MulAc :
Continuous and Discrete Gabor Multipliers

Hans G. FEICHTINGER

September 23, 2008

Preambel: still prelim. version

Gabor analysis appears at first sight as a rather complicated using infinite dimensional Hilbert spaces, non-orthogonal expansions and double series representations.
HOWEVER, one can separate the different aspects

- 1) LINEAR ALGEBRA (linear signal processing) aspect;
- 2) Algebraic structure (Weyl-Heisenberg group, phase space, time-frequency shifts);

Preambel: still prelim. version

Gabor analysis appears at first sight as a rather complicated using infinite dimensional Hilbert spaces, non-orthogonal expansions and double series representations.
HOWEVER, one can separate the different aspects

- 1) LINEAR ALGEBRA (linear signal processing) aspect;
- 2) Algebraic structure (Weyl-Heisenberg group, phase space, time-frequency shifts);
- 3) Functional analytic aspect (Hilbert spaces, frames, Riesz bases, modulation spaces);

Preambel: still prelim. version

Gabor analysis appears at first sight as a rather complicated using infinite dimensional Hilbert spaces, non-orthogonal expansions and double series representations.
HOWEVER, one can separate the different aspects

- 1) LINEAR ALGEBRA (linear signal processing) aspect;
- 2) Algebraic structure (Weyl-Heisenberg group, phase space, time-frequency shifts);
- 3) Functional analytic aspect (Hilbert spaces, frames, Riesz bases, modulation spaces);
- 4) Connection between finite and continuous models;

Preambel: still prelim. version

Gabor analysis appears at first sight as a rather complicated using infinite dimensional Hilbert spaces, non-orthogonal expansions and double series representations.
HOWEVER, one can separate the different aspects

- 1) LINEAR ALGEBRA (linear signal processing) aspect;
- 2) Algebraic structure (Weyl-Heisenberg group, phase space, time-frequency shifts);
- 3) Functional analytic aspect (Hilbert spaces, frames, Riesz bases, modulation spaces);
- 4) Connection between finite and continuous models;

Linear Algebra Background

First we are going to recall some basic notations from LINEAR ALGEBRA. Note that every matrix $\mathbf{A}$ of size $m \times n$ defines a linear mapping from $\mathbb{C}^n$ to $\mathbb{C}^m$ via matrix multiplication (and vice versa). Any such matrix has a so-called (Moore-Penrose) pseudo-inverse matrix $\mathbf{B} = pinv(\mathbf{A})$, of format $n \times m$.

If (and only if) the matrix has maximal rank, i.e. if $rank(\mathbf{A}) = min(n, m)$, then such a matrix has either a left inverse (in case of linear independent columns) or a right inverse (in the case of linear independent rows).

Linear Algebra

It is important to note that the pseudo-inverse is in fact a very nice geometric object, as it provides the MNLSQ-solution (= minimal norm least squares solution) to the problem $\mathbf{A} * \mathbf{x} = \mathbf{b}$, i.e. $\mathbf{x}_0 := pinv(\mathbf{A}) * \mathbf{b}$ satisfies the two inequalities

$$\|\mathbf{A} * \mathbf{x}_0 - \mathbf{b}\| \leq \|\mathbf{A} * \mathbf{x} - \mathbf{b}\|$$

and

$$\|\mathbf{x}_0\| \leq \|\mathbf{x}\| \quad \forall \quad \mathbf{x}$$

which minimize the estimate above. Equivalently, $\mathbf{x}_0$ is the unique minimizer in the row space of $\mathbf{A}$, i.e. the range of $pinv(\mathbf{A})$ is the row space of $\mathbf{A}$.

This *geometric* description of the pseudo-inverse is important, because it implies that the pseudo-inverse is a property of the linear mapping (and the scalar products on both ends of the linear mapping), not just a property of the concrete matrix representation. It is also true that

$$pinv(\mathbf{A}) * \mathbf{A} = \mathbf{P}_R$$

and

$$\mathbf{A} * pinv(\mathbf{A}) = \mathbf{P}_C,$$

where $\mathbf{P}_C$ is the orthogonal projection onto the column space of $\mathbf{A}$ (within $\mathbb{C}^m$) and $\mathbf{P}_R$ is the orthogonal projection onto the row space of $\mathbf{A}$ (within $\mathbb{C}^n$).

Linear Algebra

Using the fact that for two matrices of equal height m, say $\mathbf{A}_1$ and $\mathbf{A}_2$, the collection of all scalar products is obtained via matrix multiplication, namely as $\mathbf{A}_1' * \mathbf{A}_2$ (where the dash indicates according to MATLAB convention transposition and complex conjugation), and also the easy rule that $pinv(\mathbf{A}') = pinv(\mathbf{A})'$, we realize that in the case of linear independent columns we have that

orthogonal projection via biorthogonal system

$$\mathbf{P}_C(\mathbf{x}) = \mathbf{A} * (\mathbf{B}' * x) = \sum \langle \mathbf{x}, \mathbf{b}_k \rangle \mathbf{a}_k,$$

where $\mathbf{B} = pinv(\mathbf{A}')$ is the so-called biorthogonal system. It can be obtained by

$$pinv(\mathbf{A}') = \mathbf{A} * inv(\mathbf{A}' * \mathbf{A}).$$

Linear Algebra

Using the fact that for two matrices of equal height m, say $\mathbf{A}_1$ and $\mathbf{A}_2$, the collection of all scalar products is obtained via matrix multiplication, namely as $\mathbf{A}_1' * \mathbf{A}_2$ (where the dash indicates according to MATLAB convention transposition and complex conjugation), and also the easy rule that $pinv(\mathbf{A}') = pinv(\mathbf{A})'$, we realize that in the case of linear independent columns we have that

orthogonal projection via biorthogonal system

$$\mathbf{P}_C(\mathbf{x}) = \mathbf{A} * (\mathbf{B}' * x) = \sum \langle \mathbf{x}, \mathbf{b}_k \rangle \mathbf{a}_k,$$

where $\mathbf{B} = pinv(\mathbf{A}')$ is the so-called biorthogonal system. It can be obtained by

$$pinv(\mathbf{A}') = \mathbf{A} * inv(\mathbf{A}' * \mathbf{A}).$$

Frame Theory

For the case $n \geq m$ we have a set of generators for $\mathbb{C}^m$, or in modern terminology a frame. The role of a biorthogonal system is now that of a *dual frame*, i.e. the system $\mathbf{B} = pinv(\mathbf{A}')$ is a system of (column) vectors in $\mathbb{C}^m$, such that every $\mathbf{x} \in \mathbb{C}^m$ can be represented in as

minimal norm coefficients or standard frame representation

$$\mathbf{x} = \mathbf{P}_C(x) = \mathbf{A} * (\mathbf{B} \; * x) = \sum \langle \mathbf{x}, \mathbf{b}_k \rangle \mathbf{a}_k,$$

Frame Theory

For the case $n \geq m$ we have a set of generators for $\mathbb{C}^m$, or in modern terminology a frame. The role of a biorthogonal system is now that of a *dual frame*, i.e. the system $\mathbf{B} = pinv(\mathbf{A}')$ is a system of (column) vectors in $\mathbb{C}^m$, such that every $\mathbf{x} \in \mathbb{C}^m$ can be represented in as

minimal norm coefficients or standard frame representation

$$\mathbf{x} = \mathbf{P}_C(x) = \mathbf{A} * (\mathbf{B} \ * x) = \sum \langle \mathbf{x}, \mathbf{b}_k \rangle \mathbf{a}_k,$$

where the set of coefficients, given by $\mathbf{c}_k := \langle \mathbf{x}, \mathbf{b}_k \rangle \mathbf{a}_k$ is the set of *minimal norm coefficients* among all the coefficient sequences representing $\mathbf{x}$.

Frame Theory

For the case $n \geq m$ we have a set of generators for $\mathbb{C}^m$, or in modern terminology a frame. The role of a biorthogonal system is now that of a *dual frame*, i.e. the system $\mathbf{B} = pinv(\mathbf{A}')$ is a system of (column) vectors in $\mathbb{C}^m$, such that every $\mathbf{x} \in \mathbb{C}^m$ can be represented in as

minimal norm coefficients or standard frame representation

$$\mathbf{x} = \mathbf{P}_C(x) = \mathbf{A} * (\mathbf{B} \ * x) = \sum \langle \mathbf{x}, \mathbf{b}_k \rangle \mathbf{a}_k,$$

where the set of coefficients, given by $\mathbf{c}_k := \langle \mathbf{x}, \mathbf{b}_k \rangle \mathbf{a}_k$ is the set of *minimal norm coefficients* among all the coefficient sequences representing $\mathbf{x}$.

Diagonal Matrices

Among all matrices diagonal matrices are certainly a special class of matrices. They are easily composed (simple pointwise multiplication of the diagonal entries) and inverted.

Diagonal Matrices

Among all matrices diagonal matrices are certainly a special class of matrices. They are easily composed (simple pointwise multiplication of the diagonal entries) and inverted.
What is true for matrix representations with respect to a fixed basis becomes an interesting object when we come to frame expansions of operators: hence **frame multipliers** are an object of interest.

Diagonal Matrices

Among all matrices diagonal matrices are certainly a special class of matrices. They are easily composed (simple pointwise multiplication of the diagonal entries) and inverted.
What is true for matrix representations with respect to a fixed basis becomes an interesting object when we come to frame expansions of operators: hence **frame multipliers** are an object of interest. By definition an operator which is obtained by taking frame coefficients of the input vector, then multiplying these coefficients by scalars, and then doing synthesis, is called a frame multiplier.

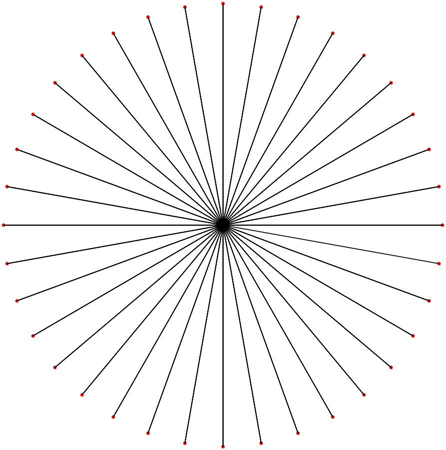
Diagonal Matrices

Among all matrices diagonal matrices are certainly a special class of matrices. They are easily composed (simple pointwise multiplication of the diagonal entries) and inverted.
What is true for matrix representations with respect to a fixed basis becomes an interesting object when we come to frame expansions of operators: hence **frame multipliers** are an object of interest. By definition an operator which is obtained by taking frame coefficients of the input vector, then multiplying these coefficients by scalars, and then doing synthesis, is called a frame multiplier.

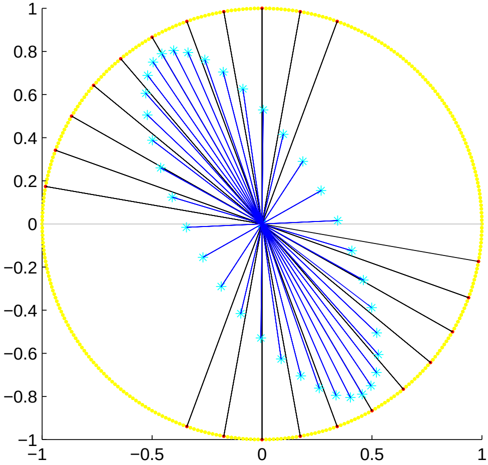
Note that the non-uniqueness of representation implies that such an operator, which **factorizes** through a diagonal matrix does not have in general a diagonal matrix in its frame representation. Hence the composition and inversion of frame multipliers is all but obvious, although frame multipliers with respect to localized frames (such as Gabor frames or wavelet frames with good atoms) are showing a number of good properties.
So what is the difference between *Fourier multipliers, wavelet multipliers* (with respect to the usual, good wavelet ONBs) and **Gabor multipliers**? The answer lies in the non-uniqueness of frame coefficients.

A generic, high redundancy frame in the plane

a frame of redundancy 18 in the plane



The effect of a frame multiplier in the plane:



GABOR ANALYSIS:
the algebraic viewpoint

Next we will be concerned specifically with Gabor families, specifically with **Gabor frames** (spanning all of $\mathbb{C}^m$) in a stable way) and with **Gabor Riesz Bases** (spanning proper subspaces of $\mathbb{C}^m$, but allowing to recover coefficients from linear combinations in a stable way: this is essentially the principle of *mobile communication*, because Gaborian families are quite stable under so-called *underspread* (linear, time-variant) channels.

Gabor Families

Among Gabor families we distinguish between "regular ones" which arise by the action (TF-shifts) of a given atom (or a finite set of atoms) along a TF-lattice. Algebraically speaking, one considers the orbit of some projective representation.

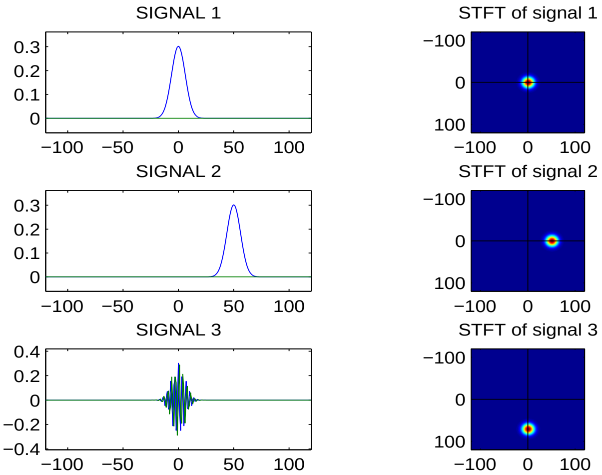
Alternatively one has "irregular families". Coorbit theory (Feichtinger/Gröchenig) implies, that such families are frames if there is enough density of the underlying pointset (given the behavior of `stft(g,g)`, the ambiguity function of g).

Gabor Families

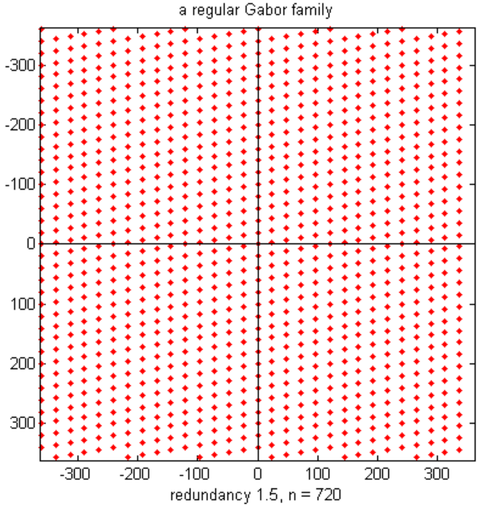
Among Gabor families we distinguish between "regular ones" which arise by the action (TF-shifts) of a given atom (or a finite set of atoms) along a TF-lattice. Algebraically speaking, one considers the orbit of some projective representation.

Alternatively one has "irregular families". Coorbit theory (Feichtinger/Gröchenig) implies, that such families are frames if there is enough density of the underlying pointset (given the behavior of `stft(g,g)`, the ambiguity function of g).

Gabor Families

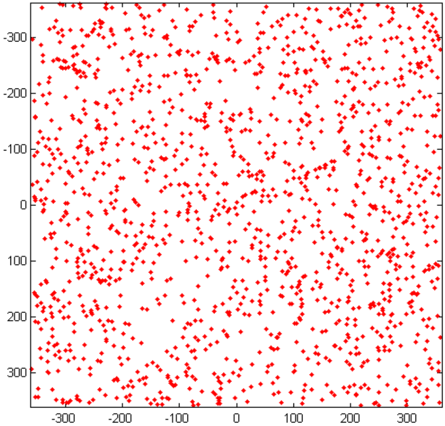


Regular lattice, redundancy 1.5, $n = 720$.

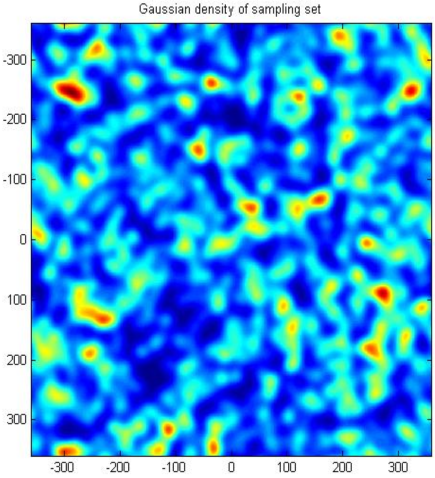


Pictures

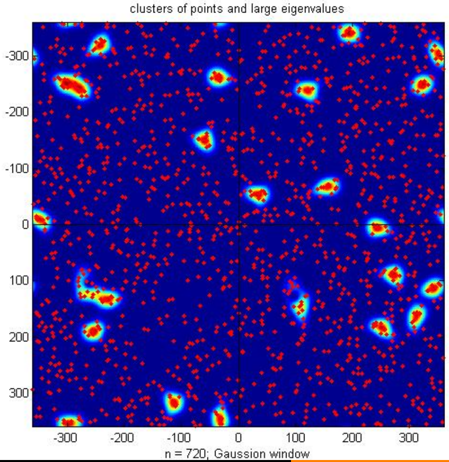
Irregular Gabor family, 1486 points, redundancy 2.0639.



Gaussian density of irregular Gabor family:

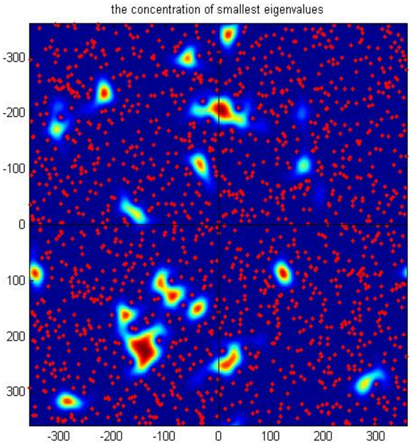


Concentration of subspace of 24 first eigenvectors: measured by calculating $\|\mathbf{P}_{24}(\pi(\lambda)g)\|^2$, using STFT of $h_1, \ldots, h_{24}$.

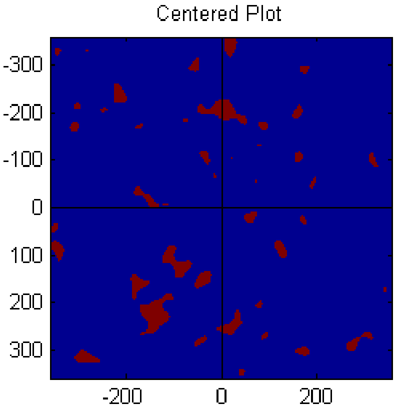


Gabor Families

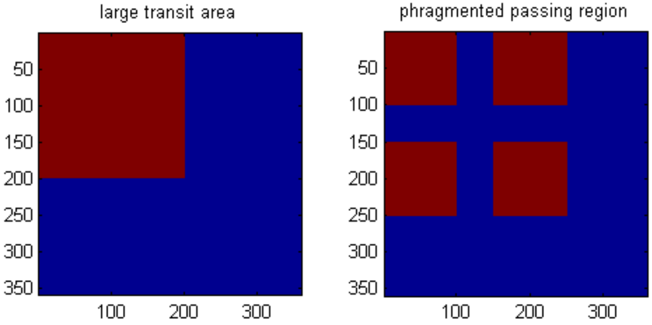
Concentration areas for the "lowest" eigenvectors of the irregular Gabor frame operator:



Showing areas where the sampling set has low (Gaussian) density:

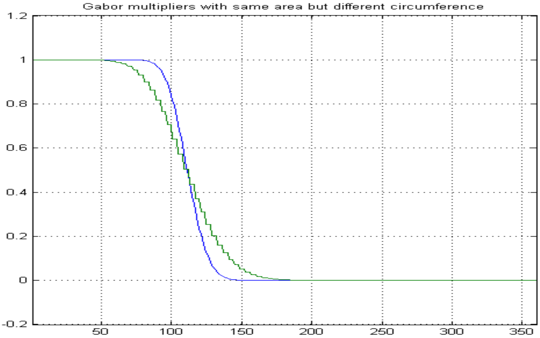


Comparison of two different *pass regions* in the TF-plane, with equal area but different length of the boundary.



Localization Operators

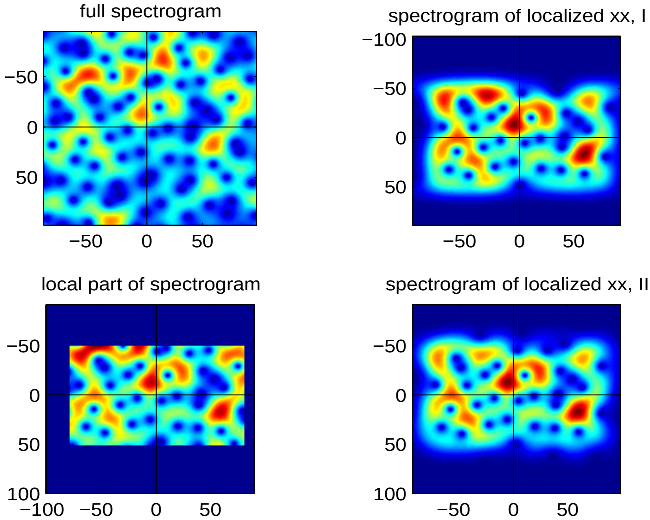
Illustrating the results of Krysztof Nowak on *plateau's* and *plunge regions* of the spectrum of such localization operators.



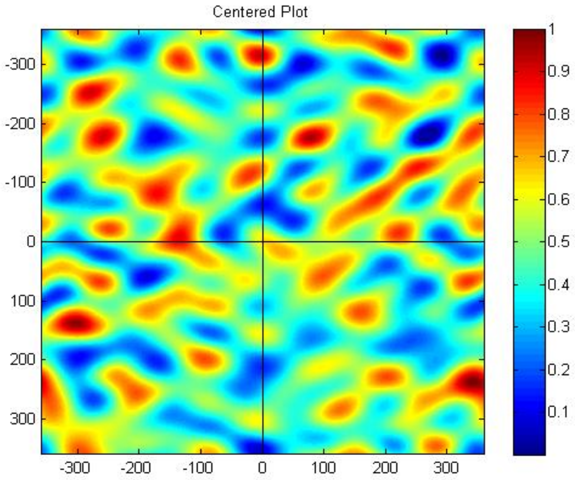
Gabor Multipliers

Before going into the study of Gabor multipliers etc. let us think of the possible applications: one would like to filter, given the spectrogram, some parts of the signal in a time-variant fashion.
It is a matter of fact, that this is not a trivial task, although it looks like a simple image processing problem, which it is not. Localizing by setting parts of the spectrogram to zero does not work, because there is (almost) never a signal whose spectrogram equals the modified spectrum (e.g. due to discontinuities). Moreover, such a 0/1-STFT-multiplier (or *localization operator* turns NOT to define a projection operator. Due to the work of K. Nowak and M. Dörfler we know a lot about such operators and ways out (in particular about the eigensystem of such operators).

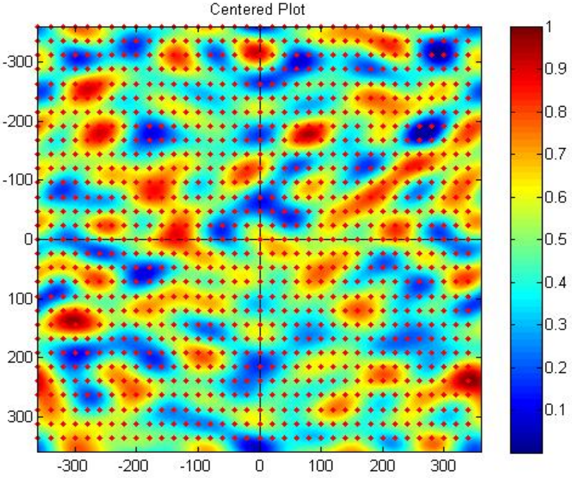
Gabor Multipliers

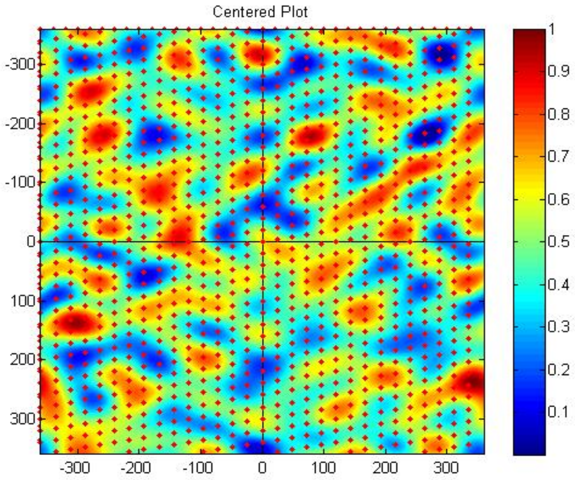


Gabor Multipliers



Gabor Multipliers




Centered Plot
-300
-200
-100
0
100
200
300
-300
-200
-100
0
100
200
300
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1

Gabor Multipliers

The Riesz-Basis business comes into the picture when we try to do the best approximation of a given operator (or matrix) by a Gabor multiplier (or a multi-window Gabor multiplier system, or a "generalized Gabor multiplier" (see the discussions in the MOHAWI project, or Monika's talk).
The link to spline-type theory (and algorithms) is through the Kohn-Nirenberg symbol. Recall, that the TF-shift or better conjugation of an operator (e.g. a projection operator on the Gabor atom) is expressed as the ordinary phase-space shift of its Kohn-Nirenberg symbol. Via the (symplectic) Fourier transform the spreading function comes in.

Gabor Expansions

Once we have the continuous short-time Fourier transform

$$V_g(f)(\lambda) = V_g(f)((t,\omega)) = \langle f, \pi(\lambda)g\rangle = \langle f, M_\omega T_t f\rangle.$$

we can reconstruct f from $V_g(f)$ by the formula:

$$f = \int_{\mathbb{R}^{2d}} V_g(f)(\lambda)\pi(\lambda)g \quad d\lambda.$$

This integral is to be understood "in the weak sense" (i.e. is turned into a valid integral by taking scalar products $, h\rangle$ with some $h \in \mathbf{L}^2(\mathbb{R}^d)$ on both sides). However, if g is "decent" then the integral is convergent in the norm sense of $\mathbf{L}^2(\mathbb{R}^d)$, i.e.

$$f = lim_{(a,b)\to(0,0)} \frac{1}{ab} \sum_{(n,k)\in\mathbb{Z}^{2d}} V_g(f)(an, bk)\pi(an, bk)g,$$

the double sums being approximated by finite partial sums.

Gabor Expansions

It is one of the central results of Gabor analysis that one can compensate for the error due to discretization, i.e. for (a, b) sufficiently close to $(0, 0)$ one simply has to replace either the analysis window g or the synthesis window $\tilde{g}$ in the above procedure in order to achieve perfect representation:

$$f = \sum_{(n,k)\in\mathbb{Z}^{2d}} V_{\tilde{g}}(f)(an, bk)\pi(an, bk)g$$

respectively

$$f = \sum_{(n,k)\in\mathbb{Z}^{2d}} V_g(f)(an, bk)\pi(an, bk)\tilde{g}.$$

with unconditional convergence in the norm of $\mathbf{L}^2(\mathbb{R}^d)$.

Given the asymmetry in the above representation it is a advisable to go for a symmetric version of the exact representation formula. Since $\tilde{g} =^{-1} S(g)$ (S is the Gabor frame operator) it is natural to choose $g_t = S^{-1/2}g$ in order to have

$$f = \sum_{(n,k)\in\mathbb{Z}^{2d}} V_{g_t}(f)(an, bk)\pi(an, bk)g_t.$$

Nowadays we have various efficient methods to calculate g_t, not just by using the binomial expansion, e.g. iterative methods analyzed in detail by A.J.M.E. Janssen.

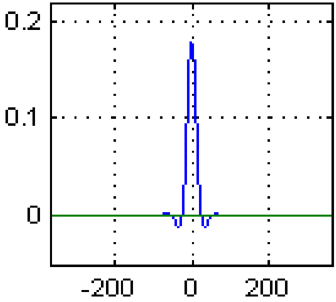
STFT and Gabor multipliers

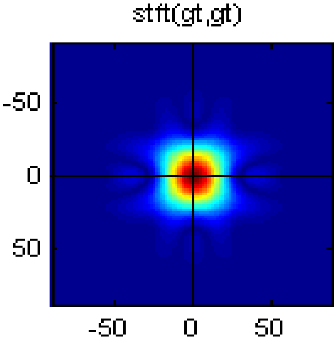
Having now the different characterizations it is natural to try to "operate" on the signal by modifying some of its parts. In fact, somehow we have taken an "image" of the (acoustic) signal, unfolding it into an image, so that image processing methods, the removal or emphasize of certain parts of the signal appear to be quite natural.
Correspondingly we talk of **STFT-multipliers** (also often called *Anti-Wick operators*) if the full STFT is multiplied, or a **Gabor multiplier**, if a specific Gabor family $(G)(g, a, b)$ is used.

Most often we prefer *symmetric* situations, i.e. operators with equal analysis and synthesis window (typically with g or g_t. The most important reason for this restriction is the fact that a real-valued multiplier gives a self-adjoint operator. For a fixed lattice $\Lambda = a\mathbb{Z}^d \times b\mathbb{Z}^d$ the use of the canonical tight widow g_t has the extra advantage that one has a minimal functional calculus, i.e. "doing nothing" with the STFT-coefficients (resp. multiplying them with the constant function) gives us the identity operator.

Gabor Multipliers

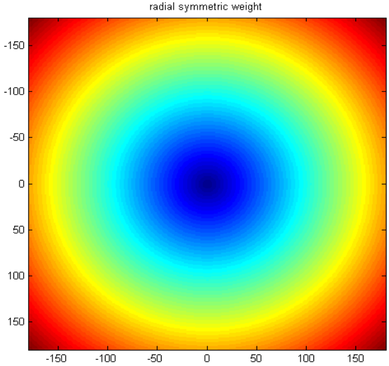
Back to the regular case: The tight Gabor atom $gt = S^{-1/2}g$, for $n = 720$; $a = 20$; $b = 24$; $red = 1.5$.





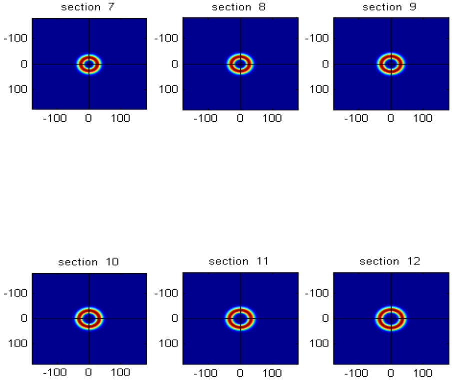
Degression concerning Discrete Hermite Signals

An interesting case is the family of STFT-multipliers with Gaussian (analysis and synthesis) window, and radial symmetric weights, such as:

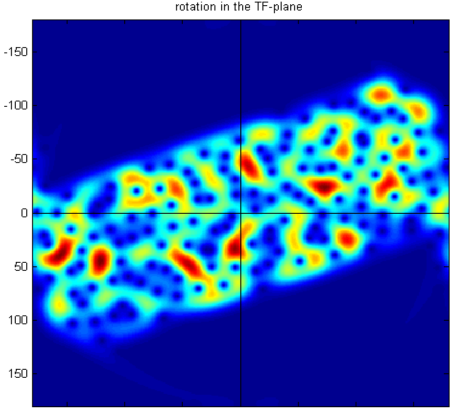


Degression concerning Discrete Hermite Signals

The TF-concentration of the corresponding eigenvectors makes them a perfect substitute for Hermite functions:



As an example we can demonstrate that a lowpass signal can be “rotated” within the TF-plane (by 20 degrees) by a suitable Hermite-multiplier:



We have a NUMBER OF QUESTIONS related to STFT-multipliers and their approximation by Gabor multipliers:

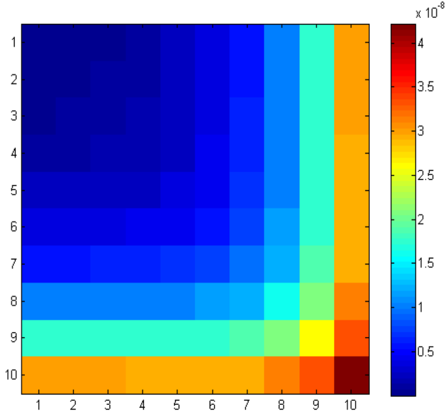
- In which sense can on approximate a general STFT multiplier by Gabor multipliers?
- How can one calculate (in the HS sense) the best approximation of a given matrix by Gabor multipliers (with given parameters (g, a, b))?
- Given the redundancy, what are good lattices (e.g. hexagonal)?
- What is the speed of convergence (say in the HS-sense) of the Gabor multipliers towards the limiting STFT multiplier (assuming that the weight is smooth and that the lattice constants tend to $(0, 0)$.

Gabor Multipliers

Another type of question:
There is a continuous dependence of Gabor multipliers (in the strong operator topology) of Gabor multipliers e.g. with respect to "varying lattices". So we assume that a Gabor multiplier T_n is obtained by sampling a nice weight function along the lattice with TF-lattice constants (a_n, b_n) (converging to (a_0, b_0)). We know that then $\|G_n(f) \to G_0(f)\|_2 \to 0$ as $n \to \infty$. But how fast is the convergence. Does it depend more on the smoothness of f (and decay), or on the question whether the window is adapted to the lattice (i.e. that one uses the tight window $S_n^{-1/2} g$ with respect to the lattice $a_n\mathbb{Z} \times b_n\mathbb{Z}$) or not (the same window g is used for all the lattices?
Of course one can take f from some Shubin class, because the unit ball of such a Shubin class is relatively compact within $\mathbf{L}^2(\mathbb{R}^d)$ and so one can expect that on has uniform convergence (of a certain speed) in this case.

This is connected with the following observation: For smooth weight functions that Gabor multiplier obtained by just going from the "fully continuous setting" to the sampled (in the TF-sense) is showing typically a much smaller error if one tries to partially compensate the discretization error by modifying the window a littlebit, i.e. by replacing the original window by the canonical tight window.
Recall that this is also characterized as the tight window which is closest to the original window in the $\mathbf{L}^2$-setting.

Given pure frequencies as weight sequences: determination of relative error in approximating STFT-multiplier by Gabor multiplier with tight atom:



Worst approximation of the continuous STFT multiplier by a discrete one (note areas in between lattice points!):