Convergence of an Iterative Time-Variant Filtering Based on Discrete Gabor Transform

Xiang-Gen Xia and Shie Qian

Abstract— An iterative time-variant filtering based on the discrete Gabor transform (DGT) has been recently proposed by the authors. In this correspondence, we present a proof of the convergence of the iterative algorithm under a sufficient condition on the analysis and synthesis window functions of the DGT. In the meantime, we show that the iterative algorithm refines the least squares solution.

Index Terms—Discrete Gabor transforms, iterative time-variant filtering.

I. INTRODUCTION

Time–frequency (TF) transforms (or analysis) add redundancy in the joint TF domain to the signal in the time domain. They spread noise over the whole TF plane and, meanwhile, contain the signal information in some localized areas as shown in Fig. 1(a)–(c). Therefore, TF transforms usually significantly increase the signal-to-noise ratio in the TF domain; see, for example, [19] for a quantitative analysis. In other words, signals in the TF domain may be more easily detected than in the time domain alone. With this observation, we might use the following idea for extracting the signal in the time domain analogous to traditional linear filtering: Take the TF transform of a noisy signal $s(t)$; mask the TF transform of the signal plane as shown in Fig. 1(d); take the inverse TF transform of the masked TF transform shown in Fig. 1(d) as $\hat{s}(t)$. With traditional linear filtering, there is no question that the Fourier spectrum of the filtered signal $\hat{s}(t)$ has the desired frequency band. This is because the Fourier transform is a one-to-one mapping for finite energy signals. Any signal in the frequency domain corresponds to a unique signal in the time domain. This is, however, no longer true, in general, for TF transforms. Not every signal in the joint TF domain corresponds to a signal in the time domain due to the fact that TF transforms are redundant and not onto. This implies that the TF transform of the filtered $\hat{s}(t)$ may not fall in the masked domain as shown in Fig. 1(d) and (e). With this observation, let us state the general TF synthesis problem (which is also known as the problem of filtering, time-varying, nonstationary wideband signals). Given a user specified, localized TF domain in the TF plane, find the corresponding time domain waveform. The traditional approach to this problem is the least squares solution method, which finds the signal in the time domain that minimizes the squared error between the signal’s TF transform and the desired one (see, for example, [1] for ambiguity functions and [2]–[4] for TF transforms). For other approaches, see, for example, [5]. There are two drawbacks to the least squares solution method. The first one is that although the error between the TF transform of the solution and the desired one is minimized in the mean squared error sense, the TF transform of the solution is not guaranteed to have the desired TF characteristics. This means that the solution may not be the desired one, as illustrated later by examples. As a result, the performance is limited, which will be seen from our numerical results later. The second drawback is the computational complexity when signals are fairly long, which is quite often the case in practice. This is because the calculation of the pseudo inverses of the matrices needed for the least squares
solution method is computationally expensive when their sizes are large. Recently, an iterative time-variant filtering method based on the discrete Gabor transform has been proposed by the authors; see, for example, [6], [7], and [18].

In this correspondence, we present a proof of the convergence of the iterative algorithm proposed in [6] and [7] under a sufficient condition on the window functions of discrete Gabor transforms. We also prove that under these conditions, the first iteration of the iterative algorithm is exactly the least squares solution. Improvement over the least squares solution occurs with more iterations, which can be seen clearly from our numerical examples. This correspondence is organized as follows. In Section II, we first briefly review discrete Gabor transforms and then restate the iterative algorithm for time-varying filtering proposed in [6] and [7]; finally, we present a proof of the convergence. In Section III, we present some numerical examples.

II. CONVERGENCE OF THE ITERATIVE TIME-VARYING FILTERING

In this section, we first describe the iterative time-varying filtering algorithm proposed in [6] and [7] and then study its convergence.

A. Iterative Algorithm

Let us first review the DTG studied by Waxler and Raz [8]. Let a signal $s[k]$, a synthesis window function $h[n]$, and an analysis window function $\gamma[n]$ all be periodic with the same period $L$. Then

$$\begin{align}
s[k] &= \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} C_{m,n} h_m[n] \gamma_n[k] \\
C_{m,n} &= \sum_{k=0}^{L-1} s[k] \gamma_m^* n[k] \\
h_{m,n}[k] &= h[k - m \Delta M] W_{L-k}^{-\Delta N} \gamma[k - m \Delta M] W_{L-k}^{-\Delta N} \\
\gamma_{m,n}[k] &= \gamma[k - m \Delta M] W_{L-k}^{-\Delta N} \gamma\gamma^* \end{align}$$

and $W_L = \exp(j2\pi/L)$, $\Delta M$, and $\Delta N$ are the time and the frequency sampling interval lengths, $M$ and $N$ are the number of sampling points in the time and the frequency domains, $M \cdot \Delta M = N \cdot \Delta N = L$, $M \geq N$ (or $\Delta M \Delta N \leq L$). The coefficients $C_{m,n}$ are called the discrete Gabor transform (DGT) of the signal $s[k]$, and the representation (2.1) is called the inverse discrete Gabor transform (IDGT) of the coefficients $C_{m,n}$.

One condition on the analysis and synthesis window functions $\gamma[k]$ and $h[k]$ obtained by Waxler and Raz [8] is the following identity:

$$\begin{align}
\sum_{k=0}^{L-1} h[k + m N] W_{L-k}^{-m N} \gamma [k] &= \delta[m] h[n]. \\
&\quad 0 \leq m \leq \Delta N - 1 \\
&\quad 0 \leq n \leq \Delta M - 1.
\end{align}$$

The DGT and IDGT can be also represented in the following matrix forms. Let

$$\begin{align}
C &= (C_{0,0}, C_{0,1}, \ldots, C_{M-1,N-1})^T \\
s &= (s[0], s[1], \ldots, s[L-1])^T.
\end{align}$$

The DGT and IDGT can be represented by the $MN \times L$ matrix $G_{MN \times L}$ and $H_{L \times MN}$, respectively, with its $(mN+n)$th row and $k$th column element $\gamma_{m,n}[k]$. The IDGT can be represented by the $L \times MN$ matrix $H_{L \times MN}$ with its $k$th row and $(mN+n)$th column element $h_{m,n}[k]$. Thus

$$\begin{align}
C &= G_{MN \times L} s \\
&\quad \text{and} \\
H_{L \times MN} C &= G_{MN \times L} s.
\end{align}$$

The condition (2.5) implies that

$$H_{L \times MN} G_{MN \times L} = I_{L \times L},$$

where $I_{L \times L}$ is the $L \times L$ identity matrix.

As mentioned in the introduction, the oversampling, that corresponds to the case when $MN > L$ of the DGT adds redundancy and is usually preferred for noise reduction applications. This can be also seen from [19], where it is proved that the SNR in the transform domain for short-time Fourier transforms increases when the sampling rate increases. From (2.1)–(2.5) and (2.6) and (2.7), we can see that an $L$-dimensional signal $s$ is transformed into an $MN$-dimensional signal $C$, and $MN$ is greater than $L$ due to the oversampling. Therefore, only a small set of $MN$-dimensional signals in the TF plane have their corresponding time waveforms with length $L$. Let $D_{MN \times MN}$ denote the mask transform, specifically, a diagonal matrix with diagonal elements either 0 or 1. Let $s$ be a signal with length $L$ in the time domain. The first step in the time-varying filtering is to mask the TF transform of $s$

$$C_1 = D_{MN \times MN} G_{MN \times L} s$$

where $D_{MN \times MN}$ masks a desired domain in the TF plane. Since the DGT $G_{MN \times L}$ is a redundant transformation, the IDGT of $C_1$, $H_{L \times MN} C_1$ may not fall in the mask. In another words, generally

$$G_{MN \times L} H_{L \times MN} C_1 \neq D_{MN \times MN} G_{MN \times L} H_{L \times MN} C_1,$$

where $MN > L$, which is illustrated in Fig. 1(e). Notice that in the critical sampling case, i.e., $MN = L$, the inequality (2.8) becomes equality. An intuitive method to reduce the difference between the right- and the left-hand sides of (2.8) is to mask the right-hand side of (2.8) again and repeat the procedure, which leads to our iterative algorithm

$$\begin{align}
s_0 &= s \\
C_{i+1} &= D_{MN \times MN} G_{MN \times L} s_i \\
s_{i+1} &= H_{L \times MN} C_{i+1}, \\
&\quad i = 0, 1, 2, \ldots
\end{align}$$

The above iterative algorithm is illustrated in Fig. 2.

The iterative algorithm (2.9)–(2.11) is an alternating projection procedure. It is used quite often in signal recovery applications, such as signal extrapolation and phase retrieval. The important issues for this algorithm include the following: When does the algorithm converge? If it converges, to what does it converge? We study these questions in the next subsection.

Before going to the convergence, let us see what the least squares solution is. Based on the definition, the least squares solution is the $L \times 1$ vector $\hat{s}$ that minimizes

$$\begin{align}
\|G_{MN \times L} \hat{s} - D_{MN \times MN} G_{MN \times L} s\| \\
&= \min_{\hat{s}} \|G_{MN \times L} \hat{s} - D_{MN \times MN} G_{MN \times L} s\| \\
&\quad \text{subject to} \\
&\quad \hat{s} = 0, 1, 2, \ldots
\end{align}$$

where $\| \cdot \|$ is the usual Euclidean norm. Then

$$\begin{align}
\hat{s} &= (G_{MN \times L} G_{MN \times L})^{-1} G_{MN \times L} D_{MN \times MN} G_{MN \times L} s \\
&\quad \text{where} \\
&\quad \text{subject to} \\
&\quad \hat{s} = 0, 1, 2, \ldots
\end{align}$$

which is the first step $s_0$ of the iterative algorithm (2.9)–(2.11).

Authorized licensed use limited to: TIANJIN UNIVERSITY. Downloaded on February 5, 2009 at 08:38 from IEEE Xplore. Restrictions apply.
B. Convergence of the Iterative Algorithm

In this subsection, we want to have a condition on the window functions \( h[n] \) and \( \gamma[n] \) for the convergence of (2.9)–(2.11). We show that under this condition, the limit of the sequence \( s_k \) in (2.11) has its DGT falling in the mask \( D_{MN \times MN} \).

Rewrite (2.9)–(2.11) as

\[
C_{l+1} = D_{MN \times MN} G_{MN \times XL} H_{L \times MN} C_l = (D_{MN \times MN} G_{MN \times XL} H_{L \times MN})^t \cdot D_{MN \times MN} G_{MN \times XL} s
\]

where \( l = 1, 2, \cdots \). If we can prove that both matrices \( D_{MN \times MN} \) and the product \( G_{MN \times XL} H_{L \times MN} \) are orthogonal projections, the above iterative algorithm converges by the alternating orthogonal projection theorem (see [14]). A matrix \( A \) is an orthogonal projection [14], [15], which means that i) \( A^2 = A \) and ii) \( A^t = A \), and vice versa.

It is clear that the mask matrix \( D_{MN \times MN} \) is an orthogonal projection. For the product matrix \( G_{MN \times XL} H_{L \times MN} \), we have, by (2.7)

\[
(G_{MN \times XL} H_{L \times MN})^t = G_{MN \times XL} H_{L \times MN}.
\]

i.e., the condition i) for an orthogonal projection is satisfied. For the Hermitian property ii), we need the following condition on the window functions \( h \) and \( \gamma \) (details can be found in the Appendix)

\[
\sum_{k=0}^{\Delta N-1} \gamma^*[N + k]h[N + k + m\Delta M] = \sum_{k=0}^{\Delta N-1} h^*[N + k]\gamma[N + k + m\Delta M]
\]

for \( k = 0, 1, \cdots, N - 1 \) and \( m = 0, 1, \cdots, M - 1 \).

With the above condition, the following lemma is not hard to prove.

**Lemma 1:** The product matrix \( G_{MN \times XL} H_{L \times MN} \) is Hermitian if and only if (2.16) for the window functions \( h \) and \( \gamma \) holds.

**Proof:** To prove Lemma 1, we first re-express both DGT and IDGT matrices \( G_{MN \times XL} \) and \( H_{L \times MN} \) by taking the advantage of the forms in (2.1)–(2.4).

For \( l = 0, 1, \cdots, M - 1 \), let \( \Gamma_l \) be the \( L \times L \) diagonal matrix

\[
\Gamma_l = \text{diag}(\gamma*[0 - l\Delta M], \gamma*[1 - l\Delta M], \cdots, \gamma*[L - 1 - l\Delta M])
\]

and let \( W_{N \times N} \) be the \( N \)-point discrete Fourier transform matrix, i.e., \( W_{N \times N} = (W_{N \times N}^{mn})_{0 \leq m, n \leq N - 1} \). Let \( W_{N \times N} \) be the following \( N \times L \) matrix consisting of \( \Delta N \) many \( W_{N \times N} \) submatrices:

\[
W_{N \times N} = (W_{N \times N}, W_{N \times N}, \cdots, W_{N \times N})
\]

(2.18)

Then, \( G_{MN \times XL} \) can be rewritten as

\[
G_{MN \times XL} = \left( \begin{array}{c} W_{N \times XL} \Gamma_0 \\ W_{N \times XL} \Gamma_1 \\ \vdots \\ W_{N \times XL} \Gamma_{M-1} \end{array} \right)
\]

(2.19)

Similarly, the matrix \( H_{L \times MN} \) can be rewritten as

\[
H_{L \times MN} = (\Lambda_0 W_{N \times XL}^{\dagger}, \Lambda_1 W_{N \times XL}^{\dagger}, \cdots, \Lambda_{M-1} W_{N \times XL}^{\dagger})
\]

(2.20)

where \( \Lambda_l \) is the following \( L \times L \) diagonal matrix similar to \( \Gamma_l \):

\[
\Lambda_l = \text{diag}(h[0 - l\Delta M], h[1 - l\Delta M], \cdots, h[L - 1 - l\Delta M])
\]

(2.21)

Therefore

\[
G_{MN \times XL} H_{L \times MN} = \left( W_{N \times XL} \Gamma_m \Lambda_n W_{N \times XL}^{\dagger} \right)_{m, n \leq M-1}
\]

(2.22)

\[
= \left( \begin{array}{cccc} W_{N \times XL} \Gamma_0 \Lambda_0 W_{N \times XL}^{\dagger} & \cdots & W_{N \times XL} \Gamma_{M-1} \Lambda_{M-1} W_{N \times XL}^{\dagger} \\ \vdots & \ddots & \vdots \\ \vdots & \ddots & \vdots \\ W_{N \times XL} \Gamma_{M-1} \Lambda_0 W_{N \times XL}^{\dagger} & \cdots & W_{N \times XL} \Gamma_{M-1} \Lambda_{M-1} W_{N \times XL}^{\dagger} \end{array} \right)
\]

For \( G_{MN \times XL} H_{L \times MN} \) to be Hermitian, we need to have

\[
W_{N \times XL} \Gamma_m \Lambda_n W_{N \times XL}^{\dagger} = W_{N \times XL} \Gamma_n \Lambda_m W_{N \times XL}^{\dagger}
\]

(2.23)
With the form (2.18) for $\mathbf{W}_{N \times L}$, the above (2.23) can be simplified as follows:

$$
\begin{align}
\mathbf{W}_{N \times L} \Gamma_m \Lambda_n \mathbf{W}_{N \times L}^t &= \sum_{l=0}^{L-1} W_{N \times N} \\
&= \sum_{l=0}^{L-1} \text{diag}( \gamma^l [N + 0 - m \Delta M], \cdots, \gamma^l [N + N - 1 - m \Delta M]) W_{N \times N}^l \\
&= W_{N \times N} \sum_{l=0}^{L-1} h^l [N + 0 - n \Delta M], \cdots, h^l [N + N - 1 - n \Delta M] W_{N \times N}^l.
\end{align}
$$

Therefore, the Hermitian property ii) for the matrix $G_{MN \times KN} H_{KL \times MN}$ holds if and only if

$$
\sum_{l=0}^{L-1} \gamma^l [N + k - m \Delta M] h^l [N + k - n \Delta M]
$$

for $k = 0, 1, \cdots, N - 1$ and $0 \leq m, n \leq M - 1$.

Since $h[n]$ and $\gamma[n]$ are periodic with period $L = M \Delta M$, (2.24) is equivalent to

$$
\sum_{l=0}^{L-1} \gamma^l [N + k - m \Delta M] h^l [N + k + n \Delta M]
$$

for $k = 0, 1, \cdots, N - 1$ and $0 \leq m, n \leq M - 1$. Notice that the difference between (2.24) and (2.25) is the difference of the signs in the front of the variable $n$.

Condition (2.16) can be obtained from condition (2.25). This proves Lemma 1.

With Lemma 1 and the alternating orthogonal projection theorem, we have proved the following convergence result.

**Theorem 1**: When the synthesis and the analysis window functions $h[n]$ and $\gamma[n]$ satisfy (2.16), the iterative algorithm (2.9)–(2.11) converges.

There are two trivial cases where (2.16) holds. The first case is the orthogonal-like case where $h[n] = \gamma[n]$ for all integer $n$. The second case is the critical sampling case in which $\Delta M = N$. Notice that the continuous Gabor transform is never orthogonal-like unless the window functions are badly localized in the frequency domain. This, however, is not the case for the DGT. The most orthogonal-like solution was studied by Qian et al. in [9]–[11]. They showed that it is possible to have the analysis window function $\gamma$ very close to the synthesis window function $h$ when $h$ is truncated Gaussian. The error between $h$ and $\gamma$ is less than $2 \times 10^{-6}$ while they are of unit energy, and therefore, the error is negligible. We will see numerical results later in the next section.

We next want to see what the limit of the iterative algorithm (2.9)–(2.11) is, under (2.16). Assume $\mathbf{x}$ is the limit of the sequence $\mathbf{u}$, and $\mathbf{C}$ is the limit of $\mathbf{C}_j$. Then

$$
\mathbf{C} = D_{MN \times KN} G_{MN \times KN} H_{KL \times MN} \mathbf{C} = D_{MN \times KN} G_{MN \times KN} \mathbf{x},
$$

and

$$
\mathbf{x} = H_{KL \times MN} D_{MN \times KN} G_{MN \times KN} \mathbf{C}.
$$

We want to prove

$$
G_{MN \times KN} \mathbf{x} = D_{MN \times KN} G_{MN \times KN} \mathbf{x},
$$

i.e., the DGT of $\mathbf{x}$ falls in the mask $D_{MN \times KN}$. Since $G_{MN \times KN} H_{KL \times MN}$ is an orthogonal projection and

$$
D_{MN \times KN} G_{MN \times KN} \mathbf{x}
$$

we have that

$$
G_{MN \times KN} \mathbf{x} = -(I - G_{MN \times KN} H_{KL \times MN}) D_{MN \times KN} G_{MN \times KN} \mathbf{x}
$$
where $\perp$ means orthogonal. Since $D_{MN \times MN}$ is also an orthogonal projection and
\[
-(I - G_{MN \times L} H_{L \times MN}) D_{MN \times MN} G_{MN \times L} \bar{x} = (I - D_{MN \times MN}) G_{MN \times L} \bar{x}
\]
we have $G_{MN \times L} \bar{x} = D_{MN \times MN} G_{MN \times L} \bar{x}$. This proves the following Theorem 2.

**Theorem 2:** Under (2.16), the DGT of the limit $\bar{x}$ of the iterative algorithm (2.9)--(2.11) falls in the mask $D_{MN \times MN}$, i.e.,
\[
G_{MN \times L} \bar{x} = D_{MN \times MN} G_{MN \times L} \bar{x}.
\]

(2.27)

With the above result, we might ask whether it violates the known fact that an image of a TF transform of a signal in the TF plane cannot be of compact support. This is because that a signal cannot be time and band limited simultaneously. To answer this question, we first need to know that the above known fact is true for continuous TF transforms. Moreover, its proof is based on the marginal properties of TF transforms. It may not be true for discrete TF transforms. In other words, discrete TF transforms may have compact support [6].

For the least squares solution $\bar{x}$ in (2.13), its Gabor transform is $G_{MN \times L} \bar{x}$. Since $G_{MN \times L} H_{L \times MN}$ is an orthogonal projection, by (2.26), we have proved that the least squares solution
\[
\bar{x} = H_{L \times MN} D_{MN \times MN} G_{MN \times L} \bar{x} = s_1.
\]

This proves the following Theorem 3.

**Theorem 3:** Under (2.16), the first iteration $s_1$ of the iterative algorithm (2.9)--(2.11) is equal to the least squares solution in (2.13), i.e., $s_1 = \bar{x}$.

With Theorem 3, we will see in the next section that the iterative algorithm (2.9)--(2.11) improves the least squares solution when the number of iterations increases, and in the meantime, we do not need to compute the inverse matrix in (2.13). Theorem 3 also provides another
way to compute the least squares solution when (2.16) holds on the window functions. Note that the least squares solution in (2.13) does not depend on the synthesis window function $h[n]$. This means that all the least squares solutions are the same for all pairs of synthesis and analysis window functions as long as the analysis window functions are the same, such as the Gaussian function. Therefore, the improvement from the iterative algorithm with window functions satisfying (2.16) is over the least squares solutions not only for the window functions satisfying (2.16) but for other window functions as well.

III. Numerical Examples

In this section, we test two sets of window functions of the DGT. The first set of window functions are the most orthogonal-like ones obtained from [11], [18], and [19]. For this set of window functions, their difference, and the absolute values of the differences between the left- and right-hand sides of (2.16) are shown in Fig. 3, respectively. The second set of window functions only satisfies the Wexler–Raz condition (2.5), and correspondingly, they are shown in Fig. 4. The test signal is $s[n] = x[n] + \eta[n]$, where $x[n] = \cos(2\pi((n + 1)/115)^2)$, and $\eta[n]$ is white Gaussian noise. The mean square errors between the true signal $x(n)$ and the filtered ones are shown in Figs. 5 and 6 for the two sets of window functions, respectively. We can clearly see the performance difference.

IV. Conclusion

In this correspondence, we presented a convergence proof of the iterative time-variant filtering algorithm proposed in [6] and [7]. We proved that under the condition, the limit of the time waveforms from the iterative algorithms has the desired TF characteristics. We also proved that, under the condition, the first iteration is equal to the least squares solution.

REFERENCES