Available online at www.sciencedirect.com
ScienceDirect
Signal Processing 87 (2007) 1670–1676
SIGNAL PROCESSING
www.elsevier.com/locate/sigpro

The inversion of Gabor-type matrices

Mario Hampejs*, Günther Kracher

Faculty of Mathematics, University of Vienna, Nordbergstrasse 15, 1090 Vienna, Austria

Received 20 October 2006; received in revised form 2 January 2007; accepted 17 January 2007
Available online 31 January 2007

Abstract

We present a fast algorithm for the inversion of certain sparse matrices with periodic side-diagonals. Matrices of this type are widely used in time–frequency analysis and Gabor theory. There are no constraints to the matrix like in other known algorithms and the algorithm computes the exact inverse and not only an approximation.

Keywords: Gabor; Time–frequency; Inversion; Algorithm; Weyl–Heisenberg; Frame operator

1. Introduction

The STFT (short-time Fourier transform, also called Gabor-transform in its sampled variant) is a valuable tool to display the energy distribution of a signal f over the time–frequency plane. Recall that for any non-zero window function g and a signal f the STFT can be defined as $\mathrm{STFT}_g(f)(\lambda) = V_g(f)(\lambda) = \langle f, \pi(\lambda)g\rangle$ with $\lambda = (t, \omega)$ and the following abbreviations $T_t f(z) = f(z-t)$ (translation operator), $M_\omega f(t) = f(t)\chi_\omega(t)$ (modulation operator with character $\chi_\omega(t) = \mathrm{e}^{2\pi i\omega t}$) and the composition $\pi(t,\omega) = M_\omega T_t$. Basic Gabor theory (cf. the introduction of [1]) tells us that the original signal can be reconstructed by

$$f = \sum_{\lambda\in\Lambda} V_g f(\lambda)\cdot\pi(\lambda)\tilde{g} \tag{1}$$

using the canonical *dual* Gabor atom $\tilde{g} = S^{-1}g$, provided that the so-called *frame-operator* S, given by $Sf = \sum_\lambda V_g(f)(\lambda)\pi(\lambda)g$, is invertible. In this case the family $(\pi(\lambda)g)_{\lambda\in\Lambda}$ is called a *Gabor-* (or also *Weyl–Heisenberg*) *frame* generated by (g, Λ).

Gabor-type matrices (we will call them blockmatrices) are matrices which occur in Gabor-analysis—the frame operator for a Weyl–Heisenberg frame has this special form. In particular, if g is a window function of length N, a, b (divisors of N) the time and the frequency-gap, then the frame operator S has the following form:

$$S_{m,n} = \begin{cases} M\sum_{k=0}^{N-1} T_{ka}g(m)\overline{T_{ka}g(n)} & \text{if } M|m-n, \\ 0 & \text{otherwise} \end{cases}$$

with $M = N/b$.

This is called the Walnut-representation. The inversion of S is important for the calculation of the dual window $\tilde{g}$ because $\tilde{g} = S^{-1}g$. See Fig. 1 for $\tilde{g}$ and g with $N = 144, a = 12, b = 9$.

For the multiwindow case (see Section 5) the frame operator is of block-diagonal type as well and can be inverted with our algorithm. Applying this inverse to the distinct atoms we obtain the dual windows.

*Corresponding author.
E-mail addresses: mario.hampejs@univie.ac.at (M. Hampejs), guenther.kracher@univie.ac.at (G. Kracher).

0165-1684/$ - see front matter © 2007 Elsevier B.V. All rights reserved.
doi:10.1016/j.sigpro.2007.01.012

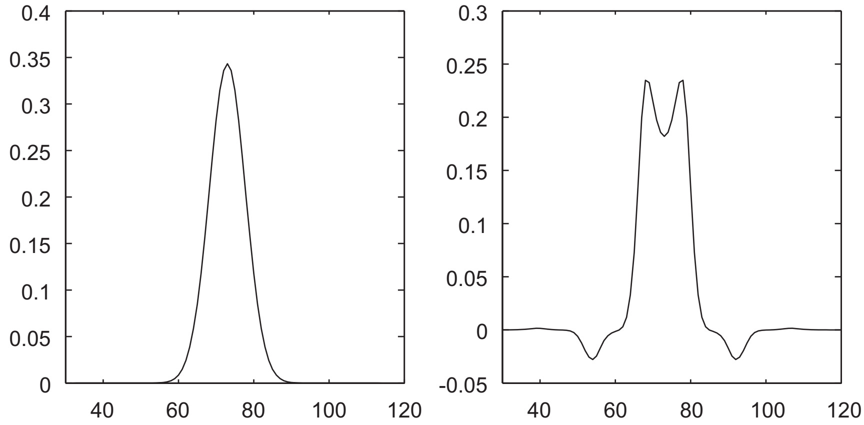

Fig. 1. Left: atom g; right: dual atom $\tilde{g}$.

2. Problem statement

Assume we are given the following $N \times N$ matrix S with $N, a, b \in \mathbb{N}$ and $a \mid N, b \mid N$:

$$S = \begin{pmatrix}
s_{11} & 0 & \cdots & 0 & s_{21} & 0 & & & 0 & s_{b1} & 0 & \cdots & 0 \\
0 & s_{12} & & & & s_{22} & & & & & s_{b2} & & \\
\vdots & & \ddots & & & & \ddots & & & & & s_{b3} & \\
0 & & & s_{1a} & & & & s_{2a} & & & & & \ddots \\
\ddots & & & & s_{11} & & & & & & & & \\
0 & s_{b1} & & & & \ddots & & & & & & & \\
 & & \ddots & & & & s_{1a} & & & & & & \\
 & & & s_{ba} & & & & \ddots & & & & s_{21} & \\
0 & & & & \ddots & & & & \ddots & & & & \ddots \\
\ddots & & & & & \ddots & & & & \ddots & & & \\
0 & s_{21} & & & & & s_{b1} & & & & s_{11} & & \\
\vdots & & \ddots & & & & & \ddots & & & & \ddots & \\
0 & & & s_{2a} & & & & & s_{ba} & & & & s_{1a}
\end{pmatrix}.$$

Without proof we claim that S commutates with every $\pi(\lambda) : \lambda \in \Lambda$ with $\lambda = (t, \omega)$ and $\{\pi(\lambda) : t, \omega = 0, \ldots, N-1\}$ is an ONB in the $\mathscr{HS}$-norm (Hilbert–Schmidt or Frobenius-norm, defined by $\|S\|_{\mathscr{HS}} = \sqrt{\sum_{i,j=0}^{N-1} |s_{i,j}|^2}$), for the matrices. The coefficients $J_{k,l}$ with respect to this basis we call the Janssen-representation $S = \sum_{k=0}^{a-1} \sum_{l=0}^{b-1} J_{k,l} \pi(l\widetilde{b}, k\widetilde{a})$, with $\widetilde{a} = N/a$, $\widetilde{b} = N/b$ and $J_{k,l} = (N/a \cdot b) V_g(g)(l\widetilde{b}, k\widetilde{a})$.

Definition 1. If $\Lambda = a\mathbb{Z} \times b\mathbb{Z}$ is a lattice then $\Lambda^o = \widetilde{b}\mathbb{Z} \times \widetilde{a}\mathbb{Z}$ is called the adjoint lattice. For a general definition of a lattice on LCA groups see [2].

Operators which are commuting with every $\pi(\lambda)$ have a specific Janssen-representation: $J_{k,l} \neq 0 \Rightarrow$

$(k, l) \in \Lambda^o$. Operators commutating with a discrete subgroup of translations are described in [3]. These are exactly the Weyl–Heisenberg-blockmatrices and the Janssen coefficients are calculated from the Walnut-representation by transposition and columnwise Fourier transform.

For the purpose of computation and storage on a digital computer system it is convenient to suppress the zeros and store S in a block form

$$\widetilde{S} := \begin{pmatrix} s_{11} & s_{12} & \cdots & s_{1a} \\ s_{21} & & \ddots & s_{2a} \\ \vdots & & & \\ s_{b1} & & & s_{ba} \end{pmatrix}.$$

We will call this type of matrices $b \times a$ blockmatrices [4]. This leads to the following:

Definition 2. Let $S \in \mathcal{M}_N$ and a, b be divisors of N, $M = N/b$. S is called a $b \times a$ blockmatrix if

(1) $S_{m,n} \neq 0 \Rightarrow M | m - n$.
(2) $S_{m,n} = S_{m+a,n+a}$.
(1) means that only every diagonal which is a multiple of M can be different to 0.
(2) means that each diagonal is a-periodic.

The space of all $b \times a$ blockmatrices ($\in \mathcal{M}_N$) will be denoted by $\mathcal{B}_{N,a,b} \subseteq \mathcal{M}_N$.

The compressed representation of $S \in \mathcal{B}_{N,a,b}$ will be denoted as $\widetilde{S} \in \mathcal{M}_{b,a}$. The aim of this article is to present a quick algorithm to find the inverse S^{-1} of S without any further constraint (except for the assumption that S be invertible). We will see that S^{-1} is also in $\mathcal{B}_{N,a,b}$ and therefore can be represented in the compressed form $\widetilde{S^{-1}}$.

3. Related work

In [5,6] an algorithm for the inversion of a special type of blockmatrices ($\in \mathcal{B}_{N,a,b}$), called Weyl–Heisenberg frame matrices, is presented which works by factorization of the analysis-operator W (it holds $S = W^*W$)—for more details see [7]. Here we present an algorithm for inverting arbitrary Gabor-type matrices—in this case, for a given blockmatrix S we do not know the corresponding W so that $S = W^*W$ and the above algorithm cannot be used here.

In [8] another well-known tool to speed up the convergence rate, namely, preconditioning, is used to further improve the numerical efficiency of the calculation. In the proposed method, a special invertible preconditioning matrix P is used, which makes $\|Id - PS\|$ small. Then, instead of $Sx = y$, the equation $PSx = Py$ is solved. So the matrix $I = P \cdot S$ is intended to be an approximation of the identity. If P is a reasonable good approximation, e.g., $\|Id - I\| < 0.1$, then a few iterations are needed in order to find the true dual atom. Moreover, if I is a very good approximation, e.g., $\|Id - I\| \ll 0.1$, then the preconditioning matrix P can already be considered as an approximation of the inverse matrix of S. Note that this is only an approximation of S^{-1} and there are additional constraints on S.

It has also been shown [9] that the inversion of a block-type matrix can be computed in the spreading domain by applying the twisted convolution.

4. Our approach

For simplicity we start our indices for rows and columns with 0 and further exploit them mod(N). Let $A \in \mathcal{M}_N$ and define $D(A)_{d,e} := A_{e,d+e}$—the eth element of the dth diagonal of A. If $A \in \mathcal{B}_{N,a,b}$ then it can be represented obviously by the $b \times a$-matrix $\tilde{A}$ via:

$$\tilde{A}_{z,s} = D(A)_{zM,s} \quad (\text{for } z = 0, \ldots, b-1; \ s = 0, \ldots, a-1).$$

4.1. Multiplication of blockmatrices

Now let $A, B \in \mathcal{B}_{N,a,b}$ and $\tilde{A}, \tilde{B}$ their corresponding $b \times a$-matrices and $C = A \cdot B$.

Recall that

$$\tilde{A}_{z,s} = D(A)_{zM,s} = A_{s,zM+s}$$

and

$$A_{z,s} = \begin{cases} D(A)_{s-z,z} = \tilde{A}_{(s-z)/M \bmod b, z \bmod a} & \text{if } M | z - s, \\ 0 & \text{otherwise.} \end{cases}$$

Then it is not very difficult to see that $C \in \mathcal{B}_{N,a,b}$. Thus $\mathcal{B}_{N,a,b}$ is an algebra. In particular, we have

$$\widetilde{C}_{i,k} = C_{k,i\ M+k} = \sum_{j=0}^{N-1} A_{k,j} \cdot B_{j,i\ M+k}$$

$$\stackrel{(1)}{=} \sum_{j=1}^{b} A_{k,k+j\;M} \cdot B_{k+j\;M,k+i\;M}$$
$$= \sum_{j=0}^{b-1} \widetilde{A}_{j,k} \cdot \widetilde{B}_{(i-j) \bmod b,(k+j\;M) \bmod a}.$$

The blockmatrix–blockmatrix multiplication is (maximum) of order $\mathrm{O}(ab^2)$ because $i = 0, \ldots, b-1$; $k = 0, \ldots, a-1$.

4.2. Inversion

With

$$G^k_{i,j} = \widetilde{B}_{(i-j) \bmod b,(k+j\;M) \bmod a} \tag{2}$$

we get

$$\widetilde{C}_{i,k} = \sum_{j=0}^{b-1} G^k_{i,j} \cdot \widetilde{A}_{j,k} \;\;\Rightarrow\;\; \widetilde{C}_{\bullet,k} = G^k \cdot \widetilde{A}_{\bullet,k}.$$

For inversion we obviously set $C = I_N$, thus

$$\widetilde{C} = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ 0 & & & 0 \\ \vdots & & & \vdots \\ 0 & & & 0 \end{pmatrix}$$

and we have $\widetilde{C}_{\bullet,k} = e_1 = (1, 0, \ldots, 0)^{\mathrm{T}}$, $\forall k$ the first unit vector. Therefore the kth column of $\widetilde{A}$ equals the first column of the inverse of G^k.

To achieve a quick algorithm we exploit the special structure of the equations.

4.2.1. Reduction of redundant equations

We will start with the following example.

Let $N = 72, a = 6, b = 9$,

$$\widetilde{B} = \begin{pmatrix} 11 & 12 & 13 & 14 & 15 & 16 \\ 21 & 22 & 23 & 24 & 25 & 26 \\ 31 & 32 & 33 & 34 & 35 & 36 \\ 41 & 42 & 43 & 44 & 45 & 46 \\ 51 & 52 & 53 & 54 & 55 & 56 \\ 61 & 62 & 63 & 64 & 65 & 66 \\ 71 & 72 & 73 & 74 & 75 & 76 \\ 81 & 82 & 83 & 84 & 85 & 86 \\ 91 & 92 & 93 & 94 & 95 & 96 \end{pmatrix}.$$

Now we compute the corresponding G^0:

$$G^0 = \begin{pmatrix} 11 & 93 & 85 & 71 & 63 & 55 & 41 & 33 & 25 \\ 21 & 13 & 95 & 81 & 73 & 65 & 51 & 43 & 35 \\ 31 & 23 & 15 & 91 & 83 & 75 & 61 & 53 & 45 \\ 41 & 33 & 25 & 11 & 93 & 85 & 71 & 63 & 55 \\ 51 & 43 & 35 & 21 & 13 & 95 & 81 & 73 & 65 \\ 61 & 53 & 45 & 31 & 23 & 15 & 91 & 83 & 75 \\ 71 & 63 & 55 & 41 & 33 & 25 & 11 & 93 & 85 \\ 81 & 73 & 65 & 51 & 43 & 35 & 21 & 13 & 95 \\ 91 & 83 & 75 & 61 & 53 & 45 & 31 & 23 & 15 \end{pmatrix}.$$

We observe that the $G^0 \in \mathscr{B}_{9,3,9} \subseteq \mathscr{M}_9$ is a blockmatrix. The following proposition holds.

Proposition 3. *With $t = \gcd(M, a)$ each G^k is an element in $\mathscr{B}_{b,a/t,b}$.*

Proof. First, note that $(a/t)|b$:

$$a|a \cdot b, a|N \;\;\Rightarrow\;\; a|\gcd(a \cdot b, N) = b \cdot \gcd(a, M) = b \cdot t.$$

Now we still have to show that for each i, j, k:

$$G^k_{i \bmod b, j \bmod b} = G^k_{i+a/t \bmod b, j+a/t \bmod b}.$$

Due to the definition of the G^k we can omit the mod in the indices and understand the G^k as $b \times b$-periodic:

$$G^k_{i+b,j} = \widetilde{B}_{(i+b-j) \bmod b,(k+j\;M) \bmod a} = \widetilde{B}_{(i-j) \bmod b,(k+j\;M) \bmod a} = G^k_{i,j},$$

$$G^k_{i,j+b} = \widetilde{B}_{(i-j-b) \bmod b,(k+(j+b)\;M) \bmod a} = \widetilde{B}_{(i-j) \bmod b,(k+j\;M+b\;M) \bmod a} = G^k_{i,j}$$

since $a|n = b \cdot M$. Therefore

$$G^k_{i+a/t,j+a/t} = \widetilde{B}_{(i+a/t)-(j+a/t) \bmod b,\; k+(j+a/t)\;M \bmod a}$$
$$= \widetilde{B}_{(i-j) \bmod b,(k+j\;M) \bmod a} = G^k_{i,j}$$

because

$$a\Big|\frac{a}{t} \cdot M = a \cdot \frac{M}{\gcd(M, a)}. \qquad \square$$

Corollary 4. *Any $T \in \mathscr{B}_{N,a,b}$ can be inverted in $\gcd(a, N/b)b^3$ operations.*

Let us return to our example (here we have $M = 8$ and $t = 2$) and examine the G^k

corresponding to $\widetilde{G^k}$:

$$\widetilde{G^0} = \begin{pmatrix} \mathbf{11} & 13 & 15 \\ \mathbf{93} & 95 & 91 \\ \mathbf{85} & 81 & 83 \\ \mathbf{71} & 73 & 75 \\ \mathbf{63} & 65 & 61 \\ \mathbf{55} & 51 & 53 \\ \mathbf{41} & 43 & 45 \\ \mathbf{33} & 35 & 31 \\ \mathbf{25} & 21 & 23 \end{pmatrix}, \quad \widetilde{G^2} = \begin{pmatrix} 13 & 15 & \mathbf{11} \\ 95 & 91 & \mathbf{93} \\ 81 & 83 & \mathbf{85} \\ 73 & 75 & \mathbf{71} \\ 65 & 61 & \mathbf{63} \\ 51 & 53 & \mathbf{55} \\ 43 & 45 & \mathbf{41} \\ 35 & 31 & \mathbf{33} \\ 21 & 23 & \mathbf{25} \end{pmatrix},$$

$$\widetilde{G^4} = \begin{pmatrix} 15 & \mathbf{11} & 13 \\ 91 & \mathbf{93} & 95 \\ 83 & \mathbf{85} & 81 \\ 75 & \mathbf{71} & 73 \\ 61 & \mathbf{63} & 65 \\ 53 & \mathbf{55} & 51 \\ 45 & \mathbf{41} & 43 \\ 31 & \mathbf{33} & 35 \\ 23 & \mathbf{25} & 21 \end{pmatrix}.$$

Obviously $\widetilde{G^2}$ and $\widetilde{G^4}$ are shifted versions of the $\widetilde{G^0}$.

The following proposition holds.

Proposition 5. $\widetilde{G^{k+t}_{\cdot,v}} = \widetilde{G^k_{\cdot,v+u}}$, *where u is given as a solution of the diophantic equation $t = uM + va$.*

Proof.

$$\begin{aligned} t = uM + va &\Leftrightarrow t \equiv uM \pmod{a} \\ &\Leftrightarrow k + t + (d+e)\ M \equiv k + (d+e+u)\ M \pmod{a} \end{aligned}$$

this can also be written as

$$\begin{aligned} &\Leftrightarrow k + t + (d+e)\ M \bmod a \\ &= k + (d+e+u)\ M \bmod a \\ &\Rightarrow \widetilde{B}_{-d \bmod b,(k+t+(d+e)\ M) \bmod a} \\ &= \widetilde{B}_{-d \bmod b,(k+(d+e+u)\ M) \bmod a} \end{aligned}$$

$$\overset{\text{Eq.(2)}}{\Rightarrow} G^{k+t}_{e,d+e} = G^k_{e+u,d+e+u}\ (\forall e,d,k)$$

$$\Leftrightarrow \widetilde{G^{k+t}_{\cdot,v}} = \widetilde{G^k_{\cdot,v+u}}. \qquad \square \tag{3}$$

Remark 6. Eq. (3) is equivalent to the statement that the diagonals of G^{k+t} are rotated versions of the diagonals of G^k.

Remark 7. The diophantic equation $t = uM + va$ can be solved very fast by Euklid's algorithm—in MATLAB for example via: $[t, u, v] = \gcd(M, a)$.

Using the following corollary we can reduce the number of equations represented by the G^k from a to t.

Corollary 8. $[G^{k+t}]^{-1} = T_{-u} \circ [G^k]^{-1} \circ T_u$.

Proof. From Remark 6 and the fact that $T_{-u} \circ G^k \circ T_u$ shifts the G^k along the diagonals it follows $G^{k+t} = T_{-u} \circ G^k \circ T_u$. $\square$

4.2.2. Fourier transform of the matrix

Definition 9. Let $T \in \mathscr{M}_n$. The matrix Fourier transformation(MFT) of T is defined by

$$\hat{T} = \mathscr{F} T \mathscr{F}^{-1},$$

where $\mathscr{F}$ is the FFT-matrix, $(\mathscr{F})_{k,l} = (1/\sqrt{n})\mathrm{e}^{-2\pi \mathrm{i} kl/n}$.

This definition implies the following commutating diagram:

$$\begin{array}{ccc} \hat{f} & \xrightarrow{\hat{T}} & \widehat{Tf} \\ \uparrow & & \uparrow \\ f & \xrightarrow{T} & Tf \end{array}$$

4.2.3. Inversion of G^k

Since the inversion of a blockmatrix S is of order ab^3 it is obviously better when $b<a$. If we are given a blockmatrix not fulfilling this condition, we transform it into that shape by using the MFT. Now we know [6,10] that if S is a $b \times a$ matrix then $\hat{S}$ is a $a \times b$ matrix and it is known that $\widehat{\hat{S}^{-1}} = S^{-1}$.

As described above we have to calculate the first column of the $[G^k]^{-1}$ (for $k = 0..a-1$). From Corollary 8 it follows $[G^{k+t}]^{-1} e_0 = T_{-u} \circ [G^k]^{-1} \circ T_u\, e_0 = T_{-u} \circ [G^k]^{-1} e_u$. This is just a rotated version of the uth column of the G^k. Therefore it is sufficient to compute the $[G^0]^{-1}, [G^1]^{-1}, \ldots, [G^{t-1}]^{-1}$. For the inversion of these G^k we have the following possibilities:

- The ordinary inversion of a $b \times b$ matrix — this takes $\mathrm{O}(b^3)$ steps.
- Consider the $G^k \in \mathscr{B}_{b,a/t,b}$ as $b \times a/t$ blockmatrix. Via MFT it can be transformed into $\widehat{G^n} \in \mathscr{B}_{b,b,a/t}$.

This blockmatrix can be inverted (see Corollary 4) in

$$\mathrm{O}\left(\gcd\left(b,\frac{b}{a/t}\right)(a/t)^3\right)=\mathrm{O}\left(\frac{b}{a/t}(a/t)^3\right)=\mathrm{O}\left(\frac{ba^2}{t^2}\right)\text{ steps.}$$

Remark 10. The costs for the MFT and the inverse-MFT are of order $ba/t\log(a/t)$ and can be neglected.

So, S can be inverted in order

$$t\ \min\left(b^3,\frac{ba^2}{t^2}\right)=\min\left(t\ b^3,\frac{ba^2}{t}\right)\text{ steps.}$$

We also can decide at the beginning of our calculation if it is cheaper to calculate via the MFT (applied to S). By neglecting the costs for the MFT and the inverse-MFT we get the total order of this algorithm by

$$\min\left(t\ b^3,\frac{ba^2}{t},s\ a^3,\frac{ab^2}{s}\right)\quad\text{with } s=\gcd\left(b,\frac{N}{a}\right).$$

4.2.4. Practical implementation of the algorithm

Finally, we present a pseudocode-like implementation for our developed algorithm:

Function $A=\text{blockinv}(B,N)$
% Input Parameters:
% $B\ldots$Block matrix to be inverted
% $N\ldots$Size of the original $N\times N$ matrix
$[a,b]=\text{size}(B)$
$M=N/b$
$t=\gcd(N/b,a), s=\gcd(N/a,b)$
If $\min(s\cdot a^3,a\cdot b^2/s)<\min(t\cdot b^3,b\cdot a^2/t)$ % is it cheaper to invert $\widehat{B}$?
Then $A=\widehat{\text{blockinv}(\widehat{B},N)}$
Else
 For $k_1=0:t-1$
  build G^{k_1} via Eq. (2)
  If $\frac{a}{t}\geqslant b$ Then $G^{k_1}_{\text{inv}}=(G^{k_1})^{-1}$ % ordinary inversion of G^{k_1}
  Else
   build $\widetilde{G^{k_1}}$
   $G^{k_1}_{\text{inv}}=\text{blockinv}(\widetilde{G^{k_1}},b)$ % inversion via our algorithm
   For $k_2=0:t:a-1$
    $k=k_1+k_2$
    $r=\bmod(c\cdot k_2/t,b)$ % Cor. 8 applied k_2-times
    $w=(G^{k_1}_{\text{inv}})_{\cdot,r}$ % equal to $(G^{k}_{\text{inv}})_{\cdot,1}$
    $A_{\cdot,k}=\text{rotate}(w,-r)$ % shifts vector w cyclic r positions to the left
   End For
  End If
 End For
End If
End Function

5. Applications

One application is the computation of the dual atoms of a multiwindow system. In this case we have—instead of one atom g in the ordinary case—R windows $g_1,\ldots,g_R$. This gives a better localization in the TF-plane. In particular we have

$$f=\sum_{j=1}^{R}\sum_{\lambda\in\Lambda}V_{g_j}f(\lambda)\cdot\pi(\lambda)\tilde{g}_j,\tag{4}$$

where $\tilde{g}_j=S^{-1}g_j$ and S is the sum of the ordinary Gabor frame operators S_j with windows g_j. Now each S_j has blockmatrix structure and hence S has. The set of duals $(g_{j_0,\lambda})_\lambda$ corresponding to a j_0 can be computed by $\pi(\lambda)S^{-1}g_{j_0}$ since S has a Janssen-representation and commutates with all $\pi(\lambda)$ with $\lambda\in\Lambda$.

6. Conclusion

We have presented a fast method for inverting an important class of block-type-matrices. The complexity is of the same order or below as for the blockmatrix-multiplication algorithms described in [11,12]. An advantage of the presented algorithm is the fact that it applies to all blockmatrices in the given class not only to Weyl–Heisenberg frame matrices [6] or to positive definite and symmetric matrices (conjugate gradient algorithm).

Acknowledgments

The authors would like to thank Hans G. Feichtinger and Tobias Werther for fruitful discussions and their helpful remarks on this paper. We also acknowledge funding by the FWF (Project No. L288) and WWTF (Project MOHAWI 2005-2008).

References

[1] H. Feichtinger, T. Strohmer, Gabor Analysis and Algorithms. Theory and Applications, Birkhäuser, Boston, 1998.

[2] H. Feichtinger, W. Kozek, Quantization of TF lattice-invariant operators on elementary LCA groups, in: H. Feichtinger, T. Strohmer (Eds.), Gabor Analysis and Algorithms. Theory and Applications, Applied and Numerical Harmonic Analysis, Birkhäuser, Boston, MA, 1998, pp. 233–266, 452–488, (Chapter. 7).

[3] H. Feichtinger, H. Führ, K. Gröchenig, N. Kaiblinger, Operators commuting with a discrete subgroup of translations, J. Geom. Anal. 16 (1) (2006) 53–67.

[4] S. Qiu, Gabor-type matrix algebra and fast computations of dual and tight Gabor wavelets, Opt. Engng. 36 (1) (1997) 276–282.

[5] P. Prinz, Theory and algorithms for discrete 1-dimensional Gabor frames, Master's Thesis, University of Vienna, 1996.

[6] T. Strohmer, Numerical algorithms for discrete Gabor expansions, in: H. Feichtinger, T. Strohmer (Eds.), Gabor Analysis and Algorithms. Theory and Applications, Birkhäuser, Boston, 1998, pp. 267–294 (Chapter 8).

[7] K. Gröchenig, Foundations of time-frequency analysis, Applied and Numerical Harmonic Analysis, Birkhäuser, Boston, MA, 2001.

[8] P. Balazs, H.G. Feichtinger, M. Hampejs, G. Kracher, Double preconditioning for Gabor frames, IEEE Trans. Signal Proces. 54 (12) (2006) 4597–4610.

[9] T. Werther, E. Matusiak, Y. Eldar, N. Subbana, A unified approach to dual Gabor windows, IEEE Signal Processing Magazine, Preprint.

[10] P. Balazs, Frame multiplier and irregular Gabor filters with application in time frequency, in: Proceedings ICSV13, 13th International Congress on Sound and Vibration, ARI; Acoustics;HASSIP;NuHAG, 2006.

[11] S. Qiu, Generalized dual Gabor atoms and best approximations by Gabor family, Signal Process. 49 (1996) 167–186.

[12] H. Feichtinger, S. Qiu, Gabor-type matrices and discrete huge Gabor transforms, in: Proceedings of ICASSP'95, vol. 2, IEEE, New York, 1995, pp. 1089–1092.