Gabor Multipliers
A Self-Contained Survey

Diplomarbeit zur Erlangung
des akademischen Grades
Magistra der Naturwissenschaften

Eingereicht von
Karin Schnass

Betreut von
Prof. Dr. H. G. Feichtinger

Ausgeführt am
Institut für Mathematik der Universität Wien

Wien, 4.4.2004

Contents

List of Figures

Deutsche Zusammenfassung

Ziel dieser Arbeit ist es eine Einführung in das Gebiet der *Gabor Multiplier* zu geben. Diese Operatoren basieren auf der Zerlegung einer Funktion mit Hilfe eines *Gabor Frames*, das heißt eines Erzeugendensystems generiert durch zeitliche Verschiebung und gleichzeitige Frequenzmodulation einer Grundfunktion (*Gabor Atom*) entlang eines Gitters. Anstatt jedoch die Funktion nun mittels eines *dualen Frames* zu rekonstruieren werden die Koeffizienten zuerst mit einer Gewichtsfunktion multipliziert.
Kapitel 1 gibt einen Überblick über die Grundbegriffe und ein kurze Einführung in das Gebiet der Gabor Frames und Riesz Basen. Außerdem wird eine für den Rest der Arbeit wichtige Funktionenklasse vorgestellt.
In zweiten Kapitel werden Gabor Multiplier definiert und auf ihre Eigenschaften abhängig von den Bausteinen (Gitter, Atom, Gewichtsfunktion) untersucht. Dafür werden Operatorsymbole ähnlich eines Integrationkerns eingeführt und weiters wird auf eine alternative Interpretation eines Gabor Multipliers hingewiesen, nämlich als unendliche Linearkombination von Rang 1 Operatoren, die wieder durch Verschieben einer Grundfunktion entlang des Gitters entstehen. Schließlich wird ein Kriterium angegeben unter welchen Bedingungen an das Gitter und das Gabor Atom die darauf basierenden Rang 1 Operatoren eine Riesz Basis bilden und wie die biorthogonale Basis aussieht.
Das dritte Kapitel beschäftigt sich mit der numerischen Realisation von Gabor Multipliern. Zuerst werden kurz Verfahren zur Berechnung von dualen und *tight* Frames vorgestellt und dann Algorithmen zur effizienten Berechnung von Gabor Multipliern, usw. beschrieben und entwickelt. Mit Hilfe dieser Algorithmen werden dann die Ergebnisse von Kapitel 2 numerisch, in MATLAB, simuliert und überprüft.
Kapitel 4 greift zurück auf die Riesz Basis Eigenschaft des generierten *Gabor Systems* und die dadurch entstehende Möglichkeit die Bestapproximation eines allgemeinen Operators einfach zu berechnen. Die Unterschiede der Approximation in Systemen mit verschiedenen Gittern und Atomen werden analytisch und numerisch untersucht und es wird mittels numerischer Experimente versucht die optimale Abstimmung von Gitter und Gabor Atom festzustellen.
Der zweite Teil des Kapitels beschäftigt sich mit der Frage welche Operatoren gut durch Gabor Multiplier approximiert werden können. Es wird ein analytisches Kriterium angegeben und dadurch motiviert werden zwei Klassen von Operatoren, *Underspread Operators* und *STFT-Multipliers*, mittels numerischer Experimente auf ihre Approximierbarkeit untersucht. Das Hauptaugenmerk liegt dabei wieder auf der Abstimmung von Gitter und Gabor Atom und zwar auf den Operator.
Die Arbeit schließt mit dem Anwendungsbeispiel eines Gabor Multipliers als digitaler, zeitvarianter Filter.

Eine Online-Version dieser Arbeit ist auf der NuHAG Homepage unter
http://www.mat.univie.ac.at/∼nuhag/papers/thesis.html
zu finden.

Danke
allen, die mir während meines Studiums geholfen haben.

Chapter 1

Basics of Time Frequency Analysis

1.1 Basic Fourier Analysis & Tools

1. **Fourier Transform**:
   For a function $f \in \boldsymbol{L}^1(\mathbb{R}^d) \cap \boldsymbol{L}^2(\mathbb{R}^d)$ the Fourier transform $\mathcal{F}$ is defined by:

   $$(\mathcal{F}f)(\omega) = \int_{\mathbb{R}^d} f(x) e^{-2\pi i x\omega} dx$$

   $\mathcal{F}$ extends to a unitary operator on $\boldsymbol{L}^2(\mathbb{R}^d)$.

2. **Lattice**:
   A lattice Λ is a (discrete) subgroup of $\mathbb{R}^d$ of the form $\Lambda = A\mathbb{Z}^d$, where A is an invertible, real $d \times d$ matrix. A lattice of the form $\Lambda = a\mathbb{Z}^d \times b\mathbb{Z}^d$ for $a, b > 0$ is called a separable or product lattice. The dual lattice $\Lambda^\perp$ to a lattice $\Lambda = A\mathbb{Z}^d$ is defined as all $\lambda^\perp \in \mathbb{R}^d$ such that $e^{-2\pi i \lambda\lambda^\perp} = 1 \quad \forall \lambda \in \Lambda$. It can be calculated as $\Lambda^\perp = (A^T)^{-1}\mathbb{Z}^d$.

3. **Time-Frequency-Shifts**:

   (a) For $x, \omega \in \mathbb{R}^d$ the translation operator T_x and the modulation operator M_ω are defined by their actions on a function f as:

   $$\begin{aligned} \mathrm{T}_x f(t) &= f(t-x) \\ \mathrm{M}_\omega f(t) &= f(t) \cdot e^{2\pi i t\omega} \end{aligned}$$

   Operators of the form $\mathrm{M}_\omega \mathrm{T}_x$ are called time-frequency-shifts (TF-shifts), for $\lambda = (x, \omega)$, $\mathrm{M}_\omega \mathrm{T}_x$ is denoted by $\pi(\lambda)$.
   TF-shifts obey the commutation relation:

   $$\mathrm{T}_x \mathrm{M}_\omega = e^{-2\pi i x\omega} \mathrm{M}_\omega \mathrm{T}_x \tag{1.1}$$

   Note that $\pi^*(\lambda) = {\mathrm{T}_x}^* {\mathrm{M}_\omega}^* = \mathrm{T}_{-x}\, \mathrm{M}_{-\omega} = e^{-2\pi i x\omega} \pi(-\lambda)$.

   (b) The tensor product $(\pi \otimes \pi^*)(\lambda)$ of time-frequency shift operators is defined through its action on an operator K as:

   $$(\pi \otimes \pi^*)(\lambda)\mathrm{K} = \pi(\lambda) \circ \mathrm{K} \circ \pi^*(\lambda)$$

$(\pi \otimes \pi^*)(\lambda)$ will be abbreviated by $\pi_2(\lambda)$. A short computation shows that the action of $\pi_2(\lambda)$ on K corresponds to a TF-shift $\mathrm{M}_{(\omega,-\omega)}\,\mathrm{T}_{(x,x)}$ of the distributional kernel $\kappa(\mathrm{K})$ of the operator:

$$\begin{aligned}
(\pi_2(\lambda)\mathrm{K})f(t) &= \pi(\lambda)\int_{\mathbb{R}^d}\kappa(\mathrm{K})(s,t)(\pi^*(\lambda)f)(t)dt = \\
&= \pi(\lambda)\int_{\mathbb{R}^d}\kappa(\mathrm{K})(s,t)f(t+x)e^{-2\pi i\omega(t+x)}dt = \\
&\overset{t+x=r}{=} \int_{\mathbb{R}^d}\kappa(\mathrm{K})(s-x,r-x)e^{2\pi i\omega s}e^{-2\pi i\omega r}f(r)dr = \\
&= \int_{\mathbb{R}^d}[\mathrm{M}_{(\omega,-\omega)}\,\mathrm{T}_{(x,x)}\,\kappa(\mathrm{K})](s,r)f(r)dr \qquad (1.2)
\end{aligned}$$

4. **Λ-Fourier transform**:
For a sequence $c \in \boldsymbol{\ell}^2(\Lambda)$, $\Lambda = A\mathbb{Z}^d$, the Λ-Fourier transform $\mathcal{F}_\Lambda$ is defined as:

$$\mathcal{F}_\Lambda c(x) = \hat{c}(x) = \det(A)\sum_{\lambda\in\Lambda} c(\lambda)e^{2\pi i x\lambda}$$

It is relatively easy to see that $\mathcal{F}_\Lambda c$ is a periodic function on the torus $\mathbb{T}_{\Lambda^\perp} = \mathbb{R}^d/\Lambda^\perp$ and it holds that $\mathcal{F}_\Lambda : \boldsymbol{\ell}^2(\Lambda) \longrightarrow \boldsymbol{L}^2(\mathbb{T}_{\Lambda^\perp})$ is unitary.
For more information on the Λ-Fourier transform see for instance [19]. Note that for the case $\Lambda = \mathbb{Z}^d$ the setting is simply that of Fourier series, although unfortunately with reversed notation. In fact all results about the Λ-Fourier transform are easy consequences of the equivalent results for Fourier series.

5. **Convolution** over a lattice Λ, $*_\Lambda$:
The convolution of two sequences $(c(\lambda))_{\lambda\in\Lambda}, (d(\lambda))_{\lambda\in\Lambda}$ over the lattice Λ is defined as:

$$(c *_\Lambda d)(\mu) = \sum_{\lambda\in\Lambda} c(\lambda)d(\mu-\lambda)$$

Under the Λ-Fourier transform the convolution over Λ turns into point wise multiplication, i.e.,

$$\mathcal{F}_\Lambda(c *_\Lambda d) = \mathcal{F}_\Lambda c \cdot \mathcal{F}_\Lambda d = \hat{c}\cdot\hat{d}$$

6. **Short-Time Fourier Transform (STFT)**:
The short-time Fourier transform of a function $f \in \boldsymbol{L}^2(\mathbb{R}^d)$ with respect to a window function $g \in \boldsymbol{L}^2(\mathbb{R}^d)$ is defined as:

$$\begin{aligned}
\mathcal{V}_g f(x,\omega) &= \int_{\mathbb{R}^d} f(t)\overline{g(t-x)e^{2\pi i\omega t}}dt = \\
&= \langle f, \mathrm{M}_\omega \mathrm{T}_x g\rangle = \qquad (1.3)\\
&= \mathcal{F}(f\cdot \mathrm{T}_x\bar{g})(\omega) \qquad (1.4)
\end{aligned}$$

The STFT is a unitary mapping from $\boldsymbol{L}^2(\mathbb{R}^d)$ to $\boldsymbol{L}^2(\mathbb{R}^{2d})$ and so a function can be reconstructed from its STFT. Via the inner product representation it can be extended to a greater class of functions/distributions. So whenever g is in the Schwartz space, i.e., $g \in \mathcal{S}$, $\mathcal{V}_g f$ is well defined for all tempered distributions $f \in \mathcal{S}'$. More details about the STFT can be found in [14] Chapter 3.

7. **Wiener Amalgam Spaces**:
Wiener Amalgam spaces are a large class of function/distribution spaces that are characterized by having elements with a well controlled global and local behaviour. As here only two types are needed, we will not give the general definition as can be found in [4] but more straightforward ones. ($Q = [0,1]^d$)

$$\boldsymbol{W}(\boldsymbol{C}, \boldsymbol{\ell}^p)(\mathbb{R}^d) = \{f \in \boldsymbol{C}(\mathbb{R}^d) : \|f\|^p_{\boldsymbol{W}(\boldsymbol{C},\boldsymbol{\ell}^p)} = \sum_{j\in\mathbb{Z}^d} (\sup_{x\in j+Q} |f(x)|)^p < \infty\}$$
$$\boldsymbol{W}(\mathrm{M}, \boldsymbol{\ell}^p)(\mathbb{R}^d) = \{\mu \in \boldsymbol{C}_c(\mathbb{R}^d)' : \|\mu\|^p_{\boldsymbol{W}(\mathrm{M},\boldsymbol{\ell}^p)} = \sum_{j\in\mathbb{Z}^d} |\mu|(j+Q)^p < \infty\}$$

Wiener Amalgam spaces obey a number of convolution relations and norm estimates, of which however we will only need two types: ($K_{1,2}$ constants)

$$\boldsymbol{L}^p * \boldsymbol{W}(\boldsymbol{C}, \boldsymbol{\ell}^1) \subseteq \boldsymbol{W}(\boldsymbol{C}, \boldsymbol{\ell}^p) \qquad \|f * g\|_{\boldsymbol{W}(\boldsymbol{C},\boldsymbol{\ell}^p)} \leq K_1 \|f\|_p \|g\|_{\boldsymbol{W}(\boldsymbol{C},\boldsymbol{\ell}^1)} \quad (1.5)$$
$$\boldsymbol{W}(\mathrm{M}, \boldsymbol{\ell}^p) * \boldsymbol{W}(\boldsymbol{C}, \boldsymbol{\ell}^1) \subseteq \boldsymbol{L}^p \qquad \|f * g\|_p \leq K_2 \|f\|_{\boldsymbol{W}(\mathrm{M},\boldsymbol{\ell}^p)} \|g\|_{\boldsymbol{W}(\boldsymbol{C},\boldsymbol{\ell}^1)} \quad (1.6)$$

8. **Gelfand Triples**:
A (Banach) Gelfand triple $(\boldsymbol{B}, \boldsymbol{\mathcal{H}}, \boldsymbol{B}')$ is the combination of a Hilbert space $\boldsymbol{\mathcal{H}}$ with a Banach space $\boldsymbol{B}$ that is continuously and densely embedded into the Hilbert space, and its dual space $\boldsymbol{B}'$, into which in turn the Hilbert space is weak* continuously and densely embedded. A linear mapping between two Gelfand triples is called bounded if it is bounded on all three levels and preserves weak* convergence. Similarly a unitary Gelfand triple isomorphism is a mapping that is isomorphic on all three levels and additionally unitary on the Hilbert space level.
The main reason for introducing Gelfand triples is their practicability when trying to extend definitions or properties from a Hilbert space to a greater class of functions. As a general rule one simply has to replace all inner products by linear functionals, i.e., $\langle h, f\rangle = f(h) = h(f) \quad \forall h \in \boldsymbol{B}, \forall f \in \boldsymbol{B}'$, and by hand waving argue that everything will be fine. Conversely it is often easier to prove something just for $\boldsymbol{B}$ and then again by hand waving extend it to the whole Gelfand triple. For general information on Gelfand triples see [12] and "hand waving" is described in 7.3.3/4 in [8].

1.2 Riesz Bases and Frames

Definition 1.1 (Riesz Basis). *A family of functions $(g_i)_{i\in I}$ in a Hilbert space $\boldsymbol{\mathcal{H}}$ is a Riesz basis for its closed linear span if there exist constants $A, B > 0$ s.t. for all sequences $c \in \boldsymbol{\ell}(I)$:*

$$A\|c\|_2 \leq \|\sum_{i\in I} c_i g_i\|_{\boldsymbol{\mathcal{H}}} \leq B\|c\|_2 \quad (1.7)$$

A direct consequence of (1.7) is that every function f in the closed linear span of $(g_i)_{i\in I}$ has an expansion of the form $f = \sum_{i\in I} c_i g_i$ with uniquely determined coefficients $(c_i)_{i\in I}$. These coefficients can be calculated by taking inner products

with a biorthogonal basis $(\tilde{g}_i)_{i\in I}$, i.e., a basis such that $\langle g_i, \tilde{g}_j\rangle = \delta_{i,j}$ holds for all $i, j \in I$. Thus we have:

$$f = \sum_{i\in I} \langle f, \tilde{g}_i\rangle g_i$$

A consequence of the uniqueness of the coefficients is that assuming $(\tilde{g}_i)_{i\in I} \subseteq \overline{\text{span}(g_i)_{i\in I}}$ the biorthogonal basis is also uniquely determined.

Definition 1.2 (Frames). *A family $(f_j)_{j\in J}$ in a Hilbert space $\boldsymbol{\mathcal{H}}$ is called a frame if there exist constants $A, B > 0$ such that for all $f \in \boldsymbol{\mathcal{H}}$*

$$A\|f\|^2_{\boldsymbol{\mathcal{H}}} \leq \sum_{j\in J} |\langle f, f_j\rangle|^2 \leq B\|f\|^2_{\boldsymbol{\mathcal{H}}} \tag{1.8}$$

The constants A,B are called frame bounds. A frame is called tight if A=B.

For a family of functions $(f_j)_{j\in J}$, not necessarily but most of the times a frame, the analysis operator $\mathrm{C} : \boldsymbol{\mathcal{H}} \longrightarrow \boldsymbol{\ell}^2$ is given by

$$\mathrm{C}f = (\langle f, f_j\rangle)_{j\in J}$$

and the synthesis operator $\mathrm{D} : \boldsymbol{\ell}^2 \longrightarrow \boldsymbol{\mathcal{H}}$ is given by

$$\mathrm{D}c = \sum_{j\in J} c_j f_j$$

The combination of the analysis and synthesis operator $\mathrm{S} = \mathrm{DC}$ is called the frame operator. Note that as D and C are adjoint to each other, $\mathrm{D} = \mathrm{C}^*$, S is self-adjoint, i.e., $\mathrm{S} = \mathrm{C}^*\mathrm{C} = \mathrm{DD}^*$.
Now by reformulation of the frame-condition (1.8).

$$\langle \mathrm{S}f, f\rangle = \langle \sum_{j\in J} \langle f, f_j\rangle, f\rangle = \sum_{j\in J} |\langle f, f_j\rangle|^2$$
$$A\|f\|^2 \leq \langle \mathrm{S}f, f\rangle \leq B\|f\|^2$$

we see that it is equivalent to S being bounded and positive and thus invertible, with inverse S^{-1}. Consequently every $f \in \boldsymbol{\mathcal{H}}$ has expansions of the form:

$$\begin{aligned} f &= \mathrm{S}^{-1}\mathrm{S}f = \textstyle\sum_{j\in J} \langle f, f_j\rangle \mathrm{S}^{-1} f_j \\ f &= \mathrm{S}\mathrm{S}^{-1}f = \textstyle\sum_{j\in J} \langle \mathrm{S}^{-1}f, f_j\rangle f_j = \sum_{j\in J} \langle f, \mathrm{S}^{-1}f_j\rangle f_j \end{aligned} \tag{1.9}$$

The family $(\tilde{f}_j)_{j\in J} = (\mathrm{S}^{-1}f_j)_{j\in J}$ is again a frame with frame bounds B^{-1}, A^{-1}. However it is not the only **dual frame**, i.e., a frame $(g_j)_{j\in J}$ that allows an expansion of the form: $f = \sum_{j\in J}\langle f, g_j\rangle f_j$ but as it yields the coefficients $(\langle f, \tilde{f}_j\rangle)_{j\in J}$ with minimal $\|\cdot\|_2$ it is called the **canonical dual** frame.
Tight frames are of special interest, because for them, as can be seen from (1.8), the frame operator is a multiple of the identity operator. Therefore the dual frame is, apart from a constant, the same as the original and expansion (1.9) takes a more simplistic form.

$$f = \frac{1}{A}\mathrm{S}f = \frac{1}{A}\sum_{j\in J} \langle f, f_j\rangle f_j$$

Starting with an arbitrary frame there exists a nice trick to get a tight frame. Since S and thus S^{-1} are positive and self-adjoint, $\mathrm{S}^{\frac{1}{2}}$ and $\mathrm{S}^{-\frac{1}{2}}$ exist and are again positive and self-adjoint. From the manipulations

$$\begin{aligned} f = \mathrm{S}^{-\frac{1}{2}}\mathrm{S}\mathrm{S}^{-\frac{1}{2}} f &= \mathrm{S}^{-\frac{1}{2}} \sum_{j\in J} \langle f, \mathrm{S}^{-\frac{1}{2}} f\rangle f_j f_j \\ &= \sum_{j\in J} \langle f, \mathrm{S}^{-\frac{1}{2}} f_j\rangle \mathrm{S}^{-\frac{1}{2}} f_j \end{aligned} \tag{1.10}$$

we get that $(\mathrm{S}^{-\frac{1}{2}} f_j)_{j\in J}$ is a tight frame with frame bounds $A = B = 1$.
For a detailed description of the concepts outlined above see [1]. Both the concept of a Riesz basis and a frame can be generalized to Banach spaces, see e.g. [11].

1.3 Gabor Frames

A special type of frames on $\boldsymbol{L}^2(\mathbb{R}^d)$ are the Gabor or Weyl-Heisenberg frames, where the atoms f_j are shifted versions of one window function g, i.e.,

$$(g_\lambda)_{\lambda\in\Lambda} = (\pi(\lambda) g)_{\lambda\in\Lambda} \tag{1.11}$$

This adds a lot of structure. So the frame operator S commutes with TF-shifts:

$$\begin{aligned} \mathrm{T}_{-x}\,\mathrm{M}_{-\omega}\,\mathrm{S}\mathrm{M}_\omega \mathrm{T}_x f &= \sum_{(y,\xi)\in\Lambda} \langle \mathrm{M}_\omega \mathrm{T}_x f, \mathrm{M}_\xi\,\mathrm{T}_y\, g\rangle\, \mathrm{T}_{-x}\,\mathrm{M}_{-\omega}\,\mathrm{M}_\xi\,\mathrm{T}_y\, g = \\ &= \sum_{(y,\xi)\in\Lambda} \langle f, e^{-2\pi i(\xi-y)x}\,\mathrm{M}_{\xi-\omega}\,\mathrm{T}_{y-x}\, g\rangle e^{-2\pi i(\xi-y)x}\,\mathrm{M}_{\xi-\omega}\,\mathrm{T}_{y-x}\, g = \\ &= \sum_{(y',\xi')\in\Lambda} \langle f, \mathrm{M}_{\xi'}\,\mathrm{T}_{y'}\, g\rangle\, \mathrm{M}_{\xi'}\,\mathrm{T}_{y'}\, g = \mathrm{S} f \end{aligned}$$

Consequently S^{-1} and $\mathrm{S}^{-\frac{1}{2}}$ commute with TF-shifts and because

$$\mathrm{S}^{-1}(g_\lambda)_{\lambda\in\Lambda} = (\mathrm{S}^{-1}\pi(\lambda) g)_{\lambda\in\Lambda} = (\pi(\lambda)\mathrm{S}^{-1} g)_{\lambda\in\Lambda} = (\pi(\lambda)\tilde{g})_{\lambda\in\Lambda}$$

the dual frame $\mathrm{S}^{-1}(g_\lambda)_{\lambda\in\Lambda}$ and in analogy the canonical tight frame $\mathrm{S}^{-\frac{1}{2}}(g_\lambda)_{\lambda\in\Lambda}$ are again Gabor frames.
Also the frame operator itself has now many representations. One of them will be presented in Chapter 3, as it allows for efficient computations.
Among the remaining problems however is how to find window functions and lattices such that $(g_\lambda)_{\lambda\in\Lambda}$ constitutes a frame. This is quite delicate as there exist $\boldsymbol{L}^2$-functions so that for no lattice $\Lambda = a\mathbb{Z}^d \times b\mathbb{Z}^d$, $a, b > 0$, $(g_\lambda)_{\lambda\in\Lambda}$ forms a frame, see [14] p110. One way to avoid these complications is to introduce a new class of functions that are especially suitable as windows.

1.4 The Space $\boldsymbol{S}_0$

We next introduce a space of test functions which will play a role similar to the Schwartz space of rapidly decreasing functions. However while the Schwartz space,

being only a Frechet space, is quite awkward to deal with this new space is technically much more simple and in combination with $\boldsymbol{L}^2$ provides a very useful Banach Gelfand triple for our purposes.

Definition 1.3. *Let g be the normalized d-dimensional Gaussian $g(x) = e^{-\pi x^2}$, then the space $\boldsymbol{S}_0(\mathbb{R}^d)$ is defined as*

$$\boldsymbol{S}_0(\mathbb{R}^d) = \{f \in \mathcal{S}', \|\mathcal{V}_g f\|_1 < \infty\} \tag{1.12}$$

Remark 1.1. $\boldsymbol{S}_0$ belongs to a class of function spaces called modulation spaces and in that context is mostly denoted by $\boldsymbol{M}^1$. They were first introduced by H. Feichtinger and so $\boldsymbol{S}_0$ is also known as Feichtinger's algebra. An extensive study of modulation spaces in general can be found in [14], especially Chapters 11 and 12, and properties of $\boldsymbol{S}_0$ are discussed in [11].

Proposition 1.1. *Properties of $\boldsymbol{S}_0$*

(i) $\boldsymbol{S}_0(\mathbb{R}^d)$ is a Banach space with dual space:

$$\boldsymbol{S}_0'(\mathbb{R}^d) = \{f \in \mathcal{S}', \|\mathcal{V}_g f\|_\infty < \infty\}$$

and together $(\boldsymbol{S}_0, \boldsymbol{L}^2, \boldsymbol{S}_0')(\mathbb{R}^d)$ form a Gelfand triple.

(ii) By replacing the Gaussian in the definition of $\boldsymbol{S}_0$ with any other window in $\mathcal{S}$ and even $\boldsymbol{S}_0$ we get an equivalent norm.

(iii) If $f \in \boldsymbol{S}_0$ then $f, \hat{f} \in \boldsymbol{W}(\boldsymbol{C}, \boldsymbol{\ell}^1)$ and consequently f is continuous.

(iv) If f, g are in $\boldsymbol{S}_0(\mathbb{R}^d)$ then the STFT $\mathcal{V}_g f$ is even in $\boldsymbol{W}(\boldsymbol{C}, \boldsymbol{\ell}^1)(\mathbb{R}^{2d})$.

(v) The Fourier transform as well as all TF-shifts establish unitary Gelfand-Triple isomorphisms on $(\boldsymbol{S}_0, \boldsymbol{L}^2, \boldsymbol{S}_0')$. In particular, $\boldsymbol{S}_0(\mathbb{R}^d)$ is isometrically invariant under TF-shifts (and it is the smallest non-trivial Banach space with this property).

(vi) $\boldsymbol{S}_0(\mathbb{R}^d) \otimes \boldsymbol{S}_0(\mathbb{R}^d) \subseteq \boldsymbol{S}_0(\mathbb{R}^{2d})$

The proofs of (i-iv) need quite some theoretical background and can be found in one of the above mentioned books (e.g. [14] Theorems 11.3.5-7, 12.1.4/11). The proofs of (v) and (vi) are straight forward calculations that are good to get a feeling for $\boldsymbol{S}_0$.

Proof:
(v) We will just show that $\boldsymbol{S}_0(\mathbb{R}^d)$ is isometrically invariant under TF-shifts and the Fourier transform. So given $f \in \boldsymbol{S}_0(\mathbb{R}^d)$ we have:

$$\begin{aligned} \|\hat{f}\|_{\boldsymbol{S}_0} &= \int_{\mathbb{R}^{2d}} |\langle \hat{f}, \mathrm{M}_\omega \mathrm{T}_x g\rangle| d\omega dx = \int_{\mathbb{R}^{2d}} |\langle \hat{f}, \mathrm{M}_\omega \mathrm{T}_x \hat{g}\rangle| d\omega dx = \\ &= \int_{\mathbb{R}^{2d}} |\langle \hat{f}, \widehat{\mathrm{T}_{-\omega} \mathrm{M}_x}\, g\rangle| d\omega dx = \int_{\mathbb{R}^{2d}} |\langle f, \mathrm{T}_{-\omega} \mathrm{M}_x\, g\rangle| d\omega dx = \|f\|_{\boldsymbol{S}_0} \end{aligned}$$

$$\begin{aligned}\|\mathrm{M}_\omega \mathrm{T}_x f\|_{\boldsymbol{S}_0} &= \int_{\mathbb{R}^{2d}} |\langle \mathrm{M}_\omega \mathrm{T}_x f, \mathrm{M}_\xi \mathrm{T}_y g\rangle| d\xi dy = \int_{\mathbb{R}^{2d}} |\langle f, \mathrm{T}_{-x}\ \mathrm{M}_{-\omega} \mathrm{M}_\xi \mathrm{T}_y g\rangle| d\xi dy = \\ &= \int_{\mathbb{R}^{2d}} |\langle f, \mathrm{M}_{\xi-\omega} \mathrm{T}_{y-x} g\rangle| d\xi dy = \int_{\mathbb{R}^{2d}} |\langle f, \mathrm{M}_{\xi'} \mathrm{T}_{y'} g\rangle| d\xi' dy' = \|f\|_{\boldsymbol{S}_0}\end{aligned}$$

(vi) Note that the 2d-dimensional Gaussian is the tensor product of two d-dimensional Gaussian. Writing $\omega = (\omega_1, \omega_2)$, $x = (x_1, x_2)$, we also have $\mathrm{M}_\omega \mathrm{T}_x g = \mathrm{M}_{\omega_1} \mathrm{T}_{x_1} g \otimes \mathrm{M}_{\omega_2} \mathrm{T}_{x_2} g$ and thus:

$$\begin{aligned}\|f_1 \otimes f_2\|_{\boldsymbol{S}_0} &= \int_{\mathbb{R}^{4d}} |\langle f_1 \otimes f_2, \mathrm{M}_{\omega_1} \mathrm{T}_{x_1} g \otimes \mathrm{M}_{\omega_2} \mathrm{T}_{x_2} g\rangle| d\omega_1 d\omega_2 dx_1 dx_2 = \\ &= \int_{\mathbb{R}^{4d}} |\langle f_1, \mathrm{M}_{\omega_1} \mathrm{T}_{x_1} g\rangle| \cdot |\langle f_2, \mathrm{M}_{\omega_2} \mathrm{T}_{x_2} g\rangle| d\omega_1 d\omega_2 dx_1 dx_2 = \|f_1\|_{\boldsymbol{S}_0} \|f_2\|_{\boldsymbol{S}_0}\end{aligned}$$

□

As a bonus with $\boldsymbol{S}_0$ comes its dual space $\boldsymbol{S}_0'$ that is consequently also invariant under the Fourier transform and TF-shifts. Since $\boldsymbol{S}_0$-functions are continuous but need not be differentiable, functions in $\boldsymbol{S}_0'$ cannot be too "wild". However it still contains all locally integrable functions and the for signal analysis important pure frequencies $e^{2\pi i x\omega}$ as well as their Fourier transforms the δ-distributions.

Finally there are three theorems that should justify the introduction of $\boldsymbol{S}_0$ as they will make everything nice and easy.

Theorem 1.2. *For a Gabor family $(\pi(\lambda)g)_{\lambda\in\Lambda}$, $\Lambda = a\mathbb{Z}^d \times b\mathbb{Z}^d$, denote the synthesis operator by D_g and the analysis operator by C_g.*
If $g \in \boldsymbol{S}_0$ then C_g and D_g are bounded for all lattice constants $a, b > 0$, i.e.,

$$\begin{aligned}&\mathrm{C}_g : (\boldsymbol{S}_0, \boldsymbol{L}^2, \boldsymbol{S}_0') \longrightarrow (\boldsymbol{\ell}^1, \boldsymbol{\ell}^2, \boldsymbol{\ell}^\infty) \quad \textit{with} \quad \|\mathrm{C}_g\|_{\mathcal{OP}} \leq K_1 \|g\|_{\boldsymbol{S}_0} \textit{ and} \\ &\mathrm{D}_g : (\boldsymbol{\ell}^1, \boldsymbol{\ell}^2, \boldsymbol{\ell}^\infty) \longrightarrow (\boldsymbol{S}_0, \boldsymbol{L}^2, \boldsymbol{S}_0') \quad \textit{with} \quad \|\mathrm{D}_g\|_{\mathcal{OP}} \leq K_2 \|g\|_{\boldsymbol{S}_0}\end{aligned}$$

For $c \in (\boldsymbol{\ell}^1, \boldsymbol{\ell}^2))$ the sum $\mathrm{D}_g c = \sum_{\lambda\in\Lambda} c(\lambda)\pi(\lambda)g$ converges unconditionally in $(\boldsymbol{S}_0, \boldsymbol{L}^2)$ and for $c \in \boldsymbol{\ell}^\infty$ weak in $\boldsymbol{S}_0{}'$.*
(Proof in [14] Theorems 12.2.3/4)

This means essentially that whenever $g \in \boldsymbol{S}_0$ the resulting Gabor family has an upper frame bound and thus is a "Bessel sequence". The next theorem finally establishes the existence of Gabor frames.

Theorem 1.3. *Given a window $g \in \boldsymbol{S}_0$ there exist lattice constants $a_0, b_0 > 0$, such that for all lattices $\Lambda = a\mathbb{Z}^d \times b\mathbb{Z}^d$ with $a < a_0, b < b_0$ the generated Gabor family $(g_\lambda)_{\lambda\in\Lambda}$ constitutes a frame.*
Additionally the frame operator S is also invertible on $\boldsymbol{S}_0$. As a consequence the canonical dual atom $\tilde{g} = \mathrm{S}^{-1}g$ is again in $\boldsymbol{S}_0$. Further the operator $\mathrm{S}^{-\frac{1}{2}}$ maps $\boldsymbol{S}_0$ into $\boldsymbol{S}_0$ and thus also the associated tight atom $g_t = \mathrm{S}^{-\frac{1}{2}}g$ is in $\boldsymbol{S}_0$?
(Proof in [14] Theorem 6.5.1 and [15] Theorems 9 resp. 15)

Theorem 1.4. *Let $\mathcal{B}$ be the Banach space of bounded linear operators from $\boldsymbol{S}_0'(\mathbb{R}^d)$ to $\boldsymbol{S}_0(\mathbb{R}^d)$, that additionally map weak*-convergent sequences into $\boldsymbol{S}_0$-norm convergent sequences [1]. Then its dual space $\mathcal{B}'$ can be naturally identified with the space of*

[1] Actually, this is equivalent to the fact that w^*-convergent nets are mapped onto norm convergent nets in S_0.

bounded linear operators from $\boldsymbol{S}_0(\mathbb{R}^d)$ *to* $\boldsymbol{S}_0'(\mathbb{R}^d)$ *the duality pairing being the natural extension of the scalar product for Hilbert Schmidt operators.*
Together with the HS-operators $\mathcal{HS}$ *on* $\boldsymbol{L}^2(\mathbb{R}^d)$ *these Banach spaces form a Gelfand triple* $(\mathcal{B}, \mathcal{HS}, \mathcal{B}')$ *that is isomorphic to* $(\boldsymbol{S}_0, \boldsymbol{L}^2, \boldsymbol{S}_0')(\mathbb{R}^{2d})$*, i.e.,*

$$\mathrm{K} \in (\mathcal{B}, \mathcal{HS}, \mathcal{B}') \longleftrightarrow \kappa(\mathrm{K}) \in (\boldsymbol{S}_0, \boldsymbol{L}^2, \boldsymbol{S}_0')(\mathbb{R}^{2d})$$

(Proof in [8] Theorems 7.4.1/2)

Because every bounded linear operator T on $\boldsymbol{L}^2(\mathbb{R}^d)$ restricts to a bounded linear operator from $\boldsymbol{S}_0(\mathbb{R}^d)$ to $\boldsymbol{S}_0'(\mathbb{R}^d)$ we can associate a kernel $\kappa(\mathrm{T})$ to it. Hence ignoring the fact that an element of $\boldsymbol{S}_0'(\mathbb{R}^d)$ may actually be a true distribution we can think of T as an integration operator.

Chapter 2

Gabor Multipliers

As was shown in the first chapter Gabor frames provide a method of analysing and synthesising functions like for instance audio signals. So let's have a look at one possible application - wireless communication. The interesting part of the signal are

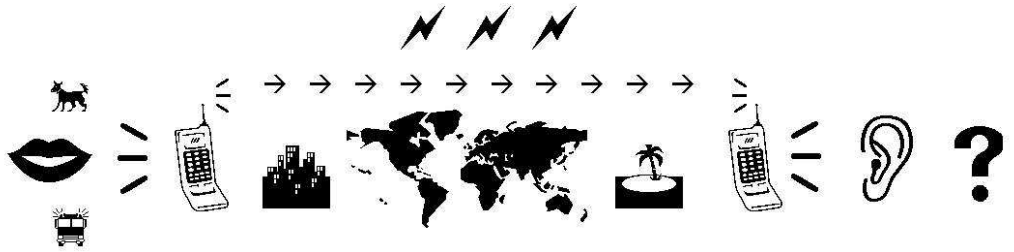


Figure 2.1: Communication Model

the spoken words but the received signal is a superposition of the voice, street noise and all kinds of transmission effects. So in order to filter away noise and correct the transmission irregularities it would be a good idea not to just re-synthesize with the received coefficients but to process them first. Mathematically this process can be described as:

$$f \longrightarrow \mathrm{G}(f)f = \sum_{\lambda\in\Lambda} \boldsymbol{m}(\lambda, f)\langle f, \pi(\lambda)g_1\rangle \pi(\lambda)g_2$$

or linearizing the dependence of G on f as:

$$f \longrightarrow \mathrm{G}f = \sum_{\lambda\in\Lambda} m(\lambda)\langle f, \pi(\lambda)g_1\rangle \pi(\lambda)g_2$$

This motivates the following definition and subsequent study of Gabor multipliers. All results in this chapter can be found in a condensed version in [9] or [3].

2.1 Definition & Basic Properties

Definition 2.1 (Gabor Multiplier). *Let g_1, g_2 be two functions in $\boldsymbol{L}^2(\mathbb{R}^d)$, Λ a time frequency lattice for $\mathbb{R}^d$ and $\boldsymbol{m} = (m(\lambda))_{\lambda\in\Lambda}$ a complex valued sequence on*

Λ. *The Gabor multiplier associated to the triple* (g_1, g_2, Λ) *with multiplier or upper symbol* $\boldsymbol{m}$ *is given by:*

$$\mathrm{G}_{\boldsymbol{m}}(f) = \mathrm{G}_{g_1,g_2,\Lambda,\boldsymbol{m}}(f) = \sum_{\lambda\in\Lambda} m(\lambda)\langle f, \pi(\lambda)g_1\rangle \pi(\lambda)g_2 \tag{2.1}$$

The above definition is very general and so it is necessary to check under which conditions on the windows g_1, g_2, the lattice Λ and the upper symbol $\boldsymbol{m}$ the resulting operator is well defined.
Let M denote the operator that acts on sequences $c = (c(\lambda))_{\lambda\in\Lambda}$ by point-wise multiplication i.e., $\mathrm{M}\, c = (m(\lambda)c(\lambda))_{\lambda\in\Lambda}$. Now the Gabor multiplier $\mathrm{G}_{g_1,g_2,\Lambda,\boldsymbol{m}}$ can be written as a combination of three already known operators.

$$\mathrm{G}_{g_1,g_2,\Lambda,\boldsymbol{m}} = \mathrm{D}_{g_2} \circ \mathrm{M} \circ \mathrm{C}_{g_1}$$

One way to ensure that G is bounded is to assume that $\mathrm{C}_{g_1}, \mathrm{D}_{g_2}$ and M are bounded. For M this means that the multiplying sequence $\boldsymbol{m}$ should be bounded i.e., $\boldsymbol{m} \in \boldsymbol{\ell}^\infty(\Lambda)$ and for $\mathrm{C}_{g_1}, \mathrm{D}_{g_2}$ that g_1, g_2 should be Bessel atoms with respect to the lattice Λ. This motivates the choice $g_1, g_2 \in \boldsymbol{S}_0(\mathbb{R}^d)$ and $\Lambda = a\mathbb{Z}^d \times b\mathbb{Z}^d$ with $a, b > 0$, a separable lattice, as every $g \in \boldsymbol{S}_0(\mathbb{R}^d)$, $g \neq 0$, is a Bessel atom for any separable lattice, independent of the lattice constants a, b.
The following theorem is a summary of these considerations.

Theorem 2.1. *Let* $g_1, g_2 \in \boldsymbol{S}_0(\mathbb{R}^d)$, $\Lambda = a\mathbb{Z}^d \times b\mathbb{Z}^d$.

(i) If $\boldsymbol{m} \in \boldsymbol{\ell}^\infty(\Lambda)$ *then* $\mathrm{G}_{\boldsymbol{m}} = \mathrm{G}_{g_1,g_2,\Lambda,\boldsymbol{m}}$ *defines a bounded operator on* $(\boldsymbol{S}_0, \boldsymbol{L}^2, \boldsymbol{S}_0')(\mathbb{R}^d)$, *with operator norm* $\|\mathrm{G}_{\boldsymbol{m}}\|_{\mathcal{OP}} \leq C\|\boldsymbol{m}\|_\infty$

(ii) If $\boldsymbol{m} \in \boldsymbol{\ell}^2(\Lambda)$ *then* $\mathrm{G}_{\boldsymbol{m}}$ *maps* $\boldsymbol{S}_0'(\mathbb{R}^d)$ *into* $\boldsymbol{L}^2(\mathbb{R}^d)$ *and* $\boldsymbol{L}^2(\mathbb{R}^d)$ *into* $\boldsymbol{S}_0(\mathbb{R}^d)$ *i.e.,* $\mathrm{G}_{\boldsymbol{m}} : \boldsymbol{S}_0'(\mathbb{R}^d) \longrightarrow \boldsymbol{L}^2(\mathbb{R}^d), \mathrm{G}_{\boldsymbol{m}} : \boldsymbol{L}^2(\mathbb{R}^d) \longrightarrow \boldsymbol{S}_0(\mathbb{R}^d)$

(iii) If $\boldsymbol{m} \in \boldsymbol{\ell}^1(\Lambda)$ *then* $\mathrm{G}_{\boldsymbol{m}}$ *maps* $\boldsymbol{S}_0'(\mathbb{R}^d)$ *into* $\boldsymbol{S}_0(\mathbb{R}^d)$ *i.e.,* $\mathrm{G}_{\boldsymbol{m}} : \boldsymbol{S}_0'(\mathbb{R}^d) \longrightarrow \boldsymbol{S}_0(\mathbb{R}^d)$

Proof:
Norm estimates for the pointwise product of two sequences of the form $\|cd\|_p \leq \|c\|_\infty \|d\|_p$ and $\|cd\|_1 = \langle |c|, |d|\rangle \leq \|c\|_2 \|d\|_2$, $|c| = |c(\lambda)|_{\lambda\in\Lambda}$, lead to the following mapping properties for M:

$$\begin{aligned}
\boldsymbol{m} \in \boldsymbol{\ell}^\infty(\Lambda) \Rightarrow \quad & \mathrm{M} : (\boldsymbol{\ell}^1, \boldsymbol{\ell}^2, \boldsymbol{\ell}^\infty) \longrightarrow (\boldsymbol{\ell}^1, \boldsymbol{\ell}^2, \boldsymbol{\ell}^\infty) \quad \text{with } \|\mathrm{M}\|_{\mathcal{OP}} \leq \|\boldsymbol{m}\|_\infty \\
\boldsymbol{m} \in \boldsymbol{\ell}^2(\Lambda) \Rightarrow \quad & \mathrm{M} : (\boldsymbol{\ell}^2, \boldsymbol{\ell}^\infty) \longrightarrow (\boldsymbol{\ell}^1, \boldsymbol{\ell}^2) \quad \text{with } \|\mathrm{M}\|_{\mathcal{OP}} \leq \|\boldsymbol{m}\|_2 \\
\boldsymbol{m} \in \boldsymbol{\ell}^1(\Lambda) \Rightarrow \quad & \mathrm{M} : \boldsymbol{\ell}^\infty \longrightarrow \boldsymbol{\ell}^1 \quad \text{with } \|\mathrm{M}\|_{\mathcal{OP}} \leq \|\boldsymbol{m}\|_1
\end{aligned}$$

The rest is a consequence of Theorem 1.2.

$$\begin{aligned}
\|\mathrm{G}_{\boldsymbol{m}}\|_{\mathcal{OP}} &= \|\mathrm{D}_{g_2}\, \mathrm{M}\, \mathrm{C}_{g_1}\|_{\mathcal{OP}} \leq \\
&\leq \|\mathrm{D}_{g_2}\|_{\mathcal{OP}} \|\mathrm{M}\|_{\mathcal{OP}} \|\mathrm{C}_{g_1}\|_{\mathcal{OP}} \leq C\|g_1\|_{\boldsymbol{S}_0}\|g_2\|_{\boldsymbol{S}_0}\|\boldsymbol{m}\|_{(1,2,\infty)} < \infty
\end{aligned}$$

□

Of course it is now possible to have a look at what happens with $\boldsymbol{L}^2$-windows and $\boldsymbol{\ell}^p$-multipliers as is for instance done in Section 3 of [9]. However for our needs the most important output is that Gabor multipliers with $\boldsymbol{\ell}^\infty$-multipliers and $\boldsymbol{S}_0$-windows are well behaved on $\boldsymbol{L}^2$ and that there is a certain stability in the dependence on the building blocks. Namely:

Corollary 2.2. *The Gabor multiplier $\mathrm{G}_{g_1,g_2,\Lambda,\boldsymbol{m}}$ depends continuously on the multiplying sequence $\boldsymbol{m}$, the analysis and the synthesis windows g_1, g_2, i.e., small changes of the multiplier, measured in the $(1,2,\infty)$-norm, or of the windows, measured in the $\boldsymbol{S}_0$-norm, cause only small changes of the Gabor multiplier, measured in the appropriate operator norms.*

Proof:
The statement is a direct consequence of the estimates in the proof of Theorem 2.1 and the triangular equation, i.e., :

$$\begin{aligned}&\|\mathrm{G}_{m,g_1,g_2,\Lambda} - \mathrm{G}_{\tilde{m},\tilde{g}_1,\tilde{g}_2,\Lambda}\|_{\mathcal{OP}} \leq \\ &\quad \leq \|\mathrm{G}_{m-\tilde{m},g_1,g_2,\Lambda}\|_{\mathcal{OP}} + \|\mathrm{G}_{\tilde{m},g_1-\tilde{g}_1,g_2,\Lambda}\|_{\mathcal{OP}} + \|\mathrm{G}_{\tilde{m},\tilde{g}_1,g_2-\tilde{g}_2,\Lambda}\|_{\mathcal{OP}} \leq \\ &\quad \leq C_1\|m-\tilde{m}\|_{(1,2,\infty)} + C_2\|g_1-\tilde{g}_1\|_{\boldsymbol{S}_0} + C_3\|g_2-\tilde{g}_2\|_{\boldsymbol{S}_0} \leq C_4\varepsilon\end{aligned}$$

□

Remark 2.1. As a conclusion to Theorem 2.1 for the rest of the thesis it will always be assumed that the windows g_1, g_2 are in $\boldsymbol{S}_0$ and that Λ is a separable lattice. However most results are also valid for the arbitrary lattices.

An alternative and very fruitful way to look at the Gabor multiplier $\mathrm{G}_{g_1,g_2,\Lambda,\boldsymbol{m}}$ is to interpret it as a sum of rank 1 operators P_λ.

$$\begin{aligned}\mathrm{G}_{g_1,g_2,\Lambda,\boldsymbol{m}}f &= \sum_{\lambda\in\Lambda} m(\lambda)\langle f,\pi(\lambda)g_1\rangle\pi(\lambda)g_2 = \sum_{\lambda\in\Lambda} m(\lambda)\underbrace{[(\pi(\lambda)g_1)\otimes(\pi(\lambda)g_2)^*]}_{\mathrm{P}_\lambda} f \\ \mathrm{G}_{g_1,g_2,\Lambda,\boldsymbol{m}} &= \sum_{\lambda\in\Lambda} m(\lambda)\mathrm{P}_\lambda = \sum_{\lambda\in\Lambda} m(\lambda)\pi_2(\lambda)\mathrm{P}_0 \qquad (2.2)\end{aligned}$$

Exploiting this representation - the validity of the second part of (2.2) will be shown in the proof of the following theorem - and the mapping properties of D_g for $\boldsymbol{S}_0$-windows again leads to a nice and useful result about the distributional kernels of Gabor multipliers.

Theorem 2.3. *For two windows $g_1, g_2 \in \boldsymbol{S}_0(\mathbb{R}^d)$ and any separable lattice Λ the Gabor multiplier $\mathrm{G}_{g_1,g_2,\Lambda,\boldsymbol{m}}$ has distributional kernel $\kappa(\mathrm{G}_{\boldsymbol{m}})$ in $(\boldsymbol{S}_0,\boldsymbol{L}^2,\boldsymbol{S}_0')(\mathbb{R}^{2d})$ whenever the upper symbol $\boldsymbol{m}$ is in $(\boldsymbol{\ell}^1,\boldsymbol{\ell}^2,\boldsymbol{\ell}^\infty)$.*

Proof:
P_0 obviously has the integration kernel $\kappa(\mathrm{P}_0) = g = g_1\otimes g_2^* \in \boldsymbol{S}_0(\mathbb{R}^{2d})$. P_λ on the other hand can be written as:

$$\mathrm{P}_\lambda = (\pi(\lambda)g_1)\otimes(\pi(\lambda)g_2)^* = \pi(\lambda)(g_1\otimes g_2^*)\pi^*(\lambda) = \pi_2(\lambda)\mathrm{P}_0$$

So remembering (1.2) its kernel $\kappa(\mathrm{P}_\lambda)$ is just a shifted version of $\kappa(\mathrm{P}_0)$.

$$\kappa(\mathrm{P}_\lambda) = \mathrm{M}_{(\omega,-\omega)}\,\mathrm{T}_{(x,x)}\,\kappa(\mathrm{P}_0)$$

If Λ is the original lattice $\Lambda = a\mathbb{Z}^d \times b\mathbb{Z}^d$ define $\Lambda_2 = a\mathbb{Z}^{2d} \times b\mathbb{Z}^{2d}$ and $\tilde{\boldsymbol{m}}$ on Λ_2 by

$$\tilde{\boldsymbol{m}}(x,x,\omega,-\omega) = \begin{cases} \boldsymbol{m}(x,\omega) = \boldsymbol{m}(\lambda) & \text{for } \lambda\in\Lambda \\ 0 & \text{else} \end{cases}$$

Clearly $\tilde{\boldsymbol{m}} \in (\boldsymbol{\ell}^1, \boldsymbol{\ell}^2, \boldsymbol{\ell}^\infty)(\Lambda_2)$ whenever $\boldsymbol{m} \in (\boldsymbol{\ell}^1, \boldsymbol{\ell}^2, \boldsymbol{\ell}^\infty)(\Lambda)$. Now $\kappa(\mathrm{G}_{\boldsymbol{m}})$ can be written as:

$$\begin{aligned} \kappa(\mathrm{G}_{\boldsymbol{m}}) &= \sum_{\lambda\in\Lambda} m(\lambda)\kappa(\mathrm{P}_\lambda) = \\ &= \sum_{(x,x,\omega,-\omega)\in\Lambda_2} \tilde{\boldsymbol{m}}(x,x,\omega,-\omega)\,\mathrm{M}_{(\omega,-\omega)}\,\mathrm{T}_{(x,x)}\,\kappa(\mathrm{P}_0) = \\ &= \sum_{\lambda_2\in\Lambda_2} \tilde{\boldsymbol{m}}(\lambda_2)\pi(\lambda_2)\kappa(\mathrm{P}_0) = \mathrm{D}_{g,\Lambda_2}\tilde{\boldsymbol{m}} \end{aligned}$$

Applying Theorem 1.2 for $g = g_1 \otimes g_2^* \in \boldsymbol{S}_0(\mathbb{R}^{2d})$ and Λ_2 yields that for $\tilde{\boldsymbol{m}} \in (\boldsymbol{\ell}^1, \boldsymbol{\ell}^2, \boldsymbol{\ell}^\infty)(\Lambda_2)$ the kernel $\kappa(\mathrm{G}_{\boldsymbol{m}})$ is a function/distribution in $(\boldsymbol{S}_0, \boldsymbol{L}^2, \boldsymbol{S}_0')(\mathbb{R}^{2d})$ and that $\|\kappa(\mathrm{G}_{\boldsymbol{m}})\|_{(\boldsymbol{S}_0,\boldsymbol{L}^2,\boldsymbol{S}_0')} \le C\|g\|_{\boldsymbol{S}_0}\|\boldsymbol{m}\|_{1,2,\infty}$. □

Remark 2.2. Actually Theorem 2.3 could have been directly deduced from Theorem 2.1 and Theorem 1.4, i.e., for $\boldsymbol{m} \in \boldsymbol{\ell}^\infty$ $\mathrm{G}_{\boldsymbol{m}}$ maps $\boldsymbol{S}_0$ into $\boldsymbol{S}_0 \subset \boldsymbol{S}_0'$ and therefore it has to have an $\boldsymbol{S}_0'$ integration kernel. The nice thing about the proof is that it simulates the proof of Theorem 1.4 and gives the idea that Gabor multipliers are operators that in some way live only on a diagonal. For another idea in that direction see the proof of Theorem 4.3 in Chapter 4.
A compact formulation of Theorem 2.3 using the language of Gelfand triples is that the mapping α from upper symbol to Gabor multiplier is bounded linear from $(\boldsymbol{\ell}^1, \boldsymbol{\ell}^2, \boldsymbol{\ell}^\infty)$ to $(\mathcal{B}, \mathcal{HS}, \mathcal{B}')$.

From now on the focus will be mainly on Gabor multipliers as operators on the Hilbert space level $\boldsymbol{L}^2(\mathbb{R}^d)$.

Theorem 2.4. *Let $g_1, g_2 \in \boldsymbol{S}_0(\mathbb{R}^d)$:*

(i) If $g_1 = g_2 = g$ and $\boldsymbol{m}$ is real-valued then $\mathrm{G}_{g,\Lambda,\boldsymbol{m}}$ is self-adjoint on $\boldsymbol{L}^2(\mathbb{R}^d)$.

(ii) If $\boldsymbol{m} \in \boldsymbol{c}_0(\Lambda)$ then $\mathrm{G}_{g_1,g_2,\Lambda,\boldsymbol{m}}$ is compact on $\boldsymbol{L}^2(\mathbb{R}^d)$.

(iii) If $\boldsymbol{m} \in \boldsymbol{\ell}^2(\Lambda)$ then $\mathrm{G}_{g_1,g_2,\Lambda,\boldsymbol{m}}$ is a Hilbert Schmidt operator on $\boldsymbol{L}^2(\mathbb{R}^d)$.

(iv) If $\boldsymbol{m} \in \boldsymbol{\ell}^1(\Lambda)$ then $\mathrm{G}_{g_1,g_2,\Lambda,\boldsymbol{m}}$ is a trace class operator on $\boldsymbol{L}^2(\mathbb{R}^d)$.

Proof:

(i) Let's calculate $\mathrm{G}_{\boldsymbol{m}}^*$ first, write $\pi(\lambda)g = g_\lambda$:

$$\begin{aligned}\langle \mathrm{G}_{\boldsymbol{m}}^* f, h\rangle &= \langle f, \mathrm{G}_{\boldsymbol{m}} h\rangle = \langle f, \sum_{\lambda\in\Lambda} m(\lambda)\langle h, g_\lambda\rangle g_\lambda\rangle = \sum_{\lambda\in\Lambda} \overline{m(\lambda)\langle h, g_\lambda\rangle}\langle f, g_\lambda\rangle = \\ &= \sum_{\lambda\in\Lambda} \overline{m(\lambda)}\langle f, g_\lambda\rangle\langle g_\lambda, h\rangle = \langle \sum_{\lambda\in\Lambda} \overline{m(\lambda)}\langle f, g_\lambda\rangle g_\lambda, h\rangle \\ \mathrm{G}_{\boldsymbol{m}}^* f &= \sum_{\lambda\in\Lambda} \overline{m(\lambda)}\langle f, g_\lambda\rangle g_\lambda\end{aligned}$$

Now it is easy to see that when $\boldsymbol{m}$ is real-valued $\mathrm{G}_{\boldsymbol{m}}$ is self adjoint.

(ii) since the finite sequences are dense in $\boldsymbol{c}_0(\Lambda)$ choose a finite sequence $\boldsymbol{m}_\varepsilon$ s.t. $\|\boldsymbol{m} - \boldsymbol{m}_\varepsilon\|_\infty \leq \varepsilon$. Now by

$$\|\mathrm{G}_{\boldsymbol{m}} - \mathrm{G}_{\boldsymbol{m}_\varepsilon}\|_{\mathcal{OP}} = \|\mathrm{G}_{\boldsymbol{m}-\boldsymbol{m}_\varepsilon}\|_{\mathcal{OP}} \leq C\|\boldsymbol{m} - \boldsymbol{m}_\varepsilon\|_\infty \leq C\varepsilon$$

$\mathrm{G}_{\boldsymbol{m}}$ is the limit of finite rank operators and thus compact.

(iii) by Theorem 2.3 for $\boldsymbol{m} \in \boldsymbol{\ell}^2(\Lambda)$ $\mathrm{G}_{\boldsymbol{m}}$ has an integration kernel in $\boldsymbol{L}^2(\mathbb{R}^d)$ and thus is Hilbert Schmidt.

(iv) since $\mathrm{G}_{\boldsymbol{m}} = \sum_{\lambda\in\Lambda} m(\lambda)\mathrm{P}_\lambda$ and

$$\|\mathrm{G}_{\boldsymbol{m}}\|_{\mathcal{OP}} \leq \sum_{\lambda\in\Lambda} |m(\lambda)|\|\mathrm{P}_\lambda\|_{\mathcal{OP}} \leq C\|\boldsymbol{m}\|_1$$

$\mathrm{G}_{\boldsymbol{m}}$ is an absolutely convergent sum of rank 1 operators and thus trace class.

□

Remark 2.3. The fact that $\mathrm{G}_{\boldsymbol{m}}$ is self-adjoint whenever $g_1 = g_2 = g$ and $\boldsymbol{m}$ is real-valued is especially interesting for the finite dimensional case. Then $\mathrm{G}_{\boldsymbol{m}}$ corresponds to a self-adjoint matrix and thus makes computations easier.
(iii) also has some important implications. The Hilbert Schmidt operators again form a Hilbert space and there the concepts of orthogonal projections and thus best approximation (by Gabor multipliers) are well defined (see Chapter 4).
Keeping this idea in mind it is good to know that by Theorem 2.3 Gabor multipliers also depends continuously on their building blocks with respect to the $\mathcal{HS}$-norm, i.e.,

$$\|\mathrm{G}_{\boldsymbol{m}}\|_{\mathcal{HS}} = \|\kappa(\mathrm{G}_{\boldsymbol{m}})\|_2 \leq C\|g_1\|_{\boldsymbol{S}_0}\|g_2\|_{\boldsymbol{S}_0}\|\boldsymbol{m}\|_2$$

As a consequence Corollary 2.2 including the proof remains valid if we replace the operator by the Hilbert Schmidt norm.

The explorations so far concerned the influence of the upper symbol $\boldsymbol{m}$ and the windows g_1, g_2 on the properties of a Gabor multiplier, but not of the lattice Λ. The only thing that is known is that together with $\boldsymbol{S}_0$-windows any separable lattice will do fine.
There are two main questions about the influence of the lattice: First concerning the

continuity, how do small changes of the lattice constants affect the Gabor multiplier and later for the approximation of arbitrary operators, how does the density or shape of the lattice affect the approximation qualities of a Gabor system.
Before answering either of these questions it is necessary to introduce two new classes of symbols especially suited to characterise a Gabor multiplier.

2.2 The Kohn Nirenberg Symbol (KNS)

Definition 2.2. *The Kohn Nirenberg Symbol* $\sigma(\mathrm{K})$ *of an operator* K *with distributional kernel* $\kappa(\mathrm{K})$ *is the function or distribution on* $\mathbb{R}^{2d}$ *defined by:*

$$\sigma(\mathrm{K})(x,\xi) := \int_{\mathbb{R}^d} \kappa(\mathrm{K})(x, x-t) e^{-2\pi i \xi t} dt$$

Theorem 2.5. *(Properties of the KNS)*

(i) The mapping $\alpha : \kappa(\mathrm{K}) \longmapsto \sigma(\mathrm{K})$ *is a unitary Gelfand triple isomorphism on* $(\boldsymbol{S}_0, \boldsymbol{L}^2, \boldsymbol{S}_0')(\mathbb{R}^{2d})$.

(ii) (Shift covariance) The action of $\pi_2(\lambda)$ *on* K *corresponds to a translation* T_λ *of the KNS* $\sigma(\mathrm{K})$*, i.e.,*

$$\sigma(\pi_2(\lambda)(\mathrm{K})) = \mathrm{T}_\lambda \sigma(\mathrm{K}) \tag{2.3}$$

Proof:

(i) For a rigorous proof see 7.5.1 in [8]. First of all on the $\boldsymbol{S}_0$-level the KNS is well defined because the integral is absolutely convergent. Decomposing α as $\alpha = \mathcal{F}_2 \circ \mathrm{T}$, where:

$$\begin{aligned} (\mathrm{T}\, f)(x,t) &= f(x, x-t) \text{ and} \\ (\mathcal{F}_2 f)(x,\xi) &= \int_{\mathbb{R}^d} f(x,t) e^{-2\pi i \xi t} dt \end{aligned}$$

we also see that α leaves $\boldsymbol{S}_0$ invariant - the calculations are similar to those in the proof of Proposition 1.1 - and that its natural extension to $\boldsymbol{L}^2$ as a combination of unitary operators is again unitary, i.e., $\alpha^{-1} = \alpha^* = \mathrm{T}^*\, \mathcal{F}_2^*$. Therefore the kernel can be recovered from the KNS by:

$$\kappa(\mathrm{K})(x,t) = \mathrm{T}^*\, \mathcal{F}_2^* \sigma(\mathrm{K})(x,t) = \int_{\mathbb{R}^d} \sigma(\mathrm{K})(x,\xi) e^{2\pi i \xi (x-t)} d\xi$$

(ii) From Chapter 1 (1.2) we know that for $\lambda = (x,\omega)$:

$$\kappa(\pi_2(\lambda)\mathrm{K})(y,t) = \kappa(\mathrm{K})(y-x, t-x) e^{2\pi i \omega (y-t)}$$

Calculating happily away we get:

$$\begin{aligned} \sigma(\pi_2(\lambda)\mathrm{K})(y,\xi) &= \int_{\mathbb{R}^d} \kappa(\pi_2(\lambda)\mathrm{K})(y, y-t) e^{-2\pi i \xi t} dt = \\ &= \int_{\mathbb{R}^d} \kappa(\mathrm{K})(y-x, y-t-x) e^{2\pi i \omega (y-(y-t))} e^{-2\pi i \xi t} dt = \\ &= \int_{\mathbb{R}^d} \kappa(\mathrm{K})(y-x, y-x-t) e^{-2\pi i t (\xi - \omega)} dt = \\ &= \sigma(\mathrm{K})(y-x, \xi-\omega) = \mathrm{T}_{(x,\omega)}\, \sigma(\mathrm{K})(y,\xi) \end{aligned}$$

□

To get an idea what a KNS may look like, let us calculate it for one of our "pet operators", the rank 1 operator P.

Example 2.1 *(KNS of the rank 1 operator* $\mathrm{P} = f \otimes g^*$*).*

$$\begin{aligned}\sigma(\mathrm{P})(x,\xi) &= \int_{\mathbb{R}^d} f(x)\overline{g(x-t)}e^{-2\pi i\xi t}dt = \\ &= f(x)\int_{\mathbb{R}^d}\overline{g(z)}e^{2\pi i\xi(z-x)}dz = \\ &= f(x)e^{-2\pi i\xi x}\int_{\mathbb{R}^d}\overline{g(z)e^{-2\pi i\xi z}}dz = \\ &= e^{-2\pi i\xi x}f(x)\overline{\hat{g}(\xi)}\end{aligned}$$

The next proposition yields an important characterization of the KNS of a Gabor-multiplier, i.e., as a weighted sum of translates of one window function.

Proposition 2.6. *The KNS of a Gabor multiplier* $\mathrm{G}_{\boldsymbol{m}}$ *is given by a convolution over* Λ *between the upper symbol* $\boldsymbol{m}$ *and the KNS of* $\mathrm{P}_0 = g_1 \otimes g_2^*$*.*

Proof: writing $z = (x, \xi)$:

$$\begin{aligned}\sigma(\mathrm{G}_{\boldsymbol{m}})(x,\xi) &= \int_{\mathbb{R}^d}\kappa(\mathrm{G}_{\boldsymbol{m}})(x,x-t)e^{-2\pi i\xi t}dt = \\ &= \int_{\mathbb{R}^d}\sum_{\lambda\in\Lambda}m(\lambda)\kappa(\pi_2(\lambda)\mathrm{P}_0)(x,x-t)e^{-2\pi i\xi t}dt = \\ &= \sum_{\lambda\in\Lambda}m(\lambda)\int_{\mathbb{R}^d}\kappa(\pi_2(\lambda)\mathrm{P}_0)(x,x-t)e^{-2\pi i\xi t}dt = \\ &= \sum_{\lambda\in\Lambda}m(\lambda)\sigma(\pi_2(\lambda)\mathrm{P}_0)(x,\xi) = \\ &= \sum_{\lambda\in\Lambda}m(\lambda)\mathrm{T}_\lambda\sigma(\mathrm{P}_0)(x,\xi) = \qquad (2.4)\\ &= \sum_{\lambda\in\Lambda}m(\lambda)\sigma(\mathrm{P}_0)(z-\lambda) = \\ &= \boldsymbol{m} *_\Lambda \sigma(\mathrm{P}_0)(z) \qquad (2.5)\end{aligned}$$

□

2.3 The Spreading Function η

Definition 2.3. *The spreading function* $\eta(\mathrm{K})$ *of an operator K with distributional kernel* $\kappa(\mathrm{K})$ *is the function/distribution on* $\mathbb{R}^{2d}$ *defined by*

$$\eta(\mathrm{K})(t,\nu) = \int_{\mathbb{R}^d}\kappa(\mathrm{K})(x,x-t)e^{-2\pi ix\nu}dx \qquad (2.6)$$

Proposition 2.7. *(Properties of the spreading function)*

(i) The mapping α *from kernel to spreading function is a unitary Gelfand triple isomorphism on* $(\boldsymbol{S}_0, \boldsymbol{L}^2, \boldsymbol{S}_0')(\mathbb{R}^d)$ *with inverse:*

$$\kappa(\mathrm{K})(t,x) = \int_{\mathbb{R}^d}\eta(\mathrm{K})(t-x,\nu)e^{2\pi it\nu}d\nu$$

(ii) The operator K can be recovered from its spreading function by

$$\mathrm{K} = \int_{\mathbb{R}^{2d}} \eta(\mathrm{K})(t,\nu)\, \mathrm{M}_\nu\, \mathrm{T}_t\, dtd\nu \tag{2.7}$$

Proof:
(i) The idea is the same as for the KNS. A rigorous proof as well as a proof for (ii) can again be found in Section 6 in [8].
We decompose α into the two unitary operators:

$$\mathrm{T}\, f(t,x) = f(x, x-t) \qquad \mathrm{T}^*\, f(t,x) = f(t-x,t)$$
$$\mathcal{F}_2 f(t,\nu) = \int_{\mathbb{R}^d} f(t,x) e^{2\pi i x\nu} dx$$

Now the integration kernel can be reconstructed by:

$$\begin{aligned} \kappa(\mathrm{K})(t,x) &= \mathrm{T}^*\, \mathcal{F}^* \eta(\mathrm{K})(t,x) = \\ &= \mathrm{T}^* \int_{\mathbb{R}^d} \eta(\mathrm{K})(t,\nu) e^{2\pi i x\nu} d\nu = \int_{\mathbb{R}^d} \eta(\mathrm{K})(t-x,\nu) e^{2\pi i t\nu} d\nu \end{aligned}$$

□

Remark 2.4. From (2.7) we can quickly calculate the spreading function of the TF-shift-operator $\pi(\lambda)$, i.e.,

$$\pi(\lambda) = \mathrm{M}_\omega \mathrm{T}_x = \int_{\mathbb{R}^{2d}} \delta_{(\omega,x)}\, \mathrm{M}_\nu\, \mathrm{T}_t\, dtd\nu \quad \Rightarrow \quad \eta(\pi(\lambda)) = \delta_\lambda$$

The fact that the spreading symbols of TF-shifts are just the δ-distributions does not only demonstrate why it will be a useful tool for the analysis of Gabor-multipliers but is actually a uniquely determining property for the Gelfand triple isomorphism from operator to spreading symbol.

Again to get a feeling for the spreading function and for later purposes it is instructive to calculate the spreading function for the rank 1 operator $P = g_\lambda \otimes {g_{\lambda'}}^*$.

Example 2.2 *(spreading function of $P = g_\lambda \otimes {g_{\lambda'}}^*$).*

$$\begin{aligned} \eta(\mathrm{P})(t,\nu) &= \int_{\mathbb{R}^d} g_\lambda(x) \overline{g_{\lambda'}(x-t)} e^{-2\pi i x\nu} dx = \langle g_\lambda, \mathrm{M}_\nu\, \mathrm{T}_t\, g_{\lambda'} \rangle = \\ &= \langle \mathrm{M}_\omega \mathrm{T}_x g, \mathrm{M}_{\omega'}\, \mathrm{T}_{x'}\, \mathrm{M}_\nu\, \mathrm{T}_t\, g_{\lambda'} \rangle = \\ &= e^{2\pi i\varphi} \langle g, \mathrm{M}_{\nu+\omega'-\omega}\, \mathrm{T}_{t+x'-x}\, g \rangle = \qquad (\varphi = x(\omega-\omega'-\nu) + t\nu) \\ &= e^{2\pi i\varphi} \mathcal{V}_g g(\nu+\omega'-\omega, t+x'-x) \end{aligned}$$

Finally equipped with the KNS we are ready to investigate the influence of the lattice Λ on a Gabor multiplier.

2.4 Varying the Lattice

One technical problem arising by varying the lattice is that multipliers on different lattices have to be compared. Therefore the multiplier needs to be defined on the full time-frequency plane $\mathbb{R}^d \times \hat{\mathbb{R}}^d$ in a way that the samples on any separable lattice $\Lambda_k = a_k\mathbb{Z}^d \times b_k\mathbb{Z}^d$ yield a suitable upper symbol. There are two main considerations. First since Gabor multipliers are in $\mathcal{HS}$, the space of Hilbert Schmidt operators, the upper symbols $\boldsymbol{m}_k = (m(na_k, lb_k))_{(n,l)\in\mathbb{Z}^{2d}}$ have to be square summable for all $k \in \mathbb{N}$ and second $m(na_k, lb_k)$ should converge to $m(na_0, lb_0)$ for $k \to \infty$ and all $(n, l) \in \mathbb{Z}^{2d}$. Together this suggests to choose the multiplier $\boldsymbol{m} \in \boldsymbol{W}(\boldsymbol{C}, \boldsymbol{\ell}^2)$.
Finally we have all the necessary tools to state and proof the promised theorem about the continuous dependence on the lattice.

Theorem 2.8. *Let $g_1, g_2 \in \boldsymbol{S}_0$ and $\boldsymbol{m} \in \boldsymbol{W}(\boldsymbol{C}, \boldsymbol{\ell}^2)$ be given. Furthermore let (a_k, b_k) be a sequence of lattice constants such that $(a_k, b_k) \to (a_0, b_0)$, for some pair of positive lattice constants (a_0, b_0). Write G_k for the Gabor multiplier with analysis window g_1, synthesis window g_2, using the time-frequency lattice $\Lambda_k = a_k\mathbb{Z}^d \times b_k\mathbb{Z}^d$ and corresponding multiplier sequence $\boldsymbol{m}_k = (m(na_k, lb_k))_{(n,l)\in\mathbb{Z}^{2d}}$, i.e.,*

$$\mathrm{G}_k(f) = \sum_{(n,l)\in\mathbb{Z}^{2d}} m(na_k, lb_k)\langle f, \mathrm{M}_{lb_k}\,\mathrm{T}_{na_k}\,g_1\rangle\,\mathrm{M}_{lb_k}\,\mathrm{T}_{na_k}\,g_2 \tag{2.8}$$

Then the G_k converge to G_0 in the Hilbert-Schmidt norm.

Proof: (compare [6])
The proof is quite lengthy and better ignored by all who do not enjoy analytic estimates.
To show that $\mathrm{G}_k \to \mathrm{G}_0$ in the $\mathcal{HS}$-norm we need to show that their operator kernels or equivalently their KNSs (see Theorem 2.5(i)) converge in $\boldsymbol{L}^2(\mathbb{R}^{2d})$. So remembering (2.4) the KNS of a Gabor multiplier can be written as

$$\sigma(\mathrm{G}_k) = \sum_{(n,l)\in\mathbb{Z}^{2d}} m(na_k, lb_k)\,\mathrm{T}_{(na_k,lb_k)}\,\sigma(\mathrm{P}_0) \text{ with } \mathrm{P}_0 = g_1 \otimes g_2^*$$

Thus the aim is to proof that for $k \to \infty$

$$\|\sum_{(n,l)\in\mathbb{Z}^{2d}} m(na_k, lb_k)\,\mathrm{T}_{(na_k,lb_k)}\,\sigma(\mathrm{P}_0) - m(na_0, lb_0)\,\mathrm{T}_{(na_0,lb_0)}\,\sigma(\mathrm{P}_0)\|_{\boldsymbol{L}^2(\mathbb{R}^{2d})} \to 0$$

Now the idea is to split the sum over $(n, l) \in \mathbb{Z}^{2d}$ into two sums. One over a finite set F, where the weight of the multiplier sequence is concentrated but each of the summands converges nicely, and one over the rest, $F' = \mathbb{Z}^{2d}/F$, that does not contribute much, i.e.,

$$\|\sum_{(n,l)\in\mathbb{Z}^{2d}} \ldots\|_{\boldsymbol{L}^2} \le \|\sum_{(n,l)\in F} \ldots\|_{\boldsymbol{L}^2} + \|\sum_{(n,l)\in F'} \ldots\|_{\boldsymbol{L}^2} \tag{2.9}$$

To choose the set F we first reformulate the KNS of G_k again. If we define the measure $\mu_k := \sum_{(n,l)\in\mathbb{Z}^{2d}} m(na_k, lb_k)\delta_{(na_k,lb_k)}$ we can write:

$$\begin{aligned}\sigma(\mathrm{G}_k)(x,\xi) &= \sum_{(n,l)\in\mathbb{Z}^{2d}} \sigma(\mathrm{P}_0)(x-na_k,\xi-lb_k)\cdot m(na_k,lb_k) = \\ &= \int_{\mathbb{R}^{2d}} \sigma(\mathrm{P}_0)(x-x',\xi-\xi')d\mu_k(x',\xi') = \\ &= \mu_k * \sigma(\mathrm{P}_0)(x,\xi)\end{aligned}$$

and if we split μ_k into $\mu_k = \mu_k|_F + \mu_k|_{F'}$ we can write the infinite sum in (2.9) as

$$\sum_{(n,l)\in F'} \ldots = \mu_k|_{F'} * \sigma(\mathrm{P}_0) - \mu_0|_{F'} * \sigma(\mathrm{P}_0)$$

Since the compactly supported functions are dense in $\boldsymbol{W}(\boldsymbol{C},\boldsymbol{\ell}^2)$ for any $\varepsilon' > 0$ we can choose a compactly supported plateau function p with values $p(x) \in [0,1]$ such that $\|m - mp\|_{\boldsymbol{W}(\boldsymbol{C},\boldsymbol{\ell}^2)} < \varepsilon'$. Then $F := \mathbb{Z}^{2d} \cap \operatorname{supp} p$ is finite and there holds the point-wise estimate $|(1-\chi_{\operatorname{supp} p})m(x)| \le |(1-p)m(x)|$.
Now wlog assuming that the density of the lattices is bounded above, for instance $a_k > \frac{a_0}{2} > 0$, $b_k > \frac{b_0}{2} > 0$, we can estimate the $\boldsymbol{W}(\mathrm{M},\boldsymbol{\ell}^2)$-norm of
$\mu_k|_{F'} = \sum_{(n,l)\in F'} m(na_k, lb_k)\delta_{(na_k,lb_k)}$. (write $Q = [0,1]^{2d}$)

$$\begin{aligned}\|\mu_k|_{F'}\|^2_{\boldsymbol{W}(\mathrm{M},\boldsymbol{\ell}^2)} &= \sum_{j\in\mathbb{Z}^{2d}} |\mu_k|_{F'}|^2(j+Q) \le \\ &\le \sum_{j\in\mathbb{Z}^{2d}} \sup_{x\in j+Q} |(1-\chi_{\operatorname{supp} p})m(x)|^2 \cdot \#^2\{(\tfrac{na_0}{2},\tfrac{lb_0}{2}) \in j+Q\} \le \\ &\le \sum_{j\in\mathbb{Z}^{2d}} \sup_{x\in j+Q} |(1-p)m(x)|^2 \cdot C_1^2 \le \\ &\le C_1^2\|(1-p)m\|^2_{\boldsymbol{W}(\boldsymbol{C},\boldsymbol{\ell}^2)} \le C_1^2\varepsilon'^2\end{aligned}$$

Finally using the convolution relation for Wiener amalgams of the form $\boldsymbol{W}(\mathrm{M},\boldsymbol{\ell}^2)(\mathbb{R}^{2d}) * \boldsymbol{W}(\boldsymbol{C},\boldsymbol{\ell}^1)(\mathbb{R}^{2d}) \subseteq \boldsymbol{W}(\boldsymbol{C},\boldsymbol{\ell}^2)(\mathbb{R}^{2d})$ we can estimate the norm of the infinite sum by:

$$\begin{aligned}\|\sum_{(n,l)\in F'} \ldots\|_{\boldsymbol{L}^2} &\le \|\mu_k|_F * \sigma(\mathrm{P}_0)\|_{\boldsymbol{W}(\boldsymbol{C},\boldsymbol{\ell}^2)} + \|\mu_0|_{F'} * \sigma(\mathrm{P}_0)\|_{\boldsymbol{W}(\boldsymbol{C},\boldsymbol{\ell}^2)} \le \\ &\le (\|\mu_k|_F\|_{\boldsymbol{W}(\mathrm{M},\boldsymbol{\ell}^2)} + \|\mu_0|_{F'}\|_{\boldsymbol{W}(\mathrm{M},\boldsymbol{\ell}^2)}) \cdot \|\sigma(\mathrm{P}_0)\|_{\boldsymbol{W}(\boldsymbol{C},\boldsymbol{\ell}^1)} \le \\ &\le 2C_1\varepsilon' \cdot \|\sigma(\mathrm{P})\|_{\boldsymbol{S}_0} \le \frac{\varepsilon}{2} \qquad (2.10)\end{aligned}$$

To estimate the finite part of the sum in (2.9) observe that $m(na_k, lb_k) \to m(na_0, lb_0)$ for all $(n,l) \in F$. Also since $\sigma(\mathrm{P}_0) \in \boldsymbol{S}_0 \subseteq \boldsymbol{L}^2$ we have that $\|T_\delta\sigma(\mathrm{P}_0) - \sigma(\mathrm{P}_0)\|_{\boldsymbol{L}^2} \le \varepsilon'$ for $\|\delta\| \le \delta_0$. Thus we can find a $k_0 \in \mathbb{N}$ such that for all $k > k_0$ and all $(n,l) \in F$:

$$\|(na_k, lb_k) - (na_0, lb_0)\| \le \delta_0 \quad \text{and} \quad |m(na_k, lb_k) - m(na_0, lb_0)| \le \varepsilon'$$

Now writing $\lambda_k = (na_k, lb_k)$, $\lambda_0 = (na_0, lb_0)$ we can estimate each term of the finite

sum by:

$$\begin{aligned}
&\|m(\lambda_k)\,\mathrm{T}_{\lambda_k}\,\sigma(\mathrm{P}_0) - m(\lambda_0)\,\mathrm{T}_{\lambda_0}\,\sigma(\mathrm{P}_0)\|_{\boldsymbol{L}^2} \leq \\
&\quad\leq \|m(\lambda_k)\,\mathrm{T}_{\lambda_k}\,\sigma(\mathrm{P}_0) - m(\lambda_0)\,\mathrm{T}_{\lambda_k}\,\sigma(\mathrm{P}_0)\|_{\boldsymbol{L}^2} + \\
&\qquad\qquad \|m(\lambda_0)\,\mathrm{T}_{\lambda_k}\,\sigma(\mathrm{P}_0) - m(\lambda_0)\,\mathrm{T}_{\lambda_0}\,\sigma(\mathrm{P}_0)\|_{\boldsymbol{L}^2} \leq \\
&\quad\leq |m(\lambda_k) - m(\lambda_0)| \cdot \|\sigma(\mathrm{P}_0)\|_{\boldsymbol{L}^2} + |m(\lambda_0)| \cdot \|\,\mathrm{T}_{\lambda_k-\lambda_0}\,\sigma(\mathrm{P}_0) - \sigma(\mathrm{P}_0)\|_{\boldsymbol{L}^2} \leq \\
&\quad\leq \varepsilon' \cdot \|\kappa(\mathrm{P}_0)\|_{\boldsymbol{S}_0} + \|m\|_{\boldsymbol{W}(\boldsymbol{C},\boldsymbol{\ell}^2)} \cdot \varepsilon' \leq C_3\varepsilon' \leq \frac{\varepsilon}{2\cdot \#F} \qquad (2.11)
\end{aligned}$$

Summarizing the estimates (2.10) and (2.11) we see that for $k > k_0$:

$$\|\sigma(\mathrm{G}_k) - \sigma(\mathrm{G}_0)\|_{\boldsymbol{L}^2} \leq \#F \cdot \frac{\varepsilon}{2\cdot\#F} + \frac{\varepsilon}{2} \leq \varepsilon$$

□

Similar results about convergence in other operator classes are discussed in [9] and [6] . The reason for choosing the $\mathcal{HS}$ version here is that in combination with Remark 2.3 the theorem guarantees complete stability for Gabor multipliers in $\mathcal{HS}$, which will be reassuring to know when trying to do best approximation in that operator class.

Corollary 2.9. *Let $g_k, \tilde{g}_k$ be two sequences of atoms converging in $\boldsymbol{S}_0$ to $g_0, \tilde{g}_0$, $\boldsymbol{m}_k$ a sequence of functions converging in $\boldsymbol{W}(\boldsymbol{C}, \boldsymbol{\ell}^2)$ to $\boldsymbol{m}_0$ and (a_k, b_k) a sequence of lattice constants converging to (a_0, b_0). Then the Gabor multipliers G_k*

$$\mathrm{G}_k(f) = \mathrm{G}_{m_k,g_k,\tilde{g}_k,\Lambda_k} f = \sum_{(n,l)\in\mathbb{Z}^{2d}} m_k(na_k, lb_k)\langle f, \mathrm{M}_{lb_k}\,\mathrm{T}_{na_k}\,g_k\rangle\,\mathrm{M}_{lb_k}\,\mathrm{T}_{na_k}\,\tilde{g}_k$$

converge to G_0 in the $\mathcal{HS}$-norm.

Proof:
Applying the triangular equation yields:

$$\begin{aligned}
&\|\mathrm{G}_{m_0,g_0,\tilde{g}_0,\Lambda_0} - \mathrm{G}_{m_k,g_k,\tilde{g}_k,\Lambda_k}\|_{\mathcal{HS}} \leq \\
&\quad\leq \|\underbrace{\mathrm{G}_{m_0,g_0,\tilde{g}_0,\Lambda_0} - \mathrm{G}_{m_0,g_0,\tilde{g}_0,\Lambda_k}}_{\leq\varepsilon \text{ (by Theorem 2.8)}}\|_{\mathcal{HS}} + \|\underbrace{\mathrm{G}_{m_0,g_0,\tilde{g}_0,\Lambda_k} - \mathrm{G}_{m_k,g_k,\tilde{g}_k,\Lambda_k}}_{\leq\varepsilon \text{ (by Remark 2.3)}}\|_{\mathcal{HS}}
\end{aligned}$$

□

After having established the continuous dependence of a Gabor multiplier on all its building blocks it would be interesting to have a look at the reversed situation. So given a Gabor multiplier $\mathrm{G}_{\boldsymbol{m}}$ can the upper symbol $\boldsymbol{m}$ be reconstructed assuming that it is known which lattice and windows have been used. This problem leads to the next questions. Which operators can be accurately represented by a Gabor multiplier and how can we approximate any given linear operator by a Gabor multiplier? The key to the answering these questions is to think of $\mathrm{G}_{\boldsymbol{m}}$ as an infinite linear combination of the projection operators $(\mathrm{P}_\lambda)_{\lambda\in\Lambda}$, compare (2.2). From this point of view $\mathcal{GM}_2$, the space of all Gabor multipliers with $\boldsymbol{\ell}^2$-multiplier can be interpreted as the closed linear span of $(\mathrm{P}_\lambda)_{\lambda\in\Lambda}$ in $\mathcal{HS}$, the space of Hilbert Schmidt operators, and so studying the family $(\mathrm{P}_\lambda)_{\lambda\in\Lambda}$ should yield information about $\mathcal{GM}_2$.

2.5 $(\mathrm{P}_\lambda)_{\lambda\in\Lambda}$ as a Riesz basis

The idea that $(\mathrm{P}_\lambda)_{\lambda\in\Lambda}$ should be a Riesz basis is suggested by the aim to be able to reconstruct the upper symbol $\boldsymbol{m}$ from the linear combination $\mathrm{G}_{\boldsymbol{m}} = \sum_{\lambda\in\Lambda} m(\lambda)\mathrm{P}_\lambda$ or in other words to have unique coefficients $m(\lambda)$. So let's assume for a moment that $(\mathrm{P}_\lambda)_{\lambda\in\Lambda}$ is a Riesz basis. Consequently there exists a biorthogonal basis $(\mathrm{Q}_\lambda)_{\lambda\in\Lambda}$, that allows us to determine the coefficients $m(\lambda)$ just by taking inner products with it. Also it is now easy to calculate the orthogonal projection of a general Hilbert-Schmidt operator K onto $\mathcal{GM}_2$, i.e., $\mathrm{K}_P = \sum_{\lambda\in\Lambda}\langle \mathrm{K}, \mathrm{Q}_\lambda\rangle \mathrm{P}_\lambda$, which is at the same time the best approximation of K by a Gabor multiplier.
However it remains to be investigated if a family $(\mathrm{P}_\lambda)_{\lambda\in\Lambda}$ can be a Riesz basis in $\mathcal{HS}$ and if yes under which additional conditions on the windows and the lattice. The answer to this question is given by the following theorem.

Theorem 2.10. *Suppose that for $g \in \boldsymbol{S}_0$, (g,Λ) generates a Gabor frame for $\boldsymbol{L}^2(\mathbb{R}^d)$. Then the family of projection operators $(\mathrm{P}_\lambda = g_\lambda \otimes {g_\lambda}^*)_{\lambda\in\Lambda}$ is a Riesz basis for its closed linear span within the Hilbert space $\mathcal{HS}$ of all Hilbert-Schmidt operators on $\boldsymbol{L}^2(\mathbb{R}^d)$ if and only if the Λ-Fourier transform of $(|\mathcal{V}_g g(\lambda)|^2)_{\lambda\in\Lambda}$ is free of zeros.*

Proof:
By definition $(\mathrm{P}_\lambda)_{\lambda\in\Lambda}$ is a Riesz basis if there exist constants $A, B > 0$ such that for all $c \in \boldsymbol{\ell}^2(\Lambda)$:

$$A\|c\|_2^2 \leq \|\sum_{\lambda\in\Lambda} c(\lambda)\mathrm{P}_\lambda\|_{\mathcal{HS}}^2 \leq B\|c\|_2^2 \tag{2.12}$$

By Remark 2.3, $c \simeq \boldsymbol{m}$, the upper Riesz bound always exists and satisfies $B \leq C\|g\|_{\boldsymbol{S}_0}$ for $g \in \boldsymbol{S}_0$.
To check the lower bound we first rewrite the inner expression of (2.12):

$$\|\sum_{\lambda\in\Lambda} c(\lambda)\mathrm{P}_\lambda\|_{\mathcal{HS}}^2 = \sum_{\lambda\in\Lambda}\sum_{\lambda'\in\Lambda} c(\lambda)\overline{c(\lambda')}\langle \mathrm{P}_\lambda, \mathrm{P}_{\lambda'}\rangle_{\mathcal{HS}}$$

and calculate $\langle \mathrm{P}_\lambda, \mathrm{P}_{\lambda'}\rangle_{\mathcal{HS}}$.

$$\begin{aligned}
\langle \mathrm{P}_\lambda, \mathrm{P}_{\lambda'}\rangle_{\mathcal{HS}} &= \langle g_\lambda \otimes {g_\lambda}^*, g_{\lambda'} \otimes g_{\lambda'}^*\rangle_{\boldsymbol{L}^2(\mathbb{R}^{2d})} = \langle g_\lambda, g_{\lambda'}\rangle_{\boldsymbol{L}^2(\mathbb{R}^d)}\langle {g_\lambda}^*, g_{\lambda'}^*\rangle_{\boldsymbol{L}^2(\mathbb{R}^d)} = \\
&= |\langle \pi(\lambda)g, \pi(\lambda')g\rangle|^2 = |\langle g, \pi^*(\lambda)\pi(\lambda')g\rangle|^2 = \\
&= |\langle g, e^{2\pi i x(\omega'-\omega)}\pi(\lambda'-\lambda)g\rangle|^2 = |\langle g, \pi(\lambda'-\lambda)g\rangle|^2 = \\
&= |\mathcal{V}_g g(\lambda'-\lambda)|^2
\end{aligned}$$

Writing $\gamma = (|\mathcal{V}_g g(\lambda)|^2)_{\lambda\in\Lambda}$ there holds:

$$\begin{aligned}
\|\sum_{\lambda\in\Lambda} c(\lambda)\mathrm{P}_\lambda\|_{\mathcal{HS}}^2 &= \sum_{\lambda\in\Lambda} c(\lambda)\sum_{\lambda'\in\Lambda}\overline{c(\lambda')}\gamma(\lambda'-\lambda) = \\
&= \sum_{\lambda'\in\Lambda}\overline{c(\lambda')}(c *_\Lambda \gamma)(\lambda') = \\
&= \langle c *_\Lambda \gamma, c\rangle_{\boldsymbol{\ell}^2(\Lambda)} = \\
&= \langle \hat{c}\cdot\hat{\gamma}, \hat{c}\rangle_{\boldsymbol{L}^2(\mathbb{T}_{\Lambda^\perp})}
\end{aligned}$$

Now it is easy to see that the first part of (2.12) is equivalent to

$$A\int_{\mathbb{T}_{\Lambda^\perp}} |\hat{c}(x)|^2 dx \le \int_{\mathbb{T}_{\Lambda^\perp}} \hat{\gamma}(x)|\hat{c}(x)|^2 dx \qquad \forall \hat{c}\in \boldsymbol{L}^2(\mathbb{T}_{\Lambda^\perp})$$

and that can only hold if $\hat{\gamma}\ge A>0$ on $\mathbb{T}_{\Lambda^\perp}$. This is equivalent to being free of zeros because $\hat{\gamma}$ is continuous, i.e., $\mathcal{V}_g g$ is in $\boldsymbol{W}(\boldsymbol{C},\boldsymbol{\ell}^1)$ and because $\|(|\mathcal{V}_g g|^2)\|_{\boldsymbol{W}(\boldsymbol{C},\boldsymbol{\ell}^1)} \le \|\overline{\mathcal{V}_g g}\|_\infty \cdot \|\mathcal{V}_g g\|_{\boldsymbol{W}(\boldsymbol{C},\boldsymbol{\ell}^1)}$ so is $|\mathcal{V}_g g|^2$. Consequently $\gamma\in\boldsymbol{\ell}^1(\Lambda)$ and so its Λ-Fourier transform is absolutely convergent and thus continuous. □

Corollary 2.11. *Under the assumptions in Theorem 2.10 the mapping α from multiplying sequence $\boldsymbol{m}$ to Gabor multiplier $\mathrm{G}_{\boldsymbol{m}}$ is invertible between $\boldsymbol{\ell}^2(\Lambda)$ and $\mathcal{GM}_2\subseteq\mathcal{HS}$.*

In order to explicitly write down the inverse of the mapping α and to see if the invertibility is also possible on the level of $\boldsymbol{\ell}^1$ and $\boldsymbol{\ell}^\infty$ it is necessary to find out more about the biorthogonal system $(\mathrm{Q}_\lambda)_{\lambda\in\Lambda}$.

Lemma 2.12. *Let $(\mathrm{P}_\lambda)_{\lambda\in\Lambda}$ be a Riesz basis in $\mathcal{HS}$. Then the uniquely determined biorthogonal basis $(\mathrm{Q}_\lambda)_{\lambda\in\Lambda}\subseteq\overline{\mathrm{span}(\mathrm{P}_\lambda)_{\lambda\in\Lambda}}$ is again of the form $\mathrm{Q}_\lambda=\pi_2(\lambda)\mathrm{Q}_0$.*

Proof:
The idea is to show that $(\pi_2(\lambda)\mathrm{Q}_0)_{\lambda\in\Lambda}$ is a biorthogonal system. Then assuming that $\mathrm{Q}_0\in\overline{\mathrm{span}(\mathrm{P}_\lambda)_{\lambda\in\Lambda}}$ it has to be the uniquely determined one.

$$\begin{aligned}\langle\pi_2(\lambda)\mathrm{Q}_0,\mathrm{P}_{\lambda'}\rangle_{\mathcal{HS}} &= \langle \mathrm{T}_\lambda\,\sigma(\mathrm{Q}_0),\mathrm{T}_{\lambda'}\,\sigma(\mathrm{P}_0)\rangle_{\boldsymbol{L}^2} = \\ &= \langle \sigma(\mathrm{Q}_0),\mathrm{T}_{\lambda'-\lambda}\,\sigma(\mathrm{P}_0)\rangle_{\boldsymbol{L}^2} = \\ &= \langle \mathrm{Q}_0,\mathrm{P}_{\lambda'-\lambda}\rangle_{\mathcal{HS}} = \delta_{0,\lambda'-\lambda}=\delta_{\lambda,\lambda'}\end{aligned}$$

□

Lemma 2.13. *The operator Q_0 generating the biorthogonal basis is given by its KNS as*

$$\sigma(\mathrm{Q}_0)=\mathcal{F}_\Lambda{}^{-1}\Big(\frac{1}{\mathcal{F}_\Lambda(|\mathcal{V}_g g_{|\Lambda}|^2)}\Big)*_\Lambda\sigma(\mathrm{P}_0) \tag{2.13}$$

Proof:
Since Q_0 is in the closed linear span of $(\mathrm{P}_\lambda)_{\lambda\in\Lambda}$ it can be written as $\mathrm{Q}_0=\sum_{\lambda\in\Lambda}c(\lambda)\mathrm{P}_\lambda$ with uniquely determined coefficient sequence $c\in\boldsymbol{\ell}^2(\Lambda)$. By biorthogonality there holds $\langle\mathrm{Q}_0,\mathrm{P}_\lambda\rangle=\delta_{0,\lambda}$ and so:

$$\begin{aligned}\delta_{0,\lambda} &= \langle\sum_{\lambda'\in\Lambda}c(\lambda')\mathrm{P}_{\lambda'},\mathrm{P}_\lambda\rangle=\sum_{\lambda'\in\Lambda}c(\lambda')\langle\mathrm{P}_{\lambda'},\mathrm{P}_\lambda\rangle= \\ &= \sum_{\lambda'\in\Lambda}c(\lambda')|\mathcal{V}_g g(\lambda-\lambda')|^2 = c*_\Lambda|\mathcal{V}_g g_{|\Lambda}|^2(\lambda) \\ \text{or}\qquad \delta_0 &= c*_\Lambda|\mathcal{V}_g g_{|\Lambda}|^2 \end{aligned}\tag{2.14}$$

Applying the Λ-Fourier transform to (2.14) yields:

$$\begin{aligned} 1 &\equiv \mathcal{F}_\Lambda c \cdot \mathcal{F}_\Lambda(|\mathcal{V}_g g_{|_\Lambda}|^2) \\ c &= \mathcal{F}_\Lambda{}^{-1}\Big(\frac{1}{\mathcal{F}_\Lambda(|\mathcal{V}_g g_{|_\Lambda}|^2)}\Big) \end{aligned}$$

Now (2.13) is a consequence of Theorem 2.5, that the KNS of a Gabor multiplier is given by a Λ-convolution of the multiplier and $\sigma(\mathrm{P}_0)$. □

Remark 2.5. In the above argument was only assumed that $c \in \boldsymbol{\ell}^2(\Lambda)$. In [5] however is shown that in fact c is even in $\boldsymbol{\ell}^1(\Lambda)$. The main argument is that $f = \mathcal{F}_\Lambda(|\mathcal{V}_g g_{|_\Lambda}|^2)$ is a continuous function (compare Theorem 2.10). As it is bounded away from zero by the Wiener-Levy theorem its inverse again has an absolutely convergent Fourier series and so $c \in \boldsymbol{\ell}^1$.
Now as a consequence of $c \in \boldsymbol{\ell}^1$ and Theorem 2.3 Q_0 again has an integration kernel in $\boldsymbol{S}_0$.

Before further exploiting the $\boldsymbol{S}_0$-property of Q_0 note that Q_0 is only almost as "nice" as P_0, i.e., if the generating Gabor system $(g_\lambda)_{\lambda\in\Lambda}$ is a redundant frame Q_0 cannot be a rank 1 operator like P_0.
Assume there exist $\gamma_1, \gamma_2 \in \boldsymbol{S}_0(\mathbb{R}^d)$ such that $\mathrm{Q}_0 = \gamma_1 \otimes \gamma_2^*$, then:

$$\begin{aligned} \delta_{\lambda,\lambda'} &= \langle \mathrm{P}_\lambda, \mathrm{Q}_{\lambda'}\rangle = \langle \pi_2(\lambda)g \otimes g^*, \pi_2(\lambda')\gamma_1 \otimes \gamma_2^*\rangle = \\ &= \langle \pi(\lambda)g \otimes (\pi(\lambda)g)^*, \pi(\lambda')\gamma_1 \otimes (\pi(\lambda')\gamma_2)^*\rangle = \\ &= \langle \pi(\lambda)g, \pi(\lambda')\gamma_1\rangle\overline{\langle \pi(\lambda)g, \pi(\lambda')\gamma_2\rangle} = \\ &= \langle \pi(\lambda)g, \pi(\lambda')(\langle \pi(\lambda)g, \pi(\lambda')\gamma_2\rangle\gamma_1)\rangle \end{aligned}$$

Therefore $\langle \pi(\lambda)g, \pi(\lambda')\gamma_2\rangle\gamma_1$ is a biorthogonal system to $(g_\lambda)_{\lambda\in\Lambda}$, which means that it is a Riesz basis and thus cannot be redundant.

Coming back to the statement of Corollary 2.11 it is now possible to explicitly write down how to reconstruct the multiplying sequence for a Gabor multiplier $\mathrm{G}_{\boldsymbol{m}} \in \mathcal{GM}_2$, i.e.,

$$\boldsymbol{m} = (m(\lambda))_{\lambda\in\Lambda} = (\langle \mathrm{G}_{\boldsymbol{m}}, \mathrm{Q}_\lambda\rangle)_{\lambda\in\Lambda} \tag{2.15}$$

However since $\kappa(\mathrm{Q}_\lambda) \in \boldsymbol{S}_0(\mathbb{R}^{2d})$ the mapping $\beta : \mathrm{T} \to (\langle \mathrm{T}, \mathrm{Q}_\lambda\rangle)_{\lambda\in\Lambda}$ can by duality be extended to all operators with kernel in $\boldsymbol{S}_0'(\mathbb{R}^{2d})$ and especially to all linear operators on $\boldsymbol{L}^2$. Applying β to T we get a bounded sequence $\boldsymbol{m}_\mathrm{T}$ with $\|\boldsymbol{m}_\mathrm{T}\|_\infty \leq C\|\kappa(\mathrm{T})\|_{\boldsymbol{S}_0'} \leq \tilde{C}\|\,\mathrm{T}\,\|_{\mathcal{OP}}$, and thus by Theorem 2.1 the Gabor multiplier $\mathrm{G}_{\boldsymbol{m}_\mathrm{T}}$ is well defined.
Similarly starting with an operator T with $\boldsymbol{S}_0$-kernel the sequence $\boldsymbol{m}_\mathrm{T}$ will be in $\boldsymbol{\ell}^1$.

$$\begin{aligned} \|\boldsymbol{m}_\mathrm{T}\|_1 &= \sum_{\lambda\in\Lambda} |\langle \kappa(\mathrm{T}), \kappa(\mathrm{Q}_\lambda)\rangle| = \sum_{\lambda\in\Lambda} e^{i\varphi(\lambda)}\langle \kappa(\mathrm{T}), \kappa(\mathrm{Q}_\lambda)\rangle = \\ &= \langle \kappa(\mathrm{T}), \sum_{\lambda\in\Lambda} e^{i\varphi(\lambda)}\kappa(\mathrm{Q}_\lambda)\rangle \leq \|\kappa(\mathrm{T})\|_{\boldsymbol{S}_0} \cdot \|\sum_{\lambda\in\Lambda} e^{i\varphi(\lambda)}\kappa(\mathrm{Q}_\lambda)\|_{\boldsymbol{S}_0'} < \infty \end{aligned}$$

Using the language of Gelfand triples these facts can be summarized in the following Corollary.

Corollary 2.14. *(i) β is a surjective bounded linear mapping from $(\mathcal{B},\mathcal{HS},\mathcal{B}')$ to $(\boldsymbol{\ell}^1,\boldsymbol{\ell}^2,\boldsymbol{\ell}^\infty)(\Lambda)$.*

(ii) If the space of all Gabor multipliers with $\boldsymbol{\ell}^{(1,2,\infty)}$-multiplier is denoted by $\mathcal{GM}_{(1,2,\infty)}$ then the mapping from multiplier to Gabor multiplier is bounded invertible between $(\boldsymbol{\ell}^1,\boldsymbol{\ell}^2,\boldsymbol{\ell}^\infty)$ and $(\mathcal{GM}_1,\mathcal{GM}_2,\mathcal{GM}_\infty)\subseteq(\mathcal{B},\mathcal{HS},\mathcal{B}')$.

Proof:
ii) is a consequence of Remark 2.2 and the fact that restricted to $(\mathcal{GM}_1,\mathcal{GM}_2,\mathcal{GM}_\infty)$ β is just the inverse of the mapping α. □

A more applicable consequence of the rather abstract considerations above is that in order to represent the identity operator by a Gabor multiplier we need to use a tight atom g and have $\boldsymbol{m}\equiv c$.

$$\begin{aligned} m(\lambda) &= \langle\kappa(\mathrm{Id}),\kappa(\mathrm{Q}_\lambda)\rangle_{(\boldsymbol{S}_0',\boldsymbol{S}_0)} = \langle\sigma(\mathrm{Id}),\sigma(\mathrm{Q}_\lambda)\rangle_{(\boldsymbol{S}_0',\boldsymbol{S}_0)} = \\ &= \langle 1,\mathrm{T}_\lambda\,\sigma(\mathrm{Q}_0)\rangle_{(\boldsymbol{S}_0',\boldsymbol{S}_0)} = \langle 1,\sigma(\mathrm{Q}_0)\rangle_{(\boldsymbol{S}_0',\boldsymbol{S}_0)} = m(0) \end{aligned}$$

So we have $f=\mathrm{Id}f=\sum_{\lambda\in\Lambda}m(0)\langle f,g_\lambda\rangle g_\lambda$ and thus $(g_\lambda)_{\lambda\in\Lambda}$ has to be a tight frame.

Remark 2.6. Of course to represent the identity it is also possible to take g and its dual $\tilde{g}$ but in that case we do not know when $\mathrm{P}_0=g\otimes\tilde{g}^*$ generates a Riesz basis or how Q_0 looks like.

Now that all details about Q_0 have been established everything is in place to state the theorem about best approximation of general linear operators by Gabor multipliers.

Theorem 2.15. *Let (g,Λ) be a Gabor system that generates a Riesz basis $(\mathrm{P}_\lambda)_{\lambda\in\Lambda}$ and let $(\mathrm{Q}_\lambda)_{\lambda\in\Lambda}$ be the biorthogonal basis. Then the best approximation of a Hilbert Schmidt operator* K *by a Gabor multiplier based on (g,Λ) is given by:*

$$\mathrm{PG}(\mathrm{K})=\sum_{\lambda\in\Lambda}\langle\mathrm{K},\mathrm{Q}_\lambda\rangle_{\mathcal{HS}}\mathrm{P}_\lambda=\sum_{\lambda\in\Lambda}\langle\mathrm{K},\mathrm{P}_\lambda\rangle_{\mathcal{HS}}\mathrm{Q}_\lambda \tag{2.16}$$

Proof:
Since $(\mathrm{P}_\lambda)_{\lambda\in\Lambda}$ is a Riesz basis (2.16) describes the orthogonal projection from $\mathcal{HS}$ onto $\mathcal{GM}_2$ and thus the best approximation. □

Remark 2.7. (i) The combination of Corollary 2.14 and Theorem 2.1 shows that PG can actually be extended to a bounded mapping from $(\mathcal{B},\mathcal{HS},\mathcal{B}')$ to $(\mathcal{GM}_1,\mathcal{GM}_2,\mathcal{GM}_\infty)$. Hence if we interpret orthogonal projection liberally any bounded linear operator on $\boldsymbol{L}^2(\mathbb{R}^d)$ can be "best" approximated by a Gabor multiplier.

(ii) Note that the coefficients of the second expansion in (2.16) can be rewritten as

$$\langle\mathrm{K},\mathrm{P}_\lambda\rangle_{\mathcal{HS}}=\int_{\mathbb{R}^{2d}}\kappa(\mathrm{K})(x,y)g_\lambda(x)\overline{g_\lambda(y)}dxdy=\langle\mathrm{K}g_\lambda,g_\lambda\rangle_{\boldsymbol{L}^2} \tag{2.17}$$

$\langle \mathrm{K} g_\lambda, g_\lambda \rangle$ can be calculated for any operator without knowing about its kernel and is know as lower symbol $\sigma_L(\mathrm{K})$. Since on the space of all Gabor multipliers PG restricts to the identity Theorem 2.15 just says that a Gabor multiplier and consequently the upper symbol $\boldsymbol{m}$ can be reconstructed from its lower symbol. The mapping from σ_L to $\boldsymbol{m}$ is given by the inverse of the Gram matrix $\langle \mathrm{P}_\lambda, \mathrm{P}_{\lambda'} \rangle_{\lambda,\lambda'}$ of the system $(\mathrm{P}_\lambda)_{\lambda \in \Lambda}$.

The interesting questions arising now that there are formulas for the best approximation are: Which operators are well represented by a Gabor multiplier and how crucial is the choice of the approximating system (g, Λ)?

However answers to both questions will be postponed to Chapter 4 because any kind of result in abstract terms would not be very satisfactory if no numerical example was provided. Therefore the next chapter is dedicated to the development of analogue theory for vectors of finite length instead of $\boldsymbol{S}_0$-functions and to the numerical verification of the continuous results.

Chapter 3

From $\boldsymbol{L}^2$ to $\mathbb{C}^L$

As already mentioned the aim of this chapter is to adapt all previous results and definitions to the "facts of real life". All calculations have to be done using computers, which unfortunately cannot deal with with continuous signals like $\boldsymbol{L}^2$-functions but only with discrete signals of finite length, i.e., elements of $\mathbb{C}^L$.
The structure of this chapter will follow mainly that of Chapters 1 and 2, looking at theory from a numerical point of view and developing a tool box of MATLAB files to underline the continuous results with numerical experiments. One technical detail that should be mentioned is that in this presentation all indices will start with 0. MATLAB however starts counting with 1 which may lead to confusion when comparing the explanation of the algorithms, all based on NuHAG code, with the m-files in the Appendix.

3.1 Fourier Analysis in $\mathbb{C}^L$

An important thing at the beginning is to find suitable discrete substitutes for the (Λ)-Fourier transform and TF-shifts because unlike later for frames and bases the continuous definitions make little or no sense in the discrete setting. One way to the discrete definition would be to generalize all the concepts of Fourier analysis to arbitrary groups and then have a look at what this means for the special case $\mathbb{C}^L$. The beauty of this approach, as can be found in [19], is that it shows how Fourier series, the (Λ-)Fourier transform, etc are basically all aspects of the same theory. It is however quite unsatisfying from an applied mathematician's point of view, who wants to be sure that the discrete calculations, done on the computer, actually simulate the continuous case. So the "on foot" approach taken here is simply to try and approximate the continuous formulas and see where this leads.
Recall the definition of the Fourier transform for a function in $\boldsymbol{L}^1(\mathbb{R})$:

$$\hat{f}(\omega) = \int_{\mathbb{R}} f(x) e^{-2\pi i x \omega} dx \tag{3.1}$$

Lets approximate the integral in (3.1) by a Riemann sum with L data points, compare Figure (3.1).

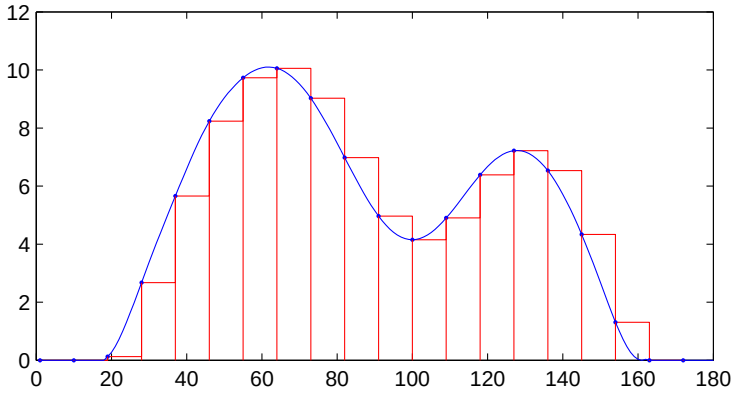


Figure 3.1: Approximating the Fourier Integral

$$\hat{f}(\omega) \approx \tilde{f}(\omega) = \sum_{k=0}^{L-1} f(\frac{bk}{L}) e^{-2\pi i \frac{bk}{L}\omega} \cdot \frac{b}{L} \tag{3.2}$$

The question arising now is for which values of ω $\tilde{f}(\omega)$ should be calculated. A look at (3.2) shows that $\tilde{f}$ is periodic with period $\frac{L}{b}$, i.e., $\tilde{f}(\omega + \frac{L}{b}) = \tilde{f}(\omega)$. Together with the consideration that, since the input consisted of L data points so should the output, this suggests choosing the values $\frac{j}{L} \cdot \frac{L}{b} = \frac{j}{b}$ for $j = 0 \dots L-1$ and so:

$$\tilde{f}(\frac{j}{b}) = \frac{b}{L} \sum_{k=0}^{L-1} f(\frac{bk}{L}) e^{-2\pi i \frac{jk}{L}}$$

Finally after some cosmetic normalisations this leads to the following definition:

Definition 3.1 (discrete Fourier transform - FFT). *For a signal $f \in \mathbb{C}^L$ the Fourier transform $\hat{f}$ is defined as*

$$\hat{f}(j) = \sum_{k=0}^{L-1} f(k) e^{-2\pi i \frac{jk}{L}} \tag{3.3}$$

Remark 3.1. (i) Because of the many efficient and fast algorithms for calculating the discrete Fourier transform it is also called Fast-Fourier-Transform (FFT).

(ii) Sometimes the discrete Fourier transform can be found with an additional factor $\frac{1}{L}$ or $\frac{1}{\sqrt{L}}$. The advantage of the definition given here is that it is consistent with the way the FFT is implemented in MATLAB (`b=fft(a)`). The drawback is that it is no longer a unitary operator, as it would be with the factor $\frac{1}{\sqrt{L}}$. Therefore the inverse Fourier transform (MATLAB `b=ifft(a)`) is given by:

$$\check{f}(j) = \frac{1}{L} \sum_{k=0}^{L-1} f(k) e^{2\pi i \frac{jk}{L}}$$

(iii) Because the n-dimensional Fourier transform can be decomposed into n one-dimensional Fourier transforms in each component Definition (3.1) also settles this case. Still for n=2 MATLAB provides the function `fft2`.

The definition of the FFT now helps motivating the right definitions of the discrete TF-shifts. At first glance the problem with the translation operator T_n is that, in analogy to T_x, setting

$$\mathrm{T}_n f(k) = f(k-n)$$

there might be values for $f(-1)$ but not for $f(L-1)$. However Definition 3.1 shows that the discretisation introduced an L-periodicity, i.e., theoretically $\hat{f}$ could be calculated for all $j \in \mathbb{Z}$ and the resulting sequence would be L-periodic. Indeed thinking of f as a periodic sequence in $\mathbb{Z}$ or $\mathbb{Z}_L$ solves the problem of too many yet not enough values.
The right definition of the modulation operator M_m is motivated by looking at the equivalent of the term $e^{-2\pi i x\omega}$ in (3.3), i.e., $e^{-2\pi i \frac{jk}{L}}$. So:

Definition 3.2 (discrete TF-shifts).

$$\begin{aligned} \mathrm{T}_n f(k) &= f((k-n) mod_L) \\ \mathrm{M}_m f(k) &= f(k) \cdot e^{2\pi i \frac{mk}{L}} \end{aligned}$$

Remark 3.2. For the rest of this chapter, unless further specified, all sums are considered to be modulo L.

3.2 Riesz Bases and Frames

Both the definition of a Riesz basis and a frame given in Chapter 1 are valid for general Hilbert spaces, so there is not need to make any adjustments. However it is still worth looking at both from a linear algebra point of view.
Recalling Definition 1.1 a Riesz basis $\{f_1 \dots f_n\}$ should span $\mathbb{C}^L$ and satisfy:

$$A\|c\|_2 \leq \|\sum_{k=1}^{n} c_k f_k\|_2 \leq B\|c\|_2$$

This is equivalent to $\{f_1 \dots f_n\}$ being a collection of linearly independent vectors or if the span is the whole of $\mathbb{C}^L$ to being a basis.
Frames turn out to be a little bit more exciting. Let's rewrite the frame condition for an (obviously finite) set of functions $\{f_1 \dots f_n\}$.

$$\begin{aligned} & & A\|f\|_2^2 \leq \sum_{k=1}^{n} |\langle f, f_k\rangle|^2 \leq B\|f\|_2^2 \\ \sum_{k=1}^{n} |\langle f, f_k\rangle|^2 &= \sum_{k=1}^{n} |f_k^* f|^2 = \\ &= \|(f_1^* f, \dots, f_n^* f)\|_2^2 = \|\underbrace{(f_1^*, \dots, f_n^*)}_{n\times L\text{-matrix C}} f\|_2^2 = \|\mathrm{C}f\|_2^2 \\ & & A\|f\|_2^2 \leq \|\mathrm{C}f\|_2^2 \leq B\|f\|_2^2 \end{aligned} \tag{3.4}$$

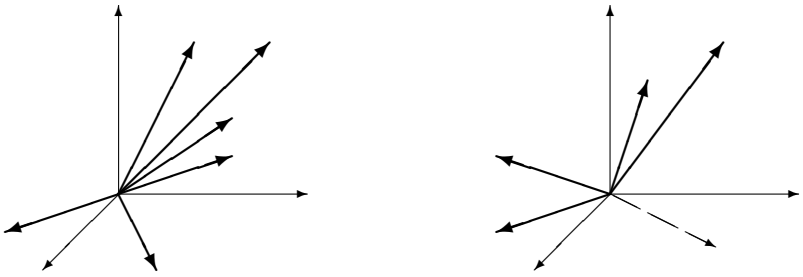
The first consequence of (3.4) is that n has to be larger than L because otherwise $\mathrm{C}f = 0$ for

$$f = (0, \ldots, 0, \underbrace{1}_{n+1}, \ldots, 1) \neq 0$$

Further since $\mathrm{C}f = 0 \Leftrightarrow f = 0$ the matrix C, that corresponds to the coefficient mapping, has to have full rank, i.e., $\mathrm{rk}(\mathrm{C}) = L$. Therefore in $\mathbb{C}^L$ a frame is any generating system.
Another conclusion that can be drawn from C having full rank is that the $L \times L$-matrix $\mathrm{S} = \mathrm{C}^*\mathrm{C}$, corresponding to the frame operator, has full rank and thus is invertible, as would be expected in analogy to the continuous case.
For numerical computations the frame bounds have an interesting relevance to the condition number $\kappa_2(\mathrm{S})$ of S:

$$\kappa_2(\mathrm{S}) = \kappa_2(\mathrm{C}^*\mathrm{C}) = \frac{\text{largest singular value(S)}}{\text{smallest singular value(S)}} = \Big(\frac{\mathrm{lsv(C)}}{\mathrm{ssv(C)}}\Big)^2 \leq \frac{B}{A}$$

Tight frames, where the frame operator is a multiple of the identity matrix, obviously have condition number $\kappa_2(\mathrm{S}) = 1$. Consequently the closer a frame is to being tight, the smaller is the condition number and thus the inversion of the frame operator, the calculation of the dual frame and the frame expansion are more stable.
Naively there are two main reasons for a frame being "bad". Either the lower frame bound A is small, because one "direction" in $\mathbb{C}^L$ is under-represented, or the upper bound B is large, because one direction is over-represented.



B large A small

Figure 3.2: Bad Frames

Both cases have direct consequences for Gabor frames. See Table 3.1 later.
Something else that is interesting for numerical calculations is the redundancy, $red = \frac{n}{L}$, that measures the percentage of frame elements which could theoretically be discarded. To keep computational cost low it is obviously desirable to keep the redundancy as close to one as possible. However, as will also become clear later, there is a trade off between the redundancy of a Gabor frame on one hand and the stability of inverting the frame operator as well as the approximation qualities of the generated Riesz basis on the other hand.

3.3 Gabor Frames

The last ingredient we need for the study of Gabor frames in $\mathbb{C}^L$ is the concept of a lattice. In the continuous case a lattice was a subgroup of the cartesian product of the time with the frequency domain of our signals, i.e., $\mathbb{R}^d \times \widehat{\mathbb{R}}^d$. This also holds for the discrete setting. There the time and the frequency domain are both $\mathbb{Z}_L$ and a subgroup of $\mathbb{Z}_L \times \hat{\mathbb{Z}}_L$ has the form $a\mathbb{Z}_L \times b\hat{\mathbb{Z}}_L$, where a,b are both divisors of L. Now if we set $N = \frac{L}{a}$ and $M = \frac{L}{b}$ we see that a discrete Gabor frame is a set of the form:

$$(g_{n,m})_{n,m} = \{\mathrm{M}_{mb}\mathrm{T}_{na}g,\ m = 0 \dots M-1,\ n = 0 \dots N-1\} \tag{3.5}$$

From the previous section we know that a necessary condition for $(g_{n,m})_{n,m}$ being a frame is that it has at least L elements or equivalently that the redundancy $red \geq 1$. So we need to choose our lattice constants a,b such that

$$1 \leq red = \frac{n}{L} = \frac{N \cdot M}{L} = \frac{\frac{L}{a} \cdot \frac{L}{b}}{L} = \frac{L}{ab} \tag{3.6}$$

To find out what else should be taken into account for the choice of lattice constants, we will calculate dual and tight frames for three differently scaled Gaussians on varying lattices and then compare the condition number of the frame operator.

3.3.1 The Dual Atom

To compute the dual frame we need to solve $\mathrm{S}\tilde{g} = g$ for the frame operator S. If we calculate the single entries $\mathrm{S}_{j,k}$ we see that S has a special structure.

$$\mathrm{S} = \mathrm{C}^*\mathrm{C} = \sum_{n,m} g_{n,m} \cdot g^*_{n,m} \tag{3.7}$$

$$\begin{aligned}
\mathrm{S}_{j,k} &= \sum_{n,m} g_{n,m}(j)\overline{g_{n,m}(k)} = \\
&= \sum_{n,m} e^{\frac{2\pi imbj}{L}} g(j-na) e^{\frac{-2\pi imbk}{L}} \overline{g(k-na)} = \\
&= \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} e^{\frac{2\pi imb(j-k)}{L}} g(j-na)\overline{g(k-na)}
\end{aligned}$$

The inner sum $\sum_{m=0}^{M-1} \left(e^{\frac{2\pi ib(j-k)}{L}}\right)^m$ is zero unless $|j-k|$ is a multiple of M and so we can write S in the following way, known as the Walnut representation.

$$\mathrm{S}_{j,k} = \begin{cases} M \cdot \sum_{n=0}^{N-1} g(j-na)\overline{g(k-na)} & (j-k)mod_M = 0 \\ 0 & \text{else} \end{cases} \tag{3.8}$$

From 3.8 we can see that S is a sparse matrix, i.e., only on every Mth (side) diagonal there are non zero entries. This suggests to use the conjugate gradient method (cgm), an iterative algorithm that basically needs only one matrix vector product in each step, for solving $\mathrm{S}\tilde{g} = g$. For details about cgm see [13]. A ready to implement algorithm can be found in [20], Theorem 8.6.1, where also details on how to use the a-periodicity of the diagonals and possible other structure are described.

3.3.2 The Tight Atom

In Section 2.4 we have seen that Gabor multipliers based on tight frames are of special importance. From (1.10) we know that whenever $(\pi(\lambda)g)_{\lambda\in\Lambda}$ is a frame $(\pi(\lambda)\mathrm{S}^{-\frac{1}{2}}g)_{\lambda\in\Lambda}$ is a tight frame. Unfortunately however calculating the square root of the matrix S is much too costly. So here we will present an iterative algorithm that is discussed in detail in [16]. Starting with an atom g_0 choose a function φ that is positive and analytic on an open neigbourhood of $[A,B] \supseteq \sigma(\mathrm{S}_0)$ and set $g_1 = \varphi(\mathrm{S}_0)g_0$. Now since $\varphi(\mathrm{S}_0)$ and S_0 commute with each other and all TF-shifts we have:

$$\begin{aligned}\mathrm{S}_1 f &= \sum_{\lambda\in\Lambda}\langle f,\pi(\lambda)\varphi(\mathrm{S}_0)g\rangle\pi(\lambda)\varphi(\mathrm{S}_0)g = \\ &= \varphi(\mathrm{S}_0)\sum_{\lambda\in\Lambda}\langle\varphi(\mathrm{S}_0)f,\pi(\lambda)g\rangle\pi(\lambda)g = \varphi(\mathrm{S}_0)\mathrm{S}_0\varphi(\mathrm{S}_0)f = \varphi^2(\mathrm{S}_0)\mathrm{S}_0 f\end{aligned}$$

By the spectral mapping theorem $(\pi(\lambda)g_1)_{\lambda\in\Lambda}$ is again a frame with frame bounds:

$$A_1 = \min\sigma(\mathrm{S}_1) \geq \min_{s\in[A,B]}\varphi^2(s)s > 0$$
$$B_1 = \max\sigma(\mathrm{S}_1) \leq \max_{s\in[A,B]}\varphi^2(s)s < \infty$$

Additionally it has the same associated tight atom as the original frame.

$$\mathrm{S}_1^{-\frac{1}{2}}g_1 = \left(\varphi^2(\mathrm{S}_0)\mathrm{S}_0\right)^{-\frac{1}{2}}\varphi(\mathrm{S}_0)g = \mathrm{S}_0^{-\frac{1}{2}}g_0$$

The idea now is to find a simple (sequence of) functions φ_k that shrink the spectrum, so that repeating the process and setting $g_{k+1} = \varphi_k(\mathrm{S}_k)g_k$ we have $\frac{B_{k+1}}{A_{k+1}} < \frac{B_k}{A_k}$. Thus g_{k+1} generates a tighter frame than g_k and if we can achieve $\frac{B_k}{A_k} \to 1$ the g_k converge to a/the tight atom.

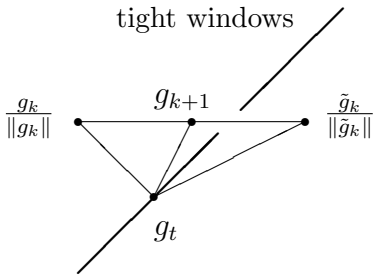


Figure 3.3: The Canonical Tight Window

For choosing φ_k note that the canonical dual $\tilde{g}$ minimizes $\|\frac{g}{\|g\|} - \frac{\gamma}{\|\gamma\|}\|$ among all duals γ and the associated tight atom g_t minimizes $\|g-h\|$ among all normalized tight windows h. So as illustrated by Figure (3.3) for:

$$g_{k+1} = \frac{1}{2}\Big(\frac{g_k}{\|g_k\|} + \frac{\tilde{g}_k}{\|\tilde{g}_k\|}\Big) \tag{3.9}$$

g_{k+1} should be closer to g_t than g_k and so formally we can set $\varphi_k(\mathrm{S}) = \frac{1}{2}(\frac{1}{\|g_k\|}\mathrm{Id} + \frac{1}{\|\tilde{g}_k\|}\mathrm{S}^{-1})$. In [16] is shown that recursion (3.9) always converges quadratically. However it is quite costly because in each step a dual atom has to be calculated. For g already close to being tight $\varphi_k(\mathrm{S}) = \frac{1}{2}(\frac{1}{\|g_k\|}\mathrm{Id} + \frac{1}{\|\mathrm{S}g_k\|}\mathrm{S})$ also gives quadratic convergence and so its best to use (3.9) only for the early steps.

Since it is not really the subject of the thesis there are no m-files for the above algorithms in the appendix. A good source for code is for instance [18] and should all else fail use `gd=g/S` for the dual and `gt = g*sqrtm(inv(S))` for the tight atom.

Coming back to the original problem of choosing the lattice constants Table 3.1 shows redundancy and condition number for various Gabor frames.

The conclusions that can be drawn from that are that although the critical case redundancy 1 should be avoided, redundancy between 1 and 2 already gives satisfactory results. Also it seems best to choose lattice constants that reflect the shape of the STFT of the atom. The reason for that is that for e.g. $a = 6$, $b = 16$ and the dilated Gaussian the frequency directions are under-represented, as illustrated by Figure 3.2. Similarly the condition number for the system $(g, a, 8)$ does not improve by changing a from 12 to 1 because the time directions just become over-represented compared to the frequency directions.
So even though situations later may call for higher redundancy or extreme lattices it is good to keep the above guidelines in mind.

3.4 Gabor Multipliers

At the beginning of Chapter 2 Gabor multipliers were motivated as reconstructions of a signal with processed coefficients. Indeed when applying a Gabor multiplier to a signal a more efficient way than constructing the matrix and doing the matrix vector product is to sample the signal via the STFT, do a pointwise multiplication of the coefficients with the multiplier matrix and then do the reconstruction. Taking things step by step let's start with the implementation of the STFT.

3.4.1 STFT

Most of the formulas in Chapter 2 do not require the values of the STFT on the whole time-frequency plane, which would correspond to a=b=1 in the discrete case but only on a sublattice.

$$\mathcal{V}_g f(x,\omega) = \mathcal{F}(f \cdot \mathrm{T}_x \bar{g})(\omega) \quad \Leftrightarrow \quad \mathrm{STFT}\, g(f)(na, mb) = \widehat{(f\, \mathrm{T}_{na}\, \bar{g})}(mb) \tag{3.10}$$

So to save computational effort it is a good idea not to blindly implement the discretized version (3.10) of the STFT but to use the "folding" trick. Writing $y =$

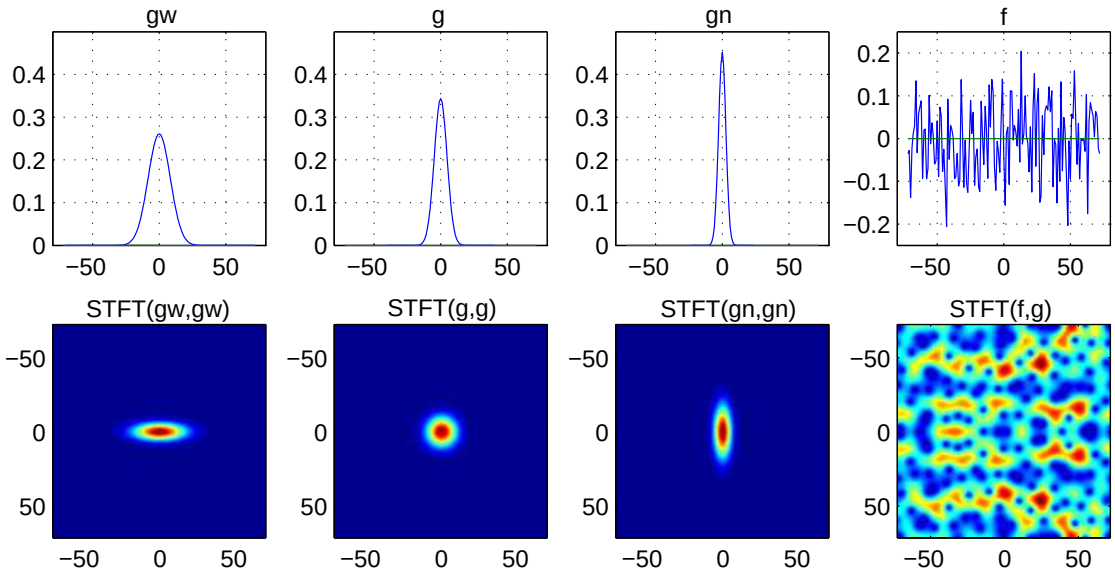


Figure 3.4: 4 Windows and their STFTs

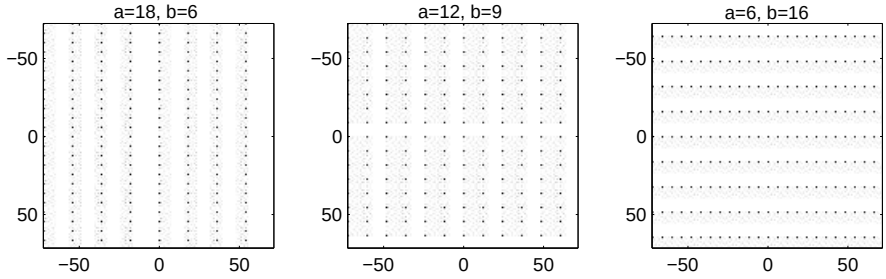


Figure 3.5: 3 Lattices

Condition number κ_2(S)

Red.	(a/b)	gw	g	gn	f
1	(12/12)	singular	singular	singular	19916
1	(9/16)	36053	137	42	8195
1.3	(12/9)	6.2	2.5	56	438
1.3	(18/6)	2.0	17	20125	5590
1.5	(12/8)	4.1	2.4	56	136
1.5	(6/16)	2174	8.1	1.8	142
2	(9/8)	4.1	1.4	6.2	91
3	(6/8)	4.1	1.1	1.6	158
18	(1/8)	4.1	1.1	1.0	1.6

Table 3.1:

$f \cdot \mathrm{T}_{na}\,\bar{g}$ for shortage, STFT $g(f)(na, mb)$ takes the form $\hat{y}(mb)$ and we have:

$$\begin{aligned}
\hat{y}(mb) &= \sum_{k=0}^{L-1} y(k) e^{-2\pi i \frac{kmb}{L}} = \\
&= \sum_{j=0}^{M-1} \sum_{k=0}^{b-1} y(j+kM) e^{-2\pi i \frac{(j+kM)m}{M}} = \qquad \left(y_f(j) = \sum_{k=0}^{b-1} y(j+kM)\right) \\
&= \sum_{j=0}^{M-1} y_f(j) e^{-2\pi i \frac{jm}{M}} = \hat{y}_f(m)
\end{aligned}$$

3.4.2 Gabor Synthesis

Since taking the STFT of a signal is just the coefficient mapping $f \to \mathrm{C}f =: d$ in disguise the reconstruction of f corresponding to the adjoint mapping $d \to C^* d$ should be realizable with the same amount of effort. Let $C = (C_0 \dots C_{N-1})$ be the coefficient matrix.

$$\begin{aligned}
f(k) &= \sum_{n,m} C(n,m) g_{n,m}(k) = \\
&= \sum_{n,m} C(n,m) g(k-na) e^{2\pi i \frac{kmb}{L}} = \\
&= \sum_{n=0}^{N-1} g(k-na) \sum_{m=0}^{M-1} C(n,m) e^{2\pi i \frac{kmb}{M}} = \\
&= \sum_{n=0}^{N-1} g(k-na) \check{C}_n(k/mod_M) = \langle g_k, \tilde{C}^{k/mod_M} \rangle
\end{aligned}$$

where $\tilde{C} := (\check{C}_0 \dots \check{C}_{N-1}) = (\tilde{C}^0 \dots \tilde{C}^{M-1})$.

For some purposes, not the least to have a look at them, it may still be necessary to calculate the actual Gabor multiplier matrices.

3.4.3 Synthesis of a Gabor Multiplier

If $C = (C_0 \dots C_{N-1})$ is the matrix that stores the upper symbol, the entries of the Gabor multiplier have the following form:

$$\begin{aligned}
\mathrm{G}_{\boldsymbol{m}}(j,k) &= \sum_{n,m} C(n,m) g_{n,m}(j) \bar{g}_{n,m}(k) = \\
&= \sum_{n,m} g(j-na) e^{2\pi i \frac{jmb}{L}} \bar{g}(k-na) e^{-2\pi i \frac{kmb}{L}} = \\
&= \sum_{n=0}^{N-1} g(j-na) \bar{g}(k-na) \sum_{m=0}^{M-1} C(n,m) e^{-2\pi i \frac{k-j}{M}} = \\
&= \sum_{n=0}^{N-1} g_n(j) \bar{g}_n(k) \hat{C}_n(k-j/mod_M)
\end{aligned}$$

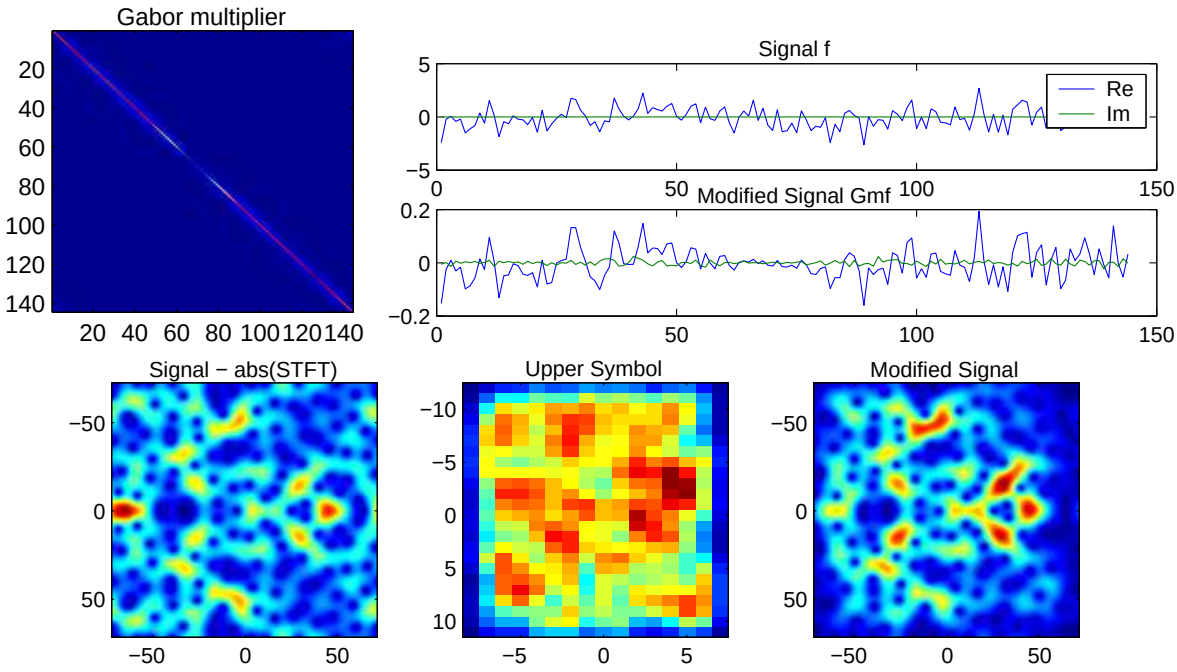


Figure 3.6: Application of a Gabor Multiplier 1

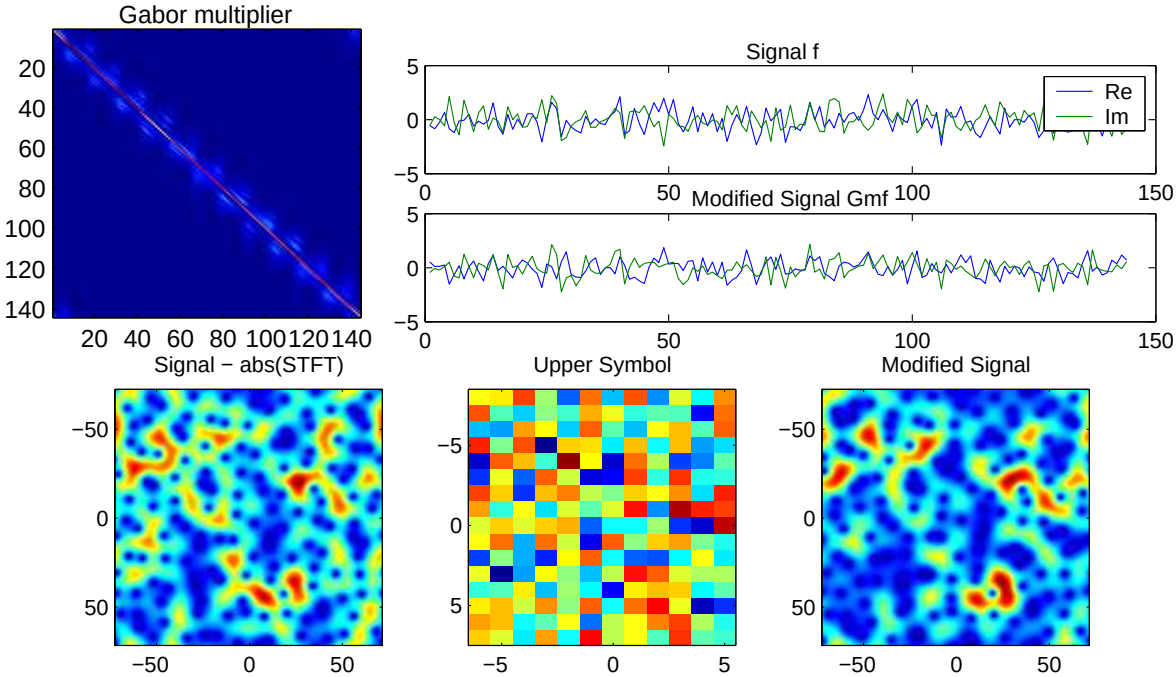


Figure 3.7: Application of a Gabor Multiplier 2

If $\tilde{C}_n$ denotes the matrix with entries $\tilde{C}_n(j,k) = \hat{C}_n(k-j/mod_M)$ $\mathrm{G}_{\boldsymbol{m}}$ can be written as:

$$\mathrm{G}_{\boldsymbol{m}} = \sum_{n=0}^{N-1} (g_n \otimes g_n^*) \cdot_{pw} \tilde{C}_n = \sum_{n=0}^{N-1} \mathrm{P}_n \cdot_{pw} \tilde{C}_n \tag{3.11}$$

With the routines derived from the above considerations it is now possible to simulate the convergence properties for Gabor multipliers in Corollary 2.2 and Theorem 2.8.

Experiment 3.1 *(Dependence on the Multiplier and Windows).* In the experiment the window g, a normalized Gaussian of length 144, and a randomly generated multiplier on the lattice $\Lambda = 9\mathbb{Z}_{144} \otimes 12\mathbb{Z}_{144}$ were perturbed with a normally distributed noise of relative size ε in the Euklidean norm, i.e., :

$$\|g_\varepsilon - g\|_2 = \varepsilon \|g\|_2 \quad \text{and} \quad \|m_\varepsilon - m\|_2 = \varepsilon \|m\|_2$$

Then the Gabor multipliers based on the original and the jittered building blocks, $\tilde{g}_\varepsilon$ denotes the dual window for g_ε, were compared. The error was calculated as the Hilbert Schmidt (Frobenius) norm of $\mathrm{G} - \mathrm{G}_\varepsilon$ divided by the $\mathcal{HS}$-norm of G.

	$\mathrm{G}_{g,\boldsymbol{m},\Lambda}$			$\mathrm{G}_{g,\tilde{g},\boldsymbol{m},\Lambda}$	
ε-noise	g	$\boldsymbol{m}$	$g, \boldsymbol{m}$	$g, \tilde{g}$	$g, \tilde{g}, \boldsymbol{m}$
1.0000	1.6357	0.9686	2.5087	1.4576	2.1649
0.1000	0.1415	0.0999	0.1711	0.1425	0.1780
0.0100	0.0140	0.0098	0.0170	0.0144	0.0174
0.0010	0.0014	0.0010	0.0017	0.0015	0.0018
10^{-14}	$1.40 \cdot 10^{-14}$	$9.82 \cdot 10^{-15}$	$1.70 \cdot 10^{-14}$	$1.49 \cdot 10^{-14}$	$1.78 \cdot 10^{-14}$

Table 3.2: Results of Experiment 3.1

As can be seen in Table 3.2 the error depends linearly, with relatively small constants, on the magnitude of the jitter, which is as good as predicted by Corollary 2.2.

Experiment 3.2 *(Dependence on the Lattice).* The window used was again a normalized Gaussian (this time of length 120). However since the continuous theory makes use of a multiplier in $\boldsymbol{W}(\boldsymbol{C}, \boldsymbol{\ell}^2)$, $\boldsymbol{m}$, defined on the full TF-plane, was chosen to be smooth. To be able to compare the Gabor multipliers based on different lattices it was also necessary to scale them with the redundancy of the lattice.

$$error = \|\tfrac{a_0 b_0}{L} \mathrm{G}_{a_0,b_0} - \tfrac{a_k b_k}{L} \mathrm{G}_{a_k,b_k}\|_{\mathcal{HS}} \Big/ \|\tfrac{a_0 b_0}{L} \mathrm{G}_{a_0,b_0}\|_{\mathcal{HS}}$$

The results are displayed in Table 3.3 and Figure 3.9. There the difference of the Gabor multipliers is plotted against the scaled Euklidean norm of the difference of the lattice constants, $1/L\|(a_0,b_0)-(a_k,b_k)\|_2$, where a negative sign indicates higher redundancy of (a_k,b_k) than that of (a_0,b_0).

Again the results confirm the findings of the underlying theorem (i.e., 2.8). An unexpected though not bad effect is that as the lattice constants a_k, b_k go to a_0, b_0

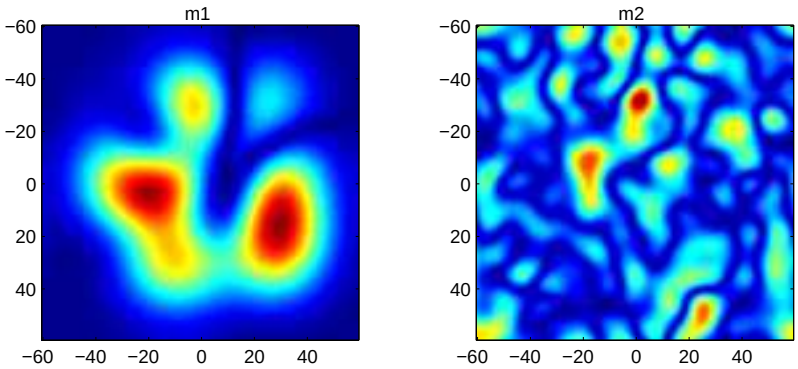


Figure 3.8: Multipliers for Experiment 3.2

$a_0 = b_0 = 2$

a/b	6/6	5/6	5/5	4/5	4/4	3/4	3/3	2/3
m1	0.0164	0.0134	0.0048	0.0046	0.0019	0.0020	0.0008	0.0007
m2	0.0871	0.0739	0.0451	0.0361	0.0192	0.0153	0.0074	0.0062

$a_0 = b_0 = 5$

a/b	3/3	3/4	4/4	4/5	5/6	6/6	6/8	8/8
m1	0.0051	0.0064	0.0063	0.0019	0.0159	0.0187	0.0898	0.1221
m2	0.0463	0.0503	0.0498	0.0277	0.0838	0.1045	0.1984	0.2442

Table 3.3: Results of Experiment 3.2

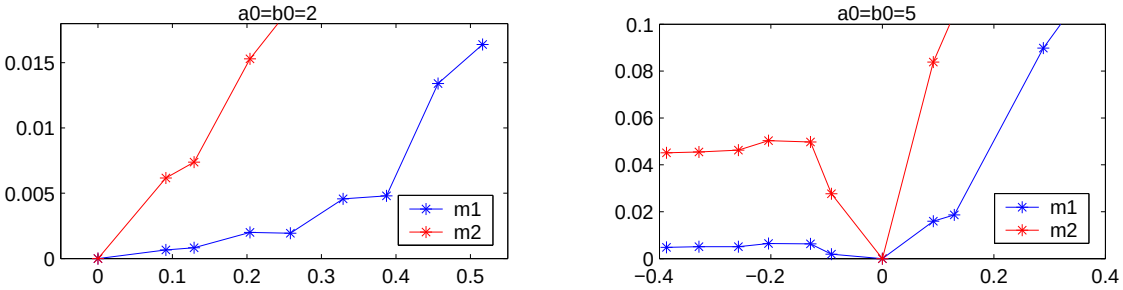


Figure 3.9: Convergence Curves

from below the difference in the corresponding Gabor multipliers starts out much less than for convergence from above but remains almost constant till the final improvement. The same curve shape even results from a less smooth multiplier where the samples vary more heavily.

3.4.4 $(\mathrm{P}_\lambda)_{\lambda\in\Lambda}$ as a Riesz Basis

From Theorem 2.10 we know that the system (g, a, b) will generate a Riesz basis if the Λ-Fourier transform of $(|\mathcal{V}_g g(\lambda)|^2)_{\lambda\in\Lambda}$ is bounded away from zero. Translating that into the discrete we therefore have to check if for the matrix $S_{n,m} = \mathrm{STFT}\, g(g)(na, mb)$, as defined above, there holds:

$$\min_{n,m} |\texttt{fft2}(S)_{n,m}| > c > 0$$

Remembering on the other hand that a Riesz basis is just a linearly independent set of vectors this should be no problem as long as the number of vectors $(M \cdot N)$ compared to the dimension of the space $(L \times L)$ is not too large.
Therefore the question we try to answer with the next experiment is up to which redundancy will a system (g, a, b) be likely to generate a Riesz basis provided that both the atom and the lattice constants are chosen reasonably nice (compare Experiment 4.3)?

Experiment 3.3 *(Riesz Basis Property).* The simple idea is to find out the limiting redundancy for which systems consisting of a normalized Gaussian and a more or less quadratic lattice will still generate a Riesz basis.

Table 3.4: Results of Experiment 3.3

L	limiting (a/b)	Redundancy
120	(2/3)	20
144	(2/4)	18
360	(4/4)	22.25
720	(6/6)	20
1080	(6/9)	20

The rule that can be deduced from Table 3.3 is that systems with redundancy up to 20 will generate a Riesz basis.
Looking at the generating atom of the biorthogonal basis Q_0, Figure 3.10, illustrates the break down of the Riesz basis property. So for low redundancy there is still enough room in the space of all $L \times L$-matrices for $(\mathrm{P}_\lambda)_{\lambda\in\Lambda}$ to be almost orthogonal and thus the shape of Q_0 still strongly resembles that of P_0. As the redundancy however increases the space becomes more "crowded" and in order to generate the biorthogonal basis Q_0 has to take account of an ever larger number of operators and hence becomes more and more de-fragmented.

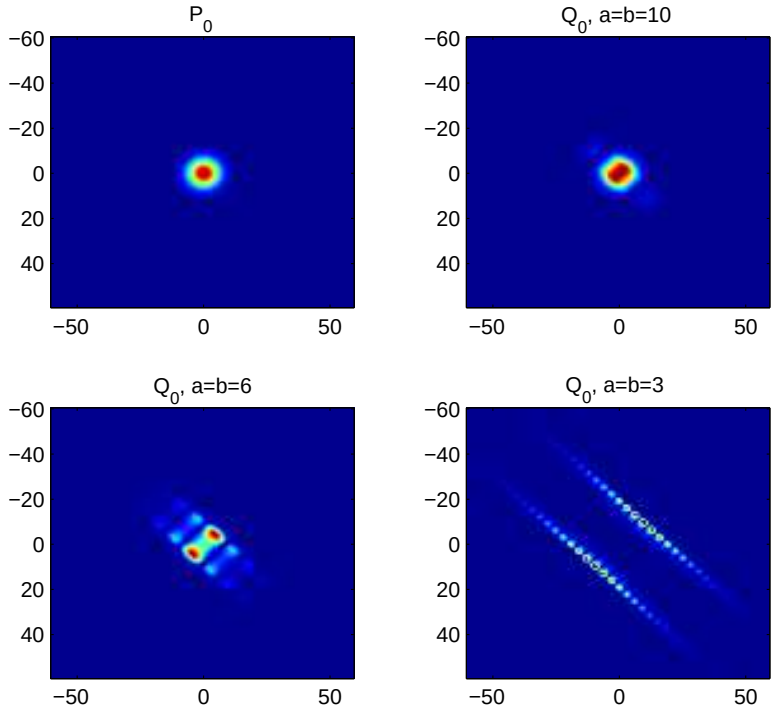


Figure 3.10: Biorthogonal Atom for Various Redundancies

3.4.5 The Best Approximation

Compare [7]. From Theorem 2.15 we have two formulas for the best approximation of an operator.

$$\mathrm{K}_{app} = \sum_{\lambda\in\Lambda} \langle \mathrm{K}, \mathrm{P}_\lambda \rangle \mathrm{Q}_\lambda = \sum_{\lambda\in\Lambda} \langle \mathrm{K}, \mathrm{Q}_\lambda \rangle \mathrm{P}_\lambda$$

The problem with them however is that they either require a sum over or inner products with the elements Q_λ of the biorthogonal basis that does not consist of rank-1-operators like $(\mathrm{P}_\lambda)_{\lambda\in\Lambda}$. Fortunately there is a way to avoid any "direct contact" with $(\mathrm{Q}_\lambda)_{\lambda\in\Lambda}$. Writing $c(\lambda) = \langle \mathrm{K}, \mathrm{P}_\lambda \rangle$ and $\boldsymbol{m}_Q$ for the upper symbol of Q_0 we see that the KNS of K_{app} has the form:

$$\begin{aligned} \sigma(\mathrm{K}_{app}) &= \sum_{\lambda\in\Lambda} c(\lambda)\, \mathrm{T}_\lambda\, \sigma(\mathrm{Q}_0) = c *_\Lambda \sigma(\mathrm{Q}_0) = \\ &= c *_\Lambda (\boldsymbol{m}_Q *_\Lambda \sigma(\mathrm{P}_0)) = (c *_\Lambda \boldsymbol{m}_Q) *_\Lambda \sigma(\mathrm{P}_0) \end{aligned}$$

On the other hand $\sigma(\mathrm{K}_{app}) = \boldsymbol{m}_{app} *_\Lambda \sigma(\mathrm{P}_0)$ and so by comparison $\boldsymbol{m}_{app} = c *_\Lambda \boldsymbol{m}_Q$. Finally applying the Λ-Fourier transform (a 2-dim. FFT) and plugging in the formula for the upper symbol $\boldsymbol{m}_Q$ of Q_0 gives: $(\gamma = |\mathcal{V}_g g_{|\Lambda}|^2)$

$$\hat{\boldsymbol{m}}_{app} = \hat{c} \cdot \hat{\boldsymbol{m}}_Q = \hat{c}/\hat{\gamma}$$

For the calculation of the lower symbol $c(\lambda) = \langle \mathrm{K}, \mathrm{P}_\lambda \rangle$ again a folding trick similar to the one for the STFT can be used.

$$
\begin{aligned}
C(n,m) &= \langle \mathrm{K}, g_{n,m} \otimes g^*_{n,m} \rangle = \langle \mathrm{K} g_{n,m}, g_{n,m} \rangle = \\
&= \sum_{j,k=0}^{L-1} \mathrm{K}(j,k) g(k-na) e^{2\pi i \frac{kmb}{L}} \bar{g}(k-na) e^{-2\pi i \frac{jmb}{L}} = \\
&= \sum_{j,k=0}^{L-1} \underbrace{\mathrm{K}(j,k) g(k-na) \bar{g}(k-na)}_{B_n(j,k)} e^{-2\pi i \frac{(j-k)m}{M}} = \\
&= \sum_{j,k=0}^{M-1} e^{-2\pi i \frac{(j-k)m}{M}} \underbrace{\sum_{j',k'=0}^{b-1} B_n(j+Mj', k+Mk')}_{B'_n f(j,k)} = \\
&= \sum_{j,k=0}^{M-1} B'_n(j,k) e^{-2\pi i \frac{(j-k)m}{M}} = \\
&= \sum_{l=0}^{M-1} e^{-2\pi i \frac{lm}{M}} \underbrace{\sum_{l=j-k/mod_M} B'_n(j,k)}_{B''_n(l)} = \\
&= \sum_{l=0}^{M-1} B''_n(l) e^{-2\pi i \frac{lm}{M}} = \hat{B}''_n(m)
\end{aligned}
$$

Finally equipped with the algorithms developed from the above considerations we can turn to the further investigation of approximating operators by Gabor multipliers.

Chapter 4

Best Approximation by Gabor Multipliers

As promised at the end of Chapter 2 this chapter will deal with the questions which operators are well represented by a Gabor multiplier and how crucial is the choice of the approximating system (g, Λ)?
Answers to the latter question will come first - on the one hand in the form of an estimate of the difference between the approximations of a general operator K in two close systems and on the other hand by explicitly calculating the approximation of a Gabor multiplier $\mathrm{G}_{m,g_1,\Lambda_1}$ in a system (g_2, Λ_2).

4.1 Approximation in Different Systems

In view of all convergence theorems in Chapter 2 it is only natural to ask if also the best approximation of an operator K depends continuously on the generating system (g, Λ). The answer at least for quadratic lattices is given by the next theorem.

Theorem 4.1. *Let g_k be a sequence of tight atoms on the quadratic lattices $\Lambda_k = a_k\mathbb{Z}^{2d}$ such that $g_k \to g_0$ in $\boldsymbol{S}_0$, $a_k \to a_0$ and each system (g_k, Λ_k) generates a Riesz basis. Then the best approximations $\mathrm{PG}_k(\mathrm{K})$ of a general Hilbert Schmidt operator K converge to $\mathrm{PG}_0(\mathrm{K})$ in the $\mathcal{HS}$-norm.*

Proof:
To get an idea for the proof it is helpful to rewrite $\mathrm{PG}_k(\mathrm{K})$ as:

$$\mathrm{PG}_k(\mathrm{K}) = \sum_{n\in\mathbb{Z}^{2d}} \underbrace{\langle \mathrm{K}, \pi_2(na_k)\mathrm{Q}_k\rangle}_{m_k(na_k)} \mathrm{P}_k = \mathrm{G}_{\boldsymbol{m}_k,g_k,\Lambda_k} \tag{4.1}$$

The above notation suggests to try and use Corollary 2.9. So what has to be shown is the the multipliers $\boldsymbol{m}_k$ can be interpreted as the samples on Λ_k of a sequence of function $\tilde{\boldsymbol{m}}_k$ on $\mathbb{R}^{2d}$, that converge in $\boldsymbol{W}(\boldsymbol{C}, \boldsymbol{\ell}^2)$.
Now defining $\psi(x) := \sigma(\mathrm{K})(-x)$, $\|\psi\|_2 = \|\mathrm{K}\|_{\mathcal{HS}}$ there holds:

$$\begin{aligned} m_k(na_k) &= \langle \mathrm{K}, \pi_2(na_k)\mathrm{Q}_k\rangle = \langle \sigma(\mathrm{K}), \mathrm{T}_{na_k}\,\sigma(\mathrm{Q}_k)\rangle = \\ &= \int_{\mathbb{R}^{2d}} \psi(-x)\overline{\sigma(\mathrm{Q}_k)(x-na_k)}dx = \Big(\psi * \overline{\sigma(\mathrm{Q}_k)}\Big)(-na_k) =: \tilde{m}_k(na_k) \end{aligned}$$

The rest of the proof is a consequence of relatively heavy machinery from theory for spline type Riesz bases, Theorem 7.2 in [10], saying that small changes of the P_0 will only result in small changes of Q_0, i.e., if $\|\sigma(\mathrm{P}_k) - \sigma(\mathrm{P}_0)\|_{\boldsymbol{W}(\boldsymbol{C},\boldsymbol{\ell}^1)} \leq C\|\sigma(\mathrm{P}_k) - \sigma(\mathrm{P}_0)\|_{\boldsymbol{S}_0} \leq C'\|g_k - g_0\|_{\boldsymbol{S}_0} \leq \varepsilon'$ then $\|\sigma(\mathrm{Q}_k) - \sigma(\mathrm{Q}_0)\|_{\boldsymbol{W}(\boldsymbol{C},\boldsymbol{\ell}^1)} \leq \varepsilon$. Together with a convolution relation for amalgam spaces of the form $\|f*g\|_{\boldsymbol{W}(\boldsymbol{C},\boldsymbol{\ell}^2)} \leq \|f\|_2\|g\|_{\boldsymbol{W}(\boldsymbol{C},\boldsymbol{\ell}^1)}$, compare (1.5), this yields the desired estimate.

$$\begin{aligned}\|\tilde{\boldsymbol{m}}_k - \tilde{\boldsymbol{m}}_0\|_{\boldsymbol{W}(\boldsymbol{C},\boldsymbol{\ell}^2)} &\leq \|\psi * \left(\overline{\sigma(\mathrm{Q}_k) - \sigma(\mathrm{Q}_0)}\right)\|_{\boldsymbol{W}(\boldsymbol{C},\boldsymbol{\ell}^2)} \leq \\ &\leq \|\psi\|_2 \cdot \|\sigma(\mathrm{Q}_k) - \sigma(\mathrm{Q}_0)\|_{\boldsymbol{W}(\boldsymbol{C},\boldsymbol{\ell}^1)} \leq \|\mathrm{K}\|_{\mathcal{HS}} \cdot \varepsilon\end{aligned}$$

□

Because there is no reason - other than that the theorem used is only valid for the quadratic case - why convergence should hold only for quadratic lattices in the obligatory numerical simulation the limiting system will make use of a rectangular lattice.

Experiment 4.1 *(Convergence of the Best Approximation).* For 3 operators the approximations in systems with lattice constants going to $(a_0/b_0) = (4/5)$ and a Gaussian of length 120 were compared. The results are displayed in Table 4.1 and Figure 4.1. The interesting observation to be made is that the better an operator is represented by a Gabor multiplier - recovery in the system $(g, 4, 5)$ is 12.9% for K1, 45.3% for K2 and 90.4% for K3 - the more the convergence curve resembles that in Experiment 3.2. So for a/b going to a_0/b_0 from below the variance of the approximations is small even for large difference of the lattice constants but only slowly decreasing. Just perturbing the atom (compare Experiment 3.1) again gives linear convergence independently of the quality of the approximation.

a/b	2/3	3/3	3/4	4/4	5/5	5/6	6/6	6/8
K1	0.6277	0.4619	0.3573	0.2187	0.2025	0.3063	0.3535	0.4793
K2	0.2405	0.1845	0.1456	0.1076	0.1152	0.2020	0.2462	0.3898
K3	0.0946	0.0944	0.0903	0.0849	0.0971	0.1853	0.2294	0.3851

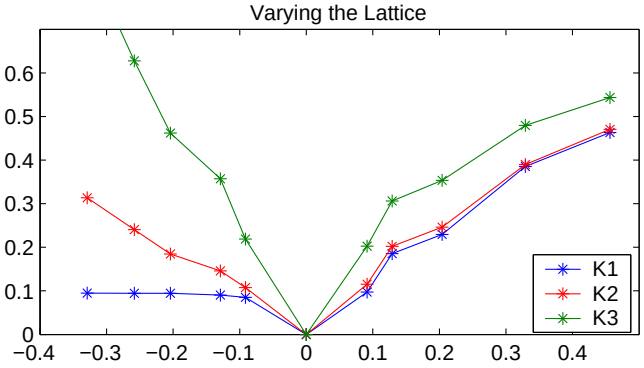
Table 4.1: Results of Experiment 4.1

While the last theorem shows that best approximations depend continuously on the underlying systems (g_k, Λ_k) and thus promises a certain stability for numerical calculation it gives no information about the difference in approximation quality of two entirely unrelated systems. This however is especially important for the discrete case where the shape of the atom and most of all the lattice constants have a direct influence on computational cost.

One way to compare two completely different systems is to calculate how well a Gabor multiplier based on the system (g, Λ) can be approximated within the system $(\tilde{g}, \tilde{\Lambda})$. For a start let's have a look at what happens by only changing the atom.

Lemma 4.2. *The difference between a Gabor multiplier* $\mathrm{G}_{\boldsymbol{m},g,\Lambda}$ *and its best approximation* G_{app} *in the system* $(\tilde{g}, \Lambda)$ *is given by a convolution over* Λ *of the multiplier* $\boldsymbol{m}$ *and the difference between* P_0 *and its best approximation* P_{app} *in* $(\tilde{g}, \Lambda)$*, i.e.,*

$$\|\mathrm{G}_{\boldsymbol{m},g,\Lambda} - \mathrm{G}_{app}\|_{\mathcal{HS}} \leq C\|\boldsymbol{m}\|_2 \cdot \|\sigma(\mathrm{P}_0) - \sigma(\mathrm{P}_{app})\|_{\boldsymbol{S}_0} \tag{4.2}$$



Varying the Atom

ε-noise of g	Difference of Approximation
0.5	0.4824
0.1	0.1455
0.01	0.0201
0.001	0.0020

Figure 4.1: Convergence of the Best Approximation

Proof: Let $\tilde{p}$ denote the upper symbol of P_{app}. Then the upper symbol $\boldsymbol{m}_{app}$ of G_{app} in the system $(\tilde{g}, \Lambda)$ can be calculated as:

$$\begin{aligned}
\boldsymbol{m}_{app}(\lambda') &= \langle \mathrm{G}_{\boldsymbol{m},g,\Lambda}, \tilde{\mathrm{Q}}_{\lambda'} \rangle = \\
&= \sum_{\lambda\in\Lambda} m(\lambda) \langle \mathrm{P}_\lambda, \tilde{\mathrm{Q}}_{\lambda'} \rangle = \\
&= \sum_{\lambda\in\Lambda} m(\lambda) \langle \mathrm{P}_0, \tilde{\mathrm{Q}}_{\lambda'-\lambda} \rangle = \\
&= \sum_{\lambda\in\Lambda} m(\lambda) \tilde{p}(\lambda'-\lambda) = \boldsymbol{m} *_\Lambda \tilde{p}(\lambda')
\end{aligned}$$

Therefore the KNS of G_{app} has the form:

$$\sigma(\mathrm{G}_{app}) = \boldsymbol{m}_{app} *_\Lambda \sigma(\tilde{\mathrm{P}}_0) = \boldsymbol{m} *_\Lambda \tilde{p} *_\Lambda \sigma(\tilde{\mathrm{P}}_0) = \boldsymbol{m} *_\Lambda \sigma(\mathrm{P}_{app})$$

Since P_0 has an $\boldsymbol{S}_0$-integration kernel so has its best approximation P_{app} and we can apply the same trick as in Theorem 2.8, i.e., we interpret the Λ-convolution with $\boldsymbol{m}$ as a normal convolution with the measure $\mu = \sum_{\lambda\in\Lambda} m(\lambda)\delta_\lambda$, where $\|\mu\|_{\boldsymbol{W}(\mathrm{M},\boldsymbol{\ell}^2)}$ can be estimated as: ($Q = [0,1]^{2d}$ and $C = \max_{j\in\mathbb{Z}^{2d}} \#\{\lambda \in j + Q\} < \infty$)

$$\begin{aligned}
\|\mu\|^2_{\boldsymbol{W}(\mathrm{M},\boldsymbol{\ell}^2)} &= \sum_{j\in\mathbb{Z}^{2d}} \Big(|\mu|(j+Q)\Big)^2 = \sum_{j\in\mathbb{Z}^{2d}} \Big(\sum_{\lambda\in j+Q} |m(\lambda)|\Big)^2 \leq \\
&\leq \sum_{j\in\mathbb{Z}^{2d}} C^2 \max_{\lambda\in j+Q} |m(\lambda)|^2 \leq C^2 \sum_{j\in\mathbb{Z}^{2d}} \sum_{\lambda\in j+Q} |m(\lambda)|^2 \leq C^2 \|\boldsymbol{m}\|_2^2
\end{aligned}$$

Therefore using the convolution relation (1.6) we get:

$$\begin{aligned}
\|\mathrm{G}_{\boldsymbol{m},g,\Lambda} - \mathrm{G}_{app}\|_{\mathcal{HS}} &= \|\sigma(\mathrm{G}_{\boldsymbol{m},g,\Lambda}) - \sigma(\mathrm{G}_{app})\|_2 = \\
&= \|\mu * \Big(\sigma(\mathrm{P}_0) - \sigma(\mathrm{P}_{app})\Big)\|_2 \leq \\
&\leq C' \|\boldsymbol{m}\|_2 \cdot \|\sigma(\mathrm{P}_0) - \sigma(\mathrm{P}_{app})\|_{\boldsymbol{S}_0}
\end{aligned}$$

□

Experiment 4.2 *(Optimal Atom for a Given Lattice).* The aim of the experiment was to determine the best atom for a given lattice and in the course of doing that verify to what extent estimate (4.2) gives an applicable criterion. So for a given lattice Λ the (average) approximation quality of P_0^1 and ten Gabor multipliers with random ($\boldsymbol{r}$), respectively smooth ($\boldsymbol{s}$) upper symbol in (g_2, Λ) were compared to the reversed situation. The Atoms g_1, g_2 were a normalized and a compressed Gaussian of size 144 (compare Figure 3.4).

Lattice	Approximation in (g_1, Λ)			Approximation in (g_2, Λ)		
(a/b)	P_0^2	$\mathrm{G}_{\boldsymbol{r}}^2$	$\mathrm{G}_{\boldsymbol{s}}^2$	P_0^1	$\mathrm{G}_{\boldsymbol{r}}^1$	$\mathrm{G}_{\boldsymbol{s}}^1$
(8/8)	0.3337	0.3309	0.3309	0.3708	0.3681	0.3658
(4/16)	0.3745	0.3735	0.4168	0.3460	0.3448	0.4167
(4/4)	0.1175	0.1162	0.0156	0.0517	0.0509	0.0153
(2/8)	0.1041	0.1033	0.0433	0.1637	0.1643	0.0434

Table 4.2: Results of Experiment 4.2

As the results in Table 4.1 show the approximation quality for Gabor multipliers with random upper symbols is almost the same as for P_0 while for smooth multipliers it is actually less for high redundancy but more for low redundancy.
In general the approximation quality is more or less the same and so rather a measure for the difference of the systems and thus a sign to carefully decide which one to use than an indicator of the better one. Still for high redundancy the atom whose STFT reflects the shape of the lattice gives better performance, while surprisingly for low redundancy the situation is reversed.

The analogue comparison of two systems with different lattices cannot be reduced to the difference of the central building block P_0^1 and its approximation in the other system. In [3] an explicit formula for the best approximation of a Gabor multiplier $\mathrm{G}_{m,g_1,\Lambda_1}$ in the system (g_2, Λ_2) can be found. As it however does not give much insight we will just do numerical experiments instead.

Experiment 4.3 *(Optimal Lattice for a Given Atom).* The basic situation was the same as in the last experiment but this time the atom, g_1 or g_2, was fixed and the average approximation of ten Gabor multipliers G_{Λ_1} with random/smooth upper symbol in the system with Λ_2 compared.

Atom	$\mathrm{G}_{(4/16)}$ in (8/8)	$\mathrm{G}_{(8,8)}$ in (4/16)	$\mathrm{G}_{(2/8)}$ in (4/4)	$\mathrm{G}_{(4/4)}$ in (2/8)
g_1	0.3001/0.0826	0.6431/0.5238	0.0167/0.0000	0.3042/0.0610
g_2	0.3883/0.1451	0.6366/0.3671	0.2123/0.0193	0.0443/0.0006

Table 4.3: Results of Experiment 4.3

This time the results in Table 4.1 clearly show that - independently of the redundancy or smoothness of the multiplier - the system with the lattice adapted to the shape to the STFT gives the better performance. So for the next section where suitable other operators are approximated by Gabor multipliers we will keep the guideline to harmonize atom and lattice in mind.

4.2 Approximation in Various Classes of Operators

In the previous section the difference of the best approximation of an operator in various systems was discussed. The really interesting question however concerns the difference between the operator and its approximation, because an estimate of that would also answer the question which operators can be well approximated by Gabor multipliers.
A look at the KNS of a Gabor multiplier

$$\sigma(\mathrm{G}_{\boldsymbol{m}}) = \sum_{\lambda\in\Lambda} m(\lambda)\mathrm{T}_\lambda\sigma(\mathrm{P}_0) \tag{4.3}$$

shows that as $\sigma(\mathrm{P}_0)$ is in $\boldsymbol{S}_0$ it has to be a rather smooth function, i.e., even if $\sigma(\mathrm{G}_{\boldsymbol{m}})$ is only in $\boldsymbol{S}_0'$ it will not be "wild" in the sense of the delta function, but only in the sense of not being integrable on $\mathbb{R}^d$, like the constant 1 or the pure frequencies $e^{2\pi ikx}$. So a heuristic conclusion is that operators with smooth KNS are well approximated by Gabor multipliers. The problem however is that "smoothness" is not only a rather vague but certainly also hard to check for. To get a more tangible criterion it is helpful to determine how much of the operator is lost in the approximation.

Theorem 4.3. *If (g,Λ) generates a tight Gabor Frame in $\boldsymbol{L}^2(\mathbb{R}^d)$ then the rank 1 operators $(g_\lambda \otimes {g_{\lambda'}}^*)_{\lambda,\lambda'}$ form a tight frame in $\mathcal{HS}$.*

Proof:
The basic idea is that for any operator K and any $f \in \boldsymbol{L}^2$ both f and $\mathrm{K}f$ have a Gabor expansion.

$$\begin{aligned}
\mathrm{K}f &= \mathrm{K}\sum_{\lambda\in\Lambda}\langle f,g_\lambda\rangle g_\lambda = \sum_{\lambda\in\Lambda}\langle f,g_\lambda\rangle \mathrm{K}g_\lambda = \\
&= \sum_{\lambda\in\Lambda}\langle f,g_\lambda\rangle \sum_{\lambda'\in\Lambda}\langle \mathrm{K}g_\lambda,g_{\lambda'}\rangle g_{\lambda'} = \\
&= \sum_{\lambda\in\Lambda}\sum_{\lambda'\in\Lambda}\langle \mathrm{K}g_\lambda,g_{\lambda'}\rangle g_{\lambda'}\langle f,g_\lambda\rangle = \\
&= \Big(\sum_{\lambda\in\Lambda}\sum_{\lambda'\in\Lambda}\langle \kappa(\mathrm{K}),g_{\lambda'}\otimes {g_\lambda}^*\rangle g_{\lambda'}\otimes {g_\lambda}^*\Big) f
\end{aligned} \tag{4.4}$$

Since

$$(g_\lambda \otimes {g_{\lambda'}}^*)(t,y) = e^{2\pi it\omega'} g(t-x')\cdot e^{2\pi iy\omega}\overline{g(y-x)} = \mathrm{M}_{(\omega',-\omega)}\,\mathrm{T}_{(x',x)}(g\otimes g^*)(t,y)$$

the convergence of the double sum in (4.4) is once again a consequence of Theorem 1.2 for $g_0 = g\otimes g^*$ and $\Lambda = a\mathbb{Z}^{2d}\times b\mathbb{Z}^{2d}$. □

A consequence of the above theorem is that all the information about an operator is incorporated in the coefficient matrix $\langle \mathrm{K}, g_\lambda\otimes {g_{\lambda'}}^*\rangle_{\lambda,\lambda'}$, whose diagonal is just the lower symbol $\sigma_L(\mathrm{PG(K)})$ of the operator K. Therefore the best approximation by a Gabor multiplier corresponds to the optimal reconstruction of the operator using the information on the diagonal but ignoring all off diagonal entries and the quality

of that is obviously the better the smaller this lost information was to begin with. So far of course this is still not an applicable criterion because it makes no sense to calculate side entries just in order to discard them with a good conscience. However there exists a nice estimate for them involving the spreading function $\eta(\mathrm{K})$.

Lemma 4.4.

$$|\langle \mathrm{K}, g_{\lambda'} \otimes {g_\lambda}^* \rangle| \leq \Big(|\eta(\mathrm{K})| * |\mathcal{V}_g g| \Big)(\lambda' - \lambda) \tag{4.5}$$

Proof: (Compare [17], 10.3.1)
The proof uses the trick that:

$$\begin{aligned} |\langle \mathrm{K}, g_{\lambda'} \otimes {g_\lambda}^* \rangle| &= |\overline{\langle \mathrm{K}, g_{\lambda'} \otimes {g_\lambda}^* \rangle}| = \\ &= |\langle \mathrm{K}^*, g_\lambda \otimes {g_{\lambda'}}^* \rangle| = |\langle \eta(\mathrm{K}^*), \eta(g_\lambda \otimes {g_{\lambda'}}^*) \rangle| \end{aligned}$$

From (2.7) it is relatively easy (as opposed to the messy direct way) to calculate $\eta(\mathrm{K}^*)$.

$$\begin{aligned} \mathrm{K}^* &= \Big(\int_{\mathbb{R}^{2d}} \eta(\mathrm{K})(t,\nu)\, \mathrm{M}_\nu\, \mathrm{T}_t\, dtd\nu \Big)^* = \int_{\mathbb{R}^{2d}} \overline{\eta(\mathrm{K})(t,\nu)}\, \mathrm{T}_t^*\, \mathrm{M}_\nu^*\, dtd\nu = \\ &= \int_{\mathbb{R}^{2d}} \overline{\eta(\mathrm{K})(t,\nu)}\, \mathrm{M}_{-\nu}\, \mathrm{T}_{-t}\, e^{-2\pi i\nu t} dtd\nu = \int_{\mathbb{R}^{2d}} \underbrace{e^{-2\pi i\nu t} \overline{\eta(\mathrm{K})(-t,-\nu)}}_{\eta(\mathrm{K}^*)}\, \mathrm{M}_\nu\, \mathrm{T}_t\, dtd\nu \end{aligned}$$

Using the formula for $\eta(g_\lambda \otimes {g_{\lambda'}}^*)$ from Experiment (2.2) we finally get:

$$\begin{aligned} |\langle \mathrm{K}, g_{\lambda'} \otimes {g_\lambda}^* \rangle| &\leq |\int_{\mathbb{R}^{2d}} e^{-2\pi i\nu t} \overline{\eta(\mathrm{K})(-t,-\nu)} e^{-2\pi i\varphi} \mathcal{V}_g g(\nu + \omega' - \omega, t + x' - x) dtd\nu \leq \\ &\leq \int_{\mathbb{R}^{2d}} |\eta(\mathrm{K})(-t,-\nu)| |\mathcal{V}_g g(\nu + \omega' - \omega, t + x' - x)| dtd\nu \leq \\ &\leq \Big(|\eta(\mathrm{K})| * |\mathcal{V}_g g| \Big)(\lambda' - \lambda) \end{aligned}$$

□

The practical conclusion that can be drawn from Lemma 4.5 is that if the spreading function of an operator is concentrated around zero the lost information in the off-diagonal coefficients is relatively small. Therefore a class of operators that should be well approximated by Gabor multipliers are the so called underspread operators.

4.2.1 Underspread Operators

An operator K is called underspread if the (essential) support of its spreading function is contained in a rectangle centered around zero, i.e.,

$$\mathrm{supp}(\eta(K)) \subseteq Q(t_0, \nu_0) = [-\frac{t_0}{2}, \frac{t_0}{2}] \times [-\frac{\nu_0}{2}, \frac{\nu_0}{2}]$$

such that $t_0\nu_0 < 1$.
Underspread operators were defined by W. Kozek in [17], where details including their approximability by Gabor multipliers can be found.
Before we can do a numerical experiment to see how well the approximation works

it is necessary to reformulate the underspread property for matrices. An $L \times L$-matrix corresponds to the approximation of the integration kernel of an operator by a Riemannian sum with increment $\frac{1}{\sqrt{L}}$ on an $\sqrt{L} \times \sqrt{L}$ square. Therefore it is underspread if its spreading matrix (see A.6) is non zero only on a $t_0 \times \nu_0$ submatrix centered around zero with

$$\frac{t_0}{\sqrt{L}} \cdot \frac{\nu_0}{\sqrt{L}} < 1 \quad \Leftrightarrow \quad t_0 \cdot \nu_0 < L$$

Experiment 4.4 *(Approximation of an Underspread Operator).* For the experiment L was chosen to be 144. To get an underspread operator a spreading matrix with a 7×7 complex non zero center was randomly generated and retransformed to the operator matrix K (see A.7). Then K was approximated in different systems where the atom was always a normalized Gaussian. The results of the experiment are displayed in Table 4.4. Figure 4.2 shows the operator, its spreading function and two approximations.

Table 4.4: Results of Experiment 4.4

a	b	Redundancy	Error
9	9	1.7778	0.1844
8	8	2.25	0.0976
6	6	4	0.0086
4	4	9	6.6233e-006
9	8	2	0.1469
6	12	2	0.3542

While the experiment confirms that underspread operators can be well approximated it also shows that one has to be careful. So looking at the results we see that the approximation quality is strongly influenced by the choice of the lattice, i.e., for redundancy 2 either about 85% or only 57% recovery. However the bad performance of the system $(g, 6, 12)$ is not completely unexspected. After all we have seen in the last section that the STFT of the atom should reflect the shape of the lattice. Nevertheless the main reason is that not only atom and lattice should be harmonized but they should also be adapted to the operator, i.e., the shape of the spreading function. The rule given in [17] for the optimal lattice (and consequently atom) is to have:

$$\frac{a}{b} \simeq \frac{t_0}{\nu_0}$$

Experiment 4.5 *(Adapting the System).* This time the matrix to be approximated had a spreading support of size 4×16. Then approximations in different systems, consisting of all combinations of a rectangular or a quadratic lattice with either a normalized (g_1) or a squeezed gaussian (g_2), were compared, see Figure 4.3.

The results in Table 4.5 show that as expected for low redundancy the best performance is gained from the all harmonized system, i.e., $(g_2, 4, 16)$. For high redundancy however there is almost no difference between the two systems where atom and lattice are synchronized, i.e., $(g_1, 4, 4)$ and $(g_2, 2, 8)$. So an important conclusion

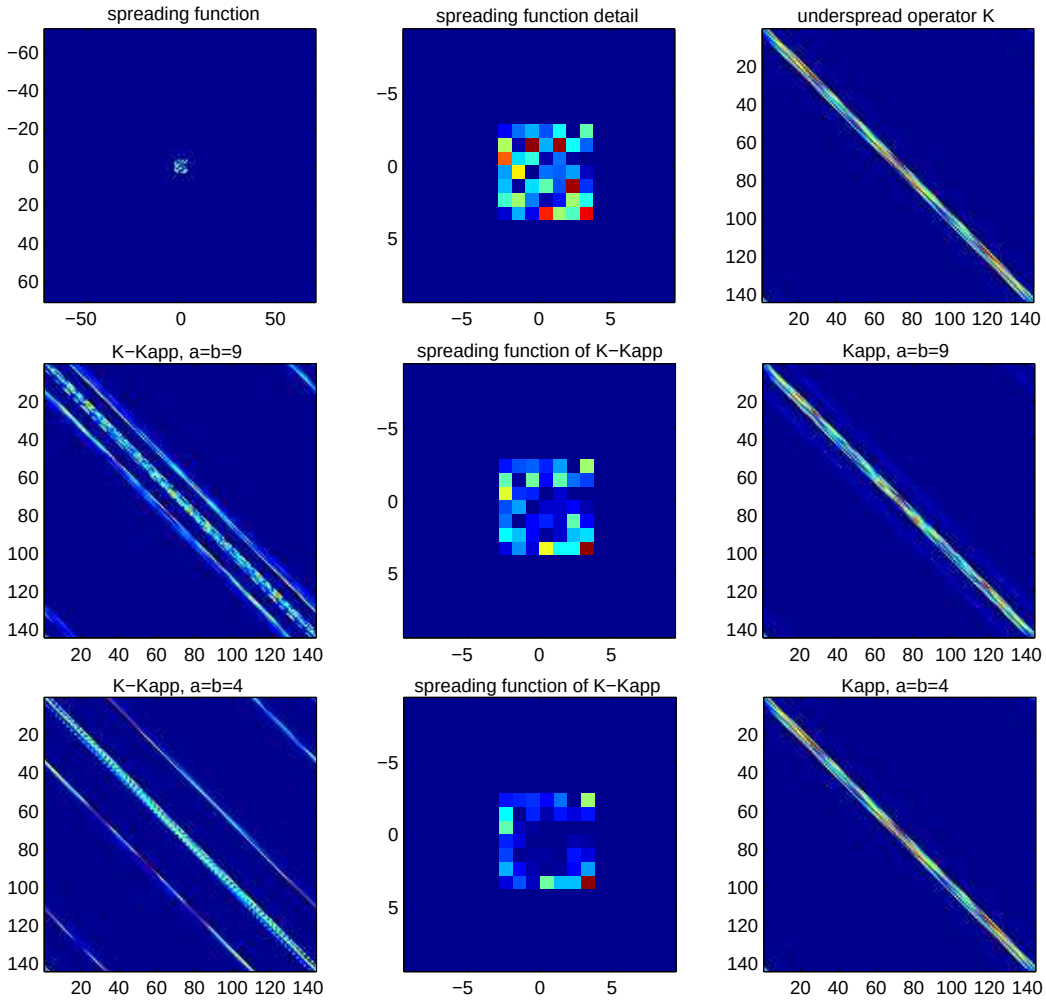


Figure 4.2: Approximation of an Underspread Operator

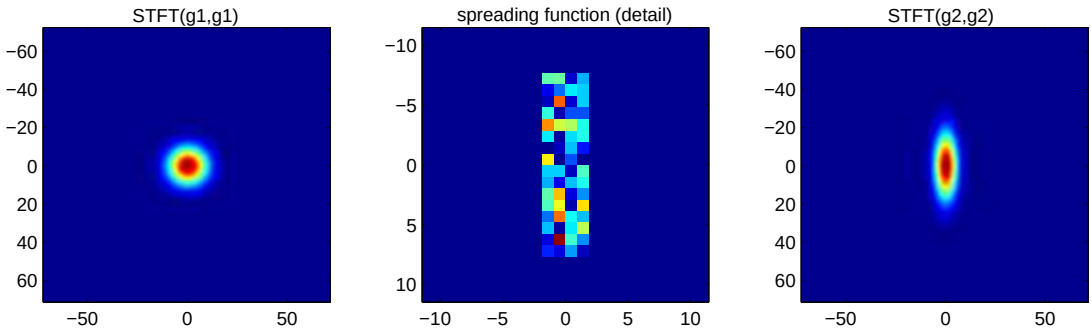


Figure 4.3: Spreading Function and Two Atoms

Table 4.5: Results of Experiment 4.5

(g, a, b)	g_1	g_2
(8/8)	0.2363	0.5043
(4/16)	0.5280	0.1597

(g, a, b)	g_1	g_2
(4/4)	0.0001	0.0344
(2/8)	0.0516	0.0001

that can be drawn is that it is much better to adjust the lattice to the atom than have only one of them adapted to the spreading support.

The aim of the final experiment with underspread operators is to find out some guidelines for the choice of redundancy depending on the spreading function and desired approximation quality and to determine how far we can go, i.e., see if it is also possible to approximate overspread matrices where $t_0 \cdot \nu_0 \geq L$.

Experiment 4.6. For the experiment operators of different size (L) and with varying shape of the spreading support (t_0, ν_0) were generated as in Experiment 4.4. Then was determined for which redundancy - always using a well adapted system - the recovery was more than 90%.

As an excerpt of the results in Table 4.6 shows more than 90% recovery is not only possible even for operators that are overspread to a degree of 25 to 30, but in some way they behave even better than truly underspread operators. So while all operators with $t_0\nu_0 < 1$ require redundancy between 2 and 4 for the desired approximation quality the necessary redundancy for overspread operators, especially when $t_0\nu_0 > 4$, seems related to the size of the spreading function by:

$$a < \frac{L}{\nu_0} \quad \text{and} \quad b < \frac{L}{t_0} \tag{4.6}$$

A combination of (4.6) with the rule for the limiting redundancy for which a system will still generate a Riesz basis (see Experiment 3.3) yields that stable approximation is possible for operators that are overspread to a degree of 20.

$$\frac{L}{ab} < 20 \qquad \overset{4.6}{\Longrightarrow} \qquad \frac{t_0\nu_0}{L} < 20$$

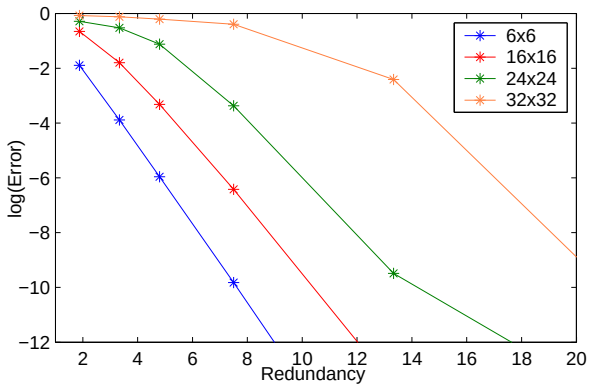


Figure 4.4: Convergence Rates ($L = 120$)

$L = 120$

$t_0 \times \nu_0$	$\frac{t_0\nu_0}{L}$	(a/b)	red	error	$(\frac{L}{\nu_0}/\frac{L}{t_0})$
3×3	0.075	(6/6)	3.33	0.0124	(40/40)
4×12	0.4	(4/12)	2.5	0.0791	(10/30)
11×11	1.0	(6/6)	3.33	0.0540	(10.9/10.9)
16×16	2.1	(5/5)	4.8	0.0417	(7.5/7.5)
19×19	3.0	(5/5)	4.8	0.0807	(6.3/6.3)
24×24	4.8	(4/4)	7.5	0.0301	(5/5)
15×45	5.6	(2/6)	10	0.0049	(2.7/8)
38×38	12.0	(3/3)	13.33	0.0929	(3.2/3.2)
52×52	20.8	$(2/2)^*$	30	0.0015	(2.3/2.3)
30×100	25	$(1/3)^*$	40	0.0176	(1.2/4)
60×60	30	$(1/1)^*$	120	0.1406	(2/2)

$L = 360$

$t_0 \times \nu_0$	$\frac{t_0\nu_0}{L}$	(a/b)	red	error	$(\frac{L}{\nu_0}/\frac{L}{t_0})$
15×15	0.6	(12/12)	2.5	0.0999	(24/24)
5×15	0.2	(6/18)	3.33	0.0165	(24/72)
25×25	1.7	(10/10)	3.6	0.0882	(14.4/14.4)
33×33	3.0	(8/8)	5.625	0.0358	(10.9/10.9)
25×75	5.2	(4/12)	7.5	0.0440	(4.8/14.4)
65×65	11.7	(5/5)	14.4	0.0227	(5.5/5.5)
84×84	19.6	(4/4)	22.5	0.0184	(4.3/4.3)
100×100	27.8	$(3/3)^*$	40	0.0838	(3.6/3.6)
101×101	28.3	$(3/3)^*$	40	0.1206	(3.6/3.6)

* no Riesz basis but algorithm still works

Table 4.6: Results of Experiment 4.6

As Figure 4.4 shows redundancy for which 90% recovery is achieved is also some kind of turning point. So while especially for highly overspread operators the convergence starts out linear at best it becomes exponential after the critical redundancy.

4.2.2 STFT-Multipliers

Definition 4.1 (STFT-Multiplier). *For a window $g \in \boldsymbol{S}_0$ and a symbol $\boldsymbol{m}$ the STFT multiplier $\mathrm{SM}_{\boldsymbol{m}}$ is given by:*

$$\mathrm{SM}_{\boldsymbol{m}}(f) = \mathcal{V}_g^*(\boldsymbol{m} \cdot \mathcal{V}_g(f)) \tag{4.7}$$

or in the weak sense on $\boldsymbol{L}^2(\mathbb{R}^d)$:

$$\mathrm{SM}_{\boldsymbol{m}}(f) = \int_{\mathbb{R}^{2d}} m(\lambda)\langle f, \pi(\lambda)g\rangle \pi(\lambda) g d\lambda \tag{4.8}$$

If $\boldsymbol{m} \in \boldsymbol{L}^2(\mathbb{R}^{2d})$ then the resulting operator $\mathrm{SM}_{\boldsymbol{m}}$ is a Hilbert-Schmidt operator.

Random Symbol m1

(a/b)	Redundancy	Original Atom	Tight Atom	Calculation Time
(9/12)	1.3333	0.6370	0.6252	0.9010s
(8/8)	2.25	0.4313	0.4287	1.0320s
(6/6)	4	0.2013	0.2012	1.3920s
(4/4)	9	0.0214	0.0214	2.3340s
(3/3)	16	0.0014	0.0014	3.4750s
Calculation Time for STFT-Multiplier: 6.8300s				

Smooth Symbol m2

(a/b)	Red.	Orig. Atom	Tight Atom	Time	Sampled Symbol	Time
(9/12)	1.3333	0.5259	0.5095	0.9220s	0.9682	0.3800s
(8/8)	2.25	0.2743	0.2683	1.0720s	0.5124	0.4500s
(6/6)	4	0.0909	0.0908	1.4320s	0.1704	0.6010s
(4/4)	9	0.0030	0.0030	2.3440s	0.0074	0.9610s
(3/3)	16	2.9196e-005	2.9196e-005	3.4350s	9.1819e-005	1.3520s
Calculation Time for STFT-Multiplier: 6.8600s						

Very Smooth Symbol m3

(a/b)	Red.	Orig. Atom	Tight Atom	Time	Sampled Symbol	Time
(9/12)	1.3333	0.3222	0.1309	0.9120s	0.3413	0.4000s
(9/9)	1.7778	0.1424	0.0603	0.9210s	0.1440	0.3910s
(8/8)	2.25	0.0712	0.0334	1.0620s	0.0714	0.4410s
(6/6)	4	0.0053	0.0033	1.4220s	0.0053	0.6110s
(4/4)	9	3.2152e-006	2.7173e-006	2.4130s	3.2152e-006	0.9620s
Calculation Time for STFT-Multiplier: 6.8500s						

Table 4.7: Results of Experiment 4.7

Comparing (4.8) to (2.1) we see that STFT-multipliers are the continuous equivalent of Gabor multipliers. In fact under certain smoothness and decay conditions on the symbol STFT-multipliers can be seen as the limiting case of Gabor multipliers, i.e., if (a_k, b_k) are a sequence of lattice constants going to $(0, 0)$ then the Gabor multipliers $\mathrm{G}^k_{\boldsymbol{m}}$ with window g and upper symbols $((a_k b_k)^{-1}\boldsymbol{m}(a_k l, b_k n))_{n,l}$ on the lattices $\Lambda_k = a_k\mathbb{Z}^d \times b_k\mathbb{Z}^d$ converge to the STFT-multiplier $\mathrm{SM}_{\boldsymbol{m}}$. See [9] for details and a proof.
It is therefore not farfetched to try to approximate STFT-multipliers also with arbitrary symbols by Gabor multipliers. In the discrete case STFT-multipliers are actually just Gabor multipliers with $a = b = 1$. As they are thus based on systems with maximal redundancy they are very costly to calculate and so it is of course desirable to approximate them by real, i.e., much less redundant, Gabor multipliers.

Experiment 4.7 *(Approximation of STFT-Multipliers).* In the experiment three STFT-multipliers resulting from symbols of varying smoothness and a normalized Gaussian as window were approximated. In one series of systems the Gabor-atom was the same as for the STFT-multiplier. In the other the tight atom for the given lattice was used because for $a = b = 1$ the starting atom also has the tightness property. In the

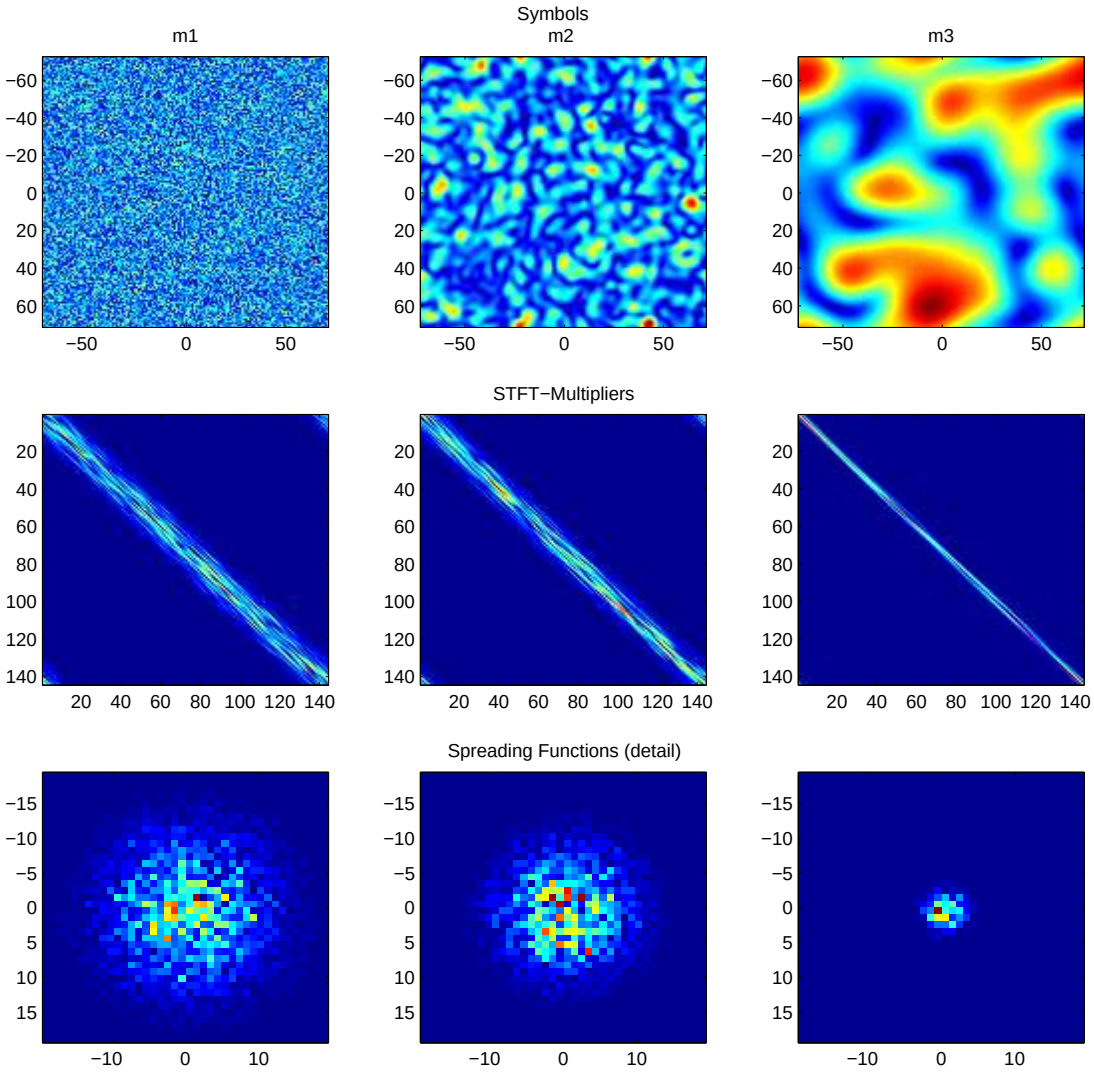


Figure 4.5: STFT-Multipliers

case of a smooth symbol (m2,m3) also the Gabor multiplier resulting from sampling the symbol on the lattice was calculated. The measure for the error was again the relative error in the Frobenius norm.

The results in Table 4.7 show that even STFT-multipliers with random symbols can be approximated using a system with sufficiently large redundancy. Taking a look at Figure 4.5 this is not surprising as they seem to be a subclass of the underspread or "slightly overspread" operators. Note that again the required redundancy for more than 90 % recovery can be read off the spreading function. The calculation time for that approximation is less than half of that for the STFT-multiplier.
For the moderately smooth symbol (m2) the approximation naturally works much better. Though there is not much difference between the original and the tight atom there is a distinctive improvement by using the best approximation instead of just the sampled multiplier. In the case of the very smooth symbol (m3) the approximation in the system with the original window does not give a smaller error than the Gabor multiplier with sampled symbol anymore. The tight system however still yields a refinement especially for low redundancies.

Thinking back to the motivation of Gabor multipliers, i.e., efficient filtering of a signal, let's see how well Gabor multipliers approximate STFT-multipliers with 0/1-masks.

Experiment 4.8 *(Digital Filters).* In analogy to the last experiment we approximate STFT-multipliers with 0/1-masks, i.e., digital filters. As the results in Table 4.8 show the best approximations especially using a tight system gibe a large improvement over just calculating the Gabor multiplier with the sampled mask. More than 90% recovery is already gained with redundancy 1.7778 for the simple square mask and 2.25 for the more complicated band mask. Thus Gabor multipliers can be very effectively used as time-variant filters.

Square Mask

(a/b)	Sampled Mask	Orig. Atom	Tight Atom
(9/12)	0.3577	0.3009	0.1090
(9/9)	0.2151	0.1461	0.0803
(8/8)	0.1905	0.0938	0.0690
(6/6)	0.1996	0.0294	0.0290
(4/4)	0.0407	0.0009	0.0009

Band Mask

(a/b)	Sampled Mask	Orig. Atom	Tight Atom
(9/12)	0.5330	0.3828	0.2481
(9/9)	0.2797	0.1700	0.1125
(8/8)	0.3119	0.1181	0.0991
(6/6)	0.1940	0.0290	0.0286
(4/4)	0.1227	0.0023	0.0023

Table 4.8: Results of Experiment 4.8

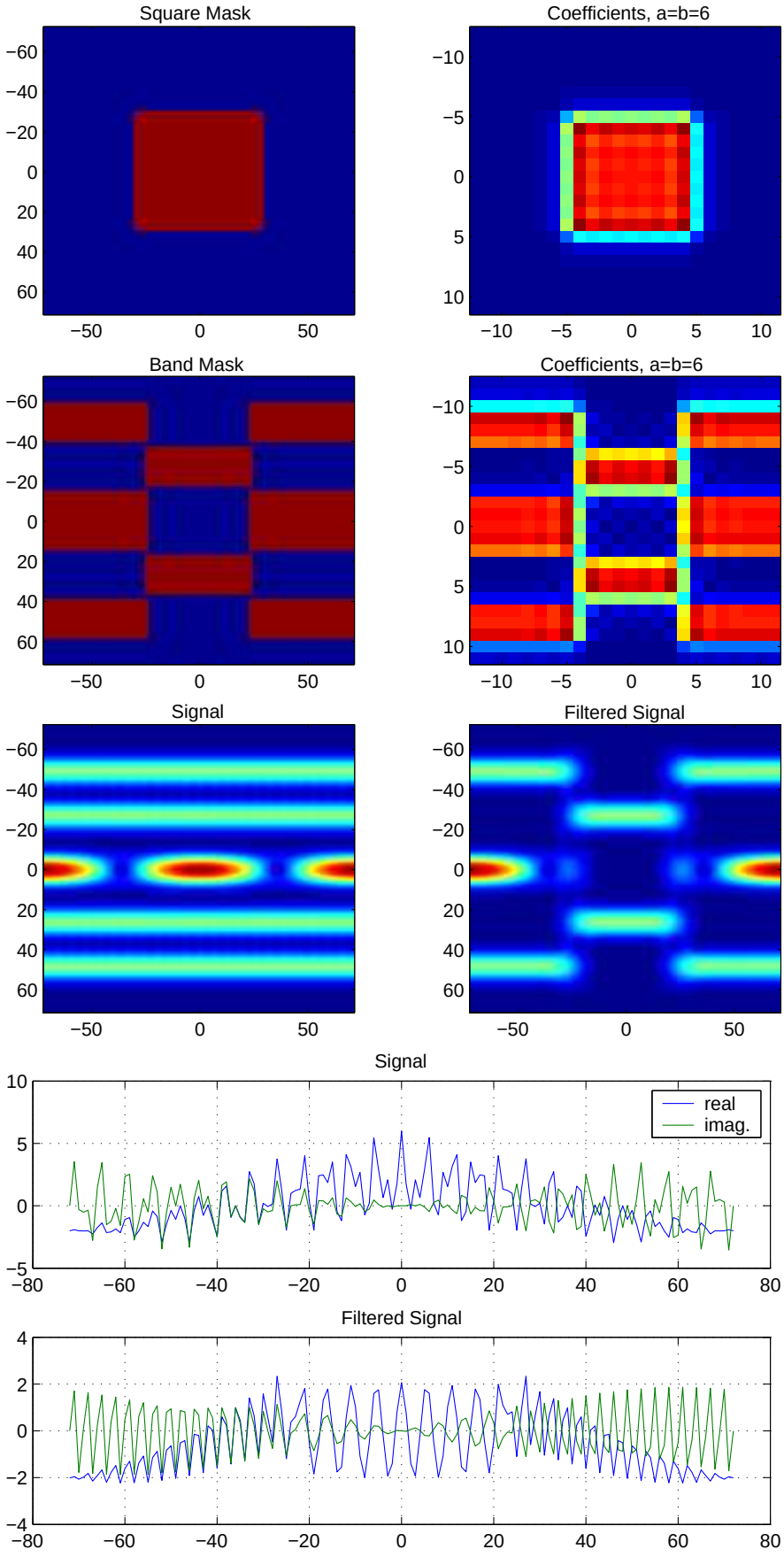


Figure 4.6: Digital Filter

Figure 4.6 shows the action of the best approximation of the STFT-multiplier with the band mask in the system made up by a Gaussian and $a = b = 6$. The signal in Figure 4.6 is a superposition of three frequencies $e^{2\pi ikx}, k = \pm 1, \pm 26, \pm 48$. With the help of the Gabor multiplier first the low and high then the middle and then again the low and high frequencies are successfully filtered out.

Two other classes of operators that one could try to approximate are the inverse and the iteration of a given Gabor multiplier. However both types of operators can again be viewed as a special subclass of the sightly overspread operators, see [7] respectively [3] for examples.
More information about Gabor multipliers in general but especially about their eigenvalue distribution and eigenvectors can be found in [9] und [3].

Appendix A

MATLAB Files

All m-files are based on NuHAG files by H. G. Feichtinger or M. Hampejs. Signals are assumed to be row vectors and matrices act on them by multiplication from the right. Additional respectively alternative code to adapt for column vectors and left multiplication is marked by '%'.

A.1 stft.m

```
% stftka.m  - calculates the STFT of a signal over a lattice
%             with lattice constants a (time) and b (frequency).
%
% Input    : f = signal    (rowvector of length L)
%            g = window    (rowvector of length L)
%            a,b = time,frequency lattice constants
%
% Output   : MxN-matrix S(m+1,n+1) =  STFTg(f)(na,mb)
%                                  = <f,Mmb Tna g>
%
% Usage    : stft(f,g,a,b) or
%            stft(f,g) (a = 1 = b  default values)


function   stft = stft(f,g,a,b);

if nargin < 4;
    a = 1;
    b = 1;
end;

L = length(f);
N=L/a;
M=L/b;

stft1 = zeros(M,N);             % stft1 = zeros(N,M)

gg =  [conj(g),conj(g)];        % gg=[conj(g);conj(g)]

for n = 0 : N-1;
    gna = gg( (L+1-n*a) : (2*L - n*a) );   % na-shifted version of g
    y= f.*gna;
    if b > 1;
        y = fold(y,b);
    end;
    stft1(:,n+1) = fft(y).';          % stft1(n+1,:)=fft(y).'
end;

stft = stft1;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function   yf = fold(y,b);

   M=length(y)/b;
   u = zeros(M,b);
   u(:) = y;
   yf = sum(u.');             % yf = sum(u.').'

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

A.2 gabsyn.m

```
% gabsynka.m    - synthesizes a vector from the coefficient matrix C
%                         and time frequency shifts of the atom g
%
% Input     : C = MxN coefficient matrix
%                         C(m+1,n+1) corresponds to
%                         M(mb)T(na)g  n=0...N-1, m=0...M-1
%             g = atom
%
% Output    : row-vector f
%
% Usage     : gabsyn(C,g)


function   f = gabsyn(C,g);

L=length(g);
[M,N]=size(C);          % [N,M]=size(C)
a=L/N; b=L/M;

f=zeros(1,L);                               % f=zeros(L,1)
Ch=L*ifft(C);                               % Ch=L*ifft(C.')
Chstack=repmat(Ch,b,1);
gg=[g,g];                                   % gg=[g;g]

for k = 1 : L;
    gk = gg((L+k) : -a : k+a);
    f(k)= Chstack(k,:)*gk.';            % f(k)=Chstack(k,:)*gk
end;
```

A.3 gabmult.m

```
% gabmultka.m   - calculates the matrix corresponding to the Gabor multiplier
%                 with multiplier C and windows g1,g2
%
% Input     :  C = multiplier on lattice, MxN matrix
%              g1 = analysis window, row vector of length L
%              g2 = synthesis window, row vector of length L
%
% Output    : LxL matrix Gm (acts on row vectors)
%
% Usage     : gabmult(m,g1,g2)
%                 or gabmult(m,g) (g1=g2=g)


function   Gm = gabmult(C,g1,g2)

if nargin == 2;
    g2=g1;
end;

L = length(g1);
[M,N]=size(C);              % [N,M]=size(C)
a=L/N; b=L/M;

gg1=[g1,g1];                % gg1=[g1;g1]
gg2=[g2,g2];                % gg2=[g2;g2]
gabm=zeros(L,L); for n=0:N-1;
    g1n=gg1( (L+1-n*a) : (2*L-n*a) );
    g2n=gg2( (L+1-n*a) : (2*L-n*a) );
    gn=g1n'*g2n;                                % gn = g2n*g1n'
    cn=fft(C(:,n+1));                           % cn = fft(C(n+1,:))
    cn=[cn;cn];                                 % cn = [cn,cn]
    Cn=zeros(M,M);
    for m=0:M-1
        Cn(:,m+1)=cn(M+1-m : 2*M-m);    % Cn(m+1,:)= ...
    end
    Cn = repmat(Cn,b,b);
    gabm=gabm+gn.*Cn;
end Gm=gabm;
```

A.4 rb.m

```
% rb.m  - checks if (g,a,b) generates a Riesz basis
%
% Input     :   g atom
%               a,b time/frequency gap
%
% Usage     : rb(g,a,b)
%

function rb(g,a,b)

mQ=abs(stftka(g,g,a,b)).^2;
ftmQ=fft2(mQ);

if min(min(ftmQ)) < 100*eps
    disp('is no Riesz basis');
else
    disp('is a Riesz basis');
end;
```

A.5 gabcoeff.m

```
% gappcoeff.m   - calculates the upper symbol of the best approximation of
%                       a general quadratic matrix K by a Gabor multiplier
%
% Input       : K quadratic matrix
%               g atom
%               a,b time/frequency gap
%
% Output    : MxN coefficient matrix
%
% Usage     : gappcoeff(K,g,a,b) or gappcoeff(K,g)


function   coeff = gappcoeff(K,g,a,b);

if nargin < 3
    a=1;
    b=1;
end

L=length(g);
N=L/a;
M=L/b;

C1=zeros(M,N);          %C1=zeros(N,M)
gg=[g,g];               %gg=[g;g]
for n=0:N-1
    gn=gg(L+1-n*a : 2*L -n*a);
    gn=gn'*gn;                  %gn=gn*gn.'
    Bn=K.*gn;
    if b>1
        Bn=fold(Bn,M);
    end
    Bn=[Bn,Bn];                 %Bn=[Bn;Bn]
    Bnf=zeros(1,M);             %Bnf=zeros(M,1)
    for l=1:M
        Bnf=Bnf+Bn(l,l:l+M-1);      %Bnf=Bnf+Bn(l:l+M-1,l)
    end
    C1(:,n+1)=fft(Bnf).';           %C1(n+1,:)=fft(Bnf).'
end

mQ=abs(stftka(g,g,a,b)).^2;
if min(min(fft2(mQ))) < 10*eps
    disp('(g,a,b) no Riesz basis');
    break;
else
    coeff=ifft2(fft2(C1)./(fft2(mQ)));
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function  cfold=fold(C,d);

L=length(C);
Cf=zeros(d,L);
for j =1:d:L
    Cf=Cf+C(j:j+d-1,:);
end
Cff=zeros(d,d);
for k=1:d:L
    Cff=Cff+Cf(:,k:k+d-1);
end
cfold=Cff;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

A.6 spread.m

```
% spread.m  - calculates the spreading function of an operator/matrix
%              that acts on row vectors by multiplication from the right
%
% Input     : K quadratic matrix
%
% Output    : quadratic matrix
%
% Usage     : spread(K)


function spread=spread(K)
                              %K=K.'
L=size(K);
K=[K;K];
sp=zeros(L);
for k=1:L
    sp(k,:)=K(L+k:-1:k+1,k).';
end
spread=sp;
spread=fft(spread);
                              %spread=spread.'
```

A.7 spr2mat.m

```
% spr2mat.m  - calculates the matrix from its spreading function
%              (matrix acts on row vectors by multiplication from the right)
%
% Input     : N quadratic matrix
%
% Output    : quadratic matrix
%
% Usage     : spr2mat(N)


function mat=spr2mat(N)
                    %N=N.'
L=size(N);
N=ifft(N);
K=[N.';N.'];
m=zeros(L);
for j=1:L
    m(j,:)=K(L+j:-1:j+1,j).';
end
mat=m.';            %mat=m
```

Bibliography

[1] O. Christensen. *An Introduction to Frames and Riesz Bases*. Birkhäuser, Boston, 2003.

[2] J. B. Conway. *A Course in Functional Analysis*. Springer New York, 1990.

[3] M. Dörfler. *Gabor Analysis for a Class of Signals called Music*. Dissertation, University of Vienna, 2002.

[4] H. G. Feichtinger. Wiener amalgams over Euclidean spaces and some of their applications In K. Jarosz, editor, *Proc.Conf.Function spaces volume 136 of Lect.Notes in Math pages 123-137*. Edwardsville, IL, 1991.

[5] H. G. Feichtinger. Gabor multipliers and spline-type spaces over lca groups. In D. X. Zhou, editor, *Wavelet Analysis: Twenty Years' Developments*. World Scientific Press, Singapore, 2002.

[6] H. G. Feichtinger. Gabor multipliers with varying lattices. *Proc. SPIE Conf.*, August 2003.

[7] H. G. Feichtinger, M. Hampejs and G. Kracher. Approximation of matrices by Gabor multipliers. *IEEE Sign. Proc. Letters*, 2004.

[8] H. G. Feichtinger and W. Kozek. Operator quantization on lca groups. In H.G. Feichtinger and T. Strohmer, editors, *Gabor Analysis and Algorithms: Theory and Applications*. Birkhäuser, Boston, 1998. Chap. 7.

[9] H. G. Feichtinger and K. Nowak. A first survey of Gabor multipliers. In H.G. Feichtinger and T. Strohmer, editors, *Advances in Gabor Analysis*. Birkhäuser, Boston, 2002.

[10] H. G. Feichtinger und T. Werther. Robustness of minimal norm interpolation in Sobolev algebras. In J. J. Benedetto and A. Zayed, editors, *Sampling, Wavelets and Tomography*. Birkhäuser, Boston, 2002.

[11] H. G. Feichtinger and G. Zimmermann. A Banach space of test functions for Gabor analysis. In H.G. Feichtinger and T. Strohmer, editors, *Gabor Analysis and Algorithms: Theory and Applications*. Birkhäuser, Boston, 1998. Chap. 3.

[12] I. M. Gelfand and N. J. Wilenkin. *Generalized Functions vol.4: Some Applications of Harmonic Analysis. Rigged Hilbert Spaces*. Academic Press, New York, 1964.

[13] G. H. Golub and C. F. Van Loan. *Matrix Computations*, 3rd edition. The Johns Hopkins University Press, Baltimore, 1996.

[14] K. Gröchenig. *Foundations of Time-Frequency Analysis*. Applied and Numerical Harmonic Analysis. Birkhäuser, Boston, 2001.

[15] K. Gröchenig and G. Leinert. Wiener's Lemma for Twisted Convolution and Gabor Frames. *J. Amer. Math. Soc. 17, 1-18*, 2004.

[16] A. J. E. M. Janssen and T. Strohmer. Characterization and computation of canonical tight windows for Gabor frames. *J. Four. Anal. Appl.*,8(1):1-28, 2002.

[17] W. Kozek. Adaption of Weyl-Heisenberg frames. In H.G. Feichtinger and T. Strohmer, editors, *Gabor Analysis and Algorithms: Theory and Applications*. Birkhäuser, Boston, 1998. Chap. 10.

[18] P. Prinz. *Theory and algorithms for discrete 1-dimensional Gabor frames*. Masters Thesis, University of Vienna, 1996.

[19] H. Reiter. *Classical Analysis and Locally Compact Abelian Groups*. Oxford University Press, 1968.

[20] T. Strohmer. Numerical algorithms for discrete Gabor expansions. In H.G. Feichtinger and T. Strohmer, editors, *Gabor Analysis and Algorithms: Theory and Applications*. Birkhäuser, Boston, 1998. Chap. 8.

Curriculum Vitae

Name: Karin Schnass
Geburtstag: 3. Mai 1980
Geburtsort: Klosterneuburg
Eltern: Dr Karl und Susanne Schnass
Staatsbürgerschaft: Österreich
Email: karin.schnass@aon.at

1986-1990: Volksschule in Klosterneuburg
1990-1998: BG und BRG Klosterneuburg
Juni 1998: Matura (mit Auszeichnung)
1998-2000: Biologiestudium an der Universität Wien
1998-2004: Mathematikstudium an der Universität Wien
WS 2001: Auslandssemester an der University of Leeds (UK)