FREQUENCY-UNDERSAMPLED SHORT-TIME FOURIER TRANSFORM

Daichi Kitahara

College of Information Science and Engineering, Ritsumeikan University, Shiga, Japan

ABSTRACT

The short-time Fourier transform (STFT) usually computes the same number of frequency components as the frame length while overlapping adjacent time frames by more than half. As a result, the number of components of a spectrogram matrix becomes more than twice the signal length, and hence STFT is hardly used for signal compression. In addition, even if we modify the spectrogram into a desired one by spectrogram-based signal processing, it is re-changed during the inversion as long as it is outside the range of STFT. In this paper, to reduce the number of components of a spectrogram while maintaining the analytical ability, we propose the frequency-undersampled STFT (FUSTFT), which computes only half the frequency components. We also present the inversions with and without the periodic condition, including their different properties. In simple numerical examples of audio signals, we confirm the validity of FUSTFT and the inversions.

Index Terms—Short-time Fourier transform, redundancy, spectrogram, inversions with and without periodicity, tridirectional system.

1. INTRODUCTION

Time-frequency analysis is to capture the temporal variations of frequency components of a target signal [1]–[18]. In audio signal processing, the short-time Fourier transform (STFT) [1]–[9] is the most commonly used time-frequency analysis method since STFT inherits the robustness of the Fourier transform, against time shifts [10]. The result of STFT is called a spectrogram, and it is often expressed as a matrix. In addition to using spectrograms for signal analysis and feature extraction, we can also generate desired time-domain signals through modification of the spectrograms themselves [19]–[26]. This paper is particularly aware of the latter usage of the spectrograms.

In most cases of the engineering field, STFT is used for discrete-time signals, and a window function has a compact support. In such a case, the support length of the discrete-time window function, called the window length, is directly equal to the length of each time frame, called the frame length. Typically, we calculate the same number of frequency components as the frame length in each time frame by using the fast Fourier transform (FFT). We call this the discrete STFT.

We can also compute more frequency components, although that are linearly dependent, in each time frame by padding zeros before FFT. We call this the frequency-oversampled STFT (FOSSTFT). Both the discrete STFT and FOSSTFT are also called the windowed discrete Fourier transform (WDFT) or the discrete Gabor transform (DGT), but in this paper we switch the names by focusing on the inequality between the frame length and the number of frequency components.

The inversions, based on the Moore–Penrose pseudoinverse, for the discrete STFT and FOSSTFT can be easily computed by using the so-called canonical dual window [22], [28], whose window length is the same as the frame length, after the inverse FFT (IFFT). These inversions for the discrete STFT and FOSSTFT are called “painless” [29].

From the facts that (i) human hearing is sensitive to block boundary artifacts and (ii) a window function makes signal values small at both ends of each time frame, we usually overlap adjacent frames by more than half in the computation of the discrete STFT and FOSSTFT. As a result, the number of components of a spectrogram matrix becomes more than twice the original signal length, and hence the spectrogram is hardly used for signal compression. Moreover, even if we set components of a spectrogram to desired values by a spectrogram-based signal processing technique such as [19]–[26], there is a risk that both magnitudes and phases would be greatly changed during the inversion unless the desired spectrogram belongs to the range of STFT.

As almost nonredundant time-frequency analysis methods, the modified discrete cosine transform (MDCT) [11], that is used in coding formats for audio signals such as MP3 and AAC, and the discrete Wilson transform (DWT) [12], [13], that is hardly used in an application because of a strict condition for a window function, are known. The results of the discrete STFT and FOSSTFT are complex-valued, while those of MDCT and DWT are real-valued and unsuitable for analysis of complex-valued signals. Moreover, MDCT and DWT are sensitive to time shifts differently from STFT. As a complex version of MDCT, the modulated complex lapped transform (MCLT) [14] is known but it is almost the same as the discrete STFT (see Footnote 8).

In this paper, to suppress the redundancy of a spectrogram while maintaining the original analytical ability, we propose the frequency-undersampled STFT (FUSTFT), which calculates only half the frequency components of the discrete STFT in each time frame. From the fact that the energy of a target signal spreads along the frequency axis by multiplying a smooth window function, FUSTFT maintains the features of the original spectrogram of the discrete STFT despite the undersampling. In fact, Stanković has already proposed the special case of FUSTFT in [30], that is equivalent to Type-I FUSTFT in (11) with $\xi = \frac{2\pi}{N}$. Hence, this paper is the generalization of [30].

By using FUSTFT, we can easily obtain efficient spectrograms, including almost nonredundant ones, while its inversion is not so simple differently from those for the discrete STFT and FOSSTFT, i.e., its inversion process changes dependently on the signal length [31]. We realize the inversions with and without the periodic condition, which is assumed in [6]–[8], by directly solving the least squares problems. In [31] the general frequency-undersampling is considered while this paper treats only the half frequency-undersampling and clarifies that both two different inversions can always be computed very quickly.

2. DEFINITIONS OF STFT AND ISTFT IN THIS PAPER

Let $\mathbb{R}$ and $\mathbb{C}$ be the sets of all real numbers and all complex numbers, respectively. The imaginary unit is denoted by $i \in \mathbb{C}$, i.e., $i^2 = -1$. We write vectors and matrices with boldface small and capital letters, respectively. We express the transpose operator as $\cdot^\top$ and the adjoint operator as $\cdot^\dagger$. We express the composition of mappings as $\circ$ and the inverse of a nonsingular matrix by $(\cdot)^{-1}$. We express the $\ell_2$ norm of a vector as $\|\cdot\|_2$ and the Frobenius norm of a matrix as $\|\cdot\|_F$. The floor and ceiling functions are denoted by $\lfloor\cdot\rfloor$ and $\lceil\cdot\rceil$, respectively. For $a > 0$, we define $\text{mod}_a : \mathbb{R} \to [0, a)$ by $\text{mod}_a (b) := b - \lfloor \frac{b}{a} \rfloor a$.

1. The redundancies of MDCT and DWT occur at the first and last frames.

2. Strictly speaking, sampling points of a window function are also changed.
2.1. Continuous-Time / Discrete-Time / Discrete STFT

Let \( x : \mathbb{R} \rightarrow \mathbb{C} \) be a real-valued or complex-valued continuous-time signal. In this paper, with a real-valued window function \( w : \mathbb{R} \rightarrow \mathbb{R} \), we define the continuous-time STFT of \( x(t) \) by

\[
X(f, t) = \int_{-\infty}^{\infty} x(\tau) w(\tau - t) e^{-2\pi i f (\tau - t)} \, d\tau
\]

and the discrete-time STFT by

\[
X_d(f, t) = \sum_{\tau = -\infty}^{\infty} x(\tau T_s) w(\tau T_s - t) e^{-2\pi i f (\tau T_s - t)}
\]

where \( f \in \mathbb{R}, t \in \mathbb{R}, T_s > 0 \) is the sampling interval of a discrete-time signal. Let \( T_s = x[\tau] := x(\tau T_s) \), and \( f = \frac{\tau}{T_s} \) is the sampling frequency. From (3), \( X_d(f, t) \) is periodic on \( f \) with period \( f_s \), and hence we can restrict \( f \) to \( (-\frac{f_s}{2}, \frac{f_s}{2}) \) or \( f \in [0, f_s) \) in the discrete-time STFT.

In what follows, let \( L_w \geq 2 \) be an integer, and we suppose that the window function \( w(t) \) has a compact support of length \( L_w T_s \), i.e., \( w(t) \neq 0 \) for almost all \( t \in [0, L_w T_s) \) and \( w(t) = 0 \) otherwise, and \( w(t) \) is a symmetric curve, i.e., \( w(\frac{L_w T_s - t}{T_s}) = w(\frac{L_w T_s - t}{T_s}) \) for all \( t \in \mathbb{R} \) and \( w(t) \neq 0 \) for almost all \( t \in (0, L_w T_s) \). Under these assumptions the continuous-time STFT in (1) is expressed as

\[
X(f, t) = \int_{0}^{L_w T_s} x(\tau + t) w(\tau) e^{-2\pi i f (\tau + \frac{t}{T_s})} \, d\tau
\]

In (2), we can define the discrete-time STFT for all \( t \in \mathbb{R} \), but there is almost no need to calculate \( X_d(f, t) \) at intervals shorter than \( T_s \). Let \( t \) be the time frame index. With an integer frame shift \( \xi \leq L_w \), we discretize the time \( t \) of \( X_d(f, t) \) by \( t = (\xi - L_w + \xi - \frac{1}{2})T_s \), i.e.,

\[
X_d(f, (\xi - L_w + \xi - \frac{1}{2})T_s) = \sum_{\tau = 0}^{L_w - 1} x[\tau + \xi - L_w + \xi] w[\tau] e^{-2\pi i f \tau T_s}
\]

where \( f \in [0, f_s) \) and \( w[\tau] := w(\tau + \frac{1}{2})T_s \) for \( f \neq \frac{\tau}{T_s} \). Next, let \( \xi \) be the frequency index, and discretize the frequency \( f \) in (5) by \( f = \frac{\tau}{T_s} \) since the maximum number of independent frequency components computed in each time frame is \( L_w \). For a continuous-time signal \( x := (x[0], x[1], \ldots, x[L_w - 1]) \in \mathbb{C}^{L_w} \) of length \( L_w > L_s \), we define

\[
\text{STFT}(x)[k, l] = X_d(k, \frac{\tau}{T_s}, (\xi - L_w + \xi - \frac{1}{2})T_s)
\]

as the discrete STFT in this paper, where \( k = 0, 1, \ldots, L_w - 1 \) and \( l = 0, 1, \ldots, \lfloor \frac{L_w + L_s - 2k}{2} \rfloor \).

In (6), by assuming that \( x(t) = 0 \) for all \( t \in (-\infty, -\frac{L_w}{2}) \cup (\frac{L_w - 1}{2}, \infty) \), we padded \( L_w - \xi \) zeros at the beginning of \( x \) and \( \lfloor \frac{L_w + L_s - 2k}{2} \rfloor \) zeros at the end. The discrete STFT in (6) is easily computed by FFT after multiplying the window function \( w[\tau] \) and extracted time frame signals of length \( L_w \).

We discretize the frequency-oversampled STFT (FOSTFT). Specifically, let \( N_s \) be a positive integer, discretize \( f \) in (5) by \( f = \frac{\tau}{L_w + f_s} \), and we define

\[
\text{FOSTFT}(x)[k, l] = X_d(k, \frac{\tau}{L_w + f_s}, (\xi - L_w + \xi - \frac{1}{2})T_s)
\]

where \( k = 0, 1, \ldots, L_w + N_s - 1 \) and \( l = 0, 1, \ldots, \lfloor \frac{L_w + L_s + N_s - 2k}{2} \rfloor \).

In each time frame, we can also compute more frequency components, than \( L_w \). We call this transform the frequency-oversampled STFT (FOSTFT). Specifically, let \( N_s \) be a positive integer, discretize \( f \) in (5) by \( f = \frac{\tau}{L_w + N_s f_s} \), and we define

\[
\text{FSTFT}(x)[k, l] = X_d(k, \frac{\tau}{L_w + N_s f_s}, (\xi - L_w + \xi - \frac{1}{2})T_s)
\]

where \( k = 0, 1, \ldots, L_w + N_s - 1 \) and \( l = 0, 1, \ldots, \lfloor \frac{L_w + L_s + N_s - 2k}{2} \rfloor \).

2.2. Inversions for the Discrete STFT and FSTFT

The discrete STFT in (6) is a linear mapping and we express its range as \( \mathbb{R}^L := \{ X \in \mathbb{C}^{L_w} \times \mathbb{C}^{L_w} \mid \exists x \in \mathbb{C}^{L_w} \} \). As long as \( \xi < L_w \), the discrete STFT is redundant, and there are innumerable linear mappings that recover, from a complex spectrogram \( X \in \mathbb{R}^L \), the corresponding signal \( x \). To recover the most consistent signal \( x \) from \( X \), we define the inverse STFT (ISTFT) by

\[
\text{ISTFT}(X) = \arg \min_{x \in \mathbb{C}^{L_w}} \| X - \text{STFT}(x) \|_2^2
\]

We express the discrete STFT as \( S : \mathbb{C}^{L_w} \rightarrow \mathbb{C}^{L_w} \times \mathbb{C}^{L_w} \). Then, since ISTFT in (10) is the Moore–Penrose pseudoinverse of \( S \), we have \( \text{ISTFT}(X) = (S^H \circ S)^{-1} \circ S^H \). The matrix \( S^H \circ S \) is diagonal, and its diagonal components are periodic with period \( \xi \). Hence, ISTFT can be quickly computed by using IFFT and the pre-designed canonical dual window (22). For FOSTFT in (9) we can compute its inversion by using the same canonical dual window.

3. FREQUENCY-UNDERSAMPLED STFT

It is known that human hearing is sensitive to block boundary artifacts. Moreover, in each time frame, a non-rectangular window \( w[\tau] \) makes signal values at both ends very small. From these facts, we usually restrict the frame shift \( \xi \) to \( \xi \leq \frac{L_s}{4} \) in (6) and (9). However, in this usual case, the number of components of a spectrogram matrix is more than twice the signal length \( L_w \), which is not suitable for signal compression. In addition, even if we obtain desired spectrograms through spectrogram-based signal processing, their components will be changed by ISTFT unless they belong to the range \( \mathbb{R}^L \).

In what follows, \( L_w \) is a multiple of \( 4 \). For more efficient time-frequency analysis, we propose the frequency-undersampled STFT (FUSTFT), that computes \( \frac{L_w}{2} \) frequency components in each frame. We discretize \( f \) in (5) by \( f = \frac{\tau}{L_w} \), and define Type-I FUSTFT as

\[
\text{FUSTFT}_1(x)[k, l] = X_d(k, \frac{\tau}{L_w}, (\xi - L_w + \xi - \frac{1}{2})T_s)
\]

\[
= \sum_{r=0}^{L_w - 1} x[r + \xi - L_w + \xi] w[\tau] e^{-2\pi i \frac{2k \cdot \tau}{L_w}}
\]

Discretize \( f \) in (5) by \( f = \frac{2k + 1}{L_w} \), and define Type-II FUSTFT as

\[
\text{FUSTFT}_2(x)[k, l] = X_d(k, \frac{2k + 1}{L_w}, (\xi - L_w + \xi - \frac{1}{2})T_s)
\]

\[
= \sum_{r=0}^{L_w - 1} x[r + \xi - L_w + \xi] w[\tau] e^{-2\pi i \frac{2k + 1 \cdot \tau}{L_w}}
\]
4. TWO DIFFERENT INVERSIONS FOR FUSTFT

4.1. Inversion Based on the Standard Pseudoinverse

Differently from cases of the discrete STFT and FOSTFT, the inversion for FUSTFT is not simple, i.e., the canonical dual window does not exist and the computation process changes dependently on \( L_x \). In this paper, we realize the inversion by solving the problem similar to (10). Let \( \mathcal{S} : \mathbb{C}^{L_w \times \ell} \rightarrow \mathbb{C}^{L_w \times \ell} \times \left\lfloor \frac{L_w + L_x - 2q\pi}{\pi} \right\rfloor \) be one of the linear mappings (11), (12), and (13). Then the inversion based on the pseudoinverse is expressed as \((\mathcal{S}^\dagger \circ \mathcal{S})^{-1} \circ \mathcal{S}^\dagger\). In FUSTFT cases, \( \mathcal{S}^\dagger \circ \mathcal{S} \) is diagonal components are

\[
a_i = \frac{L_w}{2} \sum_{l=0}^{\left\lfloor \frac{L_w}{2} \right\rfloor - 1} w^2 |m_l + i\xi|
\]

for all the three types, where \( m_l = \text{mod}_{L_x}(i + L_x) \) and \( w[\tau] = 0 \) for \( \tau \geq L_w \). For Type-I FUSTFT, nonzero nondiagonal components are

\[
b_i = \frac{L_w}{2} \sum_{l=0}^{\left\lfloor \frac{L_w}{2} \right\rfloor - 1} w[m_l + i\xi] w[m_l + i\xi + \frac{L_w}{2}].
\]

For Type-II, each \( b_i \) is equal to (17) multiplied by \(-1\). For Type-III,

\[
b_i = \frac{L_w}{2} \sum_{l=0}^{\left\lfloor \frac{L_w}{2} \right\rfloor - 1} (-1)^{i+L_x} w[m_l + i\xi] w[m_l + i\xi + \frac{L_w}{2}].
\]

\( a_i \) in (16) and \( b_i \) in (17) are periodic with period \( \xi \) while \( b_i \) in (18) is periodic with period \( 2\xi \). We only have to compute them up once.

For a given complex spectrogram \( X \in \mathbb{C}^{L_w \times \ell} \) and \( \mathcal{S}^\dagger \circ \mathcal{S} \), define \( y := (y[0], y[1], \ldots, y[L_x-1])^T := \mathcal{S}^\dagger \circ \mathcal{S} \). Then, the unique solution \( x \) to a linear system \((\mathcal{S}^\dagger \circ \mathcal{S}) \circ x = y \) is the inversion result of \( X \). This linear system is decomposed into \( \frac{L_w}{2} \) independent subsystems

\[
\begin{bmatrix}
  a_{00} & a_{01} & a_{10} & a_{11} \\
  b_{00} & b_{01} & b_{10} & b_{11} \\
  a_{L_w-1,0} & a_{L_w-1,1} & a_{L_w-1,0} & a_{L_w-1,1} \\
  b_{L_w-1,0} & b_{L_w-1,1} & b_{L_w-1,0} & b_{L_w-1,1}
\end{bmatrix}
= \begin{bmatrix}
  x[0] \\
  x[1] \\
  \vdots \\
  x[L_w - 1]
\end{bmatrix}
= \begin{bmatrix}
  y[0] \\
  y[1] \\
  \vdots \\
  y[L_w - 1]
\end{bmatrix}
\]

(19)

\( (i = 0, 1, \ldots, L_w - 1) \), where \( m_l = \frac{2L_w - 2q\pi}{\pi} \), \( a_{ii} = a_{i,i+1} \), and \( b_{ii} := b_{i,i+1} \). Since the matrices in the left side of (19) are tridiagonal matrices, their LU decompositions can be computed in \( \mathcal{O}(n) \) [33], and the inversion result \( x \) is also obtained from \( y \) in \( \mathcal{O}(n) \) [33].

In particular, when \( \text{mod}_{L_w}(\frac{L_w}{2}) = 0 \) for Type-I and Type-II, or \( \text{mod}_{2\ell}(\frac{L_w}{2}) = 0 \) for Type-III, we have \( a_{00} = a_{11} = \cdots = a_{n-1,n-1} = a_0 \) and \( b_{00} = b_{11} = \cdots = b_{n-1,n-1} = b_0 \), and hence the matrices in the left side of (19) are tridiagonal Toeplitz matrices. In such cases, the eigenvalues are \( \lambda_{ii} = a_i + 2b_i \cos\left(\frac{2\pi q\xi}{L_x}\right) > a_i - 2|b_i| > 0 \), and the eigenvectors are \( u_{ii} = \left(\sin\left(\frac{2\pi q\xi}{L_x}\right), \sin\left(\frac{2\pi q\xi}{L_x}\right), \ldots, \sin\left(\frac{2\pi q\xi}{L_x}\right)\right)^T \in \mathbb{R}^n \) \((q = 1, 2, \ldots, n)\). Therefore, the inversion result \( x \) is also obtained by using the discrete sine transform (DST) of Type-I [35].

When \( \text{mod}_{L_w}(\frac{L_w}{2}) = 0 \) and \( \text{mod}_{2\ell}(\frac{L_w}{2}) \neq 0 \) for Type-III, we have \( b_{ii} = (1)^j b_i \) for all \( i \) and \( j \). We can also compute the inversion result \( x \) by using Type-I DST, even in this case, with appropriate sign reversal process (see the actual program in [35] for more detail).

4.2. Inversion Based on the Pseudoinverse with the Periodicity

For Type-I and Type-II, let \( p \) be the minimum nonnegative integer s.t. \( \text{mod}_{L_w} \left( \frac{L_w + L_x - 2q\pi}{\pi} \right) = p \). For Type-III, \( p \) must also satisfy \( \text{mod}_{L_w} \left( \frac{L_w + L_x - 2q\pi}{\pi} \right) + |p| = 0 \). We define \( L_p := \left\lfloor \frac{L_w + L_x - 2q\pi}{\pi} \right\rfloor + p \xi \) and \( x_p := (x[0], x[1], \ldots, x[L_x - 1], x[L_x], \ldots, x[L_p - 1])^T := (x^T, 0_{L_x, L_p - L_x})^T \in \mathbb{C}^{L_p} \). Let \( \mathcal{S}_p : \mathbb{C}^{L_p} \rightarrow \mathbb{C}^{L_w \times \ell} \times \left\lfloor \frac{L_w + L_x - 2q\pi}{\pi} \right\rfloor + |p| \) be a linear mapping, that computes one of (11), (12), and (13) for \( L = 0, 1, \ldots, \left\lfloor \frac{L_w + L_x - 2q\pi}{\pi} \right\rfloor + |p| \) while assuming the periodic condition, i.e., \( x[\tau] = x[L_p + \tau] \) for \( \tau < 0 \), according to the convention [6–8].

Let \( L_P \in \mathbb{R}^{L_x \times L_w} \), \( W = \text{diag}(w[\tau]) \in \mathbb{R}^{L_x \times L_w} \), and \( F \in \mathbb{C}^{L_w \times L_w} \) be the frame extraction matrix, window matrix, and Type-I undersampled DFT matrix. We have \( \mathcal{S}^\dagger \circ \mathcal{S} = \sum_{j=0}^{P-1} W F_j^H F_j W P \) for Type-I FUSTFT.

9 The solver for tridiagonal systems is called the Thomas algorithm.
For a given complex spectrogram $X \in \mathbb{C}^{\frac{Lw}{2}\times \left[\frac{Lw+1}{2}-\frac{Lw}{2}\right]}$, we define $X_t := [X, O_{\frac{Lw}{2}}] \in \mathbb{C}^{\frac{Lw}{2}\times \left[\frac{Lw+1}{2}-\frac{Lw}{2}\right]+1}$ and compute the inversion result $x_t = (S^H_t \circ S_p)^{-1} \circ S^H_t (X_t)$ of $X_t$ under the periodic condition. Then, by extracting the first $Lw$ components of $x_t$, we obtain the final inversion result $x$ of $X$. Therefore, define $y_t := (y[0], y[1], \ldots, y[Lw-1])^T := (S^H_t (X_t))$, and we only have to compute the unique solution $x_t$ to a linear system $(S^H_t \circ S_p)x_t = y_t$. This is decomposed into $\frac{Lw}{2}$ independent systems of the same size:

$$
\begin{align*}
\begin{bmatrix}
(a_i)^{0}\cdots (a_i)^{n-1} \\
(b_i)^{0}\cdots (b_i)^{n-1} \\
\vdots \\
(b_i)^{0}\cdots (b_i)^{n-1}
\end{bmatrix}
\begin{bmatrix}
x[i] \\
x[i + \frac{Lw}{2}] \\
\vdots \\
x[i + \frac{Lw}{2} - 1]
\end{bmatrix} =
\begin{bmatrix}
y[i] \\
y[i + \frac{Lw}{2}] \\
\vdots \\
y[i + \frac{Lw}{2} - 1]
\end{bmatrix}
\end{align*}
$$

(20)

where $n = n_1 = \cdots = n = \frac{Lw}{2} - 1$, and $(a_i)^{0} = (a_i)^{1} = \cdots = (a_i)^{n-1} = a_i$ and $(b_i)^{0} = (b_i)^{1} = \cdots = (b_i)^{n-1} = b_i$, and the matrices in the left side of (20) are periodic tridiagonal matrices, whose LU decompositions are also given in $O(n)$ [37], hence $x_t$ and the final result $x$ are also quickly obtained.

In particular, when $\text{mod}_{\frac{Lw}{2}}(\frac{Lw}{2}) = 0$ for Type-I and Type-II, or $\text{mod}_{\frac{Lw}{2}}(\frac{Lw}{2}) = 1$ for Type-III, we have $(a_i)^{0} = (a_i)^{1} = \cdots = (a_i)^{n-1} = a_i$ and $(b_i)^{0} = (b_i)^{1} = \cdots = (b_i)^{n-1} = b_i$, and hence the matrices in the left side of (20) are symmetric circular matrices. In such cases, the eigenvalues are $\lambda_j^{(i)} = a_i + 2b_i \cos(\frac{j \pi}{Lw}) \geq 2|b_i| > 0$ and the eigenvectors are $u_j = (1, e^{-i\frac{2\pi j}{Lw}}, e^{-i\frac{4\pi j}{Lw}}, \ldots, e^{-i(2(n-1)\frac{2\pi j}{Lw})})^T \in \mathbb{C}^{n-1}$. Note that Stanekovic defined the discrete-time window function as $w[\tau] := w(\tau T_u)$ in [30] which results in $|\hat{w}[\ell]| = \frac{1}{Lw}$ and $\lambda^{(i)} = 0$ for $\ell = \frac{Lw}{2}$ when $\xi = \frac{Lw}{2}$. On the other hand, we defined it as $w[\tau] := w(\tau + \frac{1}{2} T_u)$ in this paper, which guarantees $|\hat{w}[\ell]| < \frac{1}{Lw}$ and $\lambda^{(i)} > 0$ for all $i$. Hence, the inversion result $x_t$ is also obtained by using FFT, but we have to note that if $\xi = \frac{Lw}{2}$ and $Lw$ is relatively large, then $S^H_t \circ S_p$ becomes ill-conditioned.

When $\text{mod}_{\frac{Lw}{2}}(\frac{Lw}{2}) = 0$ and $\text{mod}_{\frac{Lw}{2}}(\frac{Lw}{2}) \neq 0$ for Type-III, we have $(b_i)^{(i)} = (-1)^{i}b_i$ for all $i$. We can also compute the inversion result $x_t$ by using FFT, even in this case, with appropriate sign reversal process or appropriate multiplication process by $i$ dependently on $\text{mod}_{n}(n)$ (see the actual program in [36] for more detail).

4.3. Difference between the Two Inversions for FUSTFT

Many papers explain the discrete STFT under the periodic condition as shown in Sect. 4.2. In the cases of the discrete STFT and FOSTFT, actually both $S^H \circ S$ and $S^H \circ S_p$ are diagonal matrices whose diagonal components are periodic, and the two inversion results are always the same. As a result, there is almost no problem even if we explain the discrete STFT and FOSTFT without the periodic condition.

On the other hand, in the case of FUSTFT, these two inversions have different properties. Define $\mathcal{R} := \{X \in \mathbb{C}^{\frac{Lw}{2}\times \left[\frac{Lw+1}{2}-\frac{Lw}{2}\right]} | \exists \xi \in \mathbb{C}^{Lw} \exists \xi = \{\xi(\beta)\} \text{ as the range of FUSTFT } S\}$. For a complex spectrogram $X \in \mathbb{R}$, both inversions can recover $x$ s.t. $X = S(x)$. For $X \not\in \mathbb{R}$, the standard inversion $(S^H \circ S_p)^{-1} \circ S^H (X)$ can recover the most consistent $x$ with $X$, while the inversion $(S^H \circ S_p)^{-1} \circ S^H (X)$ in the periodic condition can recover the most consistent $x$ with $X$, which means that the last $Lw - L$ components of $x_p$ become non-zero. It might seem that the former inversion $(S^H \circ S_p)^{-1} \circ S^H (X)$ should always be used, but the latter inversion $(S^H \circ S_p)^{-1} \circ S^H (X)$ has a special property in the case of $\xi = \frac{Lw}{2}$. Only when $\xi = \frac{Lw}{2}$, $S_p$ becomes a non-redundant transform, and $x_p$ s.t. $X_p = S_p (x_p)$ can be always recovered by the latter inversion. As a result, for any complex spectrogram $X \not\in \mathbb{R}$, a discrete-time signal that guarantees the perfect consistency other than the first and last frames is always recovered.

We confirm the properties of the inversions in Sects. 4.1 and 4.2 for Type-II FUSTFT. A sound signal $x$ of 15 seconds is transformed into a complex spectrogram $X \in \mathbb{R}$ and we create a noisy version $X \not\in \mathbb{R}$ by adding complex white Gaussian noise of variance $10^{-6}$ to $X$. The performance of the inversion results $\hat{x}$ from $X$ and $\hat{X}$ are summarized in Tables 1 and 2, respectively. From these tables, we can confirm that both inversions work correctly. In particular, when $\xi = \frac{Lw}{2}$, the inversion results by $(S^H \circ S_p)^{-1} \circ S^H (X)$ demonstrated the slight numerical instability in blue letters of Table 1 and the perfect consistency other than the first and last frames in red letters of Table 2.

5. CONCLUSION

This paper proposed FUSTFT and its two inversions. FUSTFT gives efficient spectrogram matrices by computing only half the frequency components of the discrete STFT. By using FUSTFT, it is expected that window functions of relatively large $Lw$ such as the Kaiser window and the truncated Gaussian window will be easier to use. Since we can arbitrarily modify magnitudes and phases other than the first and last frames when $\xi = \frac{Lw}{2}$, further development of spectrogram-based techniques is also expected. There is also a possibility that flexible transforms such as Type-III FUSTFT will find new applications.

10There are many cases that are the inversions for MDCT and DWT have.

12We used a male voice, that counts numbers, of $\ell = 44,100$ Hz [32] and the normalized Hann window $w[\tau] := \frac{1}{Lw} (1 - \cos(\frac{\pi \ell}{Lw} \cdot (\tau + \frac{1}{2})))$.

13In Tables 1 and 2, we call $\| \cdot \|^0$ the interior Frobenius norm that ignores components in the first $\left\lfloor \frac{Lw}{2}\right\rfloor$ and last $\left\lceil \frac{Lw+1}{2}-\frac{Lw}{2}\right\rceil$ time frames.
6. REFERENCES

[1] M. R. Portnoff, “Implementation of the digital phase vocoder using the fast Fourier transform,” *IEEE Trans. Acoust. Speech Signal Process.*, vol. 24, no. 3, pp. 243–248, 1976.

[2] J. B. Allen, “Short term spectral analysis, synthesis, and modification by discrete Fourier transform,” *IEEE Trans. Acoust. Speech Signal Process.*, vol. 25, no. 3, pp. 235–238, 1977.

[3] J. B. Allen and L. R. Rabiner, “A unified approach to short-time Fourier analysis and synthesis,” *Proc. IEEE*, vol. 65, no. 11, pp. 1558–1564, 1977.

[4] R. E. Crochier, “A weighted overlap-add method of short-time Fourier analysis/synthesis,” *IEEE Trans. Acoust. Speech Signal Process.*, vol. 28, no. 1, pp. 99–102, 1980.

[5] L. Cohen, *Time-Frequency Analysis: Theory and Applications*. Englewood Cliffs, NJ: Prentice Hall, 1994.

[6] H. G. Feichtinger and T. Strohmer, Eds., *Gabor Analysis and Algorithms: Theory and Applications*. Secaucus, NJ: Birkhäuser, 1997.

[7] K. Gröchenig, *Foundations of Time-Frequency Analysis*. Secaucus, NJ: Birkhäuser, 2001.

[8] H. G. Feichtinger and T. Strohmer, Eds., *Advances in Gabor Analysis*. Secaucus, NJ: Birkhäuser, 2002.

[9] P. L. Søndergaard, “Finite discrete Gabor analysis,” Ph.D. thesis, Technical University of Denmark, 2007.

[10] K. Yatabe, Y. Masuyama, T. Kusano, and Y. Oikawa, “Representation of complex spectrogram via phase conversion,” *Acoust. Sci. & Tech.*, vol. 40, no. 3, pp. 170–177, 2019.

[11] H. S. Malvar, “Lapped transforms for efficient transform/subband coding,” *IEEE Trans. Acoust. Speech Signal Process.*, vol. 38, no. 6, pp. 969–978, 1990.

[12] I. Daubechies, S. Jaffard and J. Journé, “A simple Wilson orthonormal basis with exponential decay,” *SIAM J. Math. Anal.*, vol. 22, no. 2, pp. 554–573, 1991.

[13] H. Bölcskei, G. Feichtinger, K. Gröchenig, and F. Hlawatsch, “Discrete-time Wilson expansions,” in *Proc. TFFTS*, Paris, 1996, pp. 525–528.

[14] H. S. Malvar, “A modulated complex lapped transform and its applications to audio processing,” in *Proc. ICASSP*, Phoenix, AZ, 1999, pp. 1421–1424.

[15] J. C. Brown, “Calculation of a constant-Q spectral transform,” *J. Acoust. Soc. Am.*, vol. 89, no. 1, pp. 425–434, 1991.

[16] I. Daubechies, *Ten Lectures on Wavelets*, ser. CBMS-NSF Regional Conference Series in Applied Mathematics. Philadelphia, PA: SIAM, 1992, vol. 61.

[17] J. J. Benedetto and M. W. Frazier, *Wavelets: Mathematics and Applications*, ser. Studies in Advanced Mathematics. Boca Raton, FL: CRC Press, 1993, vol. 13.

[18] G. Kaiser, *A Friendly Guide to Wavelets*. Secaucus, NJ: Birkhäuser, 2011.

[19] V. Verfaille, U. Zolzer, and D. Arfib, “Adaptive digital audio effects (A-DAFs): A new class of sound transformations,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 5, pp. 1817–1831, 2006.

[20] G. Yu, S. Mallat, and E. Bacry, “Audio denoising by time-frequency block thresholding,” *IEEE Trans. on Signal Process.*, vol. 56, no. 5, pp. 1830–1839, 2008.

[21] E. Benetos, S. Dixon, D. Giannoulis, H. Kirchhoff, and A. Klapuri, “Automatic music transcription: challenges and future directions,” *J. Intel. Inf. Syst.*, vol. 41, no. 3, pp. 407–434, 2013.

[22] M. Mauch, C. Cannam, R. Bittner, G. Fazekas, J. Salamon, J. Dai, J. Bello, and S. Dixon, “Computer-aided melody note transcription using the Tony software: accuracy and efficiency,” in *Proc. TENOR*, Paris, 2015, pp. 28–30.

[23] T. Fujiwara, M. Yamagishi, and I. Yamada, “Reduced-rank modeling of time-varying spectral patterns for supervised source separation,” in *Proc. ICASSP*, Brisbane, 2015, pp. 3307–3311.

[24] Y. Wakabayashi and N. Ono, “Griffin–Lim phase reconstruction using short-time Fourier transform with zero-padded frame analysis,” in *Proc. APSIPA ASC*, Lanzhou, 2019, pp. 1863–1867.

[25] Y. Takahashi, D. Kitahara, K. Matsuura, and A. Hirabayashi, “Determined source separation using the sparsity of impulse responses,” in *Proc. ICASSP*, Barcelona, 2020, pp. 686–690.

[26] R. Nakatsu, D. Kitahara, and A. Hirabayashi, “Non-Griffin–Lim type signal recovery from magnitude spectrogram,” in *Proc. ICASSP*, Barcelona, 2020, pp. 791–795.

[27] T. Werther, Y. C. Eldar, and N. K. Subbanna, “Dual Gabor frames: theory and computational aspects,” *IEEE Trans. Signal Process.*, vol. 53, no. 11, pp. 4147–4158, 2005.

[28] B. Yang, “A study of inverse short-time Fourier transform,” in *Proc. ICASSP*, Las Vegas, NV, 2008, pp. 3541–3544.

[29] I. Daubechies, A. Grossmann, and Y. Meyer, “Painless nonorthogonal expansions,” *J. Math. Phys.*, vol. 27, no. 5, pp. 1271–1283, 1986.

[30] L. Stanković, “On the STFT inversion redundancy,” *IEEE Trans. Circuits Syst. II: Express Briefs*, vol. 63, no. 3, pp. 284–288, 2016.

[31] S. Moreno-Picot, F. J. Ferri, M. Arevalillo-Herráez, and W. Díaz-Villanueva, “Efficient analysis and synthesis using a new factorization of the Gabor frame matrix,” *IEEE Trans. Signal Process.*, vol. 66, no. 17, pp. 4564–4573, 2018.

[32] MathWorks, “Modified discrete cosine transform: MATLAB mdct.” https://www.mathworks.com/help/audio/ref/mdct.html

[33] L. H. Thomas, “Elliptic problems in linear differential equations over a network,” *Watson Sci. Comput. Lab. Report*, Columbia Univ., 1949.

[34] S. Noschese, L. Pasquini, and L. Reichel, “Triangular Toeplitz matrices: properties and novel applications,” *Numer. Linear Algebra Appl.*, vol. 20, no. 2, pp. 302–326, 2013.

[35] S. A. Martucci, “Symmetric convolution and the discrete sine and cosine transforms,” *IEEE Trans. Signal Process.*, vol. 42, no. 5, pp. 1038–1051, 1994.

[36] D. Kitahara, The MATLAB program of FUSTFT, https://lab.d-kitahara.com/codes/

[37] M. M. Chawla and R. R. Khazal, “A parallel elimination method for “periodic” tridiagonal systems,” *Int. J. Comput. Math.*, vol. 79, no. 4, pp. 473–484, 2002.