f_k^{(l)})$ are sequences such that for all ϵ there exists an $N(\epsilon)$ with

$$\sum_k \left\|g_k - f_k^{(l)}\right\|_{\mathcal{H}}^2 < \epsilon$$

for all $l \geq N(\epsilon)$, then $(f_k)^{(l)}$ is a Bessel sequences, frame, Riesz sequence respectively a Riesz basis for all $l \geq N(A)$ and for all $l > \max\{N(\epsilon), N(A)\}$ with the optimal upper frame bound $B_{opt}^{(l)} \to B_{opt}$. And

$$\left\|C_{f_k^{(l)}} - C_{g_k}\right\|_{Op} < \epsilon$$

$$\left\|D_{f_k^{(l)}} - D_{g_k}\right\|_{Op} < \epsilon$$

and for all $l > \max\{N(\epsilon), N(A), N(1)\}$

$$\left\|S_{f_k^{(l)}} - S_{g_k}\right\|_{Op} < \epsilon \cdot \left(\sqrt{B+1} \cdot \sqrt{B}\right).$$

Proof: Let $c \in C_c$, then

$$\|D_{f_k} c - D_{g_k} c\|_{\mathcal{H}} \leq \left\|\sum_k c_k (f_k - g_k)\right\|_{\mathcal{H}} \leq \sum_k |c_k| \, \|g_k - f_k\|_{\mathcal{H}} \leq$$

$$\sqrt{\sum_k |c_k|^2} \sqrt{\sum_k \|g_k - f_k\|_{\mathcal{H}}^2}$$

$$\Longrightarrow \|D_{f_k} - D_{g_k}\|_{Op} \leq \sqrt{\sum_k \|g_k - f_k\|_{\mathcal{H}}^2}$$

So in the first case $\|D_{f_k} - D_{g_k}\|_{Op} < \sqrt{A}$ and therefore (f_k) forms a Bessel sequence, frame, Riesz sequence or Riesz basis.

In the second case we get $\left\| D_{f_k^{(l)}} - D_{g_k} \right\|_{Op} < \epsilon$.

$$\left\| C_{f_k^{(l)}} f - C_{g_k} \right\|_{Op} = \left\| D^*_{f_k^{(l)}} - D^*_{g_k} \right\|_{Op} = \left\| D_{f_k^{(l)}} - D_{g_k} \right\|_{Op} < \epsilon$$

$$\left\| S_{f_k^{(l)}} - S_{g_k} \right\|_{Op} = \left\| D_{f_k^{(l)}} \circ C_{f_k^{(l)}} - D_{g_k} \circ C_{g_k} \right\|_{Op} =$$

$$= \left\| D_{f_k^{(l)}} \circ C_{f_k^{(l)}} - D_{f_k^{(l)}} \circ C_{g_k} + D_{f_k^{(l)}} \circ C_{g_k} - D_{g_k} \circ C_{g_k} \right\|_{Op} \leq$$

$$\leq \left\| D_{f_k^{(l)}} \right\|_{Op} \left\| C_{f_k^{(l)}} - C_{g_k} \right\|_{Op} + \left\| D_{f_k^{(l)}} - D_{g_k} \right\|_{Op} \|C_{g_k}\|_{Op} = (*)$$

Due to Corollary 1.1.62 there is an $N(1)$ such that for all $l \geq N(1)$

$$\left\| D_{f_k^{(l)}} \right\|_{Op} \leq \sqrt{B+1}$$

and so

$$(*) \leq \sqrt{B+1}\epsilon + \epsilon\sqrt{B} = \epsilon \cdot \left(\sqrt{B+1} \cdot \sqrt{B} \right)$$

□

Because for all sequences $\|c\|_1 \geq \|c\|_2$ a corresponding property is true in an l^1-sense:

Corollary 1.1.64 *Let (g_k) be a Bessel sequences, frame, Riesz sequence respectively a Riesz basis. Let (f_k) be a sequence with*

$$\sum_k \|g_k - f_k\|_{\mathcal{H}} < A$$

then (f_k) is a Bessel sequences, frame, Riesz sequence or Riesz basis.

If $(f_k^{(l)})$ are sequences such that for all ϵ there exists an $N(\epsilon)$ with

$$\sum_k \left\| g_k - f_k^{(l)} \right\|_{\mathcal{H}} < \epsilon$$

for all $l \geq N(\epsilon)$, then $(f_k)^{(l)}$ is a Bessel sequences, frame, Riesz sequence respectively a Riesz basis for all $l > \max\{N(\epsilon), N(A)\}$ and the optimal upper frame bound $B_{opt}^{(l)} \to B_{opt}$ for $l \to \infty$.

$$\left\| C_{f_k^{(l)}} - C_{g_k} \right\|_{Op} < \epsilon$$

$$\left\| D_{f_k^{(l)}} - D_{g_k} \right\|_{Op} < \epsilon$$

and for all $l > \max\{N(\epsilon), N(A), N(1)\}$

$$\left\| S_{f_k^{(l)}} - S_{g_k} \right\|_{Op} < \epsilon \cdot \left(\sqrt{B+1} \cdot \sqrt{B} \right).$$

1.2 Frames And Finite Dimensional Spaces

1.2.1 Frames In Finite Dimensional Spaces

As mentioned above the typical properties of frames can be understood more easily in the context of finite-dimensional vector spaces. Let us gather some results from [23], letting V denote a finite dimensional space with dimension N.

Proposition 1.2.1 ([23] 1.1.2) *Let $(f_k)_{k=1}^M$ be a sequence in V. Then it is a frame for $\overline{span}\{f_k\}$.*

So in finite dimensional spaces all finite sequences that span the whole space are exactly the finite frames. Clearly sequences that are Riesz bases are linear independent and so are bases.

A result dealing with the eigenvalues of the frame operator is the following

Proposition 1.2.2 ([23] 1.2.1 & 1.2.2)*Let $(f_k)_{k=1}^M$ be a frame for V. Then*

1. *The optimal lower frame bound is the smallest eigenvalue of S, and the optimal upper frame bound is the largest eigenvalue.*
2. *Let (λ_k) denote the sequence of eigenvalues of S. Then*

$$\sum_{k=1}^{n} \lambda_k = \sum_{k=1}^{M} \|f_k\|_{\mathbb{C}^n}^2$$

3. *The condition number of S is $\chi(S) = \frac{B_{opt}}{A_{opt}}$.*

The convergence of the sum $\sum_k \|f_k\|_{\mathbb{C}^N}^2$ will again be investigated later in this section. For finite frames this sum is certainly finite. For infinite frames the question whether this sum is convergent will be answered in Section 1.2.2. In the next section we deal with finite dimensional spaces, and if there are frames for such spaces with infinite many elements.

1.2.1.1 An Example Of An Infinite Frame In An Finite Dimensional Space

The finite sequences that span the whole space are not the only frames. There are also frames with infinite many elements in finite dimensional spaces.

Example 1.2.1 :

1. Take a basis $(e_i | i = 1, .., N)$ in $\mathbb{C}^N$ and let $e_k^{(l)} = \frac{1}{l} \cdot e_k$ for $l = 1, 2, ...$ Then $(e_k^{(l)})$ is a tight frame, as

$$\sum_{k,l} \left| \left\langle f, e_k^{(l)} \right\rangle \right|^2 = \sum_{l=1}^{\infty} \sum_{k=1}^{N} \left| \left\langle f, \frac{1}{l} \cdot e_k \right\rangle \right|^2 =$$

$$= \sum_{l=1}^{\infty} \frac{1}{|l|^2} \sum_{k=1}^{N} |\langle f, e_k \rangle|^2 = \sum_{l=1}^{\infty} \frac{1}{|l|^2} \|f\|_{\mathcal{H}} = \|f\|_{\mathcal{H}} \cdot \frac{\pi^2}{6}$$

So $S^{-1} e_k^{(l)} = \frac{6}{\pi^2} e_k^{(l)}$.

2. The same is possible for $e_k^{(l)} = \frac{1}{l^2} \cdot e_k$ for $l = 1, 2, ...$ This is again a tight frame with the bound $A = \frac{\pi^4}{90}$.

It seems rather "strange" to use infinitely many elements for a frame in a finitely dimensional space, but these examples are a good tool to find counter-examples for some properties, which might be expected if properties of ONBs are generalized.

It can be shown that the condition $\sum_{k \in K} |f_k|^2 < \infty$ is equivalent for the space to be finite dimensional, see 1.2.15.

1.2.1.2 Frames And ONBs

We will revisit Proposition 1.1.26, where it was stated that frames are exactly the images of ONBs in infinite dimensional space. Can something similar be done to finite dimensions?

A well-known result regarding this question is

Theorem 1.2.3 ([23] Theorem 1.3.2) *Let $(f_k)_{k=1}^M$ be a frame for $\mathbb{C}^N$. Then the vectors f_k can be considered as the first coordinates of some vectors $(g_k)_{k=1}^M$ in $\mathbb{C}^M$ constituting a basis for $\mathbb{C}^M$. If $(f_k)_{k=1}^M$ is tight, then these (g_k) form an orthonormal basis.*

A consequence of this result is

Proposition 1.2.4 ([23] Theorem 1.3.2)*Let M be a $m \times n$ matrix with $m \geq n$. Then the columns of this matrix constitute a basis for $\mathbb{C}^M$ if and only if the rows constitute a frame for $\mathbb{C}^N$.*

Or stated in the words of [99]: *"Every matrix with full rank represents a frame."*

Therefore we now arrive at a direct analog to Proposition 1.1.26

Proposition 1.2.5 *Let $\{e_k\}_{k=0}^{\infty}$ be an arbitrary ONB for l^2. The frames for $\mathbb{C}^N \subseteq l^2$ are precisely the families $\{Ue_k\}$, where $U : l^2 \to \mathbb{C}^N$ is a surjective operator.*

Proof: Let $(f_k)_{k=1}^M$ be a frame. Define $U : l^2 \to V$ with

$$U(e_k) = \begin{cases} f_k & k \leq M \\ 0 & otherwise \end{cases}$$

As (e_k) is an ONB this operator is well-defined and because (f_k) is a frame it is surjective. It is clearly bounded as it can be seen as a projection $l^2 \to \mathbb{C}^m$ compsed with $U|_{\mathbb{C}^M}$.

On the other hand as an operator into a finite dimensional space, U is bounded, so Proposition 1.1.26 tells us that $f_k = U(e_k)$ constitute a frame for $\mathbb{C}^n$. □

Corollary 1.2.6 *The frames with M elements in $\mathbb{C}^n$ are exactly the images of an ONB in $\mathbb{C}^M$ by a surjective operator.*

Proof: This is just a rephrasing of Theorem 1.2.3. □

As in Proposition 1.1.30 we can extend that result to

Corollary 1.2.7 *Let $\{e_k\}_{k=0}^{n}$ be an arbitrary ONB for $\mathbb{C}^n$. The Riesz bases for $\mathbb{C}^N$ are precisely the families $\{Ue_k\}$, where $U : l^2 \to \mathbb{C}^N$ is an invertible operator.*

An extension of this result to frame sequences (respectively Bessel) sequences is still possible, as every sequence is an image of an ONB with $U(e_k) = f_k$, but the result does not contain new information, as every arbitrary sequence fulfills is already a frame and Bessel sequence.

1.2.1.3 The Matrices Connected To Frames

Lemma 1.2.8 *Let $(g_k)_{k=1}^M$ be a frame in $\mathbb{C}^N$. The $N \times M$ matrix D*

$$D = \begin{pmatrix} | & | & & | \\ g_1 & g_2 & \dots & g_M \\ | & | & & | \end{pmatrix}$$

describes the synthesis operator $D : l^2 \to V$ such that $D_{g_k} c = D \cdot \mathfrak{p}_{1..M}(c)$, where $\mathfrak{p}_{1..M}$ is the projection $l^2 \to \mathbb{C}^M$.

Proof:

$$(D \cdot \mathfrak{p}_{1..M}(c))_i = \sum_{k=1}^{M} D_{i,k} c_k = \sum_{k=1}^{M} c_k \left(g_k\right)_i$$

□

We know $C = D^*$. In the finite dimensional case we just have to use transposition and complex conjugation and so:

Corollary 1.2.9 *Let $(g_k)_{k=1}M$ be a frame in $\mathbb{C}^N$. The $M \times N$ matrix C*

$$C = \begin{pmatrix} - & \overline{g_1} & - \\ - & \overline{g_2} & - \\ \vdots & & \vdots \\ - & \overline{g_M} & - \end{pmatrix}$$

describes the analysis operator $C : V \to \mathbb{C}^M \subseteq l^2$ such that $C_{g_k} f = C \cdot f$.

The frame operator is defined as $S = D \circ D^*$. S is a $n \times n$ matrix $S = D \cdot D^*$. This matrix can be represented very easily by

Proposition 1.2.10 *Let $\{g_k \,| k \in K\}$ and $\{\gamma_k \,| k \in K\}$ be families of elements in $\mathcal{H}$ and let S_{g_k,γ_k} be the associated frame matrix, then*

$$\left(S_{g_k,\gamma_k}\right)_{m,n} = \left(\sum_{k \in K} \gamma_k \otimes \overline{g}_k\right)_{m,n} = \sum_{k \in K} \left(\gamma_k\right)_m \dot{\left(\overline{g}_k\right)}_n$$

Proof:

$$S_{i,j} = \left(D_{\gamma_k} \cdot C_{g_k}\right)_{i,j} = \sum_k \left(D_{\gamma_k}\right)_{i,k} \left(C_{g_k}\right)_{k,j} = \sum_k (\gamma_k)_i (\overline{g}_k)_j$$

□

This result was motivated by the results for Gabor frames, see Section 3.1.2.

We can also express this term as product of an $N \times 1$ and a $1 \times N$ matrix:

$$S_{g_k,\gamma_k} = \sum_{k \in K} g_k^* \cdot \gamma_k.$$

For the pseudoinverse Lemma 1.1.47 can be interpreted very easily in $\mathbb{C}^n$ as

Corollary 1.2.11 *Let $\{f_k\}$ be a frame in $\mathbb{C}^n$, let D be its synthesis operator, then*

$$D^\dagger = \begin{pmatrix} - & \tilde{f}_1 & - \\ - & \tilde{f}_2 & - \\ \vdots & & \vdots \\ - & \tilde{f}_k & - \end{pmatrix}$$

1.2.1.4 Frame Transformation

In linear algebra we learn that unitary operators are exactly the transformation from one orthonormal basis to the other and invertible matrices are exactly describing the change between arbitrary bases. So "switching" from one ONB to another is rather straight forward. But what about frames? How can a representation be changed from one frame to another in the finite dimensional case?

Let $(g_k)_{k=1}^M$ and $(f_i)_{i=1}^N$ be two frames. We want to find a way to switch between these frames. The naive way to do frame transformation would be just to combine the analysis and synthesis operator of the two frames. So we could use

$$g = \sum_k \langle f, g_k \rangle f_k$$

But if $M \neq N$ we get a problem with this definition, with the different size of the index sets. Either the vector of coefficients is too long or too short. In the first case we could set all $f_k = 0$ for $k > N$ and in the second case $g_k = 0$ for $k > M$, we could call that 'zero-padding' in parallel to the concept in discrete signal processing, see e.g. [97]. But this certainly cannot be an injective or surjective operator, but the ultimate goal would be to get identity, perfect reconstruction. Instead trying to overcome this short-coming by some 'periodization' or 'aliasing' like in discrete signal processing we can just use Lemma 1.1.42 to get

Proposition 1.2.12 *Let $(g_k)_{k=1}^M$ and $(f_i)_{i=1}^N$ be two frames. The $M \times N$ matrix $G = G_{f_j, \tilde{g}_k}$ maps $ran(C_{g_k})$ onto $ran(C_{f_k})$ such that*

$$f = \sum_{i=1}^M \langle f, g_i \rangle \tilde{g}_i = \sum_{i=1}^N \left(G \cdot \{\langle f, g_k \rangle\}_{k=1}^M \right)_i \tilde{f}_i$$

The $M \times N$ matrix $G = G_{\tilde{f}_j, g_k}$ maps $ran(C_{\tilde{g}_k})$ onto $ran(C_{\tilde{f}_k})$ such that

$$f = \sum_{i=1}^M \langle f, \tilde{g}_i \rangle g_i = \sum_{i=1}^N \left(G \cdot \{\langle f, \tilde{g}_k \rangle\}_{k=1}^M \right)_i f_i$$

Proof: This is just Lemma 1.1.42 rephrased. □

So analogue to the basis transformation matrix defined in linear algebra, we can define:

Definition 1.2.1 *We call $G = G_{f_j,\tilde{g}_k}$ from Proposition 1.2.12 the* **frame transformation matrix**.

As Lemma 1.1.42 is valid also for infinite-dimensional spaces, the statements in this section are not restricted to finite-dimensional spaces.

1.2.1.5 Inverting The Frame Operator

In Section 3.4 we will present a new algorithm for inverting the frame algorithm in the case of Gabor frames. For the calculation of the canonical dual it is necessary to invert the frame operator. There are several known algorithms, but we will close this section on frames in finite dimensional spaces with a compilation of two of them, the frame algorithm and the conjugate gradient method.

We start with the frame algorithm:

Proposition 1.2.13 ([36] Theorem III) *Let $(f_k)_{k=}^{M}$ be a frame for $\mathbb{C}^N$ with frame bounds A,B. Given $f \in \mathbb{C}^N$, define the sequence $(g_k)_{k=1}^{\infty}$ of vectors in $\mathbb{C}^n$ by*

$$g_0 = 0, g_k = g_{k-1} + \frac{2}{A+B} S\left(f - g_{k-1}\right) \text{ for } k \geq 1.$$

Then

$$\|f - g_k\| \leq \left(\frac{B-A}{B+A}\right)^k \|f\|_{\mathbb{C}^n}.$$

This is the Neumann algorithm, see A.4.9, with the relaxation parameter $\frac{2}{A+B}$.

The conjugate gradient algorithm has the big advantage that the calculation of the frame bounds is not necessary:

Proposition 1.2.14 ([23] Lemma 1.2.5) *Let $(f_k)_{k=}^{M}$ be a frame for $\mathbb{C}^N$ with frame bounds A,B. Given $f \in \mathbb{C}^N$, $f \neq 0$, define the sequences $(g_k)_{k=1}^{\infty}$, $(r_k)_{k=1}^{\infty}$ and $(p_k)_{k=1}^{\infty}$ of vectors in $\mathbb{C}^N$ and $(\lambda_k)_{k=1}^{\infty}$ a sequence of numbers such that*

$$g_0 = 0, r_0 = p_0 = Sf, p_{-1} = 0$$

and for $k \geq 1$

$$\lambda_k = \frac{\langle r_k, p_k\rangle}{\langle p_k, Sp_k\rangle},$$

$$g_{k+1} = g_k + \lambda_k p_k,$$
$$r_{k+1} = r_k - \lambda_k S p_k,$$
$$p_{k+1} = S p_k - \frac{\langle S p_k, S p_k \rangle}{\langle p_k, S p_k \rangle} p_k - \frac{\langle S p_k, S p_{k-1} \rangle}{\langle p_{k-1}, S p_{k-1} \rangle} p_{k-1}$$

Then $g_k \to f$ for $k \to \infty$. Let A and B be the smallest and the largest eigenvalues of S, let $\sigma = \frac{\sqrt{B}-\sqrt{A}}{\sqrt{B}+\sqrt{A}}$, then the speed of convergence can be estimated by

$$\|f - g_k\| \leq \frac{2\sigma^k}{1+\sigma^{2k}} \|f\|$$

1.2.2 Classification Of Finite Dimensional Spaces With Frames

For an ONB (e_i) if the sum of the elements $\sum\limits_i \|e_i\|_{\mathcal{H}}$ is finite, the dimension of the space is finite and vice versa. The Example 1.2.1 shows that that is not true anymore with frames, as in this case

$$\sum_{l,k} \left\| e_k^{(l)} \right\| = \sum_{l=1}^{\infty} \sum_{k=1}^{N} \left\| \frac{1}{l} e_k \right\| = \sum_{l=1}^{\infty} \frac{N}{|l|} = \infty$$

But taking the square sum of the norms of the elements of a frame for $\mathcal{H}$ is an equivalent condition for $\mathcal{H}$ being finite dimensional:

Proposition 1.2.15 *Let (g_k) be a frame for the Hilbert space $\mathcal{H}$. Let (e_l) be an ONB for $\mathcal{H}$. Then the following statements are equivalent*

- $\sum_k \|g_k\|^2 < \infty$
- $\sum_l \|e_l\|^2 < \infty$
- *the space is finite dimensional.*

Proof: The equivalence of the second and third statements is clear.

$$\sum_k \|f_k\|^2 = \sum_k \sum_l |\langle f_k, e_l \rangle|^2 = \sum_l \sum_k |\langle f_k, e_l \rangle|^2 = (*)$$

On the one hand, as the first sum is finite, Fubini applies and this means

$$\sum_k \|f_k\|^2 = (*) \geq \sum_l A \cdot \|e_l\|^2$$

and so the sum $\sum_k \|e_k\|^2$ must be finite.

On the other hand if $\sum_l \|e_l\|^2 < \infty$ (and so again Fubini applies) then

$$\sum_k \|f_k\|^2 = (*) \leq \sum_l B \cdot \|e_l\|$$

$\square$

Corollary 1.2.16 *Let (f_k) be a frame and (e_k) an ONB for $\mathcal{H}$ then*

$$A \cdot \sum_l \|e_l\|^2 \leq \sum_k \|f_k\|^2 \leq B \cdot \sum_l \|e_l\|^2$$

or equivalently (for finite dimensional spaces)

$$A \cdot dim\,(\mathcal{H}) \leq \sum_k \|f_k\|^2 \leq B \cdot dim\,(\mathcal{H})$$

Proof: In the last proof we have shown that this is true for finite dimensional cases, and also that for infinite dimensional spaces all the sums are infinite. $\square$

And as an evident corollary we find:

Corollary 1.2.17 *Let $(f_k)_{k=1}^m$ be a tight frame in the finite dimensional $\mathcal{H}$ with $dim\mathcal{H} = n$, then*

$$\sum_k \|f_k\|^2 = A \cdot n \text{ resp. } \frac{\sum_k \|f_k\|^2}{n} = A$$

If all frame elements have equal length, i.e. $\|f_k\|_{\mathcal{H}} = d$ for all k, then

$$m \cdot d = A \cdot n \text{ resp. } \frac{m \cdot d}{n} = A$$

If this frame is normalized, then

$$A = \frac{m}{n}$$

Compare to [108], where a possibility to construct such a frame is given.

1.2.3 Frames And Hilbert-Schmidt Operators

With a very similar proof to the one of Proposition 1.2.15 it can be shown that

Proposition 1.2.18 *Let* (f_k) *be a frame and* (e_i) *an ONB in* $\mathcal{H}$*. Let* H *be an operator* $\mathcal{H} \to \mathcal{H}$*. Then*

$$A \cdot \sum_i \|H^* e_i\|_{\mathcal{H}}^2 \leq \sum_k \|H f_k\|_{\mathcal{H}}^2 \leq B \sum_i \|H^* e_i\|_{\mathcal{H}}^2$$

Proof:

$$\sum_k \|H f_k\|^2 = \sum_k \sum_l |\langle H f_k, e_l \rangle|^2 = \sum_l \sum_k |\langle f_k, H^* e_l \rangle|^2 = (*)$$

$$(*) \geq A \sum_l \|H^* e_l\|^2$$

and

$$(*) \leq B \sum_l \|H^* e_l\|^2$$

□

From the proof it is clear that the right inequality is true for Bessel sequences, so

Lemma 1.2.19 *Let* (f_k) *be a Bessel sequence and* (e_i) *an ONB in* $\mathcal{H}$*. Let* H *be an operator* $\mathcal{H} \to \mathcal{H}$*. Then*

$$\sum_k \|H f_k\|_{\mathcal{H}}^2 \leq B \sum_i \|H^* e_i\|_{\mathcal{H}}^2$$

As we know that an operator is Hilbert Schmidt if and only if it's adjoint operator is as well, we get:

Corollary 1.2.20 *An operator* $H : \mathcal{H} \to \mathcal{H}$ *is Hilbert Schmidt if and only if*

$$\sum_k \|H f_k\|_{\mathcal{H}}^2 < \infty$$

for one (and therefore for all) frame(s).

$$\sqrt{A}\, \|H\|_{\mathcal{HS}} \leq \sqrt{\sum_k \|H f_k\|_{\mathcal{H}}^2} \leq \sqrt{B}\, \|H\|_{\mathcal{HS}}$$

The idea for this was found in [24].

Of course like the frame operator of an ONB (which is the identity) every frame operator has a connection to the dimension of the space. It is very easy to prove

Lemma 1.2.21 *S is compact if and only if $\mathcal{H}$ is finite dimensional.*

Proof: If the space in finite dimensional every operator is compact.

If S is compact, then $S \circ S^{-1} = Id$ is compact (A.4.26) and therefore the space is finite dimensional. □

1.2.3.1 Matrix Representation Of $\mathcal{HS}$ Operators With Frames

We can now come back to the relationship of matrices and operators from $\mathcal{H}$ to $\mathcal{H}$, possibly infinite dimensional, stated in Section 1.1.7.3. We will look at Hilbert-Schmidt operators, see appendix A.4.5.4.

We now have the adequate tools to state that $\mathcal{HS}$ operators correspond exactly to the matrices having a bounded Frobenius norm, see Definition A.3.12:

Proposition 1.2.22 *Let (g_k) be a frame in $\mathcal{H}_1$ with bounds A, B, (f_k) in $\mathcal{H}_2$ with A', B'. Let M be a matrix in $l^{(2,2)}$ with $\|M\|_{2,2} = \sqrt{\sum\limits_i \sum\limits_j |M_{i,j}|^2}$. Then $\mathcal{O}(M) \in \mathcal{HS}(\mathcal{H}_1, \mathcal{H}_2)$, the Hilbert Schmidt class of operators from $\mathcal{H}_1$ to $\mathcal{H}_2$, with $\|\mathcal{O}(M)\|_{\mathcal{HS}} \leq \sqrt{BB'} \|M\|_{2,2}$.*

Let $O \in \mathcal{HS}$, then $\mathcal{M}(O) \in l^{(2,2)}$ with $\|\mathcal{M}(O)\|_{2,2} \leq \sqrt{BB'} \|O\|_{\mathcal{HS}}$.

Proof: 1.) Naturally the matrices in $l^{(2,2)}$ correspond to Hilbert-Schmidt operators on l^2 as

$$\|M\|_{HS}^{l^2 \to l^2} = \sqrt{\sum_i \|Me_i\|_{\mathcal{H}_1}^2}$$

$$(Me_i)_p = \sum_q M_{p,q} \delta_{i,q} = M_{p,i}$$

$$\Longrightarrow \sum_i \|Me_i\|_{\mathcal{H}_1}^2 = \sum_i \sum_p |M_{p,i}|^2$$

As the Hilbert-Schmidt class of operators is an ideal, see A.4.38, we know that

$$\|\mathcal{O}(M)\|_{\mathcal{HS}} = \|D_{f_k} \circ M \circ C_{g_k}\|_{\mathcal{HS}} \leq \|D_{f_k}\|_{Op} \cdot \|M\|_{\mathcal{HS}} \cdot \|C_{g_k}\|_{Op} = \sqrt{BB'} \|M\|_{2,2}$$

2.)

$$\|\mathcal{M}(O)\|_{2,2}^2 = \sum_k |\langle O g_l, f_k\rangle|^2 \leq B' \cdot \|O g_l\|_{\mathcal{H}_1}^2$$

$$\Longrightarrow \sum_l \sum_k |\langle Ogl, f_k\rangle|^2 \leq \sum_l B' \cdot \|O g_l\|_{\mathcal{H}_1}^2 \overset{Cor.1.2.20}{\leq} BB' \, \|O\|_{\mathcal{HS}}^2$$

□

The norm for matrices in $l^{2,2}$ is also called *Frobenius* or *Hilbert Schmidt* matrix norm, see Definition A.3.12 and also Section 3.1.2.

1.2.3.2 Frames In The Hilbert-Schmidt Class Of Operators

Section 1.1.7.3 tells us that an operator can be described by the matrix $\mathcal{M}^{(f_k, \tilde{g}_j)}(O)_{k,j} = \langle O\tilde{g}_j, f_k\rangle$. This is the matrix that maps $C_{f_k}(f) \mapsto C_{g_k}(Tf)$. It is identical to the $\mathcal{HS}$ inner product of O and $f_k \otimes \overline{\tilde{g}_j}$.

Theorem 1.2.23 *Let $(g_k)_{k\in K}$ be a sequence in $\mathcal{H}_1$, $(f_i)_{i\in I}$ in $\mathcal{H}_2$. Then*

1. *Let (g_k) and (f_i) be Bessel sequences with bounds B,B', then $(f_i \otimes \overline{g}_k)_{(i,k)\in I\times K}$ is a Bessel sequence for $\mathcal{HS}(\mathcal{H}_1, \mathcal{H}_2)$ with bound $\sqrt{B\cdot B'}$.*
2. *Let (g_k) and (f_i) be frames with bounds A, B and A, B'. Then $(f_i \otimes \overline{g}_k)_{(i,k)\in I\times K}$ is a frame for $\mathcal{HS}(\mathcal{H}_1, \mathcal{H}_2)$ with bounds $\sqrt{A\cdot A'}$ and $\sqrt{B\cdot B}$. A dual frame is $(\tilde{f}_i \otimes \overline{\tilde{g}}_k)$.*
3. *Let (g_k) and (f_i) be Riesz bases. Then $(f_i \otimes \overline{g}_k)_{(i,k)\in I\times K}$ is a Riesz basis for $\mathcal{HS}(\mathcal{H}_1, \mathcal{H}_2)$. The biorthogonal sequence is $(\tilde{f}_i \otimes \overline{\tilde{g}}_k)$.*

Proof: Suppose the operator $O \in \mathcal{HS}$, then

$$\mathcal{M}^{(f_k, g_j)}(O)_{k,j} = \langle O g_j, f_k\rangle_{\mathcal{H}_1} \overset{Cor.A.4.40}{=} \left\langle O, f_k \otimes \overline{g}_j\right\rangle_{\mathcal{HS}}$$

With Proposition 1.2.22 we know that the system $(\tilde{g}_k \otimes \overline{g}_j | k, j)$ forms a Bessel sequence for $\mathcal{HS}$ with bounds BB'. Following Theorem 1.1.12 and the fact that $\mathcal{M}$ is injective, we know that this system is a frame. Directly we can again use Proposition 1.2.22 with the dual frame. We know from Proposition 1.1.23 that

$$\|O\|_{\mathcal{HS}} = \left\|\mathcal{O}^{(\tilde{f}_k, \tilde{g}_j)} \mathcal{M}^{(f_k, g_j)}(O)\right\|_{\mathcal{HS}} = \left\|\mathcal{O}^{(\tilde{f}_k, \tilde{g}_j)}\left(\mathcal{M}^{(f_k, g_j)}(O)\right)\right\|_{\mathcal{HS}} \leq$$

$$\overset{Th.1.1.7}{\leq} \frac{1}{\sqrt{AA'}} \left\|M^{(f_k, g_j)}(O)\right\|_{\mathcal{HS}}$$

Therefore

$$AA' \|O\|_{\mathcal{H}S}^2 \leq \left\|M^{(f_k,g_j)}(O)\right\|_{\mathcal{H}S}^2 \leq BB' \|O\|_{\mathcal{H}S}^2$$

This is equal to

$$AA' \|O\|_{\mathcal{H}S}^2 \leq \sum_{k,j} \left|\langle O, f_k \otimes \overline{g}_j \rangle_{\mathcal{H}S}\right|^2 \leq BB' \|O\|_{\mathcal{H}S}^2$$

If both sequences are Riesz bases Theorem 1.1.35 tells us that $C_{g_k \otimes \overline{f}_j} = M^{(f_k,g_j)}$ is bijective and therefore $(f_k \otimes \overline{g}_j)$ is a Riesz Basis.

$$\left\langle f_{k_1} \otimes \overline{g}_{j_1}, \tilde{f}_{k_2} \otimes \overline{\tilde{g}}_{j_2} \right\rangle_{\mathcal{H}S} \overset{Lem. A.4.39}{=} \left\langle f_{k_1}, \tilde{f}_{k_2} \right\rangle_{\mathcal{H}} \cdot \langle g_{j_2}, g_{j_1} \rangle_{\mathcal{H}} = \\ = \delta_{k_1,k_2} \cdot \delta_{j_1,j_2}$$

□

In section 1.3 we will look at operators which can be described in this sense by diagonal matrices. In the Hilbert-Schmidt class we will look at operators spanned by $\gamma_k \otimes \overline{g}_k$. We now already know

1. that this will be a Bessel sequence for Bessel sequences (g_k) and (f_k)
2. and because every sub-family of a Riesz basis is a Riesz sequence, that for Riesz bases, $(\gamma_k \otimes \overline{g}_k)$ is a Riesz sequence.

1.2.3.3 Matrices And The Kernel Theorems

For $L^2(\mathbb{R}^d)$ the $\mathcal{H}S$ operators are exactly those integral operators with kernels in $\mathbb{R}^{2d}$, see [110] [43]. This means that such an operator can be described as

$$(Of)(x) = \int \kappa_O(x,y) f(y) dy$$

Or in weak formulation

$$\langle Of, g \rangle = \int \int \kappa_O(x,y) f(y) \overline{g}(x) dy dx = \langle \kappa_O, f \otimes \overline{g} \rangle \tag{1.2}$$

which can be used for other kernel theorem for Banach spaces and distributions, see below.

From 1.1.22 we know that

$$O = \sum_{k,i} \left\langle O\tilde{g}_j, \tilde{f}_k \right\rangle f_k \otimes \overline{g}_j$$

and so

Corollary 1.2.24 *Let $O \in \mathcal{HS}\left(L^2\left(\mathbb{R}^d\right)\right)$. Let (g_j) and (f_k) be frames for $L^2\left(\mathbb{R}^d\right)$. Then the kernel of O is*

$$\kappa_O = \sum_{j,k} \left\langle O\tilde{g}_j, \tilde{f}_k \right\rangle \cdot f_k \otimes \overline{g}_j$$

Proof:

$$\kappa(O) = \kappa\left(\sum_{k,i} \left\langle O\tilde{g}_j, \tilde{f}_k \right\rangle f_k \otimes \overline{g}_j\right) =$$

$$= \sum_{k,i} \left\langle O\tilde{g}_j, \tilde{f}_k \right\rangle \kappa\left(f_k \otimes \overline{g}_j\right) \overset{Lemma A.4.42}{=} \sum_{k,i} \left\langle O\tilde{g}_j, \tilde{f}_k \right\rangle f_k \otimes \overline{g}_j$$

□

There is a large variety of function spaces, where operators are exactly integral operator using equation 1.2, for example for bounded operators $L^2(\mathbb{R}^d)$ and $\mathcal{HS}$ operators, for the Schwartz space $O : \mathcal{S} \to \mathcal{S}'$, the modulation spaces $O : M_v^1(\mathbb{R}^d) \to M_{1/v}^\infty$, Feichtinger's algebra $O : S_0' \to S_0$ and $O : S_0 \to S_0'$. See [110] [63] [43]. In order to derive similar results for the case of Banach spaces of functions or distirbutions Section 1.1.7.3 would have to be generalized to these spaces.

1.2.3.4 The $\mathcal{HS}$ Inner Product Algorithm

Let us return to the finite dimensional space $\mathbb{C}^n$. As seen in the section about matrix representation in $\mathcal{HS}$, the inner product $\langle T, g_k \otimes f_l\rangle_{\mathcal{HS}}$ becomes important. The diagonal version $\langle T, g_k \otimes f_k\rangle_{\mathcal{HS}}$ will play an essential role in the next section about frame multipliers. There are several ways to calculate this $\mathcal{HS}$ inner product, which we will list in Theorem 1.2.28. We will first collect the following properties for the proof of this theorem.

Note that with $\lfloor x \rfloor$ we describe the biggest integer smaller than x.

Lemma 1.2.25 *Let $\mathcal{M}_{m,n}$ be the vector space of $m \times n$-matrices with the inner product, cf. Section A.3.5.1,*

$$\langle A, B\rangle_{fro} = \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} A_{i,j} \overline{B}_{i,j}.$$

For $M \in \mathcal{M}_{m,n}$ let

$$\mathfrak{vec}^{(n)}(M)_k = M_{k \bmod n, \lfloor \frac{k}{n} \rfloor} \ for\ k = 0, \dots, m \cdot n - 1$$

With this function this space is isomorphic to $\mathbb{C}^{m\times n}$ with the standard inner product. The inverse of this function is

$$\mathfrak{Mat}_{\mathfrak{n}}(x)_{i,j} = x_{i+j\cdot n}$$

Proof: The function $\mathfrak{vec}^{(n)}$ is clearly linear and inverse to $\mathfrak{Mat}_{\mathfrak{n}}$. □

The function $\mathfrak{vec}^{(n)}$ joins the columns together to a vector. The function $x \mapsto \mathfrak{Mat}_{\mathfrak{n}}(x)$ separates this vector again. $\mathfrak{Mat}_{\mathfrak{n}}(x)$ is well known in signal processing, it is called *Polyphase representation* there.

We will denote the complexity of a formula by $\sim$, so by $f \sim n$, we mean that the complexity of the calculation of f needs (approximately) n operations.

Lemma 1.2.26 *The complexity of the calculation of the following terms is*

1. *inner product: Let $x, y \in \mathbb{C}^p$, then*

$$\langle x, y\rangle \sim 2p - 1$$

2. *matrix-vector multiplication: Let $A \in \mathcal{M}_{p,q}$, $x \in \mathbb{C}^q$, then*

$$A \cdot x \sim p \cdot (2q - 1)$$

3. *matrix-matrix multiplication: Let $A \in \mathcal{M}_{p,q}$, $B \in \mathcal{M}_{q,r}$, then*

$$A \cdot B \sim p \cdot r \cdot (2q - 1)$$

4. *Kronecker product of matrices: Let $A \in \mathcal{M}_{p,q}$, $B \in \mathcal{M}_{r,s}$, then*

$$A \otimes B \sim p \cdot q \cdot r \cdot s$$

Proof: Use the definitions:

$$\langle x, y\rangle = \sum_{k=0}^{p-1} x_i \overline{y}_i$$

$$(A \cdot x)_i = \sum_{j=0}^{q-1} A_{i,j} x_j \; for \; i = 0, \ldots, p$$

$$(A \cdot B)_{i,k} = \sum_{j=0}^{q-1} A_{i,j} B_{j,k} \textit{ for } i = 0, \ldots, p \textit{ and } k = 0, \ldots, r$$

$$(A \otimes B)_{i,k} = A_{\lfloor \frac{i}{r} \rfloor, \lfloor \frac{i}{s} \rfloor} \cdot B_{i \bmod r, k \bmod s}$$

$$\textit{for } i = 0, \ldots, rp-1 \textit{ and } k = 0, \ldots, qs-1$$

□

Lemma 1.2.27 *Let $A \in \mathcal{M}_{r,s}$, $B \in \mathcal{M}_{p,q}$ and $C \in \mathcal{M}_{q,r}$. Then*

$$\left(A^T \otimes B\right) \cdot \left(\mathfrak{vec}^{(q)} C\right) = \mathfrak{vec}^{(p)} \left(A \cdot C \cdot \vec{B}\right)$$

Proof:

$$\left(\left(A^T \otimes B\right) \left(\mathfrak{vec}^{(q)} C\right)\right)_i = \sum_{j=0}^{q \cdot s-1} \left(A^T \otimes B\right)_{i,j} \left(\mathfrak{vec}^{(q)} C\right)_j =$$

$$\sum_{j=0}^{q \cdot s-1} A^T_{\lfloor \frac{i}{p} \rfloor, \lfloor \frac{j}{q} \rfloor} \cdot B_{i \bmod p, j \bmod q} C_{j \bmod q, \lfloor \frac{j}{q} \rfloor} = (*)$$

Let $j_1 = j \bmod q$ and $j_2 = \left\lfloor \frac{j}{q} \right\rfloor$, so

$$(*) = \sum_{j_1=0}^{q-1} \sum_{j_2=0}^{s-1} A^T_{\lfloor \frac{i}{p} \rfloor, j_2} \cdot B_{i \bmod p, j_1} C_{j_1, j_2} =$$

$$= \sum_{j_2=0}^{s-1} A_{j_2, \lfloor \frac{i}{p} \rfloor} \cdot (B \cdot C)_{i \bmod p, j_2} =$$

$$= (B \cdot C \cdot A)_{i \bmod p, \lfloor \frac{i}{p} \rfloor}$$

□

Theorem 1.2.28 *Let $(h_l)_{l=0}^L$ be a frame in $\mathbb{C}^n$, $(g_k)_{l=0}^K$ in $\mathbb{C}^m$. Let T be a linear operator $T : \mathbb{C}^n \to \mathbb{C}^m$. Then*

1. $\left\langle T, g_k \otimes \overline{h}_l \right\rangle_{\mathcal{HS}} = \left\langle \mathfrak{vec}^{(n)}(T), \mathfrak{vec}^{(n)}\left(g_k \otimes \overline{f}_l\right)\right\rangle_{\mathbb{C}^{m \cdot n}} \sim (3mn + m - 1)$ *for each single pair* (l, k).

2. $\langle T, g_k \otimes h_l \rangle_{\mathcal{HS}} = \langle Th_l, g_k \rangle_{\mathbb{C}^m} \sim (2mn + m - 1)$ *for each single pair* (l, k).

3. $\left\langle T, g_k \otimes \overline{h}_l \right\rangle_{\mathcal{HS}} = \left(C_{g_k} \cdot T \cdot D_{h_l} \right)_{l,k} \sim \left(L \left(2mn - m + 2mK - K \right) \right)$ *for all values* (l, k).

4. $\left\langle T, g_k \otimes h_l \right\rangle_{\mathcal{HS}} = \left(D_{g_k}^T \otimes C_{f_l} \right) \vec{T} \sim \left(KL \cdot (3mn - 1) \right))$ *for all values* (l, k).

Proof: We will use Lemma 1.2.26 extensively:

1.) Calculation of $g_k \otimes h_l \sim m \cdot n$. $\mathfrak{vec}^{(n)}$ is only a reordering. The complex conjugation $\sim m$. Calculation of inner product $\sim 2 \cdot (mn) - 1$. So the sum is $3mn + m - 1$.

2.) Calculation of $Th_l \sim m(2n - 1)$. Calculation of the inner product $\sim 2m - 1$. The sum is $2mn + m - 1$.

3.) That $\left\langle T, g_k \otimes h_l \right\rangle_{\mathcal{HS}} = \left(C_{g_l} \cdot T \cdot D_{h_l} \right)_{l,k}$ can be seen using Lemma A.3.7. $T \cdot D \sim mL(2n - 1)$, $C \cdot (TD) \sim KL(2m - 1)$, so altogether we get $mL2n - mL + KL2m - KL = L \left(2mn - m + 2mK - K \right)$.

4.) Using Lemma 1.2.27 we know that this equality is true. For the calculation of $D_{g_k}^T \otimes C_{f_l}$ we need $\sim KmLn$. And for the matrix vector multiplication in $\mathbb{C}^{mn} \sim KL(2mn - 1)$. So overall $KL(2mn - 1) + KmLn = KL \cdot (3mn - 1)$. □

So overall if we have to calculate the inner products for all pairs (k, l) the third method is the fastest (except when n is very big and m and K very small). If we need only the diagonal part $k = l$, the second one is the most efficient as for using the third method we would still have to calculate the whole matrix and then use its trace.

1.3 Frame Multiplier

1.3.1 Basic Definition

R. Schatten provides a detailed study of ideals of compact operators in [110] using their singular value decomposition. He investigates the operators of the form $\sum \lambda_i \varphi_i \otimes \overline{\psi_i}$ where (φ_i) and (ψ_i) are orthonormal families. We are interested in similar operators where the only difference is that the families are frames or Bessel sequences. See Section A.4.4.1 for basic properties of the rank one operators $f \otimes \overline{g}$.

Analogous to the definition of Gabor multiplier, e.g. found in [47] , we define:

Definition 1.3.1 *Let* $\mathcal{H}_1$, $\mathcal{H}_2$ *be Hilbert-spaces, let* $\mathcal{G} = \left\{ g_k \right\}_{k \in K}$ *be a frame in* $\mathcal{H}_1$, $\mathcal{F} = \left\{ f_k \right\}_{k \in K}$ *in* $\mathcal{H}_2$. *Define the operator* $\mathbf{M}_{m,\mathcal{F},G} : \mathcal{H}_1 \to \mathcal{H}_2$, *the*

frame multiplier *for the frames* $\{g_k\}$ *and* $\{f_k\}$, *as the operator*

$$\mathbf{M}_{m,\mathcal{F},\mathcal{G}} = \sum_k m_k \langle f, g_k \rangle f_k$$

where $m \in l^\infty(K)$.

For $v \in l^2$ let $M = \mathfrak{diag}(v)$ be the diagonal matrix, for which $M_{i,j} = \delta_{i,j} v_i$.

Using Theorem 1.1.22 we get

Corollary 1.3.1 *With the conventions in Definition 1.3.1*

$$\boldsymbol{M}_{m,\mathcal{G},\mathcal{G}'} = \mathcal{O}^{(f_k,g_j)}\left(\mathfrak{diag}(m)\right)$$

To be able to talk about diagonal matrices we use the condition in Definition 1.3.1 that the two frames must have the same index set.

We will often use the alternate notation $\mathbf{M}_{m,f_k,g_k}$. Let $\mathbf{M}_{m,g_k} = \mathbf{M}_{m,g_k,g_k}$. We will simplify this notation, if there is no chance of confusion, using $\mathbf{M}_m$ or even $\mathbf{M}$. Also the following notation is obviously equivalent:

Corollary 1.3.2

$$\mathbf{M}_{m,f_k,g_k} = D_{f_k}(m \cdot C_{g_k}) = \sum_k m_k \cdot f_k \otimes g_k$$

The frame multiplier is a linear combination of rank one (or zero) operators. If e.g. m_k is non-zero for only finitely many indices, then $\mathbf{M}$ has finite rank. (So it can be shown that the multiplier is compact for $m \in c_0$, see Section 1.3.5 .)

The multiplier is well defined, which will be shown in Section 1.3.5.

The term "multiplier" was used corresponding to Gabor or STFT multipliers [47]. This is not equivalent to the definition of multipliers e.g. found in [80]: A is a multiplier if and only if $AT_t = T_t A$ for the translation operator T_t and all t. For frame multipliers no connection to the shift operator can be made.

We will see that the class of frame multipliers is quite "big". All compact operators can be written as frame multipliers. Therefore it is important to investigate properties of this kind of operators. Especially interesting is the dependence on the properties of the symbol, see Section 1.3.5.1 or the frame, see e.g. Section 1.3.7.

More general we can define such an operator for a Bessel sequence:

Definition 1.3.2 *Let $\mathcal{H}$ be a Hilbert-space, let (g_k) be Bessel sequences in $\mathcal{H}_1$ and and (f_k) in $\mathcal{H}_2$, define the operator $\mathbf{M}_{m,f_k,g_k} : \mathcal{H}_1 \to \mathcal{H}_2$, the* **Bessel multiplier** *for (f_k) and (g_k), as the operator*

$$\mathbf{M}_{m,f_k,g_k}(f) = \sum_k m_k \langle f, g_k \rangle f_k$$

Clearly every frame multiplier is a Bessel multiplier. Since $m \in l^\infty$ we know that for $c \in l^2$ also $m \cdot c \in l^2$. As we use Bessel sequences we therefore know that the convergence in Definition 1.3.2 is unconditional.

Definition 1.3.3 *Let $\sigma_U(\mathbf{M}) = m$ in Definition 1.3.1. Then m is called the (upper)* **symbol** *of $\mathbf{M}$.*

This relation σ_U does not have to be a well-defined function. This is only the case if the operators $g'_k \otimes g_k$ have a basis property, cf. 1.3.7.1.

1.3.2 The Multiplier From $l^2 \to l^2$

On l^2 a pointwise multiplication, $(a \cdot c)_i = a_i \cdot c_i$, can be defined, which is certainly associative and commutative. As $l^\infty \cdot l^2 \subseteq l^2$ it is clear that this multiplication is well defined for $a \in l^p$. For $a \in l^2$ this multiplication defines an inner operation. The problem from an group-theoretical point of view is that the unit element would be the constant sequence 1, which is in l^∞ but not in any l^p for $1 \leq p < \infty$. There are certainly no inverse elements in c for elements in c_0, because if the sequence c converges to zero, its inverse sequence must tend to infinity. The space l^2 with this product is a commutative semi-group. The set of possible multipliers is the set of all polynomials of order 1, i.e. $\{ax\}$, on this commutative semi-group. For more information about polynomials over groups and semi-groups see [7]. Only if we restrict this operation to $c_N = \{c \in l^\infty \,|c_k = 0 \forall k > N\}$ for a $N \in \mathbb{N}$, we will have a group structure.

We will use the symbol $\mathcal{M}_m$ for the mapping $\mathcal{M}_m : l^2 \to l^2$ and $m \in l^p$ (for a $p > 0$) given by the pointwise multiplication $\mathcal{M}_m(\{c_k\}) = \{m_k \cdot c_k\}$. So

$$\mathbf{M}_m = D \circ \mathcal{M}_m \circ C$$

As preparation for Theorem 1.3.13 we can show that the class of operators depends on the space containing m. Let δ_i be the standard ONB in l^2, $(\delta_i)_j = \delta_{i,j}$.

Lemma 1.3.3 *1. Let $m \in l^\infty$. The operator $\mathcal{M}_m : l^2 \to l^2$ is bounded with $\|\mathcal{M}_m\|_{Op} = \|m\|_\infty$.*

2. $\mathcal{M}_m^* = \mathcal{M}_{\overline{m}}$

3. *Let $m \in l^1$. The operator $\mathcal{M}_m : l^2 \to l^2$ is trace class with $\|\mathcal{M}_m\|_{trace} = \|m\|_1$.*

4. *Let $m \in l^2$. The operator $\mathcal{M}_m : l^2 \to l^2$ is a Hilbert-Schmidt ($\mathcal{HS}$) operator with $\|\mathcal{M}_m\|_{\mathcal{HS}} = \|m\|_2$.*

5. *Let $m \in c_0$. Then there exist finite sequences $m_N = (m_0, \ldots, m_N, 0, \ldots)$ with $\mathcal{M}_{m_N} \to \mathcal{M}_m$ for all l^p norms. Therefore $\mathcal{M}_m$ is compact.*

Proof: 1.) We already know that

$$\|m \cdot c\|_2 \leq \|m\|_\infty \|c\|_2 .$$

On the other hand $\mathcal{M}_m \delta_i = m_i \Longrightarrow \|\mathcal{M}_m\|_{Op} \geq \|m\|_\infty$.

2.)

$$\langle \mathcal{M}_m c, d \rangle_{l^2} = \sum_k m_k c_k \cdot \overline{d}_k = \sum_k c_k \cdot \overline{m}_k \overline{d}_k = \langle c, \mathcal{M}_{\overline{m}} d \rangle_{l^2}$$

3.) $[\mathcal{M}_m] = \sqrt{\mathcal{M}_m^* \mathcal{M}_m} = \sqrt{\mathcal{M}_{\overline{m}} \mathcal{M}_m} = \mathcal{M}_{|m|}$ and so

$$\|\mathcal{M}_m\|_{trace} = \sum_i \langle [\mathcal{M}_m] \delta_i, \delta_i \rangle = \|m\|_1 .$$

4.)

$$\|\mathcal{M}_m\|_{\mathcal{HS}}^2 = \sum_i \|\mathcal{M}_m \delta_i\|_2 = \|m\|_2^2 .$$

5.) Let $c \in l^p$, then $\lim\limits_{N \to \infty} m_N \cdot c = m \cdot c$ as $\|m_N \cdot c - m \cdot c\|_p \leq \|m_N - m\|_\infty \cdot \|c\|_p$.

□

Compare this result also to multipliers for $L^2\left(\mathbb{R}^d\right)$, see Section A.4.3.5.

To stress the connection of multipliers to diagonal matrices, we could also prove the third part of this lemma by using Proposition 1.2.22. The operator M_m corresponds to the diagonal matrix with diagonal entries m_i, so this proposition gives us the wanted property.

1.3.2.1 Surjective And Injective Multipliers

Lemma 1.3.4 *Let $\mathcal{M}_m$ be the multiplier with a fixed element $m \in l^p$ with $p \geq 1$. Then $\mathcal{M}_m$ is not surjective from $l^2 \to l^2$.*

Proof: Suppose $\mathcal{M}_m : l^2 \to l^2$ is surjective. This means that $\forall c \in l^2 \; \exists b \in l^2 : a \cdot b = c \Longleftrightarrow a_i \cdot b_i = c_i$.

1.) Suppose there is an index i_0 such that $a_{i_0} = 0$. Let c be such that

$$c_i = \begin{cases} 1 & i = i_0 \\ a_i & otherwise \end{cases}$$

Then there is no b such that $a \cdot b = c$. So we have arrived at a contradiction.

2.) Suppose all entries $a_i \neq 0$, then there is no b such that $a \cdot b = a$. If this were the case then $a_i \cdot b_i = a_i$ and so $b_i = 1$. But then $\|b\|_p^p = \sum_i 1^p \not< \infty$. $\square$

For injectivity we can even give an equivalence property

Lemma 1.3.5 *Let $(a_i) \in l^\infty$. If and only if $a_i \neq 0$ then $m_a : l^2 \to l^2$ is injective.*

Proof: Let $m_a(b) = m_a(c)$ so $a \cdot b = a \cdot c$. This means that $a_i \cdot b_i = a_i \cdot c_i$ for all i.

If $a_i \neq 0 \Longrightarrow b_i = c_i$ and so if for all i $a_i \neq 0$ $c = d$.

On the other hand if there is an i_0 where $a_{i_0} = 0$ then for any c and $c' = c + \delta_{i_0}$ we have that $m_a(c) = m_a(c')$ and $c \neq c'$. $\square$

1.3.3 The Multiplier For An ONB

To be able to compare multipliers for ONBs to the results in this work, we will repeat the findings of [110]. Let in this sector (φ_j) and (ψ_j) be orthonormal sequences for the Hilbert space $\mathcal{H}$. Remember that $[T] = (TT^*)^{\frac{1}{2}}$ as in Definition A.4.22.

Theorem 1.3.6 ([110] I.1 Theorem 1) *Let (λ_i) be a sequence. Let $\mathbf{M}_\lambda = \sum_j \lambda_j \varphi_j \otimes \overline{\psi}_j$. Then this is well-defined and bounded if and only if $\lambda \in l^\infty$. And $\|\mathbf{M}_\lambda\|_{Op} = \|\lambda\|_\infty$,* We will show one direction for all Bessel sequences in Section 1.3.5. We can not show an equality for the norm, but an inequality using the Riesz bounds for Riesz bases, see Proposition 1.3.20.

Corollary 1.3.7 ([110] I.1 Corollary) *Let* $\lambda \in l^\infty$. $\mathbf{M}_\lambda = \sum_j \lambda_j \varphi_j \otimes \overline{\psi}_j = 0$ *if and only if* $\lambda_i = 0$ *for all* i.

We will look into the question, whether the connection symbol to operator, $m \mapsto \mathbf{M}_m$, is injective, in the following Section 1.3.7.1.

Theorem 1.3.8 ([110] I.1 Theorem 2) *Let* $\lambda \in l^\infty$. *Let* $\mathbf{M}_\lambda = \sum_j \lambda_j \varphi_j \otimes \overline{\psi}_j$.

1. $\mathbf{M}_\lambda^* = \sum_j \overline{\lambda}_j \psi_j \otimes \overline{\varphi}_j$
2. $\mathbf{M}_\lambda^* \mathbf{M}_\lambda = \sum_j |\lambda_j|^2 \psi_j \otimes \overline{\psi}_j$
3. $[\mathbf{M}_\lambda] = \sum_j |\lambda_j| \varphi_j \otimes \overline{\psi}_j$
4. *The operator* $\sum_j \lambda_j \ \varphi_j \otimes \varphi_j$ *is normal.*
5. *It is self-adjoint if and only if the* λ_i *are real.*

The first and fifth statements are true for all Bessel sequences, see Section 1.3.5. For the rest the orthonormality of the sequences is important.

Theorem 1.3.9 ([110] I.1 Theorem 2) *An operator* $O : \mathcal{H} \to \mathcal{H}$ *is*

1. *a projection if and only if* $O = \mathbf{M}_\lambda = \sum_j \varphi_j \otimes \overline{\varphi}_j$
2. *isometric if and only if* $O = \mathbf{M}_\lambda = \sum_j \varphi_j \otimes \overline{\psi}_j$ *and* (ψ_j) *are complete.*
3. *unitary if and only if* $O = \mathbf{M}_\lambda = \sum_j \varphi_j \otimes \overline{\psi}_j$ *and* (φ_j), (ψ_j) *are complete.*

All these properties need the orthonormality condition.

Theorem 1.3.10 ([110] I.2 Theorem 4) *Let* $\lambda \in l^\infty$. *The operator* $\mathbf{M}_\lambda = \sum_j \lambda_j \varphi_j \otimes \overline{\psi}_j$ *has an inverse if and only if* (φ_j) *and* (ψ_j) *are complete and* λ *is semi-normalized. Then*

$$\mathbf{M}_{\lambda_j, \varphi_j, \psi_j} = \mathbf{M}_{\frac{1}{\lambda_i}, \psi_i, \varphi_i}$$

Something similar can be shown for Riesz bases using the dual bases, see Proposition 1.3.28.

Corollary 1.3.11 ([110] I.2 Corollary) *Let* $\lambda \in l^\infty$. *Let* $\mathbf{M}_\lambda = \sum\limits_j \lambda_j \varphi_j \otimes \overline{\psi}_j$. *If and only if* $\lambda \in c_0$ *and* λ *is real then* $\mathbf{M}_\lambda$ *is compact and self-adjoint. In this case the spectrum of* $\mathbf{M}_\lambda$ *is* λ *and possibly* 0.

We will show the first part of the corollary for Bessel sequences in Theorem 1.3.13.

Overall it is clear from the remarks we made, that the handling of the frame multipliers becomes more difficult than in the case of orthonormal sequences. But we have also seen, that in the case of Riesz bases a lot of these properties stay true. We will devote the whole Section 1.3.7 to multipliers with Riesz bases.

1.3.4 Combination Of Multipliers

How do two multipliers combine? We have chosen that as one of the first questions to answer, because we will need this formula very often and it becomes evident that the situation with frames or Bessel sequences is not nearly as "smooth" as with orthonormal sequences.

Lemma 1.3.12 *For two multipliers for the sequences* (g_k),(f_k),(g'_k) *and* (f'_k)

$$\mathbf{M}_{m^{(1)},f_k,g_k} = \sum_k m_k^{(1)} \langle f, g_k \rangle f_k$$

and

$$\mathbf{M}_{m^{(2)},f_k,g_k} = \sum_l m_l^{(2)} \langle f, g'_l \rangle f'_l$$

the combination is

$$\left(\mathbf{M}_{m^{(1)},f_k,g_k} \circ \mathbf{M}_{m^{(2)},f'_k,g'_k}\right)(f) = \sum_k \sum_l m_k^{(1)} m_l^{(2)} \langle f, g'_l \rangle \langle f'_l, g_k \rangle f_k =$$

$$= D_{f_k} \mathcal{M}_{m^{(1)}} G_{g_k,f'_k} \mathcal{M}_{m^{(2)}} C_{g'_k}$$

Proof:

$$\left(\mathbf{M}_{m^{(1)},f_k,g_k} \circ \mathbf{M}_{m^{(1)},f'_k,g'_k}\right)(f) = \mathbf{M}_{m^{(1)},f_k,g_k}\left(\sum_l m_l^{(2)} \langle f, g'_l \rangle f'_l\right) =$$

$$= \sum_k m_k^{(1)} \left\langle \sum_l m_l^{(2)} \langle f, g_l' \rangle f_l', g_k \right\rangle f_k = \sum_k \sum_l m_k^{(1)} m_l^{(2)} \langle f, g_l \rangle \langle f_l, g_k \rangle f_k$$

Using the Gram matrix this can be written as

$$\mathbf{M}_{m^{(1)}, f_k, g_k} \circ \mathbf{M}_{m^{(2)}, f_k', g_k'} = D_{f_k} \mathcal{M}_{m^{(1)}} C_{g_k} D_{f_k'} \mathcal{M}_{m^{(2)}} C_{g_k'} =$$

$$= D_{f_k} \mathcal{M}_{m^{(1)}} G_{g_k, f_k'} \mathcal{M}_{m^{(2)}} C_{g_k'}$$

□

Thus in the general frame case no exact symbolic calculus can be assumed, i.e. the combination of symbols does not correspond to the combination of the operators. Even in the case of using only one Bessel sequence we get

$$\mathbf{M}_{m^{(1)}, g_k} \circ \mathbf{M}_{m^{(2)}, g_k} \neq \mathbf{M}_{m^{(1)} \cdot m^{(2)}, g_k}.$$

In general the product of two frame multipliers is not a frame multiplier any more. It is not induced by a diagonal matrix anymore, following Section 1.1.7.3, but rather

$$\left(\mathbf{M}_{m^{(1)}, f_k, g_k} \circ \mathbf{M}_{m^{(2)}, f_k, g_k} \right) = \mathcal{O}^{(f_i, g_j)} \left(\left(m_i^{(1)} \cdot \left(G_{g_i, f_j} \right)_{i,j} m_j^{(2)} \right)_{i,j} \right)$$

1.3.5 Properties Of Multipliers

1.3.5.1 The Connection Of Properties Of Symbol And Multiplier

Now equivalent results proved in [47] for Gabor multiplier can be shown for Bessel multipliers.

Theorem 1.3.13 *Let $\mathbf{M} = \mathbf{M}_{m, f_k, g_k}$ be a Bessel multiplier for the Bessel sequences $\{g_k\}$ and $\{f_k\}$ with the Bessel bounds B and B'. Then*

1. *If $m \in l^\infty$, i.e. m is bounded, $\mathbf{M}$ is a well defined bounded operator. $\|\mathbf{M}\|_{Op} \leq \sqrt{B'}\sqrt{B} \cdot \|m\|_\infty$.*

2. *$\mathbf{M}^*_{m, f_k, g_k} = \mathbf{M}_{\overline{m}, g_k, f_k}$. Therefore if m is real-valued and $f_k = g_k$, $\mathbf{M}$ is self-adjoint.*

3. *If $m \in c_0$, $\mathbf{M}$ is compact.*

4. *If $m \in l^1$, $\mathbf{M}$ is a trace class operator with $\|M\|_{trace} \leq \sqrt{B'}\sqrt{B}\ \|m\|_1$. And $tr(M) = \sum_k m_k \langle f_k, g_k \rangle$.*

5. *If $m \in l^2$, $\mathbf{M}$ is a Hilbert Schmidt operator with $\|M\|_{\mathcal{HS}} \le \sqrt{B'}\sqrt{B}\ \|m\|_2$.*

Proof: We will use the notation $D = D_{f_k}$, $C = C_{g_k}$ with $\|D\|_{Op} \le \sqrt{B'}$ and $\|C\|_{Op} \le \sqrt{B}$. From 1.1.1 we know that the members of a frame are norm bounded, so $\|g_k\|_{\mathcal{H}} \le \sqrt{B}$ and $\|g'_k\|_{\mathcal{H}} \le \sqrt{B'}$.

1.) This follows directly from the definition of the multiplier as $C \circ M_m \circ D$ and Lemma 1.3.3.

$$\|\mathbf{M}\| = \|C \circ \mathcal{M}_m \circ D\| \le \|C\| \cdot \|m\|_\infty \cdot \|D\| \le \sqrt{B}\, \|m\|_\infty \sqrt{B'}$$

2.) $\mathbf{M} = C_{g_k} \circ M_m \circ D_{f_k} = C_{g_k} \circ M_m \circ C_{f_k}^*$, so $\mathbf{M}_m^* = C_{f_k} \circ M_m^* \circ C_{g_k}^*$. From Lemma 1.3.3 we know $\mathcal{M}_m^* = \mathcal{M}_{\overline{m}}$. If $m_k \in \mathbb{R}$ $M_m^* = M_m$, so $\mathbf{M}^* = \mathbf{M}$.

3.) Let m_N be the finite sequences from Lemma 1.3.3, then for every $\epsilon > 0$ there is an N such that

$$\|\mathbf{M}_{m_N} - \mathbf{M}_m\|_{Op} = \|D\mathcal{M}_{m_N}C - D\mathcal{M}_mC\|_{Op} = \|D\left(\mathcal{M}_{m_N} - \mathcal{M}_m\right)C\|_{Op} \le$$

$$\le \|D\|_{Op} \|\mathcal{M}_{m_N} - \mathcal{M}_m\|_{Op} \|C\|_{Op} \le \sqrt{B'} \cdot \epsilon\sqrt{B}$$

$\mathbf{M}_{m_N}$ is a finite sum of rank one operators and so has finite rank. This means that $\mathbf{M}_m$ is a limit of finite-rank operators and with Corollary A.4.27 therefore compact.

4.)

$$\mathbf{M}(f) = \sum_k \langle f, g_k \rangle (m_k \cdot g'_k)$$

so according to the definition of trace class operators A.4.23 we just have to show that

$$\sum_k \|g_k\|_{\mathcal{H}} \cdot \|m_k g'_k\|_{\mathcal{H}} < \infty$$

$$\sum_k \|g_k\|_{\mathcal{H}} \cdot \|m_k g'_k\|_{\mathcal{H}} = \sum_k \|g_k\|_{\mathcal{H}} |m_k| \|g'_k\|_{\mathcal{H}} \le \sqrt{B} \cdot \sqrt{B'} \cdot \|m\|_1$$

Due to A.4.36

$$tr(M) = \sum_k \langle m_k \cdot g'_k, g_k \rangle = \sum_k m_k \langle g'_k, g_k \rangle$$

5.) The operator $\mathcal{M}_m : l^2 \to l^2$ is in $\mathcal{HS}$ due to Lemma 1.3.3 with bound $\le \|m\|_2$. The $\mathcal{HS}$ operators combined with other linear operators stay in $\mathcal{HS}$, refer to Lemma A.4.38. Therefore

$$\|D_{f_k}\mathcal{M}_m C_{g_k}\|_{\mathcal{HS}} \le \|D_{f_k}\|_{Op} \|m\|_2 \|C_{g_k}\|_{Op} \le \sqrt{B}\sqrt{B'}\, \|m\|_2$$

□

Remark:

1. Property (3) could be shown by using (1), as we know that

$$\|m - m_N\|_\infty \to 0 \text{ for } N \to \infty.$$

2. Property (4) could be very easily shown using the property for the multipliers on l^2, see Lemma 1.3.3, and the fact, that trace class operators act as ideal. But this proof uses the special form of the operators and the basic definition for trace class operators. For this proof the Bessel bounds are only needed as upper bound for the norms of the frame elements. So instead of the Bessel bound any such upper bound can be used.

3. To reach an equality for the norm inequalities above is in general not possible. In this case the map from the symbol to the multiplier is injective. We will investigate this question in Section 1.3.7.1.

4. Due to this results it is also clear that the sum $\sum m_k f_k \otimes g_k$ converges in the respective norm $\|.\|_{Op}$, $\|.\|_{\mathcal{HS}}$ or $\|.\|_{trace}$.

5. It should be possible to use analogue proofs for continuous frames and some properties. Some thought have to be given to which function spaces are used for the symbol. A natural choice in the case of Gabor frames would be the Modulation spaces, cf. Section 2.1.3. An analogue property does not hold for all such spaces, because in [65] it is shown that the pseudo differential operator is only bounded for symbols in $M^{r,1}$ for $1 \leq r \leq \infty$ and $M^{2,1}$ and $M^{2,2}$. For all other spaces counterexamples could be constructed there.

For Riesz and orthonormal base we can show, see Proposition 1.3.20 and Theorem 1.3.6, that if the multiplier is well-defined, then the symbol must be in l^∞. This is not true for general frames, as can be seen, when using the frame in Example 1.2.1 (2), $g_{k,l} = \frac{1}{l^2} e_k$ and using the symbol $m_{k,l} = l^2$. Then

$$\mathbf{M}_{m_{k,l}, g_{k,l}} = \sum_{k,l} l^2 \left\langle f, \frac{1}{l^2} \cdot e_k \right\rangle \frac{1}{l^2} \cdot e_k = \sum_{k,l} \left\langle f, \frac{1}{l} \cdot e_k \right\rangle \frac{1}{l} \cdot e_k = S_{h_{kl}}$$

where $h_{kl} = \frac{1}{l} e_k$ from Example 1.2.1 (1).

1.3.5.2 Frame Multiplier And Compact Operators

The above mentioned operator classes can be described with Bessel sequences

Lemma 1.3.14 *An operator T is compact (nuclear respectively Hilbert Schmidt) if and only if there exist two frames (g_k) (g'_k) such that $T = \sum_k m_k g_k \otimes g'_k$ with $(m_k) \in c_o$ (l^1 resp. l^2).*

Proof: See [110] I.4 Theorem 7, II.1 Theorem 3 and III.1 Theorem 5, where it is proved that every such operator can be written in this way for two ONBs, the converse follows from Theorem 1.3.13. □

For a classification it is better to use ONBs as they form the smaller class of operators.

That does not work for a fixed pair of frames, as a multiplier for one ONB does not have to be one for another one. For examples look at diagonal matrices and different ONBs in $\mathbb{R}^2$.

1.3.5.3 Frame Multipliers In $L^2(\mathbb{R}^d)$

Lemma 1.3.15 *Let (g_k) and (γ_k) be frames in $L^2(\mathbb{R}^d)$. Let $m \in l^2$, then the kernel of $\mathbf{M}_{m,g_k,\gamma_k}$ is*

$$\kappa\left(\mathbf{M}_{m,g_k,\gamma_k}\right) = \sum m_k g_k \otimes \overline{\gamma}_k$$

Proof: If $m \in l^2$, we know that $\mathbf{M}$ is in $\mathcal{HS}$ and following Theorem A.4.41 we know that this operator is an integral operator.

$$\mathbf{M}(f)(x) = \sum_k m_k \langle f, \gamma_k \rangle g_k(x) = \sum_k m_k \int_{\mathbb{R}^d} f(y)\overline{\gamma}_k(y) dy g_k(x) =$$

$$= \int_{\mathbb{R}^d} \left(\sum_k m_k \overline{\gamma}_k(y) g_k(x) \right) f(y) dy$$

□

This result can of course be extended to other spaces, where such a kernel representation is possible, e.g. modulation spaces [63].

1.3.5.4 Positive Multipliers

An easily proved statement is

Proposition 1.3.16 *Let $m \in l^\infty$ with $m_k > 0$. Then $\mathbf{M}_{m,g_k}$ is positive.*

Proof:

$$\langle \mathbf{M}f, f \rangle = \sum_k m_k \langle f, g_k \rangle \langle g_k, f \rangle = \sum_k m_k |\langle f, g_k \rangle|^2 > 0$$

□

The condition that the coefficients of the symbol have to be positive is necessary for the general case, as can be seen for an ONB. If one coefficient is zero or negative, the inner product can be zero or negative. So there is no other symbol such that the multiplier is positive for all frames.

We can now use Lemma 1.1.57 for frame multiplier:

Proposition 1.3.17 *Let $\{g_k\}$ be a frame for $\mathcal{H}$. Let $\{m_k\}$ be a semi-normalized sequence with bounds a, b. Then the multiplier $\mathbf{M}_{m,g_k}$ is just the frame operator of the frame $\{\sqrt{m_k} g_k\}$ and so is positive, self-adjoint and invertible.*

Proof:

$$\mathbf{M}_{m,g_k} = \sum_k m_k \langle f, g_k \rangle f_k = \sum_k \langle f, \sqrt{m_k} g_k \rangle \sqrt{m_k} g'_k$$

This means $M_{\{g_k\}} = S_{\{\sqrt{m_k} g_k\}}$, as by Lemma 1.1.57 we know that $\{\sqrt{m_k} g_k\}$ is a frame. So the operator is positive, surjective and invertible. □

Clearly the symbol is not in c, so the operator is not compact. For infinite dimensional spaces no compact operator is invertible.

1.3.6 Changing The Ingredients

A frame multiplier clearly depends on the chosen symbol, analysis and synthesis frame. A natural question arises, when we ask, what happens if these items are changed. Are the frame multipliers similar to each other if the symbol or the frames are similar to each other (in the right topology)? The next result answers this question:

Theorem 1.3.18 *Let $\mathbf{M}$ be a multiplier for the Bessel sequences $\{g_k\}$ and $\{f_k\}$. Then the operator $\mathbf{M}$ depends continuously on m, g_k and f_k, in the following sense. Let $(g_k^{(l)})$ and $(f_k^{(l)})$ be sequences indexed by $l \in \mathbb{N}$.*

1. *Let $m^{(l)} \to m$ in (l^∞, l^2, l^1) then*

$$M_{m^{(l)},g_k,f_k} \to M_{m,g_k,f_k} \text{ in } (Op, \mathcal{HS}, tr)$$

.

2. (a) *Let $m \in l^1$ and let the sequences $(g_k^{(l)})$ be Bessel sequences converging uniformly to (g_k), so $\forall \epsilon \exists N$ such that $\left\| g_k^{(l)} - g_k \right\|_{\mathcal{H}} < \epsilon$ for all $l \geq N$ for all k. Then*

$$\left\| M_{m,g_k^{(l)},f_k} - M_{m,g_k,f_k} \right\|_{trace} \to 0$$

   (b) *Let $m \in l^2$ and let the sequences $(g_k^{(l)})$ converge to (g_k) in an l^2 sense, meaning $\forall \epsilon \exists N$ such that $\sum_k \left\| g_k^{(l)} - g_k \right\|_{\mathcal{H}}^2 < \epsilon$ for all $l \geq N$. Then*

$$\left\| M_{m,g_k^{(l)},f_k} - M_{m,g_k,f_k} \right\|_{\mathcal{HS}} \to 0$$

   (c) *Let $m \in l^\infty$ and let the sequences $(g_k^{(l)})$ converge to (g_k) in an l^1 sense, meaning $\forall \epsilon \exists N$ such that $\sum_k \left\| g_k^{(l)} - g_k \right\|_{\mathcal{H}} < \epsilon$ for all $l \geq N$. Then*

$$\left\| M_{m,g_k^{(l)},f_k} - M_{m,g_k,f_k} \right\|_{Op} \to 0$$

3. *For Bessel sequences $(f_k^{(l)})$ converging to (f_k), corresponding properties like in 2. apply.*

4. (a) *Let $m^{(l)} \to m$ in l^1 and let the sequences $(g_k^{(l)})$ respectively $(f_k^{(l)})$ be Bessel sequences with $B^{(l)}$ and $B'^{(l)}$ as Bessel bounds such that there is a common upper frame bound for all $l \geq N_1$, $\mathbf{B}$ and $\mathbf{B}'$, i.e. $B^{(l)} \leq \mathbf{B}$ (resp. $B'^{(l)} \leq \mathbf{B}'$). Let them converge uniformly to (g_k) respectively (f_k), so $\forall \epsilon \exists N$ such that $\left\| g_k^{(l)} - g_k \right\|_{\mathcal{H}} < \epsilon$ for all $l \geq N$ for all k. Then*

$$\left\| M_{m^{(l)},g_k^{(l)},f_k^{(l)}} \to M_{m,g_k,f_k} \right\|_{trace} \to 0$$

   (b) *Let $m^{(l)} \to m$ in l^2 and let the Bessel sequences $(g_k^{(l)})$ respectively $(f_k^{(l)})$ converges to (g_k) respectively (f_k) in an l^2 sense, meaning $\forall \epsilon \exists N$ such that $\sqrt{\sum_k \left\| g_k^{(l)} - g_k \right\|_{\mathcal{H}}^2} < \epsilon$ for all $l \geq N$. Then*

$$\left\| M_{m,g_k^{(l)},f_k} \to M_{m,g_k,f_k} \right\|_{\mathcal{HS}} \to 0$$

(c) Let $m^{(l)} \to m$ in l^∞ and let the Bessel sequences $(g_k^{(l)})$ respectively $(f_k^{(l)})$ converges to (g_k) respectively (f_k) in an l^1 sense, meaning $\forall \epsilon \exists N$ such that $\sum_k \left\| g_k^{(l)} - g_k \right\|_{\mathcal{H}} < \epsilon$ for all $l \geq N$. Then

$$\left\| M_{m,g_k^{(l)},f_k} \to M_{m,g_k,f_k} \right\|_{Op} \to 0$$

Proof:

1.) For a sequence of symbols this is a direct result of 1.3.13 and

$$\left\| \mathbf{M}_{m^{(l)},g_k,f_k} - \mathbf{M}_{m,g_k,f_k} \right\|_{\mathcal{HS}} = \left\| G_{\left(m^{(l)}-m\right),g_k,f_k} \right\|_{\mathcal{HS}} \overset{1.3.13}{\leq} \left\| m^{(l)} - m \right\|_2 \sqrt{BB'}$$

The result for the operator and infinity norm respectively trace and l^1 norms can be proved in an analogue way.

2.) For points (b) and (c) we know from Corollaries 1.1.63 and 1.1.64 that the sequences are Bessel sequences. For all the norms $(Op, \mathcal{HS}, tr)$ $\|g_k \otimes f_k\| = \|g_k\|_{\mathcal{H}} \|f_k\|_{\mathcal{H}}$ and so

$$\left\| \sum m_k g_k^{(l)} \otimes f_k - \sum m_k g_k \otimes f_k \right\| = \left\| \sum m_k \left(g_k^{(l)} - g_k \right) \otimes f_k \right\| \leq$$

$$\leq \sum_k |m_k| \left\| g_k^{(l)} - g_k \right\|_{\mathcal{H}} \sqrt{B'} = (*)$$

$$\text{case a}: (*) \leq \sqrt{B'} \left(\sum_k |m_k| \right) \sup_l \left\{ \left\| g_k^{(l)} - g_k \right\|_{\mathcal{H}} \right\} \leq \sqrt{B'} \|m\|_1 \epsilon$$

$$\text{case b}: (*) \leq \sqrt{B'} \sqrt{\sum_k |m_k|^2} \sqrt{\sum \left\| g_k^{(l)} - g_k \right\|_{\mathcal{H}}^2} \leq \sqrt{B'} \|m\|_2 \epsilon$$

$$\text{case c}: (*) \leq \sqrt{B'} \|m\|_\infty \sum \left\| g_k^{(l)} - g_k \right\|_{\mathcal{H}} \leq \sqrt{B'} \|m\|_\infty \epsilon$$

3.) Use a corresponding argumentation for $f_k^{(l)}$.

4.) For points (b) and (c) Corollary 1.1.62 gives us the condition that there are common Bessel bounds for $l \geq N_1$, $\mathbf{B}$ and $\mathbf{B}'$.

$$\left\| M_{m^{(l)},g_k^{(l)},f_k^{(l)}} - M_{m,g_k,f_k} \right\| \leq$$

$$\leq \left\| M_{m^{(l)},g_k^{(l)},f_k^{(l)}} - M_{m,g_k^{(l)},f_k^{(l)}} \right\| + \left\| M_{m,g_k^{(l)},f_k^{(l)}} - M_{m,g_k,f_k^{(l)}} \right\| + \left\| M_{m,g_k,f_k^{(l)}} - M_{m,g_k,f_k} \right\| \leq$$

$$\leq \epsilon \sqrt{\mathbf{BB'}} + \|m\| \epsilon \sqrt{\mathbf{B'}} + \|m\| \sqrt{B} \epsilon = \epsilon \cdot \left(\sqrt{\mathbf{BB'}} + \|m\| \left(\sqrt{\mathbf{B'}} + \sqrt{B} \right) \right)$$

for an l bigger than the maximum N needed for the convergence conditions. This is true for all pairs or norms (Op, ∞), $(\mathcal{H}S, l^2)$ and $(trace, l^1)$. $\square$

Remark: For item $4(a)$ it is sufficient if there is common bound for the norm of the frame elements, following the remark right after Theorem 1.3.13.

1.3.7 Riesz Multipliers

1.3.7.1 From Symbol To Operator

As we have seen above, the question, whether the relation $m \mapsto \mathbf{M}$ is injective, is very interesting. This is equivalent to the following questions: When is the operator uniquely defined by the symbol? When is the relation σ a function?

Compare this problem to the word problem in polynomial algebras ([7],[81]). There the question is, when do two polynomials $p = \sum a_k x^k$ and $q = \sum b_i x^i$ give rise to the same function. In this context the function from the "formal" polynomials $G[X]$ to the polynomial functions $P_k(G)$ is investigated. So in this context the mapping $\mathfrak{m} : l^\infty \to \mathcal{B}(\mathcal{H})$ and its kernel could be investigated.

This needs further investigation, but we already know from Theorem 1.2.23 that the rank one operators $g_k \otimes \overline{f}_k$ form a Bessel sequence in $\mathcal{H}S$. So the question is equivalent to the question whether they form a Riesz sequence. Following [34] we define

Definition 1.3.4 *Let (g_k), (γ_k) be Bessel sequences. We call it a* **well-balanced** *pair if $(g_k \otimes \overline{\gamma}_k)$ forms a Riesz sequence in $\mathcal{H}S$.*

We call (g_k) a **well-balanced** *Bessel sequence if $(g_k \otimes \overline{g}_k)$ forms a Riesz sequence in $\mathcal{H}S$.*

We will show in the next section, that Riesz bases are certainly well-balanced.

1.3.7.2 Uniqueness Of The Upper Symbol

We know from Theorem 1.2.23 that for Riesz bases the family $(g_k \otimes \overline{f}_k)$ is a Riesz sequence, so in this case the question when $m \to \mathbf{M}_m$ is injective is answered. We can state a more general property

Lemma 1.3.19 *Let (g_k) be a Bessel sequence where no element is zero, and (f_k) a Riesz sequence. Then the mapping $m \mapsto \mathbf{M}_{m,f_k,g_k}$ is injective.*

Proof: Suppose $\mathbf{M}_{m,f_k,g_k} = \mathbf{M}_{m',f_k,g_k}$

$$\Longrightarrow \sum_k m_k \langle f, g_k \rangle f_k = \sum_k m'_k \langle f, g_k \rangle f_k \text{ for all } f$$

As f_k is a Riesz basis for its span

$$\Longrightarrow m_k \langle f, g_k \rangle = m'_k \langle f, g_k \rangle \quad \text{for all } f, k$$

Because $g_k \neq 0$ for all $k \in K$, there exist an f such that $\langle f, g_k \rangle \neq 0$. Therefore

$$m_k = m'_k \text{ for all } k$$

□

So if the conditions in Lemma 1.3.19 is fulfilled the Bessel sequence $(g_k \otimes \overline{f}_k)$ is a Riesz sequence.

1.3.7.3 Operator Norm Of Multipliers

As mention in Section 1.3.3 the multiplier for Riesz basis share some of the nice properties of ONB multipliers.

Proposition 1.3.20 *Let (g_k) be a Riesz basis with bounds A, B and (f_k) be one with bounds A', B'. Then*

$$\sqrt{AA'}\, \|m\|_\infty \leq \|\mathbf{M}_{m,f_k,g_k}\|_{Op} \leq \sqrt{BB'}\, \|m\|_\infty$$

Particularly $\mathbf{M}_{m,f_k,g_k}$ is bounded if and only if m is bounded.

Proof: Theorem 1.3.13 gives us the upper bound.
For the lower bound let k_0 be arbitrary, then

$$\mathbf{M}_{m,f_k,g_k}(\tilde{g}_{k_0}) = \sum_k m_k \langle \tilde{g}_{k_0}, g_k \rangle f_k = m_{k_0} f_{k_0}$$

Therefore

$$\|\mathbf{M}_{m,f_k,g_k}\|_{Op} = \sup_{f \in \mathcal{H}} \left\{ \frac{\|\mathbf{M}_{m,f_k,g_k}(f)\|_{\mathcal{H}}}{\|f\|_{\mathcal{H}}} \right\} \geq$$

$$\geq \frac{\|\mathbf{M}_{m,f_k,g_k}(\tilde{g}_{k_0})\|_{\mathcal{H}}}{\|\tilde{g}_{k_0}\|_{\mathcal{H}}} \geq \frac{\|m_{k_0} f_{k_0}\|_{\mathcal{H}}}{\|\tilde{g}_{k_0}\|_{\mathcal{H}}} \geq \frac{|m_{k_0}|\, \sqrt{A'}}{\frac{1}{\sqrt{A}}} \geq \sqrt{A'A}\, |m_{k_0}|$$

using Theorem 1.1.7 and Corollary 1.1.34.

□

1.3.7.4 Combination Of Riesz Multipliers

For an ONB this is clearly just the multiplier with $\sigma = m \cdot m'$. But this is true for all biorthogonal sequences.

Proposition 1.3.21 *Let (g_k), (g'_k), (f_k) and (f'_k) be Bessel sequences, such that (f'_k) and (g_k) are biorthogonal to each other, then*

$$\left(\mathbf{M}_{m^{(1)},f_k,g_k} \circ \mathbf{M}_{m^{(2)},f'_k,g'_k}\right)(f) = \mathbf{M}_{m^{(1)}\cdot m^{(2)},f_k,g'_k}$$

Proof: We know from Lemma 1.3.12

$$\left(\mathbf{M}_{m^{(1)},f_k,g_k} \circ \mathbf{M}_{m^{(2)},f'_k,g'_k}\right)(f) = \sum_k \sum_l m_k^{(1)} m_l^{(2)} \left\langle f, g'_l \right\rangle \left\langle f'_l, g_k \right\rangle f_k = (*)$$

So if (f'_l) and (g_k) are biorthogonal

$$(*) = \sum_k \sum_l m_k^{(1)} m_l^{(2)} \left\langle f, g'_l \right\rangle \delta_{l,k} f_k = \sum_k m_k^{(1)} m_k^{(2)} \left\langle f, g'_k \right\rangle f_k$$
$$= \mathbf{M}_{m^{(1)}\cdot m^{(2)},f_k,g'_k}$$

□

A direct consequence is

Corollary 1.3.22 *Let (g_k) be a Riesz sequence, then*

$$\mathbf{M}_{m^{(1)},\tilde{g}_k,g_k} \circ \mathbf{M}_{m^{(2)},\tilde{g}_k,g_k} = \mathbf{M}_{m^{(1)}\cdot m^{(2)},\tilde{g}_k,g_k}$$

So we see that even for Riesz bases the combination of Gabor multipliers is not trivial (like in the ONB) case, only if the basis and its dual is used for the multiplier we get the following equality:

$$\sigma\left(\mathbf{M}_m \circ \mathbf{M}_{m'}\right) = \sigma\left(\mathbf{M}_m\right) \cdot \left(\mathbf{M}_{m'}\right)$$

The reverse of Corollary 1.3.22 is also true: If for a Bessel sequence, which does not contain zero elements, the symbol of the product of any multipliers is the product of the symbols of each multiplier, it is a Riesz Basis.

For this result we first show the following property:

Proposition 1.3.23 *Let (g_k) and (f_k) be Bessel sequences in $\mathcal{H}$. If $\forall m^{(1)}, m^{(2)} \in c_c(\mathbb{K})$*

$$\mathbf{M}_{m^{(1)},g_k,f_k} \circ \mathbf{M}_{m^{(2)},g_k,f_k} = \mathbf{M}_{m^{(1)}\cdot m^{(2)},g_k,f_k}$$

then for all k, k' either

- $f_{k'} = 0$ *or*
- $g_k = 0$ *or*
- $\langle g_k, \tilde{g}_{k'} \rangle = \delta_{kk'}$

Proof: Choose k_0, k_1 in the index set. Let $m = \delta_{k_0}$ and $\tilde{m} = \delta_{k_1}$.

$$\mathbf{M}_{m^{(1)},g_k,f_k} \circ \mathbf{M}_{m^{(2)},g_k,f_k} = \mathbf{M}_{m^{(1)} \cdot m^{(2)},g_k,f_k}$$

is in this case equivalent via equation 1.3.12 to

$$\langle f, g_{k_1} \rangle \langle f_{k_1}, g_{k_0} \rangle \cdot f_{k_0} = \delta_{k_0,k_1} \langle f, g_{k_1} \rangle f_{k_0} \ \forall\ f \in \mathcal{H}$$

<u>1. case</u>: Let $k_1 \neq k_0$ then this means that we obtain

$$\langle f, g_{k_1} \rangle \langle f_{k_1}, g_{k_0} \rangle \cdot f_{k_0} = 0$$

So either
1a.) $f_{k_0} = 0$ or
1b.) $\langle f, g_{k_1} \rangle = 0$ for all f, but then $g_{k_1} = 0$, or
1c.) $\langle f_{k_1}, g_{k_0} \rangle = 0$

<u>2. case</u>: Let $k_1 = k_0$.

$$\langle f, g_{k_1} \rangle \left(\langle f_{k_1}, g_{k_0} \rangle - 1 \right) f_{k_0} = 0$$

Either
2a.) $f_{k_0} = 0$ or
2b.) $\langle f, g_{k_1} \rangle = 0$ for all f and so $g_{k_1} = 0$ or
2c.) $\langle f_{k_1}, g_{k_0} \rangle = 1$

□

So we can find a way to classify Riesz bases by multipliers:

Theorem 1.3.24 *Let (g_k) and (f_k) be frames with $g_k \neq 0$ and $f_k \neq 0$ for all $k \in K$. If and only if $\sigma\left(\mathbf{M}_{m^{(1)},f_k,g_k} \circ \mathbf{M}_{m^{(2)},f_k,g_k}\right) = \sigma\left(\mathbf{M}_{m^{(1)},f_k,g_k}\right) \cdot \sigma\left(\mathbf{M}_{m^{(2)},f_k,g_k}\right)$ for all multiplier $\mathbf{M}_{m^{(1)},f_k,g_k}$, $\mathbf{M}_{m^{(1)},f_k,g_k}$ with $m^{(1)}$,$m^{(2)}$ finite, then these frames are biorthogonal to each other and so are Riesz bases.*

Proof: One direction is 1.3.23. The other 1.3.22. □

1.3.7.5 Commutation Of Multiplier

For the commutation of multipliers we get

Corollary 1.3.25 *Let (g_k) be a Riesz sequence, then*

$$\mathbf{M}_{m^{(1)},\tilde{g}_k,g_k} \circ \mathbf{M}_{m^{(2)},\tilde{g}_k,g_k} = \mathbf{M}_{m^{(2)},\tilde{g}_k,g_k} \circ \mathbf{M}_{m^{(1)},\tilde{g}_k,g_k}$$

Proof: We will use Corollary 1.3.22

$$\left(\mathbf{M}_{m^{(1)},\tilde{g}_k,g_k} \circ \mathbf{M}_{m^{(2)},\tilde{g}_k,g_k}\right)(f) = \sum_k m_k^{(1)} m_k^{(2)} \langle f, \tilde{g}_k \rangle g_k$$

On the other hand

$$\left(\mathbf{M}_{m^{(2)},\tilde{g}_k,f_k} \circ \mathbf{M}_{m^{(2)},\tilde{f}_k,g_k}\right)(f) = \sum_l m_l^{(2)} m_l^{(1)} \langle f, \tilde{g}_l \rangle g_l$$

□

1.3.7.6 Injectivity And Surjectivity

Using Lemma 1.3.4 in the case of Riesz basis means that a multiplier G_{m,g_k,g_k} can never be surjective, if the symbol is in l^2. Also if the symbol is not zero, the multiplier is injective:

Corollary 1.3.26 *A Hilbert Schmidt multiplier with Riesz bases is*

1. *not surjective.*
2. *injective, if and only if the symbol m is non-zero everywhere.*

Proof: $\mathbf{M}_m = D \circ \mathcal{M}_m \circ C$, D and C are bijective. And so $\mathbf{M}_m$ is injective respectively surjective if $\mathcal{M}_m$ is. □

The last property is true for all operator classes:

Lemma 1.3.27 *Let (f_k) and (g_k) Riesz Bases. Let $\mathbf{M} = \mathbf{M}_{m,g_k,f_k}$. Then $\mathbf{M}$ is injective $\Longleftrightarrow m_k \neq 0$ for all k.*

Proof: As (f_k) is a Riesz basis

$$\mathbf{M}f = \mathbf{M}g \Longleftrightarrow m_k \cdot \langle f, g_k \rangle = m_k \cdot \langle g, g_k \rangle \Longleftrightarrow m_k \cdot (\langle f, g_k \rangle - \langle g, g_k \rangle) = 0$$

Suppose $m_k \neq 0$ for all $k \in K \Longrightarrow \langle f, g_k \rangle = \langle g, g_k \rangle$ As g_k is a frame $\Longrightarrow f = g$.

Suppose $m_{k'} = 0$, let $c_k = \begin{cases} d & k = k' \\ \langle f, g_k \rangle & otherwise \end{cases}$. As (g_k) is a Riesz basis, there is a g such that $\langle g, g_k \rangle = c_k$. But then $\mathbf{M}f = \mathbf{M}g$. This is a contradiction to $\mathbf{M}$ being injective.

□

1.3.7.7 Inverse Riesz Multiplier

We can now ask, when is a multiplier invertible, or more precise when is the inverse another multiplier. Clearly in infinite dimensional spaces no compact operator is invertible, so the symbol can not be in c_0.

Proposition 1.3.28 *Let (g_k) and (f_k) be Riesz bases and let the symbol m be semi-normalized. Then*

$$\mathbf{M}_{m_k,f_k,g_k}^{-1} = \mathbf{M}_{\frac{1}{m_k},\tilde{g}_k,\tilde{f}_k}$$

Proof: Lemma 1.3.12 tells us that

$$\left(\mathbf{M}_{m,f_k,g_k} \circ \mathbf{M}_{\frac{1}{m},\tilde{g}_k,\tilde{f}_k}\right)(f) = \sum_k \sum_l m_k \frac{1}{m_l} \left\langle f, \tilde{f}_l \right\rangle \langle \tilde{g}_l, g_k \rangle f_k =$$

$$= \sum_k \sum_l m_k \frac{1}{m_l} \left\langle f, \tilde{f}_l \right\rangle \delta_{l,k} f_k = \sum_k m_k \frac{1}{m_k} \left\langle f, \tilde{f}_k \right\rangle f_k = f$$

□

For frames which are not Riesz bases, this proposition could give an idea how to find an approximation of the inverse operator. See [63] Section 14.1 where something similar for pseudodifferential operators is mentioned. Also see Section 3.4 where we will investigate a possibility to approximate the inverse of a non-diagonal matrix by diagonal matrices.

1.3.8 The Identity As Multiplier

The question in this section is: Can the identity be described as multiplier? For infinite dimensional spaces we already know, that the symbol cannot belong to c_0, if the identity is a multiplier. Due to Theorem 1.3.13 multipliers with symbols in c_0 are compact, but the identity is compact only in finite dimensional spaces.

We can show

Lemma 1.3.29 *If and only if the identity is a multiplier for the Bessel sequence $\{g_k\}$ with constant symbol $c \neq 0$, then $\{g_k\}$ is a tight frame.*

Proof: If $c = 0$ obviously $\mathbf{M}_m \equiv 0$. If the identity $I \equiv 0$, the space is a trivial vector space.

$$m_k = c \Leftrightarrow \mathbf{M}_m = \sum c g_k \otimes g_k = c \cdot S$$

$$c \cdot S = I \Leftrightarrow S = \frac{1}{c} I \Leftrightarrow \{g_k\} \text{ tight with } A = \frac{1}{c}$$

□

The question of when the identity is a frame multiplier is identical to the one, when a frame can be made tight be applying weights for example found in [101].

In the case of regular well-balanced Gabor frames, it is shown in [34] that if the identity can be written as multiplier for the frame $\{g_k\}$, its symbol is a constant sequence. Then Lemma 1.3.29 is clearly equivalent to: The identity is a multiplier if and only if $\{g_k\}$ is a tight frame. A simple exercise shows that this is not possible for general frames:

Example 1.3.1 :

Let $\{e_i | i = 1, 2, ..\}$ be an ONB for $\mathcal{H}$. Take $e_0 = e_1 + e_2$. Then $\{e_0, e_1, e_2, ...\}$ is a frame with the bounds 1 and 2. The identity can be described as multiplier with $m = (0, 1, 1, 1, 1, ...)$, but not as one with constant symbol. Suppose the multiplier M_c is the identity, then

$$e_3 = Id(e_3) = M_c(e_3) = \sum_{k=0}^{\infty} c \langle e_3, e_k \rangle e_k = c \cdot e_3$$

and therefore $c = 1$. But

$$M_1(e_1) = 1 \cdot \langle e_1, e_0 \rangle e_0 + 1 \cdot \langle e_1, e_1 \rangle e_1 = \langle e_1, e_1 + e_2 \rangle (e_1 + e_2) + e_1 =$$

$$= \langle e_1, e_1 \rangle (e_1 + e_2) + e_1 = 2e_1 + e_2 \neq e_1$$

1.3.9 Approximation Of Hilbert-Schmidt operators

We have investigated a certain class of operators, the frame multipliers. We now want to find the best approximation of operators in this class.

1.3.9.1 The Lower Symbol

We know from Theorem 1.2.23 that for Bessel sequences (g_k) and (f_k) the family $(g_k \otimes \overline{f}_k)$ is again a Bessel sequence. So the synthesis operator is well defined:

$$C_{g_k \otimes \overline{f}_k} : \mathcal{HS} \to l^2 \text{ with } C(T) = \left\langle T, g_k \otimes \overline{f}_k \right\rangle_{\mathcal{HS}}$$

Using Theorem 1.2.28 we can express that inner product as

$$C(T) = \left\langle T, g_k \otimes \overline{f}_k \right\rangle_{\mathcal{HS}} = \left\langle T\overline{f}_k, g_k \right\rangle_{\mathcal{H}}$$

So we define

Definition 1.3.5 *Let* (g_k) *and* $(\overline{f}_k)$ *be Bessel sequences for* $\mathcal{H}$*, then the* **lower symbol** *of an operator* $T \in \mathcal{HS}$ *is defined as*

$$\sigma_L(T) = \langle T f_k, g_k \rangle_{\mathcal{H}}$$

The function $\sigma_L : \mathcal{HS} \to l^2$ is just the synthesis operator of the Bessel sequence $g_k \otimes \overline{f}_k$ in $\mathcal{HS}$ and therefore well defined in l^2. The name is deserved in the case when the rank one operators $(g_k \otimes \overline{f}_k)$ fulfill the lower frame boundary condition for elements in its closed span. These elements form a frame sequence in this case. Following 1.1.14 we can find the best approximation by using the analysis and the dual synthesis operator for the projection on the closed span of the elements $V = \overline{span} \left\{ g_k \otimes \overline{f_k} \right\}$, which are exactly those $\mathcal{HS}$ operators that can be expressed as frame multipliers with the given frames. Let Q_k be the canonical dual frame of $g_k \otimes \overline{f}_k$ in V then the best approximation is

$$P_V(T) = \sum_k \left\langle T, g_k \otimes \overline{f}_k \right\rangle_{\mathcal{HS}} Q_k = \sum_k \sigma_L(T) Q_k$$

Due to Proposition 1.1.9 we know $\|\sigma_U\|_2 \leq \|\sigma_L(T)\|_2$ for any other coefficients σ_U such that the projection P_V can be expressed in this way and hence the name 'lower symbol'. Also for bounded operators T which are not in $\mathcal{HS}$ this inner product is defined and bounded by $\|T\|_{Op} \sqrt{BB'}$.

An interesting result is

Lemma 1.3.30 *Let* (g_k) *be a well-balanced Bessel sequence, which is not a Riesz sequence. Then none of the biorthogonal sequences* Q_k *in* $\mathcal{HS}$ *of* $P_k = g_k \otimes \overline{g}_k$ *can consist only of rank one operators* $\gamma_k \otimes \overline{\gamma}_k$.

Proof: Let suppose the opposite $Q_k = \gamma_k \otimes \gamma_k$ for all k

$$\langle P_k, Q_l \rangle_{\mathcal{HS}} = \langle g_k \otimes g_k, \gamma_l \otimes \gamma_l \rangle_{\mathcal{HS}} = \langle g_k, \gamma_l \rangle_{\mathcal{H}} \cdot \langle \gamma_l, g_k \rangle_{\mathcal{H}} = |\langle g_k, \gamma_l \rangle_{\mathcal{H}}|^2$$

Therefore (g_k) has a biorthogonal sequence (γ_k), which is a contradiction. $\square$

Let us repeat: Section 1.2.3.2 tells us that the rank one operators $(g_k \otimes \overline{f}_j)$ form a Bessel sequence, frame or Riesz basis if the sequences (g_k) and (f_k)

do. Therefore we have deduced that for Bessel sequences and Riesz bases the operators $(g_k \otimes \overline{f}_k)$ form a Bessel sequence or Riesz sequence. Subsequences of a frame do not have to be frame sequences, so it is not possible to deduce a similar property like above for frames. Also while Riesz bases in $\mathcal{H}$ give rise to Riesz bases in $\mathcal{HS}$, it would be interesting to classify all Bessel sequences respectively frames where this is true. In the regular Gabor case for example it can be shown [12] that for Bessel sequences (g_λ) and (f_λ) the family is either a Riesz sequence or no frame at all.

For the general case, due to the lack of a underlying group structure, such a classification seems hard to find. Only for orthonormal bases the connection is easy:

Lemma 1.3.31 *Let (g_k) be a Bessel sequence. If and only if it is an orthonormal basis, the sequence $(g_k \otimes \overline{g}_k)$ is an orthonormal system.*

Proof:

$$\langle g_k \otimes \overline{g}_k, g_l \otimes \overline{g}_l \rangle_{\mathcal{HS}} = \delta_{k,l} \iff$$
$$\langle g_k, g_l \rangle_{\mathcal{H}} \cdot \langle g_l, g_k \rangle_{\mathcal{H}} = \delta_{k,l} \iff$$
$$|\langle g_k, g_l \rangle_{\mathcal{H}}|^2 = \delta_{k,l} \iff$$
$$\langle g_k, g_l \rangle_{\mathcal{H}} = \delta_{k,l}$$

□

1.3.9.2 Perturbation For $\mathcal{HS}$ Riesz Sequences

The perturbation results in Section 1.1.12 give us tools to formulate a perturbation result for the rank one operators in $\mathcal{HS}$:

Theorem 1.3.32 *Let $(g_k), (\gamma_k)$ be a well-balanced pair of Bessel sequences with Bessel bounds B and B'. Let $(f_k^{(l)}), (\varphi_k^{(l)})$ be sequences such that for all ϵ there exists an $N(\epsilon)$ with*

$$\sum_k \left\| g_k - f_k^{(l)} \right\|_{\mathcal{H}}^2 < \epsilon \text{ and } \sum_k \left\| \gamma_k - \varphi_k^{(l)} \right\|_{\mathcal{H}}^2 < \epsilon$$

for all $l \geq N(\epsilon)$, then the Bessel sequences $((f_k^{(l)} \otimes \varphi_k^{(l)}))$ also form a Riesz sequence.

Proof: From Corollary 1.1.63 we know that for $l \leq N(A)$ $(f_k)^{(l)}$ is a Bessel sequences, frame respectively a Riesz basis and $C_{f_k^{(l)}} \to C_{g_k}$, $D_{f_k^{(l)}} \to D_{g_k}$ and $S_{f_k^{(l)}} \to S_{g_k}$ for $l \geq \max\{N(A), N(1)\}$, $l \to \infty$.

So we know that $(g_k \otimes \gamma_k)$ and $(f_k^{(l)} \otimes \varphi_k^{(l)})$ are Bessel sequences for $l \geq N(A)$.

$$\left(D_{g_k \otimes \gamma_k} - D_{f_k^{(l)} \otimes \varphi_k^{(l)}}\right)(c) = D_{g_k} \circ \mathcal{M}_c \circ C_{\gamma_k} - D_{f_k^{(l)}} \circ \mathcal{M}_c \circ C_{\varphi_k^{(l)}}$$

$$\Rightarrow \left\| \left(D_{g_k \otimes \gamma_k} - D_{f_k^{(l)} \otimes \varphi_k^{(l)}}\right)(c) \right\|_{\mathcal{HS}} = \left\| D_{g_k} \circ \mathcal{M}_c \circ C_{\gamma_k} - D_{f_k^{(l)}} \circ \mathcal{M}_c \circ C_{\varphi_k^{(l)}} \right\|_{\mathcal{HS}} =$$

$$= \left\| D_{g_k} \circ \mathcal{M}_c \circ C_{\gamma_k} - D_{f_k^{(l)}} \circ \mathcal{M}_c \circ C_{\gamma_k} + D_{f_k^{(l)}} \circ \mathcal{M}_c \circ C_{\gamma_k} - D_{f_k^{(l)}} \circ \mathcal{M}_c \circ C_{\varphi_k^{(l)}} \right\|_{\mathcal{HS}} \leq$$

$$\leq \left\| D_{g_k} - D_{f_k^{(l)}} \right\|_{Op} \left\| \mathcal{M}_c \circ C_{\gamma_k} \right\|_{Op} + \left\| D_{f_k^{(l)}} \circ \mathcal{M}_c \right\|_{Op} \left\| C_{\gamma_k} - C_{\varphi_k^{(l)}} \right\|_{Op} = (*)$$

For $l \geq N' = \max\{N(A), N(\epsilon)\}$.

$$(*) \leq \epsilon \left\| \mathcal{M}_c \right\|_{l^2 \to l^2} \left\| C_{\gamma_k} \right\|_{Op} + \left\| D_{f_k^{(l)}} \right\|_{Op} \left\| \mathcal{M}_c \right\|_{l^2 \to l^2} \epsilon$$

From Corollary 1.1.62 we know that there is a $N(1)$ such that $\left\| D_{f_k^{(l)}} \right\|_{Op} < \sqrt{B+1}$ for $l \geq N(1)$. So using Lemma 1.3.3 we get

$$(*) \leq \epsilon \left\| c \right\|_2 \sqrt{B'} + \sqrt{B+1} \left\| c \right\|_2 \epsilon = \epsilon \cdot \left\| c \right\|_2 \left(\sqrt{B'} + \sqrt{B+1}\right)$$

for all $l \geq N = \max\{N(1), N(A), N(\epsilon)\}$.

Therefore

$$\left\| D_{g_k \otimes \gamma_k} - D_{f_k^{(l)} \otimes \varphi_k^{(l)}} \right\|_{l^2 \to \mathcal{HS}} \leq \epsilon \cdot \left(\sqrt{B'} + \sqrt{B+1}\right)$$

Following Proposition 1.1.61 we can finish the proof as

$$\left\| D_{g_k \otimes \gamma_k} - D_{f_k^{(l)} \otimes \varphi_k^{(l)}} \right\|_{c_c^2 \to \mathcal{HS}} \leq \left\| D_{g_k \otimes \gamma_k} - D_{f_k^{(l)} \otimes \varphi_k^{(l)}} \right\|_{c_c^2 \to \mathcal{HS}} \leq \epsilon'$$

□

Again this can be specialized to

Corollary 1.3.33 *Let $(g_k), (\gamma_k)$ be Bessel sequences with Bessel bounds B and B', such that $(g_k \otimes \gamma_k)$ form a Riesz sequence . Let $(f_k), (\varphi_k)$ be sequences such that for all ϵ there exists an $N(\epsilon)$ with*

$$\sum_k \left\| g_k - f_k^{(l)} \right\|_{\mathcal{H}} < \epsilon \text{ and } \sum_k \left\| \gamma_k - \varphi_k^{(l)} \right\|_{\mathcal{H}} < \epsilon$$

for all $l \geq N(\epsilon)$, then the Bessel sequences $((f_k^{(l)} \otimes \varphi_k^{(l)}))$ also form a Riesz sequence.

1.3.9.3 Approximation Of Matrices By Frame Multipliers

In infinite-dimensional spaces not every subsequence of a frame is a frame sequence, but in the finite-dimensional case, all sequences are frame sequences. So we can use the ideas in Section 1.1.10.1 and apply it to frame multipliers.

We want to find the best approximation (in the Frobenius norm) of a $m \times n$ matrix T by a frame multiplier with the frames $(g_k)_{k=1}^K \subseteq \mathbb{C}^n$ and $(f_k)_{k=1}^K \subseteq \mathbb{C}^m$. This whole section is a generalization of the ideas in [50].

`Algorithm:`

1. `Inputs: T, D, Ds`
   T is a $m \times n$ matrix, D is the $n \times K$ synthesis matrix of the frame (g_k) and so following Section 1.2.1.3 this means that the element of the frame are the columns of D. Ds is the synthesis matrix of the frame (f_k). Often we will use the case $(f_k) = (g_k)$ so $D_s = D$ by default.

2. `Lower Symbol :`
   Using Lemma 1.2.26 the most efficient way to calculate the inner product $\left\langle T, g_k \otimes \overline{f}_k \right\rangle_{\mathcal{HS}}$ is $\langle T f_k, g_k \rangle_{\mathbb{C}^n}$. This can be implemented effectively using the matrix multiplication by
   ```
   (MATLAB :)     lowsym(i) = conj(D(:,i)'*(T*Ds(:,i)));
   ```

3. `Hilbert Schmidt Gram Matrix :`
   We calculate the Gram matrix of $(g_k \otimes \overline{f}_k)$
   $$(G_{\mathcal{HS}})_{l,k} = \left\langle g_k \otimes \overline{f}_k, g_l \otimes \overline{f}_l \right\rangle_{\mathcal{HS}} = \langle g_k, g_l \rangle_{\mathcal{H}} \cdot \langle f_l, f_k \rangle_{\mathcal{H}} = (G_{g_k})_{l,k} \cdot (G_{f_k})_{k,l}$$
   ```
   (MATLAB :) Gram = (D'*D).*((Ds'*Ds).');
   ```
   If $(g_k) = (f_k)$ then
   $$(G_{\mathcal{HS}})_{l,k} = |\langle g_k, g_l \rangle_{\mathcal{H}}|^2$$
   It is more efficient to use this formula in
   ```
   (MATLAB :) Gram = abs((D'*D)).^2;
   ```
   as this has complexity, using Lemma 1.2.26, $\sim K^2 \cdot (n^2 + 2)$ compared to the original calculation with $\sim K^2 \cdot (n^2 + m^2 + 1)$.

4. `Upper Symbol :`
   Using Theorem 1.1.52 we get the coefficients of the approximation by using the pseudoinverse of the Gram matrix. In the case of frame multipliers the coefficients are an upper symbol σ.
   ```
   (MATLAB :) uppsym = pinv(Gram)*lowsym;
   ```

5. `Outputs: TA, uppsym`
   For the calculation of the approximation we just have to apply the synthesis operator of the sequence $(g_k \otimes \overline{f}_k)$ to the upper symbol.

$$\mathtt{TA} = P_V(T) = \sum_{k=1}^{K} \sigma_k g_k \otimes \overline{f}_k$$

   From Lemma A.4.42 we know that the matrix of the operator $g_k \otimes \overline{f}_k$ can just be calculated by $(g_k)_i \cdot \left(\overline{f}_k\right)_j$.
   `(MATLAB :) P = D(:,i)*Ds(:,i)';`

For an implementation of this algorithm in MATLAB see Section B.1.

Example 1.3.2 :

We will look at two simple example in $\mathbb{C}^2$.

1. Let $A = \begin{pmatrix} 3 & 0 \\ 0 & 5 \end{pmatrix}$. This is clearly a multiplier for the standard orthonormal basis of $\mathbb{C}^2$. The sequence $f_1 = \left(\frac{1}{2}, \frac{\sqrt{3}}{2}\right), f_2 = \left(\frac{\sqrt{3}}{2}, -\frac{1}{2}\right)$ is also an ONB. But the best approximation of A with this basis is $P_V(A) = \begin{pmatrix} 3.7500 & 0.4330 \\ 0.4330 & 4.2500 \end{pmatrix}$. So this is an example that not even for ONBs a frame multiplier for one basis stays a frame multiplier for another one.

2. Let $T = Id_{\mathbb{C}^2}$. and let $D = \begin{pmatrix} \cos(30°) & 1 & 0 \\ \sin(30°) & 1 & -1 \end{pmatrix}$. This is a frame with bounds $A = 0.5453$, $B = 3.4547$ and therefore not tight. Still the identity can be approximated perfectly (up to numerical precision) with the coefficients $\sigma = (3.1547, -1.3660, 1.5774)$. So this is an example, where the identity is a frame multiplier for a non-tight system.

The MATLAB-codes for these examples can be found in the appendix in Section B.1.2.

Example 1.3.3 :

We will now use this algorithm for the Gabor case, as a connection to the next chapter. We are using a Gauss window in $\mathbb{C}^n$ with $n = 32$. We are changing the lattice parameters a and b. The resulting approximation of the identity can be found in Figure 1.4.

1. $(g, a = 2, b = 2)$. This is nearly a tight frame with the lower frame bound $A = 7.99989$ and the upper frame bound $B = 8.00011$. As expected the identity is approximated very well.

2. $(g, a = 4, b = 4)$: This frame is not tight anymore, as $A = 1.66925$ and $B = 2.36068$ and we can see that the approximation is deviating from identity.

3. $(g, a = 8, b = 8)$: This is not a frame anymore, but a Bessel sequence with $B = 1.18034$. At least some of the structure (the diagonal dominance) is still kept.

4. $(g, a = 16, b = 16)$: This is not a frame anymore, but a Bessel sequence with $B = 1.00001$. All structure is (more or less) lost.

This algorithm is not very efficient for the Gabor case as the special structure is not used. For the regular case the algorithm presented in [50] is preferable. We will try to speed up this algorithm for irregular Gabor systems in Section 2.7.3.

The MATLAB-codes for these examples can be found in the appendix in Section B.1.3.

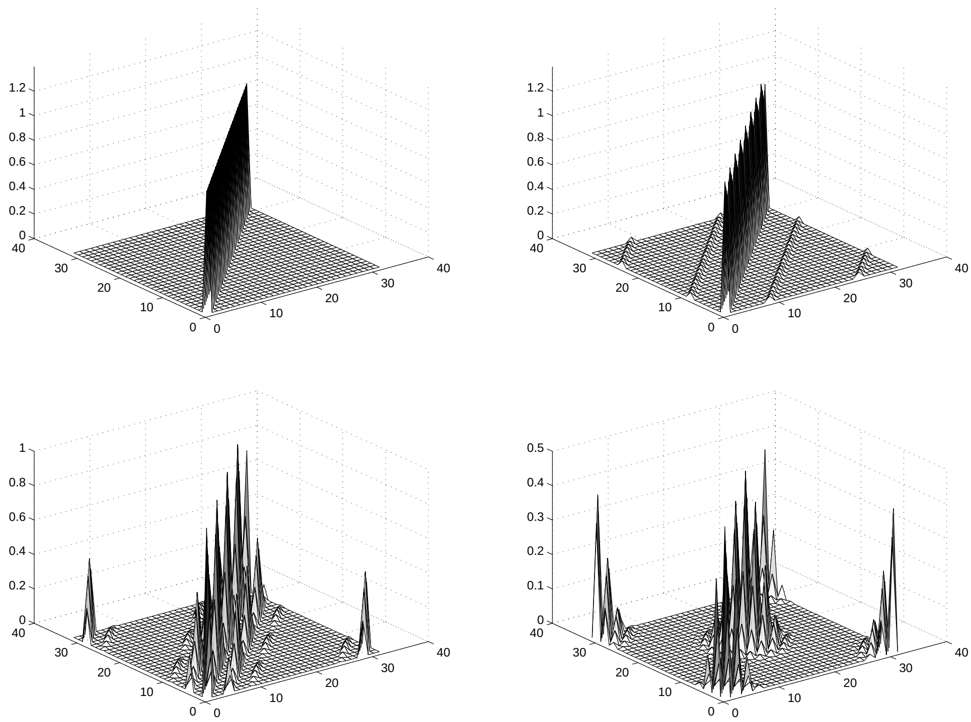

Figure 1.4: Using the algorithm for approximation with frame multipliers in the Gabor case: Approximating the identity by Gabor multiplier with Gauss window ($n = 32$) and changing lattice parameters. Top Left: ($a = 2$,$b = 2$), Top Right: ($a = 4$,$b = 4$), Bottom Left: ($a = 8$,$b = 8$), Bottom Right: ($a = 16$,$b = 16$)

Chapter 2

Gabor Frames And Multipliers

The *Fourier Transformation*, refer to Section A.4.7, is a well known mathematical tool to analyze the frequency content of a signal. Due to the very efficient algorithms of the *fast Fourier transformation*, *FFT*, see e.g. [126], many applications and developments are possible. If humans listen to a sound, a voice or music, they do not only hear frequencies and their amplitudes but also their dynamic development. So it is very natural to search for a joint time frequency analysis, for a two dimensional representation, that shows the frequency and time information of the signal. This is not possible in an exact way, as with the uncertainty principle there is always a trade off between the precision in time and in frequency, see [63] and Theorem 2.1.15.

A well known method for a time frequency representation is the *short time Fourier transformation*, *STFT*, see Definition 2.1.1. One possibility to look at this method is to take the signal $x(\tau)$ and multiply it with a window function $w(\tau - t)$ to get a version of the signal that is concentrated at the time t (if the window is chosen accordingly, centered at zero). Then the Fourier transformation is applied to the result:

$$X_w(t, \omega) = \int_{-\infty}^{\infty} x(\tau) w(\tau - t) e^{-i2\pi\omega\tau} d\tau$$

For the moment let us ignore the question, when and where this is well-defined or can be generalized, at the moment. And let us suppose that the window $w(t)$ is real-valued.

In application (and this work) the discrete finite case is important, so the finite, discrete equivalent for the above definition at the time sample n and the frequency bin k is

$$X_w[n, k] = \sum_{m=0}^{N-1} x[m] w[m - n] e^{\frac{-2\pi i k m}{N}}$$

which is the (regularly) sampled version of the continuous STFT using the sampling points $\tau = n \cdot T$, $\omega = \frac{k}{N \cdot T}$. For more on the discrete finite case see Chapter 3.

Another possibility to look at the STFT is to see it as a *filter bank*. These filters $f(t)$, as convolution operators $x \mapsto (x * f)$, have certain properties, they have to add up to 1, they have to have all the same frequency response characteristics except the center frequency and these center frequencies have to be evenly spaced. Of course these filters stand in close connection to the windowing function of the "Fourier view". A schematic drawing of this filter bank is given in Figure 2.1.

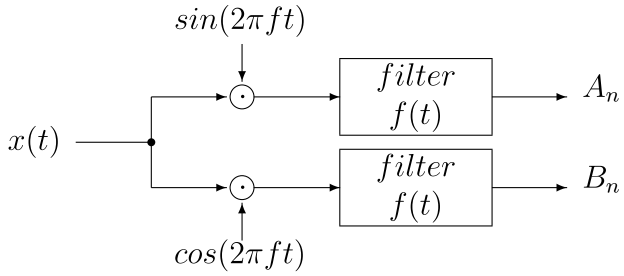

Figure 2.1: n^{th} channel of a channel vocoder

If the two outputs A_n and B_n are seen as the real and imaginary part of a complex number, this is exactly the result of the STFT. If only the amplitude, $\sqrt{A_n^2 + B_n^2}$, is worked with, the STFT is the equivalent of a so called *Channel Vocoder* [61]. This method is called *Phase Vocoder*, if the phase is not ignored, but its temporal difference is used to get a better estimation of the frequency of sinusoidal parts. It is a very common tool in modification of audio signals, see e.g. [98] [8]. For an investigation of the equivalence between Gabor analysis and filter-banks see [14].

A different view point than the two above (which are also those two mentioned in [32]) can be taken: the *Gabor* way, in which time and frequency are seen as symmetric. In this context there is no ordering of time and frequency processing, it is not an analysis first over time and then over frequency or the other way around. The signal is projected on atoms, that have a certain time frequency spread. These atoms are found by time and frequency shifted versions of a function $g(t)$, the Gabor atom. This atom again corresponds with the window and the filter mentioned above. So the projection of the signal $x(t)$ on the shifted atom $M_{kb}T_{la}g(t)$ with $a = T, b = \frac{1}{N \cdot T}$ yields the coefficients

$$f \mapsto \langle f, M_{kb}T_{la}g \rangle$$

The time frequency spread of the shifted atoms is depicted in Figure 2.2:

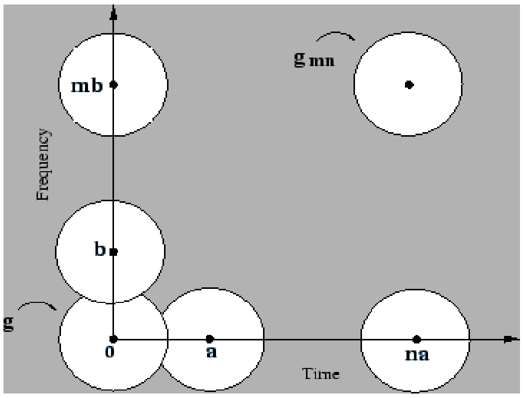

Figure 2.2: If g is centered around zero, the atoms $g_{m,n}$ are centered around (na, mb). (from [44])

All three view points are mathematically (and therefore as algorithms) equivalent, refer to Lemma 2.1.4, but give three different methods to analyze problems. The Fourier view is useful e.g. for implementing an algorithm, the filter bank view to know what to expect acoustically and the Gabor view gives a compact mathematical description so that certain properties of windows can be found.

In applications it is often interesting to reduce the amount of data and therefore the number of computations, which means reducing the *redundancy* of the representation. The redundancy can be defined for this setting as

$$red = \frac{1}{\Delta f \cdot \Delta t}$$

It is clear from the statements above that the redundancy of the full STFT in the finite discrete setting is

$$red = \frac{1}{\Delta f \cdot \Delta t} = \frac{N \cdot T}{T} = N$$

The full STFT in the finite discrete case of $\mathbb{C}^N$ has N^2 entries. So we describe an N-dimensional vector by an N^2-dimensional one, which is a factor N. Hence this factor is called redundancy, see also Theorem 2.1.14.

To reduce it, one can (in the Fourier view) calculate the spectrum not at every time step, but only every H seconds (or samples). H is called *hop size*. In the filter view this is a down sampling of the filtered signal. In the

Gabor view many different choice for a sampling lattice are possible. If not the full STFT but a sampled version of it is used, the method is called *Gabor transform*. We will investigate Gabor systems in Section 2.1.2.

If the goal is modification of signals, like in an masking algorithm, for every analysis method a synthesis method is needed. In this context it is the *overlap add* method, the *synthesis filter bank* respectively *oscillator bank*, or the projection on *dual atoms*, depending on the chosen viewpoint. For synthesis again a window (filter, atom) γ has to be chosen, although in some practical application "no" window ($\gamma \equiv 1$) or the analysis window ($\gamma = g$) can be chosen. If for the chosen parameters the Gabor system (g, a, b) forms a frame, this is a sufficient condition when perfect reconstruction from discrete samples of the STFT is possible.

Gabor [60] proposed, that in the case of Gauss windows the redundancy could be reduced to $red = 1$. It could be shown later that these functions constitute a *frame* if and only if $red < 1$. This has the consequence that there is a synthesis atom, which guarantees perfect reconstruction. The question whether certain windowing functions form frames for certain redundancies could be answered for many systems. It is clear that there is a kind of *"Nyquist criteria"* for Gabor frames, as it has be shown that no window function can be a frame for $red > 1$, see also Theorem 2.1.14. In application an overlap of 75%, i.e. a redundancy of 4, and a "standard" window function like a *Hanning* or *Blackman Harris* window will lead very often to satisfactory results. It is still an open problem to classify, when a Gabor system constitutes a frame. For certain classes of windows, there are positive results, for which lattice parameters frames are formed. Refer to Section 2.1.2.

If one is mainly interested in perceptual features, any part of the signal that cannot be heard is obviously redundant: So the representation can be made more sparse by restricting it to to the psychoacoustical relevant parts. This is exactly what masking filters do. We will investigate this issue further in chapter 4.

Many modern tools rely on signal processing algorithms. Due to the fast algorithm of the FFT in the last 50 years many practical application of time-invariant filters have been found. In recent years a lot of attention has been given to time-variant filtering, cf. e.g. [70]. One way to implement a time-variant filter is to use a multiplier on the STFT coefficients. For the sampled version we will have Gabor multipliers, which we will look at in Section 2.3.

There are many other time frequency representations like the wavelet analysis [29], Gabor analysis with irregular lattices, cf. Section 2.2 or the Wigner Ville representation [86].

In this chapter we will first look at the basic definitions of the STFT and the Gabor transform in Section 2.1. In Section 2.2 we will shortly investigate irregular Gabor systems. In Section 2.3 we will use the theory developed in Section 1.3 and 2.1.2 for Gabor multipliers, most notable irregular Gabor multipliers.

2.1 Introduction And Preliminaries

2.1.1 Short Time Fourier Transformation

The well known definition of the STFT is

Definition 2.1.1 *Let* f,$g \neq 0$ *in* $L^2\left(\mathbb{R}^d\right)$*, then we call*

$$\mathcal{V}_g f(t,\omega) = \int_{\mathbb{R}^d} f(x)\overline{g(x-t)} e^{-2\pi i \omega x} dx\,.$$

the **Short Time Fourier Transformation (STFT)** *of* f *with the* **window** g.

In applications often the quadratic representation $\left|\mathcal{V}_g f(t,\omega)\right|^2$ is used. This is called the *spectrogram*. For a picture of a typical spectrogram of an audio signal see Figure 2.3.

We can give alternate ways to describe the STFT. For that we need the following transformation

Definition 2.1.2 *Let* $f : \mathbb{R}^d \to \mathbb{C}$*. Then define for* τ *and* $\omega \in \mathbb{R}$ *the* **translation** *by* τ

$$\left(T_\tau f\right)(t) = f(t-\tau)$$

and the **modulation** *by* ω

$$\left(M_\omega f\right)(t) = e^{2\pi i \omega t} f(t)$$

The operators $\pi\left((\tau,\omega)\right) = M_\omega T_\tau$ *are called* **time-frequency shifts**.

In the context of time frequency we will use the notation $\omega(t)$ (and symmetrically $t(\omega)$) for the factor $e^{2\pi i \omega t}$. Especially in the context of irregular Gabor systems we will use the notation $\lambda = (t,\omega) \in \mathbb{R}^{2d}$ for time-frequency points. The transformations M_ω and T_τ are clearly unitary operators with $M_\omega^* = M_\omega^{-1} = M_{-\omega}$ and $T_\tau^* = T_\tau^{-1} = T_{-\tau}$.

Let us collect the most important properties of this operators which can be found for example in [63]

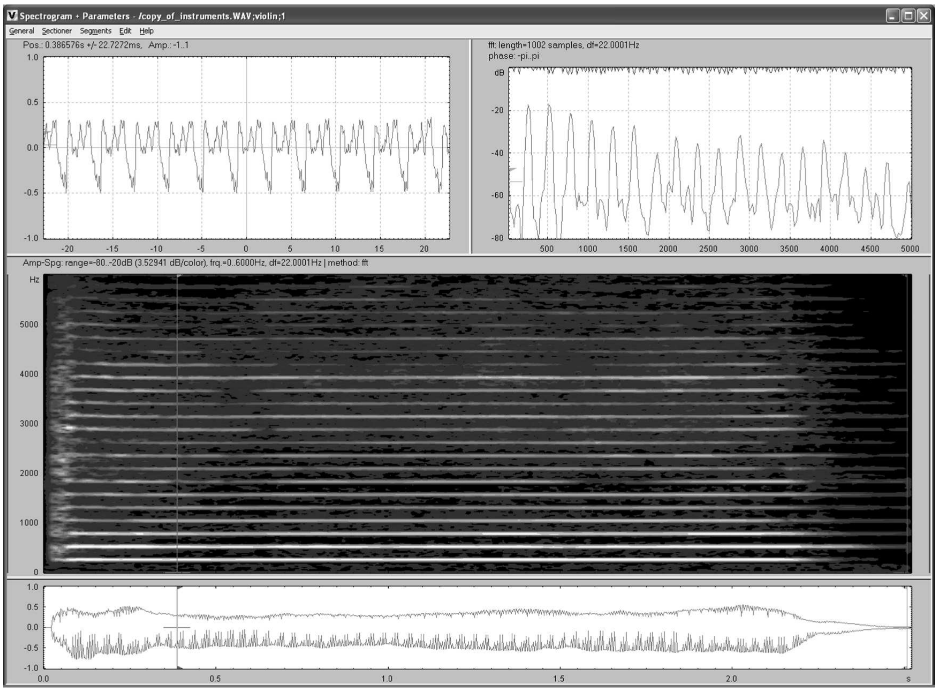

Figure 2.3: The STFT: The spectrogram of a violin sound (screenshot of ST^X[96]). Left Top: the part of the signal at the cursor. Right Top: the spectrum at the cursor. Middle: the spectrogram. Bottom: the whole signal in the time domain

Lemma 2.1.1 *1.*

$$M_\omega T_\tau = \omega(\tau) T_\tau M_\omega$$

2.

$$\|M_\omega T_\tau f\|_p = \|f\|_p$$

3.

$$\widehat{(M_\omega f)} = T_\omega \hat{f} \quad \widehat{(T_\tau f)} = M_{-\tau} \hat{f}$$

where $f \mapsto \hat{f}$ is the Fourier transformation.

4.

$$\lim_{\tau \to 0} \|T_\tau f - f\|_2 = 0$$

5.

$$\lim_{\omega \to 0} \|M_\omega f - f\|_2 = 0$$

6.

$$\pi^*(\tau,\omega) = \overline{\omega(\tau)}\pi(-\tau,-\omega)$$

7.

$$\pi(\tau_1,\omega_1)\circ\pi(\tau_2,\omega_2) = \overline{\omega_2(\tau_1)}\pi\left(\tau_1+\tau_2,\omega_1+\omega_2\right)$$

Banach spaces, which fulfill point 2., 4. and 5., i.e. spaces, where the translations and modulations are isometric and strongly continuous, are called *time-frequency homogeneous*.

As a direct consequence of the last lemma and the fact that M_ω and T_τ are unitary operators we get

Corollary 2.1.2 *1.*

$$\lim_{\tau\to\tau_0}\|T_\tau f - T_{\tau_0} f\|_2 = 0$$

2.

$$\lim_{\omega\to\omega_0}\|M_\omega f - M_{\omega_0} f\|_2 = 0$$

Let us collect two more results needed in the following:

Corollary 2.1.3 *For $\lambda = (\tau,\omega)$ and $\lambda' = (\tau',\omega')$*

1.

$$\pi^*(\lambda)\circ\pi(\lambda') = \overline{(\omega+\omega')}(\tau)\pi$$

2.

$$\pi^*(\lambda) = \pi^{-1}(\lambda)$$

Proof: 1.)

$$\pi(\lambda)^*\pi(\lambda') \overset{Lem.2.1.1}{=} \overline{\omega(\tau)}\cdot\pi(-\lambda)\pi(\lambda') =$$

$$\overset{Lem.2.1.1}{=} \overline{\omega(\tau)}\cdot\overline{\omega'(-\tau)}\pi(\lambda'-\lambda) = \overline{(\omega-\omega')}(-\tau)\pi(\lambda'-\lambda)$$

2.)

$$\pi^*(\lambda)\pi(\lambda) \overset{Lem.2.1.1}{=} \overline{\omega(\tau)}\cdot\pi(-\lambda)\circ\pi(\lambda) =$$

$$\overset{Lem.2.1.1}{=} \overline{\omega(\tau)}\cdot\overline{\omega(-\tau)}\pi(\lambda-\lambda) == \overline{\omega(0)}\pi(0) = Id$$

For the opposite direction to proof can be done in an analogue way. □

We find different ways to describe the STFT because the following result is well-known.

Lemma 2.1.4 ([63] Lemma 3.1.1)*Let f,g be in $L^2\left(\mathbb{R}^d\right)$, then $\mathcal{V}_g(f)$ is uniformly continuous and*

1.

$$\mathcal{V}_g(f)(\tau,\omega) = (\widehat{f \cdot T_\tau \overline{g}})(\omega)$$

2.

$$\mathcal{V}_g(f)(\tau,\omega) = \langle f, M_\omega T_\tau g\rangle$$

3.

$$\mathcal{V}_g(f)(\tau,\omega) = e^{-2\pi i x\cdot\omega}\left(f * M_\omega g^*\right)(\tau)$$

From Item 1 we now learn that for the STFT we multiply the signal by the window shifted by τ and then do a Fourier transformation. From item 3 we see how to interpret it as convolution. So the initial comments in this chapter about the equivalence of the different views have been certified.

Item 2 gives a change to generalize the idea of the STFT to other function space respectively function spaces and their duals. For example for tempered distributions $\mathcal{S}'$ [63], S_0' [42], Modulations spaces [63] or on locally compact Abelian groups [62]. For time-frequency homogeneous function spaces a lot of properties stay the same, see e.g. [42].

The STFT is invertible as stated in the *inversion formula for the STFT.*

Corollary 2.1.5 ([63] Corollary 3.2.3) *Let* $g,\gamma \in L^2\left(\mathbb{R}^d\right)$ *and* $\langle g,\gamma\rangle_{L^2} \neq 0$. *Then*

$$f(t) = \frac{1}{\langle g,\gamma\rangle_{L^2\left(\mathbb{R}^d\right)}} \int\limits_{\mathbb{R}^{2d}} \mathcal{V}_g f(s,\omega)\gamma(t-s)e^{2\pi i\omega t} ds d\omega\,.$$

This is a direct consequence of the *orthogonality relations for the STFT*:

Theorem 2.1.6 ([63] Theorem 3.2.1) *Let* $f_1, f_2, g_1, g_2 \in L^2\left(\mathbb{R}^d\right)$, *then* $\mathcal{V}_{g_j} f_j \in L^2\left(\mathbb{R}^{2d}\right)$ *for* $j = 1, 2$ *and*

$$\langle \mathcal{V}_{g_1} f_1, \mathcal{V}_{g_2} f_2\rangle_{L^2\left(\mathbb{R}^{2d}\right)} = \langle f_1, f_2\rangle_{L^2\left(\mathbb{R}^d\right)} \cdot \overline{\langle g_1, g_2\rangle_{L^2\left(\mathbb{R}^d\right)}}$$

This theorem implies

Corollary 2.1.7 ([63] Corollary 3.2.2) *Let* $f, g \in L^2\left(\mathbb{R}^d\right)$, *then*

$$\|\mathcal{V}_g f\|_2 = \|f\|_2 \|g\|_2$$

In particular, if $\|g\|_2 = 1$ *then*

$$\|\mathcal{V}_g f\|_2 = \|f\|_2 \text{ for all } f \in L^2\left(\mathbb{R}^d\right)$$

So for $\|g\|_2 = 1$ *the STFT is an isometry from* $L^2\left(\mathbb{R}^d\right)$ *to* $L^2\left(\mathbb{R}^{2d}\right)$.

2.1.2 Gabor Systems

For applications discrete time-frequency representations are necessary. The STFT is sampled at certain time-frequency points, $STFT(t_l, \omega_l)$. In this case the inversion is not always possible. We will apply the theory of frames from Section 1.1 for the sampled STFT. In this case we know that inversion is possible, if the a frame is formed:

Definition 2.1.3 *Let $g \in L^2(\mathbb{R}^d)$ be a non zero function, the so called* **window**. *Given parameters $\alpha, \beta > 0$, α is called the time and β the frequency shift respectively. The set of time-frequency shifts*

$$\mathcal{G}(g, \alpha, \beta) = \{T_{\alpha k} M_{\beta n} g : k, n \in \mathbb{Z}^d\}$$

is called a **Gabor system**. *If it is a frame, it is called* **Gabor frame**.

The set $\{(\alpha k, \beta n) : k, n \in \mathbb{Z}^d\}$ is called the **lattice**.

Other sampling sets are possible, either with a group structure, see e.g. [99] or without, see Section 2.2.

It can be shown [63] that the dual frame for a Gabor frame is just the Gabor system of the **dual window** $\tilde{g} = S^{-1} g$. And $S_g^{-1} = S_{\tilde{g}}$.

The Gabor frame operators are time frequency operators, in the sense that the analysis operator is just the STFT, sampled at the time frequency points $(\alpha k, \beta n)$, because the inner product $\langle f, T_{\alpha k} M_{\beta n} g \rangle = V_g f(\alpha k, \beta n)$.

Similar to the general frame case we will write the (associated) analysis operator for the window g as $C_g f = \{\langle f, g_{k,n} \rangle\}$, the (associated) synthesis operator as $D_\gamma f = \sum\limits_{k,n} c_{k,n} g_{k,n}$ and $S_{g,\gamma} = D_g C_\gamma$ for the (associated) frame operator.

An important window class is the following

Definition 2.1.4 *Let $Q = [0,1]^d$. A function $g \in L^\infty\left(\mathbb{R}^d\right)$ belongs to the* **Wiener space** *$W = W(\mathbb{R}^d)$, if*

$$\|g\|_W = \sum_{n \in \mathbb{Z}^d} ess \sup_{x \in Q} |g(x+n)| < \infty$$

The subspace of continuous functions of W will be denoted by W_0.

These spaces are special cases of Amalgam spaces, refer to Section 2.1.3.1.

This norm can also be written as $\|g\|_W = \sum\limits_{n \in \mathbb{Z}^d} \|g T_n \chi_Q\|_\infty$. As all bounded functions with compact support are in W, we know that W is a subspace of every $L^p(\mathbb{R}^d)$ for $1 \leq p < \infty$. W is even densely embedded in these $L^p(\mathbb{R}^d)$ [63].

This window class is important, because it can be shown, that for this class of windows the analysis, synthesis and frame operator are bounded operators, and so the Gabor system always forms a Bessel sequence, see [63] chapter 6.2. Even more

Theorem 2.1.8 ([63] Theorem 6.5.1) *Let $g \in W$ and let $\alpha > 0$ be such that for constants $a, b > 0$*

$$a \leq \sum_{k \in \mathbb{Z}^d} |g(x - \alpha k)|^2 \leq b < \infty \text{ almost everywhere.}$$

Then there is a value $\beta_0 > 0$ depending on α such that $\mathcal{G}(g, \alpha, \beta)$ is a Gabor frame for all $\beta < \beta_0$.

There are several possible way to represent the Gabor frame operator, for the so called Walnut's representation we need the following definition:

Definition 2.1.5 *Let $g, \gamma \in L^2(\mathbb{R}^d)$ and $\alpha, \beta > 0$, the* **correlation function** *of the pair (g, γ) is defined as*

$$G_n(x) = \sum_{k \in \mathbb{Z}^d} \overline{g}\left(x - \frac{n}{\beta} - \alpha k\right) \gamma(x - \alpha k)$$

With this definition the *Walnut's representation* [127] can be found

Theorem 2.1.9 ([63] 6.3.2) *Let $g, \gamma \in W(\mathbb{R}^d)$ and let $\alpha, \beta > 0$. Then operator $S_{\gamma,g}$ can be represented as*

$$S_{\gamma,g} f = \beta^{-d} \sum_{n \in \mathbb{Z}^d} G_n \cdot T_{\frac{n}{\beta}} f$$

Thus in the finite, discrete case the Gabor frame matrix has only β non-zero side-diagonals and they are α-periodic. These n-th side-diagonals are just the G_n defined here. In some case, i.e. when the support of g is shorter than $\frac{1}{\beta}$, S is even diagonal. This sparse structure is important for inverting this operator to find the canonical dual window, see Section 3.1.2.

We can find another possible representation the so called *Janssen representation* [74].

Theorem 2.1.10 ([63] Theorem 7.2.1) *Let $g, \gamma \in L^2\left(\mathbb{R}^d\right)$ such that for $\alpha, \beta > 0$*

$$\sum_{k,l \in \mathbb{Z}^d} \left| \left\langle \gamma, T_{\frac{k}{\beta}} M_{\frac{l}{\alpha}} g \right\rangle \right| < \infty$$

then

$$S_{g,\gamma} = (\alpha\beta)^{-d} \sum_{l,n \in \mathbb{Z}^d} \left\langle \gamma, T_{\frac{k}{\beta}} M_{\frac{l}{\alpha}} g \right\rangle T_{\frac{k}{\beta}} M_{\frac{l}{\alpha}} =$$

$$= (\alpha\beta)^{-d} \sum_{l,n \in \mathbb{Z}^d} \left\langle \gamma, M_{\frac{l}{\alpha}} T_{\frac{k}{\beta}} g \right\rangle M_{\frac{l}{\alpha}} T_{\frac{k}{\beta}}$$

In finite dimensional spaces, this representation becomes important again and is used to define an alternative matrix norm, see Section 3.4.

This representation is important for the very useful *Wexler-Raz biorthogonality relation*, which can be used for a classification of dual windows.

Theorem 2.1.11 ([63] Theorem 7.3.1) *Assume that g and γ form Gabor Bessel sequences. Then they form dual frames if and only if*

$$(\alpha\beta)^{-d} \left\langle \gamma, M_{\frac{l}{\alpha}} T_{\frac{n}{\beta}} g \right\rangle = \delta_{l0}\delta_{n0}.$$

In the finite-dimensional case this relation provides the tools to find a dual window just by solving a system of equations.

Definition 2.1.6 *For a given lattice $\Lambda = \alpha\mathbb{Z}^d \times \beta\mathbb{Z}^d$ the lattice*

$$\Lambda^\circ = \frac{1}{\beta}\mathbb{Z}^d \times \frac{1}{\alpha}\mathbb{Z}^d$$

is called the **adjoint lattice**.

The points in the adjoint lattice can be represented by a commutation property:

Lemma 2.1.12 *Let $\Lambda = \alpha\mathbb{Z}^d \times \beta\mathbb{Z}^d$. A point $\lambda = (\tau, \omega)$ in the phase space $\mathbb{R}^d \times \mathbb{R}^d$ is in the adjoint lattice Λ° if and only if*

$$\left(T_\tau M_\omega\right)\left(T_{\alpha k} M_{\beta l}\right) = \left(T_{\alpha k} M_{\beta l}\right)\left(T_\tau M_\omega\right)$$

for all $k, l \in \mathbb{Z}^d$.

As a consequence from that the *Ron-Shen duality principle* can be shown

Theorem 2.1.13 ([23] Theorem 9.2.6) *Let $g \in L^2(\mathbb{R}^d)$ and $\alpha, \beta > 0$. Then the Gabor system $\mathcal{G}(g, \alpha, \beta)$ is a frame for $L^2(\mathbb{R}^d)$ with bounds A,B if and only if $\mathcal{G}(g, \frac{1}{\beta}, \frac{1}{\alpha})$ is a Riesz sequence with bounds abA, abB.*

This principle leads to the description of Gabor frames in dependency of their redundancy, defined as $red = \frac{1}{\alpha\beta}$.

Theorem 2.1.14

1. ([63] Corollary 7.5.1) *If $\mathcal{G}(g, \alpha, \beta)$ is a frame for $L^2(\mathbb{R}^d)$, then $red \geq 1$.*
2. ([63] Corollary 7.5.2) *The Gabor system $\mathcal{G}(g, \alpha, \beta)$ is a Riesz basis for $L^2(\mathbb{R}^d)$, if and only if it is a frame and $red = 1$.*
3. ([63] Corollary 7.5.2) *The Gabor system $\mathcal{G}(g, \alpha, \beta)$ is an ONB for $L^2(\mathbb{R}^d)$, if and only if it is a tight frame, $\|g\|_2 = 1$ and $red = 1$.*

The last theorem could be seen as a way to describe a uncertainty in the time-frequency plane, cf. [63]. Another, clearer form of uncertainty principle is the *Balian Low Theorem* :

Theorem 2.1.15 *Let $g \in L^2(\mathbb{R})$ and let $\alpha, \beta > 0$ satisfy $\alpha \cdot \beta = 1$. If the Gabor system (g, α, β) is an exact frame for $L^2(\mathbb{R})$, then*

$$\|tg(t)\|_2 \|\nu \hat{g}(\nu)\|_2 = +\infty$$

For more on this theorem, related phenomena and a comparison to the classical uncertainty principle see [11].

So Theorem 2.1.14 together with the Balian Low theorem tells us that no Gabor systems can be a Riesz bases and also have a "good" time frequency resolution.

2.1.3 Function Spaces

2.1.3.1 Amalgam Spaces

Definition 2.1.7 *A measurable function F on $\mathbb{R}^{2d}$ belongs to the* **amalgam space** $W(L_m^{p,q})$, *if the sequence of local suprema*

$$a_{k,n} = \left\| F \cdot T_{k,n} \chi_{[0,1[^{2d}} \right\|_\infty$$

belongs to $l_m^{p,q}$.

The Wiener space defined in Definition 2.1.4 is only a special case of this definition as $W = W(L^1)$.

The above definition mixes global and local behaviors. It is possible to get the same kind of space by using locally other norms than just the sup-norm. This can be seen in the investigation of a more general class of amalgams, the Wiener amalgam spaces, following [52]:

Definition 2.1.8 *A Banach space $B \in \mathcal{S}'$ is called* **localizable***, if*

1. *$(B, \|.\|_B)$ is continuously embedded in $S'(\mathbb{R}^d)$ in the weak*-topology.*
2. *B is isometrically translation invariant, i.e.*

$$\|T_x f\|_B = \|f\|_B \ \forall x \in \mathbb{R}^d, \ f \in B$$

3. *$C_c^\infty \cdot B \subseteq B$*

Definition 2.1.9 *A family $\Psi = \{T_{ak}\psi\}_{k \in \mathbb{Z}^d}$ is called a* **bounded uniform partition of unity (BUPU)***, if*

1. *$\psi \in L_c^\infty$, i.e. ψ is essentially bounded and has compact support,*
2. *$\sum_{k \in \mathbb{Z}^d} T_{ak}\psi(x) = 1$.*

The BUPU is called **smooth** *if $\psi \in C_c^\infty(\mathbb{R}^d)$.*

Definition 2.1.10 *Let $\Psi = \{T_{ak}\psi\}_{k \in \mathbb{Z}^d}$ be a smooth BUPU on $\mathbb{R}^d$. Let B a localizable Banach space. Then*

$$W(B, l^p) = \left\{ f \in B_{loc} \left| \left(\sum_{k \in \mathbb{Z}^d} \|f \cdot T_{\alpha k}\psi\|_B^p \right)^{1/p} < \infty \right. \right\}$$

is called a **Wiener amalgam space***.*

It can be shown that this definition does not depend on the choice of the BUPU. The smoothness-condition of the BUPU can be dropped for $B = L^p, C_0$ or $M = C_0'$ to get an equivalent condition, cf. [52]. So for these spaces we can use the indicator function for cubes in $\mathbb{R}^d$, for an application refer to Section 2.2.2. Clearly every amalgam space is a Wiener amalgam space $W\left(L_m^{p,q}\right) = W(L^\infty, l^{p,q})$.

For the spaces C_0 and C we use the supremum norm. In this sense $W_0 = W(C_0, l^1)$. Clearly $W(C_0, l^p) = C_0 \cap W(L^p)$ which is therefore densely embedded in $C_0 \cap L^p$, so $W(C_0, l^\infty) = C_0$.

Proposition 2.1.16 [52] *Let A, B be localizable Banach space, then*

1. $A_{loc} \subseteq B_{loc} \Longrightarrow W(A, l^p) \subseteq W(B, l^p)$ *for* $1 \leq p \leq \infty$.
2. $W(C_0, l^2) \cdot W(M, l^\infty) \subseteq W(M, l^2)$.
3. $\mathcal{F}W(\mathcal{F}L^p, l^q) \subseteq W(\mathcal{F}L^q, l^p)$ *for* $1 \leq q \leq p \leq \infty$.

For more details on Wiener amalgam spaces see e.g. [41] or [52].

Let us collect some properties of $W(M, l^p)$, which will be needed in Section 2.6:

Corollary 2.1.17 1. *For* $\mu \in W(M, l^\infty)$

$$\|\mu\|_{W(M,l^\infty)} = \sup_{k\in\mathbb{Z}^d} \sup_{\|f\|_{C_0}=1} \left| \int_{Q_1(k)} f d\mu \right|$$

2. *For* $\mu \in W(M, l^p)$, $1 \leq p < \infty$

$$\|\mu\|^p_{W(M,l^p)} = \sum_{k\in\mathbb{Z}^d} \sup_{\|f\|_{C_0}=1} \left| \int_{Q_1(k)} f d\mu \right|$$

Proof: 1.) $\mu \in M = C_0'$, $\mu : C_0 \to \mathbb{C}$

$$\|\mu\|_{Op} = \sup_{\|f\|_{C_0}=1} |\mu(f)| = \sup_{\|f\|_{C_0}=1} \left| \int_{\mathbb{R}^d} f d\mu \right|$$

$$\|\mu\|_{W(M,l^\infty)} = \sup_{k\in\mathbb{Z}^d} \left\|\mu \cdot \chi_{Q_1(k)}\right\|_M = \sup_{k\in\mathbb{Z}^d} \sup_{\|f\|_{C_0}=1} \left| \int_{Q_1(k)} f d\mu \right|$$

2.) use a similar proof as above. □

Corollary 2.1.18 *Let* $1 \leq p < \infty$.

$$W(C_0(\mathbb{R}^d), l^p) \otimes W(C_0(\mathbb{R}^d), l^p) \subseteq W(C_0(\mathbb{R}^{2d}), l^p)$$

Proof:

$$\|f \otimes g\|^p_{W(C_0,l^p)} = \sum_{k \in \mathbb{Z}^{2d}} \left\| f \otimes g \cdot \chi_{Q_1(k)} \right\|^p_{C_0} = \sum_{(k,l) \in \mathbb{Z}^{2d}} \left\| f \otimes g \cdot \chi_{Q_1(k,l)} \right\|^p_\infty$$

$$= \sum_{k \in \mathbb{Z}^d} \sum_{l \in \mathbb{Z}^d} \sup_{(x,y) \in Q_1(k,l)} |f(x) \cdot g(y)|^p \leq \sum_{k \in \mathbb{Z}} \sum_{l \in \mathbb{Z}} \sup_{x \in Q_1(k)} |f(x)|^p \cdot \sup_{y \in Q_1(l)} |g(y)|^p \leq$$

$$\leq \|f\|^p_{W(C_0,l^p)} \cdot \|g\|^p_{W(C_0,l^p)}$$

□

One can show that the inclusion is a proper one, as it can already by shown for $C(\mathbb{T})$.

2.1.3.2 Modulation Spaces

First introduced in [46] we define a special class of functions. Note the definition of v-moderate weight functions in Definition A.5.7.

Definition 2.1.11 *Fix a non-zero window $g \in \mathcal{S}$ a v-moderate weight function m on $\mathbb{R}^{2d}$ and $1 \leq p, q \leq \infty$. Then the* **modulation space** *$M^{p,q}_m(\mathbb{R}^d)$ consists of all tempered distributions $f \in \mathcal{S}'(\mathbb{R}^d)$ such that $\mathcal{V}_g f \in L^{p,q}_m(\mathbb{R}^d)$. The norm on $M^{p,q}_m(\mathbb{R}^d)$ is*

$$\|f\|_{M^{p,q}_m(\mathbb{R}^d)} = \|\mathcal{V}_g f\|_{L^{p,q}_m(\mathbb{R}^d)}$$

We will write $M^{p,q} = M^{p,q}_0$ and $M^q_m = M^{q,q}_m$. Under these circumstances it can be shown that the spaces defined above do not depend on the special choice of the non-zero test function g, as long as it is sufficiently well concentrated in the time frequency sense. Different functions define the same space with equivalent norms. Moreover these functions spaces are Banach spaces, invariant under time-frequency shifts [63].

2.1.3.3 Feichtinger's Algebra: S_0

A very special class of functions is the so called *Feichtinger's algebra* $S_0(\mathbb{R}^d) = M^{1,1}_0(\mathbb{R}^d)$.

Theorem 2.1.19 *[42] For $S_0(\mathbb{R}^d) = M^{1,1}_0(\mathbb{R}^d)$ we have the following properties:*

1. *$S_0 = W(\mathcal{F}L^1, l^1)$. Moreover it is continuously and densely embedded in $W_0 = W(C_0, l^1)$.*

2. *It is continuously and densely embedded in* $L^2(\mathbb{R}^d)$.

3. *It is continuously embedded in any (non-trivial) time-frequency homogeneous Banach space.*

4. *A function* $f \in L^2(\mathbb{R}^d)$ *is in* $S_0(\mathbb{R}^d)$ *if and only if for a non-zero* $g \in S_0(\mathbb{R}^d)$ *we have* $\mathcal{V}_g f \in L^1(\mathbb{R}^{2d})$

5. *A function* $g \in L^2(\mathbb{R}^d)$ *is in* $S_0(\mathbb{R}^d)$ *if and only if* $\mathcal{V}_g g \in L^1(\mathbb{R}^{2d})$.

6. *For* $g \in S_0(\mathbb{R}^d)$ *also* $\hat{g} \in S_0(\mathbb{R}^d)$.

For more properties and a general overview see [42]. We already know that $\mathcal{V}_g f$ is uniformly continuous for all functions in $L^2(\mathbb{R}^d)$, so in particular also for S_0-functions. Even more

Proposition 2.1.20 *[42] A function* f *is in* $S_0(\mathbb{R}^d)$ *if and only if for a non-zero* $g \in S_0(\mathbb{R}^d)$ *we have* $\mathcal{V}_g f \in W(C_0, l^1)$

Therefore we can classify S_0 by:

Corollary 2.1.21 *A function* g *is in* $S_0(\mathbb{R}^d)$*, if and only if* $\mathcal{V}_g g \in W(C_0, l^1)$.

The norm of the STFT in $W(C_0, l^1)$ can be estimated by

Lemma 2.1.22 ([42] Lemma 3.2.15) *For* $f, g \in S_0(\mathbb{R}^d)$ *we have* $\mathcal{V}_g f \in W(C_0, l^1)$ *and there exists a constant* $C > 0$ *such that*

$$\|\mathcal{V}_g f\|_{W(C_0,l^1)} \leq C \|f\|_{S_0} \|g\|_{S_0}$$

It can be shown that Gabor systems with windows from this class form Bessel sequences for all regular lattices [42]. They form a frame for parameters that are small enough:

Theorem 2.1.23 ([42] 3.6.6) *Let* $g \in S_0(\mathbb{R})$. *The Gabor system* (g, a, b) *generates a frame for* $L^2(\mathbb{R}^d)$ *for all sufficiently small* a, b.

Let us state one corollary needed in Section 2.6:

Lemma 2.1.24 *For* $g, \gamma \in S_0(\mathbb{R}^d)$ *we have* $g \otimes \gamma \in S_0(\mathbb{R}^{2d})$.

Proof: With Corollary 2.1.21 $g \in S_0(\mathbb{R}^d)$ if and only if $\mathcal{V}_g g \in W(C_0, l^1)$. Let $\lambda \in \mathbb{R}^{4d}$ with $\lambda = (\lambda_1, \lambda_2)$.

$$\mathcal{V}_{g\otimes\gamma} g \otimes \gamma(\lambda) = \langle g \otimes \gamma, \pi(\lambda) g \otimes \gamma \rangle =$$
$$= \langle g \otimes \gamma, \pi(\lambda_1) g \otimes \pi(\lambda_2)\gamma \rangle = \langle g, \pi(\lambda_1) g \rangle \cdot \overline{\langle \gamma, \pi(\lambda_2)\gamma \rangle} =$$
$$= \mathcal{V}_g g(\lambda_1) \otimes \overline{\mathcal{V}_\gamma \gamma(\lambda_2)}.$$

With Corollary 2.1.18 we get the result. □

2.2 Irregular Gabor Frames

After all these well-known facts, we will come to a part of Gabor theory, which is less explored, although in recent years there have been several publication, e.g. [53].

2.2.1 Basic Definitions

Instead of sampling the STFT at the points $(n \cdot a, m \cdot b) \in \mathbb{R}^{2d}$ for $m \in \mathbb{Z}^d, n \in \mathbb{Z}^d$, which means looking at the lattice $a\mathbb{Z} \times b\mathbb{Z} \subseteq \mathbb{R}^{2d}$, we look at a set Λ of countable but arbitrarily distributed points in the time frequency plane $\mathbb{R}^{2d}$. Such a set will still be called *lattice*.

Definition 2.2.1 *Let $g \in L^2(\mathbb{R}^d)$ be a non zero function. Let Λ be a countable subset of $\mathbb{R}^{2d}$. The set of time-frequency shifts*

$$\mathcal{G}(g, \Lambda) = \{\pi(\lambda) g : \lambda \in \Lambda\}$$

is called an (irregular) **Gabor system**. *If it is a frame, it is called (irregular)* **Gabor frame**.

The set Λ is called its **lattice**.

For $\lambda \in \mathbb{R}^{2d} = \mathbb{R}^d \times \mathbb{R}^d$ we will use t_λ and ω_λ as symbols for the first and second coordinates, such that $\pi(\lambda) g = T_{t_\lambda} M_{\nu_\lambda} g$

One of the most important results can be found in [40], which we will repeat in Theorem 2.2.5 after collecting the necessary tools.

Special cases of irregular sets have also been investigated. For example products of two irregular subsets $(\tau_k) \times (\nu_l)$, where the time frequency plane is still split into boxes, but they are of varying sizes, see [82]. *Semi-irregular lattices* where one dimension, time or frequency, is sampled regularly are investigated in [17]. In these cases classification result for frames have been formulated. In [53] several sufficient conditions for irregular Gabor frames are investigated.

2.2.2 Irregular Sampling

We will collect some basic definition for example found in [23].

Definition 2.2.2 *Let I be a countable index set and $\Lambda = (\lambda_k)_{k \in I}$ a sequence in $\mathbb{R}^d$. We say that*

1. Λ *is called* **separated** *if* $\inf_{j\neq k} |\lambda_j - \lambda_k| > o$ *for all* $j, k \in I$. *It is called* δ-**separated** *if* $|\lambda_j - \lambda_k| > \delta$ *for all* $j, k \in I$.

2. Λ *is called* **relatively separated***if it is a finite union of separated sequences.*

We use the notation Λ for such sets, as we will use the notion of separability mostly in connection with lattices.

We can give a classification for relatively separated sequences. Following [23] let us denote the half-open cube with length $h > 0$ in $\mathbb{R}^d$ centered at x with $Q_h(x)$. So

$$Q_h(x) = \prod_{i=1}^{d} [x_j - h/2, x_j + h/2|$$

where the x_j are the coordinates of x. This is clearly a disjoint cover of $\mathbb{R}^d$.

Let $\nu^+(h)$ and $\nu^-(h)$ denote the largest and smallest number of points in $\Lambda \cap Q_h(x)$, i.e.

$$\nu^+(h) = \sup_{x\in\mathbb{R}^d} \# \left(\Lambda \cap Q_h(x)\right) \ , \quad \nu^-(h) = \inf_{x\in\mathbb{R}^d} \# \left(\Lambda \cap Q_h(x)\right)$$

Definition 2.2.3 *With the above conventions define the* **upper Beurling density** $D^+(\Lambda)$ *and the* **lower Beurling density** $D^-(\Lambda)$ *by*

$$D^+(\Lambda) = \limsup_{h\to\infty} \frac{\nu^+(h)}{h^d} \ , \quad D^-(\Lambda) = \limsup_{h\to\infty} \frac{\nu^-(h)}{h^d}$$

If $D^+(\Lambda) = D^-(\Lambda)$, *then this number is called the* **uniform Beurling density** $D(\Lambda)$.

Lemma 2.2.1 ([23] Lemma 7.1.3) *For* $\Lambda \subseteq \mathbb{R}^d$ *the following properties are equivalent:*

1. $D^+(\Lambda) < \infty$

2. Λ *is relatively separated.*

3. *For some (and therefore every)* $h > 0$, *there is a natural number* $N_h(\Lambda)$ *such that*

$$\sup_{n\in\mathbb{Z}^d} \# \left(\Lambda \cap Q_h(h \cdot n)\right) < N_h(\Lambda)$$

The last point is equivalent to

$$\forall x : \sup_{n\in\mathbb{Z}^d} \#\left(\Lambda \cap Q_h(x)\right) < 2 \cdot d \cdot N_h$$

because every $Q_h(x)$ intersects only with $2 \cdot d$ boxes $Q_h(hn)$.

Let us state one result needed in the following:

Corollary 2.2.2 *Let Λ be δ-separated subset of $\mathbb{R}^d$. Let $(a_\lambda) \in l^\infty(\Lambda)$.*

$$\left\| \sum_{\lambda\in\Lambda} a_\lambda \delta_\lambda \right\|_{W(M,l^\infty)} = \sup_{k\in\mathbb{Z}^d} \sup_{\|f\|_{C_0}=1} \sum_{\lambda\in\Lambda\cap Q_1(k)} |a_\lambda| \, |f(\lambda)|$$

2. For $1 \leq p < \infty$

$$\left\| \sum_{\lambda\in\Lambda} a_\lambda \delta_\lambda \right\|^p_{W(M,l^p)} = \sum_{k\in\mathbb{Z}^d} \sup_{\|f\|_{C_0}=1} \sum_{\lambda\in\Lambda\cap Q_1(k)} |a_\lambda|^p \, |f(\lambda)|^p$$

Proof: 1.)

$$\left\| \sum_{\lambda\in\Lambda} a_\lambda \delta_\lambda \right\|_{W(M,l^\infty)} = \sup_{k\in\mathbb{Z}^d} \sup_{\|f\|_{C_0}=1} \left| \int\limits_{Q_1(k)} f(x) \sum_{\lambda\in\Lambda} a_\lambda \delta_\lambda(x) dx \right| =$$

$$= \sup_{k\in\mathbb{Z}^d} \sup_{\|f\|_{C_0}=1} \left| \sum_{\lambda\in\Lambda\cap Q_1(k)} a_\lambda f(\lambda) \right|$$

With $f \in C_0$, $\|f\|_\infty = 1$ and because Λ is δ-separated, we can extend

$$\lambda \mapsto sgn(a_\lambda) \, |f(\lambda)|$$

to a function $\tilde{f}$ in C_0 with $\left\| \tilde{f} \right\|_\infty = 1$. Therefore

$$\sup_{\|f\|_{C_0}=1} \left| \sum_{\lambda\in\Lambda\cap Q_1(k)} a_\lambda f(\lambda) \right| = \sup_{\|f\|_{C_0}=1} \sum_{\lambda\in\Lambda\cap Q_1(k)} |a_\lambda| \, |f(\lambda)|$$

2.) use an analogous proof as for item 1.) □

2.2.3 Irregular Sampling In Amalgam Spaces

The Amalgam spaces have nice sampling properties for regular sampling, see e.g. [63]. As stated in a remark there, it is also possible to extend these results to irregular sampling. The proof matches the proof of the regular case in [42]:

Proposition 2.2.3 *Let Λ be a relatively separated countable set, let $f \in W(C_0, l^p)$. Then there is a constant $C_\Lambda = N_1(\Lambda)$, such that for all $1 \leq p < \infty$*

$$\sum_{\lambda \in \Lambda} |f(\lambda)|^p \leq C_\Lambda \|f\|^p_{W(C_0,l^p)}$$

Proof:

$$\sum_{\lambda \in \Lambda} |f(\lambda)|^p = \sum_{k \in \mathbb{Z}^d} \sum_{\lambda \in k+[-\frac{1}{2},\frac{1}{2}[^d} |f(\lambda)|^p$$

By assumption Λ is relatively separated and so there is $N_1(\Lambda)$ such that with Lemma 2.2.1:

$$\sum_{\lambda \in k+[-\frac{1}{2},\frac{1}{2}[^d} |f(\lambda)|^p \leq N_1(\Lambda) \cdot \left\| f \cdot \chi_{k+[-\frac{1}{2},\frac{1}{2}[^d} \right\|^p_\infty$$

$$\Longrightarrow \sum_{\lambda \in \Lambda} |f(\lambda)|^p \leq N_1(\Lambda) \cdot \sum_{k \in \mathbb{Z}^d} \left\| f \cdot \chi_{k+[-\frac{1}{2},\frac{1}{2}[^d} \right\|^p_\infty = N_1(\Lambda) \cdot \|f\|^p_{W(C_0,l^p)}$$

□

This result can be extended to $W(C, l^\infty) = C(\mathbb{R}^d)$:

Proposition 2.2.4 *Let Λ be a relatively separated countable set, let $f \in W(C, l^\infty)$. Then*

$$\|f|_\Lambda\|_\infty \leq \|f\|_{W(C,l^\infty)}$$

Proof:

$$\|f|_\Lambda\|_\infty = \sup_{\lambda \in \Lambda} |f(\lambda)| \leq \sup_{x \in \mathbb{R}^{2d}} |f(x)| = \|f\|_\infty = \|f\|_{W(C_0,l^\infty)}$$

□

2.2.4 Irregular Bessel Sequences

In [40] Feichtinger and Gröchenig investigated atomic decomposition in the context of locally compact groups. They proved a lot of results in this very general case with deep representation theory tools. One of these results can be specialized in the irregular Gabor framework to

Theorem 2.2.5 ([40] Theorem 6.1.) *Let $g \in S_0$ non-zero. Then there exists an open set $U \subseteq \mathbb{R}^{2d}$ such that (g, Λ) forms a frame in $L^2\left(\mathbb{R}^d\right)$ for every relatively separated lattice Λ for which*

$$\bigcup_{k \in I} (\lambda_k + U) = \mathbb{R}^{2d}$$

Without using arguments from representation theory like in [40], we can still show that, similar to the regular case, for relatively separated lattice S_0-windows always form Bessel sequences:

Theorem 2.2.6 *Let $g \in S_0$ and let Λ be a relatively separated lattice $\subseteq \mathbb{R}^{2d}$. Then the system (g, Λ) forms a Bessel sequence in $L^2\left(\mathbb{R}^d\right)$, i.e. there exists a $B > 0$ such that for all $f \in L^2\left(\mathbb{R}^d\right)$*

$$\sum_{\lambda \in \Lambda} \left|\mathcal{V}_g f(\lambda)\right|^2 \leq B \cdot \|f\|^2_{L^2\left(\mathbb{R}^d\right)}$$

Proof: We will use the properties of the Gram matrix. With Theorem 1.1.37 we have to show that the Gram matrix G_g gives rise to a bounded operator. With Schur's Lemma, Lemma A.4.19, it is enough to show that for this self-adjoint matrix there is a B such that for all λ':

$$\sum_{\lambda \in \Lambda} \left|\langle g_\lambda, g_{\lambda'} \rangle\right| \leq B \iff \sum_{\lambda \in \Lambda} \left|\langle \pi(\lambda) g, \pi(\lambda') g \rangle\right| \leq B$$

As we are looking at the absolute value of the inner product we can ignore phase factors and so:

$$\iff \sum_{\lambda \in \Lambda} \left|\langle g, \pi(\lambda' - \lambda) g \rangle\right| \leq B \iff \sum_{\lambda \in \Lambda} \left|\left(\mathcal{V}_g \pi(\lambda') g\right)(-\lambda)\right| \leq B$$

By assumption $g \in S_0$ and as S_0 is isometric time-frequency shift invariant for all λ' we know that $\pi(\lambda') g \in S_0$. So $\mathcal{V}_g \pi(\lambda') g \in W(C_0, l^1)$. Therefore with Proposition 2.2.3 we know that

$$\sum_{\lambda \in \Lambda} \left|\left(\mathcal{V}_g \pi(\lambda') g\right)(\lambda)\right| \leq C_\Lambda \left\|\mathcal{V}_g \pi(\lambda') g\right\|_{W(C_0, l^p)} \leq$$

$$\overset{Lem.2.1.22}{\leq} C_\Lambda C \left\| \pi(\lambda')g \right\|_{S_0} \left\| g \right\|_{S_0} = C' \left\| g \right\|_{S_0}^2$$

□

The proof does not depend on the space $L^2\left(\mathbb{R}^d\right)$ so it can be extended to other spaces. This theorem is a generalization of Lemma 3.3. in [53].

2.2.5 Perturbation Of Irregular Gabor Frames

We will look at a way of how to measure if two lattices are similar to each other. We will start with an investigation on what happens if one point is removed from a regular overcomplete Gabor frame.

2.2.5.1 From Regular To Irregular Gabor Frames

Let us look at an example on how to get an irregular frame by taking out one element of a regular Gabor frame. Clearly for exact frames this is not a frame anymore, but for every overcomplete regular Gabor frame we will get an irregular frame.

Lemma 2.2.7 *Let (g, a, b) form an overcomplete regular Gabor frame in $L^2\left(\mathbb{R}^d\right)$. Let $\Lambda = \left\{(la, kb) \middle| (l,k) \in \mathbb{Z}^{2d}\right\}$. Let $\lambda_0 \in \Lambda$ be any time frequency point. Let $\Lambda' = \Lambda \backslash \{\lambda_0\}$. Then (g, Λ') forms an irregular Gabor frame.*

Proof: For every f we know $f = \sum\limits_{\lambda \in \Lambda} \langle f, \tilde{g}_\lambda \rangle g_\lambda$. Therefore

$$g_{\lambda_0} = \sum_{\lambda \in \Lambda} \langle g_{\lambda_0}, \tilde{g}_\lambda \rangle g_\lambda \Longrightarrow g_{\lambda_0} \cdot (1 - \langle g_{\lambda_0}, \tilde{g}_{\lambda_0} \rangle) = \sum_{\lambda \in \Lambda'} \langle g_{\lambda_0}, \tilde{g}_\lambda \rangle g_\lambda$$

Clearly

$$\langle g_{\lambda_0}, \tilde{g}_{\lambda_0} \rangle = \langle \pi(\lambda_0) g, \pi(\lambda_0) \tilde{g} \rangle = \langle g, \pi(\lambda_0)^* \pi(\lambda_0) \tilde{g} \rangle = \langle g, \tilde{g} \rangle$$

From the Wexler-Raz biorthogonality relation, Theorem 2.1.11, we know

$$(ab)^{-d} \left\langle \tilde{g}, M_{\frac{l}{\alpha}} T_{\frac{n}{\beta}} g \right\rangle = \delta_{l0} \delta_{n0}$$

And so as $a \cdot b < 1$ for inexact Gabor frames, cf. Theorem 2.1.14

$$\langle g_{\lambda_0}, \tilde{g}_{\lambda_0} \rangle = 1 \cdot (ab)^d < 1$$

Therefore $g \in \overline{span} \{g_\lambda | \lambda \in \Lambda'\}$ and (g, Λ') is complete and following Proposition 1.1.2 it must be a frame. □

2.2.5.2 The Similarity Of Lattices

For the sake of a shorter notation let us define

Definition 2.2.4 *Let Λ, Λ' be two countable sets. If they have a common index set K such that*

$$|\lambda_k - \lambda'_k| \leq \delta \quad \forall k \in K,$$

*then we say that the two sets are δ-**similar** and write $\mathfrak{s}(\Lambda, \Lambda') \leq \delta$. If this is not possible we set $\mathfrak{s}(\Lambda, \Lambda') = \infty$*

We will again use this definition mostly in connection with lattices for Gabor analysis. This means to be able to compare two lattices they must have a common index set. If $\mathfrak{s}(\Lambda, \Lambda') \leq \delta$ this means that there is a index set fulfilling the above definition.

This kind of measuring the similarity of lattice seems only to make sense in the irregular case, because in the regular case, if $|a - a'| = \delta \neq 0$ the distance between (ma, nb) and (ma', nb') will get arbitrarily large. But with an infinite index set a reordering might do the trick. It is clear that this measurement of similarity is not suitable for all questions regarding similar lattices, but at least for jitter-like question this seems to be useful.

If we have lattices with similarity δ and $\delta \to 0$ the infinity norm of the difference of elements of the Gabor systems (g, Λ) and (g, Λ') tend to zero, because we know that the time-frequency shifts are continuous:

Lemma 2.2.8 *The mapping $\lambda \mapsto \pi(\lambda)g$ from $\mathbb{R}^{2d}$ to $L^2\left(\mathbb{R}^d\right)$ is uniformly continuous for every $g \in L^2\left(\mathbb{R}^d\right)$. I.e. for $\lambda \to \lambda'$*

$$\|\pi(\lambda)g - \pi(\lambda')g\|_{L^2\left(\mathbb{R}^d\right)} \to 0$$

Proof: Let $\lambda = (\tau, \omega)$ and $\lambda' = (\tau', \omega')$, then

$$\|\pi(\lambda)g - \pi(\lambda')g\| = \|M_\omega T_\tau g - M_{\omega'} T_{\tau'} g\| \leq$$
$$\leq \|M_\omega T_\tau g - M_{\omega'} T_\tau g\| + \|M_{\omega'} T_\tau g - M_{\omega'} T_{\tau'} g\| =$$
$$= \|(M_\omega - M_{\omega'}) T_\tau g\| + \|M_{\omega'} (T_\tau - T_{\tau'}) g\| =$$
$$\overset{Lem.2.1.1}{=} \|(M_\omega - M_{\omega'}) T_\tau g\| + \|(T_\tau - T_{\tau'}) g\| \overset{Cor.2.1.2}{\to} 0 \text{ for } \lambda \to \lambda'.$$

□

We have seen in Section 1.1.12 that this is not a good measure for similarity of a frame in general. In the Gabor case we can show that this similarity is at least well suited for the continuity of Gabor multipliers, see Section 2.6. For that we need some results and definitions:

Definition 2.2.5 *For a function f on $\mathbb{R}^d$ and $\delta > 0$, the function*

$$x \mapsto osc_\delta(f)(x) = \sup_{|y| \leq \delta} |T_y f(x) - f(x)|$$

*is called the δ-**oscillation** of f.*

Lemma 2.2.9 ([2] Lemma 6.3) *Let $1 \leq p < \infty$.*

1. *$f \in W(C_0, l^p)$ implies $osc_\delta(f) \in W(C_0, l^p)$, and*

$$\|osc_\delta(f)\|_{W(C_0,l^p)} \leq \|f\|_{W(C_0,l^p)}$$

2. *For every $f \in W(C_0, l^p)$*

$$\|osc_\delta(f)\|_{W(C_0,l^p)} \to 0 \text{ for } \delta \to 0$$

Now we can formulate

Theorem 2.2.10 *Let $g \in W(C_0, l^p)$ for $1 \leq p < \infty$, let Λ be a relatively separated countable set in $\mathbb{R}^d$. Let Λ_δ be countable sets such that $\mathfrak{s}(\Lambda, \Lambda_\delta) \leq \delta$. Then*

$$\sum_{k \in K} |g(\lambda_k) - g(\lambda'_k)|^p \to 0 \text{ for } \delta \to 0$$

Proof: We know that

$$|g(\lambda_k) - g(\lambda'_k)| \leq \sup_{|y| \leq \delta} |g(\lambda_k) - g(\lambda_k + y)| = osc_\delta(g)(\lambda_k).$$

And therefore

$$\sum_{k \in K} |g(\lambda_k) - g(\lambda'_k)|^p \leq \sum_{k \in K} |osc_\delta(g)|^p (\lambda_k) \overset{Prop.2.2.3}{\leq} C_\Lambda \|osc_\delta(g)\|_{W(C_0,l^p)}.$$

With Lemma 2.2.9 (2) we know

$$\sum_{k \in K} |g(\lambda_k) - g(\lambda'_k)|^p \to 0$$

for $\delta \to 0$. □

If a set is similar enough to a δ-separated lattice, it is δ-separated as stated in the next result:

Lemma 2.2.11 *Let Λ be a δ-separated lattice. Let Λ' be a lattice with $\mathfrak{s}(\Lambda, \Lambda') \leq \delta_0 < \frac{\delta}{2}$. Then Λ' is a $(\delta - 2\delta_0)$-separated lattice.*

Let Λ be a relatively separated lattice. Then there is a Δ such that for all $\delta_0 \leq \Delta$ and for any lattice Λ' with $\mathfrak{s}(\Lambda, \Lambda') \leq \delta_0$ the lattice Λ' is relatively separated.

Proof: Let $\lambda_i', \lambda_j' \in \Lambda'$. Then

$$\begin{aligned}
\left|\lambda_i' - \lambda_j'\right| &= \left|\lambda_i' - \lambda_i + \lambda_i - \lambda_j + \lambda_j - \lambda_j'\right| = \\
&= \left|(\lambda_i - \lambda_j) - \left(\lambda_i - \lambda_i' + \lambda_j' - \lambda_j\right)\right| \geq \\
&\geq |\lambda_i - \lambda_j| - \left|\lambda_i - \lambda_i' + \lambda_j' - \lambda_j\right| \geq \\
&\geq |\lambda_i - \lambda_j| - \left(|\lambda_i - \lambda_i'| + \left|\lambda_j' - \lambda_j\right|\right) \geq \\
&\geq \delta - 2\delta_0 > 0
\end{aligned}$$

For the second part apply the proof for the finitely many δ_n-separated subsets of Λ. □

For relatively separated sets a much stronger results is possible:

Proposition 2.2.12 *Let Λ be a relatively separated countable set. Let $r > 0$. Let Λ' be a countable set with $\mathfrak{s}(\Lambda, \Lambda') < r$. Then Λ' is relatively separated.*

Proof: Lemma 2.2.1 respectively the comment following states that a set Λ is relatively separated if and only if for all $x \in \mathbb{R}^d$ for one (and therefore for all) $h > 0$, there is $N_h(\Lambda)$ such that

$$\forall x : \sup_{n \in \mathbb{Z}^d} \# \left(\Lambda \cap Q_h(x)\right) < 2 \cdot d \cdot N_h(\Lambda)$$

Let $h = 1$ and $x \in \mathbb{R}^d$. Λ is relatively separated and so there is a $N_{r+1}(\Lambda)$. As $\mathfrak{s}(\Lambda, \Lambda') < r$

$$\Lambda' \cap Q_1(x) \subseteq \Lambda \cap Q_{r+1}(x)$$

and therefore

$$\sup_{x \in \mathbb{R}^d} \# \left(\Lambda' \cap Q_1(x)\right) \leq \sup_{x \in \mathbb{R}^d} \# \left(\Lambda \cap Q_{r+1}(x)\right) <$$

$$< 2 \cdot d \cdot N_{r+1}(\Lambda) =: 2 \cdot d \cdot N_1(\Lambda')$$

□

As a direct consequence of the last Proposition and Theorem 2.2.6 we get:

Corollary 2.2.13 *Let $g \in S_0$, let Λ be an irregular relatively separated lattice. Let Λ' be another irregular lattice, such that there is a $r > 0$ with $\mathfrak{s}(\Lambda, \Lambda') < r$. Then the Gabor system (g, Λ') forms a Bessel sequences in $L^2\left(\mathbb{R}^d\right)$.*

Every regular lattice is clearly relatively separated. Therefore every irregular lattice witch is similar to a regular lattice is relatively separated. In particular every irregular lattice created by a jittering of a regular lattice fulfills this condition, independent on how big this error is.

2.3 Gabor Multipliers

2.3.1 Preliminaries

Let us state basic definitions and properties needed for the regular and irregular case of Gabor multipliers in this introduction. For clarity we will define the Gabor multiplier as

Definition 2.3.1 *Let $L^2\left(\mathbb{R}^d\right)$ be a Hilbert-space, let (g, Λ), (γ, Λ) be Gabor systems in $L^2\left(\mathbb{R}^d\right)$ that form Bessel sequences. For $m \in l^\infty(\Lambda)$ define the operator $\mathbf{G}_{m,\gamma,g} : L^2\left(\mathbb{R}^d\right) \to L^2\left(\mathbb{R}^d\right)$, the* **Gabor multiplier** *for (γ, Λ) and (g, Λ), as the operator*

$$\mathbf{G}_{m,\gamma,g}(h) = \sum_{\lambda \in \Lambda} m_\lambda \langle f, g_\lambda \rangle \gamma_\lambda$$

Let m be a bounded function on $\mathbb{R}^{2d}$, then we define

$$\mathbf{G}_{m,\gamma,g}(h) = \sum_{\lambda \in \Lambda} m(\lambda) \langle f, g_\lambda \rangle \gamma_\lambda$$

This is, of course, just the Bessel multiplier, see Definition 1.3.2, for a Gabor system. Again to be able to define this operator, the two sequences have to share their index set, here the lattice. This definition can be extended to other spaces, where Gabor systems can be defined. We will stick to the $L^2\left(\mathbb{R}^d\right)$ case. Note that this definition does not make any assumption on the regularity of the underlying discrete set Λ, as long as the Gabor systems form Bessel sequences.

Every Gabor multiplier $\mathbf{M}$ can be expressed as linear combination of the projections

$$P_{g,\gamma,\lambda} = \pi(\lambda) g \otimes \overline{\pi(\lambda)\gamma}.$$

Following Lemma A.4.24 this means

$$P_{g,\gamma,\lambda} = \pi(\lambda) \circ (g \otimes \overline{\gamma}) \circ \pi(\lambda)^*.$$

In this context it seems very natural to define

Definition 2.3.2 *Let $T, A, B : L^2\left(\mathbb{R}^d\right) \to L^2\left(\mathbb{R}^d\right)$ then define*

$$(A \otimes B)\, T = A \circ T \circ B$$

For $\lambda \in \mathbb{R}^{2d}$ let

$$\pi_2(\lambda)T = (\pi(\lambda) \otimes \pi^*(\lambda))\, T$$

$\pi_2(\lambda)$ is a unitary representation from the phase space $\mathbb{R}^{2d}$ to $\mathcal{HS}$, see [43]. Therefore $\pi_2^*(\lambda) = \pi_2(-\lambda)$.

If we want to look at the Gram Matrix of this Hilbert Schmidt Projection we can use A.4.39 and see:

Corollary 2.3.1 *The entries of the Gram matrix of the projection P_λ in $\mathcal{HS}$ are*

$$G^{(\mathcal{HS})}_{\lambda,\lambda'} = \langle P_{g,\gamma,\lambda}, P_{g,\gamma,\lambda'}\rangle = \mathcal{V}_g(g)\,(\lambda' - \lambda) \cdot \overline{\mathcal{V}}_\gamma(\gamma)\,(\lambda' - \lambda)$$

For $g = \gamma$ we get

$$G_{\lambda\lambda'} = \left|\mathcal{V}_g(g)\,(\lambda' - \lambda)\right|^2$$

Proof:

$$\langle P_\lambda, P_{\lambda'}\rangle_{\mathcal{HS}} = \langle g_\lambda \otimes \overline{\gamma}_\lambda, g_{\lambda'} \otimes \overline{\gamma}_{\lambda'}\rangle = \langle \pi(\lambda)g, \pi(\lambda')g\rangle \cdot \overline{\langle \pi(\lambda)\gamma, \pi(\lambda')\gamma\rangle} =$$

$$\overset{Cor.2.1.3}{=} \omega(\tau) \cdot \omega'(-\tau) \cdot \langle g, \pi(\lambda' - \lambda)g\rangle \cdot \overline{\omega(\tau) \cdot \omega'(-\tau) \cdot \langle \gamma, \pi(\lambda' - \lambda)\gamma\rangle} =$$

$$= \langle g, \pi(\lambda' - \lambda)g\rangle \cdot \overline{\langle \gamma, \pi(\lambda' - \lambda)\gamma\rangle} = \mathcal{V}_g(g)(\lambda' - \lambda) \cdot \overline{\mathcal{V}_\gamma}(\gamma)(\lambda' - \lambda)$$

□

2.3.2 Pseudodifferential Operator

For the discussion of Gabor multiplier it is important to look at the connection of operators and time-frequency analysis, in a study of pseudo-differential operators. This section is based on [63] and [43].

The Fourier transformation and partial differentiation are connected, so the theory of partial differential equations can be seen as study of operators, which can be written as

Definition 2.3.3 *Let* $\sigma \in L^2\left(\mathbb{R}^{2d}\right)$. *Then the operator* $K_\sigma : L^2\left(\mathbb{R}^d\right) \to L^2\left(\mathbb{R}^d\right)$ *defined by*

$$K_\sigma f(x) = \int\limits_{\mathbb{R}^d} \sigma(x,\omega)\hat{f}(\omega)e^{2\pi i x\cdot\omega}d\omega \text{ for } f \in L^2\left(\mathbb{R}^d\right)$$

is called the **pseudodifferential operator (PDO)** *with* **Kohn-Nirenberg symbol** σ.

The mapping $\sigma \mapsto K_\sigma$ *is called the* **Kohn-Nirenberg correspondence**.

This definition can be extended to other measurable function spaces or tempered distributions on $\mathbb{R}^{2d}$. For example in [43] it is used for the so-called Gelfand triple S_0, L^2, S_0'.

Definition 2.3.4 *For two functions* $f, g \in L^2\left(\mathbb{R}^d\right)$ *we call*

$$u_{f,g}(x,\omega) = e^{-2\pi i x\cdot\omega}\overline{\hat{f}(\omega)}g(x) = \overline{\omega(x)}\left(g \otimes \hat{f}\right)(x,\omega)$$

the **Rihaczek distribution**.

If $\sigma(x,\omega) = m(x)$ the resulting operator is just the multiplication operator with $m(x)$, if $\sigma(x,\omega) = \mu(\omega)$ the resulting operator is the convolution operator with $\hat{h} = \mu$. If $\hat{f}(\omega) = \int\limits_{\mathbb{R}^d} f(y)e^{-2\pi i y\omega}dy$ is substituted in the definition of the PDO, we receive [63]

$$K_\sigma f(x) = \int\limits_{\mathbb{R}^d} \underbrace{\int\limits_{\mathbb{R}^d} \sigma(x,\omega)e^{2\pi i(x-y)\omega}d\omega}_{=:h(x,y)}\, f(y)dy$$

Thus the PDO corresponds to integral operators, cf. Theorem 2.3.2.

For $g, f \in \mathcal{S}(\mathbb{R}^d)$ clearly $u_{f,g} \in \mathcal{S}(\mathbb{R}^{2d})$. Let $\sigma \in \mathcal{S}'(\mathbb{R}^{2d})$ so the following expression is well defined

$$\langle \sigma, u_{f,g}\rangle = \int\limits_{\mathbb{R}^d}\int\limits_{\mathbb{R}^d} \sigma(x,\omega)e^{2\pi i x\cdot\omega}\hat{f}(\omega)\overline{g(x)}d\omega dx =$$

$$= \int\limits_{\mathbb{R}^d} K_\sigma f(x)\overline{g(x)}dx = \langle K_\sigma f, g\rangle$$

Therefore the Kohn-Nirenberg correspondence can be extended to symbols in $\mathcal{S}'(\mathbb{R}^d)$ or $S_0'(\mathbb{R}^d)$, refer to [43]. We will use the results in [43] only for the Hilbert space setting $L^2\left(\mathbb{R}^d\right)$:

Theorem 2.3.2 ([43] Theorem 7.5.1) *The Kohn-Nirenberg correspondence is an invertible operator from the integral operator kernels to the Kohn-Nirenberg symbols.*

$$\sigma\left(K\right)(x,\omega) = \int\limits_{\mathbb{R}^d} \kappa(K)(x,x-y)e^{-2\pi i\omega y}dy$$

$$\kappa(K)(x,y) = \int\limits_{\mathbb{R}^d} \sigma(K)(x,\omega)e^{2\pi i\omega\cdot(x-y)}d\omega$$

It is a unitary isomorphism $\sigma : \mathcal{HS} \to L^2\left(\mathbb{R}^d\right)$ *which implies that*

$$\langle S,T\rangle_{\mathcal{HS}} = \langle \kappa(S),\kappa(T)\rangle_{L^2\left(\mathbb{R}^d\right)} = \langle \sigma(S),\sigma(T)\rangle_{L^2\left(\mathbb{R}^d\right)}$$

With the following lemma it becomes clear, why the mapping $\pi_2(\lambda)$ is also called a *time-frequency shift of operators*, cf. e.g. [47]

Lemma 2.3.3 ([43] Lemma 7.5.3) *The action of* $\pi_2(\lambda)$ *on* $K \in \mathcal{HS}$ *corresponds to a translation of the symbol:*

$$\sigma(\pi_2(\lambda)K) = T_\lambda\sigma\left(K\right)$$

So especially for the rank one operators:

$$\sigma(P_\lambda) = T_\lambda\sigma\left(P_0\right)$$

Let $\mathcal{F}_2$ be the Fourier transformation in the second variable for $F(x,y)$. Let $\mathcal{T}_a$ be the coordinate transformation $\left(\mathcal{T}_aF\right)(x,y) = F(x,y-x)$. Then we can write

$$\kappa(K)(x,y) = \mathcal{T}_a\mathcal{F}_2\sigma(K)$$

Using that, we want to find yet another way to describe a Hilbert-Schmidt operator. Clearly

$$\mathcal{F}_2\sigma = \mathcal{F}_1^{-1}\mathcal{F}_1\mathcal{F}_2\sigma = \mathcal{F}_1^{-1}\hat{\sigma}$$

and therefore

$$\kappa(K)(x,y) = \mathcal{F}_2\sigma(x,y-x) = \mathcal{F}_1^{-1}\hat{\sigma}(x,y-x) = \int\limits_{\mathbb{R}^d} \hat{\sigma}(\eta,y-x)e^{2\pi i\eta\cdot x}d\eta$$

$$K_\sigma f(x) = \int\limits_{\mathbb{R}^d}\int\limits_{\mathbb{R}^d} \hat{\sigma}(\eta,y-x)e^{2\pi i\eta\cdot x}f(y)d\eta dy =$$

$$= \int_{\mathbb{R}^d} \int_{\mathbb{R}^d} \hat{\sigma}(\eta, u) e^{2\pi i \eta \cdot (x)} f(x+u) d\eta du =$$

$$= \int_{\mathbb{R}^d} \int_{\mathbb{R}^d} \hat{\sigma}(\eta, u) \left(M_\eta T_{-u} f \right)(x) d\eta du$$

And so the PDO K_σ can also be represented as superposition of time-frequency shifts

$$K_\sigma = \int \int_{\mathbb{R}^{2d}} \hat{\sigma}(\eta, -u) M_\eta T_u du d\eta$$

where the operator-valued integral is understood "in a strong way", i.e.

$$\left(\int O(x) dx \right)(f)(y) = \int \left(O(x) f \right)(y) dx.$$

With this motivation we define

Definition 2.3.5 *The* **spreading function** *of a linear operator $K \in \mathcal{HS}$ with kernel $\kappa(K)$ is defined as*

$$\eta(K)(t, \nu) = \int_{\mathbb{R}^d} \kappa(K)(x, x-t) e^{-2\pi i \eta \cdot x} dx$$

Theorem 2.3.4 ([43] Theorem 7.6.3.) *The assignment of the spreading function $\eta(K)$ to a linear operator $K \in \mathcal{HS}$ is an invertible mapping.*

$$\kappa(K)(x, y) = \int_{\mathbb{R}^d} \eta(K)(x-y, \nu) e^{2\pi i \nu \cdot x} d\nu$$

It is a unitary isomorphism $\sigma : \mathcal{HS} \to L^2\left(\mathbb{R}^d\right)$ which implies that

$$\langle S, T \rangle_{\mathcal{HS}} = \langle \eta(S), \eta(T) \rangle_{L^2\left(\mathbb{R}^d\right)}$$

We will not use the spreading function in the following sections extensively. We have nevertheless repeated the basic definition and properties of the spreading function, because it gives a different way to view these operators: They are represented as a superposition of time-frequency shifts. This is easier to interpret in some applications. This representation is connected to the Janssen representation and matrix, refer to Section 3.1.2.4. For more on the spreading functions see e.g. [55].

2.4 Regular Gabor Multipliers

As regular lattices are obviously relatively separated, according to Theorem 2.2.6 every S_0-atom forms a Gabor Bessel sequence. Therefore the Gabor multiplier is a well-defined operator on $L^2(\mathbb{R}^d)$ for all bounded symbols.

Regular Gabor multipliers have been investigated to some extent. In [47] the equivalent statements to Theorem 1.3.13 and Lemma 1.3.29, among others, have been proved for regular Gabor multipliers. The investigation of how the multiplier depends on the sequences and the symbols motivated Section 1.3.6, which we are going to refine in Section 2.6. In [34] Gabor multipliers have been investigated with special focus on application in music. Some results for irregular Gabor multipliers in Section 2.5 are just generalizations from results there.

For regular Gabor multipliers with $g = \gamma$ the $\mathcal{HS}$ Gram matrix of the sequence P_λ is

$$(G_{\mathcal{HS}})_{k,l} = \left|\mathcal{V}_g \gamma \left(\lambda_k - \lambda_l\right)\right|^2$$

and is therefore a circulant matrix. So an equivalent condition for being invertible can be found [34] by using this condition with the Fourier-transformation on Λ. Compare this to Section 3.4.1.3, which deals with circulant matrices in the finite-dimensional case. We refer also to [12], where the question of well-balanced Gabor systems are further investigated. There it is shown that for regular Gabor frames the Bessel sequences P_λ in $\mathcal{HS}$ are either Riesz bases or have no further structure, i.e. they can not form a frame sequences without forming a Riesz sequence. For a related property refer to Section 2.5.4.2. In all these references it can be seen that the big 'advantage' of the regular case is the group structure of the lattice. So e.g. it can be shown [51] for windows in S_0 that the dual atom continuously depends on the lattice parameters. The lack of this useful structure gave rise to the investigation of multipliers for the general frame case in Section 1.3.

2.4.1 Spline-Type Spaces

Following [23] we will call a sequence of $T_{\lambda_k} g$ for a fixed $g \in \mathcal{H}$ a *sequence of translates*. As special case when these elements form a Riesz sequence, we call the closed span of these elements a *Spline-type spaces*. With the Kohn-Nirenberg symbol we get a connection between the sequences (P_λ) and Spline-type space following Lemma 2.3.3 . This connection was investigated in [45] and [49]. One of the main results will be extended to the irregular case in Theorem 2.5.6.

2.5 Irregular Gabor Multiplier

One of the main reasons to investigate frame multipliers in the general case was to specialize the results to the case of irregular Gabor systems. As a lot of tools is lost by dropping the group structure, for this work it was decided to investigate an even more general case. As the Gabor multipliers are just Bessel multipliers in the sense of Section 1.3 all the results there are valid in this context.

2.5.1 Basic Properties

Let us just repeat the main theorem as an example and specialize it to relatively separated lattices, windows in S_0 and continuous symbols.

Theorem 2.5.1 *Let $g, \gamma \in S_0$ and let Λ be a relatively separated lattice $\subseteq \mathbb{R}^{2d}$. Let*

$$\mathbf{G} = \mathbf{G}_{m,g,\gamma} = \sum_{\lambda} m(\lambda) g_\lambda \otimes \gamma_\lambda$$

be the Gabor multiplier with symbol $m \in C(\mathbb{R}^{2d})$, then

1. *Let $m \in W(C, l^\infty)$, then $\mathbf{G}$ is a well defined bounded operator with $\|\mathbf{G}\|_{Op} \leq C \cdot \|m\|_\infty$.*
2. *$\mathbf{G}^*_{m,f_k,g_k} = \mathbf{G}^*_{\overline{m},g_k,f_k}$. Therefore if m is real-valued and $g = \gamma$, $\mathbf{G}$ is self-adjoint.*
3. *If $m \in C_0(\mathbb{R}^{2d})$, then $\mathbf{G}$ is compact.*
4. *If $m \in W(C_0, l^1)$, then $\mathbf{G}$ is a trace class operator with $\|\mathbf{G}\|_{trace} \leq C \|m\|_{W(C_0,l^1)}$, and $tr(\mathbf{G}) = \langle g, \gamma \rangle \cdot \sum\limits_{\lambda} m_\lambda$.*
5. *If $m \in W(C_0, l^2)$, then $\mathbf{G}$ is a Hilbert Schmidt operator with $\|\mathbf{G}\|_{\mathcal{HS}} \leq C \|m\|_{W(C_0,l^2)}$.*

Proof: From Section 2.2.3 we know that for $m \in W(C_0, l^p)$ we have $\|m(\lambda)\|_p \leq C \cdot \|m\|_{W(C_0,l^p)}$. The rest is a direct consequence of Theorem 1.3.13.

For the trace formulas we know from Theorem 1.3.13

$$tr(\mathbf{G}) = \sum_{\lambda} m_\lambda \langle \gamma_\lambda, g_\lambda \rangle = \sum_{\lambda} m_\lambda \langle \gamma, \pi^*(\lambda)\pi(\lambda) g \rangle = \langle \gamma, g \rangle \sum_{\lambda} m_\lambda.$$

□

In the rest of this section we show many results, that are just generalizations to the irregular case of results in [34], [47] and [45].

2.5.2 The Kohn-Nirenberg Symbol

Proposition 2.5.2 *Let (g, γ) be an irregular Gabor system that forms a Bessel sequence.. Let $P_\lambda = \pi(\lambda) g \otimes \overline{\pi(\lambda)\gamma}$. Then*

1. $\sigma(\mathbf{G}_{m,\gamma,g}) = \sum_\lambda m_\lambda \cdot T_\lambda \sigma(P_0) =: m *_\Lambda \sigma(P_0)$

2. $\sigma(P_0) = u_{\gamma,g}$

Proof:

$$\sigma\left(\sum_\lambda m_\lambda P_\lambda\right) = \sum_\lambda m_\lambda \sigma\left(P_\lambda\right) =$$

$$= \sum_\lambda m_\lambda \sigma\left(\pi_2(\lambda) P_0\right) \overset{Lem.2.3.3}{=} \sum_\lambda m_\lambda T_\lambda \sigma\left(P_0\right)$$

From Lemma 1.3.15 we know that

$$\kappa\left(\mathbf{G}_{m,g,\gamma}\right) = \sum_\lambda m_\lambda g_\lambda \otimes \overline{\gamma}_\lambda$$

and so especially

$$\kappa\left(g \otimes \overline{\gamma}\right) = g \otimes \overline{\gamma}$$

Therefore

$$\sigma\left(P_0\right)(x, \omega) = \int\limits_{\mathbb{R}^d} g(x) \cdot \overline{\gamma}(x-y) e^{-2\pi i \omega y} dy =$$

$$= g(x) \cdot \int\limits_{\mathbb{R}^d} \overline{\gamma}(x-y) e^{-2\pi i \omega y} dy = g(x) \cdot \int\limits_{\mathbb{R}^d} \overline{\gamma}(u) e^{-2\pi i \omega (x-u)} du =$$

$$= g(x) e^{-2\pi i \omega x} \cdot \int\limits_{\mathbb{R}^d} \overline{\gamma}(u) e^{2\pi i \omega u} du = e^{-2\pi i \omega x} \cdot g(x) \cdot \int\limits_{\mathbb{R}^d} \overline{\gamma}(u) \overline{e^{-2\pi i \omega u} du} =$$

$$= e^{-2\pi i \omega x} \cdot g(x) \cdot \hat{\overline{\gamma}}(\omega) = u_{\gamma,g}(x, \omega)$$

□

2.5.3 Well-balanced Gabor systems

Lemma 2.5.3 *Let (g, γ, Λ) form a well-balanced pair of Gabor Bessel sequence. Then there is an operator $Q_0 \in \overline{span}\{P_\lambda\}$ such that $Q_\lambda = \pi(\lambda) Q_0$ forms a biorthogonal sequence for $P_\lambda = g_\lambda \otimes g_\lambda$.*

Proof: We know that P_λ is a Riesz sequence, so there is a biorthogonal sequence $(Q'_\lambda) \subseteq \overline{span}\{P_\lambda\}$. Set $Q_0 = Q'_0$. Then

$$\langle P_\lambda, \pi(\lambda') Q_0 \rangle_{\mathcal{HS}} = \langle \pi(\lambda) P_0, \pi(\lambda') Q_0 \rangle_{\mathcal{HS}} =$$

$$= \langle \pi(\lambda - \lambda') P_0, Q_o \rangle_{\mathcal{HS}} = \delta_{\lambda,\lambda'}$$

□

Different to the regular case we can not conclude that the $Q_\lambda = \pi_2(\lambda) Q_0$ are in $\overline{span}\{P_\lambda\}$, because (from the Kohn-Nirenberg point of view) in general irregular frames of translates are not translation invariant.

2.5.4 Hilbert Schmidt Operators

In this section we will investigate the case, in which the Gabor multipliers are in $\mathcal{HS} = \mathcal{HS}(L^2(\mathbb{R}^d))$.

2.5.4.1 Frames In $\mathcal{HS}$

As η, σ and κ are unitary isomorphisms, they transfer the properties of one space exactly into the others. For clarity we will state some results explicitly.

Theorem 2.5.4 *Let $T_k \in \mathcal{HS}$ for $k \in K$. Then*

1. $S \in \overline{span}\{T_k\} \Leftrightarrow \sigma(S) \in \overline{span}\{\sigma(T_k)\}$
2. $S \in \overline{span}\{T_k\} \Leftrightarrow \eta(S) \in \overline{span}\{\eta(T_k)\}$
3. $S \in \overline{span}\{T_k\} \Leftrightarrow \kappa(S) \in \overline{span}\{\kappa(T_k)\}$

Proof: Let $T \in \overline{span}\{T_k\}$, then for all $\epsilon > 0$ there exists a finite set I and coefficients c_{k_i} such that

$$\left\| T - \sum_{i \in I} c_{k_i} T_{k_i} \right\|_{\mathcal{HS}} < \epsilon$$

With Theorem 2.3.2 this is equal to

$$\left\| \sigma(T) - \sum_{i \in I} c_{k_i} \sigma T_{k_i} \right\|_{L^2\left(\mathbb{R}^d\right)} < \epsilon$$

So one direction of point 1 is proved, for the other direction the proof has just to be swapped.

For η and κ, literally only the symbols have to be exchanged in the above proof. □

Theorem 2.5.5 *Let $T_k \in \mathcal{HS}$ for $k \in K$*

1. *If and only if the sequence (T_k) forms a Bessel sequence, frame sequence, frame, Riesz sequence, Riesz basis, orthogonal sequences or orthonormal basis within the Hilbert space $\mathcal{HS}$ then $(\sigma(T_k))$ does in the phase space $L^2\left(\mathbb{R}^{2d}\right)$, too.*

2. *If and only if the sequence (T_k) forms a Bessel sequence, frame sequence, frame, Riesz sequence, Riesz basis, orthogonal sequences or orthonormal basis in $\mathcal{HS}$ then $(\eta(T_k))$ does in $L^2\left(\mathbb{R}^{2d}\right)$, too.*

3. *If and only if the sequence (T_k) forms a Bessel sequence, frame sequence, frame, Riesz sequence, Riesz basis, orthogonal sequences or orthonormal basis in $\mathcal{HS}$ then $(\kappa(T_k))$ does in $L^2\left(\mathbb{R}^{2d}\right)$, too.*

Proof:

$$A \cdot \|S\|_{\mathcal{HS}} \leq \sum_k |\langle S, T_k \rangle_{\mathcal{HS}}| \leq B \cdot \|S\|_{\mathcal{HS}} \Longleftrightarrow$$

$$A \cdot \|\sigma(S)\|_{\mathcal{HS}} \leq \sum_k |\langle \sigma(S), \sigma(T_k) \rangle_{\mathcal{HS}}| \leq B \cdot \|\sigma(S)\|_{\mathcal{HS}}$$

So the Bessel sequence and frame property is preserved.

For Riesz sequences and orthogonal sequences we just have to note, that the sequence (T_k) has the same Gram matrix as $\sigma(T_k)$ because of Theorem 2.3.2.

With Theorem 2.5.4 we obtain the missing results for frame sequences, Riesz bases and ONBs.

For η and κ, literally only the symbols have to be exchanged in the above proof. □

The last result can be applied to $P_\lambda = \gamma_\lambda \otimes \overline{g}_\lambda$. Using Proposition 2.5.2 we can show the corresponding result of [45] Theorem 5.20 for the irregular case. Remember the definition of well-balanced systems in Definition 1.3.4.

Proposition 2.5.6 *Let $g, \gamma \in L^2\left(\mathbb{R}^d\right)$ form a well-balanced pair of Gabor Bessel sequences. Let $T \in \mathcal{HS}$. Then T is a Gabor multiplier for these Gabor systems if and only if $\sigma(T)$ belongs to $\overline{span}\left\{T_\lambda u_{\gamma,g}\right\} = \left\{\sum_\lambda c_\lambda u_{\gamma,g} \,|(c_\lambda) \in l^2(\Lambda)\right\}$.*

Proof: With Proposition 2.5.2 one direction is clear.

On the other hand let $\sigma(T) \in \overline{span}\left\{T_\lambda u_{\gamma,g}\right\}$. Because the Gabor systems are well-balanced, we know that $\overline{span}\left\{T_\lambda u_{\gamma,g}\right\} = \left\{\sum_\lambda c_\lambda T_\lambda u_{\gamma,g}\right\}$. Therefore there exists a $c \in l^2(\Lambda)$ such that

$$\sigma(T) = \sum_\lambda c_\lambda T_\lambda u_{\gamma,g} = \sum_\lambda c_\lambda \sigma(P_\lambda)$$

$$\Longleftrightarrow T = \sum_\lambda c_\lambda P_\lambda$$

□

For the proof it is enough that the projections P_λ form a frame sequence.

2.5.4.2 The Sequence P_λ In $\mathcal{HS}$

In [23] the following result has been proved for frames of irregular translates:

Theorem 2.5.7 ([23] Proposition 7.4.2.) *Assume that $(\lambda_k)_{k\in K}$ is a sequence for which $\lambda_k \neq \lambda_i$ for $k \neq i$. If $g \in L^2\left(\mathbb{R}^d\right)$, $g \neq 0$ then the functions $(T_{\lambda_k} g)$ are linearly independent.*

Applying this result to Gabor multipliers with the Kohn-Nirenberg correspondence we immediately get:

Theorem 2.5.8 *Assume that $(\lambda_k)_{k\in K}$ is a sequence for which $\lambda_k \neq \lambda_i$ for $k \neq i$. Let $g, \gamma \in L^2\left(\mathbb{R}^d\right)$, $g, \gamma \neq 0$ then the sequence $(P_{g,\gamma,\lambda})$ in $\mathcal{HS}(L^2\left(\mathbb{R}^d\right))$ is linearly independent.*

Proof: If $g, \gamma \in L^2\left(\mathbb{R}^d\right)$ then as a special case of Theorem 1.3.13 $P_0 \in \mathcal{HS}(L^2\left(\mathbb{R}^d\right))$ and therefore $\sigma(P_0) \in L^2\left(\mathbb{R}^{2d}\right)$.

Suppose there is a finite index set I such that

$$\sum c_{\lambda_i} P_{\lambda_i} = 0 \Rightarrow$$

$$0 = \sigma\left(\sum c_{\lambda_i} P_{\lambda_i}\right) = \sum c_{\lambda_i} \sigma\left(P_{\lambda_i}\right) = \sum c_{\lambda_i} T_{\lambda_i} \sigma\left(P_0\right)$$

Therefore the $c_{\lambda_i} = 0$. □

In the general case this does not necessarily mean that the sequence forms a Riesz sequence, as Riesz sequences have to be minimal, cf. Theorem 1.1.32. For finite-dimensional case Theorem 2.5.7 can not be easily adapted. This case certainly has to be investigated further.

2.6 Changing The Ingredients For Irregular Gabor Multipliers

For the irregular case we can show a similar property as stated in [47] Theorem 5.6.2 for the regular case:

Theorem 2.6.1 *Let $g, \gamma \in W(C, l^\infty)$, let Λ be a relatively separated irregular lattice, such that (g, Λ) (γ, Λ) form a pair of Bessel sequences for $L^2\left(\mathbb{R}^d\right)$. For $m \in W(C_0, l^1)$ let $\mathbf{G} = \mathbf{G}_{m,g,\gamma}$. Then the trace-class operator $\mathbf{G}$ continuously depends on m, g_k, f_k and Λ, in the following sense:*

Let $(\gamma_k^{(l)})$, $(g_k^{(l)})$ be Bessel sequences indexed by $l \in \mathbb{N}$ with $g^{(l)} \to g$, $\gamma^{(l)} \to \gamma$ in $W(C_0, l^\infty)$. Let $\Lambda^{(\delta)}$ be lattices such that $\mathfrak{s}\left(\Lambda, \Lambda^{(\delta)}\right) \leq \delta$. Let $m^{(l)} \to m$ in $W(C, l^1)$. Then

$$\mathbf{G}_{m^{(l)},g,\gamma,\Lambda^{(\delta)}} \to \mathbf{G}_{m,g,\gamma,\Lambda} \text{ in the trace class}$$

for $\delta \to 0$, $N \to \infty$.

Proof: As $\mathfrak{s}\left(\Lambda^{(l)}, \Lambda\right) \neq \infty$, there is a common index set I for all these lattices. Let in the following be $\lambda'_i \in \Lambda^{(\delta)}$ and $\lambda_i \in \Lambda$.

With Lemma 2.1.1 we know that

$$\left\| \pi(\lambda'_i) g^{(l)} - \pi(\lambda_i) g \right\|_{L^2\left(\mathbb{R}^d\right)} \leq$$

$$\leq \left\| \pi(\lambda'_i) g^{(l)} - \pi(\lambda'_i) g \right\|_{L^2\left(\mathbb{R}^d\right)} + \left\| \pi(\lambda'_i) g - \pi(\lambda_i) g \right\|_{L^2\left(\mathbb{R}^d\right)} =$$

$$= \left\| g^{(l)} - g \right\|_{L^2\left(\mathbb{R}^d\right)} + \left\| \pi(\lambda'_i) g - \pi(\lambda_i) g \right\|_{L^2\left(\mathbb{R}^d\right)}$$

And so, because of Corollary 2.2.8, for every $\epsilon_2 > 0$ there is N_1 and a δ_2 such that for all $\delta < \delta_2$ and $l > N_1$

$$\left\| \pi(\lambda'_i) g^{(l)} - \pi(\lambda_i) g \right\|_{L^2\left(\mathbb{R}^d\right)} < \epsilon_2$$

As $g^{(l)} \to g$, there is a N_2 such that there is a common bound $\mathbf{B}$, such that

$$\left\| \pi(\lambda'_i) g^{(l)} \right\|_{L^2\left(\mathbb{R}^d\right)} < \mathbf{B}$$

Clearly this properties are also valid for $\gamma^{(l)}$.

With Theorem 1.3.18 and the remark following it, we now have to show that $m^{(l)}(\lambda_i) \to m(\lambda_i)$ in l^1 to prove the result.

$$\sum_{i \in I} \left| m^{(l)}(\lambda'_i) - m(\lambda_i) \right| \leq \sum_{i \in I} \left| m^{(l)}(\lambda'_i) - m(\lambda'_i) \right| + \sum_{i \in I} \left| m(\lambda'_i) - m(\lambda_i) \right| \leq$$

$$\stackrel{Prop.2.2.3}{\leq} N_1(\Lambda^{(\delta)}) \cdot \left\| m^{(l)} - m \right\|_{W(C_0,l^1)} + \sum_{i \in I} \left| m(\lambda'_i) - m(\lambda_i) \right| \leq$$

There is a δ_3 such that for $\delta < \delta_3$ there exists a C with $N_1(\Lambda^{(\delta)}) < C$. With Theorem 2.2.10 for every $\epsilon_5 > 0$ there is a δ_4 and N_2 such that

$$\sum_{i \in I} \left| m^{(l)}(\lambda'_i) - m(\lambda_i) \right| \leq \epsilon_5$$

So we have shown that for every $\epsilon > 0$ there is a $N = \max\{N_1, N_2\}$ and $\delta_0 = \min\{\delta_1, \ldots, \delta_4\}$ such that for all $l > N$ and $\delta < \delta_0$

$$\left\| \mathbf{G}_{m^{(l)},g,\gamma,\Lambda^{(\delta)}} - \mathbf{G}_{m,g,\gamma,\Lambda} \right\|_{trace} < \epsilon$$

□

The equivalent statement to Theorem 3.3. in [49] can be shown by adapting the proof there to the irregular case:

Theorem 2.6.2 *Let $g, \gamma \in S_0(\mathbb{R}^d)$, let Λ be a δ-separated irregular lattice. For $m \in W(C_0, l^2)$ let $\mathbf{G} = \mathbf{G}_{m,g,\gamma}$. Then the Hilbert Schmidt operator $\mathbf{G}$ continuously depends on m, g_k, f_k and Λ, in the following sense: Let $(\gamma_k^{(l)})$, $(g_k^{(l)})$ be sequences indexed by $l \in \mathbb{N}$ with $g^{(l)} \to g$, $\gamma^{(l)} \to \gamma$ in $S_0(\mathbb{R}^d)$. Let $\Lambda^{(\delta)}$ be lattices such that $\mathfrak{s}\left(\Lambda, \Lambda^{(\delta)}\right) \leq \delta$. Let $m^{(l)} \to m$ in $W(C_0, l^2)$. Then*

$$\mathbf{G}_{m^{(l)},g,\gamma,\Lambda^{(\delta)}} \to \mathbf{G}_{m,g,\gamma,\Lambda} \text{ in } \mathcal{HS}$$

for $\delta \to 0$, $N \to \infty$.

Proof: As $\mathfrak{s}\left(\Lambda^{(l)}, \Lambda\right) \neq \infty$, there is a common index set I for all these lattices. With Corollary 2.2.13 we know that there is a δ_1 such that $(g^{(l)}, \Lambda^{(\delta)})$ and $(\gamma^{(l)}, \Lambda^{(\delta)})$ form a Bessel sequence. Let $\lambda'_i \in \Lambda^{(\delta)}$ and $\lambda_i \in \Lambda$.

As $g^{(l)} \to g$, there is a N_1 such that for all $l > N_1$ there is a common bound $\mathbf{B_1}$

$$\left\| \pi(\lambda'_i) g^{(l)} \right\|_{S_0} < \mathbf{B_1}$$

Let $\mu_\delta = \sum_{i \in I} \delta_{\lambda'_i}$. There is a δ_2 such that μ_δ is uniformly bounded in $W(M, l^\infty)$ for $\delta < \delta_2$ as

$$\|\mu_\delta\|_{W(M,l^\infty)} \overset{Cor.2.2.2}{=} \sup_{k \in \mathbb{Z}^d} \sup_{\|f\|_\infty = 1} \sum_{\lambda'_i \in Q_1(k)} |f(\lambda'_i)| \leq$$

$$\leq \sup_{k \in \mathbb{Z}^d} \sup_{\|f\|_\infty = 1} N_1(\Lambda^{(\delta)}) \cdot \|f\|_\infty = N_1(\Lambda^{(\delta)})$$

where the notations $Q_1(k)$ and $N_1(\Lambda)$ are those used in Section 2.2.2. There is a δ_3 such that there is a $\mathbf{B_2}$ with $N_1(\Lambda^{(\delta)}) < \mathbf{B_2}$. As $C_c \subseteq C_0$ dense, with Corollary 2.1.16 $W(C_c, l^2) \subseteq W(C_0, l^2)$ and so there is for every $\epsilon_1 > 0$ there is a $\phi \in W(C_c, l^2)$ with values in $[0, 1]$ such that

$$\|m \cdot \phi - m\|_{W(C_0,l^2)} < \epsilon_1$$

From Corollary 2.1.16 we have $W(C_0, l^2) \cdot W(M, l^\infty) \subseteq W(M, l^2)$ and so

$$\|m \cdot \mu_\delta - m \cdot \phi \cdot \mu_\delta\|_{W(M,l^2)} \leq B_2 \|\mu_\delta\|_{W(M,l^\infty)} \cdot \|m - m \cdot \phi\|_{W(C_0,l^2)} \leq \mathbf{B_2} \cdot \epsilon_1$$

Therefore

$$\left\| \sum_{i \in I} (1 - \phi(\lambda_i)) \cdot m(\lambda_i) \cdot \delta_{\lambda_i} \right\|_{W(M,l^2)} \leq \mathbf{B_2} \cdot \epsilon_1$$

Let δ_4 be fixed, e.g. $\delta_4 = 1/2$. Then there is a finite index set $I_1 \subseteq I$ such that for all $\delta < \delta_4$ the set $\{\lambda_i | \lambda_i \in \Lambda^{(\delta)}, i \notin I_1\} \cap supp(\phi)$ is empty. Following Corollary 2.1.17 we get for $\lambda' \in \Lambda^{(\delta)}$

$$\left\| \sum_{i \notin I_1} (1 - \phi(\lambda'_i)) \cdot m(\lambda'_i) \cdot \delta_{\lambda'_i} \right\|_{W(M,l^2)} \leq \left\| \sum_{i \in I} (1 - \phi(\lambda'_i)) \cdot m(\lambda'_i) \cdot \delta_{\lambda'_i} \right\|_{W(M,l^2)} \leq \mathbf{B_2} \cdot \epsilon_1$$

As all λ which are not in $\Lambda_2^{(\delta)}$ are not in $supp(\phi)$ we have

$$\left\| \sum_{i \notin I_1} (1 - \phi(\lambda'_i)) \cdot m(\lambda'_i) \cdot \delta_{\lambda'_i} \right\|_{W(M,l^2)} = \left\| \sum_{i \notin I_1} m(\lambda'_i) \cdot \delta_{\lambda'_i} \right\|_{W(M,l^2)} \leq \mathbf{B_2} \cdot \epsilon_1$$

Let

$$G_{l,\delta} = \sum_{\lambda \in \Lambda^{(\delta)}} m^{(l)}(\lambda) g_\lambda^{(l)} \otimes \gamma_\lambda^{(l)} \text{ and } G_0 = \sum_{\lambda \in \Lambda} m(\lambda) g_\lambda \otimes \gamma$$

With Proposition 2.2.3 and Lemma 2.2.11 there is a δ_5 such that for all $\delta < \delta_5$ and $\lambda' \in \Lambda^{(\delta)}$ we obtain $m^{(l)}(\lambda') \in l^2$ and so these operators are in $\mathcal{HS}$.

$$\|G_{l,\delta} - G_0\|_{\mathcal{HS}} = \|\sigma(G_{l,\delta}) - \sigma(G_0)\|_{L^2\left(\mathbb{R}^d\right)}$$

From Proposition 2.5.2 we know that

$$\sigma(G_{l,\delta}) = \sum_{i \in I} m^{(l)}(\lambda_i') \cdot T_{\lambda_i'} u_{g^{(l)},\gamma^{(l)}}$$

With $\gamma^{(l)} \in S_0(\mathbb{R}^d)$ due to Theorem 2.1.19 also $\hat{\gamma^{(l)}} \in S_0$ as With $g^{(l)}, \gamma^{(l)} \in S_0(\mathbb{R}^d)$ we have $u_{g^{(l)},\gamma^{(l)}}(t,\omega) = \overline{\omega(t)} \cdot g^{(l)}(t) \cdot \hat{\gamma^{(l)}}(\omega) \in S_0(\mathbb{R}^{2d})$, cf. Corollary 2.1.24.

From Corollary 2.1.16 we also know that $W(M, l^\infty) * W(C_0, l^1) \subseteq W(C_0, l^2)$.

$$\left\| \sum_{i \notin I_1} m(\lambda_i') T_{\lambda_i'} u_{g^{(l)},\gamma^{(l)}} \right\|_{W(C_0,l^2)} = \left\| \left(\sum_{i \notin I_1} m(\lambda_i') \delta_{\lambda_i'} \right) * u_{g^{(l)},\gamma^{(l)}} \right\|_{W(C_0,l^2)} \leq$$

$$\leq \mathbf{B_3} \cdot \left\| \sum_{i \notin I_1} m(\lambda_i') \delta_{\lambda_i'} \right\|_{W(M,l^\infty)} \cdot \left\| u_{g^{(l)},\gamma^{(l)}} \right\|_{W(C_0,l^1)} \leq$$

$$\leq \mathbf{B_3} \cdot \mathbf{B_2} \epsilon_1 \cdot \mathbf{B_1}^2$$

for all $\delta < \min\{\delta_1, \ldots, \delta_4\}$ and $l > N_1$.

So we know for $\lambda_i' \in \Lambda^{(\delta)}$

$$\|\sigma(G_{l,\delta}) - \sigma(G_0)\|_{L^2\left(\mathbb{R}^d\right)} = \left\| \sum_{i \in I} \left(m^{(l)}(\lambda_i') \delta_{\lambda_i'} * u_{g^{(l)},\gamma^{(l)}} - m(\lambda_i) \delta_{\lambda_i} * u_{g,\gamma} \right) \right\|_{L^2\left(\mathbb{R}^d\right)} \leq$$

$$\leq \left\| \sum_{i \notin I_1} m^{(l)}(\lambda_i') \delta_{\lambda_i'} * u_{g^{(l)},\gamma^{(l)}} \right\|_{L^2\left(\mathbb{R}^d\right)} + \left\| \sum_{i \notin I_1} m(\lambda_i) \delta_{\lambda_i} * u_{g,\gamma} \right\|_{L^2\left(\mathbb{R}^d\right)} +$$

$$+ \left\| \sum_{i \in I_1} \left(m^{(l)}(\lambda_i') \delta_{\lambda_i'} * u_{g^{(l)},\gamma^{(l)}} - m(\lambda_i) \delta_{\lambda_i} * u_{g,\gamma} \right) \right\|_{L^2\left(\mathbb{R}^d\right)} \leq$$

$$\leq 2 \cdot \mathbf{B_3} \cdot \mathbf{B_2} \epsilon_1 \cdot \mathbf{B_1}^2 + \left\| \sum_{i \in I_1} m^{(l)}(\lambda_i') \delta_{\lambda_i'} * u_{g^{(l)},\gamma^{(l)}} - m(\lambda_i) \delta_{\lambda_i} * u_{g,\gamma} \right\|_{L^2\left(\mathbb{R}^d\right)}$$

Finally let $P^{(l)} = g^{(l)} \otimes \gamma^{(l)}$ and $P^0 = g \otimes \gamma$. Then

$$\left\| \sum_{i \in I_1} m^{(l)}(\lambda_i') \delta_{\lambda_i'} * u_{g^{(l)},\gamma^{(l)}} - m(\lambda_i)\delta_{\lambda_i} * u_{g,\gamma} \right\|_{L^2\left(\mathbb{R}^d\right)} =$$

$$= \left\| \sum_{i \in I_1} m^{(l)}(\lambda_i') \pi_2(\lambda_i') P^{(l)} - m\lambda_i \pi_2(\lambda_i) P^0 \right\|_{\mathcal{HS}} \leq$$

$$\left\| \sum_{i \in I_1} m^{(l)}(\lambda_i') \pi_2(\lambda_i') P^{(l)} - m(\lambda_i') \pi_2(\lambda_i') P^{(l)} \right\|_{\mathcal{HS}} + \tag{2.1}$$

$$\left\| \sum_{i \in I_1} m(\lambda_i') \pi_2(\lambda_i') P^{(l)} - m(\lambda_i) \pi_2(\lambda_i') P^{(l)} \right\|_{\mathcal{HS}} + \tag{2.2}$$

$$\left\| \sum_{i \in I_1} m(\lambda_i) \pi_2(\lambda_i') P^{(l)} - m(\lambda_i) \pi_2(\lambda_i) P^{(l)} \right\|_{\mathcal{HS}} + \tag{2.3}$$

$$\left\| \sum_{i \in I_1} m(\lambda_i) \pi_2(\lambda_i) P^{(l)} - m(\lambda_i) \pi_2(\lambda_i) P^0 \right\|_{\mathcal{HS}} \tag{2.4}$$

$$(2.1) \overset{Th.1.3.13}{\leq} \left\| m^{(l)}(\lambda_i') - m(\lambda_i') \right\|_2 \mathbf{B_1}^2 \overset{Prop.2.2.3}{\leq} N_1(\Lambda^{(\delta)}) \cdot \left\| m^{(l)} - m \right\|_{W(C_0,l^2)} \mathbf{B_1}^2$$

For all $\epsilon_2 > 0$ there is a N_2 such that for all $\delta < \min\{\delta_1, \dots, \delta_5\}$ and $l > \max\{N_1, N_2\}$

$$(2.1) \leq \epsilon_2 \mathbf{B_2} \cdot \mathbf{B_1}^2$$

$$(2.2) \overset{Th.1.3.13}{\leq} \left\| m(\lambda_i') - m(\lambda_i') \right\|_2 \mathbf{B_1}^2$$

And so with Theorem 2.2.10 there is for all $\epsilon_3 > 0$ a δ_6 such that for all $\delta < \min\{\delta_1, \dots, \delta_6\}$ and $l > N_1$

$$(2.2) \leq \epsilon_3 \cdot \mathbf{B_1}^2$$

$$(2.3) \leq \# I_1 \left\| m(\lambda_i) \right\|_\infty \left\| \pi_2(\lambda_i') P^{(l)} - \pi_2(\lambda_i) P^0 \right\|_{\mathcal{HS}} \leq$$
$$\leq \# I_1 \left\| m \right\|_{W(C_0,l^\infty)} \left\| \pi_2(\lambda_i' - \lambda_i) P^{(l)} \right\|_{\mathcal{HS}}$$

So for every $\epsilon_4 > 0$ there is a δ_7 such that for all $\delta < \min\{\delta_1, \dots, \delta_5, \delta_7\}$ and $l > N_1$

$$(2.3) \leq \# I_1 \left\| m \right\|_{W(C_0,l^\infty)} \cdot \epsilon_4$$

Equally

$$(2.4) \leq \#I_1 \|m\|_{W(C_0,l^\infty)} \left\|\pi_2(\lambda_i)P^{(l)} - \pi_2(\lambda_i)P^0\right\|_{\mathcal{H}S} =$$

$$= \#I_1 \|m\|_{W(C_0,l^\infty)} \left\|P^{(l)} - P^0\right\|_{\mathcal{H}S}$$

$$\left\|P^{(l)} - P^0\right\|_{\mathcal{H}S} = \left\|g^{(l)} \otimes \gamma^{(l)} - g \otimes \gamma\right\|_{\mathcal{H}S} \leq$$
$$\leq \left\|\left(g^{(l)} - g\right) \otimes \gamma\right\|_{\mathcal{H}S} + \left\|g \otimes \left(\gamma^{(l)} - \gamma\right)\right\|_{\mathcal{H}S}$$

Therefore for every $\epsilon_5 > 0$ there is a N_3 such that for all $\delta < \min\{\delta_1, \ldots, \delta_5\}$ and $l > \max\{N_1, N_3\}$

$$\left\|P^{(l)} - P^0\right\|_{\mathcal{H}S} \leq 2 \cdot \epsilon_5 \cdot \mathbf{B_1}$$

and so

$$(2.4) \leq \#I_1 \|m\|_{W(C_0,l^\infty)} 2 \cdot \epsilon_5 \cdot \mathbf{B_1}$$

Overall for all $\epsilon > 0$ there is a $N = \max\{N_1, ..., N_3\}$ and $\delta_0 = \min\{\delta_1, ..., \delta_7\}$ such that for all $l > N$ and $\delta < \delta_0$ we have

$$\|G_{l,\delta} - G_0\|_{\mathcal{H}S} < \epsilon$$

□

2.7 The Gabor Multiplier in $\mathbb{C}^L$

2.7.1 The Kohn-Nirenberg Symbol And Spreading Function In $\mathbb{C}^L$

For a Kohn-Nirenberg algorithm we can use the formula

$$\sigma\left(K\right)\left(x,\omega\right) = \int\limits_{\mathbb{R}^d} \kappa(K)(x, x-y)e^{-2\pi i\omega y}dy$$

and translate it to the discrete setting by noting that the matrix representation is the kernel of an operator and the integral reduces to a sum. So for the $L \times L$ matrix M we receive

$$\sigma_{m,n} = \sum_{l=0}^{L-1} M_{m,m-l} e^{\frac{-2\pi inl}{L}}$$

The corresponding algorithm can be found in Section B.2.2. In the test file in Section B.2.2.1 the algorithm was validated by checking the identity

$$\sigma(P_0) = u_{\gamma,g}$$

from Proposition 2.5.2.

For the spreading function we show in Section 3.2.1 that the time-frequency shifts are an ONB in $\mathcal{HS}(\mathbb{C}^n)$. So every matrix can be represented as $H = \sum \eta_\lambda \pi\lambda$ with $\eta_\lambda = \langle H, M_k T_l \rangle_{\mathcal{HS}}$. Following results in the next chapter, cf. Section 3.2,

$$\langle T, M_k T_l \rangle_{\mathcal{HS}} = \sum_{i,j=0}^{L-1} T_{i,j} \cdot (M_k T_l)_{i,j} = \sum_{i,j=0}^{L-1} T_{i,j} \cdot \omega_L^{i \cdot k} \delta_{i,j+l} =$$

$$= \sum_{i=0}^{L-1} T_{i,i-l} \cdot \omega_L^{i \cdot k}$$

This was implemented for example by W. Kozek in `spread.m` [54].

2.7.2 The Irregular Gabor System

Section B.2.1 includes a MATLAB-algorithm, that calculates the synthesis matrix for an irregular Gabor system. This algorithm uses row vectors and matrix-multiplication from the right. The single elements of the Gabor system are the rows of this matrix. The elements of the Gabor system are centered at points in the $L \times L$ time-frequency given by a $n \times n$ matrix `xpo`. Every point in this matrix, which is not zero, is used for one element of the Gabor system.

A test file is included as well.

2.7.3 Approximation of Hilbert Schmidt operators by irregular Gabor Multipliers

In [50] an algorithm was presented that approximated a matrix by a regular Gabor matrix. The regularity of the Gabor system was used extensively to implement a numerically efficient algorithm. This is of course no option for irregular Gabor multipliers. But different to the case for general frames, Section 1.3.9.3, the $\mathcal{HS}$ Gram matrix can be calculated using the efficient FFT algorithm by using Corollary 2.3.1. For an algorithm refer to Section B.3.1. A test file is included.

The algorithm for an approximation of any matrix by irregular Gabor multipliers can be found in Section B.3.2.

2.7.3.1 Comparison To The Regular Version

The algorithm developed in this work for irregular Gabor multipliers is compared with the one from [50], which uses regular lattices and the same window for analysis and synthesis. Although slower the irregular approximation gives the same result, cf. Figure 2.4, when used with a regular lattice. We will give one example for a random matrix T, which was approximated.

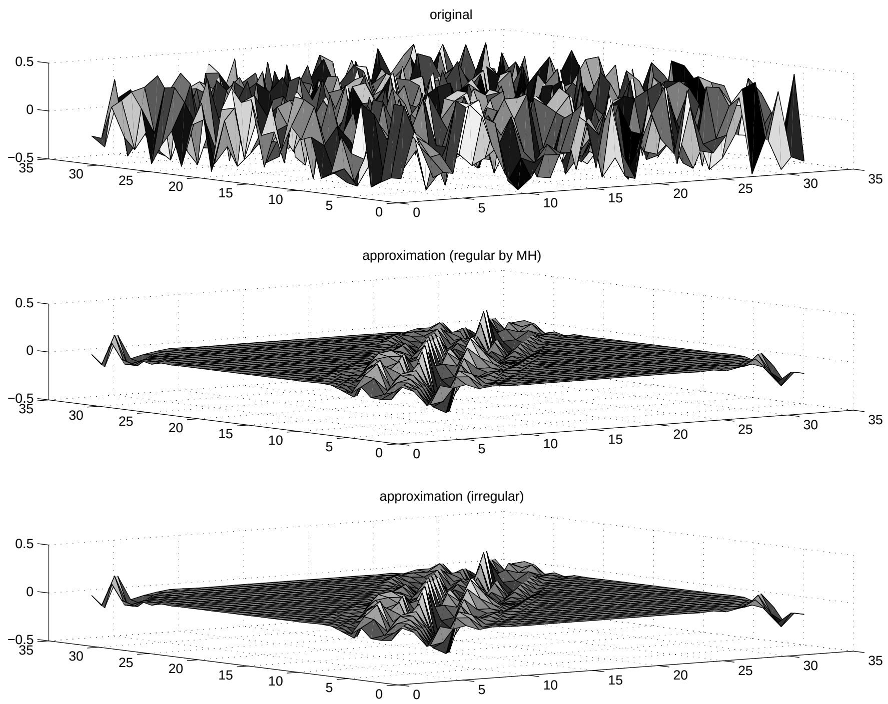

Figure 2.4: *(Top:) The original matrix: a random matrix. (Middle:) The approximation by the regular version of [50]. (Bottom:) Approximation by the irregular version u.*

For this experiment the parameters have been chosen for good graphical properties. The dimension of the signal space is $n = 32$, the lattice parameters are $a = 4, b = 4$ and a Gaussian window has been chosen for analysis and synthesis. For this example the difference of the two approximations in the Operator-norm is $8.83744 \cdot 10^{-15}$.

In Figure 2.4 we see that components "away" from the diagonal can not be approximated well by Gabor multipliers. This fact is further illustrated in Figure 2.5, where a translation matrix is approximated using the same parameters.

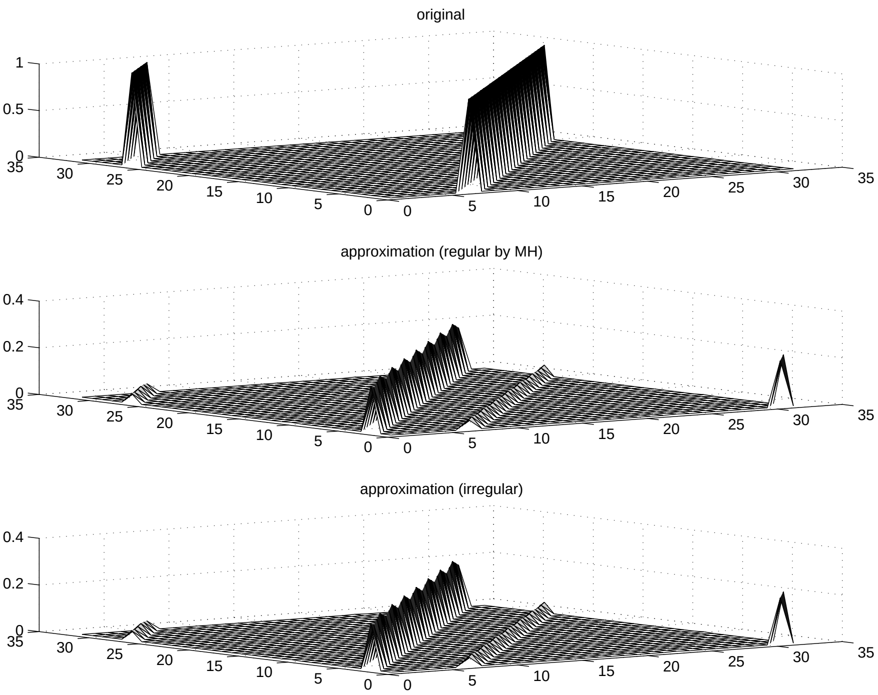

Figure 2.5: *(Top:) The original matrix: a translation matrix. (Middle:) The approximation by the regular algorithm (Bottom:) The approximation by the irregular algorithm.*

2.7.3.2 Approximation Of The Identity

As an example we will look at the approximation of the identity with $n = 32$ as in Section 1.3.9.3. The lattice points are chosen randomly for a redundancy of 2. For the synthesis atom again a Gaussian is chosen, for the analysis atom a normalized zero-padded hamming window, see Figure 2.6 and 2.7.

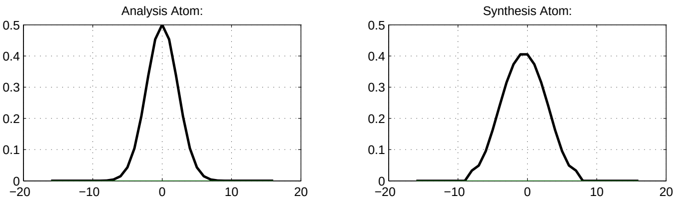

Figure 2.6: *(Left:) The analysis window (Gaussian). (Right:) The synthesis window (Hamming).*

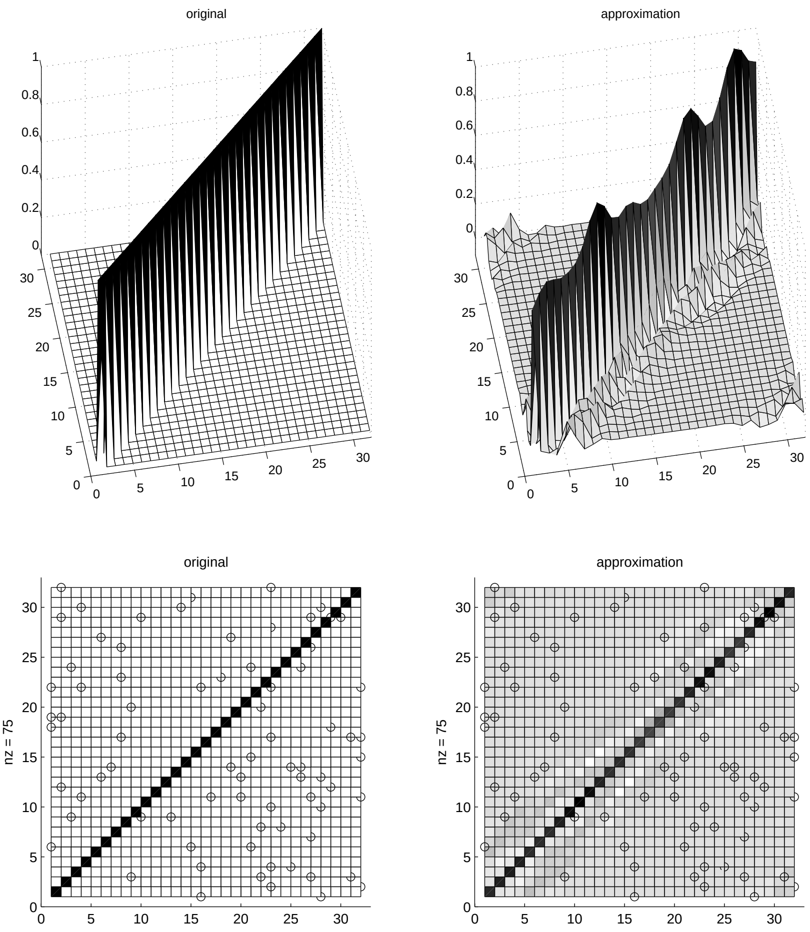

Figure 2.7: *(Top Left:) The original matrix, the identity (3D plot). (Top Right:) The approximation by Gabor multipliers (3D plot). (Bottom Left:) The original matrix (With Lattice Points) . (Bottom Right:) The approximation by Gabor multipliers. (With Lattice Points)*

Chapter 3

Discrete Finite Gabor Analysis

The goal in the next chapter, Chapter 4, is to find an algorithm for the modification of an audio signal. As we have already seen the Short-time Fourier transformation (STFT) is a valuable tool for displaying the energy distribution of a signal f over the time-frequency plane. For a number of applications (for example in audio processing like time stretching without changing the frequency content [32], more complex modifications like psychoacoustical masking see Chapter 4, or other applications see [31, 72, 132]), the time domain signal needs to be reconstructed using the time-frequency domain coefficients. The dual problem of atomic decomposition is also needed in applications. In it a given signal is built as a series using a time-frequency shifted window as building blocks (see e.g. [5]). Application and algorithms always work with finite dimensional data. So we work with finite discrete signals and have also to ask questions of the numerical efficiency. Therefore some properties of the general theory are 'translated' to this special case.

In this chapter we will first look at a summary of known results for the finite, discrete Gabor theory and some 'translation' from general frame theory, from Section 1.2. We will collect well-known results in Section 3.1 to find out that the Gabor frame matrix has a very special structure. It is a block-matrix. Because the Gabor frame matrix has this special structure we will investigate matrices with a special block structure and investigate the Fourier transformation defined for matrices in Section 3.2. We will refer to [122] a lot in this chapter, so we will look at some details at the work 'Numerical Algorithms For Discrete Gabor Expansions' by Thomas Strohmer in [44]. We will correct some minor errors there. Finally in Section 3.4 we will look at a new method for inverting the Gabor frame method with Double Preconditioning already published in [9]. The main question in this section is how can we find an effective analysis-synthesis system. To find an answer we will investigate a method for finding an approximate dual by using preconditioning matrices.

A big part of this chapter, most notable Section 3.2.6, 3.2.6.1 and 3.4 has been published as [9].

3.1 Preliminaries

3.1.1 Computational Linear Algebra : Iterative Algorithms

We want to solve the equation

$$Ax = b. \tag{3.1}$$

Direct algorithms like e.g. the Gauss elimination are known to be numerically very expensive and also instable. Often other methods, like iterative algorithms, are used. We define

Definition 3.1.1 *An* **iteration** *for $\mathbb{C}^n$ is a function $\Phi : \mathbb{C}^n \times \mathbb{C}^n \mapsto \mathbb{C}^n$ with $x_{m+1} = \Phi(x_m, b)$.*

It is called **linear** *if $\Phi(x, b) = Mx + Nb$ for two matrices $M, N \in \mathbb{C}^{n\times n}$.*

It is called **consistent with the matrix** *A, if for all $b \in \mathbb{C}^n$ $A^{-1}b$ is a fixed point of $x \mapsto \Phi(x, b)$.*

It is called **convergent**, *if for all $b \in \mathbb{C}^n$ the iteration converges for all starting vectors $x_0 \in \mathbb{C}^n$ to the same limit $\hat{x} = \lim\limits_{m\to\infty} x_m$.*

Proposition 3.1.1 ([87] Chapter 4)

1. *A linear iteration is consistent with the matrix A if and only if*

$$M = I - NA.$$

2. *A linear iteration Φ is convergent, if and only if the spectral radius of the iteration matrix M is smaller than one, $\rho(M) < 1$.*

3. *Let Φ be a linear iteration, that is convergent and consistent with A. Then $x = \lim\limits_{i\to\infty} x_m$ for*

$$x_m = \Phi(x_{m-1}, b) \text{ for } m = 1, 2, \ldots$$

and for every starting value x_0 the equation $Ax = b$ holds.

3.1.1.1 Splitting Method

Every matrix A can be written as $A = B + (A - B)$ for any matrix B. This leads to the system

$$Bx = (B - A)\, x + b$$

which is equivalent to Equation 3.1.

If B is invertible, then the following system is equivalent:

$$x = B^{-1}(B - A)\, x + B^{-1} b$$

Definition 3.1.2 *The linear iteration*

$$x_{m+1} = \Phi(x_m, b) = M x_m + N b \text{ for } m = 1, 2, \ldots$$

with

$$M = B^{-1}(B - A) \text{ and } N = B^{-1}$$

is called the **splitting method** .

Proposition 3.1.2 ([87] Section 4.1)

1. *If B is invertible, the splitting method is consistent with A.*
2. *If $\|B^{-1}(B - A)\|_{Op} < 1$, the splitting method is convergent.*

3.1.1.2 Jacobi Algorithm

Let A be a $n \times n$ matrix with non-zero diagonals. The *Jacobi algorithm* is the splitting method used with with $D = \mathfrak{diag}(S)$.

$$x_{m+1} = D^{-1}(D - S)x_m + D^{-1} b$$

Theorem 3.1.3 ([87] Section 4.1.1) *Let A be an invertible $n \times n$ matrix, if*

1. $\max\limits_{i=1,\ldots,n} \sum\limits_{\substack{j=1,\ldots,n \\ j \neq i}} \frac{|a_{i,j}|}{|a_{i,i}|} < 1$,
2. $\max\limits_{j=1,\ldots,n} \sum\limits_{\substack{i=1,\ldots,n \\ j \neq i}} \frac{|a_{i,j}|}{|a_{i,i}|} < 1$ *or*
3. $\sum\limits_{\substack{i,j=1,\ldots,n \\ j \neq i}} \left(\frac{|a_{i,j}|}{|a_{i,i}|}\right)^2 < 1$

then the Jacobi algorithm converges for every starting value x_0 and every b to a solution of Equation 3.1:

$$x_m \to A^{-1} b \text{ for } m \to \infty$$

3.1.1.3 Preconditioning

A way to improve the numerical efficiency of an iterative algorithm to solve a linear system of equations is preconditioning. Instead of solving the linear system of equations $Ax = b$ the the system $PAx = Pb$ is solved for a properly chosen preconditioning matrix P. To this end, the matrix P should be chosen according to the following criteria:

1. P should be constructed within few operations, e.g. $O(n \log n)$.

2. P should be able to be stored in an efficient way

3. $\kappa(PA) \ll \kappa(A)$.

Here $\kappa(S) = \|S^{-1}\|_{Op} \cdot \|S\|_{Op}$ is the *condition-number* of the matrix, which measures the stability of a linear equation system. The first two criteria are intended to keep the number of operations and memory requirements below those of the non-preconditioned system. The third criterion is intended to control the numeric stability of the system. A sufficient condition for the third criterion is a clustered spectrum, as $\kappa(A) = \frac{\sigma_n}{\sigma_1}$ where σ_n and σ_1 are the largest and smallest singular values, respectively. A clustered spectrum also yields a faster convergence (see [4, 84]).

Using the splitting method in Section 3.1.1.1 is equivalent to solving the preconditioned equation $B^{-1}Ax = B^{-1}b$ with the Neumann algorithm, see Proposition A.4.9.

3.1.1.4 Remarks On The Operator Norm

We have seen above, that the use of the operator norm is the natural way to measure the quality of an approximation as it satisfies $\|A \cdot x\| \leq \|A\|_{Op} \|x\|$, for all $x \in \mathbb{C}^n$. Another important application of this norm is the condition-number for invertible matrices. The problem with the operator norm is that its computation is very costly. For example, it can be shown that the operator norm of a self-adjoint operator is equal to its largest eigenvalue, and the numerical calculation of the eigenvalues of an operator is numerically very expensive, even if elaborated methods, see e.g. [123], are used.

3.1.2 Discrete Gabor Expansions

In this whole chapter we will consider the Hilbert space $\mathbb{C}^L$, and that the lattice parameter a and b are factors of L (i.e., there exist integers $\tilde{a}$ and $\tilde{b}$

such that $a \cdot \tilde{a} = L$ and $b \cdot \tilde{b} = L$). We regard the vectors $x \in \mathbb{C}^L$ as L-periodic, and therefore we interpret the *Kronecker symbol* in a consistent way:

$$\delta_{i,j} = \begin{cases} 1 & i = j \bmod L \\ 0 & otherwise \end{cases}$$

We will also always regard matrices to be periodic in the columns and rows to make the notation shorter.

In this case, the modulation and time shift operators are discretized, i.e.,

$$T_l x = (x_{L-l}, x_{L-l+1}, \ldots, x_0, x_1, \ldots, x_{L-l-1})$$

and

$$M_k x = \left(x_0 \cdot \omega_n^0, x_1 \cdot \omega_L^{1 \cdot k}, \ldots, x_{L-1} \cdot \omega_L^{(L-1)k} \right) \text{ with } \omega_L = e^{\frac{2\pi i}{L}}$$

Note that the translation is acting in a cyclic way.

Therefore we will consider the Gabor system

$$\mathcal{G}(g, \alpha, \beta) = \left\{ M_{bn} T_{ak} g : k = 0, \ldots, \tilde{a}; n = 0, \ldots, \tilde{b} \right\}$$

Notice that this is equivalent to sampling with sampling period T and setting $\omega = \frac{k}{LT}$ and $\tau = l \cdot T$.) The *redundancy* of $\mathcal{G}(g, \alpha, \beta)$ is then $red = L/(ab)$.

In the discrete, finite-dimensional case, it is well known, see e.g. [122] that the Gabor frame operator has a very special structure. The matrix S is zero except in every $\tilde{b}$-th side-diagonals. These side-diagonals are also a-periodic. This can be seen by using the *Walnut representation* of the operator, analogue to Theorem 2.1.9, which can be expressed in the finite discrete case in the following way:

Corollary 3.1.4

$$S_{\gamma, g_{m,n}} = \begin{cases} \tilde{b} \cdot \sum\limits_{k=0}^{\tilde{a}-1} g\,(m - ak)\,\overline{\gamma}\,(n - ak) & |m - n| = 0 \ mod \ \tilde{b} \\ 0 & otherwise \end{cases}$$

and therefore

$$S_{\gamma, g} f_m = \tilde{b} \cdot \sum_{p=0}^{b-1} f(m - p\tilde{b}) \underbrace{\sum_{k=0}^{\tilde{a}-1} g(m - ak) \overline{\gamma}(m - ak - p\tilde{b})}_{G_p(m)}$$

This can easily be proved directly or be found in [103]. Compare this result to the continuous case and note the similarities to the correlation function $G_p(m)$.

This means that the matrix has a sparse nature, only every p-th side-diagonal, G_p, is non-zero and they are a-periodic. The relevant entries can be seen in the following figure:

$$\left(\begin{array}{cccc|ccc|cccc|cc}
s_{0,0} & & & & & & & s_{0,M} & & & & \cdots & \\
 & s_{1,1} & & & & & & & s_{1,M} & & & & \\
 & & \ddots & & & & & & & \ddots & & & \\
 & & & s_{a-1,a-1} & & & & & & & s_{0+a-1,M+a-1} & & \\
\hline
 & & & & s_{0,0} & & & & & & & s_{0.M} & \\
 & & & & & \ddots & & & & & & & \ddots \\
 & & & & & & s_{M-1,M-1} & & & & & & \\
\hline
s_{M,0} & & & & & & & s_{M,M} & & & & & \\
 & \ddots & & & & & & & \ddots & & & &
\end{array}\right)$$

Clearly it can be represented by

$$\left(S_{\gamma,g}\right)_{m,n} = \tilde{b} \cdot Ш_{\tilde{b}}(m-n) \cdot \sum_{k=0}^{\tilde{a}-1} g\left(m-ak\right) \overline{\gamma}\left(n-ak\right). \tag{3.2}$$

This means that S can be represented as a block matrix, either as a block circulant matrix or as a block diagonal matrix. The set of matrices with any of these structures form a matrix algebra, cf. Section 3.2. Therefore the set of matrices which has both structures at the same time forms a matrix algebra as well. We call them *Gabor-type* [104] or *Walnut* matrices, for more on these special matrices see Section 3.2.

3.1.2.1 The Non-Zero Block Matrix

From the remarks stated above we know that we can find a smaller matrix which describes S uniquely. To this end, the first a entries of the non-zero side-diagonals are considered. In [104] the following block-matrix was introduced:

Definition 3.1.3 *Let (g, a, b), (γ, a, b) be Gabor frames, and $S = S_{\gamma,g}$ be their associated frame operator. Let B be the $b \times a$ matrix given by*

$$B_{i,j} = S_{i \cdot \tilde{b}+j,j}$$

We call B a **non-zero block matrix** *or the* **auto-correlation matrix** *of the Gabor system $\mathcal{G}(g, a, b)$.*

So this matrix looks like this:

$$B = \begin{pmatrix} S_{0,0} & S_{1,1} & \dots & S_{a-1,a-1} \\ S_{\tilde{b},0} & S_{\tilde{b}+1,1} & \dots & S_{\tilde{b}+a-1,a-1} \\ \vdots & & & \vdots \\ S_{(b-1)\cdot\tilde{b},0} & S_{(b-1)\cdot\tilde{b}+1,1} & \dots & S_{(b-1)\cdot\tilde{b}+a-1,a-1} \end{pmatrix}$$

This matrix can be found in the frame matrix by choosing the left-most $a \times n$ sub-matrix, regarding only the non-zero side-diagonals and using these as rows of the block matrix. See figure 3.1.

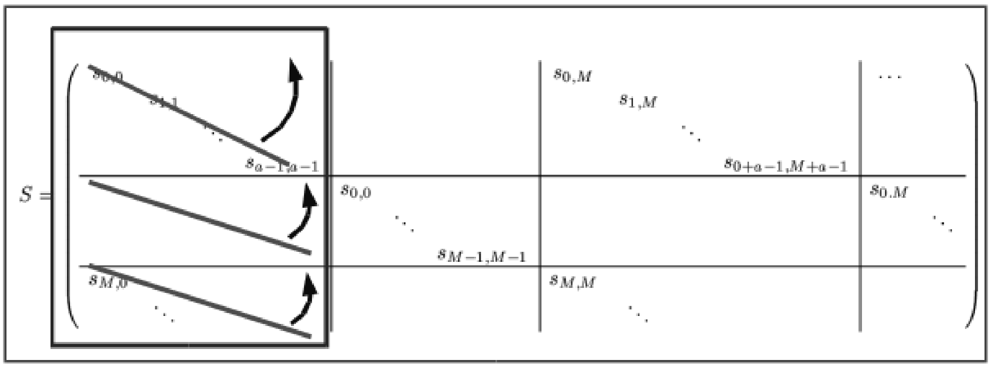

Figure 3.1: The Non-Zero Block Matrix.

The non-zero block matrix describes the frame matrix in a unique way. Keep in mind that in this chapter matrices are also regarded as periodic in rows and columns.

Corollary 3.1.5 *Let B be the non-zero block matrix of the Gabor frame matrix $S = S_{\gamma,g}$, then*

$$S_{i,j} = Ш_{\tilde{b}}(i-j) B_{\lfloor \frac{i-j}{\tilde{b}} \rfloor, j}.$$

Proof: For a given S let $B_{i,j} = S_{i\cdot\tilde{b}+j,j}$. Then let

$$S'_{i,j} := Ш_{\tilde{b}}(i-j) B_{\lfloor \frac{i-j}{\tilde{b}} \rfloor, j} = Ш_{\tilde{b}}(i-j) S_{\lfloor \frac{i-j}{\tilde{b}} \rfloor \cdot \tilde{b}+j,j}$$

Let $i - j \bmod \tilde{b} \neq 0$, then $S'_{i,j} = 0 = S_{i,j}$, see Equation 3.2.

Let $i - j \bmod \tilde{b} = 0$, then there is a k such that $i = j + k \cdot \tilde{b}$. Then

$$S'_{i,j} = Ш_{\tilde{b}}(j + k\cdot\tilde{b} - j) S_{\lfloor \frac{j+k\cdot\tilde{b}-j}{\tilde{b}} \rfloor \cdot \tilde{b}+j,j} = S_{k\cdot\tilde{b}+j,j} = S_{i,j}$$

□

The auto-correlation matrix B provides the following useful properties: S is diagonal if and only if B is zero except in the first row, and S is circulant if and only if the rows of B are constant.

Combining Definition 3.1.3 and Corollary 3.1.4 the non-zero block matrix can be expressed as

$$B_{i,j} = \tilde{b} \sum_{k=0}^{\tilde{a}-1} g\left(i\tilde{b} + j - ak\right) \overline{\gamma}\left(j - ak\right). \tag{3.3}$$

From Corollary 3.1.5 it is clear that the reconstruction can be done by using the following formula:

$$(Sx)_j = \sum_{p=0}^{b-1} x_{j+p\tilde{b}} \cdot B_{p,j+p\tilde{b}} \tag{3.4}$$

There are two strategies to factorize S and exploit the structure of S, one strategy uses the periodicity, the other the regular sparsity of this matrix. We will look at matrices with this special structure, having diagonal blocks or being block-circulant, in Section 3.2.

It is also possible to realize the multiplication of two Gabor operators by using only 'non-zero' block matrices, refer also to Section 3.2.4. This leads to a very efficient algorithm with $\mathcal{O}(a \cdot b \cdot log(b))$ operations, if the FFT is used [107]. This idea can be incorporated in iterative schemes like the conjugate gradient method.

3.1.2.2 Conditions For Gabor Frames

Lemma 3.1.6 ([103] Corollary 2)*If (g, a, b) is a Gabor frame triple, all entries in the main diagonal are strictly positive. If for (g, a, b) there is $k_0 \in \{0, \ldots, a-1\}$ such that $g(k_0 + na) = 0$ for $n = 0, \ldots, \overline{a} - 1$, then it cannot generate a frame.*

Remember that with $\hat{g}$ we denote the discrete Fourier transformation of g, see Section A.3.6

Theorem 3.1.7 ([103] Theorem 3) *The Gabor system (g, a, b) is a Gabor frame if and only if $(\hat{g}, b, a)$ forms a Gabor frame. Then*

$$S_g \circ \mathcal{F}_L = L \cdot \mathcal{F}_L \circ S_{\hat{g}}$$

Corollary 3.1.8 ([103] Corollary 7) *A necessary condition for the Gabor system (g, a, b) to generate a Gabor frame is that there are at least* a *nonzero coordinates of g and there are at least* b *nonzero coordinates of $\hat{g}$.*

In [103] there is another corollary dealing with sufficient conditions for Gabor frames, which is corrected and extended in Corollary 3.4.2.

3.1.2.3 Special Conditions On The Window And The Lattice

If g or $\hat{g}$ fulfills the following conditions, it is well known that the frame matrix has a very simple structure, see e.g. [124] or [122], which can be seen as consequence of the Walnut representation and Theorem 3.1.7.

Theorem 3.1.9

1. *If the length of the support of g $|supp(g)| \leq \tilde{b}$ then the associated Gabor frame operator for the Gabor system (g, a, b) is represented by a diagonal matrix.*
2. *If the bandwidth of g, $|supp(\hat{g})| \leq \tilde{a}$ then the associated Gabor frame operator for the Gabor system (g, a, b) is represented by a circulant matrix.*

Also if the biggest and smallest possible choices for the lattice parameters are chosen, then we get a very special structure, which also can be readily deduced from the Walnut representation:

Theorem 3.1.10

1. *For the Gabor system $(g, a, 1)$ the associated Gabor frame operator is represented by a diagonal matrix.*
2. *For the Gabor system $(g, 1, b)$ the associated Gabor frame operator is represented by a circulant matrix.*

Proof: Theorem 3.1.10 (1) was proved in [103]. This is also a direct consequence of the Walnut representation and the fact that in this case $\tilde{b} = L$.

Theorem 3.1.10 (2) is also apparent from the Walnut representation, as we know that the side-diagonals are a-periodic. If $a = 1$ then they are constant. □

With these results Theorem 3.1.9 can be reformulated: If the frequency sampling is dense enough, then the Gabor frame matrix is diagonal. If the time sampling is dense enough, then the Gabor frame matrix is circulant.

In all these special cases, it is easy to determine, whether the matrix is invertible and so the system forms a frame. It is also easy to find the inverse matrix, which will be used in Section 3.4.

3.1.2.4 The Janssen Matrix

In addition to the auto-correlation matrix defined above, there is another "small" $(b \times a)$ matrix, which fully describes the frame matrix S as a discrete analog to Theorem 2.1.10:

Definition 3.1.4 *The* **Janssen matrix** *of S is the $a \times b$ matrix J, given by*

$$(J)_{k,l} = \frac{L}{a \cdot b} \cdot c_{k,l}$$

with $c_{k,l} = (\mathcal{V}_g(\gamma))\,(lL/b, kL/a)$.

The set of time-frequency shifts normed with the factor $\frac{1}{\sqrt{L}}$ forms an orthonormal for the Hilbert-Schmidt inner product, see Proposition 3.2.3. The entries of the Janssen-matrix are given by $c_{k,l} = \left\langle \gamma, M_{kL/a} T_{lL/b} g \right\rangle$ and according to Lemma 3.2.25 they are the coefficients (up to a factor) of the following expansion:

Definition 3.1.5 *We call*

$$S_{g,\gamma} = \frac{L}{a \cdot b} \sum_{k=0}^{a-1} \sum_{l=0}^{b-1} c_{k,l} M_{k\tilde{a}} T_{l\tilde{b}} \tag{3.5}$$

the **Janssen-representation** *of S.*

We will revisit this matrix in Section 3.2.

3.1.2.5 Higher Dimensional Approach

For the biggest part of this chapter we will mostly use one-dimensional spaces. In Section 3.4.3.5 the new double preconditioning algorithm will be used for the two-dimensional case. We will only use separable windows, which means that $g = g_1 \otimes g_2 \otimes \ldots \otimes g_m$, i.e. $g(x_1, x_2, \ldots, x_m) = g_1(x_1) \cdot g_2(x_2) \cdot \ldots \cdot g_m(x_m)$. In this case $S_g = S_{g_1} \otimes S_{g_2} \otimes \ldots \otimes S_{g_m}$ and all questions can be reduced to the one-dimensional case, cf. [103].

3.1.2.6 The Strohmer Algorithm

Important algorithms for inverting a frame matrix are the frame algorithms, cf. 1.2.13, and the conjugate gradient method , cf. 1.2.14, which both work for any frame. Strohmer [122] and Prinz [99] have invented a fast algorithms for Gabor frames, exploiting the special block structure of the Gabor frame matrix: From the Walnut representation it is clear that the Gabor frame matrix on a lattice with parameters a, b can be represented by a matrix with b diagonal blocks or an a-block-circulant matrix. For a further investigation of this structure we refer to Section 3.2. In [122] the Gabor frame matrix is represented by even smaller block-matrices, which enables the numerically efficient computation of the inverse matrix by the inverse of the small matrices. There it has been proved that even if a non-iterative and slow algorithm for the inversion of a matrix is used, this algorithm is quite effective if there is a number-theoretical relation between a and $\tilde{b}$ or b and $\tilde{a}$, meaning there is a common factor dividing both numbers. For more see [122].

3.2 Matrices

Some results here have already been used e.g. in [122] [124]. So, especially at the beginning, this section is meant as rigorous summary of these statements.

3.2.1 The Matrix For The Translation And Modulation

The translation and modulation on $\mathbb{C}^L$ can, as linear operators, be obviously expressed as matrices.

Corollary 3.2.1 *For the (circular) translation on* $\mathbb{C}^L$

$$T_k^{(L)} x = (x_{L-k}, x_{L-k+1}, \ldots, x_0, x_1, \ldots, x_{L-k-1})$$

$$\left(T_k^{(L)} f\right)_l = f_{(l-k) \ mod \ L}$$

the matrix representation is

$$T_k^{(L)} = \Pi_L^k$$

where

$$(\Pi_L)_{i,j} = \delta_{i,j+1} = \delta_{i-1,j}$$

$$\Pi_L = \begin{pmatrix} 0 & 0 & \cdots & 0 & 1 \\ 1 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{pmatrix}$$

The multiplication from the left, $\Pi_L^k \cdot A$*, results in a shift of the rows down, the multiplication from the right results in a shift of columns to the left:*

$$\left(\Pi_L^k \cdot A\right)_{i,j} = A_{i-k,j} \text{ and } \left(A \cdot \Pi_L^k\right)_{i,j} = A_{i,j+k}$$

For the adjoint $\Pi_L^* = \Pi_L^{L-1}$ *the opposite properties hold.*

Proof:

$$(\Pi_L \cdot c)_i = \sum_{l=0}^{L-1} \delta_{i-1,l} c_l = c_{i-1}$$

so T_1 is represented by Π_L. As $T_k = \underbrace{T_1 \circ \ldots \circ T_1}_{k\ times}$ we have $T_k = \Pi_L^k$.

Using Lemma A.3.7 it is clear that $\Pi_L \cdot A$ results in a shift of the rows down, $(\Pi_L A)_{k,l} = A_{k-1,l}$.

$$(A \cdot \Pi_L)_{k,l} = \sum_j A_{k,j} (\Pi_L)_{j,l} = \sum_j A_{k,j} \cdot \delta_{j,l+1} = A_{k,l+1}$$

□

Obviously a similar result can also be stated for modulations:

Corollary 3.2.2 *For the modulation on* $\mathbb{C}^L$

$$M_k x = \left(x_0 \cdot \omega_L^0, x_1 \cdot \omega_L^{1 \cdot k}, \ldots, x_{L-1} \cdot \omega_L^{(L-1)k}\right) \text{ with } \omega_L = e^{\frac{2\pi i}{L}}.$$

$$(M_k f)_l = e^{-2\pi i k l / L} f_l$$

the matrix representation is

$$M_p = \Omega_L^p$$

where

$$\Omega_L = \mathfrak{diag}(1, \omega_L, \omega_L^2, \cdots, \omega_L^{L-1})$$
$$(\Omega_L)_{i,j} = \delta_{i,j} \omega_L^i$$

and the adjoint is

$$\Omega_L^* = \mathfrak{diag}(1, \omega_L^{-1}, \omega_L^{-2}, \cdots, \omega_L^{-(L-1)})$$

From Section A.3.5.1 we know that the $L \times L$ matrices are algebraically isomorph to the vector space $\mathbb{C}^{L^2}$ with

$$M \mapsto \mathfrak{vec}^{(L)}(M).$$

Proposition 3.2.3 *1. The norms of the time-frequency shifts are*

(a) $\|M_k T_l\|_{Op} = 1$

(b) $\|M_k T_l\|_{fro} = \sqrt{L}$

2. *The system* $\left(\frac{M_k T_l}{\sqrt{L}} \, | k, l = 0, \dots L-1\right)$ *is an orthonormal basis for the vector space of all* $L \times L$ *matrices with* $\langle ., . \rangle_{\mathcal{HS}}$ *as inner product.*

Proof: 1a) is the equivalent to Lemma 2.1.1. It can also be shown directly:

$$(M_k T_l f)_i = \omega_L^{ki} f_{i-l}$$

$$\Longrightarrow \|M_k T_l f\|_2^2 = \sum_{i=0}^{L-1} \left|\omega_L^{ki} f_{i-l}\right|^2 = \sum_{i=0}^{L-1} |f_{i-l}|^2 = \|f\|_2^2$$

1b)

$$(M_k T_l)_{p,q} = \sum_{j=0}^{L-1} \delta_{p,j} \cdot \omega_L^{pk} \cdot \delta_{j,q+l} = \delta_{p,q+l} \cdot \omega_L^{pk} \tag{3.6}$$

$$\Longleftrightarrow \|M_k T_l\|_{fro}^2 = \sum_{p=0}^{L-1} \sum_{q=0}^{L-1} \left|\delta_{p,q+l} \cdot \omega_L^{pk}\right|^2 = \sum_{q=0}^{L-1} \left|\omega_L^{(q+l)k}\right|^2 = L$$

2)

$$\langle M_k T_l, M_{k'} T_{l'} \rangle_{fro} = \sum_{p=0}^{L-1} \sum_{q=0}^{L-1} (M_k T_l)_{p,q} \cdot \overline{(M_{k'} T_{l'})_{p,q}} =$$

$$= \sum_{p=0}^{L-1} \sum_{q=0}^{L-1} \delta_{p,q+l} \cdot \omega_L^{pk} \cdot \delta_{p,q+l'} \omega_L^{-pk'} = \sum_{q=0}^{L-1} \omega_L^{(q+l)\cdot k} \cdot \delta_{q+l,q+l'} \cdot \omega_L^{-(q+l')k'} =$$

$$= \omega_L^{l \cdot k} \cdot \omega_L^{-l' \cdot k'} \sum_{q=0}^{L-1} \omega_L^{q \cdot (k-k')} \cdot \delta_{l,l'} \overset{Poisson}{=} \omega_L^{l \cdot k} \cdot \omega_L^{-l' \cdot k'} \cdot L \cdot \delta_{k,k'} \cdot \delta_{l,l'} = L \cdot \delta_{(k,l),(k',l')}$$

□

The commutation of time-frequency shifts, see Lemma 2.1.1, in the finite case is the following statement:

Corollary 3.2.4 *Let* T_k *be the translation and* M_l *the modulation on* $\mathbb{C}^L$*. Then*

$$\omega_L^{lk} (T_k M_l) = (M_l T_k)$$

Proof:

$$(T_k M_l)_{i,j} = \sum_{p=0}^{L-1} \delta_{i,p+k} \delta_{p,j} \omega_L^{lj} = \delta_{i,j+k} \omega_L^{lj}$$

$$(M_l T_k)_{i,j} = \sum_{p=0}^{L-1} \delta_{i,p} \omega_L^{li} \delta_{p,j+k} = \omega_L^{li} \delta_{i,j+k} = \delta_{i,j+k} \omega_L^{l(j+k)} = \omega_L^{lk} \cdot \delta_{i,j+k} \omega_L^{lj}$$

$$\Longrightarrow \omega_L^{lk}(T_k M_l) = (M_l T_k)$$

□

3.2.2 Diagonal And Circulant Matrices

Definition 3.2.1 *An $L \times L$-matrix M is called* **circulant***, if there exists $h \in \mathbb{C}^L$ such that*

$$M_{ij} = h_{(i-j) mod L}$$

Circulant matrices correspond to cyclic convolution operators as

$$M \cdot x = \sum_{j=0}^{L-1} M_{i,j} \cdot x_j = \sum_{j=0}^{L-1} h_{i-j} \cdot x_j = (h * x)(i) \tag{3.7}$$

Lemma 3.2.5 *A matrix M is circulant if and only if it commutes with all (cyclic) translations.*

$$M \cdot T_k = T_k \cdot M$$

Proof: For all k

$$M \cdot T_k = T_k \cdot M \Longleftrightarrow T_k^* \cdot M \cdot T_k = M$$

$$M_{i+k,j+k} = M_{i,j} \Longleftrightarrow M_{i,j} = h_{i-j}$$

□

It's easy to see that

Proposition 3.2.6 *A matrix M is circulant if and only if it can be described uniquely as linear combination of (cyclic) translation matrices.*

$$M = \sum_{k=0}^{L-1} c_k \cdot T_k$$

Proof: A matrix is circulant, if and only if the side-diagonals are constant. Clearly we can represent this matrix by matrices M_k, where only the l-th side-diagonal is non-zero and constant, c_l. From Corollary 3.2.1 we know that this matrix can be represented by $c_l \cdot \Pi_L^l$. The opposite direction of the inclusion is obvious. □

Similar statements can be made for diagonal matrices:

Lemma 3.2.7 *A matrix A is diagonal if and only if it commutes with all modulations.*

$$A \cdot M_k = M_k \cdot A$$

Proof: For $p, q = 0, \ldots, L-1$:

$$\left(M_k \cdot A \cdot M_k^*\right)_{p,q} = \sum_i \sum_j \delta_{i,p} \cdot \omega_L^{pk} \cdot A_{i,j} \cdot \delta_{j,q} \cdot \omega_L^{-jk} =$$

$$= \omega_L^{pk} \cdot A_{p,q} \cdot \omega_L^{-qk} \Longrightarrow$$

$$A_{p,q} = \left(M_k \cdot A \cdot M_k^*\right)_{p,q} \Longleftrightarrow A_{p,q} = \omega_L^{(p-q)\cdot k} \cdot A_{p,q}$$

Let $p = q$, then this is always true for all k. So we see that diagonal matrices always commute with the modulation. Let $p \neq q$ and suppose $A_{p,q} \neq 0$, then for $k = 1$

$$\omega_L^{(p-q)} = 1 \Longrightarrow p - q = 0 \bmod L \Longrightarrow p - q = 0$$

This is a contradiction, so for all $p \neq q$ $A_{p,q} = 0$. □

Proposition 3.2.8 *A matrix S is diagonal if and only if it can be described uniquely as linear combination of modulations.*

$$S = \sum_{p=0}^{L-1} c_k \cdot M_k$$

Let $d = \mathfrak{diag}(S)$ then $c_k = \frac{1}{L}\left(\hat{d}\right)_k$.

Proof: If S looks like that, it clearly is diagonal.

Let S be a diagonal matrix, and let $d = \mathfrak{diag}(S)$ be its diagonal. Following Theorem A.3.10 we can represent d with the ONB $f_k(i) = \frac{1}{\sqrt{L}}\omega_L^{k\cdot i}$.

$$d_i = \sum_{k=0}^{L-1} c'_k \frac{1}{\sqrt{L}} \omega_L^{k\cdot i}$$

Therefore

$$M = \mathfrak{diag}(d) = \sum_{k=0}^{L-1} c'_k \frac{1}{\sqrt{L}} \mathfrak{diag}(\omega_L^{k\cdot i}) = \sum_{k=0}^{L-1} c_k \Omega_L^k$$

where $c_k = \frac{c'_k}{\sqrt{L}}$.

The coefficients are

$$c'_k = \langle d, f_k \rangle = \sum_{i=0}^{L-1} d_i \cdot \frac{1}{\sqrt{L}} \cdot \omega_L^{-k \cdot i} = \frac{1}{\sqrt{L}} \hat{d}(k)$$

And therefore

$$c_k = \frac{1}{L} \left(\hat{d} \right)_k$$

□

3.2.3 Matrix Fourier Transformation

The notion of Fourier transformation can be easily extended to matrices, as used in [122]. Remember that $\mathcal{F}_n$ is the *Fourier-matrix*, $\mathcal{F}_{n i,j} = \omega_n^{-i \cdot j}$, cf. Section A.3.6.

Definition 3.2.2 *Let* $A \in M_{m,n}$. *The* **Matrix Fourier Transformation (MFT)** *of* A *is defined by*

$$\mathfrak{F}(A) = \hat{A} = \mathcal{F}_m \circ A \circ \mathcal{F}_n^*$$

Therefore

$$\mathfrak{F}(A)_{i,j} = \sum_{p=0}^{m-1} \sum_{q=0}^{n-1} \omega_m^{-i \cdot p} \cdot A_{p,q} \cdot \omega_n^{q \cdot j} \tag{3.8}$$

This is not the same as the 2-dimensional Fourier transformation, which is $\mathcal{F}^2(A) = \mathcal{F}_m \cdot A \cdot \mathcal{F}_n$, but obviously $\mathcal{F}^2(A)(p,q) = \hat{A}(p,-q)$.

We can show the following properties

Lemma 3.2.9 *For* $A, A' \in M_{m,n}$, $B \in M_{n,p}$ *we have*

- $\mathfrak{F}^{-1}(A) = \check{A} = F_m^* \circ S \circ F_n$ *is the inverse transformation.*
- $\widehat{A \cdot B} = \hat{A} \cdot \hat{B}$.
- $\widehat{M_k} = T_k$ *and* $\widehat{T_k} = M_{-k}$.
- $\widehat{(Af)} = \hat{A}\hat{f}$
- $\widehat{A * A'} = \hat{A} \odot \hat{A}'$ *where* $\odot$ *is the pointwise product (see also A.3.17) and* $*$ *is the convolution of matrices as defined in Section A.3.6.1.*

Proof: 1)

$$\mathfrak{F}^{-1}\left(\mathfrak{F}\left(A\right)\right) = \mathcal{F}_m^*\left(\mathcal{F}_m \circ A \circ \mathcal{F}_n^*\right)\mathcal{F}_n = A.$$

With an analogue argument $\mathfrak{F}\left(\mathfrak{F}^{-1}\left(A\right)\right) = A$.

2.)

$$\mathfrak{F}(A \cdot B) = \mathcal{F}_m \circ A \circ B \circ \mathcal{F}_p^* = \mathcal{F}_m \circ A \circ F_n^* \circ F_n \circ B \circ \mathcal{F}_p^*$$

3.)

$$\left(\widehat{M_k}\right)_{l_1,l_2} = \sum_{i=0}^{L-1}\sum_{j=0}^{L_1} \omega_L^{-l_1 \cdot i} \cdot \delta_{i,j} \cdot \omega_L^{kj} \cdot \omega_L^{j \cdot l_2} =$$

$$= \sum_{j=0}^{L_1} \omega_L^{-l_1 \cdot j} \cdot \omega_L^{kj} \cdot \omega_L^{j \cdot l_2} = \sum_{j=0}^{L_1} \omega_L^{(-l_1+k+l_2)\cdot j} = \delta_{l_1,l_2+k} = (T_k)_{l_1,l_2}$$

$$\left(\widehat{T_k}\right)_{l_1,l_2} = \sum_{p=0}^{L-1}\sum_{q=0}^{L-1} \omega_L^{-i \cdot p}\delta_{p,q+k}\omega_L^{q \cdot j} =$$

$$= \sum_{q=0}^{n-1} \omega_L^{-i\cdot(q+k)}\omega_L^{q \cdot j} = \sum_{q=0}^{n-1} \omega_L^{-i \cdot k}\omega_L^{q\cdot(j-i)} = \omega_L^{-i \cdot k}\delta_{i,j} = (M_{-k})_{i,j}$$

4.)

$$\widehat{(Tf)} = \mathcal{F}_m \cdot T \cdot f = \mathcal{F}_m \cdot T \cdot (\mathcal{F}_n^* \cdot \mathcal{F}_n) \cdot f = \hat{T}\hat{f}$$

5.)

$$(A * A')_{k,l} = \sum_{i_1=0}^{m-1}\sum_{i_2=0}^{n-1} A_{i_1,i_2} \cdot A'_{k-i_1,l-i_2} \Longrightarrow$$

$$(\mathcal{F}_m \cdot (A * A') \cdot \mathcal{F}_n^*)_{p,q} = \sum_{k=0}^{m-1}\sum_{l=0}^{n-1} \omega_m^{-p\cdot k} \cdot \sum_{i_1=0}^{m-1}\sum_{i_2=0}^{n-1} A_{i_1,i_2} \cdot A'_{k-i_1,l-i_2} \cdot \omega_n^{l \cdot q} =$$

$$= \sum_{i_1=0}^{m-1}\sum_{i_2=0}^{n-1} A_{i_1,i_2} \cdot \sum_{k=0}^{m-1}\sum_{l=0}^{n-1} \omega_m^{-p\cdot k} A'_{k-i_1,l-i_2} \cdot \omega_n^{l \cdot q} =$$

$$= \sum_{i_1=0}^{m-1}\sum_{i_2=0}^{n-1} A_{i_1,i_2} \cdot \sum_{k'=0}^{m-1}\sum_{l'=0}^{n-1} \omega_m^{-p\cdot(k'+i_1)} A'_{k',l'} \cdot \omega_n^{(l'+i_2)\cdot q} =$$

$$= \sum_{i_1=0}^{m-1}\sum_{i_2=0}^{n-1} \omega_m^{-p \cdot i_1} A_{i_1,i_2}\omega_n^{i_2 q} \cdot \sum_{k'=0}^{m-1}\sum_{l'=0}^{n-1} \omega_m^{-p\cdot k'} A'_{k',l'} \cdot \omega_n^{l' \cdot q} =$$

$$= \hat{A}_{p,q} \cdot \hat{A'}_{p,q}$$

□

Theorem 3.2.10 *1.* (Plancherel) *The MFT is an isometry for* $\|.\|_{Op}$

2. (Parseval) $\left\langle \hat{A}, \hat{B} \right\rangle_{fro} = \langle A, B \rangle_{fro}$

3. The MFT is an isometric isomorphism for $\|.\|_{Op}$ *and* $\|.\|_{fro}$

Proof: 1.) Let T be a matrix. We know that $f \mapsto \hat{f}$ is a bijective isometric function from $\mathbb{C}^L$ to $\mathbb{C}^L$ with $\|.\|_2$ and so

$$\left\| \hat{T} \right\|_{Op} = \sup_{\|g\|_2 = 1} \left\{ \left\| \hat{T} g \right\| \right\} = \sup_{\left\| \hat{f} \right\|_2 = 1} \left\{ \left\| \hat{T} \hat{f} \right\| \right\} =$$

$$= \sup_{\|f\|_2 = 1} \left\{ \left\| \widehat{Tf} \right\| \right\} = \sup_{\|f\|_2 = 1} \{ \|Tf\| \} = \|T\|_{Op}$$

2.) Let $A, b \in M_{m,n}$.

$$\left\langle \hat{A}, \hat{B} \right\rangle_{fro} = \langle \mathcal{F}_m A \mathcal{F}_n^*, \mathcal{F}_m B \mathcal{F}_n^* \rangle_{fro} \overset{A.3.8}{=}$$

$$= \langle \mathcal{F}_m^* \mathcal{F}_m A, B \mathcal{F}_n^* \mathcal{F}_n \rangle_{fro} = \langle A, B \rangle_{fro}$$

3.) Clear from above and Lemma 3.2.9. □

Theorem 3.2.11 *For a circulant matrix* M *the matrix* $\hat{M}$ *is diagonal.*
For a diagonal matrix D *the matrix* $\hat{D}$ *is circulant.*

Proof: This is now a direct consequence of Lemma 3.2.6, Proposition 3.2.8 and Lemma 3.2.9 as

$$M = \sum_{k=0}^{L-1} c_k \cdot T_k \Longleftrightarrow \hat{M} = \sum_{k=0}^{L-1} c_k \cdot M_{-k}$$

and vice versa. □

Therefore a circulant matrix A is invertible, if and only if $\hat{A}$ has no zeros in the diagonal. This connection will be used in Section 3.4.1.3.

3.2.4 Block Matrices

Every matrix $L \times L$ A can be written as block matrix in the following sense:

$$A = \begin{pmatrix} B_{0,0} & B_{0,1} & \cdots & B_{0,b-1} \\ B_{1,0} & B_{1,1} & \cdots & B_{1,b-1} \\ \vdots & \vdots & \ddots & \vdots \\ B_{b-1,0} & B_{b-1,1} & \cdots & B_{b-1,b-1} \end{pmatrix} \tag{3.9}$$

where the $B_{i,j}$ are $\tilde{b} \times \tilde{b}$ matrices, where $\tilde{b} = \frac{L}{b}$.

Let us fix notations: let the $b \times b$ matrix be

$$E_k^{(b)} = \mathfrak{diag}(\delta_k), \left(E_k^{(b)}\right)_{i,j} = \delta_{k,i} \cdot \delta_{k,j}$$

with δ_k the k-th unit vector in $\mathbb{C}^b$. Also let for $i, j = 0, \ldots, b-1$

$$\left(\mathcal{E}_{k,l}^{(b \times b)}\right)_{i,j} = \delta_{k,i} \cdot \delta_{l,j} = \begin{cases} 1 & k = i, l = j \\ 0 & otherwise \end{cases}$$

Clearly

Lemma 3.2.12

$$\mathcal{E}_{k,l}^{(b \times b)} = T_{k-l} \cdot E_l^{(b)} \text{ and } T_k E_l^{(b)} = \mathcal{E}_{k+l,l}^{(b \times b)}$$

Proof:

$$\left(T_{k-l} \cdot E_l^{(b)}\right)_{i,j} = \sum_{p=0}^{b-1} \delta_{i,p+(k-l)} \cdot \delta_{l,p} \cdot \delta_{l,j} =$$

$$= \delta_{i,l+(k-l)} \cdot \delta_{l,j} = \delta_{i,k} \cdot \delta_{l,j} = \left(\mathcal{E}_{k,l}^{(b \times b)}\right)_{i,j}$$

□

Using the Kronecker product defined in Section A.3.7, another way to express the block structure in 3.9 is :

$$A = \sum_{i=0}^{b-1} \sum_{j=0}^{b-1} \mathcal{E}_{i,j}^{(b \times b)} \otimes B'_{i,j} = \sum_{i=0}^{b-1} \sum_{j=0}^{b-1} \left(T_{i-j} \cdot E_j^{(b)}\right) \otimes B'_{i,j}$$

3.2.4.1 Matrices with Diagonal Blocks

Definition 3.2.3 *An $L \times L$ matrix M is called a b-***block-diagonal matrix** *if*

$$M = \begin{pmatrix} D_0 & 0 & \cdots & 0 \\ 0 & D_1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & D_{\tilde{b}-1} \end{pmatrix}$$

where $\tilde{b} = \frac{L}{b}$ and the D_j are arbitrary $b \times b$-matrices.

A matrix M is called a **matrix with b diagonal blocks** *if*

$$M = \begin{pmatrix} D_{0,0} & D_{0,1} & \cdots & D_{0,b-1} \\ D_{1,0} & D_{1,1} & \cdots & D_{1,b-1} \\ \vdots & \vdots & \ddots & \vdots \\ D_{b-1,0} & D_{b-1,1} & \cdots & D_{b-1,b-1} \end{pmatrix}$$

where $L = \tilde{b} \cdot b$ and the D_j are diagonal $\tilde{b} \times \tilde{b}$-matrices.

Clearly matrices with b diagonal blocks have only b non-zero side-diagonals, more precisely only every $\tilde{b}$-the side-diagonal is non-zero. So the frame matrix for a Gabor frame has such a structure.

Proposition 3.2.13 *1. A matrix M is b-block-diagonal if and only if*

$$M = \sum_{k=0}^{\tilde{b}-1} E_k^{(\tilde{b})} \otimes D_k$$

2. A matrix M has b diagonal blocks if and only if

$$M = \sum_{k=0}^{\tilde{b}-1} B_k \otimes E_k^{(\tilde{b})}$$

The D_k are defined as in definition 3.2.3 and $(B_k)_{l,m} = M_{k+l\tilde{b},k+m\tilde{b}} = (D_{l,m})_k$ with the $D_{l,m}$ as in 3.2.3.

Proof: 1) is obvious from the definition of the Kronecker product in Section A.3.7.

2) First let M be a matrix with b diagonal blocks (like in Definition 3.2.3). Then

$$M = \sum_{i=0}^{b-1} \sum_{j=0}^{b-1} \mathcal{E}_{i,j}^{(b \times b)} \otimes D_{i,j}$$

Let $D_{i,j} = \mathfrak{diag}(d^{(i,j)})$, then for each p let $p_1 = \left\lfloor \frac{p}{\tilde{b}} \right\rfloor$ and $p_2 = p \bmod \tilde{b}$ such that $p = p_1 \cdot \tilde{b} + p_2$ and a similar decomposition for $q = q_1 \cdot \tilde{b} + q_2$, then

$$M_{p,q} = \sum_{i=0}^{b-1} \sum_{j=0}^{b-1} \delta_{i,p_1} \cdot \delta_{j,q_1} \cdot \delta_{p_2,q_2} \cdot d_{p_2}^{(i,j)} = \delta_{p_2,q_2} \cdot d_{p_2}^{(p_1,q_1)}$$

On the other hand let $(B_k)_{l,m} = (D_{l,m})_k = d_k^{(l,m)}$, then clearly the matrix M is uniquely described by these matrices. Then

$$\left(B_k \otimes E_k^{(\tilde{b})}\right)_{p,q} = d_k^{(p_1,q_1)} \cdot \delta_{p_2,k} \cdot \delta_{q_2,k} \Longrightarrow$$

$$\left(\sum_{k=0}^{\tilde{b}-1} B_k \otimes E_k^{(\tilde{b})}\right)_{p,q} = \sum_{k=0}^{\tilde{b}-1} d_k^{(p_1,q_1)} \cdot \delta_{p_2,k} \cdot \delta_{q_2,k} =$$

$$= d_{p_2}^{(p_1,q_1)} \cdot \delta_{q_2,p_2}$$

□

This representation is obviously unique. So in $\mathbb{M}_{L\times L}$ there are $\tilde{b} \cdot b^2 = L \cdot b$ b-block-diagonal matrices resp. $\tilde{b}^2 \cdot b = L \cdot \tilde{b}$ matrices with b diagonal blocks.

Lemma 3.2.14 *The product of two b-block-diagonal matrices again is b-block-diagonal. If* $A = \sum_k E_k \otimes D_k$ *and* $B = \sum_k E_k \otimes C_k$ *then*

$$C = A \cdot B = \sum_k E_k \otimes (D_k \cdot C_k)$$

The product of two matrices with b diagonal blocks is again a matrix with b diagonal blocks. Let $D_{l,j}^{(A)}$ *be the diagonal blocks of A and* $D_{j,m}^{(B)}$ *of B, then the diagonal blocks of* $C = A \cdot B$ *are*

$$D_{l,m}^{(C)} = \sum_j D_{l,j}^{(A)} \odot D_{j,m}^{(B)}$$

Proof: Using Proposition A.3.12 we get:

$$A \cdot B = \sum_k \sum_l (E_k \otimes D_k) \cdot (E_l \otimes C_l) = \sum_k \sum_l (E_k \cdot E_l) \otimes (D_k \cdot C_l)$$

$$(E_k \cdot E_l)_{p,q} = \sum_i \delta_{k,p} \cdot \delta_{k,i} \cdot \delta_{i,l} \cdot \delta_{q,l} = \delta_{k,p} \cdot \delta_{k,l} \cdot \delta_{q,l} = \delta_{k,l} \cdot (E_k)_{p,q}$$

and so the first part is proved. With an analogous argument the same is true for matrices with diagonal blocks.

Using Proposition 3.2.13 we know that

$$\left(D^{(C)}_{l,m}\right)_{k,k} = (D_k \cdot C_k)_{l,m} =$$

$$= \sum_j (D_k)_{l,j} \cdot (C_k)_{j,m} = \sum_j \left(D^{(A)}_{l,j}\right)_{k,k} \cdot \left(D^{(B)}_{j,m}\right)_{k,k}$$

□

Lemma 3.2.15 *A matrix A has b diagonal blocks if and only if it commutes with all translations M_{lb}.*

$$A \cdot M_{lb} = M_{lb} \cdot A$$

Proof: This is analogous to Lemma 3.2.7: For $p, q = 0, \ldots, L-1$:

$$(M_{lb} \cdot A \cdot M^*_{lb})_{p,q} = \sum_i \sum_j \delta_{i,p} \cdot \omega_L^{plb} \cdot A_{i,j} \cdot \delta_{j,q} \cdot \omega_L^{-jlb} =$$

$$= \omega_L^{plb} \cdot A_{p,q} \cdot \omega_L^{-qlb} \Longrightarrow$$

$$A_{p,q} = (M_{lb} \cdot A \cdot M^*_{lb})_{p,q} \Longleftrightarrow A_{p,q} = \omega_L^{(p-q)\cdot lb} \cdot A_{p,q}$$

Let $p - q \bmod \tilde{b} = 0$, then $(p-q)\, lb \bmod L = 0$ for all l and so the above statement is true. This means that matrices, where only every $\tilde{b}$-th side-diagonal is non-zero, fulfill this condition. So one direction is proved.

Let $p - q \bmod \tilde{b} \neq 0$ and suppose $A_{p,q} \neq 0$, then for $l = 1$

$$\omega_L^{(p-q)b} = 1 \Longleftrightarrow \omega_{\tilde{b}}^{(p-q)} = 1$$

$$\Longrightarrow p - q \bmod \tilde{b} = 0$$

This is a contradiction, so for all $p - q \bmod \tilde{b} \neq 0$ $A_{p,q} = 0$. □

It is also possible to use modulation to describe such a matrix

Proposition 3.2.16 *A matrix M has b diagonal blocks if and only if*

$$M = \sum_{k=0}^{b-1} \Pi_L^{k\tilde{b}} D'_k \ ,$$

where $D'_k = \sum\limits_{l=0}^{b-1} E_l \otimes D_{l+k,l+k}$ is a diagonal $L \times L$ matrix. ($D_{i,j}$ as in 3.2.3.)

Proof: These matrices are exactly those, where only every $\tilde{b}$-th side-diagonal is non-zero. A matrix consisting only of the $k \cdot \tilde{b}$ side-diagonal can be represented as $T_{k\tilde{b}} \cdot \mathfrak{diag}(d_k)$. So every matrix with b diagonal blocks has this form, where the D'_k are the entries in the $k\tilde{b}$-th side-diagonal. This diagonal matrix is composed by the diagonal blocks $D_{l,m}$ situated at these side-diagonals. So using the definition of the Kronecker product, cf. Definition A.3.16, this clearly means

$$D'_k = \sum_{l=0}^{b-1} E_l \otimes D_{l+k,l+k}$$

□

For these matrices the following properties have been shown in [122], which use the connection in Proposition 3.2.13 between $(B_k)_{l,m} = (D_{l,m})_k$ to do a reordering:

Theorem 3.2.17 ([122] Theorem 8.3.1) *Let S be a matrix with b diagonal blocks D_i, then S can be unitarily factorized into a b-block-diagonal matrix $diag(D_0, \ldots, D_{b-1})$.*

3.2.4.2 Block Circulant Matrices

Definition 3.2.4 *A matrix M is called a a-**block-circulant** matrix if*

$$M = \begin{pmatrix} A_0 & A_1 & \cdots & A_{\tilde{a}-1} \\ A_{\tilde{a}-1} & A_0 & \cdots & A_{\tilde{a}-2} \\ \ddots & \ddots & \ddots & \ddots \\ A_1 & A_2 & \cdots & A_0 \end{pmatrix}$$

where $L = a \cdot \tilde{a}$ and the A_j are arbitrary $a \times a$ matrices.

*A matrix M is called a **matrix with a circulant blocks** if*

$$M = \begin{pmatrix} A_{0,0} & A_{0,1} & \cdots & A_{0,a-1} \\ A_{1,0} & A_{1,1} & \cdots & A_{1,a-1} \\ \vdots & \vdots & \ddots & \vdots \\ A_{a-1,0} & A_{a-1,1} & \cdots & A_{a-1,a-1} \end{pmatrix}$$

where $L = a \cdot \tilde{a}$ and the $A_{j,k}$ are circulant $\tilde{a} \times \tilde{a}$-matrices.

Clearly

Corollary 3.2.18 *The a-block-circulant matrices are exactly those matrices that have a-periodic side diagonals.*

Following Equation (3.2) a Gabor frame matrix is an a-block-circulant matrix.

Lemma 3.2.19 *A matrix M is a-block-circulant if and only if it commutes with all translations $T_{k \cdot a}$.*

$$M \cdot T_{ka} = T_{ka} \cdot M$$

Proof: Proof is similar to the one of Lemma 3.2.5: For all k

$$M \cdot T_{ka} = T_{ka} \cdot M \Longleftrightarrow T_{ka}^{*} M \cdot T_{ka} = M \Longleftrightarrow M_{i+ka, j+ka} = M_{i,j}$$

and so the side-diagonals are all a-periodic. □

Proposition 3.2.20 *1. A matrix M is* a*-block-circulant if and only if*

$$M = \sum_{k=0}^{\tilde{a}-1} T_k^{(\tilde{a})} \otimes A_k$$

2. A matrix M has a *circulant blocks if and only if*

$$M = \sum_{k=0}^{\tilde{a}-1} C_k \otimes T_k^{(\tilde{a})}$$

Here A_k is like in 3.2.4 and $(C_k)_{j,l} = (A_{j,l})_{0,k}$

Proof: 1) is obvious from the definition of the Kronecker product in Section A.3.7.

2) First let M be a matrix with b diagonal blocks (like in Definition 3.2.4) . Then

$$M = \sum_{i=0}^{a-1} \sum_{j=0}^{a-1} \mathcal{E}_{i,j}^{(a \times a)} \otimes A_{i,j}$$

Let for each p let $p_1 = \left\lfloor \frac{p}{\tilde{a}} \right\rfloor$ and $p_2 = p \bmod \tilde{a}$ such that $p = p_1 \cdot \tilde{a} + p_2$ and a similar decomposition for $q = q_1 \cdot \tilde{b} + q_2$, then

$$M_{p,q} = \sum_{i=0}^{a-1} \sum_{j=0}^{a-1} \delta_{i,p_1} \cdot \delta_{j,q_1} \cdot (A_{i,j})_{p_2,q_2} = (A_{p_1,q_1})_{p_2,q_2}$$

On the other hand let $(C_k)_{j,l} = (A_{j,l})_{0,k}$ and let

$$M = \sum_{k=0}^{a-1} C_k \otimes T_k^{(a)}$$

then

$$M_{p,q} = \sum_{k=0}^{a-1} (A_{p_1,q_1})_{0,k} \cdot \delta_{p_2,q_2+k} = \sum_{k=0}^{a-1} (A_{p_1,q_1})_{0,k} \cdot \delta_{p_2-q_2,k} = (A_{p_1,q_1})_{0,p_2-q_2}$$

This is equal to the above formula as all the $A_{i,j}$ are circulant. □

Again a property shown in [122] is valid for these matrices:

Theorem 3.2.21 ([122] Theorem 8.3.3) *Let S be a a-circulant matrix. It can be unitarily factorized with $(\mathcal{F}_N \otimes I_a)^*$ into a block diagonal matrix, where I_a is the identity of $\mathbb{C}^a$.*

Lemma 3.2.22 *The product of two a-block-circulant matrices again is a-block-circulant. If $A = \sum_k T_k \otimes A_k$ and $B = \sum_k T_k \otimes B_k$ then $C = A \cdot B = \sum_k T_k \otimes C_k$ where*

$$C_k = \sum_l A_{(k-l) \text{ } mod \text{ } \tilde{a}} \cdot B_l$$

The product of two matrices with a *circulant blocks is again a matrix with* a *circulant blocks. Let $A_{l,j}$ be the circulant blocks of A and $B_{j,m}$ of B, then the circulant blocks of $C = A \cdot B$ are*

$$(C_{p,q})_{0,k} = \sum_{l=0}^{\tilde{a}-1} \sum_{r=0}^{\tilde{a}-1} (A_{p,r})_{l,k} \cdot (B_{r,q})_{0,l}$$

Proof:

$$A \cdot B = \left(\sum_{k=0}^{\tilde{a}-1} T_k \otimes A_k \right) \cdot \left(\sum_{l=0}^{\tilde{a}-1} T_l \otimes B_l \right) =$$

$$\overset{Prop.A.3.12}{=} \sum_{k=0}^{\tilde{a}-1} \sum_{l=0}^{\tilde{a}-1} (T_k \cdot T_l) \otimes (A_k \cdot B_l) = \sum_{k=0}^{\tilde{a}-1} \sum_{l=0}^{\tilde{a}-1} T_{k+l} \otimes (A_k \cdot B_l) =$$

$$\overset{k'=k+l}{=} \sum_{k'=0}^{\tilde{a}-1} \sum_{l=0}^{\tilde{a}-1} T_{k'} \otimes (A_{k'-l} \cdot B_l) = \sum_{k=0}^{\tilde{a}-1} T_k \otimes \left(\sum_{l=0}^{\tilde{a}-1} (A_{k-l} \cdot B_l) \right)$$

For matrices with b circulant blocks the same proof shows

$$C = A \cdot B = \sum_{k=0}^{\tilde{a}-1} \left(\sum_{l=0}^{\tilde{a}-1} (A_{k-l} \cdot B_l) \right) \otimes T_k$$

So

$$(C_{p,q})_{0,k} = (C_k)_{p,q} = \left(\sum_{l=0}^{\tilde{a}-1} (A_{k-l} \cdot B_l) \right)_{p,q} =$$

$$= \sum_{l=0}^{\tilde{a}-1} \sum_{r=0}^{\tilde{a}-1} (A_{k-l})_{p,r} \cdot (B_l)_{r,q} = \sum_{l=0}^{\tilde{a}-1} \sum_{r=0}^{\tilde{a}-1} (A_{p,r})_{0,k-l} \cdot (B_{r,q})_{0,l} =$$

$$= \sum_{l=0}^{\tilde{a}-1} \sum_{r=0}^{\tilde{a}-1} (A_{p,r})_{l,k} \cdot (B_{r,q})_{0,l}$$

□

Theorem 3.2.23 *1. Let M be a matrix with b diagonal blocks, then $\hat{M}$ is a b-block-circulant matrix.*

2. Let M be an a*-block-circulant matrix, then $\hat{M}$ has* a *diagonal blocks.*

Proof: This is a direct consequence of Lemma 3.2.19, 3.2.15 and 3.2.9.

$$M \cdot T_{ka} = T_{ka} \cdot M \ \forall k \Longleftrightarrow \hat{M} \cdot M_{-ka} = M_{-ka} \cdot \hat{M} \ \forall k$$

□

3.2.5 Gabor-Type Matrices

Definition 3.2.5 *We will call an a-block circulant matrix with b diagonal blocks an (a, b)***-Walnut-matrix** *or* **Gabor-type matrix**.

The example for such a matrix is of course the Gabor frame matrix for (g, a, b). Every (a, b)-Walnut-matrix S can be represented by a $a \times b$-block B (for a MATLAB-code see B.4.2.2), refer to [103] and [104]. With the investigation of block matrices above, we have the following results as direct consequences:

Corollary 3.2.24 *For a given matrix the following properties are equivalent*

1. *having a Walnut representation, i.e. being an a-block-circulant matrix with b diagonal blocks.*
2. *commuting with all* M_{ka} *and* T_{lb} *for all* $k, l \in \mathbb{Z}$.
3. *being represented by a Janssen matrix, i.e. being in the space spanned by* $\{M_{k\tilde{a}}T_{l\tilde{b}}\}$ *for* $k = 0, \ldots, a-1, l = 0, \ldots, b-1$.

Proof: The equivalence 1) $\Longleftrightarrow$ 2) is a direct consequence of Lemma 3.2.19 and Lemma 3.2.15.

If a matrix has a Walnut representation, we know from Section 3.1.2.1, that it can be represented uniquely by a small $b \times a$ matrix, the non-zero block matrix. So the space of all such matrices has the dimension $a \cdot b$. Following Lemma 2.1.12 it is evident that the matrices $\{M_{k\tilde{a}}T_{l\tilde{b}}\}$ are in this space, and following Proposition 3.2.3 they are linear independent. The space of matrices spanned by the sequence $\{M_{k\tilde{a}}T_{l\tilde{b}} : k = 0, \ldots, a-1, l = 0, \ldots, b-1\}$ has the dimension $a \cdot b$. So the two spaces coincide.

□

This means that for Gabor-type matrices the definition of the Janssen matrix, the Walnut representation respectively the non-zero block matrix can be used.

3.2.5.1 The Janssen Matrix

The set of time-frequency shifts normed with the factor $\frac{1}{\sqrt{L}}$ forms an orthonormal sequence for the Hilbert-Schmidt inner product, as stated in Proposition 3.2.3. Let us investigate the coefficients for a representation using this basis:

Corollary 3.2.25 *The Gabor frame matrix for* (g, γ, a, b) *can be represented by*

$$S_{g,\gamma} = \frac{L}{a \cdot b} \sum_{l=0}^{b-1} \sum_{k=0}^{a-1} c_{k,l} M_{k\tilde{a}} T_{l\tilde{b}}$$

with $c_{k,l} = (\mathcal{V}_g(\gamma)) \left(l\frac{L}{b}, k\frac{L}{a}\right)$

Proof: S is a a-circulant matrix with b diagonal blocks. We know that this space is spanned by the ONB $\frac{1}{\sqrt{L}} \cdot M_{l\tilde{a}}T_{k\tilde{b}}$. So

$$S = \sum_{k=0}^{a-1} \sum_{l=0}^{b-1} \left\langle S, \frac{M_{k\tilde{a}}T_{l\tilde{b}}}{\sqrt{L}} \right\rangle_{fro} \cdot \frac{M_{k\tilde{a}}T_{l\tilde{b}}}{\sqrt{L}} = \frac{1}{L} \sum_{k=0}^{a-1} \sum_{l=0}^{b-1} \langle S, M_{k\tilde{a}}T_{l\tilde{b}} \rangle_{fro} \cdot M_{k\tilde{a}}T_{l\tilde{b}}$$

$$\langle S, M_{k\tilde{a}} T_{l\tilde{b}} \rangle_{fro} = \left\langle \sum_{\lambda \in \Lambda} \gamma_\lambda \otimes g_\lambda, M_{k\tilde{a}} T_{l\tilde{b}} \right\rangle_{fro} =$$

$$= \sum_{\lambda \in \Lambda} \langle \gamma_\lambda, M_{k\tilde{a}} T_{l\tilde{b}} g_\lambda \rangle_{\mathbb{C}^L} = \sum_{\lambda \in \Lambda} \langle \gamma, \pi(\lambda)^* M_{k\tilde{a}} T_{l\tilde{b}} \pi(\lambda) g \rangle_{\mathbb{C}^L} =$$

$$\overset{Lem.2.1.12}{=} \sum_{\lambda \in \Lambda} \langle \gamma, \pi(\lambda)^* \pi(\lambda) M_{k\tilde{a}} T_{l\tilde{b}} g \rangle_{\mathbb{C}^L} = \#\Lambda \cdot \langle \gamma, M_{k\tilde{a}} T_{l\tilde{b}} g \rangle_{\mathbb{C}^L} =$$

$$= \frac{L}{a} \frac{L}{b} \mathcal{V}_g \gamma(l\tilde{b}, k\tilde{a})$$

And so the coefficients of these representations are

$$c_{k,l} = \frac{1}{L} \cdot \frac{L}{a} \frac{L}{b} \mathcal{V}_g \gamma(l\tilde{b}, k\tilde{a}) = \frac{L}{a \cdot b} \mathcal{V}_g \gamma(l\tilde{b}, k\tilde{a})$$

□

It is possible to show, see e.g. [124], that result by using the matrix representation of $S_{g,\gamma}$ and $M_{l\tilde{a}} T_{k\tilde{b}}$, but we have used a shorter, albeit more abstract proof.

3.2.5.2 Janssen Multiplication

We know that the product of a-circulant matrices and matrices with b diagonal blocks have the same property again. So it is clear that the product of Gabor-type matrices is a Gabor-type matrix again, as stated in [104] and [124]. We will extend these results and investigate how the Janssen representation of this product looks.

In [104] Theorem 2 and algorithm was presented to do the multiplication of two Gabor frame matrices on the block matrix level:

Theorem 3.2.26 *Let S_1, S_2 be two Gabor-type matrices and $S_3 = S_1 \cdot S_2$. Let B_i the non-zero block matrix of S_i, then*

$$(B_3)_{p,q} = \sum_{p=0}^{b-1} (B_1)_{p,q} \cdot (B_2)_{r_1(p,q), r_2(p,q)}$$

with $r_1(p,q) = b + q - p + 1 \ mod \ b$ and $r_2(p,q) = q + (p-1)\tilde{b} \ mod \ a$.

We are going to give a similar result for the Janssen-Matrix. For that we need the definition of the twisted convolution, following [64]:

Definition 3.2.6 *Let $\theta > 0$ and let M, N be (possibly) infinite matrices. Then we define the* **twisted convolution** *by*

$$(M\natural_\theta N)_{k,l} = \sum_p \sum_q M_{p,q} \cdot N_{k-p,l-q} \cdot e^{2\pi i\theta(k-p)\cdot q}$$

Clearly $|M\natural_\theta N|_{k,l} \leq (|M| * |N|)_{k,l}$ and so some properties from the (normal) convolution can be extended to the twisted convolution.

Theorem 3.2.27 *The product of two Gabor-type matrices S_1,S_2 again is a Gabor-type matrix*

$$S_3 = S_1 \cdot S_2$$

Let J_i be the Janssen matrix of S_i, then

$$J_3^* = J_2^* \natural_{\frac{L}{a\cdot b}} J_1^*$$

Proof: Gabor-type matrices are a-circulant-block matrices and matrices with b diagonal blocks. Following Lemma 3.2.22 and Lemma 3.2.14 their product is, too.

According to Corollary 3.2.24 there are $c_{k,l}$,$d_{k,l}$ such that

$$S_1 = \sum_{k_1=0}^{a-1} \sum_{l_1=0}^{b-1} c_{k_1,l_1} M_{k_1\tilde{a}} T_{l_1\tilde{b}}$$

$$S_2 = \sum_{k_2=0}^{a-1} \sum_{l_2=0}^{b-1} d_{k_2,l_2} M_{k_2\tilde{a}} T_{l_2\tilde{b}}$$

Then

$$S_3 = S_1 \cdot S_2 = \sum_{k_1,k_2=0}^{a-1} \sum_{l_1,l_2=0}^{b-1} c_{k_1,l_1} \cdot d_{k_2,l_2} M_{k_1\tilde{a}} T_{l_1\tilde{b}} \cdot M_{k_2\tilde{a}} T_{l_2\tilde{b}} =$$

$$\overset{Cor.3.2.4}{=} \sum_{k_1,k_2=0}^{a-1} \sum_{l_1,l_2=0}^{b-1} c_{k_1,l_1} \cdot d_{k_2,l_2} \cdot i_L^{-k_2\cdot\tilde{a}\cdot l_1\cdot\tilde{b}} M_{k_1\tilde{a}} \cdot M_{k_2\tilde{a}} \cdot T_{l_1\tilde{b}} \cdot T_{l_2\tilde{b}} =$$

$$= \sum_{k_1,k_2=0}^{a-1} \sum_{l_1,l_2=0}^{b-1} c_{k_1,l_1} \cdot d_{k_2,l_2} \cdot e^{-\frac{2\pi i k_2\cdot\tilde{a}\cdot l_1\cdot\tilde{b}}{L}} M_{(k_1+k_2)\tilde{a}} \cdot T_{(l_1+l_2)\tilde{b}} =$$

Let $k_3 = k_1 + k_2$ and $l_3 = l_1 + l_2$, then

$$= \sum_{k_2=0}^{a-1} \sum_{l_2=0}^{b-1} \sum_{k_3=k_2}^{a-1+k_2} \sum_{l_3=l_2}^{b-1+l_2} c_{k_3-k_2,l_3-l_2} \cdot d_{k_2,l_2} \cdot e^{-\frac{2\pi i k_2\cdot\tilde{a}\cdot(l_3-l_2)\cdot\tilde{b}}{L}} M_{k_3\tilde{a}} \cdot T_{l_3\tilde{b}} =$$

In this chapter the matrices are regarded as periodic in rows and columns and the factor $e^{-\frac{2\pi i k_2 \cdot \tilde{a} \cdot (l_3 - l_2) \cdot \tilde{b}}{L}}$ is b-periodic in l_3, so

$$\sum_{k_2=0}^{a-1} \sum_{l_2=0}^{b-1} \sum_{k_3=0}^{a-1} \sum_{l_3=0}^{b-1} c_{k_3-k_2, l_3-l_2} \cdot d_{k_2,l_2} \cdot e^{-\frac{2\pi i k_2 \cdot \tilde{a} \cdot (l_3 - l_2) \cdot \tilde{b}}{L}} M_{k_3 \tilde{a}} \cdot T_{l_3 \tilde{b}} =$$

$$\sum_{k_3=0}^{a-1} \sum_{l_3=0}^{b-1} \left(\sum_{k_2=0}^{a-1} \sum_{l_2=0}^{b-1} c_{k_3-k_2, l_3-l_2} \cdot d_{k_2,l_2} \cdot e^{-\frac{2\pi i k_2 \cdot \tilde{a} \cdot (l_3 - l_2) \cdot \tilde{b}}{L}} \right) M_{k_3 \tilde{a}} \cdot T_{l_3 \tilde{b}}$$

Let J_i be the Janssen matrix of S_i. The time-frequency shifts are an orthogonal system, so for $S_3 = \sum\limits_{k_3=0}^{a-1} \sum\limits_{l_3=0}^{b-1} J_{k_3,l_3} M_{k_3 \tilde{a}} T_{l_3 \tilde{b}}$ we know

$$J_{k_3,l_3} = \sum_{k_2=0}^{a-1} \sum_{l_2=0}^{b-1} c_{k_3-k_2, l_3-l_2} \cdot d_{k_2,l_2} \cdot e^{-\frac{2\pi i k_2 \cdot \tilde{a} \cdot (l_3 - l_2) \cdot \tilde{b}}{L}} =$$

$$= \sum_{k_2=0}^{a-1} \sum_{l_2=0}^{b-1} \overline{(J_2^*)}_{l_2,k_2} \cdot \overline{(J_1^*)}_{l_3-l_2,k_3-k_2} \cdot \overline{e^{\frac{2\pi i k_2 \cdot \tilde{a} \cdot (l_3 - l_2) \cdot \tilde{b}}{L}}} =$$

$$= \overline{\left(J_2^* \natural_{\frac{L}{a \cdot b}} J_1^* \right)}_{l_3,k_3}.$$

□

Point 2 stresses the connection of the Janssen matrix to the spreading function, see Section 2.3.2. In [43] Lemma 7.6.5 an analogue result was proved for the continuous case and the spreading function.

3.2.6 The Walnut And Janssen Norms

In Sections 3.1.2.4 and 3.1.2.1 we have found two types of "small" matrices (with $b \cdot a$ elements) which characterize Gabor-type matrices. In Section 3.1.1.4 we have stated that the calculation of the operator norm is numerically not very efficient. So we will define new norms using the smaller matrices. We will show that they are upper bounds for the operator norm. We will also study the relationship between these norms. Note the definition of the mixed norms in Section A.3.5.1 $\|A\|_{p,q} = \left(\sum\limits_{j=0}^{m-1} \left(\sum\limits_{i=0}^{n-1} |a_{i,j}|^p \right)^{\frac{q}{p}} \right)^{\frac{1}{q}}$ with a natural extension to infinity.

Definition 3.2.7 *Let S be a Gabor-type matrix, B be its non-zero block matrix and J be its Janssen matrix. Then, we define*

$$\|S\|_{Wal} = \left\|B^T\right\|_{\infty,1}$$

the **Walnut-norm** *and*

$$\|S\|_{Jan} = \|J\|_{1,1}$$

the **Janssen-norm**.

Using Definition 3.1.3 we see that

$$\|S\|_{Wal} = \left\|B^T\right\|_{\infty,1} = \sum_{i=0}^{b-1} \max_{j=0,\ldots,a-1} \{|B_{i,j}|\} = \sum_{i=0}^{b-1} \max_{j=0,\ldots,a-1} \left\{\left|S_{i\cdot\tilde{b}+j,j}\right|\right\}$$

and so the walnut norm takes the maxima of the side-diagonals and sums them up.

As the matrices B and J are smaller than the Gabor-type matrix S, the computation of the norms above is relatively simple. More precisely, the Walnut norm, but also the Janssen norm can be calculated from the block matrix see Theorem 3.2.28.

In Section 3.2.6.2 it will be shown that the norms above are bounds for the operator norm, and that in the Gabor frame case they can be ordered as follows:

$$\|S\|_{Op} \leq \|S\|_{Wal} \leq \|S\|_{Jan} \leq \|S\|_{fro} \tag{3.10}$$

This means that the Walnut norm is the best approximation of the operator norm, and therefore it can be used as an efficient way to find a (close) upper bound for it.

On the other hand, the Janssen matrix and norm give us some insight on the behavior in the time-frequency plane For example in the case of matrix approximation the Janssen representation gives some insight where in the time-frequency plane the difference between original and approximation matrix occurs, see Section 3.4.3.4. In Section 3.4.3, the numerical investigation of double preconditioning, all the algorithms use the block structure of the frame matrix. In that section, the Walnut and Janssen norms are very convenient as they can be calculated directly from the block matrix.

Regarding the Frobenius norm or equivalently the Frobenius inner product $M_{m,n}$ forms a Hilbert space. Although it is not a very close approximation for the operator norm, as can be seen e.g. in Section 3.4.3.2, the Hilbert space property is very useful from the analytical point of view.

In summery, each one of the norms introduced above has its usefulness. As we will work with finite-dimensional spaces, all norms have to be equivalent, see the next section.

3.2.6.1 The Connection Of The Janssen And Non-Zero Block Matrix

In this section the non-zero block matrix and the Janssen matrix are investigated in more detail. Using Corollary 3.2.24 we know that these matrices are connected. Even more, we can give a algorithm, how to switch between the Janssen matrix and the non-zero block matrix:

Theorem 3.2.28 *Let $B_{g,\gamma,a,b}$ be the $b \times a$ associated non-zero block matrix for g, γ, a, b, and $J_{g,\gamma,a,b}$ the corresponding Janssen-matrix. Then*

$$F_a \cdot B^t_{g,\gamma,a,b} = a \cdot J_{g,\gamma,a,b}$$

$$\|B\|_{fro} = \sqrt{a}\,\|J\|_{fro}$$

and therefore for the corresponding frame matrix S

$$\|S\|_{fro} = \sqrt{L}\,\|J\|_{fro}$$

Proof:

$$J_{k,l} = \frac{L}{ab}\langle \gamma, M_{k\tilde{a}} T_{l\tilde{b}} g\rangle = \frac{1}{a}\tilde{b}\left(\mathcal{V}_g \gamma\right)(l\tilde{b}, k\tilde{a}) =$$

$$= \frac{1}{a}\tilde{b}(\widehat{\gamma \cdot T_{l\tilde{b}}\overline{g}})\,(k\tilde{a})$$

Let us look at the l-th row of B $B^{(l)} \in \mathbb{C}^a$ with $B^{(l)}_j = B_{l,j}$.

$$\hat{b^{(l)}}_k \overset{(3.3)}{=} \tilde{b}\left(\widehat{\sum_{p=0}^{\tilde{a}-1} T_{ap}\left(T_{l\tilde{b}}\overline{g}\cdot\gamma\right)}\right)_k \overset{Poisson}{=} \tilde{b}\cdot(\widehat{T_{l\tilde{b}}\overline{g}\cdot\gamma})_{k\tilde{a}}$$

Note that we start with a Fourier transform in $\mathbb{C}^a$ but end up in $\mathbb{C}^n$ in this equation.

$$\Longrightarrow a \cdot J_{k,l} = \hat{b^{(l)}}_k$$

$$\|B\|_{fro} = \sqrt{\sum_{i=0}^{b-1}\sum_{j=0}^{a-1}|B_{i,j}|^2} = \sqrt{\sum_{i=0}^{b-1}\|b^{(i)}\|^2} =$$

$$= \sqrt{\sum_{i=0}^{b-1}\frac{1}{a}\left\|\widehat{b^{(i)}}\right\|^2} = \sqrt{\sum_{i=0}^{b-1}\frac{1}{a}\cdot\sum_{j=0}^{a-1}a^2\,|J_{i,j}|^2} = \sqrt{a}\cdot\|J\|_{fro}$$

As S consists of $\frac{L}{a}$ rotated versions of the $n \times a$ block-matrix and this larger block-matrix has the same Frobenius norm as the non-zero block matrix, clearly

$$\|S\|_{fro} = \sqrt{\frac{L}{a}} \cdot \|B\|_{fro} \tag{3.11}$$

and therefore

$$\|S\|_{fro} = \frac{\sqrt{L}}{\sqrt{a}} \cdot \sqrt{a} \cdot \|J\|_{fro} = \sqrt{L} \cdot \|J\|_{fro}$$

□

3.2.6.2 Norm Equivalence

In this section we will investigate the norm equivalences for the norm introduced above. For a better overview we will split the results in several statements and propositions.

Lemma 3.2.29 *1.*

$$\|S\|_{Op} \leq \|S\|_{Jan}$$

2.

$$\|S\|_{Wal} \leq \|S\|_{Jan}$$

3.

$$\|S\|_{fro} \leq \sqrt{L}\, \|S\|_{Jan}$$

Proof: We know from Corollary 3.2.24 that we can represent the frame operator as sum of time and frequency shifts and so for every norm

$$\|S_{g,\gamma}\| = \frac{L}{a \cdot b} \left\| \sum_{l=0}^{b-1} \sum_{k=0}^{a-1} c_{l,k} M_{l\tilde{a}} T_{k\tilde{b}} \right\| \leq$$

$$\leq \frac{L}{a \cdot b} \sum_{l=0}^{b-1} \sum_{k=0}^{a-1} |c_{l,k}| \, \|M_{l\tilde{a}} T_{k\tilde{b}}\| =$$

$$= \sum_{l=0}^{b-1} \sum_{k=0}^{a-1} |J_{l,k}| \, \|M_{l\tilde{a}} T_{k\tilde{b}}\|$$

We know from Proposition 3.2.3 that $\|M_{l\tilde{a}} T_{k\tilde{b}}\|_{Op} = 1$ and $\|M_{l\tilde{a}} T_{k\tilde{b}}\|_{fro} = \sqrt{L}$ points 1) and 3) are proved.

Point 2) is also true, as we can show that $\|M_{l\tilde{a}}T_{k\tilde{b}}\|_{Wal} = 1$ because this matrix has only one non-zero side-diagonal, where the entries have all norm 1.

$\square$

Lemma 3.2.30

$$\|S\|_{Op} \leq \|S\|_{Wal}$$

Proof:

$$\|S\|_{Op} = \max_{x:\|x\|_2=1} \{\|Sx\|_2\}$$

Let b_p in $\mathbb{C}^n$ with $(b_p)_j = B_{p,j \bmod a}$. We know from (3.4) that

$$\|Sx\|_2 = \left\| \sum_{p=0}^{b-1} T_{-p\tilde{b}} x \cdot b_p \right\|_2 \leq$$

$$\leq \sum_{p=0}^{b-1} \left\| T_{-p\tilde{b}} x \cdot b_p \right\|_2 \leq \sum_{p=0}^{b-1} \left\| T_{-p\tilde{b}} x \right\|_2 \cdot \|b_p\|_\infty =$$

$$= \sum_{p=0}^{b-1} \|x\|_2 \cdot \max_{j=0,\ldots,a-1} \left\{ B_{p,j \bmod a} \right\} =$$

$$= \|x\|_2 \cdot \underbrace{\sum_{p=0}^{b-1} \max_{j=0,\ldots,a-1} \left\{ B_{p,j \bmod a} \right\}}_{\|S\|_{Wal}}$$

$\square$

Lemma 3.2.31

$$\|J\|_{fro} \leq \|J\|_{1,1} \leq \sqrt{a \cdot b}\, \|J\|_{fro}$$

Proof: This is just an analogous property to the norm equivalence for $\|.\|_2$ and $\|.\|_1$ in $\mathbb{C}^n$: $\|x\|_2 \leq \|x\|_1 \leq \sqrt{n}\, \|x\|_2$. $\square$

Proposition 3.2.32

$$\sqrt{\frac{L}{a \cdot b}}\, \|S\|_{Jan} \leq \|S\|_{fro} \leq \sqrt{L} \cdot \|S\|_{Jan}$$

Proof: We know the second part from Lemma 3.2.29.

$$\|S\|_{Jan} = \|J\|_{1,1} \leq \sqrt{a \cdot b}\,\|J\|_{fro} \overset{Lem.3.2.28}{=} \sqrt{\frac{a \cdot b}{L}}\,\|S\|_{fro}$$

□

So if $red = \frac{L}{a \cdot b} \geq 1$, what we always need for the used Gabor system to be a frame, then $\|S\|_{Jan} \leq \|S\|_{fro}$ and so the walnut norm approximates the operator norm better.

Lemma 3.2.33

$$\frac{1}{\sqrt{a}}\,\|B\|_{fro} \leq \|B\|_{\infty,1} \leq \sqrt{b}\,\|B\|_{fro}$$

Proof:

$$\|B\|_{\infty,1} = \sum_{i=0}^{b-1} \max_{j=0,\ldots,a-1} \{|B_{i,j}|\} = (*)$$

Clearly

$$\max_{j=0,\ldots,a-1} \{|B_{i,j}|\} \leq \sqrt{\sum_{j=0}^{a-1} |B_{i,j}|^2}$$

and

$$(*) \leq \sum_{i=0}^{b-1} \sqrt{\sum_{j=0}^{a-1} |B_{i,j}|^2} \leq \sqrt{b}\sqrt{\sum_{i=0}^{b-1}\sum_{j=0}^{a-1} |B_{i,j}|^2}$$

On the other hand

$$\sqrt{\sum_{j=0}^{a-1} |B_{i,j}|^2} \leq \sqrt{a} \cdot \max_{j=0,\ldots,a-1} \{|B_{i,j}|\} \Longrightarrow$$

$$\sqrt{\sum_{i=0}^{b-1}\sum_{j=0}^{a-1} |B_{i,j}|^2} \leq \sum_{i=0}^{b-1} \sqrt{\sum_{j=0}^{a-1} |B_{i,j}|^2} \leq$$

$$\leq \sum_{i=0}^{b-1} \sqrt{a} \cdot \max_{j=0,\ldots,a-1} \{|B_{i,j}|\} = \sqrt{a} \cdot (*)$$

□

With Theorem 3.2.28 we get immediately

Proposition 3.2.34

$$\frac{\sqrt{L}}{a \cdot b} \|S\|_{Wal} \leq \|S\|_{fro} \leq \sqrt{L} \|S\|_{Wal} \quad .$$

Combining Lemma 3.2.29, Proposition 3.2.32, Lemma 3.2.33 and Theorem 3.2.28 we get

Proposition 3.2.35

$$\frac{1}{\sqrt{a \cdot b}} \|S\|_{Jan} \leq \|S\|_{Wal} \leq \|S\|_{Jan}$$

So in combination we get Equation 3.10:

Theorem 3.2.36

$$\|S\|_{Op} \leq \|S\|_{Wal} \leq \|S\|_{Jan} \leq \|S\|_{fro}$$

3.3 Some Thoughts On 'Numerical Algorithms For Discrete Gabor Expansions' [122]

In [44] chapter 8 [122] T. Strohmer has written a celebrated article about "numerical algorithms for discrete Gabor expansions", which introduced an efficient algorithm for inverting the Gabor frame matrix. It was also the starting point for the investigation of the block matrices and the Fourier matrix transformation in Section 3.2, and Double Preconditioning for Gabor frames in Section 3.4. Most of the work in this chapter is based on this article, which is a very good entry point to the theory of finite dimensional Gabor analysis.

In this article there are some small errors, which can be easily eradicated. This is done in this section.

The translation and modulation are used here in this section only as in this article, so

$$\left(T_k^{(L)} f\right)_l = f_{(l+k) mod L}$$

and

$$(M_p f)_l = e^{-2\pi i p l / L} f_l = \omega_L^{pl} f_l$$

We also use other conventions of this article, like $a \cdot N = L$ and $b \cdot M = L$.

3.3.1 ad [122] 8.3.4.

We will cite the original wording from [122] between horizontal lines, using the original numbering.

Proposition 8.3.4 *The matrix G can be factorized into a block diagonal matrix D_G with M rectangular blocks W_k of size $b \times N$ via*

$$D_G = P^*_{M,L} G \left(I_N \otimes F^*_M\right) P^*_{N,MN}$$

where

$$D_G = diag\left(W_0, \ldots, W_{M-1}\right)$$

with

$$\left(W_k\right)_{mn} = \sqrt{M} g\left(k + mM - na\right)$$

for $m = 0, \ldots, b-1$,$n = 0, \ldots, N-1$ and $k = 0, \ldots, M-1$.

The proof is based on the fact that $\left(G\left(I_N \otimes F^*_M\right)\right)_{m,n} = \sqrt{M} g\left(n - aM\right)$, if $|m - n|\, modM = 0$ and 0 else, [.....]

'The fact' is not true, it should be

$$\left(G\left(I_N \otimes F^*_M\right)\right)_{m,n} =$$

$$= \begin{cases} \sqrt{M} g\left(m - \left\lfloor \frac{n}{M} \right\rfloor a\right) & \text{if } |m-n|\, modM = 0 \\ 0 & \text{otherwise} \end{cases}$$

But the proposition stays completely true, except that D_G should rather be $D_G = P^*_{M,L} G\left(I_N \otimes F^*_M\right) P_{N,MN}$.

Proof.:

$$G = \begin{pmatrix} | & | & & | & \\ g_{0,0} & g_{1,0} & \ldots & g_{m,n} & \ldots \\ | & | & & | & \end{pmatrix}$$

and so

$$G_{p,q} = \omega_M^{q_2 \cdot p} \cdot g(p - q_1 a)$$

where $q_1 = \lfloor \frac{q}{M} \rfloor$ and $q_2 = q \bmod M$. Let $l_1 = \lfloor \frac{l}{M} \rfloor$ and $l_2 = l \bmod M$.

$$\left(I_N \otimes F^*_M\right)_{q,l} = \delta_{q_1 l_1} \cdot \frac{1}{\sqrt{M}} \cdot \omega_M^{-l_2 q_2}$$

And so

$$\left(G\left(I_N \otimes F^*_M\right)\right)_{p,l} = \sum_{q=0}^{MN-1} G_{p,q} \cdot \left(I_N \otimes F^*_M\right)_{q,l} =$$

$$= \sum_{q_1=0}^{N-1} \sum_{q_2=0}^{M-1} \omega_M^{q_2 \cdot p} \cdot g(p - q_1 a) \delta_{q_1 l_1} \cdot \frac{1}{\sqrt{M}} \cdot \omega_M^{-l_2 q_2} =$$

$$= \frac{1}{\sqrt{M}} \cdot g(p - l_1 a) \underbrace{\sum_{q_2=0}^{M-1} \omega_M^{q_2 \cdot (p - l_2)}}_{= M \delta_{p_2 l_2}} = \delta_{p_2 l_2} \sqrt{M} g(p - l_1 a)$$

with $p_1 = \lfloor \frac{p}{M} \rfloor$ and $p_2 = p \bmod M$. With this property it is clear, that $G\left(I_N \otimes F_M^*\right)$ is a matrix with b diagonal $M \times M$-blocks and so the proof of 8.3.1 can be used analogously.

$$G\left(I_N \otimes F_M^*\right) = \sum_{k=0}^{M-1} \underbrace{W_k}_{b \times N} \otimes E_k^{(M)} \text{ with}$$

$$\left(W_k\right)_{m,n} = \left(G\left(I_N \otimes F_M^*\right)\right)_{k+m \cdot M, k+n \cdot M} =$$

$$= \delta_{kk} \sqrt{M} g((k + m \cdot M) - na)$$

□

3.3.2 ad [122] 8.4.1.

Proposition 8.4.1 *Given a function g of length L and lattice parameters a, b. Denote the greatest common divisor (gcd) of a and M by c. Let W_k be the submatrices of the block diagonal factorization of G stated in proposition 8.3.4, i.e. $\left(W_k\right)_{mn} = Mg\left(k + mM - na\right)$. Then the W_k satisfy following relation:*

$$\Pi_b^q W_k \Pi_N^l = W_{(k+la) mod M} (8.4.1)$$

with $q = \lfloor \frac{lab}{L} \rfloor$. Thus there are (up to permutations) c different submatrices W_k for $k = 0, M-1$.

There is a small typing error as $\left(W_k\right)_{mn} = \sqrt{M} g\left(k + mM - na\right)$. The calculation of q is not right and in the proof the property $(k + la) \bmod M = k + (la) \bmod M$ is used, which is not true.

Let $q = \left(- \left\lfloor \frac{kb + lab}{L} \right\rfloor \bmod b\right)$, then again the proposition is true, although the q now also depends on the k.

So it should finally be:

Proposition 8.4.1' *Given a function g of length L and lattice parameters a, b. Denote the greatest common divisor (gcd) of a and M by c. Let W_k be the submatrices of the block diagonal factorization of G stated in proposition*

8.3.4, i.e. $(W_k)_{mn} = \sqrt{M} g(k + mM - na)$. *Then the* W_k *satisfy following relation:*

$$\Pi_b^q W_k \Pi_N^l = W_{(k+la) mod M} (8.4.1)$$

with $q = -\left\lfloor \frac{k+la}{M} \right\rfloor$ *mod b. Thus there are (up to permutations) c different submatrices* W_k *for* $k = 0, M-1$.

Proof.:

Choose a q such that for $k = 0, \ldots, M-1$, $m = 0, \ldots, b$, $n = 0, \ldots, N-1$ and $l = 0, \ldots, N-1$

$$\left(\Pi_b^q W_k \Pi_N^l\right)_{m,n} = \left(W_{(k+la) modM}\right)_{m,n}$$

$$\iff (W_k)_{m+q,n-l} = \left(W_{(k+la) modM}\right)_{m,n}$$

$$\iff g(k + (m+q) M - (n-l) a) =$$

$$= g((k + la) \mod M + mM - na)$$

This is valid for all g if and only if

$$k + (m+q) M - (n-l) a \equiv_L (k + la) \mod M + mM - na$$

$$\iff k + la + qM \equiv_L (k + la) \mod M$$

$$\iff \left\lfloor \frac{k+la}{M} \right\rfloor M + qM \equiv_L 0$$

$$\iff \left\lfloor \frac{k+la}{M} \right\rfloor + q \equiv_b 0$$

$$\iff q = -\left\lfloor \frac{k+la}{M} \right\rfloor \mod b$$

□

3.3.3 ad [122] 8.4.2.

Proposition 8.4.2 *Given a function g of length L and lattice parameters a, b. Denote* $d = gcd(b, N)$. *Then* W_k *is a block circulant matrix with d generating blocks of size* $\frac{b}{d} \times \frac{N}{d}$ *for* $k = 0, \ldots, M-1$.

For this proposition the following proof might be easier to follow:

Proof.:

$$(W_k)_{m,n} = (W_k)_{\left(m+\frac{b}{d}\right) \bmod b, \left(n+\frac{N}{d}\right) \bmod N}$$

if and only if

$$g\left(k+mM-na\right)=$$
$$=g\left(k+\left(\left(m+\frac{b}{d}\right) \bmod b\right)M-\left(\left(n+\frac{N}{d}\right) \bmod N\right)a\right)$$

This is true for all g if and only if

$$\begin{aligned} mM-na= \\ =\left(\left(m+\tfrac{b}{d}\right) \bmod b\right)M-\left(\left(n+\tfrac{N}{d}\right) \bmod N\right)a \end{aligned} \tag{3.12}$$

We know that for all $m_1=0,\dots,L-1$

$$m_1=\lfloor\frac{m_1}{b}\rfloor\cdot b+(m_1 \bmod b)$$

$$\Longrightarrow m_1\cdot M=\lfloor\frac{m_1}{b}\rfloor\cdot\underbrace{b\cdot M}_{=L}+(m_1 \bmod b)\cdot M$$

$$\Longrightarrow m_1\cdot M\equiv_L (m_1 \bmod b)\cdot M$$

Therefore, and with an equivalent argument for a and N, (3.12) is true if and only if

$$mM-na\equiv_L\left(m+\frac{b}{d}\right)M-\left(n+\frac{N}{d}\right)a$$

$$\Longleftrightarrow 0\equiv_L\frac{b}{d}M-\frac{N}{d}a=\frac{L}{d}-\frac{L}{d}$$

□

3.4 Double Preconditioning For Gabor Frames

An important question, which we will investigate in this section, is how to find a Gabor analysis-synthesis system with perfect (or depending on the application a satisfactorily accurate) reconstruction in a numerical efficient way. Basic Gabor frame theory, see Chapter 2.1.2, tells us, that when using the canonical *dual* Gabor atom $\tilde{g}=S^{-1}g$, perfect reconstruction is always achieved, if the *frame-operator* S (cf. Section 1.1) is invertible. Thus the dual atom is obtained by solving the equation $S\tilde{g}=g$, and to this end the frame algorithm, see Proposition 1.2.13, can be applied. This is a Neumann algorithm, see Figure A.1, with a relaxation parameter. If the inequality $\left\|Id-\lambda S\right\|_{Op}<1$ holds, then this algorithm converges, S is invertible and the algorithm approximates the dual Gabor atom $\tilde{g}$.

Instead of finding the canonical dual, other dual windows can be searched, and sometimes they can be found in a numerically more efficient way as demonstrated in [130]. But in general, the computation of a dual window can be very complicated and numerically inefficient. The Zak transform, cf. [136] [73], is extensively used for theoretical purposes, but its use for numerical calculations is limited [122]. The celebrated paper from Wexler and Raz [131] gives an important bi-orthogonality relation, which reduces the problem to a simple linear system. In order to find a very efficient algorithm, Qiu and Feichtinger use the sparse structure of the frame operator [103], which leads them to an algorithm for the inversion of the frame matrix with $\mathcal{O}(abn)$ operations, where n is the signal length and a, b are the time and frequency shift parameters.

In this section another well known tool to speed up the convergence rate, namely *preconditioning*, cf. Section 3.1.1.3, is used to further improve the numerical efficiency of this calculation. In our proposed method, we use a special invertible preconditioning matrix P, which makes $\|Id - PS\|$ small. Then, instead of $S\tilde{g} = g$, the equation $PS\tilde{g} = Pg$ is solved. So the matrix $M = P \cdot S$ is intended to be an approximation of the identity. If M is a reasonably good approximation, e.g. $\|Id - M\| < 0.1$, then only a few iterations are needed in order to find the true dual atom (up to precision limitations). Moreover, if M is a very good approximation, e.g., $\|Id - M\| \ll 0.1$, then the preconditioning matrix P can already be considered a good approximation of the inverse matrix of S.

The aim of this section is to investigate the idea of *double preconditioning* of the frame operator S. This method was already suggested as an idea in the very last paragraphs of [122] and [124]. In this section the double preconditioning method will be fully developed, examined and backed up with systematic experimental numerical data. This scheme relies, again, on the very special structure of the Gabor frame operator S, it is an a block-circulant matrix with b diagonal blocks, refer also to Section 3.2.5. From Theorem 3.1.9 we know, that there are two extreme cases for this nice structure. (1) If the frequency sampling is dense enough and g has support inside an interval I, with the length $\leq b$, then S is a diagonal matrix. (2) If the time sampling is dense enough and $\hat{g}$ has compact support on an interval with length $\leq a$, then $\hat{S}$ is diagonal and therefore S is circulant. In both cases it is easy to find the inverse matrix. If the window g is not supported on I, then S becomes non-diagonal. However, if S is strictly diagonal dominated it is well known for $D = (d_{i,j})$, with $d_{i,j} = \delta_{i,j} s_{i,j}$ i.e. the best approximation of S by diagonal matrices, S^{-1} can be approximated well by using the preconditioning matrix $P = D^{-1}$, see the *Jacobi* method in Section 3.1.1.2. An analogous property holds if $\hat{S}$ is strictly diagonal dominated, obtaining a circulant matrix as preconditioning matrix. When using these two preconditioning matrices at the same time, hence the name *double preconditioning*, we will get a new method.

The main observation is the fact that the use of double preconditioning often leads to better results than the use of single preconditioning. Moreover, in the cases where this is not true, the difference is in general not significant. This behavior is observed in numerical experiments. More precisely, we will first study single cases and then proceed with systematic experiments, where the efficiency of the double preconditioning method is investigated for different windows.

In Section 3.4.1 we will review and extend the use of diagonal and circulant matrices as preconditioners for the Gabor frame operator. In Section 3.4.2 we will explain how to combine these preconditioners to invert the frame matrix S, and finally, in Section 3.4.3, we will demonstrate the efficiency of this idea.

3.4.1 Single Preconditioning Of The Gabor Frame Operator

We propose two preconditioning methods. In the first we consider the best approximation of S with diagonal matrices, and approximate S^{-1} by inverting its diagonal approximation. The second method is based on the same idea but considering circulant matrices.

3.4.1.1 Diagonal Matrices

The inverse of the diagonal part of the frame operator is used as a preconditioning matrix, depicted in Figure 3.2.

$$P = D(S)^{-1}$$

Figure 3.2: The diagonal preconditioning matrix

For every square matrix A we can find a diagonal matrix just by "cutting out" the diagonal part of A, to shorten the notation we use $D(A)$ instead of $\mathfrak{Diag}(A)$:

Definition 3.4.1 *Let* $A = (a_{i,j})_{i,j}$ *be a square* $n \times n$ *matrix, then let* $D(A) = (d_{i,j})_{i,j}$ *with* $d_{i,j} = \begin{cases} a_{i,i} & i = j \\ 0 & otherwise \end{cases}$ *the* diagonal part of A.

The set of all diagonal $n \times n$ matrices forms a matrix algebra. This algebra is spanned by the matrices E_k with $E_k = D(\delta_k)$. They clearly form an orthonormal basis (ONB) (with the Frobenius inner product) and therefore $D : A \mapsto D(A)$ is an orthogonal projection. This means that the best approximation of A in $\|.\|_{HS}$ by diagonal matrices is exactly $D(A)$.

The diagonal part of a Gabor-type matrix clearly is block-circulant, and therefore also of Gabor-type. This allows us to use the efficient block-matrix algorithms from [104] refer also to Section 3.2.5.2.

If the window g is compactly supported on an interval with a length smaller than $\tilde{b}$ then $S_{g,g}$ is a diagonal matrix, see Section 3.1.2.3. In this case the inverse matrix is very easy to calculate by just taking the reciprocal value of the diagonal entries, which are always non-zero for a Gabor frame matrix, cf. Lemma 3.1.6.

Even in the case where the window g is not compactly supported, but S is strictly diagonal dominant, S^{-1} is well approximated by D^{-1}. It is known [121] that, if the matrix A is *strictly diagonal dominant*, i.e.

$$\max_{i=0,\dots,n-1} \sum_{k=0,k\neq i}^{n-1} \frac{|a_{ik}|}{|a_{ii}|} < 1,$$

then the *Jacobi algorithm*, $x_m = D^{-1}(D - A)\, x_{m-1} + D^{-1}b$, converges for every starting vector x_0 to $A^{-1}b$, see Section 3.1.1.2. The efficiency of the Jacobi algorithm comes from the fact that it is easy to find the diagonal part of a matrix and to invert it. As can be seen from the above formula the Jacobi algorithm is equivalent to preconditioning with $D(S)^{-1}$.

The use of block-matrices leads to very efficient algorithms. Motivated by this fact, we would like to find criteria for the convergence of the Jacobi algorithm for non-zero block matrices, which means that by just using the diagonal preconditioning matrix and an iterative scheme we will get the inverse matrix and the canonical dual window respectively.

Corollary 3.4.1 *Let S be a Gabor-type matrix and B be the associated non-zero block matrix. Then the following conditions are sufficient for the Jacobi-algorithm to converge*

1. $$\max_{i=0,\dots,a-1} \left\{ \sum_{k=1}^{b-1} \frac{\left|B_{k,i-k\cdot\tilde{b}}\right|}{B_{0,i}} \right\} < 1$$

2. $$\max_{j=0,\dots,a-1} \left\{ \sum_{k=1}^{b-1} \frac{\left|B_{k,j}\right|}{B_{0,j+k\cdot\tilde{b}}} \right\} < 1$$

3. $$\sum_{i=0}^{a-1} \sum_{k=1}^{b-1} \left(\frac{\left|B_{k,i-k\cdot\tilde{b}}\right|}{B_{0,i}} \right)^2 < \frac{a}{L}$$

4. $\sum_{j=0}^{a-1} \sum_{k=1}^{b-1} \left(\frac{\left|B_{k,j}\right|}{B_{0,j+k\cdot\tilde{b}}} \right)^2 < \frac{a}{L}$

Proof: We are just going to insert Corollary 3.1.5

$$S_{i,j} = Ш_{\tilde{b}}(i-j) B_{\lfloor \frac{i-j}{\tilde{b}} \rfloor, j}$$

into Theorem 3.1.3.

For the Jacobi algorithm the quotient $\frac{|S_{i,j}|}{|S_{i,i}|}$ is important. $S_{i,j} \neq 0$ only for i, j such that $i - j \bmod \tilde{b} = 0$. Let $\frac{i-j}{\tilde{b}} = k$. Then

$$\frac{|S_{i,j}|}{|S_{i,i}|} = \frac{\left|B_{k,i-k\cdot\tilde{b}}\right|}{|B_{0,i}|} = \frac{|B_{k,j}|}{\left|B_{0,j+k\cdot\tilde{b}}\right|}$$

Notice that the first column of B is always positive, as the diagonal of the Gabor frame operator has this property for frames, see Lemma 3.1.6.

So for point 1

$$\max_{i=0,\ldots,L-1} \sum_{\substack{j=0,\ldots,L-1\\ j\neq i}} \frac{|S_{i,j}|}{|S_{i,i}|} = \max_{i=0,\ldots,a-1} \sum_{k=1,\ldots,b-1} \frac{\left|B_{k,i-k\cdot\tilde{b}}\right|}{B_{0,i}} < 1$$

Point 2 can be proved in an analogous way.

For Point 3

$$\sum_{\substack{i,j=0,\ldots,L-1\\ j\neq i}} \left(\frac{|S_{i,j}|}{|S_{i,i}|} \right)^2 = \sum_{i=0,\ldots,L-1} \sum_{k=1,\ldots,b-1} \left(\frac{\left|B_{k,i-k\cdot\tilde{b}}\right|}{|B_{0,i}|} \right)^2 =$$

$$= \tilde{a} \sum_{i=0,\ldots,a-1} \sum_{k=1,\ldots,b-1} \left(\frac{\left|B_{k,i-k\cdot\tilde{b}}\right|}{B_{0,i}} \right)^2 < \tilde{a} \cdot \frac{a}{L} = 1$$

Point 4 can be proved in an analogous way.

□

3.4.1.2 Excursus: A Criterion For Gabor Frames

Not directly connected to our main question of finding an approximate dual window, the above mentioned algorithm nevertheless gives us some sufficient conditions for a Gabor system to generate a frame by using known criteria for the convergence of the Jacobi algorithm.

Corollary 3.4.2 *Sufficient conditions for a Gabor triple (g, a, b) to generate a Gabor frame are:*

1. $\left|\sum_{k=0}^{\tilde{a}-1} \overline{g}\,(i - ak)\, g\left(i - j\tilde{b} - ak\right)\right| < \frac{1}{b-1} \sum_{k=0}^{\tilde{a}-1} |g\,(i - ak)|^2$
   for all $i = 0, \dots, a-1$ *and* $j = 1, \dots, b-1$.

2. $\left|\sum_{k=0}^{\tilde{a}-1} \overline{g}\left(j - i\tilde{b} - ak\right) g\,(j - ak)\right| < \frac{1}{b-1} \sum_{k=0}^{\tilde{a}-1} \left|g\left(j + i\tilde{b} - ak\right)\right|^2$
   for all $j = 0, \dots, a-1$ *and* $i = 1, \dots, b-1$.

Proof: Using Equation 3.3 we know that

$$B_{i,j} = \tilde{b} \sum_{l=0}^{\tilde{a}-1} \overline{g}\left(i\tilde{b} + j - al\right) g\,(j - al)\,.$$

Under the assumption of point (1), we know that

$$\frac{\left|B_{k,i-k\cdot\tilde{b}}\right|}{B_{0,i}} = \frac{\tilde{b}\left|\sum_{l=0}^{\tilde{a}-1} \overline{g}\left(k\tilde{b} + i - k\cdot\tilde{b} - al\right) g\left(i - k\cdot\tilde{b} - al\right)\right)}{\tilde{b}\left|\sum_{l=0}^{\tilde{a}-1} \overline{g}\,(i - al)\, g\,(i - al)\right)} =$$

$$= \frac{\left|\sum_{l=0}^{\tilde{a}-1} \overline{g}\,(i - al)\, g\left(i - k\cdot\tilde{b} - al\right)\right|}{\sum_{l=0}^{\tilde{a}-1} |g\,(i - al)|^2} < \frac{1}{b-1}$$

And therefore

$$\max_{i=0,\dots,a-1} \left\{\sum_{k=1}^{b-1} \frac{\left|B_{k,i-k\cdot\tilde{b}}\right|}{B_{0,i}}\right\} < \max_{i=0,\dots,a-1} \left\{\sum_{k=1}^{b-1} \frac{1}{b-1}\right\} < 1$$

Therefore with Corollary 3.4.1 the Jacobi-algorithm is converging for S, therefore S is invertible, and therefore (g, a, b) forms a frame.

Point (2) can be shown by using a very similar argument. □

A similar result was stated in a corollary in [103], which is amended and expanded by this result.

$$P = C(S)^{-1}$$

Figure 3.3: The circulant preconditioning matrix

3.4.1.3 Circulant Matrices

Instead of considering diagonal matrices we can approximate S by projecting on the algebra of circulant matrices and using the inverse as preconditioning matrix.

Definition 3.4.2 *Let* $C(S) = (c_{i,j})_{i,j}$ *with* $c_{i,j} = \frac{1}{L}\sum_{k=0}^{L-1} S_{k+(j-i),k}$.

The matrix $C(S)$ is clearly a circulant matrix. It even is the best approximation of S by circulant matrices, as stated in the next result:

Corollary 3.4.3 *Let A be a matrix, then the best approximation (regarding the Frobenius norm) of A on the circulant matrices is to take the mean value of the side diagonals as entries of the circulant matrix.*

$$C(A) = \sum_k \left(\frac{1}{L} \sum_{j=0}^{L-1} A_{j+k,j} \right) \Pi_L^k \tag{3.13}$$

Proof: The circulant matrices are spanned by the Π_L^k, see Proposition 3.2.6. We know from Proposition 3.2.3 that the matrices $\frac{\Pi_L^k}{\sqrt{L}}$ form an orthonormal system, so

$$A = \sum_k \left\langle A, \frac{1}{\sqrt{L}} \cdot \Pi_L^k \right\rangle_{HS} \frac{1}{\sqrt{L}} \cdot \Pi_L^k = \frac{1}{L} \sum_k \left\langle A, \Pi_L^k \right\rangle_{HS} \cdot \Pi_L^k$$

$$\left\langle A, \Pi_L^k \right\rangle_{HS} = \sum_{i,j=0}^{L-1} A_{i,j} \cdot \delta_{i,j+k} = \sum_{j=0}^{L-1} A_{j+k,j}$$

□

Due to properties of the Matrix Fourier Transformation, see Section 3.2.3 $C(S)$ can be calculated by using

$$C(S) = \mathfrak{F}^{-1}\left(D\left(\mathfrak{F}\left(S\right)\right)\right) = \mathcal{F}_n \cdot \left[D\left(\mathcal{F}_n \cdot S \cdot \mathcal{F}_n^*\right)\right] \cdot \mathcal{F}_n^*$$

which implies that

$$C(S)^{-1} = \mathcal{F}_n^* \cdot [D(\mathcal{F}_n \cdot S \cdot \mathcal{F}_n^*)]^{-1} \cdot \mathcal{F}_n$$

Therefore the computation of $C(S)^{-1}$ can be done in a very efficient way by using the FFT-algorithm.

3.4.2 Double Preconditioning Of The Gabor Frame Operator

The main result of this Section is the double-preconditioning method. In a rather natural way, as seen in Figure 3.4, we will combine the two single preconditioning methods introduced above. More precisely, after an approximation with diagonal matrices and inversion we do an approximation with circulant matrices. The double preconditioning algorithm can be implemented very efficiently using the block multiplication algorithm of [103], since, if S is a Gabor-type matrix, then $C(S)$ and $D(S)$ are also Gabor-type matrices and hence can also be represented by $b \times a$ block matrices.

$$P = C\left(D(S)^{-1} \cdot S\right)^{-1} D(S)^{-1}$$

Figure 3.4: The double preconditioning matrix

For a basic description of the algorithm see figure 3.5. In this figure the subscript *'block'* indicates a calculation on the block matrix level, which makes this algorithm very efficient. Let us explain some of the expressions:

1. $block(g, a, b)$ stands for the calculation of the non-zero block matrix using Equation 3.3.

2. $diag_{block}(M)$ stands for the calculation of the block matrix of $D(M)$, which is done by calculating the block matrix of S and setting all rows but the first to zero, as this row corresponds to the main diagonal of S.

3. $circ_{block}(M)$ is the calculation of the block-matrix of $S(M)$ which has constant columns with the mean value of the columns of $block(g, a, b)$ as entries.

4. $inv_{block}(M)$ is the calculation of M^{-1} on the block level.

   (a) For diagonal M and therefore block matrices with non-zero entries in the first row only, this is done by using the reciprocal values of the entries in the first row.

   (b) For circulant M and therefore block matrices with constant rows, we use the first row of the block matrix, apply the inverse FFT to it, take the reciprocal values and apply the FFT. This is the inverse, because let M be a circulant matrix, then

$$M = \sum_k c_k \cdot T_k \Longrightarrow \mathfrak{F}^{-1}(M) = \sum_k c_k \cdot M_{-k}$$

   $\mathfrak{F}^{-1}(M)$ is a diagonal matrix with diagonal entries $L\left(\mathcal{F}_L^{-1}c\right)_k$, see Proposition 3.2.8. Therefore $\left(\mathfrak{F}^{-1}(M)\right)^{-1}$ has the diagonal entries $\frac{1}{L\cdot\left(\hat{d}\right)_k}$ and so

$$M^{-1} = \sum_k \left(\mathcal{F}_L\left(\frac{1}{\left(\mathcal{F}_L^{-1}c\right)}\right)\right)_k T_k$$

5. $\bullet_{block}$ is the matrix multiplication on block matrix level using Theorem 3.2.26.

In the subsequent two sections we will look at two special properties of our algorithm, how to do the second preconditioning step and in which order to multiply the matrices. We will justify, why we have chosen this particular setting.

For an implementation in MATLAB, see Section B.4.

3.4.2.1 Choice of method

Roughly speaking, the double preconditioning method consists of two single preconditioning steps. There are two possibilities, either, to use the original matrix S for every step or to use the result of the first step in the second one. More precisely:

(Method 1) $C\left(D(S)^{-1}S\right)^{-1}D(S)^{-1}$ or the more naive

(Method 2) $C(S)^{-1}D(S)^{-1}$

The first method seems to be more sensible, as each single preconditioning step uses projections. Even more, it also provides the following property: if S

- Parameter:

the window g, the lattice parameter a,b

- Initialization:

$$B = block(g, a, b)$$

- Preconditioning :

  - (first preconditioning)

$$P_1 = diag_{block}(B)$$

$$S_1 = inv_{block}(P_1) \bullet_{block} B$$

  - (second preconditioning)

$$P_2 = circ_{block}(S_1)$$

$$S_2 = inv_{block}(P_2) \bullet_{block} S_1$$

Figure 3.5: The double preconditioning algorithm

is diagonal, after the first step we will reach identity and this will stay identity in the second step (up to the machine precision). Also, if S is circulant, after the first step we still have a circulant matrix as the multiplication of an arbitrary matrix A and a diagonal matrix D is $D \cdot A = \left(d_{i,i} \cdot a_{i,j}\right)_{i,j}$. So for the circulant matrix $C = \left(c(i-j)\right)_{i,j}$ we get $\left(D^{-1}(C) \cdot C\right)_{i,j} = \left(c(0)^{-1} \cdot c(i-j)\right)_{i,j}$. Hence the second step leads to identity again. Note that the Gabor-type structure also is preserved with this method.

On the other hand, the second method does not enjoy aforementioned property in the case of circulant matrices. For example take $n = 6$, $a = 1$, $b = 6$ and $g = (1, 2, 3, 4, 5, 6)$. Then S is a circulant matrix, but the double preconditioning deteriorate the approximation, as

$$\left\| C(S)^{-1} D(S)^{-1} S - I \right\|_{Wal} = 0.994505.$$

This is a big disadvantage, since, for these simple matrices, the method should give satisfactory results.

So we always use the first method. In order to simplify the notation we will use $\mathcal{C}(S)$ to denote $C(D(S)^{-1}S)$.

3.4.2.2 Order of preconditioning matrices

If the preconditioning matrix is diagonal, it makes no difference if it is multiplied to S from the left or from the right. As S is self-adjoint (see Theorem 1.1.3), $D = D(S)$ is too, and therefore, $(D \cdot A)^* = A \cdot D$ and $(D \cdot A - I)^* = A \cdot D - I$. Finally

$$\|D \cdot A - I\| = \|A \cdot D - I\|$$

So the norm of the difference to the identity is equal for

1. $D(S)^{-1}S$ or
2. $SD(S)^{-1}$

The same property holds for single preconditioning with circulant matrices.

In the case of double-preconditioning, the influence of the order in the multiplication has still to be investigated. Numerical experiments (see Section 3.4.3.2) suggest that also for the double preconditioning method the order is not of relevant importance. In this chapter, unless specified otherwise, the order $\mathcal{C}(S)^{-1}D(S)^{-1}S$ will always be used.

3.4.2.3 Algorithm for an approximate dual

The double preconditioning method has two applications:

1. It can be used to speed up the convergence of an iterative scheme, here the Neumann algorithm, using S_2 in Figure 3.5 to get the canonical dual (up to a certain, predetermined error).

2. In order to get a real fast algorithm for the calculation of an approximate dual we propose the following method: The double preconditioning matrix itself, P_2^{-1} in Figure 3.5, is used as an approximation of the inverse Gabor frame operator. Then the approximated dual can be calculated as
$$\widetilde{g}^{(ap)} = P_2^{-1} g = \left(\mathcal{C}(S)^{-1} D(S)^{-1}\right) g \ .$$
This can be used for example for adaptive Gabor frames in real time, where the computation of the canonical dual window needs to be done repeatedly.

3.4.3 Numerical Results

3.4.3.1 The shapes of the approximated duals

In this first, introductory example we will use the double preconditioning matrix to get an approximate dual (as mentioned above) to see

1. that the different single preconditioning steps can capture certain properties of the dual window but fail to do so for others

2. the double preconditioning leads to a good approximation of the dual.

This experiment was done with a Gaussian window with $n = 640$, $a = 20$ and $b = 20$. In this case it is interesting to see the difference between the diagonal and the circulant 'dual' windows. We will use the names *diagonal dual*, *circulant dual* and *double dual* for the window we get when we apply the preconditioning matrix to the original window. Of course this does not have to be a real dual. See Figure 3.6.

The first seems similar to the canonical dual 'away from the center' but not near the center, while the second window just has the opposite property. Opposed to these 'single duals' the 'double dual' seems to combine these properties to become very similar to the true dual everywhere.

3.4.3.2 Order

We can now try to investigate whether the order has any influence. In this case we use a Gaussian window, $n = 144$, $a = 6$ and $b = 9$ and we look at the norms of the difference to identity:

method \ norm	Operator	Walnut	Janssen	Frobenius
$D^{-1}S$	0.1226	0.1232	0.1234	1.0397
SD^{-1}	0.1226	0.1232	0.1234	1.0397
$D^{-\frac{1}{2}}SD^{-\frac{1}{2}}$	0.1226	0.1232	0.1233	1.0397
$C^{-1}S$	0.0038	0.0045	0.0046	0.0324
SC^{-1}	0.0038	0.0045	0.0046	0.0324
$\mathcal{C}^{-1}D^{-1}S$	0.0006	0.0007	0.0008	0.0048
$\mathcal{D}^{-1}C^{-1}S$	0.0006	0.0007	0.0009	0.0048
$\mathcal{C}^{-1}SD^{-1}$	0.0006	0.0007	0.0009	0.0048
$\mathcal{D}^{-1}SC^{-1}$	0.0006	0.0008	0.0009	0.0048
$SC^{-1}\mathcal{D}^{-1}$	0.0006	0.0007	0.0009	0.0048
$SD^{-1}\mathcal{C}^{-1}$	0.0006	0.0007	0.0008	0.0048

We see in this case that the order is irrelevant. Also other experiments lead the authors to believe, that the order is not relevant. This has to be investigated further.

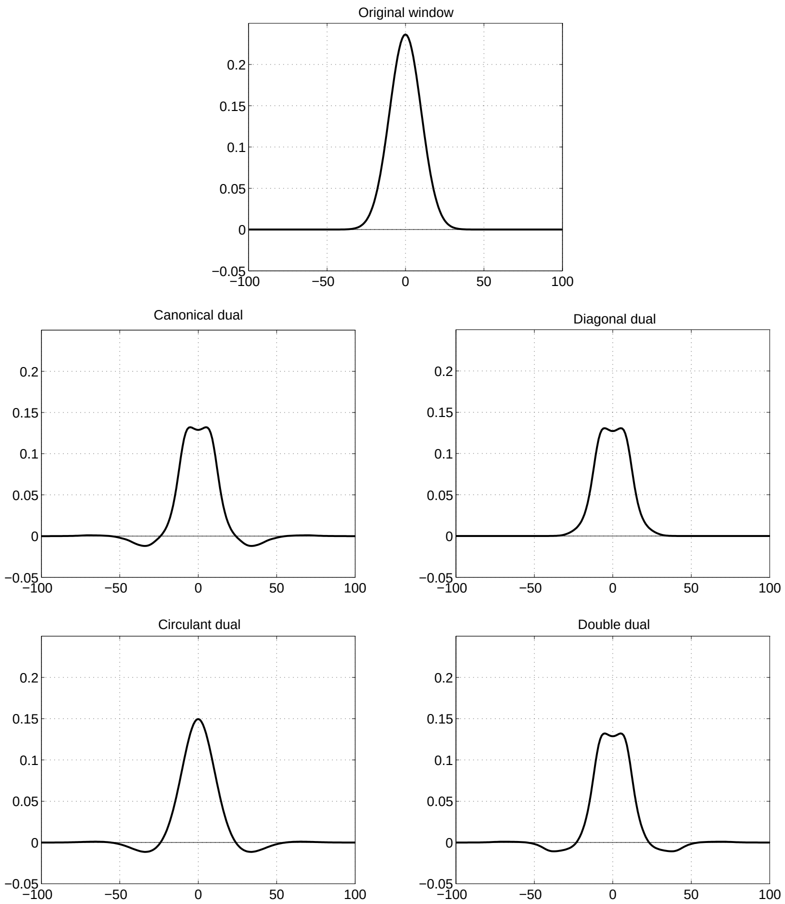

Figure 3.6: *Windows*: Top: the full original window; Mid left: the true canonical dual window, Mid right: 'diagonal dual', bottom left: 'circulant dual', bottom right: 'double dual'.

In this experiment we also see a good example of the norm inequality (3.10).

3.4.3.3 Iteration

Instead of using the preconditioning matrix as approximation of the inverse, we can iterate this scheme using the Neumann algorithm.

Let us look at an example with a Gaussian window, $n = 144$, $a = 12$ and $b = 9$. See Figure 3.7. We look at the preconditioning steps and the frame algorithm with optimal relaxation parameter. The calculation of the frame bound was done beforehand and so the results should be comparable to algorithm, which avoid this costly calculation.

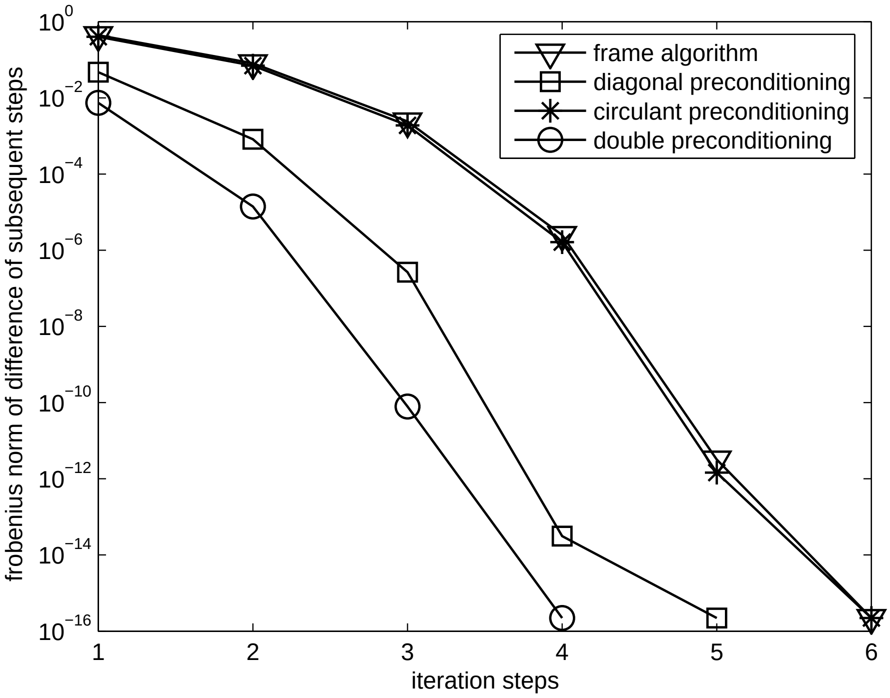

Figure 3.7: *Convergence with iteration*: Relative difference of iteration steps.

In this figure we see that the results of the circulant preconditioning step is nearly as bad as those of the frame algorithm. As the the sampling of the time axis is quite 'wide', it could be expected that circulant preconditioning is not very good. But still the double preconditioning brings an improvement compared to the single preconditioning with diagonal matrices.

3.4.3.4 The Janssen representation

To investigate the time-frequency plane let us look at the Janssen coefficients of the involved matrices. See Figure 3.8, where we have used a Gaussian window with $n = 144$, $a = 12$ and $b = 9$. Note that these are centered pictures. This means that the entry of the matrix $J_{1,1}$ is at the center of the picture as it corresponds to no time or frequency shift.

In the top left picture we see the time-frequency spread of the difference of identity and the original frame operator, $I - S$. It is clearly neither di-

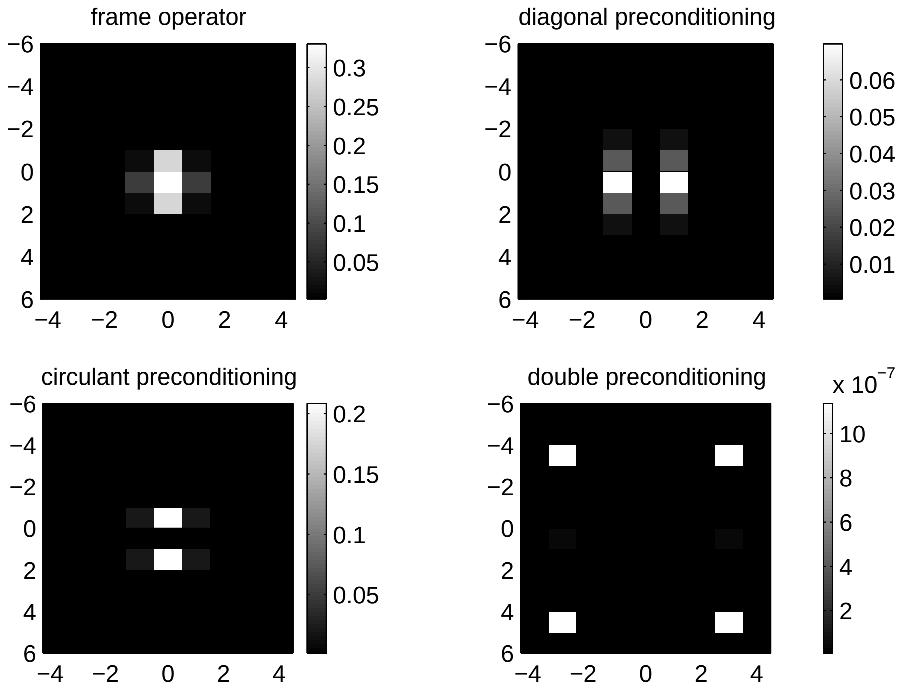

Figure 3.8: Time-frequency spread of differences to identity (Centered graphs)

agonal nor circulant, as diagonal matrices, which are linear combinations of modulations, would only have non-zero coefficients in the first row, whereas circulant matrices are non-zero only in the first column.

In the top right picture we see $I - D^{-1}S$ in the Janssen representation. The first column is zero, as the diagonal part was canceled out, but some other parts remain. An analogous property is valid for the circulant preconditioning.

For the double preconditioning method we see that in this case the Janssen norm would be very small. We further notice that the coefficients around the center, 'near the diagonal and circulant case' have been approximated well. The error occurs 'far from the center'.

So the Janssen representation gives us some insight on where in the time-frequency plane the coefficients of the difference to identity is high. As the Janssen norm just sums up the absolute value of these coefficients and is an upper bound for the operator norm, these gives us some insight on the error of the approximation in the time-frequency plane.

3.4.3.5 Higher Dimensional Double Preconditioning

For this 2D example, see Figure 3.9, we use a separable window, the tensor product $g \otimes g$. We use a Gaussian 1D window g with $n = 288, a = 12, b = 18$, so the redundancy is $red = 1.\overline{3}$. Here we do not get perfect reconstruction, but the reconstruction with the double dual is clearly much better than with the other two approximate duals. This can also be seen in the norm of the difference $\|Id - P^{-1} \cdot S\|_{Op} = 0.1796, 0.0914, 0.0300$ for the diagonal, circulant and double preconditioning case respectively.

For the calculation of the canonical dual with an iterative scheme $0.441s$ was needed, the 'double dual' needed only $0.060s$ on a MS Windows workstation with a Pentium III (937 MHz).

Figure 3.9: *2D Reconstruction*: Top left: the original image, top right: reconstruction with 'diagonal dual', bottom left: with 'circulant dual', bottom right: with 'double dual'.

3.4.3.6 Tests with Hanning window

For this experiment a zero-padded Hanning window was used as window.

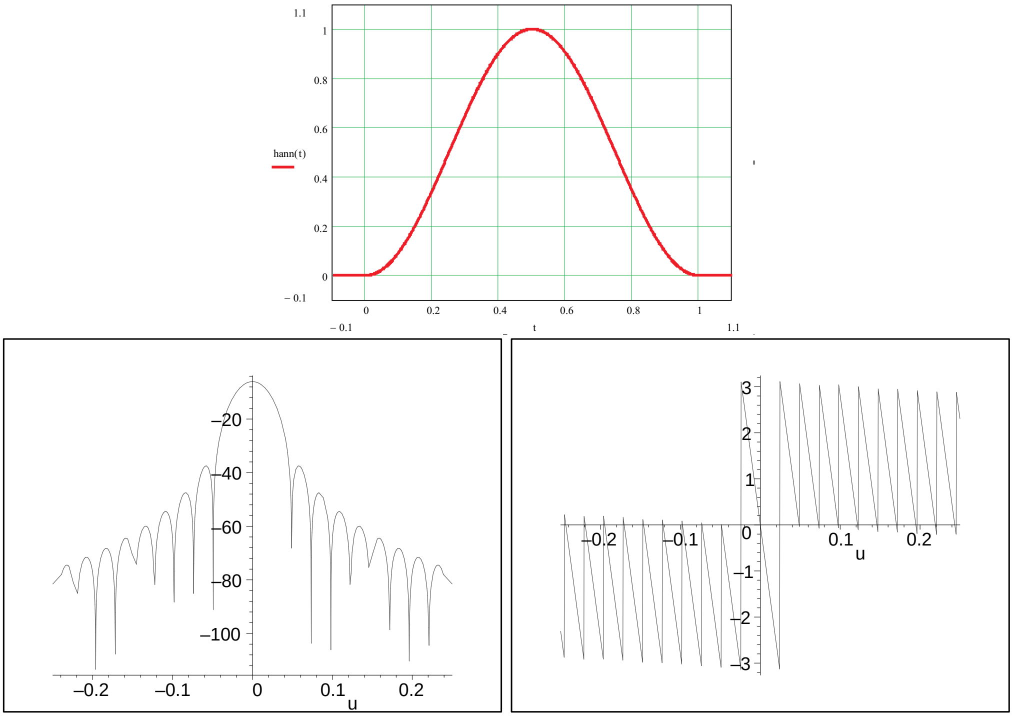

Figure 3.10: Top: The shape of the Hanning window. Bottom: Amplitude [dB] (left) and Phase (right) spectrum of a Hanning window

The length n of the signal space was randomly chosen between 1 and 300. Out of all divisors of n the length of the Hanning window w_{supp} was chosen, as well as a and b. Because we are interested in Gabor frames, we have restricted our parameters to $a \leq w_{supp}$ and $a \cdot b \leq n$. The parameters were randomized one thousand times. Some results can be found in Table 3.1. Note that this table is not ordered and sums up interesting results, not representing the statistics, so e.g. the cases are taken out, where we have no frame or the matrix clearly is diagonal due to the support criteria.

We use the terms *Diagonal Norm* for $\|D^{-1}S - Id\|$, *Circulant Norm* for $\|C^{-1}S - Id\|$, calling both of them *Single Norms*, and *Double Norm* for

$$\left\|D^{-1}C^{-1}S - Id\right\|.$$

The choice of the particular norm used depends on the context. Here in Table 3.1 the operator norm has been used, as this experiment was intended

n	redundancy	Diagonal Norm	Circulant Norm	Double Norm
688	8	13.375	2.92104e-008	2.67309e-008
891	3.66667	0.698704	0.00209154	0.00208434
704	1.375	1.67428	64.8568	114.485
868	1	0.998769	1.0134	0.99787
144	1	0.562595	0.669362	0.391396
418	1.15789	0.398039	0.449851	0.24604
300	1	0.999551	0.626595	0.571454

Table 3.1: Typical results for a Hanning window

to be short and introductory. Thus no attention has been given to numerical efficiency. The distribution of the cases can be found in Figure 3.11.

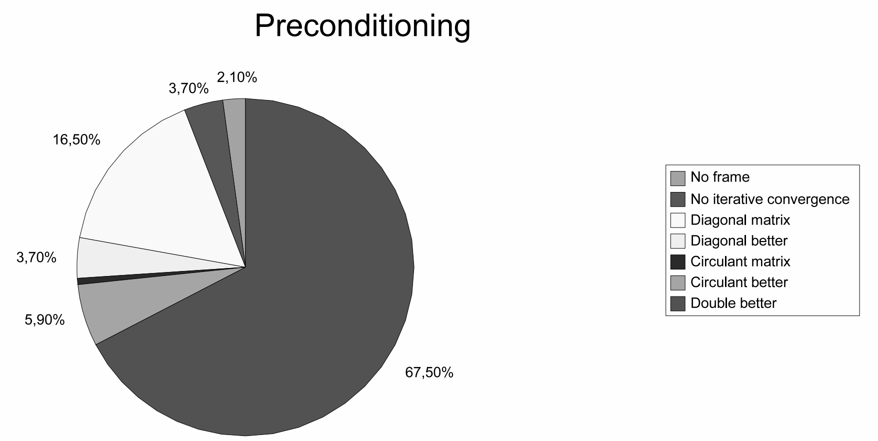

Figure 3.11: *Experiment with Hanning window*: Distribution of cases.

The double preconditioning method is in 67.50% of all cases convergent and preferable to the single preconditioning methods, measured by the norm of the distance to identity of the preconditioned matrix, $\|I - P^{-1}S\|$. Out of the remaining 32.50% the diagonal method is often preferable. As will be seen in 3.4.3.7 this property does not depend only on the shape of the window, but a lot on the lattice parameters, most notably how small b is, and the chosen settings of the experiment. When the above mentioned quality criterion is used to measure significant differences, in this experiment only in 0.1% of the cases one of the single preconditioning methods was 'essentially' preferable (more than 10% difference).

In these experiments we could also observe that if the norms of both the single preconditioning cases are around 1, the norm of the error of the double preconditioning method is also around 1.

3.4.3.7 Systematic experiments

In order to verify that the double preconditioning method is not only highly efficient for very special cases (such as the Gaussian), but for most windows typically used in Gabor analysis we have carried out systematic investigations. We have used several different windows (Gauss, Hanning, Hamming, Kaiser-Bessel, Blackman Harris, Rectangle and even noise), with various zero-padding factors, random signal lengths $n < 1000$, random lattice parameters a, b with $a \cdot b | n$ and random support of the window $w_{supp} \geq a$.

Here we have tried to minimize the cases, where the matrix is diagonal because of the lattice parameters (if $supp \leq n/b$). In this case it would still be possible to use the double preconditioning, we would only lose precision due to calculation and round-off errors, and the calculation is slower as the double preconditioning is more complex. Due to the conditions on the lattice parameters and the support of the window mean, we do get a certain bias into our statistical investigation. But this bias seems acceptable.

The complexity of the algorithm and these tests have been further decreased by staying completely at the block matrix level, by doing all calculation with the efficient block algorithms and by using the Walnut norm.

The results are summarized in Table 3.2. For each window the experiments have been repeated $20,000$ times, so overall in the following table $120,000$ random parameters have been used.

In the rows we see the following percentages

1. the Gabor system was no frame
2. none of the (above) iteration scheme would converge, i.e. none of the norms was smaller than 1
3. the diagonal norm was smaller than the double norm, where in
   - 3') the frame matrix was already diagonal (and so both methods were essentially equal).
4. the circulant norm was smaller than the double norm, where in
   - 4') the frame matrix was already circulant (and so both methods were essentially equal).

	Han	Ham	Bla	Kai	Gau	Noi
1)	0.00 %	0.00 %	0.39 %	0.00 %	3.66 %	0.00 %
2)	36.42 %	37.36 %	29.80 %	34.05 %	22.42 %	56.00 %
3)	28.53 %	28.53 %	29.80 %	28.62 %	30.17 %	27.87 %
3')	28.50 %	28.53%	29.39 %	28.46 %	30.17 %	27.80 %
4)	13.52 %	12.30%	9.58 %	16.46 %	2.41 %	1.73 %
4')	0.00 %	0.49%	9.19 %	0.74 %	0.11 %	0.00 %
5)	0.12 %	0.50%	1.02 %	0.00 %	0.02 %	00.12 %
6)	0.00 %	0.00 %	0.00 %	0.00 %	0.14 %	0.00 %
7)	0.00 %	0.11%	0.04 %	0.08 %	7.77 %	0.00 %
8)	49.83 %	50.83%	60.45 %	50.07 %	74.34 %	49.83 %
9)	78.37 %	81.15%	86.25 %	75.92 %	96.89 %	78.37 %

Table 3.2: Systematic Tests: (Han)ning, (Ham)ming, (Bla)ckman-Harris, (Kai)ser-Bessel ($\beta = 6$), (Gau)ss and (Noi)se

5. the double norm was bigger than 1, the best single norm was smaller than 0.9.
6. the double norm was essentially larger (by a factor 10) than the best of the single norms.
7. the double norm was essentially smaller (by a factor 10) than the best of the single norms.
8. The double preconditioning method is better or essentially equal if the system is a frame. We sum up the cases, where the double preconditioning norm is smaller and the matrices are already diagonal or circulant (because then the difference is only due to calculation errors).
9. The double preconditioning method is better or essentially equal if any of the iterative scheme works.

Nearly in all cases these windows form a frame. A prominent exception is the Gaussian window, which is due to the zero-padding. About the same percentage for all windows did not allow any of the preconditioning iterative algorithm to converge, exceptions being the Blackman-Harris with a somewhat low percentage, the Gaussian with a very low percentage and the noise window with a very high percentage. This leads us to the statement that the preconditioning algorithm works better for 'nice' windows.

For the windows tested it appears that the percentage of diagonal matrices is comparable, even in the case of a noise window. This is partly due to the particular properties of the chosen experiment. The percentages for circulant matrices respectively for convergence of the circulant preconditioning method seem to be quite different for different windows.

In very few cases, a single preconditioning algorithm would converge, but the double preconditioning would not. If investigated more closely it can be seen that this happens nearly always only in cases, when also the single preconditioning norms are high, near to one. For the cases when all relevant norms are smaller than one, we see that for the Gaussian window we have only a very small chance that the best single preconditioning method is essentially better than the double preconditioning method, but a rather high chance for the opposite. For all other windows the chance for an essential improvement using the double preconditioning method is not very big, but there is no chance for a deterioration. Note that here the double preconditioning method still keeps an advantage, since it can be more easily used as 'default' method than the single preconditioning methods as seen in 9) in Table 3.2.

Overall we see that with all windows the double preconditioning algorithm works in about half of the cases, if we have a frame. And it works in about 80 percent of the cases, when any of the preconditioning would work, with the notable exception of the Gaussian window, where it works nearly always. The Hanning and Hamming windows are quite similar to each other but contrary to common believe they are not very similar to the Gaussian window. We see that the behavior of the double and single preconditioning method significantly depends on the chosen window. Hence the connection of analytical properties of the windows with the efficiency of the preconditioning methods should be investigated. Results in this direction can be expected, e.g. due to the behavior of the Gaussian on one side and noise on the other side.

3.4.4 Perspectives

We believe that this algorithm can be very useful in situations, where the calculation of the inverse frame operator or dual window is very expensive or cannot be done at all. For example in the situation of *quilted Gabor frames* [34] or the *Time-Frequency Jigsaw Puzzle* [72], there exists a frame, which globally is not a Gabor frame. Hence the dual Gabor window cannot be found, but the dual frame can be approximated by the dual windows of the local Gabor frame in these cases. It might be preferable to use a good and fast approximation of the local Gabor dual windows to a precise calculation of the local canonical dual, as precision is lost at the approximation of the global dual frame anyway.

Some issues will have to be investigated further in the future. For example a more easily interpretable condition for the window, when the Jacobi algorithm is convergent, would be nice. Section 3.4.3.7 gives reasons to believe that an investigation of the analytic properties of a window and the connection to its 'preconditioning behavior' is fruitful. Furthermore the idea can be extended by using preconditioning matrices produced by projection using other commutative subgroups of the time-frequency plane, not only the translations and modulations.

Chapter 4

Application to Psychoacoustical Masking

We have already seen that to minimize computation time, memory requirements and bandwidth, the redundancy of a signal representation should be kept low. Many audio coders, for example, try to minimize the bit rate for audio signals. For audio signals, where the main interest lies in the human perception of sound, any part of the signal that cannot be heard is obviously redundant. This means the representation can be made more sparse, if it is restricted only to the psychoacoustical relevant parts, which is exactly what masking filters do.

Masking filter algorithms are for example used in the MP3 coding, refer for example to [79]. Their primary task is to filter signal components, which cannot be perceived by the human auditory system. This is certainly a non-trivial task. It strongly depends on the signal itself, and so it can be seen as an adaptive filtering, which is highly non-linear. But this filtering can be separated into two steps, first the calculation of the operator, which then is applied to the signal.

$$x \mapsto G(x)x$$

The second part is linear again. In the case of masking this means that first the mask for the time-frequency coefficients is calculated, which then is applied as an irregular Gabor multiplier.

This is a mathematical work, so only the basic ideas are formulated and the basic idea for an algorithm is presented. The author is in no way a fully-fledged psychacoustican, but in discussion with psychoacousticans and acousticans, most notably B. Laback, some ideas were formulated how to find a model and algorithm for time-frequency masking. These ideas certainly have to be validated by psychoacousticians. Psychoacoustical tests and experiments have to be performed. There are *no* concrete implementations in this chapter or in the appendix, as this has to be adapted to the system and the programming language used for these experiments.

4.1 Psychacoustical Basics

For details on aural perception and on psychoacoustics see [67] or [92].

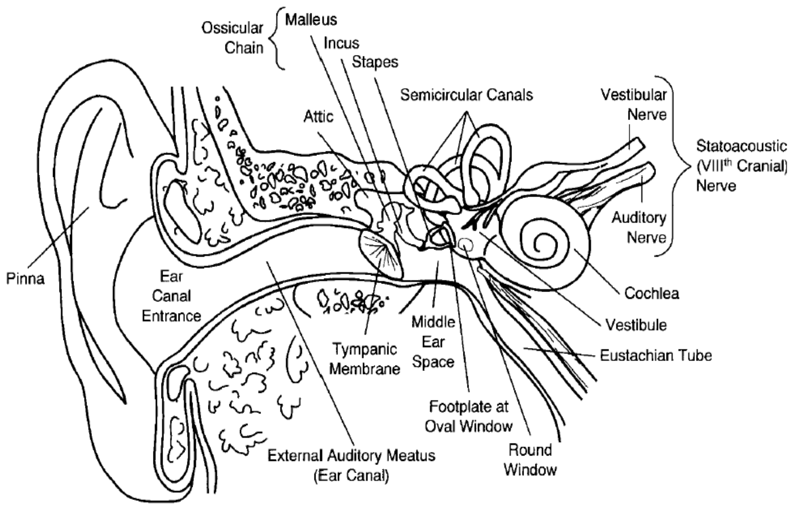

Figure 4.1: The human ear, from [59].

Sound waves travel through the air (or any other media) to reach the *pinna*, the outer ear, cf. Figure 4.1. The pressure wave spreads through the ear canal, which has a certain frequency response, to reach the *tympanic membrane*, where the *ossicular chain* transfers the big, but small-force vibrations to strong, but small vibrations at the *oval window* in the *cochlea*. There the *basilar membrane* is excited, which stimulates the hair cells to send impulses through nerve cells.

It is clear, that humans cannot only perceive temporal or spectral features, but have the ability to perceive both, otherwise speech or music perception would not be possible. The human auditory system, therefore, does perform some time-frequency analysis. Since the beginning of the investigation of human audio perception, scientists have searched for a good model for this time frequency analysis.

4.1.1 Aural perception

The pure tone can serve as the reference for the psychological dimension of pitch, because the pitch is closely related (but not identical) to the physical quantity of frequency. It can be said that the human auditory system behaves much like a frequency analyzer. The basilar membrane in the cochlea vibrates. It is excited by the traveling sound wave, which moves from the oval window to the apex of the cochlea. The maximum vibration for high-frequency tones occurs near the oval window, for low-frequency near the apex. This correlation between frequency and spatial point on the membrane is called *tonotopy*, cf. Figure 4.2. The hair cells on the basilar membrane are stimulated and generate electrical pulses.

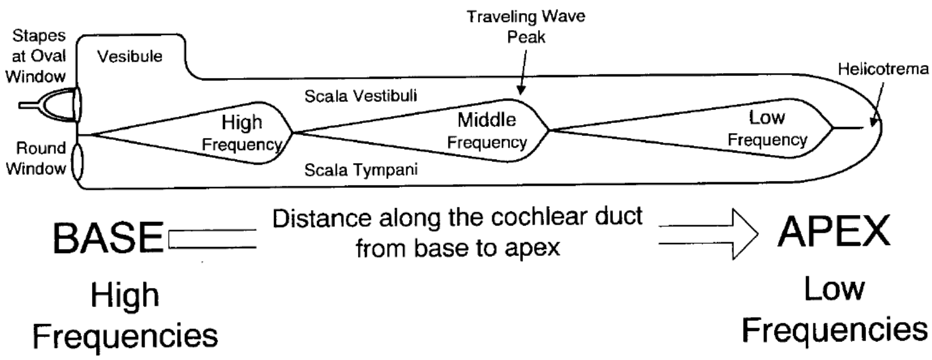

Figure 4.2: The tonotopy on the Basilar membrane, from [59].

A very basic model for the time-frequency analysis of the human ear is the STFT with window length of about 40 ms. For a basic model the phase information can be neglected.

4.1.1.1 The Basilar Membrane

In the human ear, frequencies are perceived with hair cells on the basilar membrane. The sound wave travels through the pinna, the eardrum, the ossicular chain to the oval window of the cochlea. It excites the basilar membrane and so activates the nerve cells. Closer to the window they react to high, farther away to low frequencies.

Physiological measurements on the basilar membrane and on the auditory nerve show that the frequency of a pure tone is encoded in the site of the maximum activation along the tonotopic organization. This analysis is an integral part of the initial transduction of a tone from mechanical vibrations to neural impulses. Therefore, different harmonics of a complex tone end up in different neural channels.

One single frequency is not only exciting a single cell, but it causes a certain excitation pattern on the basilar membrane, also called spreading function. This pattern depends on the frequency and amplitude, but a good approximation, see [92], is a band pass filter with a certain center frequency and certain slopes at the edges of the pass band. Linear slopes, in dB/Bark, are a good approximation. The Bark scale, see e.g. [117], is a frequency scale better fitted to human perception. This scale is empirically determined, but a formula, which describes it fairly well, is

$$f_{bark} = 13\ tan^{-1}(0.76 \cdot f_{Hz}) + 3.5\ tan^{-1}(\frac{f_{Hz}}{7.5}^2)$$

where f_{Hz} is the frequency in Hz. In first approximation, the Bark scale resembles the tonotopy. Refer to Figure 4.3.

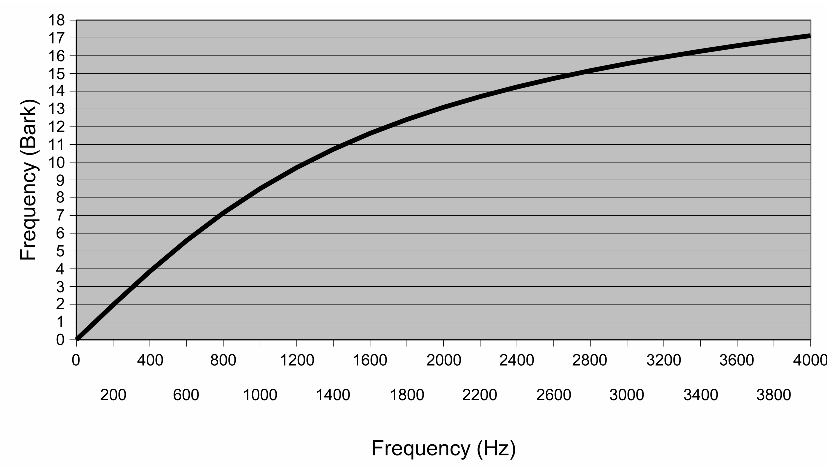

Figure 4.3: The Bark scale

For this scale the concept of the *critical band* is important, as a critical band corresponds to 1 bark. A basic definition of critical band is the bandwidth within which signal components interact fundamentally differently than for larger bandwidths. For example for two signal components within a critical band their power is additive, for components separated by more their loudness is, cf. [67]. Again in first approximation the critical bandwidth can be seen as the basis for an auditory filter.

As the central system interacts with the hair cells, such an interaction is called *efference*, excitation on the basilar membrane is highly amplified through this process, resulting in a highly non-linear process. A linear model can only be an approximation, and so the critical bandwidth depends on the signal class, the amplitude and the phase information.

4.1.2 Masking

Masking can be defined generally as the situation, where the presence of one stimulus, the masker, decreases the response to another stimulus, the target.

There are, of course, several configurations where this effect occurs in audio perception. We start with a recap of the basic ideas of the so-called simultaneous or frequency masking.

For more details see [137] or [39].

4.1.2.1 Simultaneous Masking

A basic model for the frequency masking effect can be found in the following way. Suppose we have one signal component, the *masker*. The auditory system can only detect a second, simultaneously presented, signal component, the *target*, if the excitation pattern of the resulting signal is significantly different from the one evoked by the masker. If it is not, the second sound cannot be perceived, it is masked. This kind of masking is called *simultaneous masking* as the two components are presented at the same time.

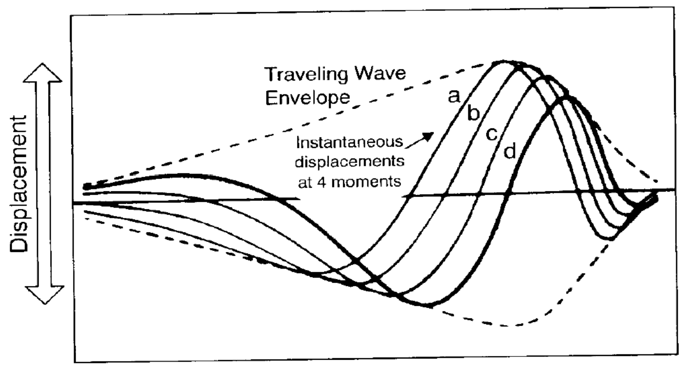

Figure 4.4: Traveling wave on the Basilar membrane, from [59].

A single sinusoidal component does not excite the basilar membrane only at a single point, but evokes a certain *excitation pattern*. See Figure 4.4.

For perceptual issues a logarithmic scale of amplitudes is best fitted, see [67]. This has, of course, a big influence on the additivity of two excitation patterns of two sinusoidal components, see Figure 4.5.

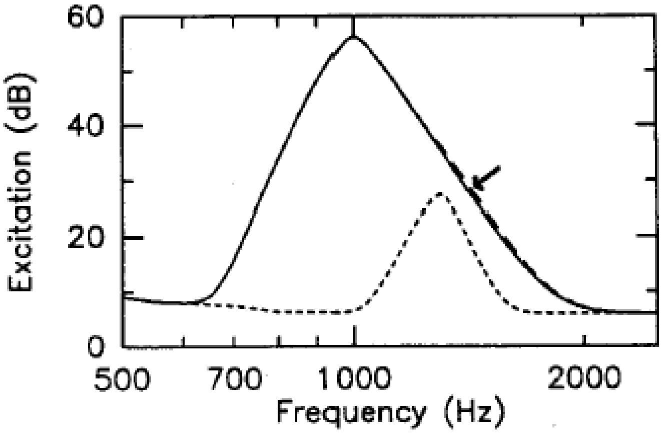

Figure 4.5: The addition of excitation patterns plotted on logarithmic frequency and amplitude scales [68]. The masker is shown with a solid, the target with a dotted and the sum with a dashed line.

The effects can be seen for example in a masking experiment using a pure tone probe and a complex tone masker, refer e.g. to [67]. The idea of this experiment is that the pure tone cannot be heard if the excitation pattern of the pure tone plus masker is not distinguishable from the one by the masker only. The amount of masking, measured in dB, is defined as the threshold level of the sine, i.e. where the sine tone is barely audible, in presence of the masker, minus the threshold level when the sine tone is presented alone. By measuring the amount of masking as a function of the probe frequency, the experiment traces out the excitation pattern of the masker.

4.2 The Masking Algorithm By Eckel

A good approximation of the spreading function for a sinusoid signal is a triangle function (in the Bark scale). This was used in [37] to formulate a simple masking model, from which an algorithm for simultaneous masking was implemented. The simultaneous masking algorithm is implemented as an adaptive filter, using a phase vocoder as analysis/synthesis system. The masking threshold is calculated for each spectrum. First the spectrum is converted to bark scale and the amplitude is converted to a logarithmic scale. The amplitude spectrum is convolved with a nearly triangular function. It is set to amplitude zero at frequency zero (Bark) as no component can mask itself. Convolution is used for the addition of the masking effect, because it is known that even signal components below the threshold have an influence. This is called the masking or *relevance* threshold. Only the components exceeding it, are used for further re-synthesis. See figure 4.6 for the implementation in ST^X.

4.2.1 Simultaneous Masking For Complex Signals

This algorithm was tested in [37] only for certain parameters, a sampling frequency of 16 kHz, a window length of 256 samples and a hop size of 32 samples. Based on data from the literature the lower slope was set to 27 dB / bark and the upper slope to -24 dB / bark. Also a so called *damping factor* is used, which describes the "sharpness" of the edge of the triangular function.

In the evaluation experiment in [37] 312 musical signals were presented, which were chosen to represent a variety of different musical styles and instruments. The test with these complex signals was chosen to obtain significant results representing real-life situations. The selected stimuli had lengths of 300 ms, 600 ms and 1200 ms.

These signals were presented to 43 persons. It was tested for which level of 'offset' the irrelevance-filtered signals can not be distinguished from the original signal. The 'offset' is a value in dB by which the threshold level is increased or decreased. These values were chosen such that there was no statistical difference in the answers of the subjects, whether two signals were different, using either twice the original signal or the original signal and the masked signal. The hypothesis was tested with a student-t-test.

4.2.2 The Algorithm In ST^X

This was implemented in ST^X[96] the signal processing software system of the Acoustics Research Institute of the Austrian Academy of Sciences. In Figure 4.6 we see the basic routine. The parameters of the masking filter can be changed freely, e.g. all the FFT parameters or the form of the spreading function used, like the slopes of the linear parts the sharpness of the peak and the height of the peak plus the offset of the function. Two special options have been implemented, where the parameters can be chosen such that a certain part of the energy or spectral bins is masked. To calculate the spreading function from the given parameters, a faster, analytic method is implemented in ST^X.

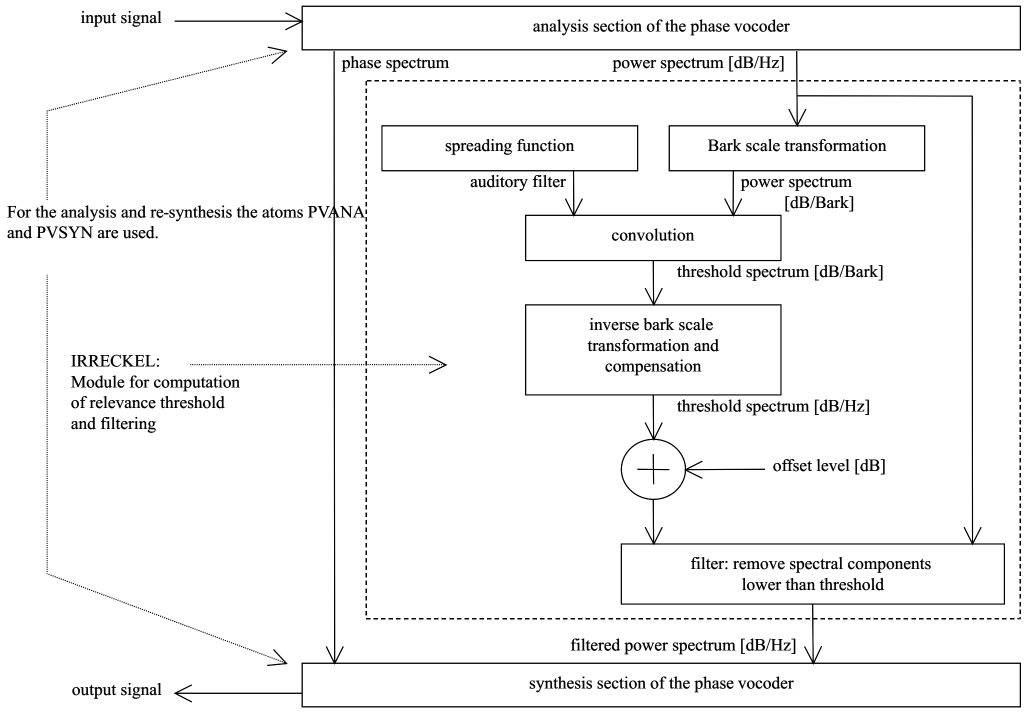

Figure 4.6: Masking filter in ST^X[96]

At the Acoustics Research Institute this filter in ST^X was used for various applications as stated in Section 4.2.3. Experience tells that the parameters of [37] can be generalized to other settings. The model used for this algorithm is based on the Fourier view, so every single spectrum is used and it is assumed that the signal is "quasi-stationary", see Figure 4.7. If the window size is getting larger and larger and so more and more time information is encoded in the spectrum, and it becomes evident that the algorithm does not give a proper resemblance of the actual situation.

For an example see Figure 4.8, where not the irrelevance threshold but overmasking is used. *Overmasking* was investigated in [30]. In that paper the parameters were chosen to filter out more signal components than with a relevance threshold. In Figure 4.8 the parameters for "musical irrelevance",

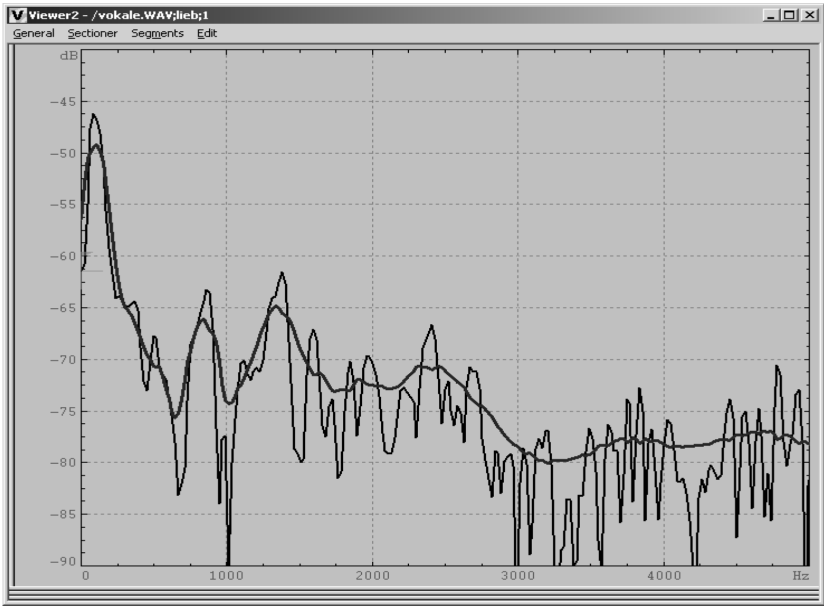

Figure 4.7: From ST^X: The amplitude spectrum (dB) and the relevance threshold.

cf. [30], were chosen, which means that a difference to the original signal can be heard, but the relevant musical information is still preserved.

4.2.3 Typical Application

Typical application of masking filters include

1. *Sound / Data Compression* : For applications where perception is relevant, there is no need to encode perceptually irrelevant information. Data which can not be heard should be simply omitted. A similar algorithm is for example used in the MP3 encoding. If over-masking is used, this means that the threshold is moved beyond the level of relevance, and so a higher compression rate can be reached.

2. *Sound Design* : For the visualization of sound the perceptually irrelevant part can be disregarded. This is for example used for car sound design, see e.g. [94].

3. *Background - Foreground Separation* : With over-masking it is possible to separate the leading instrument in a piece of music, refer to [30].

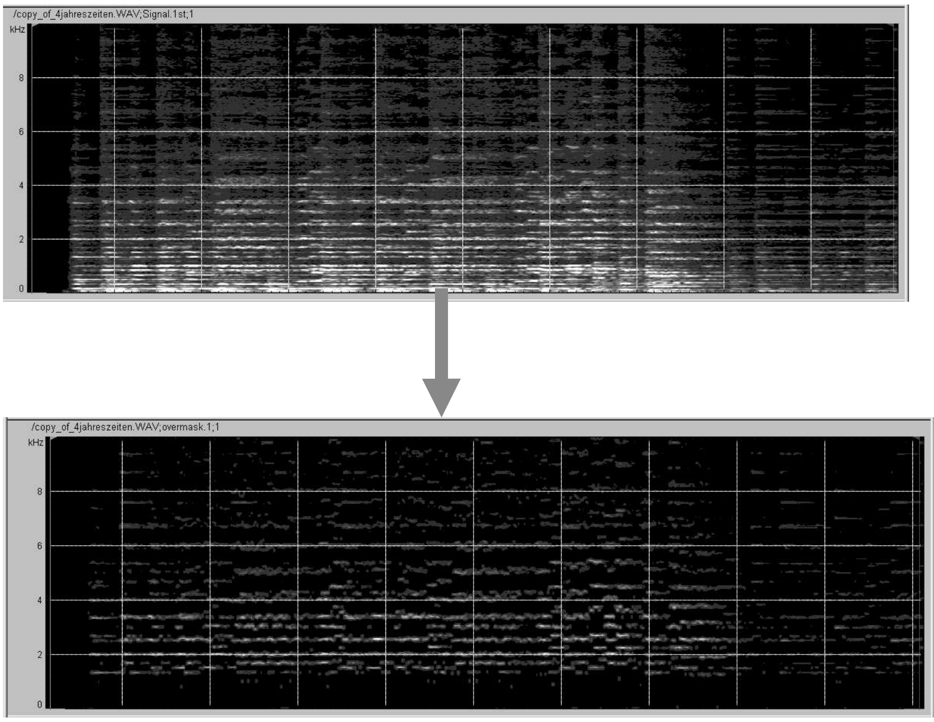

Figure 4.8: Simultaneous masking in ST^X: Top: original sound (music), Bottom: perceptual irrelevant signal parts removed

4. Amplification of masked components: Under certain conditions, the enhancement of weak components, which fall below the relevance threshold, can improve speech recognition in noise, see [78], or music perception [77].

5. *Contrast Increase* : If more spectral parts are deleted, the hearing comfort of hearing impaired people often increases. [77]

4.3 Time Frequency Masking

In the Eckel model one important restriction is to single spectra modeling only the simultaneous frequency masking. Here we propose a (very) simple algorithm to include temporal masking effects as a natural extension of the Eckel algorithm. It has to be investigated in psychoacoustical experiments. This masking model was found by trying to find a simple time frequency algorithm extending the Eckel algorithm and not by trying to incorporate the plentiful experiments done on time or frequency masking or existing models for time frequency masking, for example found in [116] or [125]. Please keep in mind that psychoacoustics was not the main focus of this work.

4.3.1 Temporal Masking

Beside the frequency masking effect, temporal masking effects are well-documented. Interestingly it is possible to mask both in forward and backward direction. If a masker is presented before or after the target, its perception is influenced. The masking level for temporal masking can be found in Figure 4.9. This was for example studied in [39]. One prominent example for this effect is, that pauses in music, compared to tones, are always perceived shorter than they physically are, cf. also [39].

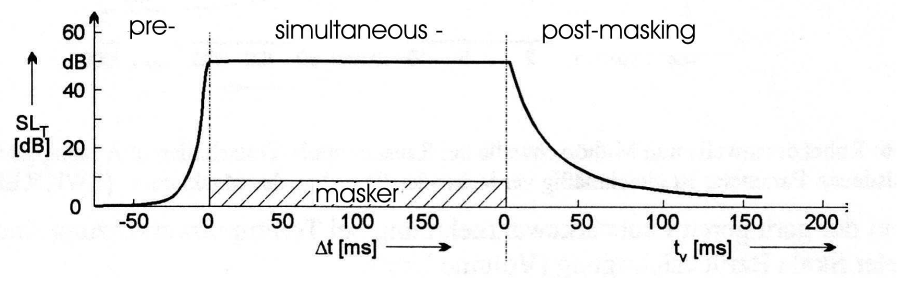

Figure 4.9: Forward and backward masking, from [137]

Possible explanations for the temporal masking effects are, according to [91]

1. a decay of the response of the basilar membrane to the masker,
2. reduction of sensitivity of recently stimulated neurons (adaptation)
3. persistence in pattern of neural activity evoked by the masker (more central)

4.3.2 Heuristic Tests

The first idea of testing, how masking in the time-frequency plane would be to compare two chirps, sinusoidal tones with rising (or falling) frequency respectively a chirp and a stationary tone, as depicted as spectrograms in Figure 4.10.

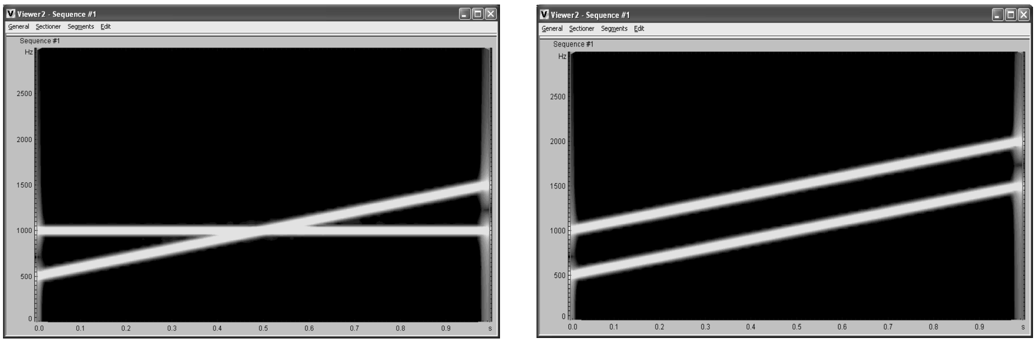

Figure 4.10: Simple (but naive) experiment for two chirps

But in (heuristic) tests it could be seen that these two chirps interact in a special way, comparable to two sinusoidal components. Due to the non-linearity of the auditory system, two similar sinusoidal components produce a difference tone, which is very easily detectable, refer to e.g. [67]. It is well known, that because of this effect, it is impossible to directly measure the simultaneous masking effects of two sinusoidal components, see [137]. A similar problem seems to be in effect, when chirps are used.

For a possible experiment to investigate time-frequency masking and how to avoid this problem refer to Section 4.3.4.3.

4.3.3 The Masking Gabor Multiplier

The goal is to get a time-frequency masking model, as presented in Figure 4.11.

A simple explanation for simultaneous masking is the excitation pattern on the basilar membrane, for temporal masking a explanation can be found, which is more central in the human auditory perception. So a simple model for time-frequency masking is first to calculate the simultaneous masking effect, simulating the excitation of the hair cells with a convolution of the short-time spectrum by the simple triangular function introduced in [37]. The temporal masking effect can be seen to be more central and so 'later' in the perception process, this data now is convolved with another triangular function to simulate the temporal effect. This is done because it can be assumed that a multitude of nerve cells are stimulated. This has to be considered.

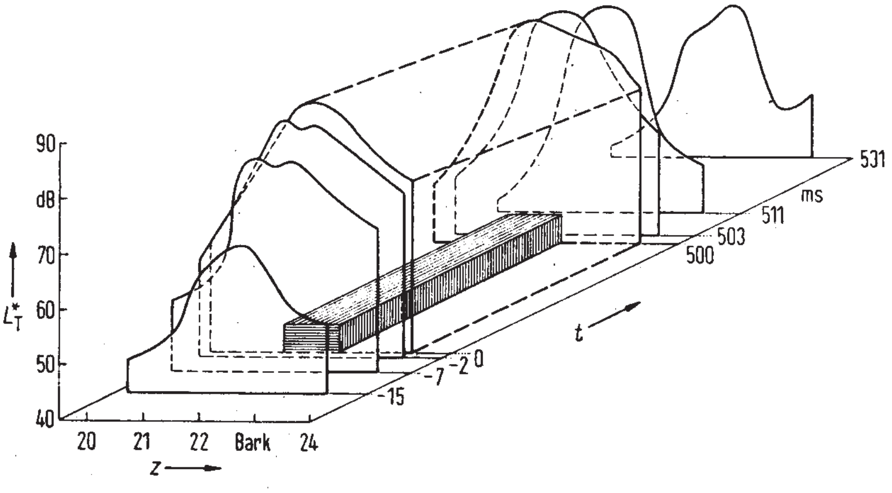

Figure 4.11: Time frequency masking pattern, from [39]. Transient masking pattern of a single critical band noise masker impulse. The hatched bar is the spectral and temporal extent of the masker impulse.

To make the main the main concept clearer let us repeat: The basic idea is to use single time-frequency points, calculate the spreading function with a simple triangular model, to simulate the excitation of the basilar membrane. This excitation is processed parallel by the nervous system. Therefore in this model the temporal masking effect is applied at every single frequency point. Again a simple triangular function is chosen. In combination an masking effect between time-frequency points is modeled, which does not occur in temporal or spectral direction only. This is done for all time-frequency points and the threshold levels are summed up. See Figure 4.12.

Like in [37] an offset parameter is used, which shifts the time-frequency pattern up or down. This parameter can be seen as the value corresponding to how two different excitation patterns have to be to be distinguishable. To add more flexibility two different offset values can be chosen.

4.3.3.1 Parameters:

A Gaussian window is chosen for the STFT as default, as psychoacoustical tests indicate that the essential support in the time-frequency plane of the window underlying the human auditory perception is close to the minimum, see [109], and so is near to a Gaussian window.

As default for many parameters the values from [37] are used, for example

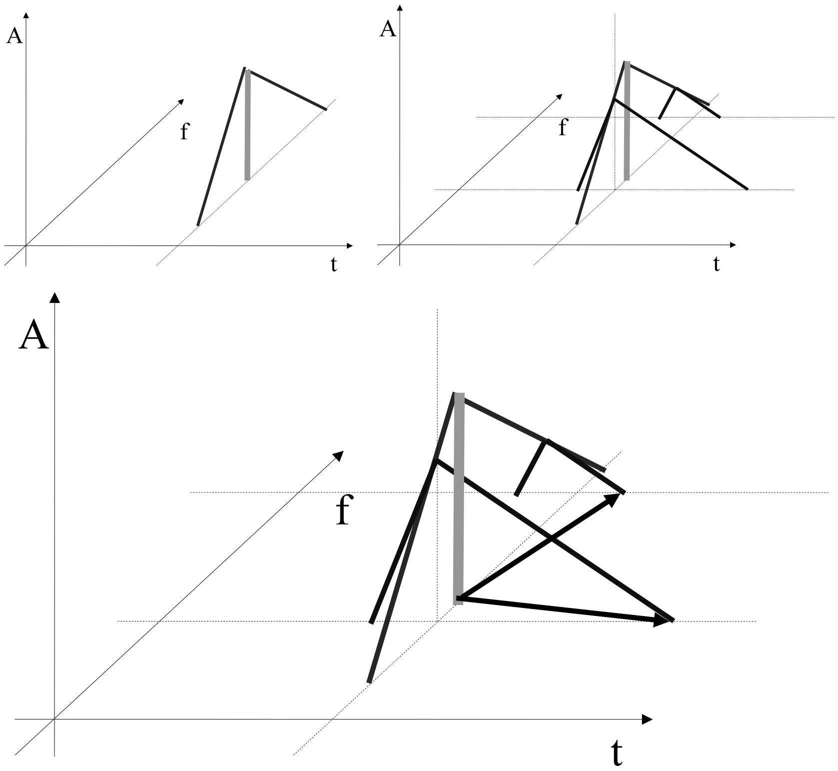

Figure 4.12: Time frequency masking idea. (Top left:) Spreading function (Top right:) Temporal masking pattern (Bottom:) Combination effect in the time-frequency plane.

the slopes of the triangular functions are set to 24 dB and -27 db for the frequency part. For the temporal masking function the slopes suggested in [137] and seen in Figure 4.9 are used , 2 dB/ms respectively -0.5 dB/ms, are used as default.

Overall, we have the following parameters, which should be adapted to the properties of the auditory system by thorough psychoacoustical experiments:

1. the sampling rate SR of the audio file. (Default: 16 kHz).

2. the window g, especially the length win of the window. (Default: Gaussian window with 256 samples).

3. the temporal shift b (the hop size). (Default: 32 samples).

4. the number of frequency sample n_b (bark points). This should be chosen rather high, to have a high redundancy and therefore to increase the

chance of having an irregular Gabor frame. (Default: 128, redundancy is 4).

5. the offset value o_f for the frequency masking function. (Default: 1 dB).

6. the offset value o_t for the temporal masking function. (Default: 1 dB).

7. the 'excitation pattern function' $ep(\omega)$:

   (a) the lower slope: k_d (Default: 27 dB)

   (b) the upper slope: k_u (Default: -24 dB)

   such that

$$ep(\omega) = \begin{cases} k_d \cdot \omega + o_f & \omega < 0 \\ 0 & \omega = 0 \\ k_u \cdot \omega + o_f & \omega > 0 \end{cases}$$

   The value at zero is set to zero, because no signal component can, per definition, mask itself.

8. the 'temporal masking function' $tm(t)$:

   (a) the backward slope: τ_b (Default: 2 dB / ms)

   (b) the forward slope: τ_f (Default: -0.5 dB / ms)

   such that

$$tm(t) = \begin{cases} \tau_b \cdot t + o_t & \omega < 0 \\ 0 & \omega = 0 \\ \tau_f \cdot t + o_t & \omega > 0 \end{cases}$$

   This function has to be converted to the time-frequency sampling, where $\Delta t = \frac{b}{SR}$. So for the default values of b and SR we get $\tau_b = 4$ db/sample and $\tau_f = -1$ db/sample.

4.3.3.2 Algorithm:

Let the signal be in $\mathbb{C}^L$. Let $\tilde{b} = L/b$.

1. Calculate the STFT of the signal with the window g. Use the painless non-orthogonal expansion method [63], which is equivalent to the algorithm used in engineering see e.g.[3], to speed up this calculation.

2. Do the semi-irregular sampling of the STFT, using frequency sampling points corresponding to n_b equally spaced bark points.

3. Check if the frame operator for this irregular Gabor system is invertible.

- If not, increase the number of sampling points in the frequency domain n_b.
- If yes, calculate the canonical dual window $\tilde{g}$.

This has to be done only once for each set of parameters, independent of the data.

4. Calculate the logarithmic amplitude information $A(t, \omega)$. Keep the phase information.

5. Calculate $ep \otimes tm$ and do a $2D$ convolution with $A(t, \omega)$ to get the 'masking pattern' $MP = A * (ep \otimes tm)$.

   Convolving $A(t, \omega)$ first with $ep(\omega)$ and then with $tm(t)$ is the same as convolving in both dimensions with $ep \otimes tm$, as

$$((A * ep) * tm)(t, \omega) = \sum_{\tau=0}^{\tilde{b}-1} (A * ep)(\tau, \omega) \cdot tm(t - \tau) =$$

$$= \sum_{\tau=0}^{\tilde{b}-1} \left(\sum_{\nu=0}^{n_b-1} A(\tau, \nu) \cdot ep(\omega - \tau) \right) \cdot tm(t - \tau) =$$

$$= \sum_{\tau=0}^{\tilde{b}-1} \sum_{\nu=0}^{n_b-1} A(\tau, \nu) \cdot (tm(t - \tau) \cdot ep(\omega - \tau)) =$$

$$= \sum_{\tau=0}^{\tilde{b}-1} \sum_{\nu=0}^{n_b-1} A(\tau, \nu) \cdot (tm \otimes ep)(\omega - \tau, t - \tau) =$$

$$(A * (ep \otimes tm))(t, \omega)$$

   This $2D$ convolution can be speeded up, by using property (4) from Lemma 3.2.9 for the Fourier matrix transformation and using a FFT algorithm.

6. Use the result as an 1/0 mask for an irregular Gabor multiplier, by setting

$$A^1(t, \omega) = \begin{cases} 0 & A(t, \omega) < MP(t, \omega) \\ A(t, \omega) & otherwise \end{cases}$$

7. Use A^1 as amplitude information and the original phase information for a Gabor synthesis with $\tilde{g}$.

4.3.3.3 Advantages

The advantages of this algorithm to the one in ST^X are

1. the incorporation of temporal forward and backward masking
2. the incorporation of Gabor theory results, especially that
   (a) it can be checked, if the chosen irregular sampling can lead to a Gabor frame, needed for a chance for perfect reconstruction (if no modification would be done).
   (b) If this is possible the synthesis window can be calculated for which perfect reconstruction is guaranteed.

4.3.4 Perspectives:

First and foremost the above model has to be implemented and tested in experiments, validated and adapted by psychoacousticans.

Further investigation of this topic can include the following ideas:

4.3.4.1 Newer Psychoacoustical Knowledge

At the Acoustics Research Institute some further extensive psychoacoustic tests of masking have been performed. These tests can result in an improvement of the masking algorithms, including using nonlinear auditory filters for the level dependence, outer/middle ear filtering (ISO-phones), taking into account the absolute threshold associated with internal noise and the dependency of masking on tonality of masker components for non-linear additivity of masking for non-tonal components.

4.3.4.2 Using Parts Of The Signals

It is well known, that different type of signals result in different masking patterns, see e.g. [137], especially tonal, transient and stochastical parts have different properties both as maskers and targets.

One way to improve the model would be to use algorithms to find tonal, transient and statistical components in masker and target and the signal and then use different masking patterns for each combination. For the separation the algorithm found e.g. in [72] or [89] can be used.

4.3.4.3 Time-Frequency Masking Experiment

An especially interesting questions is, whether the time-frequency masking effect is only a superposition of frequency and temporal masking or whether there is some other, more complex interaction. To obtain more knowledge about the properties of masking in the time-frequency domain, the basics for an experiment have been developed.

The basic idea is to use a broad band, uniformly masking noise (D) as masker. As targets one chirp (C) and sinusoidal signals (A,B) are compared to each other, cf. Figure 4.13

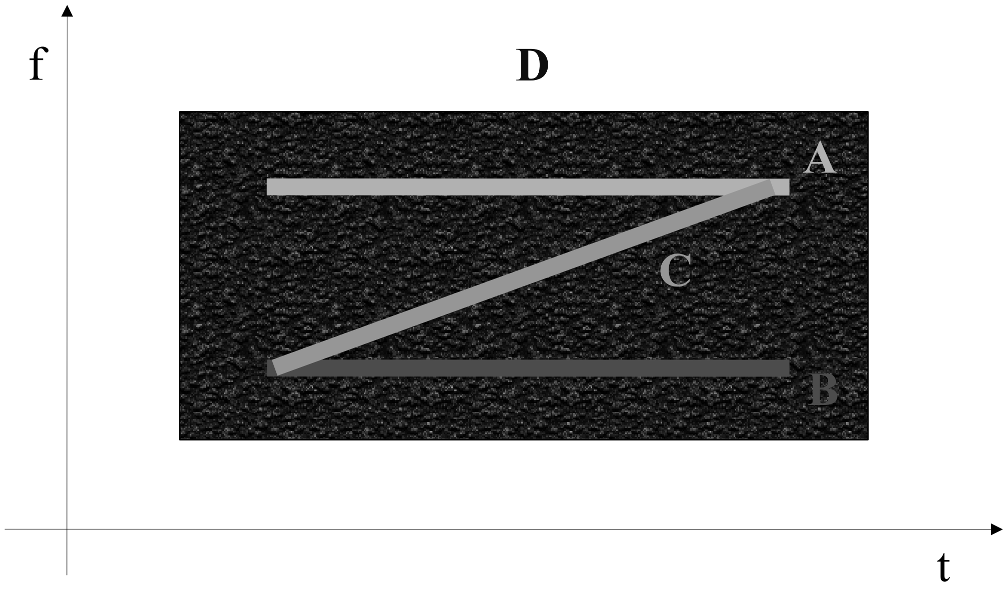

Figure 4.13: Experiment for time-frequency effect of masking.

In the classical model, if the signals are chosen with equal energy in the critical bands, there should be no difference in masking effect, but recent studies have shown that there are differences, refer to [15]. Similar experiments have already been done, for example in [95], but there are still a lot of open questions, for example a systematic investigation of the influence of the bandwidth and slope of the sweep.

Chapter 5

Conclusion

5.1 Summary

In this work we have spanned the whole range from mathematical theory to application. We have started with the theory of frame multipliers, proceeded with Gabor multipliers (regular and irregular), investigated the numerics of the discrete Gabor analysis and applied the theory to give an idea for a time-frequency masking algorithm.

In the first chapter we have investigated the new concept of frame multipliers, which generalizes the idea of Gabor multipliers.

We have started with an extensive overview of frame theory to help the reader familiarizing with this theory and getting a basic impression of frames. In this introductory part we have shown new results, mostly by extending known results to Bessel sequences, frame sequences or Riesz bases. We have also introduced new compilations and reformulations of known result, with the intention to show them from a different viewpoint. As an example of new results, we mention the connection of operators and frames in Section 1.1.7.

As this work aims at application, we were interested in the investigation of frames in finite dimensional spaces. To this end, we devoted Section 1.2 to deal with this issue. More precisely, we have shown that finite-dimensional spaces can be classified by frames, we provided new results on frames for the Hilbert-Schmidt class of operators and we compared the numerical complexity of the $\mathcal{HS}$ inner product of an operator and the rank one operator $g_k \otimes \overline{f}_l$.

Since the concept of frame multipliers has not been investigated before, we have studied their basic properties for the first time. The 'main theorem' in this context is the following: When the symbols are in a certain sequence space (like $l^\infty, l^2, l^1, ...$), then the multipliers are in certain operator spaces (bounded, $\mathcal{HS}$, trace class). Another important result states that the frame multiplier depends continuously on the symbol and on the involved frames (in a special sense). We have also proved that for Riesz bases the frame multipliers behave 'nicely', i.e. mostly importantly that the connection symbol to operator is an injective one. Finally, we have introduced an algorithm for the approximation of arbitrary matrices by frame multipliers, for a given frame. This algorithm has been programmed in MATLAB and can be found in the appendix.

In the beginning of the second chapter we have first given a short introduction to Gabor systems and have taken a close look at irregular systems. We have, for example, directly shown that for relatively separated irregular lattices, the Gabor system with a window in S_0 (i.e., the Feichtinger's Algebra) forms a Bessel sequence. In relation with Gabor multipliers, we have extended known results on the regular case to the irregular case. For example we have shown that under certain conditions these operators depend continuously on the symbol, atoms and lattices, where the similarity of lattices are measured by a 'Jitter-like' norm.

Several MATLAB algorithms have been implemented, e.g. the calculation of an irregular Gabor family for a given atom and a given set of time-frequency points. In a way parallel to what we had done in the first chapter, we have given an algorithm for the approximation of a matrix by irregular Gabor multipliers. To this end, we have used an algorithm to calculate the Gram matrix of an irregular Gabor frame, which makes the approximation algorithm numerically more efficient than that in Chapter 1. Experiments have shown that, in the case of one window and a regular lattice, this algorithm yields the same result as the algorithm in [50].

In the third chapter we have considered Gabor analysis in the finite-dimensional discrete case. After a short introduction to the special properties of Gabor analysis on $\mathbb{C}^n$ we have investigated classes of block-matrices which play an important role in this context. We have pointed out a close connection between the non-zero block matrix and the Janssen matrix, and have introduced corresponding norms. We have also pointed out the connection between these norms and why they can be useful in different situations.

We dedicated Section 3.3 to an article by Thomas Strohmer [122], in which an algorithm for inverting the Gabor frame operator is introduced. This algorithm is numerically very efficient in the case of integer redundancy. Many algorithms from this article are used, for example, in the MATLAB toolbox by P. Soendergard [118]. We have pointed out and corrected some small errors in the original article.

We have also introduced an iterative method for finding the inverse of a Gabor frame operator, which can also be used to compute very good approximate dual windows, at very low computational costs, if the window and the lattice fulfill certain properties. We have introduced a fast algorithm using existing block matrix methods. The method has been constructed so that diagonal and circulant matrices are perfectly approximated (up to precision). We have shown that this method is very often preferable to other iterative schemes. For 'nice' windows and lattice parameters, it has been made evident that the first approximation, i.e., the preconditioning matrix, is already a good approximation of the inverse frame matrix.

For the single preconditioning case we have specified sufficient conditions on the window which guarantees that the algorithm converges, and therefore the Gabor system forms a frame. We have also provided conditions on the non-zero block matrix for the convergence of the Jacobi algorithm. The condition on the window is not very intuitive, but as the block matrix can be established quickly, this check can be done in a convenient way.

In the fourth chapter we have introduced the basic idea for a time-frequency masking algorithm. To this end, we have given a short introduction of the basic ideas of psychoacoustical masking and the algorithm implemented in ST^X. We also have presented an idea, which was developed with the help of psychoacousticans, on how this algorithm can be extended to incorporate temporal forward and backward masking as well as results from Gabor theory.

5.1.1 Perspectives And Future Work:

No scientific work can claim that all connected questions are answered. In this last section we will state a few of the open problems and possible future investigations connected to this work:

Many question in connection with the new concept of *frame multipliers* should be investigated. For example we firmly believe that symbols in l^p lead to multipliers in the Schatten class [134]. It would be interesting to use the concept of the localization of frames [58] for multipliers ad it could be seen that the Gram matrix plays an essential role here. The concept of weighted frame has been introduced in [13]. The frame operator for such frames are just frame multipliers, the connection of these two notions should be delved into. Also the investigation of how to make a frame "tighter" by weights currently gets some attention [101]. This is closely related to the investigation of how well the identity can be approximated by a frame multiplier and this connection should be investigated further.

The irregular Gabor theory still has to receive a lot of attention. For example an important question is how the dual frame for such a family can be described. Frames of irregular translates are a current active topic of research [23]. The connection to Gabor Multipliers by the Kohn-Nirenberg symbol should be used for synergy effects between these two concepts. Especially for irregular Gabor multipliers an investigation of the eigenfunctions of these operators seem attractive. First experiments indicate that the eigenfunctions corresponding to big eigenvalues of the frame operator, as special case of a Gabor multiplier, have their peaks in the time-frequency plane at the sampling points, while "small" eigenfunctions "live" in between them. Last but not least it would be interesting to investigate perturbation results for well-balanced Gabor frames.

The algorithm for the inversion of a Gabor frame matrix by double preconditioning seems to be very useful in situations, where the calculation of the inverse frame operator or dual window is very expensive or cannot be done at all. For example in the situation of *quilted Gabor frames* [34] or the *Time-Frequency Jigsaw Puzzle* [72], there exists a frame, which globally is not a Gabor frame. Hence the dual Gabor window cannot be found, but the dual frame can be approximated by the dual windows of the local Gabor frame in these cases. It might be preferable to use a good and fast approximation of the local Gabor dual windows to a precise calculation of the local canonical dual, as precision is lost anyway at the approximation of the global dual frame. This application of the presented algorithm should be explored. Other issues which seem to justify future work are for example an investigation of more easily interpretable condition for the window, when the Jacobi algorithm is convergent. Furthermore the idea of double preconditioning can be extended by using other preconditioning matrices. For example such matrices as produced by projection using other commutative subgroups of the time-frequency plane. The new norms, the Walnut and Janssen norm, can be extended to infinite matrices and operators.

For the time frequency masking concept the most important future work will be the testing and validation by psychoacoustical experiments. Newer psychoacoustical models should be incorporated, for example using a outer-middle ear filtering (ISO-phones) or taking into account the absolute threshold associated with internal noise. As its is well-known fact that different classes of signal components have different masking effects, it seems very promising to investigate algorithm which separate signals in tonal, transient and statistical components. For the separation for example algorithm found in [72] or [89] can be used.

Appendix A

Mathematical Background

In this appendix we collect

1. the mathematical background, needed for the results in the main part, in this chapter. The author has decided to make that collection rather extensive to help to make that work more self-contained. Not all results have the exact citations, but in every section in the beginning you find references for some standard works, where you find all results, that don't have special citations.

2. the MATLAB-algorithms, produced for the main part, in the next one.

A.1 Basic Notations

With $K \subset\subset M$ we will mean a *compact* subset K of the set M.

We will use the words *injective* and *one-to-one*, *surjective* and *onto*, *kernel* and *null-space* as analogue formulations.

With χ_M we will denote the characteristic function of the set M:

$$\chi_m(x) = \begin{cases} 1 & x \in M \\ 0 & otherwise \end{cases}$$

For a list of symbols, see the main index under the heading "symbols".

A.2 Tonelli's And Fubini's Theorem

Taken from [63]: Let μ and ν be positive Borel measure on $\mathbb{R}^d$ and let $\mu \times \nu$ be their product measure on $\mathbb{R}^{2d}$.

Theorem A.2.1 (Tonelli) *If $f \geq 0$ on $\mathbb{R}^{2d}$, then*

$$\int\int_{\mathbb{R}^{2d}} f(x,\omega)d(\mu \times \nu) = \int_{\mathbb{R}^d} \left(\int_{\mathbb{R}^d} f(x,\omega)d\nu(x) \right) d\mu(\omega) =$$

$$= \int_{\mathbb{R}^d} \left(\int_{\mathbb{R}^d} f(x,\omega)d\mu(\omega) \right) d\nu(x).$$

This means that either all integrals are finite and equal or all are infinite.

Theorem A.2.2 (Fubini) *If $f \in L^1(\mathbb{R}^{2d}, \mu \times \nu)$ then the equations in Tonelli's theorem hold. Furthermore, for almost all $\omega \in R^d$ the section $x \mapsto f(x,\omega)$ is in $L^1(\mathbb{R}^d, \mu)$ and for almost all $x \in R^d$ the section $\omega \mapsto f(x,\omega)$ is in $L^1(\mathbb{R}^d, \nu)$. Further more $x \mapsto \int_{\mathbb{R}^d} f(x,\omega)d\nu(\omega)$ and $x \mapsto \int_{\mathbb{R}^d} f(x,\omega)d\mu(x)$ are in $L^1(\mathbb{R}^d, \mu)$ and $L^1(\mathbb{R}^d, \nu)$ respectively.*

This is of course also true for the discrete measure, so if $\sum_{k,n} |a_{k,n}| < \infty$,

$$\sum_{k,n} |a_{k,n}| = \sum_k \left(\sum_n |a_{k,n}| \right) = \sum_n \left(\sum_k |a_{k,n}| \right)$$

A.3 Linear Algebra

For details see for example [123] or [26].

A.3.1 Vector Spaces

Definition A.3.1 *A set V with the binary operations $+ : V \times V \to V$ and $\cdot : \mathbb{K} \times V \to V$ is called a* **vector space** *over the field $\mathbb{K}$, if*

1. *$(V, +)$ forms a commutative group, i.e.*
   (a) *$+$ is associative, i.e. $x + (y + z) = (x + y) + y$.*
   (b) *$+$ is commutative, i.e. $x + y = y + x$.*
   (c) *there exists a 0, such that $x + 0 = x$ for all $x \in V$.*
   (d) *for every $x \in V$ there exists an element $(-x)$ such that $x + (-x) = 0$.*

2. $\cdot$ *is associative, i.e* $\lambda \cdot (\mu \cdot x) = (\lambda \cdot \mu) \cdot x$

3. $\cdot$ *and* $+$ *are distributive.*

   (a) $(\lambda + \mu) \cdot x = \lambda \cdot x + \mu \cdot x$

   (b) $\lambda \cdot (x + y) = \lambda \cdot x + \lambda \cdot y$

4. *For* $1 \in \mathbb{K}$ *for all* $x \in V$ $1 \cdot x = x$.

Definition A.3.2 *Let* V *be a vector space. A function* $\|.\| : V \to \mathbb{R}^+$ *is called* **norm**, *if*

1. $\|\lambda \cdot x\| = |\lambda| \cdot \|x\|$

2. $\|x + y\| \leq \|x\| + \|y\|$

3. $\|x\| = 0 \Longleftrightarrow x = 0$

In the following V_i will denote normed vector spaces, i.e. vector spaces with a norm.

Definition A.3.3 *A sequence* (f_k) *is called* **linearly independent** *, if for all linear combinations, that are zero, the coefficients are zero.*

$$0 = \sum_{k \in K_f} c_k f_k, K_f \text{ finite } \Longrightarrow c_k = 0 \ \forall k \in K_f$$

The span of a sequence of elements $(f_k)_{k=}^N$ *in* V *is*

$$span(f_k) = \left\{ f \,\middle|\, \exists (c_k) \subseteq \mathbb{C}^N : f = \sum_{k=0}^{N} c_k f_k \right\}$$

A sequence $\{f_k\}$ *is called a (finite) basis for* V *if it spans* V *and is linearly independent.*

We will sometimes use the canonical basis elements $\delta_k = (0, \dots, 0, \overbrace{1}^{k}, 0, \dots, 0)$. This can be seen as a *Kronecker symbol*:

$$\delta_{i,j} = \begin{cases} 1 & i = j \\ 0 & otherwise \end{cases}$$

This can also be defined for infinite dimensional spaces. A periodization leads to the *Shah symbol*

$$Ш_M(j) = \sum_k \delta_{j,k \cdot M}.$$

A.3.2 Inner Product

Definition A.3.4 *We call a function* $\langle .,. \rangle : V \times V \to \mathbb{K}$ **inner product**, *if and only if for all* $x, y \in V$

1. $x \mapsto \langle x, y \rangle$ *is linear* ,
2. $\langle x, y \rangle = \overline{\langle y, x \rangle}$,
3. $\langle x, x \rangle \geq 0$
4. $\langle x, x \rangle = 0 \iff x = 0$.

Every inner product induces a norm by $\|x\|_V = \sqrt{\langle x, x \rangle}$.

Theorem A.3.1 (Cauchy-Schwarz Inequality) *For all* $x, y \in V$ *we have*

$$|\langle x, y \rangle|^2 \leq \langle x, x \rangle \cdot \langle y, y \rangle$$

Especially important for applications (e.g., acoustics) are finite-dimensional spaces. We regard discrete signals $x = (x_0, x_1, \ldots, x_{n-1}) \in \mathbb{C}^n$ as row vectors. On this vector space we have a (Euclidean) norm $\|x\|$ which is induced by the scalar product $\langle x, y \rangle = \sum_{i=0}^{n-1} x_i \overline{y}_i$.

A.3.3 Norms in $\mathbb{C}^n$

Apart from the Euclidean norm we can define

Definition A.3.5 *On* $\mathbb{C}^n$ *let*

$$\|x\|_p = \sqrt[p]{\sum_{i=0}^{n-1} |x_i|^p}$$

be the **p-norm** *and*

$$\|x\|_\infty = \max_{i=0}^{n-1} \{|x_i|\}$$

the **infinity norm**.

The Euclidean norm, defined above, is clearly equivalent to the 2-norm.

Corollary A.3.2 $(\|x\| + \|y\|)^2 \leq 2 \cdot \left(\|x\|^2 + \|y\|^2\right)$

Proof:

$$(\|x\| - \|y\|)^2 \geq 0$$

and therefore

$$\|x\|^2 + \|y\|^2 \geq 2 \cdot \|x\| \, \|y\|$$

□

Clearly a similar argument can be used for $\left\| \sum_{i \in I} x_i \right\|^2 \leq |I| \sum_{i \in I} \|x_i\|^2$.

On finite-dimensional vector spaces all norms have to be equivalent, cf. [129]:

Proposition A.3.3 *In $\mathbb{C}^n$ we have*

1.
$$\|x\|_2 \leq \|x\|_1 \leq \sqrt{n} \, \|x\|_2$$

2. *for all* $1 \leq p \leq \infty$
$$\|x\|_p \leq \|x\|_1$$
and
$$\|x\|_\infty \leq \|x\|_p \leq \sqrt[p]{n} \, \|x\|_\infty$$

3. *for* $1 \leq p \leq q$
$$\|x\|_q \leq \|x\|_p \leq \sqrt[p]{n} \, \|x\|_q$$

A.3.4 Linear Functions

Definition A.3.6 *A function $O : V_1 \to V_2$ is called* **linear***, if and only if*

$$O(x + y) = O(x) + O(y) \text{ and } O(\lambda \cdot x) = \lambda \cdot O(x)$$

for $x, y \in V_1$ and $\lambda \in \mathbb{C}$.

The class of all linear functions from V_1 to V_2 will be denoted by $L(V_1, V_2)$ and $L(V_1) = L(V_1, V_1)$.

Definition A.3.7 *A function $O : V_1 \to V_2$ between normed spaces is called* **bounded***, if and only if there exits an $M > 0$ such that*

$$\|O(x)\|_{V_2} \leq M \cdot \|x\|_{V_1}$$

Definition A.3.8 *The minimum of all M possible in the above inequality is called the* **operator norm** *induced by the vector norms $\|.\|_{V_!}$ and $\|.\|_{V_2}$:*

$$\|O\|_{Op} = \sup_{\|x\|_{V_1} \leq 1} \left\{ \|O(x)\|_{V_2} \right\}$$

Definition A.3.9 *1. The linear and bounded functions from V_1 to V_2 are called* **operators**. *The class of this functions will be denoted by $\mathcal{B}(V_1, V_2)$*

2. The operators from V to $\mathbb{K}$ are called **functionals**. *The class of this functions will be denoted by V' and will be called the* **dual space**.

Again we will use the notation $\mathcal{B}(V_1)$ for $\mathcal{B}(V_1, V_2)$.

Theorem A.3.4 *Let $O : V_1 \to V_2$ be a linear operator. Then the following properties are equivalent:*

1. *O is continuous*
2. *O is continuous in* 0
3. *O is bounded*
4. *O is uniformly continuous.*

The dual space of a normed vector space is a normed vector space with norm $\|x'\|_{\mathfrak{B}'} = \sup_{\|x\|_{\mathfrak{B}} \leq 1} \{x'(x)\}$. But also the reverse is true:

Corollary A.3.5 ([129] III.1.7.)

$$\|x\| = \sup_{\|x'\|=1} \{|x'(x)|\}$$

Definition A.3.10 *Let $O : V_1 \to V_2$ be an operator. Then the* **adjoint operator** *is $O^* : V_2' \to V_1'$ defined by*

$$(O^* y')(x) = y'(Ox)$$

for $x \in V_1$, $y' \in V_2'$.

Definition A.3.11 *A function A is called* **isometry** *if $\|Ax\| = \|x\|$*

Lemma A.3.6 *Let $B \in \mathcal{B}(V_1, V_2)$ and $A \in \mathcal{B}(V_3, V_1)$ a surjective isometry, $C \in \mathcal{B}(V_2, V_3)$ an isometry, then*

$$\|B\|_{Op} = \|B \circ A\|_{Op} = \|C \circ B\|_{Op} = \|C \circ B \circ A\|_{Op}$$

Proof:

$$\|B\|_{Op} = \sup_{x\in X}\left\{\frac{\|Bx\|_{\mathcal{H}}}{\|x\|_{\mathcal{H}}}\right\} = \sup_{z\in Z'}\left\{\frac{\|BAz\|_{\mathcal{H}}}{\|Az\|_{\mathcal{H}}}\right\} =$$
$$= \sup_{z\in Z'}\left\{\frac{\|BAz\|_{\mathcal{H}}}{\|z\|_{\mathcal{H}}}\right\} = \|BA\|_{Op}$$

and

$$\|CB\|_{Op} = \sup_{x\in X}\left\{\frac{\|CBx\|_{\mathcal{H}}}{\|x\|_{\mathcal{H}}}\right\} = \sup_{x\in X}\left\{\frac{\|Bx\|_{\mathcal{H}}}{\|x\|_{\mathcal{H}}}\right\} = \|B\|_{Op}$$

□

A.3.5 Matrices

Every linear operator $A : \mathbb{C}^n \to \mathbb{C}^m$ can be identified with a matrix. Respectively the action of every operator corresponds to a matrix vector multiplication:

$$A(x) = x \cdot A = \sum_{j=0}^{n} a_{i,j} x_i,$$

where $A = (a_{i,j})_{m,n}$ is an $m \times n$ matrix, $A \in M_{m,n}$. The notation A^T will signify the transpose of the matrix A: $\left(A^T\right)_{i,j} = A_{j,i}$. The adjoint of a matrix A is $A^* = \overline{A^T}$.

There are nice ways to interpret the matrix multiplication respectively the matrix-vector multiplication:

Lemma A.3.7

$$T \cdot \begin{pmatrix} | & | & & | \\ g_1 & g_2 & \dots & g_M \\ | & | & & | \end{pmatrix} = \begin{pmatrix} | & | & & | \\ Tg_1 & Tg_2 & \dots & Tg_M \\ | & | & & | \end{pmatrix}$$

$$\begin{pmatrix} | & | & & | \\ g_1 & g_2 & \dots & g_M \\ | & | & & | \end{pmatrix} \cdot \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_M \end{pmatrix} = \sum_{m=1}^{M} c_m g_m$$

$$\left(\begin{pmatrix} - & \overline{h_1} & - \\ - & \overline{h_2} & - \\ \vdots & & \vdots \\ - & \overline{h_N} & - \end{pmatrix} \cdot \begin{pmatrix} | & | & & | \\ g_1 & g_2 & \dots & g_M \\ | & | & & | \end{pmatrix}\right)_{m,n} = \langle g_n, h_m \rangle$$

For any matrix M let us use the notation M_i for the ith columns, $M^{(j)}$ the j-the row and $M_{i,j}$ the entry at the i-th row and j-th column. Note that $M_i = M \cdot \delta_i$.

For $v \in \mathbb{C}^L$ let $M = \mathfrak{diag}(v)$ be the diagonal matrix, for which $M_{i,j} = \delta_{i,j} v_i$. . For $M \in M_{n,n}$ let $d = \mathfrak{diag}(M)$ be the diagonal of M, $d_i = M_{i,i}$. These are clearly linear operators.

A.3.5.1 Matrix Norms And Spaces

Definition A.3.12 *Let A be an m by n matrix, then*

$$\|A\|_{Op} = \sup_{x \in \mathbb{C}^n : \|x\|=1} \{\|A \cdot x\|\}$$

is the (induced) **operator norm**. *Also,*

$$\|A\|_{fro} = \sqrt{\sum_{i=0}^{n-1} \sum_{j=0}^{m-1} |a_{i,j}|^2}$$

is the **Frobenius** *or* **Hilbert Schmidt norm**.

The Frobenius norm can be defined by the Hilbert-Schmidt inner product, $\|A\|_{fro} = \langle A, A \rangle_{HS}$, where

$$\langle A, B \rangle_{HS} = \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} a_{i,j} \bar{b}_{i,j}$$

Together with this norm, the space of all $m \times n$ matrices $M_{m,n}$ forms a Hilbert space. This provides us with a number of Hilbert space tools like orthonormal bases and the uniqueness of the best approximation on subspaces. The space $M_{m,n}$ is isomorphic to $\mathbb{C}^{m \cdot n}$ (for example by writing the columns one below each other, i.e. using $\mathfrak{vec}^{(n)}$ from Lemma 1.2.25), in this case, the Hilbert-Schmidt inner product coincides with the ordinary scalar product, see Lemma 1.2.25), such that

$$\langle M, S \rangle_{HS} = \sum_{k=0}^{n-1} \langle M_k, S_k \rangle_{\mathbb{C}^m} =$$

$$= \left\langle \mathfrak{vec}^{(m)}(M), \mathfrak{vec}^{(m)}(S) \right\rangle_{\mathbb{C}^{m \times n}} = \sum_{p=0}^{L-1} \sum_{q=0}^{L-1} M_{p,q} \cdot \overline{S_{p,q}}$$

where we define $M_{(k)}$ as the k-th column of a matrix M.

For this inner product a lot of nice properties are valid, like

Proposition A.3.8 *For $m \times n$ matrices A, B, X*

- $\langle XA, B \rangle_{HS} = \langle A, X^* B \rangle_{HS}$
- $\langle AX, B \rangle_{HS} = \langle A, BX^* \rangle_{HS}$

This can be proved just as in the continuous case, see [110].

We find a generalization of the Hilbert Schmidt norm by

Definition A.3.13 *Let A be an m by n matrix, then for $1 \leq p, q < \infty$*

$$\|A\|_{p,q} = \left(\sum_{j=0}^{m-1} \left(\sum_{i=0}^{n-1} |a_{i,j}|^p \right)^{\frac{q}{p}} \right)^{\frac{1}{q}}$$

is called the **mixed norm**. *The definition above extends in a natural way to $p = \infty$ as follows:*

$$\|A\|_{\infty,p} = \left(\sum_{j=0}^{n-1} \left(\max_{i=0,\ldots,n-1} \{|a_{i,j}|\} \right)^p \right)^{\frac{1}{p}}$$

This can be extended to infinite matrices to get the matrix space $l^{p,q}$ and weights to get $l^{p,q}_m$, $\|M\|_{l^{p,q}_m} = \|m \cdot M\|_{l^{p,q}}$.

A.3.5.2 Spectral Radius

For details in this section see [87] or [123].

Definition A.3.14

1. *A complex number λ is called* **eigenvalue** *of the matrix A if there is a vector x, called* **eigenvector** *, such that*

$$Ax = \lambda x$$

2. *The set*

$$\sigma(A) = \{\lambda \,|\, \lambda \text{ is eigenvalue of } A\}$$

is called the **spectrum**.

3. *The number*

$$\rho(A) = \max\{|\lambda|, \lambda \in \sigma(A)\}$$

is called the **spectral radius** .

Theorem A.3.9 *Let A be a $m \times n$ matrix, then*

1. $\|A\|_{Op} = \sqrt{\rho(A * A)}$

2. $\rho(A) \leq \|A\|_{Op}$

3. $\rho(A) = \|A\|_{Op}$ *for* A *self-adjoint.*

4. $\rho(A) = \lim\limits_{n\to\infty} \|A^n\|_{Op}^{1/n}$ (spectral radius formula) .

This is true also for operator norms induced by other vector norms, e.g. $\|x\|_1 = \sum_{i=0}^{n-1} |x_i|$ or $\|x\|_\infty = \max\limits_{i=0,\ldots,n-1} |x_i|$.

A.3.6 Discrete Fourier Transformation

For details on the discrete Fourier transformation, see for example [97] or [75].

Definition A.3.15 *Let* $\mathcal{F}_L$ *be the* $L \times L$ *matrix with entries* $(\mathcal{F}_L)_{j,k} = \omega_L^{-jk}$ *with* $\omega_L = e^{\frac{2\pi i}{L}}$. *We call* $\mathcal{F}_L$ *the* **Fourier** *or* **FFT-matrix**. *We will write* $\hat{x}$ *for* $\mathcal{F}_L \cdot x$.

Theorem A.3.10 ([23] Theorem 1.4.1) *The vectors* $f_k(l) = \frac{1}{\sqrt{L}} \omega_L^{k \cdot l}$ *constitute an orthonormal basis for* $\mathbb{C}^L$.

It can be shown [123] that $\mathcal{F}_l$ is unitary matrix and so is just a change of basis. With the above theorem we have all the results for ONBs, Section A.4.3.2, like the Parseval or Plancherel theorems, where it is important to remember the normalization factor $\frac{1}{\sqrt{L}}$.

A.3.6.1 Convolution

The convolution of two vectors in $\mathbb{C}^n$ is defined by

$$(x * y)_k = \sum_{i=0}^{n-1} x_i \cdot y_{k-i}$$

the convolution of two $m \times n$ matrices by

$$(A * B)_{k,l} = \sum_{i_1=0}^{m-1} \sum_{i_2=0}^{n-1} A_{i_1,i_2} \cdot B_{k-i_1,l-i_2}$$

As we regard vectors as well as columns and rows of matrices as periodic, this is a cyclic convolution.

The convolution of two vectors corresponds to the pointwise multiplication of their Fourier transformation

$$\widehat{x * y} = \hat{y} \cdot \hat{x}$$

For a similar result for matrices see Lemma 3.2.9.

A.3.6.2 Poisson Theorem

The discrete version of the Poisson Theorem A.4.50 can be written

Theorem A.3.11 *For $g \in \mathbb{C}^L$ we have for $l = 0, \dots, a-1$*

$$\left(\widehat{\sum_{k=0}^{L-1} T_{ka} g} \right)_l = (\hat{g})_{l \cdot \frac{L}{a}}$$

with the left Fourier transformation in $\mathbb{C}^a$, the right in $\mathbb{C}^L$.

A.3.7 Kronecker product

Definition A.3.16 *Let A be a $p \times q$, B a $r \times s$ matrix. Then the* **Kronecker product** of A and B is the $p \cdot r \times q \cdot s$ matrix C with

$$C_{i,j} = a_{\left\lfloor \frac{i}{r} \right\rfloor, \left\lfloor \frac{j}{s} \right\rfloor} \cdot b_{i \bmod r, j \bmod s}$$

$$A \otimes B = \begin{pmatrix} a_{0,0}B & a_{1,0}B & \dots & a_{p-1,0}B \\ a_{0,1}B & a_{1,1}B & \dots & a_{n-1,1}B \\ \vdots & \vdots & \vdots & \vdots \\ a_{0,q-1}B & a_{1,q-1}B & \dots & a_{p-1,q-1}B \end{pmatrix}$$

Proposition A.3.12 *Properties:*

- $(A \otimes B) \otimes C = A \otimes (B \otimes C)$
- $(A + B) \otimes C = A \otimes C + B \otimes C$
- $(A \otimes B)^* = A^* \otimes B^*$
- $(A \otimes B) \cdot (C \otimes D) = (A \cdot C) \otimes (B \cdot D)$.
- $(A \otimes B)^{-1} = A^{-1} \otimes B^{-1}$

This can be shown directly or see e.g. [83].

A.3.8 Hadamard product

We will sometimes use the pointwise product of two matrices. We will use the following notation

Definition A.3.17 *Let A and B be two $m \times n$ matrices, then*

$$(A \odot B)_{i,j} = A_{i,j} \cdot B_{i,j}$$

A.4 Functional analysis

For details, see e.g. [129] or [26].

A.4.1 Functions

Definition A.4.1 *Let $f : V_1 \to V_2$ be a function. Then*

1. *The* **kernel** *or* **null-space** *of f is*

$$ker(f) = \{x \in V_1 \,|\, f(x) = 0\}$$

2. *The* **range of f is**

$$ran(f) = \{y \in V_2 \,|\, \exists x : f(x) = y\}$$

Definition A.4.2 *The support of a function f between topological vector spaces is*

$$supp(f) = \overline{\{x \,|\, f(x) \neq 0\}}$$

A.4.2 Banach spaces

Definition A.4.3 *A vector space V is called* **complete**, *if every Cauchy sequence converges, i.e. for every sequence (x_n)*

$$\|x_n - x_m\|_V \to 0 \text{ for } n, m \to \infty \Longrightarrow \exists x : \|x_n - x\|_V \to 0 \text{ for } n \to \infty$$

A complete normed vector space $(\mathfrak{B}, \|.\|)$ is called a Banach space.

Let in the following $\mathfrak{B}_i$ be complex Banach spaces.

Proposition A.4.1 *Let V be a normed vector space and $\mathfrak{B}$ a Banach space, then $\mathcal{B}(V, \mathfrak{B})$ is complete. As every finite-dimensional space is complete, V' is complete.*

A.4.2.1 Unconditional Convergence

Definition A.4.4 *Let* $(f_k)_{k\in K}$ *be a countable set in* $\mathfrak{B}$*. The series* $\sum\limits_{k\in K} f_k$ *is said to* **converge unconditionally** *to* $f \in \mathfrak{B}$*, if for all* $\epsilon > 0$ *there exists a finite set* $F_0 \subseteq K$ *such that*

$$\left\| f - \sum_{k\in F} f_k \right\|_{\mathfrak{B}} < \epsilon \text{ for all finite sets } F \supseteq F_0$$

A more intuitive interpretation of unconditional convergence is convergence independent of permutation, which can be seen from the following result:

Proposition A.4.2 ([63] Proposition 5.3.1) *Let* $(f_k)_{k\in K}$ *be a countable set in* $\mathfrak{B}$*. Then the following properties are equivalent:*

1. $f = \sum\limits_{k\in F} f_k$ *converges unconditionally*
2. *For every enumeration* $\pi : \mathbb{N} \to K$ *the sequence of partial sums* $\sum\limits_{k=1}^{N} f_{\pi(k)}$ *converges to f.*

So the limit f *is independent of the enumeration* π*.*

Lemma A.4.3 ([63] Lemma 5.3.3) *Suppose that* $\sum\limits_{k,l} f_{k,l}$ *converges unconditionally to* $f \in \mathcal{B}$*. Then the inner partial sum* $s_{k,N} = \sum\limits_{|l|\leq N} f_{k,l}$ *converges to an element* $g_k \in \mathcal{B}$ *for each* k *and* $f = \sum\limits_{k} g_k$*. Likewise* $\sum\limits_{|k|\leq M} f_{k,l}$ *converges to an element* $h_l \in \mathcal{B}$ *for each* k *and* $f = \sum\limits_{l} h_l$*.*

Thus the order of summation can be interchanged in the double sum.

A.4.2.2 Bases In Banach Spaces

Definition A.4.5 *A sequence* $(f_k) \subseteq \mathfrak{B}$ *is called* **complete** *or* **total** *in* $\mathfrak{B}$ *if* $\overline{span\{f_k\}} = \mathcal{B}$*.*

For infinite-dimensional space it is important to note that the set

$$\text{"}span_\infty(f_k)\text{"} :=$$

$$= \left\{ f = \lim_{n\to\infty} \sum_{k\in K_n} c_k f_k \,|K_1 \subseteq \ldots \subseteq K_n \subseteq \ldots \subseteq K \text{ with } |K_n| \leq n, c_k \in \mathbb{K} \right\}$$

is in general not identical to $\overline{span}(f_k)$. An important question is, which sequence nevertheless have this property.

Definition A.4.6 *A sequence* (e_k) *is called a* **(Schauder) basis** *for the Banach space* $\mathcal{B}$*, if for each* $f \in \mathcal{B}$ *there are unique scalar coefficients* $c_k(f)$*, such that*

$$f = \sum_k c_k(f) e_k$$

Apart from the *linear independence* for infinite-dimensional spaces other definitions of the "independence" of sequences make sense:

Definition A.4.7

1. (f_k) *is called* ω**-independent** *if whenever* $\sum\limits_K c_k f_k$ *is convergent and equal to zero then* $c_k = 0$ *for all* $k \in K$.
2. (f_k) *is* **minimal** *if* $f_j \notin \overline{span}\{f_k\}_{k \neq j}$ *for all* $j \in K$.

These concepts are connected by the following chain, see [23]

$$\text{minimal} \Longrightarrow \omega\text{-independent} \Longrightarrow \text{linearly independent.}$$

A.4.2.3 Operators In Banach Spaces

Proposition A.4.4 ([129] II.1.5) *Let* $D \subseteq \mathfrak{B}_1$ *be a dense subspace, and* $T \in \mathcal{B}(D, \mathfrak{B}_2)$*. Then there exists a uniquely defined operator* $\tilde{T} \in \mathcal{B}(\mathfrak{B}_1, \mathfrak{B}_2)$*, such that* $\tilde{T}_{|D} = T$ *and* $\|T\|_{Op} = \left\|\tilde{T}\right\|_{Op}$.

The dual space of a Banach space is always also a Banach space with norm $\|x'\|_{\mathfrak{B}'} = \sup\limits_{\|x\|_{\mathfrak{B}}} \{x'(x)\}$.

Definition A.4.8 *An operator* $\in \mathcal{B}(\mathfrak{B}_1, \mathfrak{B}_2)$ *is called an* **isomorphism** *if it bijective and has a bounded inverse.*

It is clear that isomorphisms are exactly those surjective operators C for which

$$A \cdot \|f\|_{\mathcal{H}} \leq \|C(f)\|_2 \leq B \cdot \|f\|_{\mathcal{H}}$$

A.4.2.4 Open Mapping Theorem

Theorem A.4.5 (Open Mapping Theorem) *Let $\mathfrak{B}_1$ and $\mathfrak{B}_2$ be Banach spaces and let $T : \mathfrak{B}_1 \to \mathfrak{B}_2$ be linear, bounded and surjective. Then T is an open mapping, i.e. mapping open sets to open sets.*

A direct consequence is the following corollary

Corollary A.4.6 [129] IV.3.6 *Let $T : \mathfrak{B}_1 \to \mathfrak{B}_2$ be a bounded linear injective map between Banach spaces. $T^{-1} : ran(T) \to \mathfrak{B}_1$ is continuous if and only if $ran(T)$ is closed.*

As a simple consequence the converse is true:

Corollary A.4.7 *$T \in \mathcal{B}(X, Y)$, X, Y Banach spaces. T is injective (one-to-one) and has closed range if and only if there exists number $c > 0$ such that there exists number $c > 0$ such that*

$$\|x\| \le c \cdot \|Tx\| \;\; \forall x \in X$$

Proof: Injective and closed range $\overset{A.4.6}{\iff}$

$$\exists B : \left\|T^{-1}y\right\| \le B \cdot \|y\| \;\; \forall y \in ran(T) \iff$$

$$\exists B : \|x\| \le B \cdot \|Tx\| \;\; \forall x \in X$$

□

It can be shown that injective bounded operators map minimal sets to minimal sets. (Use A.4.7 and suppose the converse.) But contrary to the finite dimensional case that does not mean, that the function is necessarily surjective, as can be seen in the next example.

Example A.4.1 :

Let $r_S : l^1 \to l^1$ be the *right shift*, meaning $r_S(c_1, c_2, c_3, ...) = (0, c_1, c_2, ...)$. This is an isometry (and so injective and bounded) but not surjective.

Let $\mathcal{C} : l^1 \to l^1$ defined by $\mathcal{C}(x) = y$ with $y_k = \frac{x_k}{k}$. Then $\mathcal{C}$ is linear, injective and bounded (with bound $\frac{\pi}{\sqrt{6}}$). But the inverse is clearly not bounded. So this injective, bounded operator does not have a closed range.

Let X be a Banach space and $U \subseteq X$, $V \subseteq X'$, then $U^\perp := \{x' \in X' : x'(x) = 0 \; \forall \; x \in U\}$ and $V_\perp := \{x \in X : x'(x) = 0 \; \forall \; x' \in V\}$

Theorem A.4.8 [129] IV.5.1 *Let $T \in \mathcal{B}(X,Y)$. Then the following statements are equivalent*

- $ran(T)$ *is closed*
- $ran(T) = ker(T^*)_\perp$
- $ran(T^*)$ *is closed*
- $ran(T^*) = ker(T)^\perp$.

One part of this is also know as the *Closed Range Theorem* : Let T be bounded. Then T has closed range if and only if T^* does.

Proposition A.4.9 ([23] A.5.3.)*If $U : \mathfrak{B}_1 \to \mathfrak{B}_2$ is bounded and $\|I-U\|_{Op} < 1$, then U is invertible and*

$$U^{-1} = \sum_{k=0}^{\infty} (I-U)^k$$

Furthermore $\|U^{-1}\|_{Op} \leq \frac{1}{1-\|I-U\|_{Op}}$.

Initialization:

$$x_0 = g, h_0 = g, A = Id - S$$

Iteration :

- Set $h_{k+1} = Ah_k$;
- Set $x_{k+1} = x_k + h_{k+1}$;
- Check exit condition $\|x_{k+1} - x_k\| < \epsilon$.

Figure A.1: The Neumann algorithm

A.4.2.5 Banach Algebra

For detail see e.g. [102].

Definition A.4.9 *A Banach space $\mathfrak{B}$ for which a binary operation $\cdot : \mathfrak{B} \times \mathfrak{B} \to \mathfrak{B}$ is defined, such that for all $\lambda \in \mathbb{K}$ and $x, y, z \in \mathfrak{B}$*

1. (a) $x \cdot (y \cdot z) = (x \cdot y) \dot{z}$

(b) $(x+y)\,z = x \cdot z + y \cdot z$

(c) $x\,(y+z) = x \cdot y + x \cdot z$

(d) $\lambda \cdot (x \cdot y) = (\lambda \cdot x) \cdot y = x \cdot (\lambda \cdot y)$

2. $\|x \cdot y\|_{\mathfrak{B}} \leq \|x\|_{\mathfrak{B}} \cdot \|y\|_{\mathfrak{B}}$

is called a **Banach algebra**.

This can be seen in the following way: A Banach algebra is a Banach space that is also an algebra with unity, that "respects" the norm.

Definition A.4.10 *A function between Banach algebras* $\varphi : \mathfrak{B}_1 \to \mathfrak{B}_2$ *is called a* **Banach algebra homomorphism**, *if*

1. φ *is linear and*
2. *for all* $x, y \in \mathfrak{B}_1$ *we have* $\varphi(x \cdot y) = \varphi(x) \cdot \varphi(y)$.

It is called a **monomorphism**, *if it is injective also.*

A.4.3 Hilbert Spaces

Definition A.4.11 *A complete vector space with inner product* $(\mathcal{H}, \langle .,. \rangle)$ *is called a* Hilbert space.

Let in the following $\mathcal{H}_i$ be complex Hilbert spaces.

A.4.3.1 Bases in Hilbert spaces

Definition A.4.12 *Two sequences* (g_k), (f_k) *in a Hilbert space are called* **biorthogonal** *if*

$$\langle g_k, h_j \rangle = \delta_{kj}$$

Lemma A.4.10 ([23] Lemma 3.3.1) *Let* $(f_k)_{k \in K}$ *be a sequence in* $\mathcal{H}$. *Then*

1. *if and only if there is biorthogonal sequence* $(g_k)_{k \in K}$, *then* (f_k) *is minimal.*
2. *if* (f_k) *has a biorthogonal sequence, then it is unique if and only if* (f_k) *is a total.*

A.4.3.2 ONBs

Definition A.4.13 *1. A sequence* $(e_k)_{k\in K} \subseteq \mathcal{H}$ *is called* **orthonormal**, *if it is biorthogonal to itself, i.e.*

$$\langle e_k, e_j \rangle = \delta_{k,j} \text{ for all } k, j \in K$$

2. A **orthonormal basis** *is a sequence, that is a basis and orthonormal.*

Theorem A.4.11 ([23] 3.4.2)*For an orthonormal sequence* $(e_k)_{k\in K} \subseteq \mathcal{H}$ *the following properties are equivalent:*

1. (e_k) *is an ONB*
2. *For all* $f \in \mathcal{H}$
$$f = \sum_{k\in K} \langle f, e_k \rangle e_k$$
3. *For all* $f, g \in \mathcal{H}$
$$\langle f, g \rangle = \sum_{k\in K} \langle f, e_k \rangle \overline{\langle g, e_k \rangle}$$
4. *For all* $f \in \mathcal{H}$
$$\|f\|_{\mathcal{H}}^2 = \sum_{k\in K} |\langle f, e_k \rangle|^2$$
5. (e_k) *is complete*
6. *If* $\langle f, e_k \rangle = 0$ *for all* $k \in K$*, then* $f = 0$.

Theorem A.4.11 3.) is known as *Plancherel's theorem.* Theorem A.4.11 4.) is known as *Parseval's theorem.*

A.4.3.3 Operators In Hilbert Spaces

Every functional of a Hilbert space can be represented by an inner product:

Theorem A.4.12 (Riesz' representation theorem, [26] Ch. 3, 3.4.)*Let* $F \in \mathcal{B}(\mathcal{H}, \mathbb{K})$*, then there is a unique* f_0 *such that*

$$F(x) = \langle x, f_0 \rangle$$

for every $x \in \mathcal{H}$*. Moreover* $\|F\|_{Op} = \|f_0\|_{\mathcal{H}}$.

So Corollary A.3.5 can be restated in Hilbert spaces:

Corollary A.4.13

$$\|x\| = \sup_{\|y\|=1} \{|\langle x, y\rangle|\}$$

Proof: See [129]. It is also easy to show directly: Let $c = \sup_{\|y\|=1} \{|\langle x, y\rangle|\}$

On one hand $c \leq \|x\|$ because $c = \sup_{\|y\|=1} \{|\langle x, y\rangle|\} \overset{C.S.}{\leq} \|x\|$. On the other hand set $y = \frac{x}{\|x\|}$. $\square$

Definition A.4.14 *Let O be an operator $\mathcal{H}_1 \to \mathcal{H}_2$, then we called the uniquely defined operator O^* for which for all $x, y \in \mathcal{H}_1$*

$$\langle Ox, y\rangle = \langle x, O^* y\rangle$$

the **adjoint operator**.

This is exactly the same adjoint operator defined in Section A.3.4 .

Definition A.4.15 *An operator $T \in \mathcal{B}(\mathcal{H}_1, \mathcal{H}_2)$ is called*

1. **unitary**, *if T is invertible, and $T^{-1} = T^*$.*
2. **self-adjoint**, *if $\mathcal{H}_1 = \mathcal{H}2$ and $T = T^*$.*
3. **normal**, *if $T^*T = TT^*$.*

Proposition A.4.14 ([129] V.5.2. (f)) *For $A \in \mathcal{B}(\mathcal{H}_1, \mathcal{H}_2)$ we have $A^* \in \mathcal{B}(\mathcal{H}_2, \mathcal{H}_1)$ and $\|A^*\| = \|A\|$ and $\|AA^*\| = \|A\|^2$.*

Proposition A.4.15 ([129] V.5.7) $\|T\|_{Op} = \sup_{\|f\|_{\mathcal{H}} \leq 1} |\langle Tf, f\rangle|$ *for self-adjoint operators $T \in \mathcal{B}(\mathcal{H}_1, \mathcal{H}_2)$.*

Lemma A.4.16 [129] *Let $T \in \mathcal{B}(\mathcal{H}_1, \mathcal{H}_2)$. Then all the following statements are true*

- $\overline{ran(T)} = ker(T^*)^{\perp}$. *So T is injective if and only if T^* has dense range.*
- $\overline{ran(T^*)} = ker(T)^{\perp}$. *So T^* is injective if and only if T has dense range.*
- $ran(T)^{\perp} = ker(T^*)$.
- $ran(T^*)^{\perp} = ker(T)$.

So from this we know that $ran(T) \subseteq ker(T^*)^\perp$, so to prove that T has a closed range, it is enough to show that $ker(T^*)^\perp \subseteq ran(T)$ (following A.4.8) or the same with switched roles.

Proposition A.4.17 *[26] Let $A \in \mathcal{B}(\mathcal{H}_1, \mathcal{H}_2)$. Then*

$$\langle Af, f \rangle = 0 \ \forall f \in \mathcal{H}_1 \implies A = 0.$$

Definition A.4.16 *An operator $T \in \mathcal{B}(\mathcal{H}_1, \mathcal{H}_2)$ is called* **positive**, *if for all $x \in \mathcal{H}_1$, $x \neq 0$*

$$\langle Tx, x \rangle > 0$$

Proposition A.4.18 *If A is a positive operator, then A is injective.*

Proof: $\langle Ac, c \rangle > 0 \ \forall c \neq 0 \Rightarrow (Ac = 0 \Rightarrow c = 0)$ □

In the finite dimensional case this already means that the matrix is invertible.

We know, [129] V.5.6, that an operator is self-adjoint if and only if $< Tx, x > \in \mathbb{R}$. This also means that every positive Operator is self-adjoint.

A.4.3.4 Matrix Representation Of Operators

Let us call elements of $l^\infty(\mathbb{Z}^2)$ infinite matrices. Define the action of such a matrix M on l^p formally as

$$(Mc)_j = \sum_k M_{j,k} c_k \tag{A.1}$$

for $c \in l^p$. The sum is given formally as we don't know if it converges.

If for two matrices M, N the sum in Equation A.1 converges unconditionally for all $c \in l^p$, then this is also true for $M \circ N$ and the matrix of this operator is just the product of the two matrices with the well-known matrix multiplication. If the matrix M induces an operator we will denote that by $\mathcal{O}(M)$.

Every linear, bounded operator $A : \mathcal{H}_1 \to \mathcal{H}_2$ can be written as an infinite matrix with the entries $a_{ij} = \langle Ae_i, f_j \rangle$ with (e_i) and ONB in $\mathcal{H}_1$ and (f_l) in $\mathcal{H}_2$. Then $Af = \sum\limits_k \left(\sum\limits_j a_{i,j} \langle f, \tilde{e}_j \rangle \right) f_i$. The matrix M is clearly in $l^{\infty,\infty}$ as $|\langle Ae_i, f_j \rangle| \leq \|A\|_{Op}$. We denote $M = \mathcal{M}(A)$.

See Section 1.1.7.3 for an extension to frames.

For the opposite direction we can state *Schur's lemma*:

Lemma A.4.19 ([63] 6.2.1) *Let $A = (a_{ij})$ be an infinite matrix with*

$$\sup_j \sum_k |a_{jk}| \leq K_1$$

$$\sup_k \sum_j |a_{jk}| \leq K_2$$

Then the operator defined by this matrix is bounded from l^p to l^p

An equivalent lemma is possible for integral operators. If the matrix is self-adjoint, clearly one of the above conditions suffices.

A.4.3.5 Multipliers

Let us collect the result for this special class of operators, connected to the main subject of this work:

Theorem A.4.20 ([26] Ch. II 1.5 Theorem) *Let $\phi \in L^\infty(\mathbb{K}^d)$, define $M_\phi : L^2(\mathbb{K}^d) \to L^2(\mathbb{K}^d)$ by $M_\phi(f) = \phi \cdot f$. Then M_ϕ is bounded and linear and $\|M_\phi\| = \|\phi\|_\infty$.*

This is valid for all L^p with $1 \leq p \leq \infty$. If $1 \leq p < \infty$ and $1/p + 1/p' = 1$ then $M_\phi^* : L^{p'} \to L^{p'}$ and $M_\phi = M_\phi^*$

The following statement can also be found in the same book[26]:

- $M_\phi^2 = M_\phi \iff \phi$ is a characteristic function χ_A.
- Let $\{a_n\}$ be a sequence, $A : l^2 \to l^2$ is bounded if and only if $\{a_n\}$ is uniformly bounded. (Then $\|A\| = \|\{a_n\}\|_\infty$.)
- No nonzero multiplication operator is compact on $L^2(0,1)$.

Theorem A.4.21 ([26] Ch. II 4.6 Theorem) *If N is a normal operator on $\mathcal{H}$, then there is a measure space (X, Ω, μ) and a function $\phi \in L^\infty(X, \Omega, \mu)$ such that N is unitarily equivalent to M_ϕ on $L^2(X, \Omega, \mu)$.*

A.4.4 Tensor Products

From algebra [76] we know that

Definition A.4.17 *Let X, Y, Z be modules over a ring R, then a function $\otimes : X \times Y \to Z$ is called* **tensor product**, *if it is* **bilinear** , *meaning for $\forall a, b \in X$, $\forall c, d \in Y$ and $\forall \lambda \in R$*

$$(a + b) \otimes c = a \otimes c + b \otimes c$$

$$a \otimes (c + d) = a \otimes c + a \otimes d$$
$$(\lambda \cdot a) \otimes c = a \otimes (\lambda \cdot c) = \lambda \cdot (a \otimes c)$$

We can also find tensor products by defining the free product of two modules and the defining a relation which corresponds to the bilinear properties above.

Some properties can be stated for vector spaces V (without topology) see [76]:

Proposition A.4.22 *Let $y_1, .., y_n \in V$ linearly independent, then $(x_i \otimes y_i) = 0 \Longrightarrow x_i = 0$ $i = 1..n$.*

Let (x_i), (y_i) be bases for V_1 resp. V_2 then $x_i \otimes y_i$ bases for $V_1 \times V_2$.

The Kronecker product for matrices defined in Section A.3.7 is an example for a tensor product.

A.4.4.1 The "Outer" Tensor Product

We will get to know a couple of tensor products, one of them is

Definition A.4.18 *Let X, Y, Z be sets, $f : X \to Z$,$g : Y \to Z$ be functions. Then define the* **Kronecker product** $\otimes_{X \times Y} : X \times Y \to Z$ *by*

$$f \otimes_{X \times Y} g(x, y) = f(x) \cdot g(y)$$

We will often write $f \otimes g$ instead of $f \otimes_{X \times Y} g$, if there is no chance of misinterpretation. Although in most cases it should be apparent, which tensor product is meant, we have to give a (rather arbitrary) name to the different tensor products, so we call that the outer tensor product. It is easy to prove that this is a tensor product.

A.4.4.2 The "Inner" Tensor Product

Definition A.4.19 *Let $f \in \mathcal{H}_1$, $g \in \mathcal{H}_2$ then define the* **rank-one operator -** *or* **inner tensor product** *as a function from $\mathcal{H}_2$ to $\mathcal{H}_1$ by*

$$(f \otimes_{\mathcal{H}} \overline{g})(h) = \langle h, g \rangle f$$

We will often write $f \otimes g$ instead of $f \otimes_{\mathcal{H}} g$ if the meaning is clear. We call that the inner tensor product as a inner product is involved and also two elements of the same space are used.

For this operator we know

Lemma A.4.23 ([110] I.1. Lemma 1) *The tensor product of two elements $f \in \mathcal{H}_1$ and $g \in \mathcal{H}_2$ is in $\mathcal{B}(\mathcal{H}_2, \mathcal{H}_1)$ with $\|f \otimes \overline{g}\| = \|f\|_{\mathcal{H}_1} \cdot \|g\|_{\mathcal{H}_2}$. Its range is of dimension 1 or 0 (and it's therefore compact).*

Clearly these tensor products are exactly the rank one (and zero) operators on Hilbert spaces.

Furthermore we know and it's easy to prove

Lemma A.4.24 ([110] I.1. Lemma 2)

1. *This product is a tensor product (it's bilinear).*
2. $(f \otimes_{\mathcal{H}} \overline{g})^* = \left(g \otimes_{\mathcal{H}} \overline{f}\right)$
3. $(f \otimes_{\mathcal{H}} \overline{g})\,(f' \otimes_{\mathcal{H}} \overline{g}') = \langle f', g\rangle\,(f \otimes_{\mathcal{H}} \overline{g}')$
4. $A \circ (f \otimes_{\mathcal{H}} \overline{g}) = ((Af) \otimes_{\mathcal{H}} \overline{g})$
5. $(f \otimes_{\mathcal{H}} \overline{g}) \circ A = \left(f \otimes_{\mathcal{H}} \left(\overline{A^* g}\right)\right)$

We will see in A.4.5.4 a connection between these two tensor products on special function spaces.

It is easy to show:

Lemma A.4.25 *Let $f, g \in \mathcal{H}$. Then $f \otimes g$ is a projection if $\langle f, \overline{g}\rangle_{\mathcal{H}} = 1$.*

A.4.5 Compact Operators

A.4.5.1 Compact Operators In Banach Spaces

Definition A.4.20 *$T \in L(\mathfrak{B}_1, \mathfrak{B}_2)$ is called* **compact**, *if $\overline{T(B_1)}$ is compact with $B_1 = \{x \in \mathfrak{B}_1 \,|\, \|x\|_{\mathfrak{B}} \leq 1\}$. The set of all these functions will be denote by $\mathcal{K}(\mathfrak{B}_1, \mathfrak{B}_2)$.*

Again we use the notation $\mathcal{K}(\mathfrak{B}_1) = \mathcal{K}(\mathfrak{B}_1, \mathfrak{B}_1)$.

Theorem A.4.26 ([129] II.3.2)*$\mathcal{K}(\mathfrak{B}_1, \mathfrak{B}_2)$ is a closed subspace of $\mathcal{B}(\mathfrak{B}_1, \mathfrak{B}_2)$. $K(\mathfrak{B}_1, \mathfrak{B}_2)$ is therefore a Banach space.*

Let $\mathfrak{B}_3$ be also a Banach space. If $T \in \mathcal{B}(\mathfrak{B}_1, \mathfrak{B}_2)$ and $S \in \mathcal{B}(\mathfrak{B}_2, \mathfrak{B}_3)$ and either T or S compact, then $S \circ T$ is compact.

A function f is called to have *finite rank* , if $ran(f)$ is finite dimensional.

Corollary A.4.27 *Let $T \in \mathcal{B}(\mathfrak{B}_1, \mathfrak{B}_2)$. If there exist $T_n \in BL(\mathfrak{B}_1, \mathfrak{B}_2)$ with finite rank, such that $\|T_n - T\|_{Op} \to 0$ for $n \to \infty$, T is compact.*

Definition A.4.21 *Let* $T \in L(\mathfrak{B}_1)$*, then*

1. *the set*
$$\sigma(T) = \{\lambda \in \mathbb{K} \,|(\lambda \cdot Id - T) \text{ is not invertible } \}$$
is called the **spectrum** *of* T*.*

2. *A number* $\lambda \in \sigma(T)$ *is called* **eigenvalue** *and* $x \in \mathfrak{B}_1$ *is called* **eigenvector***, if*
$$Tx = \lambda \cdot x$$

Theorem A.4.28 ([129] VI 2.5) *Let* $T \in \mathcal{K}(\mathfrak{B}_1)$*, then*

1. *If* $\mathfrak{B}_1$ *has infinite dimension,* $0 \in \sigma(T)$

2. *Every* $\lambda \in \sigma(T)\backslash\{0\}$ *is an eigenvalue.*

Therefore for infinite-dimensional space, the identity Id can not be compact, as it is clearly invertible. This means that on infinite-dimensional space no compact operator can be invertible.

Corollary A.4.29 [110] *Two compact operators* $S, T \in \mathcal{B}(\mathfrak{B}_1)$ *commute if and only if there is a basis for* $\mathfrak{B}_1$ *consisting of eigenvectors of both.*

A.4.5.2 Compact Operators On Hilbert Spaces

Different to the situation in Banach spaces, in Hilbert spaces the compact operators are exactly those, that are limit of finite ranks operators:

Corollary A.4.30 [26] *Let* $T \in \mathcal{B}(\mathcal{H}_1, \mathcal{H}_2)$*. If and only there exist* $T_n \in BL(\mathcal{H}_1, \mathcal{H}_2)$ *with finite rank, such that* $\|T_n - T\|_{Op} \to 0$ *for* $n \to \infty$*,* T *is compact.*

The well-known *spectral theorem* for normal compact operator says, in the terminology used in the main part of this work, that such operators can be represented as multiplier of an ONB. Remember that we use $\mathbb{K} = \mathbb{C}$.

Theorem A.4.31 ([129] Vi. 3.2)*Let* $T \in \mathcal{K}(\mathcal{H}_1)$ *be normal. Then there is a ONB* $(e_k) \subseteq \mathcal{H}_!$ *and a sequence* $(\lambda_k) \subseteq \mathbb{C}\backslash\{0\}$ *such that*
$$Tx = \sum_k \lambda_k \langle x, e_k \rangle \cdot e_k$$
where the e_k *are eigenvectors of* T *for the eigenvalues* λ_k*.* $\{\lambda_k\} = \sigma(T)\backslash\{0\}$*.*

This can be extended to the class of all compact operators:

Theorem A.4.32 *Let $T \in \mathcal{K}(\mathcal{H}_1, \mathcal{H}_2)$. Then there exists an orthonormal sequences $(e_i) \subseteq \mathcal{H}_1$ and $(f_i) \subseteq \mathcal{H}_2$ and numbers $s_1 \geq s_2 \geq \ldots \geq 0$ with $s_i \to 0$ for $i \to \infty$ such that for all $x \in \mathcal{H}_1$*

$$Tx = \sum_i s_i \langle x, e_i \rangle f_i$$

The numbers s_i^2 are the eigenvalues of T^*T.

The numbers s_k are called *singular values*. The finite matrix version of this theorem is therefore called the *singular value decomposition*.

A spectral theorem is also true for possibly non-compact self-adjoint operators.

Theorem A.4.33 ([129] VII.1.21) *Every self-adjoint operator on a Hilbert-space is unitarily equivalent to a multiplication operator. More precisely: For every $T \in \mathcal{B}(\mathcal{H}_1)$ there exists a measure space (Ω, μ), a bounded, measurable function $\lambda : \Omega \to \mathbb{R}$ and a unitary operator $U : \mathcal{H}_1 \to L^2(\Omega, \mu)$ such that*

$$(UTU^*)\, \varphi = \lambda \cdot \varphi$$

nearly $\mu-$everywhere for all $\varphi \in L^2(\Omega, \mu)$.

This can be used to define a functional calculus on the operators in $\mathcal{B}(\mathcal{H}_1)$ by

$$(Uf(T)U^*)\, \varphi = f(\lambda) \cdot \varphi$$

for bounded measurable functions f, see also [26].

Definition A.4.22 *For an operator $T : \mathfrak{B}_1 \to \mathfrak{B}_2$ let $[T] = (TT^*)^{\frac{1}{2}}$.*

Sometimes this operator is denoted by $|T|$, we have decided to stick to the notation introduced by [110].

A.4.5.3 Trace class operators

For more detail on this class of compact operators refer to [110] or [129].

Definition A.4.23 *Let $\mathfrak{B}_1, \mathfrak{B}_2$ be Banach spaces. An operator $T \in \mathcal{L}(\mathfrak{B}_1, \mathfrak{B}_2)$ is called* **trace class** *(or nuclear) if there exist sequences $(x_n') \in \mathfrak{B}_1'$ and $(y_n) \in \mathfrak{B}_2$ with*

$$\sum_{n=1}^{\infty} \|x_n'\|_{\mathfrak{B}_1'} \|y_n\|_{\mathfrak{B}_2} < \infty$$

such that for all $x \in \mathfrak{B}_1$

$$Tx = \sum_{n=1}^{\infty} x'_n(x) y_n \quad \forall\, x \in X$$

Let $\mathcal{N}(\mathfrak{B}_1, \mathfrak{B}_2)$ be the set of all trace class operators.

For Hilbert spaces this means due to the Riesz representation theorem, that T is a trace class operator, if there exist sequences $(x_n) \in \mathcal{H}_1$ and $(y_n) \in \mathcal{H}_2$ with

$$\sum_{n=1}^{\infty} \|x_n\|_{\mathcal{H}_1} \|y_n\|_{\mathcal{H}_2} < \infty$$

such that

$$Tx = \sum_{n=1}^{\infty} \langle x, y_n \rangle\, x_n = \sum_{n=1}^{\infty} x_n \otimes y_n \quad \forall\, x \in \mathcal{H}_1$$

The class $\mathcal{N}(\mathfrak{B}_1, \mathfrak{B}_2)$ is a Banach space with the following norm

Definition A.4.24 *For $T \in \mathcal{N}(\mathfrak{B}_1, \mathfrak{B}_2)$ like in the definition above let*

$$\|T\|_{trace} = \inf \sum_{n=1}^{\infty} \|x_n\|_{\mathfrak{B}'_1} \|y_n\|_{\mathfrak{B}_2}$$

be the **nuclear** *or* **trace class** *norm.*

The trace class operators are an operator ideal:

Proposition A.4.34 ([129] VI.5.4) *Let $N \in \mathcal{N}(\mathfrak{B}_1, \mathfrak{B}_2)$, $S \in \mathcal{B}(\mathfrak{B}_2, \mathfrak{B}_3)$ and $T \in \mathcal{B}(\mathfrak{B}_0, \mathfrak{B}_1)$ then $S \circ N \circ T \in \mathcal{N}(\mathfrak{B}_0, \mathfrak{B}_3)$ with*

$$\|SNT\|_{trace} \leq \|S\|_{Op} \|N\|_{trace} \|T\|_{Op}.$$

For Hilbert-space it is known, cf. [110], that trace-class operators are compact. We can find equivalence conditions, when a compact operator is trace-class:

Corollary A.4.35 *Let $N \in \mathcal{K}(\mathcal{H}_1)$.*

$$N \in \mathcal{N}(\mathcal{H}_1) \Longleftrightarrow \sum s_i < \infty,$$

where the s_i are the singular values for N.

Is can be shown that the sum of the singular values is equal to the following sum:

Definition A.4.25 *For $N \in \mathcal{K}(\mathcal{H}_1)$ then let*

$$tr(N) = \sum_i \langle Ne_i, e_i \rangle$$

be the **trace** *of N.*

The trace class operators are exactly the class of operators for which the trace is defined.

This can be written in a different form :

Lemma A.4.36 *Let $T \in N(\mathcal{H}_1, \mathcal{H}_2)$. Then there exist sequences $(y_n), (x_n)$ such that*

$$tr(T) = \sum_n \langle y_n, x_n \rangle$$

This sequences coincide with the sequences from the remark following definition A.4.23.

Proof: Following the remark after definition A.4.23 there are sequences such that

$$Tx = \sum_{n=1}^{\infty} \langle x, x_n \rangle \, y_n \quad \forall \, x \in X$$

So let (e_k) be an ONB

$$tr(T) = \sum_i \langle Te_i, e_i \rangle = \sum_i \left\langle \sum_{n=1}^{\infty} \langle e_i, x_n \rangle \, y_n, e_i \right\rangle =$$

$$= \sum_i \sum_{n=1}^{\infty} \langle e_i, x_n \rangle \, \langle y_n, e_i \rangle = \sum_{n=1}^{\infty} \langle y_n, x_n \rangle$$

□

So it is clear that the tensor product $f \otimes g$ is a trace class operator as

$$tr(f \otimes g) = \sum_i \langle (f \otimes g) \, e_i, e_i \rangle = \sum_i \left\langle e_i, \overline{f} \right\rangle \langle g, e_i \rangle = \langle f, g \rangle < \infty.$$

The trace-class norm can also be calculated by using the following result:

Lemma A.4.37 *[110] Let $T \in N(\mathcal{H}_1, \mathcal{H}_2)$ and let (e_i) be any ONB of $\mathcal{H}_1$, then*

$$\|T\|_{trace} = \sum_i \langle [T] \, e_i, e_i \rangle$$

Using the matrix representation of an operator with ONBs, Section A.4.3.4, we get

$$tr(T) = tr(\mathcal{M}(T))$$

and

$$tr(M) = tr(\mathcal{O}(M))$$

A.4.5.4 Hilbert Schmidt operators

Definition A.4.26 *Let $\mathcal{H}_1, \mathcal{H}_2$ be Hilbert spaces. A bounded operator $T \in \mathcal{B}(\mathcal{H}_1, \mathcal{H}_2)$ is called* **Hilbert Schmidt** *($\mathcal{HS}$) operator if there exists an ONB $\{e_n\} \subseteq \mathcal{H}_1$ such that*

$$\|T\|_{\mathcal{HS}} := \sqrt{\sum_{n=1}^{\infty} \|Te_n\|_{\mathcal{H}}^2} < \infty$$

Let $\mathcal{HS}(\mathcal{H}_1, \mathcal{H}_2)$ be the space of Hilbert Schmidt operators from $\mathcal{H}_1$ to $\mathcal{H}_2$.

This definition is independent of the choice of the ONB. The class of Hilbert-Schmidt operators is a subclass of the compact operators, cf. [110].

Lemma A.4.38 [110] II.Lemma 2 & 3

- $\|T\|_{Op} \leq \|T\|_{\mathcal{HS}}$
- $T \in \mathcal{HS} \iff T^* \in \mathcal{HS}$ *and* $\|T\|_{\mathcal{HS}} = \|T^*\|_{\mathcal{HS}}$.
- $T \in \mathcal{HS}$ *and* $A \in \mathcal{B}$ *then* TA *and* $AT \in \mathcal{HS}$. $\|AT\|_{\mathcal{HS}} \leq \|A\|_{Op} \|T\|_{\mathcal{HS}}$ *and* $\|TA\|_{\mathcal{HS}} \leq \|A\|_{Op} \|T\|_{\mathcal{HS}}$.
- *For all* $f, g \in \mathcal{H}$ $f \otimes g \in \mathcal{HS}$ *and* $\|f \otimes g\|_{\mathcal{HS}} = \|f\|_{\mathcal{H}} \|g\|_{\mathcal{H}}$.
- $T \in \mathcal{HS} \iff [T] \in \mathcal{HS}$. $\|T\|_{\mathcal{HS}} = \|[T]\|_{\mathcal{HS}}$.

Definition A.4.27 *For $T, S \in \mathcal{HS}$ and (e_k) an ONB in $\mathcal{H}$. Then let*

$$\langle T, S \rangle_{\mathcal{HS}} = \sum_k \langle Te_k, Se_k \rangle_{\mathcal{H}}$$

This definition is again independent on the chosen ONB.

Lemma A.4.39 [110] II Lemma 5 & 6

- $\langle ., . \rangle_{\mathcal{HS}}$ *is an inner product.*

- *$\mathcal{HS}$ is a Hilbert space with this inner product.*
- $\langle S^*, T^*\rangle_{\mathcal{HS}} = \overline{\langle S, T\rangle}_{\mathcal{HS}}$
- $\langle XA, B\rangle_{\mathcal{HS}} = \langle A, X^* B\rangle_{\mathcal{HS}}$
- $\langle AX, B\rangle_{\mathcal{HS}} = \langle A, BX^*\rangle_{\mathcal{HS}}$
- $\left\langle f \otimes \overline{g}, h \otimes \overline{l}\right\rangle_{\mathcal{HS}} = \langle f, h\rangle_{\mathcal{H}} \langle l, g\rangle_{\mathcal{H}}$

Corollary A.4.40 *Let $A \in \mathcal{HS}$, then*

$$\langle A, f \otimes \overline{g}\rangle_{\mathcal{HS}} = \langle Ag, f\rangle_{\mathcal{H}}$$

Proof:

$$\langle f \otimes \overline{g}, A\rangle_{\mathcal{HS}} = \sum_k \langle (f \otimes \overline{g})\, e_k, Ae_k\rangle_{\mathcal{H}} = \sum_k \langle e_k, g\rangle \langle f, Ae_k\rangle =$$

$$= \sum_k \langle e_k, g\rangle \langle A^* f, e_k\rangle = \langle A^* f, g\rangle$$

$$\Longrightarrow \langle A, f \otimes \overline{g}\rangle_{\mathcal{HS}} = \overline{\langle f \otimes \overline{g}, A\rangle_{\mathcal{HS}}} = \overline{\langle A^* f, g\rangle} = \langle g, A^* f\rangle = \langle Ag, f\rangle$$

□

Theorem A.4.41 *Let T be an operator from $L^2\left(\mathbb{R}^d\right)$ to $L^2\left(\mathbb{R}^d\right)$. T is an integral operator with kernel in $L^2\left(\mathbb{R}^d\right)$ if and only if it belongs to $\mathcal{HS}(L^2\left(\mathbb{R}^d\right))$.*

$$\|T\|_{\mathcal{HS}} = \|\kappa(T)\|_{L^2\left(\mathbb{R}^d\right)}$$

For $d = 1$ this result can be found in [110], for higher dimension for example in [43].

This correspondence is even unitary as $\langle S, T\rangle_{\mathcal{HS}} = \langle \kappa(S), \kappa(T)\rangle_{L^2\left(\mathbb{R}^{2d}\right)}$, cf. [134].

Lemma A.4.42

$$\kappa\left(g \otimes_{L^2\left(\mathbb{R}^d\right) \times L^2\left(\mathbb{R}^d\right)} \overline{h}\right) = g \otimes_{L^2\left(\mathbb{R}^d\right)} h$$

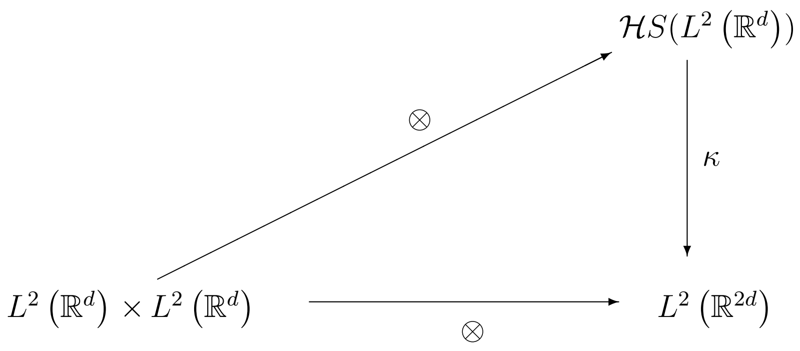

Figure A.2: Tensor product: Kernel of $g \otimes \overline{f}$.

Proof:

$$\left(\left(g \otimes \overline{h}\right) f\right)(x) = \langle f, h \rangle_{L^2\left(\mathbb{R}^d\right)} g(x) = \left(\int_{\mathbb{R}^d} f(y) \cdot h(y) dy\right) \cdot g(x) =$$

$$= \int_{\mathbb{R}^d} \underbrace{(f(y) \cdot g(x))}_{k(x,y)} \cdot h(y) dy$$

$\square$

This property is depicted in Figure A.2.

So overall we get for the rank one tensor product operators:

Corollary A.4.43

$$\langle g \otimes h, g' \otimes h' \rangle_{\mathcal{H}S} = \langle g \otimes h, g' \otimes h' \rangle_{L^2\left(\mathbb{R}^{2d}\right)} = \langle g, g' \rangle_{L^2\left(\mathbb{R}^d\right)} \langle h, h' \rangle_{L^2\left(\mathbb{R}^d\right)}$$

Using the matrix representation of an operator in Section A.4.3.4, we get

$$\langle T, S \rangle_{\mathcal{H}S} = \langle \mathcal{M}(T), \mathcal{M}(S) \rangle_{fro}$$

and

$$\langle M, N \rangle_{fro} = \langle \mathcal{O}(M), \mathcal{O}(N) \rangle_{\mathcal{H}S}$$

A.4.6 Pseudoinverse Of An Operator

For more detail on this topic let us refer to [21].

Lemma A.4.44 *Let $A \in \mathcal{B}(\mathcal{H}_1, \mathcal{H}_2)$ with closed range. Then there exists a bounded operator $A^\dagger : \mathcal{H}_2 \to \mathcal{H}_1$ for which*

$$AA^\dagger f = f, \forall f \in ran(A)$$

Definition A.4.28 *This $A^\dagger$ is called the (Moore-Penrose)* **pseudoinverse** *of A.*

If A is invertible, $A^\dagger = A^{-1}$. If U, V are invertible, then $(UAV)^\dagger = V^{-1}A^\dagger U^{-1}$. But in general $(A \circ B)^\dagger \neq B^\dagger \circ A^\dagger$. (Even in finite dimensional spaces.)

Proposition A.4.45 ([23] A.7.2) *Let $T \in \mathcal{B}(\mathcal{H}_1, \mathcal{H}_2)$ with closed range. Then*

1. *The orthogonal projection of $\mathcal{H}_1$ onto $ran(T)$ is $TT^\dagger$*
2. *The orthogonal projection of $\mathcal{H}_2$ onto $ran(T^\dagger)$ is $T^\dagger T$*
3. *T^* has closed range and $(T^*)^\dagger = \left(T^\dagger\right)^*$.*
4. *On $ran(T)$ the operator $T^\dagger$ is given by*

$$T^\dagger = T^* \left(TT^*\right)^{-1}$$

Proposition A.4.46 ([23] Theorem A.7.3) *Let $U : \mathcal{K} \to \mathcal{H}$ be a bounded surjective operator. Given $y \in \mathcal{H}$ the equation $Ux = y$ has a unique solution of minimal norm, namely $x = U^\dagger y$.*

Proposition A.4.47 ([21] Theorem 2.2)*Let $T : \mathcal{H}_1 \to \mathcal{H}_2$ and $U : \mathcal{H}_2 \to \mathcal{H}_1$ bounded linear operators with closed ranges. Then*

$$(U \circ V)^\dagger = V^\dagger U^\dagger \tag{A.2}$$

if and only if

1. *$U \circ V$ has closed range*
2. *$ran(U^*)$ is invariant under VV^* and*
3. *$ran(U^*) \cap ker(V^*)$ is invariant under U^*U.*

Even for matrices the simple equality A.2 is not true in general. But as a corollary from A.4.47 we get

Corollary A.4.48 ([21] Corollary 2.3)*Let $T : \mathcal{H}_1 \to \mathcal{H}_2$ be a linear bounded operator with closed range. Then*

$$(U^* \circ U)^\dagger = U^\dagger U^{*\dagger}$$

A.4.7 Fourier Transform

For details on this topic we refer for example to [75] or [63].

Definition A.4.29 *Let $f \in L^1(\mathbb{R}^d)$, then we define the* **Fourier transformation** *as*

$$\mathcal{F}(f)(\omega) = \hat{f}(\omega) = \int_{\mathbb{R}^d} f(t) e^{-2\pi i \omega t} dt$$

For $f \in L^1(\mathbb{R}^d)$ we have $\hat{f}$ is uniformly continuous and vanishes at infinity.

Theorem A.4.49 (Plancherel) *Let $f \in L^1 \cap L^2(\mathbb{R}^d)$ then*

$$\|f\|_2 = \left\|\hat{f}\right\|_2$$

Therefore $\mathcal{F}$ extends to a unitary operator on $L^2(\mathbb{R}^d)$ and satisfies Parseval's formula *:*

$$\langle f, g \rangle = \left\langle \hat{f}, \hat{g} \right\rangle$$

for all $f, g \in L^2(\mathbb{R}^d)$.

Theorem A.4.50 (Poisson) *Suppose that for some $\epsilon > 0$ and $C > 0$ we have $|f(x)| \leq C \cdot (1 + |x|)^{-d-\epsilon}$ and $\left|\hat{f}(\omega)\right| \leq C \cdot (1 + |\omega|)^{-d-\epsilon}$. Then*

$$\sum_{n \in \mathbb{Z}^d} f(x+n) = \sum_{n \in \mathbb{Z}^d} \hat{f}(n) e^{2\pi i n \cdot x}$$

pointwise for all $x \in \mathbb{R}^d$ and both sums converge absolutely.

A.4.7.1 Convolution

Definition A.4.30 *For $f, g \in L^1(\mathbb{R}^d)$ the* **convolution** *is defined as*

$$(f * g)(x) = \int_{\mathbb{R}^d} f(y) g(x-y) dy$$

It satisfies

$$\|f * g\|_1 \leq \|f\|_1 \|g\|_1 .$$

This definition is equivalent to

$$(f * g)(x) = \langle f, T_x g^* \rangle_{L^2(\mathbb{R}^d)}$$

where both sides are defined, and $g^*(x) = \overline{g(-x)}$ is the *involution* of g. This can be used to define the convolution in other spaces (including spaces of measures or distributions).

Again it can be shown:

Proposition A.4.51 *Let $f, g \in L^1(\mathbb{R}^d)$. Then*

$$\widehat{f * g} = \hat{f} \cdot \hat{g}$$

A.4.7.2 Derivatives

Let α be a so-called *multi-index*, i.e $\alpha \in \mathbb{N}_0^d$. Then we write $|\alpha| = \sum_{j=1}^{d} \alpha_j$. For $\omega \in \mathbb{R}^d$ let $\omega^\alpha = \prod_{j=1}^{d} \omega_j^{\alpha_j}$.

Definition A.4.31 *1.* $D^\alpha = \prod_{j=1}^{d} \frac{\delta^{\alpha_j}}{\delta x_j^{\alpha_j}}$ *the* **partial derivative operator***.*

2. $X^\alpha f(x) = x^\alpha f(x)$ *the* **multiplication operator***.*

Then

1. $\widehat{(D^\alpha f)}(\omega) = (2\pi i \omega)^\alpha \hat{f}(\omega)$
2. $(\widehat{(-2\pi i x)^\alpha f}) = D^\alpha \hat{f}(\omega)$

if both sides of this equations are well defined. Written in operator notation this is

1. $\mathcal{F} D^\alpha = (2\pi i \omega)^{|\alpha|} X^\alpha \mathcal{F}$
2. $\mathcal{F} X^\alpha = \left(\frac{i}{2\pi}\right)^{|\alpha|} D^\alpha \mathcal{F}$

A.5 Special Spaces

A.5.1 Spaces of sequences

Sequences in the field $\mathbb{K}$ can be seen as functions from the natural numbers $\mathbb{N}$ into $\mathbb{K}$. Let us denote that class by $\mathbb{K}^\mathbb{N}$. Let us use the norms from Section A.3.3 here, generalized in a natural way to this infinite-dimensional case. For detail refer for example to [129].

We define special subclasses:

Definition A.5.1 *Then we denote by*

1. $c_c = \left\{(c_n) \subseteq \mathbb{K}^\mathbb{N} : c_n \neq 0 \text{ for only finitely many } n\right\}$

2. $c_0 = \left\{ (c_n) \subseteq \mathbb{K}^{\mathbb{N}} : \lim_{n\to\infty} c_n = 0 \right\}$ *and*

3. $c = \left\{ (c_n) \subseteq \mathbb{K}^{\mathbb{N}} : \exists \mathcal{C} : \mathcal{C} = \lim_{n\to\infty} c_n \right\}$

All these sets are vector spaces with norm $\|.\|_\infty$. c_0 and c are Banach spaces, c_c is not closed.

Definition A.5.2 *Let* $1 \le p \le \infty$. *Then let*

$$l^p = \left\{ c \subseteq \mathbb{K}^{\mathbb{N}} : \|c\|_p < \infty \right\}$$

With the respective norms all these spaces are Banach spaces.

Following Section A.3.3 and using limit arguments we get for $p > 2$

$$\|(c_k)\|_\infty \le \|(c_k)\|_p \le \|(c_k)\|_2 \le \|(c_k)\|_1$$

where some values might be ∞.

So these spaces are connected to each other:

$$c_c \subseteq l^1 \subseteq l^2 \subseteq \ldots \subseteq l^p \subseteq \ldots \subseteq c_0 \subseteq c \subseteq l^\infty$$

These definitions can be extended to other countable index sets, when we will use the notation $c_c(I), c_0(I), c(I)$ and $l^p(I)$.

A.5.1.1 Hölder's Inequality

Set formally "$\frac{1}{\infty}$" $= 0$. A well known theorem, see e.g. [129] I.1.4, is

Theorem A.5.1 (Hölder's Inequality) *Let* $1 \le p \le \infty$. *Let* $q \ge 1$ *so, that* $\frac{1}{p} + \frac{1}{q} = 1$. *Then for* $x \in l^p$, $y \in l^q$

$$\|x \cdot y\|_1 \le \|x\|_p \cdot \|x\|_q$$

We know that

$$c_c \subseteq l^1 \subseteq l^2 \subseteq \ldots \subseteq c_0 \subseteq c \subseteq l^\infty$$

and so

$$l^p \cdot l^1 \subseteq l^1 \; for \; 1 \le p \le \infty.$$

For all $p \ge 1$ the product $l^\infty \cdot l^p \subseteq l^p$, therefore $l^q \cdot l^p \subseteq l^p$ for all $p, q \ge 1$.

A.5.2 Spaces of functions

Definition A.5.3 $C(\mathbb{R}^d) = \{f : \mathbb{R}^d \to \mathbb{C} \,|\, f \textit{ continuous}\}$

For integration we use the Lebesque measure

Definition A.5.4 *Let*

1. $L^p(\mathbb{R}^d) = \left\{ f : \mathbb{R}^d \to \mathbb{C}, \textit{measurable} \;\middle|\; \int\limits_{\mathbb{R}^d} |f(x)|^p \, dx < \infty \right\}$

2. $L^\infty(\mathbb{R}^d) = \left\{ f : \mathbb{R}^d \to \mathbb{C}, \textit{measurable} \;\middle|\; \textit{ess} \sup\limits_{x\in\mathbb{R}^d} \{|f(x)|\} < \infty \right\}$

These spaces are not considered as sets of functions, but of classes of functions, which are identical nearly everywhere, i.e. $f \equiv_{L^p} g \iff \int\limits_{\mathbb{R}^d} |f(x) - g(x)|^p \, dx = 0$.

Definition A.5.5 1. *For any function space $F(\mathbb{R}^d)$ let $F(\mathbb{R}^d)^b$ be $F(\mathbb{R}^d)^b = F(\mathbb{R}^d) \cap L^\infty(\mathbb{R}^d)$, the* bounded *$F(\mathbb{R}^d)$ functions, using the induced norm.*

2. *For any function space $F(\mathbb{R}^d)$ let $F(\mathbb{R}^d)_c$ be*

$$F(\mathbb{R}^d)_c = \left\{ f \in F(\mathbb{R}^d) \,\middle|\, \exists K \subset\subset \mathbb{R}^d : supp(f) \subseteq K \right\}$$

the functions of $F(\mathbb{R}^d)$ with compact support*, using the induced norm.*

Analogous to the sequence spaces, if $f \in L^\infty$ and $g \in L^p$ then $f \cdot g$ and $g \cdot f \in L^p$.

Example A.5.1 :

These examples can by investigated by using a computer algebra system like e.g. MAPLE [88].

1. $f(x) = sinc(x)$ is not in $L^1(\mathbb{R})$ but in $L^2(\mathbb{R})$.

2. $f(x) = \begin{cases} \sqrt{\frac{1}{x}} & |x| \leq 9 \\ 0 & \textit{otherwise} \end{cases}$ is not in $L^p(\mathbb{R})$ for $p > 1$ or in $L^\infty(\mathbb{R})$ but in $L^1(\mathbb{R}^d)$.

3. $f(x) = \begin{cases} \sqrt[q]{\frac{1}{x}} & |x| \leq 9 \\ 0 & otherwise \end{cases}$ is certainly not in $L^p(\mathbb{R})$ for $p > q$ or in $L^\infty(\mathbb{R})$ but in $L^p(\mathbb{R})d$ for $p <= q$.

   Note, that the essential sup here is ∞.

4. $1 \in L^\infty(\mathbb{R})$ but clearly not in $L^p(\mathbb{R})$.

5. $f(x) = \begin{cases} x & |x| \leq 1 \\ \frac{1}{x} & otherwise \end{cases}$ is not in $L^1(\mathbb{R})$ but is in $C_0(\mathbb{R})$.

So all these functions spaces are different from each other.

Lemma A.5.2 *1. $L^p(\mathbb{R}^d)$ is a Banach space for all $1 \leq p \leq \infty$.*

2. $C_c(\mathbb{R}^d) \subseteq C_0(\mathbb{R}^d) \subseteq C_b(\mathbb{R}^d) \subseteq L^\infty(\mathbb{R}^d)$.

3. $C_b(\mathbb{R}^d)$ and $C_0(\mathbb{R}^d)$ are closed for $\|.\|_\infty$ and therefore Banach spaces.

4. $C_c(\mathbb{R}^d) \subseteq C_0(\mathbb{R}^d)$ dense for $\|.\|_\infty$

5. $C_c(\mathbb{R}^d) \subseteq L^p(\mathbb{R}^d)$ dense for $\|.\|_p$

Like for sequence space also for function spaces a Hölder inequality holds:

Theorem A.5.3 *1. $L^p(\mathbb{R}^d) \cdot L^q(\mathbb{R}^d) \subseteq L^1(\mathbb{R}^d)$ for $\frac{1}{p} + \frac{1}{q} = 1$.*

2. Especially $L^2(\mathbb{R}^d) \cdot L^2(\mathbb{R}^d) \subseteq L^1(\mathbb{R}^d)$

For function spaces we can also defined "local versions" of them

Definition A.5.6

$$F_{loc}(\mathbb{R}^d) = \left\{ f : \mathbb{R}^d \to \mathbb{C} \,\middle|\, \forall K \subset\subset \mathbb{R}^d \; \exists g \in F(\mathbb{R}^d) : f|_K = g|_K \right\}$$

So e.g.

$$L^1_{loc} = \left\{ f \,\middle|\, \forall K \subset\subset \mathbb{R}^d : f \in L^1(K) \right\} = \left\{ f \,\middle|\, \forall K \subset\subset \mathbb{R}^d : \int_K |f(x)|\, dx < \infty \right\}$$

This function space can be equipped with the seminorms $\|f\|_K = \int_K |f(x)|\, dx$.

A.5.2.1 Mixed Norm Spaces

A *weight* function is a non-negative, locally integrable function on $\mathbb{R}^{2d}$.

Definition A.5.7 *1. A weight function v on $\mathbb{R}^{2d}$ is called* **submultiplicative**, *if*

$$v(z_1 + z_2) \leq v(z_1) \cdot v(z_2) \text{ for all } z_1, z_2 \in \mathbb{R}^{2d}$$

*2. A weight function m on $\mathbb{R}^{2d}$ is called v-***moderate**, *if there exists a $C > 0$ such that*

$$m(z_1 + z_2) \leq C \cdot v(z_1) \cdot m(z_2)$$

Definition A.5.8 *Let m be a weight function on $\mathbb{R}^{2d}$ and let $1 \leq q, p < \infty$. Then the* **weighted mixed-norm space** $L_m^{p,q}(\mathbb{R}^{2d})$ *consists of all (Lebesque) measurable functions on $\mathbb{R}^{2d}$, such that*

$$\|F\|_{L_m^{p,q}(\mathbb{R}^{2d})} = \left(\int\limits_{\mathbb{R}^d} \left(\int\limits_{\mathbb{R}^d} |F(x,\omega)|^p \, m(x,\omega)^p dx \right)^{\frac{q}{p}} d\omega \right)^{\frac{1}{q}} < \infty$$

This definition can be extended in a natural way to $p, q = \infty$ by using the *ess* sup. This function class is a Banach space, translation invariant and a Hölder inequality is valid, if the weights are v-moderate, with v a submultiplicative weight, cf. [63].

Compare this to Definition A.3.12.

A.6 Distributions

We will not use this concept extensively, so only the basic ideas are given. For details on this topic we refer for example to [71].

A.6.1 Schwartz Class

Definition A.6.1 *The* Schwartz class $\mathcal{S}$ *consists of all C^∞-functions on $\mathbb{R}^d$ for which*

$$\sup_{x \in \mathbb{R}^d} \left| D^\alpha X^\beta f(x) \right| < \infty.$$

This is a Fréchet space with the semi-norms $\|\varphi\|_{\alpha,\beta} = \sup\limits_{x \in \mathbb{R}^d} \left\{ \left| D^\alpha X^\beta f(x) \right| \right\}$. The Fourier transformation is a continuous bijection $\mathcal{F} : \mathcal{S} \to \mathcal{S}$.

A.6.2 Tempered Distributions

Distributions are a generalization of differentiable functions. This concept was used by physicists long before the mathematical theory was developed. The "most famous" distribution is probably the *Dirac function* : Let $\delta(x) = 0$ for $x \neq 0$ and $\int_{\mathbb{R}} \delta(x) = 1$. The problem with this definition is that there is no Lebesque integrable functions that fulfills this definition. But the "inner product" of this "function" with a function $\in \mathcal{S}$ is well defined by $\langle \delta, f \rangle = \int \delta(x) f(x) dx = f(0)$. It also clear that for functions g in $L^p(\mathbb{R}^d)$ this inner product $\langle g, f \rangle$ is well-defined. So the next definition can be seen as generalization of functions:

Definition A.6.2 *The elements of the dual space $\mathcal{S}'$ are called* **tempered distributions**.

Motivated by the Riezs representation theorem we will use the notation $\langle \varphi, f \rangle = \varphi(\overline{f})$ respectively $\langle f, \varphi \rangle = \varphi(f)$ for $\varphi \in \mathcal{S}'$ and $f \in \mathcal{S}$. Clearly this is no inner product, $\mathcal{S}$ is not even a Hilbert space. But this notation is very helpful for seeing how distributions can be seen as generalization of functions respectively properties of them. The duality of the inner product in $L^2\left(\mathbb{R}^d\right)$ and this notation is useful for many properties, for example:

Proposition A.6.1 *Let $\varphi_k \in \mathcal{S}'$, if there exists an u such that for all $f \in \mathcal{S}$*

$$\lim_{k \to \infty} \langle f, \varphi_k \rangle = \langle f, u \rangle$$

then $u \in \mathcal{S}'$.

In this case we say $u = \lim_{k \to \infty} \varphi_k$ in $\mathcal{S}'$.

Often a result for functions can be used as the basic idea for a definition for distributions, for example:

1. For $f, g \in L^2(\mathbb{R})$, both differentiable, we know that

$$\int f'g = f \cdot g|_{-\infty}^{\infty} - \int fg'$$

   as $f(x), g(x) \to 0$ for $x \to \pm\infty$, we get

$$\langle f', g \rangle = -\langle f, g' \rangle$$

   and this can be used for the definition of the derivation of a distribution.

2. For the Fourier transformation the Paresval's formula is used as definition of the Fourier transformation for distributions:

$$\left\langle \hat{f}, \hat{\varphi} \right\rangle := \langle f, \varphi \rangle$$

All functions in $L^p(\mathbb{R}^d)$ can be seen a distribution via $\langle f, \varphi \rangle = \int\limits_{\mathbb{R}^d} f(x)\overline{\varphi}(x)dx$ for $f \in \mathcal{S}'$, $\varphi \in L^p(\mathbb{R}^d)$. All bounded Radon measures μ are also included in $\mathcal{S}'$ with $\langle f, \mu \rangle = \int\limits_{\mathbb{R}^d} f(x)d\mu(x)$. Plus we get a well-defined form for the Dirac-distribution :

Definition A.6.3 *Let $\delta_{x_0} \in \mathcal{S}'$ be the tempered distribution for which*

$$\langle f, \delta_{x_0} \rangle = f(x_0)$$

for all $f \in \mathcal{S}$. This is called the **Dirac-distribution**.

The periodized version of it

$$Ш_M = \sum_{k \in \mathbb{Z}^d} \delta_{k \cdot M}$$

is called the **Shah-Distribution**,

The famous Schwarz kernel theorem states that every operator from $\mathcal{S}$ to $\mathcal{S}'$ corresponds to a kernel using the above inner product notation:

Theorem A.6.2 *Let $A : \mathcal{S} \to \mathcal{S}'$ be an operator, then there exists a $k \in \mathcal{S}'$ such that*

$$\langle Af, g \rangle = \langle k, g \otimes f \rangle$$

This theorem can for example be found in [71]. A similar one can be formulated for modulation spaces, see [63].

As a last remark let us refer to [43], where S_0 is used as test functions for another class of distributions S_0'. Apart from the connection to time-frequency analysis a big advantage of this approach is that S_0 is a Banach space.

Appendix B

MATLAB codes

B.1 Frame Multiplier

B.1.1 Basic Algorithm

```
% Best Approximation of a matrix by a frame multiplier
% function [TA uppsym] = ApprFramMult(T,D,Ds)
%
% A matrix is best approximated (in the Hilbert-Schmidt sense) by a
% frame multiplier. The elements of the frames are given in the synthesis
% matrices D and Ds columnswise.
%
% inputs : T ......... the matrix (m x n)
%          D ......... the elements of the analysis frame (columnwise)
%                      (n x K)
%          Ds ........ the elements of the synthesis frame (columnwise)
%                      (m x K)
%
% output: TA ........ the best apporximation of the matrix T with frame
%                     multipliers using the frames in D and Ds
%
% usage:  TA=ApprFramMult(T,D)
%
% test:   T = eye(2,2)
%         D = [0 1/sqrt(2) -1/sqrt(2); 1 -1/sqrt(2) -1/sqrt(2)]
%         [TA coeff] = ApprFramMult(T,D);
%         See testapprfarmmult.m and testapprGabmultKap1.m
%
% date: 10/03/2005 - 03/04/2005
%
% notes : D and Ds are just the synthesis operators of the respective frames.
%        To be able to define frame multipliers they have to have the same
%        number of elements.
%
% Author:  XXL .. Peter Balazs, contact: a8927259@unet.univie.ac.at
%
% Literature : [Xxl] P. Balazs; Irregular And Regular Gabor frame multipliers
%                    with application to psychoacoustical masking
%                    (Ph.D. thesis, in preparation 2005)
%
% See also: GMappir,low2uppIr
```

```
%
% Copyright :  (c) NUHAG, Faculty of Math., University of Vienna, AUSTRIA
%              http://nuhag.mat.univie.ac.at/
%              (c) Acoustics Research Institute, Austrian Academy of
%              Science
%              http://www.kfs.oeaw.ac.at
%
%              Permission is granted to modify and re-distribute this
%              code in any manner as long as this notice is preserved.
%              All standard disclaimers apply.
function [TA,uppsym]=ApprFramMult(T,D,Ds)

if nargin < 2
    error('At leatst two inputs are needed; T and D');
end;
[N M] = size(T);
[Nd Kd] = size(D);

if N ~= Nd
    error('The number of rows in D and T have to be the same.');
    % as T:C^m -> C^n and D=( g_1 g_2 ...) with g_i \in C^n
end;

if nargin < 3
    Ds = D;
else
    [Ns Ks] = size(Ds);
    if N ~= Ns
        error('The number of rows in Ds and T have to be the same.');
    end;
    if Kd ~= Ks
        error('The frames must have the same number of elements.');
    end;
end;

lowsym = zeros(Kd,1); %lower symbol
for i=1:Kd
%    d = D(:,i);
%    ds = Ds(:,i);
%    lowsym(i) = conj(d'*(T*ds));
    lowsym(i) = conj(D(:,i)'*(T*Ds(:,i)));
end;
% the more elegant
% lowsym = diag(D'*T*D)
% is slower, O(k(n^2+n^2)))
% see [Xxl]

% Gram-matrix in Hilbert-Schmidt sense
if nargin < 3
    Gram = abs((D'*D)).^2;
else
    Gram = (D'*D).*((Ds'*Ds).');
end;

% upper symbol:
uppsym = pinv(Gram)*lowsym;

% synthesis
TA = zeros(N,M);
for i = 1:Kd
%    d = D(:,i);
%    ds = Ds(:,i);
```

```
%    P = d*ds';
    P = D(:,i)*Ds(:,i)';
    TA = TA + uppsym(i)*P;
end;
% found no faster or more elgant way
```

B.1.2 Test File

```
A = [ 3 0 ; 0 5];
D = [ 1/2 sqrt(3)/2 ;  sqrt(3)/2  -1/2];
[AD cD] = ApprFramMult(A,D)

T = eye(2,2)
% D = [cosd(30) cosd(150) cosd(270); sind(30) cosd(150) sind(270)]
D = [cosd(30) 1 0; sind(30) 1 -1]
[TA coeff] = ApprFramMult(T,D)
% % this frame is NOT tight, still the identity could be approximated
S = D*D';
eig(S)
tS = mpower(S,-0.5)
tD = tS*D
[TAt coefft] = ApprFramMult(T,tD)
```

B.1.3 Test File For Application To Gabor Systems

```
% Test for ApprFramMult.m in the Gabor case
% see ApprFramMult.m for more information
load colormapsw_xxl; %optimized colormapr for printing
n = 32;
g = gaussnk(n);
%gamma = hanning(n/2).';
gamma = hamming(n/2).';
gamma= [gamma(1:n/4) zeros(1,n/2) gamma((n/4+1):(n/2))];
% Gauss function. Algorithm can be found in NuHAG Gabmin Toolbox
G = gabbasp(g,2,2);
Ga = gabbasp(gamma,2,2);
% Gabor Synthesis Operator (from the right!).
% Algorithm can be found in NuHAG Gabmin Toolbox
Id = eye(n);
[IA1 coeff1] = ApprFramMult(Id,G.',Ga.');
figure(1);surf(abs(IA1));colormap(cmap);
S = G'*G;
e = eig(S);
disp(sprintf(...  % continued on the next line for printout
    '(g,2,2) : Lower Frame Bound : A = %g Upper Frame Bound: B = %g',...
    min(e),max(e)));
Sa = Ga'*Ga;
e = eig(Sa);
disp(sprintf(...  % continued on the next line for printout
    '(gamma,2,2) : Lower Frame Bound for g : A = %g Upper Frame Bound: B = %g',...
    min(e),max(e)));

G = gabbasp(g,4,4);
```

```
Ga = gabbasp(gamma,4,4);
[IA2 coeff2] = ApprFramMult(Id,G.',Ga.');
figure(2);surf(abs(IA2));colormap(cmap);
S = G'*G;
e = eig(S);
disp(sprintf(...
    '(g,4,4) : Lower Frame Bound : A = %g Upper Frame Bound: B = %g',...
    min(e),max(e)));
G = gabbasp(g,4,8);
Ga = gabbasp(gamma,4,8);
[IA3 coeff3] = ApprFramMult(Id,G.',Ga.');
figure(3);surf(abs(IA3));colormap(cmap);
S = G'*G;
e = eig(S);
disp(sprintf(...
    '(g,4,8) : Lower Frame Bound : A = %g Upper Frame Bound: B = %g',...
    min(e),max(e)));
G = gabbasp(g,8,8);
Ga = gabbasp(gamma,8,8);
Id = eye(n);
[IA4 coeff4] = ApprFramMult(Id,G.',Ga.');
figure(4);surf(abs(IA4));colormap(cmap);
S = G'*G;
e = eig(S);
disp(sprintf(...
    '(g,8,8) : Lower Frame Bound : A = %g Upper Frame Bound: B = %g',...
    min(e),max(e)));
G = gabbasp(g,16,16);
Ga = gabbasp(gamma,16,16);
[IA5 coeff5] = ApprFramMult(Id,G.',Ga.');
figure(5);surf(abs(IA5));colormap(cmap);
S = G'*G;
e = eig(S);
disp(sprintf(...
    '(g,16,16) : Lower Frame Bound : A = %g Upper Frame Bound: B = %g'...
    ,min(e),max(e)));
```

B.2 Irregular Gabor Frames And Multipliers

B.2.1 Irregular Gabor System

The following routine creates the full Gabor system over a given lattice. It returns the synthesis matrix. To be compatible with the routine gabbbasp from the Nuhag Toolbox, see [54], it is seen as a matrix for multiplication from the right, acting on row-vectors.

```
function     GBI = gabbaspirr(win,xpo);
% Creates the irregular Gabor frame n the lattice xpo
% gabbaspirr.M,   20.04.2005  XXL
%
% This program creates the irregular Gabor system using the window win at
% the points which are non-zero in xpo. This algorithm uses row vectors
% and matrix multiplication from the right to be compatible to gabbasp.m.
%
% Inputs :  win ... window, length n; row vector
%           xpo ... frequency time matrix, with K non-zero entries,
```

```
%                   rows: frequency
%
% Output :  GBI ... n x n matrix, the synthesis matrix of the system
%
% usage :   GBI = gabbaspirr(win,xpo);
%
% see also: gabbasp.m
%
% Author: XXL ..  a8927259@unet.univie.ac.at
%              or  xxl@kfs.oeaw.ac.at
%
% Notes: xpo was chosen to have the frequency values in the rows fitting
% the standard picture of a spectogram. So xpo(i,j) is the i-th frequency
% bin and j-the time frame. To fit to NuHAG-conventions, rotmod is used!!
% To get a matrix for left-multiplication use GBI.'
% We use first translation, then modulation: g_k = M_x_k  T_y_k  g%
%
% Lit.: [XXL] Peter Balazs, "Irregular and regular Gabor multiplier with
%             application to psychoacoustic masking"
%
% Copyright :  (c) NUHAG, Faculty of Math., University of Vienna, AUSTRIA
%              http://nuhag.mat.uunivie.ac.at/
%              (c) Acoustics Research Institute, Austrian Academy of
%              Science
%              http://www.kfs.oeaw.ac.at
%
%              Permission is granted to modify and re-distribute this
%              code in any manner as long as this notice is preserved.
%              All standard disclaimers apply.

if nargin < 2;  error('two inputs needed.');  end;
n = length(win);
[nn n] = size(win);
if nn ~= 1
    if n ~= 1
        error('the window has to be a vector!');
    end
    win = win.';
    n = nn;
end;
if size(xpo) ~= [n,n];
    error('The Lattice does not fit the size of the window.');
end;
[xpx xpy] = find(xpo > 0);
k = size(xpx,1);
if k == 0
  error('The matrix xpo is containing only zeros');
end

GBI = zeros(k,n);
for ii = 1:k
%     xpx(ii)
%     xpy(ii)
    gg = rotmod(win,xpy(ii)-1,xpx(ii)-1);
%    plot(gg)
    GBI(ii,:)=gg;
%    pause
end

% F=ifft(eye(n))*n;
% tso = -1;
% for ii = 1:k
```

```
%       tsn = xpy(ii) % time-shift-new
%       if tsn ~= tso
%           g = rot(win,tsn-1)
%           tso = tsn
%       end;
%       for jj = 1:k
%          GBI(jj,:)=F(xpx(jj),:).*g;
%      end
% end
```

B.2.1.1 Testfile

```
% test gabbaspIrr
reg = 1; % test with regular lattice ?
show = 0; % show movie?
n = 144;
g = randc(1,n);
gamma = g;
f = randc(1,n);
% g = gaussnk(n);
% g = rand(1,n);
if reg == 1
    a = 9;
    b = 8;
    xpo=zeros(n);
    xpo(1:b:n,1:a:n) = 1;
else
    xpo = rand(n) > 1-2/n; % red = 2
end;
GBI=gabbaspirr(g,xpo);
%GBIa = gabbaspirr(g,xpo);
GBIa = GBI;
[K N] = size(GBI);
if show == 1
    clear M;
    for ii=1:K
        hold on;
        imagesc(abs(stft(GBI(ii,:),g)));spy(xpo,'w');
        hold off;
    %    pause
        M(ii) = getframe;
    end
    movie(M)
end;
Si = GBIa'*GBI; % Analyse: gamma
disp(sprintf('Rank of Si: %g',rank(Si)));

if reg == 1
    for ii=1:K
         k2 = floor((ii-1)*b/n);
         k1 = mod(ii-1,n/b);
         % => ii = k2*n/b+k1+1
         g1 = GBI(ii,:);
         g2 = rotmod(g,k2*a,k1*b);
         % g2 = rotmod(g,k2*a,k1*b);
         if norm(g1-g2)> 0.1
             disp(sprintf('Problem: time: %g, freq.: %g',k2,k1));
             plot(real(g1)); hold; plot(real(g2),'r')
             return
         else
             % disp(sprintf('Okay: time: %g, freq.: %g',k2,k1));
```

```
        end;
        %imgc(stft(G(ii,:),g));
        %M(ii) = getframe;
    end
    disp('Everything okay!');
end;
[xpx xpy] = find(xpo > 0);

Vfull = stft(f,g,1,1);
% row index of STFT (Gabmin or NuHAGTB05a) are frequencies !!!!
V = zeros(1,K);
for ii = 1:K
    V(ii) = Vfull(xpx(ii),xpy(ii));
end;

W = f*GBI';
compnorm(V,W)
```

B.2.2 Kohn-Nirenberg Symbol

```
function   KN=kohnniren(M);
% Calculate the Kohn-Nirenberg symbol of M
% kohniren.M,   20.04.2005  XXL
%
% This program calculates the Kohn-Nirenberg Symbol of the matrix M. In the
% finite-dimensional case the matrix corresponds to the integration kernel
% and therefore:
% KN (M) = F_2 T_a M = fft(col2diag(M)); (see [XXL])
%
% Inputs :  M ... square n x n matrix
%
% Output :  CD ... n x n matrix: the Kohn-Nirenberg symbol
%
% usage :    DM  =  kohnniren(M);
%
% see also: ker2kohn.m (same functionality, no comments)
%
% Author: XXL ..  a8927259@unet.univie.ac.at
%              or  xxl@kfs.oeaw.ac.at
%
% Notes: this is exactly the operation used for the Kohn-Nirenberg symbol!
%
% Lit.: [XXL] Peter Balazs, "Irregular and regular Gabor multiplier with
%             application to psychoacoustic masking"
%
% Copyright :  (c) NUHAG, Faculty of Math., University of Vienna, AUSTRIA
%              http://nuhag.mat.uunivie.ac.at/
%              (c) Acoustics Research Institute, Austrian Academy of
%              Science
%              http://www.kfs.oeaw.ac.at
%
%              Permission is granted to modify and re-distribute this
%              code in any manner as long as this notice is preserved.
%              All standard disclaimers apply.


if nargin < 1;  error('no input'); end;
[h,n] = size(M);
```

```
if h ~= n ;
    disp('M has to be square matrix')
    return;
end;

KN = fft(col2diagxxl(M).');
```

B.2.2.1 Testfile:

```
% test Kohn-Nirenberg symbol implementation with Rihaczek distribution
n = 144;
F = fft(eye(n));
g = gaussnk(n);
Rih = (g.'*conj(fft(g))).*F; % Rihaczek distribution
KN = kohnniren(g'*g);
compnorm(Rih,KN);
```

B.3 Approximation Of Matrices By Irregular Gabor Multiplier

B.3.1 The Gram Matrix Of The TF Projections

```
% determination of Gramian Matrix of Gabor rank one operators
%  HSGramMatrXXL.M   XXL,  13.05.2004
%
% function   hsgm = HsGramMatrXXL(xpo,g,gamma,show,full)
%
%  determines either the full Hilbert Schmidt Gram Matrix of the
%  rank one operators  f -> <f, \pi(\lambda) \gamma> \pi(\lambda')
%  g (full = 1) or only the entries important for a Gabor
%  multiplier (full = 0)
%
%  inputs : xpo   .... (0/1) square matrix of points in the TF plane
%                      row index - frequency, column index -time
%                      (following the convention for normal spectograms)
%           g     .... synthesis window row vector (1 x m),
%                      Default: Gauss
%           gamma .... analysis window row vector (1 x m'),
%                      Default: g
%           show  .... show graphics (slower) (*/0), Default: 1
%                      (if input ~= 0, show = 1!)
%           full  .... calculate the HS Gram Matrix of all
%                      possible TF projections
%                      or only
%                      the entries which are used in Gabor
%                      multipliers, where the tensor product only
%                      depends on one TF point (1/*),
%                      Default: 0 (if input ~= 1, full = 0!)
%
%  output: the k^2 x k^2 HS gram matrix, where k is the number of
%          non zero entries in xpo. (full = 1)
%          the k x k part of HS gram matrix, important for Gabor
%          multiplier (otherwise)
%
%  usage:  HsGramMatrXXL(xpo,g,gamma,show,full)
%
```

```
%  notes: The entries of this HS Gram Matrix are (full == 1)
%         < g_k \otimes gamma*_j , g_i \otimes gamma*_l >_HS =
%       = < g_k \otimes gamma*_j , g_i \otimes gamma*_l >_L2 =
%       = < g_k , g_i > < gamma_j, gamma_l >* =
%       = STFT_gamma(gamma)_(k,i) * STFT*_g(g)_(j,l)
%
%       (full == 0) => k = j, i = l
%
% See also: GMAPPirr, teststft
%
% Ref.: [xxl]... P. Balazs, "Regular And Irregular Gabor Multipliers With
%                Application To Psychoacoustic Masking"
%
% complexity: O(n^4) (full == 1) or O(n^2)
%             (so can be time consuming)
%
%  XXL .. Peter Balazs, contact: a8927259@unet.univie.ac.at
%
% notes:
% The matrices A,B are (up to a phase factor) the "Cross"-Gram
% Matrices of the TF atoms g_j, g_l resp. gamma_k, gamma_i.
%
% For full == 1, the complexity is about O(n^4), most of the time
% is used by the function "kron" if n is big enough. This function
% is a built in function, so should be fairly optimized.
% Otherwise most of the time is used up by the assignement of the
% matrcies.
% For full == 0, the complexity is O(n^2).
% For n=144, k = 0,5 % of N (full == 0) this functions needs about
% 12 sec. (PIII 937)
% For n=144, k = 1,5 % of N (full == 0) this functions needs about
% 101 sec.
%
% Copyright :  (c) NUHAG, Faculty of Math., University of Vienna, AUSTRIA
%              http://nuhag.mat.univie.ac.at/
%              (c) Acoustics Research Institute, Austrian Academy of
%              Science
%              http://www.kfs.oeaw.ac.at
%
%              Permission is granted to modify and re-distribute this
%              code in any manner as long as this notice is preserved.
%              All standard disclaimers apply.

function   hsgm = HsGramMatrXXL(xpo,g,gamma,show,full)
% to test how much time is used by which part, use profiling:
% profile on -detail builtin;
if nargin < 1
  error('Function HSGramMatrXXL has to get at least one parameter : xpo');
end
[N M] = size(xpo);
if M ~= N
    error('XPO has to be a square matrix');
end;
[xpx xpy] = find(xpo > 0);
% xpx ... frequencies
% xpy ... time
k = size(xpx,1);
if k == 0
  error('The matrix xpo is containing only zeros');
end
if nargin < 2
  g = gaussnk(N);
```

```
  disp('using default gaussian analysis window');
else
  % Maybe create a subfunction check(g);
  [n m] = size(g);
  if m < n
    g = g.';
    disp('transposing g');
    [n m] = size(g);
  end
  if n ~= 1
    error('Input g should be a vector')
  end
  if m > N
    error('The size of g is bigger than the (spectral) size of xpo')
  elseif m < N
    gaga = zeros(1,N - size(g)) % zeropadding
    gamma = [gamma;gaga]
  end
end

if nargin < 3
  gamma = g;
  disp('using default synthesis window: gamma = g');
else
  [n m] = size(gamma);
  if m < n
    gamma = gamma.';
     disp('transposing gamma');
     [n m] = size(gamma);
  end
  if n ~= 1
    error('Input gamma should be a vector')
  end
  if m > N
    error('The size of g is bigger than the (spectral) size of xpo')
  elseif m < N
    gg = zeros(1,N - m) % zeropadding
    g = [g;gg]
  end
end

if nargin < 4
  show = 0;
elseif show ~= 0
  show = 0;
end

if nargin < 5
  full = 0;
elseif full ~= 1
  full = 0;
end

gst = stft(g,g,1,1); % full stft
gast = stft(gamma,gamma,1,1); % full stft

if show == 1
  figure(1);
  subplot(2,1,1);
  imgc(gst);
  title('Wigner-like distribution of analysis atom');
  subplot(2,1,2);
```

```
  imgc(gast);
  title('Wigner-like distribution of synthesis atom');
end
A = zeros(k);
B = zeros(k);

% try to programm this part using MATLAB matrix functions
for ii = 1:k
  for jj = 1:k
     A(ii,jj) = gst(mod(xpx(jj)-xpx(ii),N)+1,mod(xpy(jj)-xpy(ii),N)+1);
     % row index of STFT are time values, see teststft.m
     % using periodic extension
     B(ii,jj) = conj(gast(mod(xpx(jj)-xpx(ii),N)+1,mod(xpy(jj)-xpy(ii),...
         N)+1));
     % conjugate for a 'real' tensor product !!!!!
    end
end

if full == 1
  hsgm = kron(A,B);
else
  hsgm = A.*B;
end

if show == 1
  figure(2);
  SURF(abs(hsgm));
end
% Profiling end part:
% profile report;
% figure(3);
% profile plot;
% profile off;
```

B.3.1.1 Testfile:

```
% test HSGRamMatrXXL
%
% Ref.: P. Balazs, "Regular And Irregular Gabor Multipliers With
%       Application To Psychoacoustic Masking"
%
% See testproj, HSGramMatrXXL

n = 144;

g = randc(1,n); % Synthesis Atom
gamma = randc(1,n); % Analysis Atom

% g = gaussnk(n);
% gamma = hamming(n/2).';
% gamma= [gamma(1:n/4) zeros(1,n/2) gamma((n/4+1):(n/2))];

% for regular case :
% xpo = zeros(n,n);
% a = 9;
% b = 9;
% xpo(1:b:n,1:a:n) = 1;

xpo = rand(n) > 1-2/n; % red = 2
G = gabbaspirr(g,xpo);
```

```
Ga = gabbaspirr(gamma,xpo);
Gram1 = HSGramMatrXXL(xpo,g,gamma);

[xpx xpy] = find(xpo > 0);
K = length(xpx);
Gram2 = zeros(K);
Vfull = stft(g,g,1,1);
Wfull = stft(gamma,gamma,1,1);
for ii = 1:K;
    for jj = 1:K;
    Gram2(ii,jj) = ...
        Vfull(mod(xpx(jj)-xpx(ii),n)+1,mod(xpy(jj)-xpy(ii),n)+1).*...
        conj(Wfull(mod(xpx(jj)-xpx(ii),n)+1,mod(xpy(jj)-xpy(ii),n)+1));
    end;
end;

%PHS = zeros(K,n*n);
%Pnull = gamma'*g;
%for ii = 1:K;
%    Ptemp = tfconj(Pnull,xpy(ii)-1,xpx(ii)-1);
%    PHS(ii,:) = Ptemp(:);
%end;
%Gram3 = PHS*PHS';
% Gram3 = abs(PHS*PHS'); % why abs ????
%
% ATTENTION: Tfconj (Gabmin) imeplements modrot, not rotmod !!!!!!!!!!!!
%

Gram4 = (G*G').*conj(Ga*Ga'); % conj. !!

PHS2 = zeros(K,n*n);
for ii = 1:K;
    gal = rotmod(gamma,xpy(ii)-1,xpx(ii)-1);
    gl = rotmod(g,xpy(ii)-1,xpx(ii)-1);
%   gal = modrot(gamma,xpx(ii)-1,xpy(ii)-1); %for comparison with tfconj
%   gl = modrot(g,xpx(ii)-1,xpy(ii)-1);
    Ptemp = gal'* gl;
    PHS2(ii,:) = Ptemp(:);
end;
% disp('*** PHS - PHS2:');compnorm(PHS,PHS2);
Gram5 = PHS2*PHS2';

disp('                                          ');
disp('*** Gram1-Gram2');compnorm(Gram1,Gram2);
%disp('*** Gram1-Gram3');compnorm(Gram1,Gram3);
disp('*** Gram1-Gram4');compnorm(Gram1,Gram4);
disp('*** Gram1-Gram5');compnorm(Gram1,Gram5);
%disp('*** Gram2-Gram3');compnorm(Gram2,Gram3);
disp('*** Gram2-Gram4');compnorm(Gram2,Gram4);
disp('*** Gram2-Gram5');compnorm(Gram2,Gram5);
%disp('*** Gram3-Gram4');compnorm(Gram3,Gram4);
%disp('*** Gram3-Gram5');compnorm(Gram3,Gram5);
disp('*** Gram4-Gram5');compnorm(Gram4,Gram5);
% rank(G'*G)
%rank(Ga'*Ga)

% Tests:
% n = 144;
% *** Gram1-Gram2
% Norm of first input x: 9387.5 , norm of second input y: 9387.5 , quotient =  1 .
% norm of the difference of the normalized versions  = 0 .
% *** Gram1-Gram4
```

```
% Norm of first input x: 9387.5 , norm of second input y: 9387.5 , quotient =  1 .
% norm of the difference of the normalized versions  = 4.2874e-015 .
% *** Gram1-Gram5
% Norm of first input x: 9387.5 , norm of second input y: 9387.5 , quotient =  1 .
% norm of the difference of the normalized versions  = 5.11798e-015 .
% *** Gram2-Gram4
% Norm of first input x: 9387.5 , norm of second input y: 9387.5 , quotient =  1 .
% norm of the difference of the normalized versions  = 4.2874e-015 .
% *** Gram2-Gram5
% Norm of first input x: 9387.5 , norm of second input y: 9387.5 , quotient =  1 .
% norm of the difference of the normalized versions  = 5.11798e-015 .
% *** Gram4-Gram5
% Norm of first input x: 9387.5 , norm of second input y: 9387.5 , quotient =  1 .
% norm of the difference of the normalized versions  = 2.85202e-015 .
```

B.3.2 Approximation Algorithm

```
% Approximation of Matrices by irregular Gabor multiplier
% function [TAI,COEFF]=gmappirr(T,xpo,g,gamma,show)
%
% The best approximation of a matrix by a Gabor multiplier with
% analysis window g and synthesis window gamma on the (irregular) grid xpo
% is calculated.
%
% inputs : T ......... the matrix
%          g ......... Gabor synthesis window, row vector
%          gamma ..... Gabor analysis window, row vector
%          xpo ....... the (possibly) irregular grid, row index = frequency
%          show ...... flag if error should be calculated
%
% output: TAI ....... the best approximation by Gabor multipliers
%         coeff ..... the lower symbol
%
% usage:  [TAI,COEFF] = Blo2WalXXL(A,n)
%
% test:   use TestGabMulAppIrr.m
%
% last change: 01/05/2005
%
% notes: Some files out of the NuHAG Toolbox are used, so the convetion
%        there for using row vectors and matrix multiplication from the
%        right is used !!
%
% see also: ApprFramMult.m, HSGramMatrxxl.m, testhsgrammatr.m
%
% Author:  XXL .. Peter Balazs, contact: a8927259@unet.univie.ac.at
%
% Literature : [Xxl] P. Balazs; Gabor frame multipliers with application to
%                    psychoacoustical masking
%                    (Ph.D. thesis, in preparation)
%              [FHK] H. G. Feichtinger, M. Hampjes, G. Kracher;
%                    Approximation of Matrices by Gabor Multipliers,
%                    IEEE Signal Procesing Letters Vol. 11, No. 11 (2004)
%              [Doe] M. Doerfler; Gabor Analysis for a Class of Signals
%                    called Music, PhD thesis Univ. Wien (2002)
%
%
% Copyright :  (c) NUHAG, Faculty of Math., University of Vienna, AUSTRIA
%              http://nuhag.mat.univie.ac.at/
%              (c) Acoustics Research Institute, Austrian Academy of
```

```
%              Science
%              http://www.kfs.oeaw.ac.at
%
%              Permission is granted to modify and re-distribute this
%              code in any manner as long as this notice is preserved.
%              All standard disclaimers apply.
function [TAI,uppsym]=gmappirr(T,xpo,g,gamma,show)

if nargin < 2
  error('Function GMAPPIr has to get at least two parameters : T,xpo');
end
[N M] = size(xpo);
xxx = find(xpo.' > 0);
k = size(xxx,1);
if k == 0
  error('The matrix xpo is containing only zeros');
end

[TN TM] = size(T);
if TN ~= TM
    error('At the moment only square matrices can be used.');
elseif TN ~= N
    error('Matrix size does not fit grid size.');
end;

if nargin < 3
  g = gaussnk(N);
  disp('using default gaussian analysis window');
else
  % Maybe create a subfunction check(g);
  [n m] = size(g);
  if m < n
    g = g.';
    disp('transposing g');
    [n m] = size(g);
  end
  if n ~= 1
    error('Input g should be a vector')
  end

  if m < N | m < M
    error('The length of g is too small for xpo.');
  end
end

if nargin < 4
  gamma = g;
  disp('using default synthesis window: gamma = g');
else
  [n m] = size(gamma);
  if m < n
    gamma = gamma.';
     disp('transposing gamma');
     [n m] = size(gamma);
  end
  if n ~= 1
    error('Input gamma should be a vector')
  end
  if m < N | m < M
    error('The length of g is too small for xpo.');
  end
end
```

```
GBI = gabbaspIrr(g,xpo);
GBIa = gabbaspIrr(gamma,xpo);
lowsym = zeros(k,1); %lower symbol
for ii=1:k
    lowsym(ii) = (GBI(ii,:)*T)*(GBIa(ii,:)');
end;

% Gram-matrix in Hilbert-Schmidt sense
Gram = HSGramMatrXXL(xpo,g,gamma);

% upper symbol:
uppsym = pinv(Gram)*lowsym;

% synthesis
TAI = zeros(N,M);
for ii = 1:k
    P = GBIa(ii,:)'*GBI(ii,:);
    TAI = TAI + uppsym(ii)*P;
end;

if nargin < 5 | show ~= 1
   show=0;
end
if show==1
    disp('Fehler:');
    norm(T-TA,'fro')
end;
```

B.3.2.1 Testfile:

```
% Test the Approximation of Matrices by Gabor Mulitpliers
% Script-File
% see GMAPPirr.m for more information
reg = 0;
single = 0;
n = 32;
g = gaussnk(n);
if single == 1
    gamma = g;
else
    gamma = hamming(n/2).';
    gamma= [gamma((n/4+1):(n/2)) zeros(1,n/2) gamma(1:n/4)];
    gamma = gamma/norm(gamma);
    figure(5); subplot(1,2,1); plotc(g); title('Analysis Atom:');
    subplot(1,2,2); plotc(gamma); title('Synthesis Atom:');
end;
%gamma= [gamma(1:n/4) zeros(1,n/2) gamma((n/4+1):(n/2))];
%imagesc(abs(stft(g,g)));
%pause;
%imagesc(abs(stft(gamma,g)));

if reg == 1
    a = 4;
    b = 4;
    xpo = zeros(n,n);
    xpo(1:b:n,1:a:n) = 1;
    T = randc(n,n);
    % T = rot(eye(n),6);
```

```
else
    T = eye(n);
    xpo = rand(n) > (1-2/n); % red = 2
end;
[TAI,COEFFI] = GMAPPir(T,xpo,g);
G = gabbaspirr(g,xpo);
% Ga = gabbaspirr(gamma,xpo);
S = G'*G;
rank(S)
e = eig(S);

if reg == 1
    disp(sprintf(...  % continued on the next line for printout
    '(g,%g,%g) : Lower Frame Bound for g : A = %g Upper Frame Bound: B = %g',...
    a,b,min(e),max(e)));
else
    disp(sprintf(...  % continued on the next line for printout
    '(g,xpo) : Lower Frame Bound for g : A = %g Upper Frame Bound: B = %g',...
    min(e),max(e)));
end

[IA1 coeff1] = ApprFramMult(T,G.');
compnorm(TAI,IA1);
figure(6);  subplot(1,2,2); surf(TAI); hold; spy(xpo); title('approximation');
 subplot(1,2,1); surf(T); hold; spy(xpo); title('original');

figure(1);
subplot(1,2,1);surf(T);title('original');
subplot(1,2,2);surf(TAI); title('approximation');
figure(2);
subplot(1,2,1); spy(xpo); hold; contour(T); title('original');
subplot(1,2,2); spy(xpo); hold; contour(TAI); title('approximation');

if reg == 1 & single == 1
 TA = gmappmh(T,g,a,b); % regular version
 figure(4)
 subplot(3,1,1);surf(T);title('original');
 subplot(3,1,2);surf(TA); title('approximation (regular by MH)');
 subplot(3,1,3);surf(TAI); title('approximation (irregular)');
 compnorm(TA,TAI);
end
```

B.4 Discrete Gabor Transformation

For the applications in this thesis and the numerical tests a lot of programs were needed. Some of these programs are based of the work and algorithms of H.G. Feichtinger, S. Qiu and M. Hampejs, see e.g. [104] and [50]. A few of these files are only minor reformulations of existing algorithms, but e.g. adding comments. I have to thank NuHAG and especially H.G. Feichtinger for providing these basics.

Some existing algorithms have been done completly anew, as either there was some error in the code, which was hard to find and / or the documentation was not good enough, so that it so sometimes took less time to reprogram it, than to try to understand the existing code respectivly find the error.

Some of these algorithms have been developed simultaneously by G. Kracher and M. Hampejs during the joint work on 'Double Preconditioning of the Gabor frame operator' [9].

B.4.1 Basic routines

B.4.1.1 Matrix Fourier Transformation

```
% FMF.M   Fourier Transformation of a matrix
% by XXL
%
% inputs : A .... m x n Matrix with the non trivial data
%
% output: the mxn matrix FSF with
%           FSF = F_m * A * F'_n
%
% this uses the unitary FFT!
%
% notes: fast version (01/03/2005)
%
% XXL .. Peter Balazs, contact: a8927259@unet.univie.ac.at

function U = FMF(A);

[n m] = size(A);

% (1) slow but instructive:
% F = fftu(eye(n));
% F1 = fftu(eye(m));
% UT = F*A*F1';
%
% (2) fast
UT = ifft(fft(A).').';
UT = UT*(sqrt(m)/sqrt(n));
% to get unitary result!
indx = find(abs(UT)>10*eps);
U=zeros(m,n);
U(indx)=UT(indx);
```

B.4.1.2 Inverse Matrix Fourier Transformation

```
% iFMF.M   inverse Fourier Transformation of a matrix
% by XXL
%
% inputs : A .... m x n Matrix with the non trivial data
%
%  output: the mxn matrix FSF with
%           FSF = F_m * A * F'_n
%
% notes: fast version (01/03/2005)
%
% XXL .. Peter Balazs, contact: a8927259@unet.univie.ac.at

function U = iFMF(A);
```

```
[n m] = size(A);

% (1) slow but instructive:
% F = fftu(eye(n));
% F1 = fftu(eye(m));
% UT = F'*A*F1;
%
% (2) fast
UT = fft(ifft(A).').';
UT = UT*(sqrt(n)/sqrt(m));
% to get unitary result!

indx = find(abs(UT)>10*eps);
U=zeros(m,n);
U(indx)=UT(indx);
```

B.4.1.3 Initialisation of Gabor atoms et al.

This is used to run all basic algorithms from the Gabmin Toolbox see [54].

```
% GABminINIT   Gabor initialization file
%
% initalize variables for gabor analysis
% -> original (g), dual (gd) and tight window (gt)
% -> corresponding analysis operator (G, GD, GT), also for adjoint lattice
% (GA)
% -> frame operator for original window and lattice (S)
% Show windows and their spectra
%
% derived from gabinit.m ny HGFei
% modified by XXL to work with the GABMIN tools
% but check for other files

% XXL: contact a8927259@unet.univie.ac.at
% last modification 09.12.2004

%clc;
disp(' Attention. This script tries to use existing variables. ');
disp(' If you want to make sure, that you use the desired variables, ');
disp(' assign g,a,b and n. At the moment ')
whos g;

if exist('a') == 1;
    a
end;
if exist('b') == 1;
    b
end;
if exist('n') == 1;
    n
else
    if exist('a') ~= 1 & exist('b') ~= 1
        disp ('No variable is  already set. So');
    end
    n = 144
%    disp('n is set to 144');
end;
if exist('g') == 1;
    [N M] = size(g);
```

```
    if N > 1;
        g=g.';
        [N M] = size(g);
    end;
    if N ==  1 & M > 1;
       n = M;
       disp(sprintf('g exists so n is set to %g',M));
   else
        disp('Gauss function as window g created.');
        g = gaussnk(n);
    end;
else
    disp('Gauss function as window g created.');
    g = gaussnk(n);
end;

alph = alldiv(n); alph = alph(:)';
mx1 = max(find(alph <= sqrt(n)));
divs = alph(1:mx1),
if exist('a') ~= 1 | a < 0 | a > n | mod(n,a) ~= 0;
    a = alph(mx1)
     %disp(sprintf('a is set to %g',a));
end;
if exist('b') ~= 1 | b < 0 | b > n |  mod(n,b)~= 0; % eval('b',0) ~= 0 | a*b > n |
    if mx1 == 1;
        b = round(n/a)
        if b == 0;
            b = 1
        end;
    else
        b = alph(mx1-1)
    end;
    %disp(sprintf('b is set to %g',b));
end;
if a*b > n
    disp('The redundancy is smaller than one, so this cant be a frame');
end;

red = n/(a*b)
xpo = lattp(n,a,b);
xpa = lattp(n,n/b,n/a);
gtt = '?';

disp('calculating dual atom');
% pause(1);
if rem(red,1) == 0 & exist('zd') == 2;
    gd = zd(g,a,b);
    disp('using Zak transform with integer oversampling');
elseif red == 1;
    gd = zakdfei(g,a,b);
    disp('using Zak transform in the critical case');
else
    gd = gabddd(g,a,b);
end;
disp('dual atom done');
% pause(0.2);
if rem(red,1) == 0 & exist('zt') == 2;
   disp('calculating tight atom, using Zak transform');
   gt = zt(g,a,b);
elseif exist('gabtgf') == 2;
   disp('dual atom done, doing tight atom now');
   gt = gabtgf(g,a,b);
```

```
else
    disp('no tight atom is calculated, no tool in Gabmin');
    gtt = 'no';
    % gt is a built-in function, so use actual assignment and not clear
end;
% f3sp(g,gt,gd);
disp('now establishing matrices');
pause(1);
if red*n < 540
    G = gabbasp(g,a,b);
    GD = gabbasp(gd,a,b);
    if gtt ~= 'no';
        GT = gabbasp(gt,a,b);
    end;
    S = G'*G;
    GA =  gabbasp(g,n/b,n/a);
else
    sprintf('red*n is too large (%d) to automatically establish matrices',red*n);
end;
disp('call "whos" for more details');
% disp('f3sp(g,gt,gd); shows Gabor atoms, tight + dual');
if gtt ~= 'no' & exist('gt') == 1 & exist('f3sp') == 2;
    f3sp(g,gt,gd);
else
    f2sp(g,gd);
end;
```

B.4.1.4 Diag2Row

```
function   DM=diag2row(M,F);
% Make the diagonals of a matrix the rows of a new matrix
% diag2col.M,   29.10.04  XXL
%
% This program produces a compressed collection of the side diagonals
% as rows of a new matrix, the program transfers the
% sidediagonals to the rows of the new matrix, only using every
% F-th diagonal.
% This program uses the main diagonal as first row and
% the F-th sidediagonal under the main diagonal as second row
% and so on:
% DM_i,j = M_{i*F+j,j}
%
% Inputs :  M ... square n x n matrix
%           F ... a divisor of n
%                 (Default) F = 1;
%
% Output :  DM ... n/F x n   matrix,
%
% usage :    DM  =  diag2row(M,F);
%
% see also: diag2col, SIDEDIGM, Blo2WalXXl, Wal2BloXXL
%
% Author: XXL ..  a8927259@unet.univie.ac.at
%              or  xxl@kfs.oeaw.ac.at
% based on:
%       SIDEDIGM.M,   18.11.93 by  Sigang Qiu
%
% Copyright :  (c) NUHAG, Faculty of Math., University of Vienna, AUSTRIA
%              http://nuhag.mat.uunivie.ac.at/
%              (c) Acoustics Research Institute, Austrian Academy of
%              Science
```

```
%              http://www.kfs.oeaw.ac.at
%
%              Permission is granted to modify and re-distribute this
%              code in any manner as long as this notice is preserved.
%              All standard disclaimers apply.

if nargin == 1;  F = 1; end;
[h,n] = size(M);
if h ~= n ;
    disp('M has to be square matrix')
    return;
end;
if mod(n,F) ~= 0;
    sprintf('F (= %g) should be a divisor of n (%g)',F,n)
    return;
end;

DM = zeros(n/F,n);

for jj = 1 : n/F;
    tt = mod((jj-1)*F:(jj-1)*F+n,h)+1;
    temp = M(tt,:);
    DM(jj,:)=diag(temp);
end;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Old code:
% st = (jj-1)*F*n ;
% ndx = rem(st-1 + (1: (n+1) : n^2), n^2) + 1 ;
% sd = M(ndx);
% DM(jj,:) =  sd;
%
% st = (jj-1)*F*n ;
% ndx = rem(st-1 + (1: (n+1) : n^2), n^2) + 1 ;
% sd = M(ndx);
% DM(jj,:) =  sd;
% translation matrix(j):
% P = rot(eye(n),k);
```

B.4.1.5 Diag2Col

```
function   DM=diag2col(M,F);
% Make the diagonals of a matrix the columns of a new matrix
% diag2col.M,   12.01.2005  XXL
%
% This program produces a compressed collection of the side diagonals
% as columns of a new matrix, the program transfers the
% sidediagonals to the columns of the new matrix, only using every
% F-th diagonal.
% This program uses the main diagonal as first column and
% the F-th sidediagonal right of the main diagonal as second column
% and so on:
% DM_i,j = M_{i,i+j*F}
%
% Inputs :  M ... square n x n matrix
%           F ... a divisor of n
%                 (Default) F = 1;
%
% Output :  DM ... n/F x n   matrix,
%
% usage :    DM  =  diag2col(M,F);
```

```
%
% see also: diag2row, SIDEDIGM, Blo2WalXXl, Wal2BloXXL
%
% Author: XXL ..  a8927259@unet.univie.ac.at
%              or  xxl@kfs.oeaw.ac.at
% based on:
%       SIDEDIGM.M,   18.11.93 by  Sigang Qiu
%
% Copyright :  (c) NUHAG, Faculty of Math., University of Vienna, AUSTRIA
%              http://nuhag.mat.uunivie.ac.at/
%              (c) Acoustics Research Institute, Austrian Academy of
%              Science
%              http://www.kfs.oeaw.ac.at
%
%              Permission is granted to modify and re-distribute this
%              code in any manner as long as this notice is preserved.
%              All standard disclaimers apply.


if nargin == 1;  F = 1; end;
[h,n] = size(M);
if h ~= n ;
    disp('M has to be square matrix')
    return;
end;
if mod(n,F) ~= 0;
    sprintf('F (= %g) should be a divisor of n (%g)',F,n)
    return;
end;

DM = zeros(n,n/F);

for jj = 1 : n/F;
tt = mod((jj-1)*F:(jj-1)*F+n,h)+1;
temp = M(:,tt);
DM(:,jj)=diag(temp);
end;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

B.4.1.6 Col2Diag

```
function   CD=col2diagxxl(M);
% Reorder matrix to switch the columns to the sidediagonals
% col2diagxxl.M,   20.04.2005  XXL
%
% This program reorders the matrix, switching the entries from the same
% column to the same side-diagonal. The first row stays fixed.
% CD_i,j = M_{i,i-j}
%
% Inputs :  M ... square n x n matrix
%
% Output :  CD ... n x n matrix
%
% usage :    DM  =  col2diagxxl(M);
%
% see also: diag2row, SIDEDIGM, Blo2WalXXl, Wal2BloXXL
%
% Author: XXL ..  a8927259@unet.univie.ac.at
%              or  xxl@kfs.oeaw.ac.at
%
```

```
% Notes: this is exactly the operation used for the Kohn-Nirenberg symbol!
%
% Lit.: [XXL] Peter Balazs, "Irregular and regular Gabor multiplier with
%             application to psychoacoustic masking"
%
% Copyright :  (c) NUHAG, Faculty of Math., University of Vienna, AUSTRIA
%              http://nuhag.mat.uunivie.ac.at/
%              (c) Acoustics Research Institute, Austrian Academy of
%              Science
%              http://www.kfs.oeaw.ac.at
%
%              Permission is granted to modify and re-distribute this
%              code in any manner as long as this notice is preserved.
%              All standard disclaimers apply.


if nargin < 1;  error('no input'); end;
[h,n] = size(M);
if h ~= n ;
    disp('M has to be square matrix')
    return;
end;

CD = zeros(n,n);

jj = 1 : n;
for ii = 1 : n;
CD(ii,jj)=M(ii,mod(jj-ii,n)+1);
end;
```

B.4.2 The block structure of the Gabor frame matrix

B.4.2.1 Block to full matrix

```
% Create a full (Gabor-type) matrix out of a block matrix
%
% Blo2WalXXL.M   Block-to-Walnut-matrix
% by XXL
%
% This program creates a matrix that looks like a Gabor frame
% matrix, a so called Gabor-like or Walnut matrix,  meaning only every
% M=L/b sidediagonal is not zero and the side diagonals are a-periodic.
% So the matrices created have b*b diagonal blocks and are
% a-block circulant.
%
% Attention: it is important how the gabor like matrix is built.
%            With this program both possibilities are supported.
%
% inputs : A ........ block Matrix with the non trivial data
%          n ......... (minimum) size of result
%          method .... 0 - b x a block matrix
%                      1 - a x b block matrix
%
%  output: the n x n matrix having only the data of A as non trivial
%          values. The rest is defined by being a "Gabor frame like"
%          matrix, see above.
%
%  usage:  B = Blo2WalXXL(A,n)
%
%  notes : method == 0 corresponds to the algorithm first used and
```

```
%          implemented by Qiu and Feichtinger, where the block matrix is
%          the condensed left b x n submatrix of the frame operator
%          method == 1 uses the a x b matrix, which is derived from the
%          upper submatrix a x n submatrix
%
% Author:   XXL .. Peter Balazs, contact: a8927259@unet.univie.ac.at
%
% Literature : [Qiu] S. Qiu, H.G. Feichtinger, Gabor-Type Matrices And
%                    Discrete Huge Gabor Transforms
%              [Xxl] P. Balazs, Gabor frame multipliers with application to
%                    psychoacoustical masking
%                    (Ph.D. thesis, in preparation)
%
% See also: blockxxl.m - which builds a block matrix for a gabor system
%           bm2fm.m    - version of Mario Hampjes
%
% Copyright :  (c) NUHAG, Faculty of Math., University of Vienna, AUSTRIA
%              http://nuhag.mat.uunivie.ac.at/
%              (c) Acoustics Research Institute, Austrian Academy of
%              Science
%              http://www.kfs.oeaw.ac.at
%
%              Permission is granted to modify and re-distribute this
%              code in any manner as long as this notice is preserved.
%              All standard disclaimers apply.

% news (12/01/2005) both kind of block matrices possible.

function U = Blo2Walxxl(A,n,method);

if nargin < 1
    disp('At least one input (A) is needed');
    return;
elseif nargin < 3 | method ~= 1
    method = 0;
end;

[b a] = size(A);

if rem(n, lcm(a,b)) ~=0 ;
    sprintf(...
    'n has to be a multiple of the number of rows (a= %i) and columns (b= %i) of A'...
    ,a,b)
    sprintf('Setting it to the smallest common multiple bigger than n=%i',n)
    n = exp(floor(log(n)/log(mult))+1)
end;

U = zeros(n,n);
rr = 0:(b-1);

if method == 0
    M = n/b;
    for ii = 0:(n-1)
        p = mod(ii + rr*M,n)+1;
        U(p,ii+1) = A(mod(rr,b)+1,mod(ii,a)+1);
    end
else
   [a b] = size(A);
    M = n/b;
    for ii = 0:(n-1)
        p = mod(ii + rr*M,n)+1;
        U(ii+1,p) = A(mod(ii,a)+1,mod(rr,a)+1);
```

```
    end
end;
```

B.4.2.2 Full (walnut) matrix to block

```
% Create the block matrix our of the full Gabor-type matrix
% Wal2BloXXL.M   Gabor Frame Like Matrix
% by XXL, 22.06.2004
%
% This program creates the data block from a matrix
% expecting it to be a Walnut matrix, Gabor-like , meaning
% it is expected to have a-periodic side-diagonals and only
% every M=L/b the sidediagonal is not zero. Depending on the parameter
% method a b x a or a x b matrix is created.
%
% inputs : A ...... n x n Matrix with the non trivial data
%          a ...... first parameter (block-circulant,
%                   periodic side diagonals; in Gabor case time)
%          b ...... second parameter (diagonal blocks,
%                   non-zero side diagonals; in Gabor case frequency)
%          method . 0 - b x a block matrix (Default)
%                   1 - a x b block matrix
%
%  output: the block matrix with the "essential" data.
%
%  usage:  B = Wal2Blo(A,n)
%
%  notes : method == 0 corresponds to the algorithm first used and
%          implemented by Qiu and Feichtinger, where the block matrix is
%          the condensed left b x n submatrix of the frame operator
%          method == 1 uses the a x b matrix, which is derived from the
%          upper submatrix a x n submatrix
%
% Author:   XXL .. Peter Balazs, contact: a8927259@unet.univie.ac.at
%
% Literature : [Qiu] S. Qiu, H.G. Feichtinger, Gabor-Type Matrices And
%                    Discrete Huge Gabor Transforms
%              [Xxl] P. Balazs, Gabor frame multipliers with application to
%                    psychoacoustical masking
%                    (Ph.D. thesis, in preparation)
%
%  see also: blockxxl.m, blo2walxxl.m, blocknon.m
%
% Copyright :  (c) NUHAG, Faculty of Math., University of Vienna, AUSTRIA
%              http://nuhag.mat.uunivie.ac.at/
%              (c) Acoustics Research Institute, Austrian Academy of
%              Science
%              http://www.kfs.oeaw.ac.at
%
%              Permission is granted to modify and re-distribute this
%              code in any manner as long as this notice is preserved.
%              All standard disclaimers apply.

% news (12/01/2005) both kind of block matrices possible.


function U = Wal2Bloxxl(A,a,b,method);

if nargin < 3;
    disp('At least three inputs (A,a,b) are needed');
end;
```

```
if nargin < 4 | method ~= 1
    method = 0;
end

[n m] = size(A);
if n ~= m;
    disp('The matrix should be quadratic');
    return;
end;

T = n/a;
F = n/b;

mult = lcm(a,b);

if rem(n, mult) ~=0;
    disp(sprintf(...
    'The number of rows (a= %i) and columns (b= %i) of A have %s%i)',...
    '  to be divisors of the size of the Matrix (n=',a,b,n));
    return;
end;

if method == 0
    U = diag2row(A,F)
    U = U(:,1:a)
else
    U = diag2col(A,F)
    U = U(1:a,:)
end
```

B.4.2.3 Gabor Frame Matrix

```
% GABFRMATXXL.M   Gabor Frame Matrix
% by XXL, 13.10.2004
%
% based on GABFRMATRIX.M   Sigang Qiu  11.11.93,  generated by DIASUNS
%
% This algorithms creates the Gabor frame matrix with the window g,
% the time shift a and the frequency shift b. It is a very fast and
% efficient way of calculating the Gabor frame matrix.
%
% inputs : g .... the Gabor window (a vector of length n)
%          a .... time shift parameter (lattice parameter)
%          b .... frequency shift parameter (lattice parameter)
%
%  output: the n x n Gabor frame matrix .
%
%  usage:  S = Gabfrmatxxl(g,a,b)
%
%  see also: Blo2WalXXL.m, GabFrMatrix.m

% Example:
%         n = 144;
%         a = 9;
%         b = 12;
%         g = gaussnk(n);
%         S = gabfrmatxxl(g, a, b);
%
% Changes:
% this function now only calculates the non-zero-block-matrix, which then is
```

```
% used by Blo2WalXXL to create S.
%
% (Gabfrmatrix.m): This is the same version as gabgab with the block and
% rotrotm replaced  by the trick codes.
%
% XXL .. Peter Balazs, contact: a8927259@unet.univie.ac.at

function UU = gabfrmatXXL(g, a, b);

n = length(g);

% error checking not necessary is done in the next function
u = blockxxl(g,a,b);

UU = Blo2WalXXL(u,n);

% ---------------------------------
% profile on;
% profile report;
% figure(2);
% profile plot;
% profile off;

% OLD:
% T = n/a; F = n/b;
%if (rem(n, a) ~=0) |(rem(n, b) ~=0);
%disp('Check the size of the matrix, please'); return; end
%gg = [g, g];
% uu = zeros(a,b);
% for jj=1:a,
%   for rr = 1:b;
%      per1 = jj:a:(n+jj-1);
%      pp = jj + (rr - 1)*F;
%      per2 = pp:a:(n+pp-1);
%      uu(jj,rr) = gg(per1)*gg(per2)';
%    end
% end
```

B.4.2.4 blockxxl.m

```
function  M = blockxxl(g, a, b, gamma, method, thresh);
% Create the block (= correlation) matrix from a Gabor system
% BLOCKXXL
%
% Description : This creates the Walnut representation
%               of the Gabor frame operator associated with g and gamma.
% Author    : Peter Balazs (XXL)
%
%
% Input     :   g     = a vector representing the Gabor atom (= window)
%               a,b   = two integers representing the lattice constants
%                       (they must be dividers of n)
%               gamma = a vector representing a Gabor atom (optional)
%                       (Default: gamma = g)
%               method = 0 - b x a block matrix (Default)
%                        1 - a x b block matrix
%               thresh = threshold for ignoring values (and setting to
%                        zero) (Default: 10^{-10})
%
% Output      :   M=a matrix (size n*a)
% Usage       :   blockxxl(g,a,b[,gamma]);
```

```
%
% Notes :
% method == 0 corresponds to the algorithm first used and implemented by
% Qiu and Feichtinger, where the block matrix is the condensed left b x n
% submatrix of the frame operator
% method == 1 uses the a x b matrix, which is derived from the upper
% a x n submatrix
% So (for mtehod == 1) this function calculates the non-zero block, the
% a x b matrix M such that, M[i,j] = S[i,i+j*n/b], where S is the Gabor
% frame operator for g, gamma, a, b.
% Every Gabor frame matrix can be reduced to this block. See [Qiu].
%
% See also:  Blo2WalXXL.m which can be used to get the full Gabor matrix
%            blocknon.m (basic program by Qiu)
%
% Literature : [Gro] K. Grchenig, Foundations of Time-Frequency
%                    Representation
%              [Str] T. Strohmer,  Numerical algorithms for discrete Gabor
%                    expansions
%              [Qiu] S. Qiu, H.G. Feichtinger, Gabor-Type Matrices And
%                    Discrete Huge Gabor Transforms
%              [Xxl] P. Balazs, Gabor frame multipliers with application to
%                    psychoacoustical masking
%                    (Ph.D. thesis, in preparation)
%
% Copyright :  (c) NUHAG, Faculty of Math., University of Vienna, AUSTRIA
%              http://nuhag.mat.uunivie.ac.at/
%              (c) Acoustics Research Institute, Austrian Academy of
%              Science
%              http://www.kfs.oeaw.ac.at
%
%              Permission is granted to modify and re-distribute this
%              code in any manner as long as this notice is preserved.
%              All standard disclaimers apply.

% News      : (18/06/2004)
%             different synthese atom possible
%             comments
%             error checking
%             (28/10/2004)
%             some other small modifications
%             (11/01/205)
%             calculating either the a x b or the b x a matrix (which are
%             not simple transposes of each other)
%             V0.9

if nargin < 3;
     disp('3 inputs required.');
     return;
end;
n = length(g);

if nargin < 4;
     gamma = g;
else
     if n ~= length(gamma);
         disp('g and gamma must have the same size.');
         return;
     end;
end;

if nargin < 5 | method ~= 1
```

```
    method = 0;
end;
if nargin < 6
    thresh = 0.0000000001;
end;

F = n/b;  T = n/a;

if F ~= round(F)
    disp(sprintf('Input b = %i is not a divisor of the n = length of g = %i.',b,n));
    return;
end;
if T ~= round(T)
    disp(sprintf('Input a = %i is not a divisor of the n = length of g = %i.',b,n));
    return;
end;

gg = [g, g]; % periodization
gaga = [gamma, gamma];

if method == 0 % b x a block (like qiu)
    uu = zeros(b,a);
    for jj=1:a,
      per1 = jj:a:(n+jj-1);
      for rr = 1:b;
         pp = jj + (rr - 1)*F;
         per2 = pp:a:(n+pp-1);
         uu(rr,jj) = conj(gg(per2))*gaga(per1)';
         % This is (part of) the Walnut representation (see [Str]
         % This is (more or less) the correlation function G_{(r-1).n} (j)
         % (see [Gro]).
         % 'more or less' because it uses negative translation and circular rotation
         % see also [Xxl] for details
       end
    end
    M = zeros(b,a);
else % a x b block
    uu = zeros(a,b);
    for jj=1:a,
      per1 = jj:a:(n+jj-1);
      for rr = 1:b;
         pp = jj + (rr - 1)*F;
         per2 = pp:a:(n+pp-1);
         uu(jj,rr) = conj(gg(per1))*gaga(per2)';
       end
    end
    M = zeros(a,b);
end;

ndx = find(abs(uu) > thresh);
M(ndx) = F*uu(ndx);
```

B.4.3 Double Preconditioning For Gabor Frames

Included in this testfile, which was used for the numerical test in Section 3.4.3.7, is the code for single and double preconditioning (Lines 180-200).

```
% test double preconditioning
% by XXL (Peter Balazs)
% contact: xxl@kfs.oeaw.ac.at
% double preconditioning algorithm at lines 180-200
% for documentation see
% [xxl] Peter Balazs, 'Irregular And Regular Gabor Multipliers With
%       Application To Psychoacutsic Masking'
%
% Known Problems: Some Strings are repeated because of the usage of sprintf
% and %s. This is necessary for Layout of this help, but seems to make some
% problems.


iter0 = 10; % number of different lattice parameters
iter1 = 1; % number of different tests (= different windows) for every
           %set of parameter
t0 = cputime; % for measuring calculation time

if exist('circessbet') == 1
    % incorporate past experiment in this one
    if show == 1
        disp('Using old data to get bigger basis.');
    end;
    n0 = n0+iter0*iter1;
else
  if exist('show') ~= 1
        show = 0; % print output? Or be quite (==0)
        % use it outside to set it
    end;

  maxn  = 1000; % maximal signal length

    % initiate variables
  diagsmal = 0;
  noframe = 0;
  diagesssmal = 0;
  diagnonconv = 0;
  diagmat = 0;
  diagmaterr = 0;
  diagessbet = 0;
  rrd = maxn+1;
    rrde = maxn+1;
    rrnd = maxn+1;
  nodouble = 0;
    diagsupp = 0;
    doubleessbet = 0;
    essnodouble = 0;

  circsmal = 0;
  circesssmal = 0;
  circnonconv = 0;
  circmat = 0;
  circmaterr = 0;
  noiter = 0;
  circessbet = 0;
  rrc = maxn+1;
```

```
    rrce = maxn+1;
    n0 = iter0*iter1;

    methstr = ...
        'First choosing b, then a and supp. So condition on a and supp.';
    winstr = 'Hamming'; % change with window !!!!

end;

for jj = 1:iter0
     n = 1; %reset
     nAlph = 1;
     % make sure that we don't choose a prime number (or one)
     while isprime(n) | n == 1
         n = ceil(maxn*rand(1,1));
         % choose a random n in [1,maxn]
     end;

     alph = alldiv(n); % divisors or n
     nAlph = length(alph);

     % choose a support length (now done later)
     % supp = alph(ceil(nAlph*rand(1,1)));
     % supp = n;
     % nAlph = length(find(alph <= supp));
     % a <= nAlpg % else no frame is possible.
     % choose arbitrary lattice parameter

     b = alph(ceil(nAlph*rand(1,1)));
   M = n/b;

   % choose b such that a*b <= n, otherwise no frame is possible

     naaa = length(find(alph <= M));
   a = alph(ceil(naaa*rand(1,1)));
   red = n/(a*b); %redundancy
      N = n/a;

     % choose a support length
     % bigger than M such that we don't have an diagonal frame operator
     % alsu = alph(find(alph >= M));
     % bigger than a such that we don't have a frame
%     alsu = alph(find(alph > M));

     nsss = nAlph-naaa;
     if nsss == 0
         % supp <= M
         % see tschurtschenthaler or qiu
         % in this case it must be diagonal

         % We could not avoid it
         % either cound it
         if 1 == 1
            diagmat = diagmat+1;
            diagsupp = diagsupp+1;
            diagsmal = diagsmal+1;
            if show == 1
                disp('The operator is diagonal.');
            end;
        else
            n0 = n0-1;
        end
```

```
      % or just continue
         continue;
else
     supp = alph(naaa+ceil(nsss*rand(1,1)));
end;

 if show == 1
     disp(sprintf(...
         'n = %g, a = %g, b = %g, red = %g; supp = %g',n,a,b,red,supp));
 end;

 for ii = 1:iter1
     % use this inner loop only if the window can change for the
     % same n,a,b and supp for example with random windows.
     g = zeros(1,n);

     % (1) random window (white noise)
     w = rand(1,supp);
     % g(1:(floor(supp/2)+1))=w(ceil(supp/2):supp);
     % g(n-ceil(supp/2)+2:n)=w(1:(ceil(supp/2)-1));

     % (2) hanning window
     % w = hanning(supp+2);
     % g(1:(floor(supp/2)+1))=w(ceil(supp/2)+1:supp+1);
     % g(n-ceil(supp/2)+2:n)=w(2:ceil(supp/2));
     % winstr='Hanning'
     % change above !!!!

     % (2b) hamming window
     % winstr = 'Hamming';
     % change above
     % w = hamming(supp);
     g(1:(floor(supp/2)+1))=w(ceil(supp/2):supp);
     g(n-ceil(supp/2)+2:n)=w(1:ceil(supp/2)-1);
     %        g(M+2) = 1
     % in this case use iter1 = 1
     % blackman(n)
     % kaiser(n,6)

     % (3) full Gaussian
     % w = gaussnk(supp);
     % g(1:(floor(supp/2)))=w(1:floor(supp/2));
     % g(n-ceil(supp/2)+1:n)=w(floor(supp/2)+1:supp);
     % in this case use iter1 = 1

     %% (4) cut-off Gaussian
     % supp2 = ceil(maxn*rand(1,1));
     % w = gaussnk(supp2)
     % g(1:(floor(supp/2)+1))=w(supp2-ceil(supp/2):supp2); (??)
     % g(n-ceil(supp/2)+2:n)=w(1:(ceil(supp/2)-1));
     % in this case use iter1 = 1
     %% !!! Use Kaiser-Bessel with beta < 1 instead ?


     B = blockxxl(g,a,b); % b x a block matrix

     % mde = min(abs(B(1,:)));
     if min(abs(B(1,:))) == 0
         % disp(sprintf('Minimal diagonal entry = %g',mde));
         if show == 1
             disp(sprintf('No frame'));
         end;
```

```
    noframe = noframe+1;
    continue;
end;

% Id=eye(n);
Id=zeros(b,a);
Id(1,:)=ones(1,a);

% Single Preconditioning: diagonal
P1=zeros(b,a);
P1(1,:)=1./(B(1,:));
% Block-Matrix with inverse diagonal elements
D=blockm(B,P1,n);
% Preconditioning by multiplication of blocks
diagnorm = walnorm(D-Id);
% 'distance' of preconditioned matrix to the identiy

% Single Preconditioning: circulant
v=mean(B.');      % mean of rows of B
w=ifft(oneover(fft(v))); % deconvolution
P1=w.'*ones(1,a);
A=blockm(B,P1,n);
circnorm = walnorm(A-Id);

% Double Preconditioning
v=mean(D.');
w=ifft(oneover(fft(v)));
P1=w.'*ones(1,a);
C=blockm(D,P1,n);
doublenorm = walnorm(C-Id);

% full n x n matrix
% S = blo2walxxl(B,n);
% rS = rank(S);
rS = n;

if rS ~= n
    % n elements are linear dependent, only create a space
    % with dimension = rank, so cannot be a frame.
    noframe = noframe+1;
    if show == 1
        disp(sprintf('No frame'));
    end;
    continue;
else
    if show == 1
        disp(sprintf(...
         'Rank of S: %g, Diagonal Dominance Norm: %g, %s %g; %s : %g', ...
         '  Circulant Norm:', '  Double Preconditioning', ...
         rS, diagnorm, circnorm, doublenorm));
    end;
end;
if doublenorm >= 0.99
    if diagnorm >= 0.99 & circnorm >=0.99
        % to prevent calculation erros
        noiter = noiter+1;
        if show == 1
            disp(...
                'The system is a frame, but no iterative scheme converges.');
        end;
    else
        nodouble = nodouble+1;
```

```
            if show == 1
               disp('-----------------------');
               disp('!!! Bad Moon Rising !!!');
               disp('-----------------------');
            end;
                 aand = a;
                 bbnd = b;
                 ggnd = g;
                 rrnd = min(rrnd,red);
             %    return;
             % end if this bad case occurs
             if diagnorm <= 0.9 | circnorm <= 0.9
                 essnodouble = essnodouble+1;
             end;
        end;
    else
        if show == 1
           disp('Double Preconditioning is convergent!!');
        end;
        if diagnorm < doublenorm & diagnorm < circnorm
            diagsmal = diagsmal+1;
            if diagnorm <= 10*eps
                diagmat = diagmat+1;
                diagmaterr = max(diagmaterr,doublenorm);
            else
                relerr = diagnorm/doublenorm;
                %if relerr > 0.01
                if relerr < 0.1
                     diagessbet = diagessbet+1;
                    % if show == 1
                         disp('Diagnorm is essentially smaller');
                         %end;
                     aaed = a;
                     bbed = b;
                     gged = g;
                     rred = min(rrd,red);
                 end;
                 if relerr > diagesssmal
                    diagesssmal = relerr;
%                     if show == 1
%                         disp('Diagnorm is smaller');
%                     end;
%                     aad = a;
%                     bbd = b;
%                     ggd = g;
%                     rrd = min(rrd,red):
                end;
            end;
        elseif circnorm < doublenorm
            circsmal = circsmal+1;
            if circnorm >= 1
                circnonconv = circnonconv+1;
            elseif circnorm <= 10*eps
                circmat = circmat+1;
                circmaterr = max(circmaterr,doublenorm);
            else
              relerr = circnorm/doublenorm;
              %  if relerr > 0.01
              if relerr < 0.1
                    circessbet = circessbet+1;
                     %if show == 1
                         disp('Circnorm is essentially smaller');
```

```
                          %end;
                      aace = a;
                      bbce = b;
                      ggce = g;
                      rrce = min(rrd,red);
                 end;
                 if relerr > circesssmal
                     circesssmal = relerr;
    %                   aac = a
    %                   bbc = b
    %                   ggc = g
    %                   rrc = min(rrc,red)
                 end;
             end;
         elseif 10*doublenorm < diagnorm & 10*doublenorm < circnorm
                     doubleessbet = doubleessbet+1;
                      % if show == 1
                          disp(sprintf(...
                'Doublenorm is essentially smaller, a = %g, b=%g',a,b));
                          % end;
                      adde = a;
                      bdde = b;
                      gdde = g;
                      rdde = min(rrd,red);
         end;
    end;
    end;
end;
t1 = cputime;
disp(sprintf('time = %g s',t1-t0));
disp(methstr);
disp(sprintf('We have tested %g times using a %s window',n0,winstr));
disp(sprintf(...
    'In %g cases (%g percent) the Gabor system was no frame.',noframe,...
    noframe/n0*100));
disp(sprintf(...
    'In %g cases (%g percent) no norm was smaller than 1.',noiter,...
    noiter/n0*100));
disp(sprintf(...
    '%g times the diagonal norm was smaller than the double precond norm',...
    diagsmal));
disp(sprintf('1.) In %g cases the Frame Matrix is already diagonal', ...
    diagmat));
disp(sprintf(...
    '    and the difference is only due to precision errors (%g).',...
    diagmaterr));
disp(sprintf('    In %g cases this is due to the support property.',...
    diagsupp));
disp(sprintf(...
 '2.) the rest (%g cases): here the maximal difference was (relative) %g.',...
  diagsmal-diagmat,diagesssmal));
disp(sprintf('Only in %g cases the difference was bigger than 1 percent.',...
    diagessbet));
% if rrde ~= maxn+1
%     disp(sprintf(...
%    'The minimal redundancy for which the diagonal norm
%     was essential smaller was %g.',rrde));
% end;
% disp('-------------------------------------------------------------');
disp(sprintf(...
    '%g times the circulant norm was smaller than the double precond norm',...
    circsmal));
```

```
disp(sprintf(...
    '1.) In %g cases the Frame Matrix is already circulant',circmat));
disp(sprintf(...
    '    and the difference is only due to precision errors (%g).',...
    circmaterr));
disp(sprintf(...
    '2.) the rest (%g cases): here the maximal difference was (relative) %g.',...
    circsmal-circmat,circesssmal));
disp(sprintf(...
    'Only in %g cases the difference was bigger than 1 percent.',...
    circessbet));
% if rrc ~= maxn+1
%   disp(sprintf(...
%     'The minimal redundancy for the circulant case was %g, for which
%       the error was
%        essential in the circulant case was %g.',rrc,rrce));
%    if rrce ~= maxn+1
%         disp(sprintf('The minimal redundancy for which the error was essential
%        in the circulant case was %g.',rrce));
%    end;
% end;
if nodouble ~= 0
    disp(sprintf('*ATTENTION* There was %g case, %s',...
     'when a single preconditiner would have converged, but the double didnt',...
     nodouble));
    disp(sprintf('But only in %g cases, the smaller norm was < 0.9.',...
        essnodouble));
end;
disp(sprintf('Heureka! In %g cases the double preconditioning norm %s',...
    'was essentially smaller (factor:10!).',doubleessbet));
```

Bibliography

[1] S.S. Abeyseker, K.P. Padhi et al, *Investigation of different frequency estimation techniques using the phase vocoder*, The 2001 IEEE International Symposium on Circuits and Systems, ISCAS 2001, Volume: 2, pp. 265–268, Sydney (2001)

[2] A. Aldroubi and K. Gröchenig, *Non-uniform sampling and reconstruction in shift-invariant spaces,* SIAM Review 43, p. 585-620 (2001)

[3] J.B. Allen, *Short term spectral analysis, synthesis, and modification by discrete Fourier transform*, IEEE Transactions on Acoustics, Speech and Signal Processing, Vol. ASSP-25, No.3, pp. 235-238 (1977)

[4] O. Axelson and V. Barker, *Finite Element Solution of Boundary Value Problems, Theory and Computation*, Academic Press, Orlando, 1984

[5] M. J. Bastiaans, *Gabor's signal expansion of a signal into Gaussian elementary signals*, Proc. IEEE, 68(4), pp. 538-539 (1980)

[6] R. Balan, P.G. Casazza, D. Edidin, G. Kutyniok, *A new frame identity*, Submitted (2005)

[7] P. Balazs, *Polynome über Gruppen* (Polynomials Over Groups), Master Thesis Univ. Wien (2001)

[8] P. Balazs, A. Noll, *Masking filter, phase vocoder and* ST^X *- an introduction*, Poster SAMPTA03 (2003)

[9] P. Balazs, H. G. Feichtinger, M. Hampejs, G. Kracher, *Double preconditioning for Gabor frames*, submitted to IEEE Trans. Signal Processing (2005)

[10] P. Balazs, W. A. Deutsch, A. Noll, J. Rennison, J. White, *STx Programmer Guide*, Version: 3.6.2, http://www.kfs.oeaw.ac.at/software/stx_manual/programmerguide/Programmer%20Guide.pdf (2005)

[11] J. J. Benedetto, C. Heil, D. F. Walnut, *Gabor systems and the Balian-Low theroem*, Chapter 2 in [44]

[12] J. J. Benedetto, G. Pfander, *Frame expansions for Gabor multipliers*, Applied and Computational Harmonic Analysis, to appear (2005)

[13] I. Bogdanova, P. Vandergheynst, J. Antoine, L. Jacques, M. Morvidone M, *Stereographic Wavelet Frames on the Sphere*, Applied and Computational Harmonic Analysis (in press) (2005)

[14] H. Bölcskei, F. Hlawatsch, H.G. Feichtinger, *Frame-Theoretic Analysis of Oversampled Filter Banks*, IEEE Trans. Signal Processing, Vol. 46, No. 12, pp. 3256-3268 (1998)

[15] W. A. C. van den Brink, T. Houtgast, *Spectro-temporal integration in signal detection*, JASA 88, pp. 1703-1711 (1990)

[16] P. G. Casazza, *The Art of Frame Theory*, Taiwanese J. Math., vol.4, no. 2, pp. 129-202 (2000)

[17] P. G. Casazza, O. Christensen, *Gabor frame over irregular lattices*, Adv Comp. Math. 18, 329-344 (2003)

[18] T. Chan, *An optimal circulant preconditioner for Toeplitz systems*, SIAM J. Sci. Stat. Comput., vol 9, iss. 4, pp. 766-771 (1989)

[19] O. Christensen, *Frame decompositions in Hilbert Spaces*, PhD. thesis, Univ. Aarhus (1993)

[20] O. Christensen, *Operators with closed range, pseudo-inverses, and perturbation of frames for a subspace*, Canad. Math. Bull. Vol. 42, No. 1, pp. 37–45 (1999)

[21] O. Christensen, *Frames and pseudo-inverses*, J. Math. Anal. Appl. 195, No.2, pp 401-414 (1995)

[22] O. Christensen, *Frames, Riesz bases, and discrete Gabor/wavelet expansions*, Bull. Amer. Math. Soc. 38, pp. 273-291. (2001)

[23] O. Christensen, *An Introduction To Frames And Riesz Bases*, Birkhäuser Boston (2003)

[24] O. Christensen, private conversation (2004)

[25] E. Cordero, K. Gröchenig, *Necessary conditions for Schatten class localization operators*, preprint (2004)

[26] J. B. Conway, *A Course in Functional Analysis*, Springer New York (1990)

[27] J. W. Cooley, O. W. Tukey, *An Algorithm for the Machine Calculation of Complex Fourier Series.*, Math. Comput. 19, pp. 297-301, (1965)

[28] R. E. Crochiere, *A weighted overlap-add method of short-time Fourier analysis / synthesis*, IEEE Transactions on Acoustics, Speech and Signal Processing, Vol. ASSP-28, No.1, pp. 99-102, Feb. 1980 (1980)

[29] I. Daubechies, *Ten Lectures On Wavelets*, SIAM, Philadelphia, 1992.

[30] W. A. Deutsch, A. Noll, *Simulation auditorischer Signaltrennung in komplexen musikalischen Signalen durch Übermaskierung*, Fortschritte der Akustik DAGA'93, pp. 828 – 831 (1993)

[31] N. Delprat, B. Escudie, P. Guillemain, R. Kronland-Martinet, P. Tchamitchian, B. Torrésani, *Asymptotic wavelet and Gabor analysis: extraction of instantaneous frequencies*, IEEE Transactions on Information Theory, vol.38, no.2, pp. 644 – 664. (1992)

[32] M. Dolson, *The phase vocoder: a tutorial*, Computer Musical Journal, Vol. 10., No. 4, pp. 14 – 27 (1986)

[33] H. Dudley, *The Vocoder*, Bell Labs Rec., vol. 18, p122 -126 (1939); reprinted in [111]

[34] M. Dörfler, *Gabor Analysis For A Class Of Signals Called Music*, PhD thesis, Univ. Wien (2002)

[35] M. Dörfler, H. G. Feichtinger *Quilted Gabor families I: reduced multi-Gabor frames*, preprint (2003)

[36] R. J. Duffin, A. C. Schaeffer, *A Class of nonharmonic Fourier series*, Trans. Amer. Math. Soc., vol.72, pp. 341-366 (1952)

[37] G. Eckel, *Ein Modell der Mehrfachverdeckung für die Analyse musikalischer Schallsignale*, PhD Thesis, Univ. of Vienna (1989)

[38] Y. Eldar, H. Bölcskei, *Geometrically Uniform Frames*, IEEE Trans. Inform. Theory, vol. 49, pp. 993-1006, Apr. 2003. (2003)

[39] H. Fastl, *Beschreibung dynamischer Hörempfindungen anhand von Mithörschwellen-Mustern*, Habilitation TU München (1981)

[40] H. G. Feichtinger, K. Gröchenig, *Banach spaces related to integrable group representations and their atomic decompositions. I.*, J. Funct. Anal., 86(2), pp. 307 – 340, (1989)

[41] H. G. Feichtinger, *Wiener amalgams over Euclidean spaces and some of their applications*, in Function Spaces (Edwardsville, IL, 1990), volume 136 of Lecture Notes in Pure and Appl. Math., pages 123 – 137 (1991)

[42] H. G. Feichtinger, G. Zimmermann, *A Banach space of test functions for Gabor analysis*, Chapter 3 in [44]

[43] H.G. Feichtinger, W. Kozek, *Quantization of TF lattice-invariant operators on elementary LCA groups*, Chapter 7 in [44]

[44] H. G. Feichtinger, T. Strohmer, *Gabor Analysis and Algorithms - theory and Applications*, Birkhäuser Boston (1998)

[45] H. G. Feichtinger, *Spline-type spaces in Gabor analysis,*pp. 100 – 122 in Wavelet Analysis: Twenty Years' Developments (edited by D. X. Zhou), volume 1 of Ser. Anal., World Scientific Press, Singapore (2002)

[46] H. G. Feichtinger, *Modulation spaces of locally compact Abelian groups*, in R. Radha M. Krishna and S. Thangavelu, editors, *Proc. Internat. Conf. on Wavelets and Applications*, pages 1–56, Chennai, January 2002. Allied Publishers, New Delhi. (2003)

[47] H. G. Feichtinger, K. Nowak, *A first survey of Gabor multipliers*, Chapter 5 in [48]

[48] H. G. Feichtinger, T. Strohmer, *Advances in Gabor Analysis*, Birkhäuser Boston (2003)

[49] H. G. Feichtinger, *Gabor multipliers with varying lattices*, in Wavelets: Applications In Signal And Image Processing, Edited by Unser, Michael A.; Aldroubi, Akram; Laine, Andrew F., Proceedings of the SPIE, Volume 5207, pp. 381-392 (2003)

[50] H. G. Feichtinger, M. Hampjes, G. Kracher, *Approximation of matrices by Gabor multipliers*, IEEE Signal Procesing Letters Vol. 11, Issue 11, pp 883– 886 (2004)

[51] H. G. Feichtinger, N. Kaiblinger, *Varying the time-frequency lattice of Gabor frames*, Trans. Amer. Math. Soc. 356, no. 5, pp. 2001-2023 (2004)

[52] H.G. Feichtinger, T. Werther, *Robustness of minimal norm interpolation in Sobolev algebras*, in: Sampling, Wavelets, and Tomography, eds. J.J Benedetto and A. Zayed, Birkhäuser, pp. 83-113 (2004)

[53] H. G. Feichtinger, W. Sun, *Sufficient conditions for irregular Gabor frames*, preprint(2005) to appear in Advances in Computational Mathematics

[54] H. G. Feichtinger et al., *NuHAG Matlab Center*, http://gerda.univie.ac.at/NuHAG/matlab/ (2005)

[55] H. G. Feichtinger, F. Luef, T. Werther, *A guided tour from linear algebra to the foundations of Gabor analysis*, preprint (2005)

[56] J.L. Flanagan; R. M. Golden, *Phase Vocoder*, Bell Syst. Tech., vol. 45, pp. 1493 – 1509 (1966); reprinted in [111]

[57] G. Folland, *Harmonic Analysis in Phase Space*, Princeton University Press, Princeton, NJ (1989)

[58] M. Fornasier; K. Gröchenig, *Intrinsic Localization of Frames*, to appear in Constructive Approximation (2005)

[59] S. A. Gelfand, *Essentials Of Audiology*, Thieme New York (1997)

[60] D. Gabor, *Theory of communications*, J. IEE, 93 (III), pp. 429-457 (1946)

[61] B. Gold, C.M. Rader, *The channel vocoder*, IEEE Trans. Audio Electroacoust., vol. AU-15, pp. 148-161 (1967) ; reprinted in [111]

[62] K. Gröchenig, *Aspects of Gabor analysis on locally compact abellian groups*, Chapter 6 in [44]

[63] K. Gröchenig, *Foundations Of Time-Frequency Analysis*, Birkhäuser Boston (2001)

[64] K. Gröchenig, M. Leinert, *Wiener's lemma for twisted convolution and Gabor frames*, J. Amer. Math. Soc. 17, pp. p. 1-18 (2004)

[65] K. Gröchenig, C. Heil, *Counterexamples for boundedness of pseudodifferential operators*, Osaka J. Math., to appear (2005)

[66] A. Grossmann, M. Holschneider, R. Kronland-Martinet R, J. Morlet, *Detection of abrupt changes in sound signals with the help of wavelet transforms*, Advances in Electronic and Electron Physics, Supplement: Inverse Problems, 281-306. (1987)

[67] W.M. Hartmann, *Signals, Sounds, and Sensation*, Springer (1998)

[68] M. van der Heijden, A. Kohlrausch, *Using an excitation-pattern model to predict auditory masking*, Hear Res. 1994 Oct;80(1), pp. 38-52. (1994)

[69] H. Heuser, *Lehrbuch der Analysis 2*, Teubner Stuttgart (1991)

[70] F. Hlawatsch, G. Matz, *Linear time-frequency filters*, in Time-Frequency Signal Analysis and Processing: A Comprehensive Reference, B. Boashash, eds., pp. 466-475, Oxford (UK): Elsevier, (2003)

[71] L. Hörmander, *The Analysis Of Linear Partial Differential Operators*, Springer New York, 1985–1989

[72] F. Jaillet, B. Torrésani, *Time-frequency jigsaw puzzle: adaptive and multilayered Gabor expansions.* Preprint, submitted to IEEE Trans. Signal Processing. (2004)

[73] A.J.E.M. Janssen, *The Zak transform: a signal transform for sampled time-continous signals*, Philips J. Res., vol. 43, pp 23-69 (1988)

[74] A.J.E.M. Janssen, *Duality and biorthogonality for Weyl-Heisenberg frames*, J. Fourier Anal. Appl., 2(4), pp 403-436 (1995)

[75] Y. Katznelson, *An Introduction To Harmonic Analysis*, Dover Publications New York (1976)

[76] G. Köthe, *Topologische lineare Räume I*, Springer Berlin (1966)

[77] B. Laback, *Effekte der Simultanmaskierung auf die Musikperzeption bei sensorineuralen Hörschäden und ihre Anwendung für Signalverarbeitungsalgorithmen in Hörgeräten*, PhD Thesis, Universität Wien (1998)

[78] B. Laback, W. A. Deutsch, *Relevanz des spektralen Hintergrundes für die Wortdiskrimination in Störgeräusch*, Zeitschrift für Audiologie (Audiological Acoustics), Supplementum II, 47-49 (1999)

[79] C. Lanciani, *Auditory Perception And The MPEG Audio Standard*, report, Georgia Institute of Technology, Atlanta. (1995)

[80] R. Larsen, *An Introduction To The Theory of Multipliers*, Springer Berlin (1971)

[81] H. Lausch, W. Nöbauer, *Algebra Of Polynomials*, North-Holland Publishing, London (1973)

[82] Y. Liu, Y. Wang, *The uniformity of non-uniform Gabor bases*, Advances in Computational Mathematics, 18, pp. 345 – 355 (2003)

[83] C. F. Van Loan, *Computational Frameworks For The Fast Fourier Transform*, SIAM (1992)

[84] D. Luenberger, *Linear And Nonlinear Programming*, Addison-Wesley, Reading, 1984

[85] P. Majdak, P. Balazs, *Fast system identification with log sweeps and Gabor multipliers*, in preparation (2005)

[86] W. Mecklenbräucker, F. Hlawatsch (Hrsg.), *The Wigner Distribution - Theory and Applications in Signal Processing*, Amsterdam: Elsevier (1997)

[87] A. Meister, *Numerik linearer Gleichungssysteme*, Friedr. Vieweg & Sohn Braunschweig (1999)

[88] M. B. Monagan et al., *MAPLE 7 Programming Guide*, Waterloo Maple Inc. (2001)

[89] S. Molla, B. Torrsani, *Determining local transientness in audio signals*, IEEE Signal Processing Letters, 11:7, pp. 625-628 (2004)

[90] F.R. Moore, *Elements of Computer Music*, Prentice-Hall, Inc. (1990)

[91] B. Moore, *Perceptual Consequences of Cochlear Damage*, Oxford Medical Publications (1995)

[92] B. Moore, *An Introduction to the Psychology of Hearing*, Academic Press Limited London (1989)

[93] G. F. Margrave, P. C. Gibson, J. P. Grossman, D. C. Henley, V. Iliescu, M. P. Lamoureux, *The Gabor Transform, pseudodifferential operators, and seismic deconvolution*, Integreated Computer-Aided Engineering, 9, 1-13 (2004)

[94] A. Mors, W. A. Deutsch, H. Waubke, *Audibility of resolved tonal components of gear noise in passenger car cabins by trained listeners*, Fortschritte der Akustik, DAGA, Strassburg (2004)

[95] I. V. Nabelek, *Temporal summation of constant and gliding tones at masked auditory threshold*, JASA 64, p. 751-763 (1978)

[96] A. Noll, J. White, P. Balazs, W. A. Deutsch, *STX - Intelligent Sound Processing, Programmer's Reference*, http://www.kfs.oeaw.ac.at

[97] A. V. Oppenheim, R.W. Schafer, *Zeitdiskrete Signalverarbeitung*, 3.Auflage, Oldenbourg (1999)

[98] M.R. Portnoff, *Implementation of the digital phase vocoder using the fast Fourier transform*, IEEE Trans. Acoust., Speech, and Signal Processing, vol. ASSP-24, pp. 243-248 (1976); reprinted in [111]

[99] P. Prinz, *Theory And Algorithms For Discrete 1-Dimensional Gabor Frames*, master thesis, University of Vienna (1996)

[100] R. Pichevar, J. Rouat, C. Feldbauer, and G. Kubin, *A bio-inspired sound source separation technique based on a spiking neural network in combination with an enhanced analysis/synthesis filterbank*, EUSIPCO 2004, Vienna, Austria

[101] T. Rodionov, *Tight frames of Cauchy exponents* , Poster, Modern Methods of Time-Frequency Analysis, Strobl (2005)

[102] W. Rudin, *Functional Analysis*, Second Ed., McGraw Hill, 1991

[103] S. Qiu, H.G. Feichtinger, *Discrete Gabor structures and optimal representations*, IEEE Transactions on Signal Processing, Vol. 43, No. 10, pp. 2258 – 2268(1995)

[104] S. Qiu, H.G. Feichtinger, *Gabor-type matrices and discrete huge Gabor transforms*, ICASSP-Proceeding, Detroit, pp. 1089-1092 (1995)

[105] S. Qiu, *Generalized dual Gabor atoms and best approximations by Gabor family*, Signal Processing 49, pp. 167-186 (1996)

[106] S. Qiu, *Discrete Gabor transforms: the Gabor-Gram matrix approach*, The Journal of Fourier Analysis and Applications, pp. 1-17 Vol.4 (1) (1998)

[107] S. Qiu, *Gabor-type matrix algebra and fast computations of dual and tight Gabor wavelets*, Opt. Eng. 36(1) pp. 276-282 (1997)

[108] R. Reams, S. Waldron, *Isometric tight frames*, The Electronic Journal of Linear Algebra, vol. 9, 122-128 (2002)

[109] N. H. van Schijndel, T. Houtgast, J. M. Festen, *Intensity discrimination of Gaussian-windowed tones: indications for the shape of the auditory frequency-time window*, J. Acoust. Soc. Am. 1999 Jun; 105(6), pp. 3425-35 (1999)

[110] R. Schatten, *Norm Ideals of Completely Continious Operators*, Springer Berlin (1960)

[111] R.W. Schafer, J.D. Markel (editor), *Speech Analysis*, IEEE Press (1979)

[112] M. R. Schroeder, *Vocoders: analysis and synthesis of speech*, Proc. IEEE, vol. 54, pp. 720-734 (1966); reprinted in [111]

[113] Z. Semadeni, *Schauder Bases In Banach Spaces Of Continous Functions*, Springer Berlin (1982)

[114] I. Singer, *Bases In Banach Spaces I*, Springer Berlin (1970)

[115] I. Singer, *Bases In Banach Spaces II*, Springer Berlin (1981)

[116] J. Skoglund, W. B. Kleijn, *On time-frequency masking in voiced speech*, IEEE Transactions on Speech and Audio Processing, Volume: 8, Issue: 4, p. 361 – 369 (2000)

[117] J. O. Smith and J. Abel, *Bark and ERB bilinear transforms*, IEEE Trans. Sig. Proc. Vol.7 No.6, Nov. 1999, p. 697 – 708 (1999)

[118] P. Soendergaard, *LTFAT: A Matlab/C toolbox for linear time-frequency analysis*, preprint (2005)

[119] S. D. Stearns, D.R. Hush, *Digitale Verarbeitung analoger Signale* , 7. Auflage, Oldenbourg (1999)

[120] J. Y. Stein, *Digital Signal Processing*, John Wiley & Sons (2000)

[121] J. Stoer, *Introduction To Numerical Analysis*, Springer New York (2002)

[122] T. Strohmer, *Numerical algorithms for discrete Gabor expansions*, Chapter 8 in [44]

[123] L. N. Trefethen, D. Bau, III, *Numerical Linear Algebra*, Siam Philadelphia (1997)

[124] T. Tschurtschenthaler , *The Gabor Frame Operator (Its Structure And Numerical Consequences)*, Master thesis, University of Vienna (2000)

[125] R. Vafin, S. V. Andersen, W. B. Kleijn, *Exploiting time and frequency masking in consistent sinusoidal analysis-synthesis*, Acoustics, Speech, and Signal Processing, 2000. ICASSP '00 Proceedings. pp. II901 - II904 vol.2 (2000)

[126] J. S. Walker, *Fast Fourier Transforms*, CRC Press (1991)

[127] D. F. Walnut, *Continuity properties of the Gabor frame operator*, J. Math. Anal. Appl., 165(2), pp. 479-504 (1992)

[128] E.W. Weisstein et al., *Eric Weisstein's World of Mathematics*, http://mathworld.wolfram.com

[129] D. Werner, *Funktionalanalysis*, Springer Berlin (1995)

[130] T. Werther, Y.C.Eldar, N.K.Subbanna, *Dual Gabor frames: theory and computational aspects*, submitted to IEEE Transactions on Signal Processing

[131] J. Wexler, S. Raz, *Discrete Gabor expansion*, Signal Processing, vol. 21, pp. 207-220 (1990)

[132] P. J. Wolfe, M. Dörfler, S. J. Godsill, *Multi-Gabor dictionaries for audio time-frequency analysis*, Proceedings of the IEEE Workshop of Signal Processing to Audio and Acoustics, p. 43 – 46 (2001)

[133] Yau-Chuen Wong, *Schwartz Spaces, Nuclear Spaces and Tensor Products*, Lecture Notes in Mathematics, Springer Berlin (1979)

[134] K. Zhu, *Operator Theory In Function Spaces*, Marcel Dekker New York (1990)

[135] M. Zibulski, Y.Y. Zeevi, *Oversampling in the Gabor scheme*, IEEE Transactions on Signal Processing, Vol. 41, No. 8, pp. 2679 – 2687 (1993)

[136] M. Zibulski, Y.Y. Zeevi, *Frame analysis of the discrete Gabor scheme*, IEEE Transactions on Signal Processing, Vol. 42, No. 4, pp. 942 – 945 (1994)

[137] Zwicker E, Fastl H, *Psychoacoustics*, Springer-Verlag, Berlin, Germany, 1990

Index