SIAM J. SCI. COMPUT.
Vol. 31, No. 5, pp. 4000–4012

FAST DISCRETE ORTHONORMAL STOCKWELL TRANSFORM*

YANWEI WANG† AND JEFF ORCHARD‡

Abstract. We present an efficient method for computing the discrete orthonormal Stockwell transform (DOST). The Stockwell transform (ST) is a time-frequency decomposition transform that is showing great promise in various applications, but is limited because its computation is infeasible for most applications. The DOST is a nonredundant version of the ST, solving many of the memory and computational issues. However, computing the DOST of a signal of length N using basis vectors is still $\mathcal{O}(N^2)$. The computational complexity of our method is $\mathcal{O}(N \log N)$, putting it in the same category as the FFT. The algorithm is based on a simple decomposition of the DOST matrix. We also explore the way to gain conjugate symmetry for the DOST and propose a variation of the parameters that exhibits symmetry, akin to the conjugate symmetry of the FFT of a real-valued signal. Our fast method works equally well on this symmetric DOST. In this paper, we provide a mathematical proof of our results and derive that the computational complexity of our algorithm is $\mathcal{O}(N \log N)$. Timing tests also confirm that the new method is orders-of-magnitude faster than the brute-force DOST, and they demonstrate that our fast DOST is indeed $\mathcal{O}(N \log N)$ in complexity.

Key words. Stockwell transform, discrete orthonormal Stockwell transform, time-frequency representation, multiresolution, computational complexity

AMS subject classifications. 03D15, 42C05, 65T60, 68Q15

DOI. 10.1137/080737113

1. Introduction.

In signal and image processing, the Fourier transform (FT) is commonly used to decompose a signal into its frequency components. But the global property of the FT—that each sample affects every Fourier coefficient (and vice versa)—makes it unfavorable in applications where local information is preferred (e.g., signal denoising, compression, phase analysis). The wavelet transform [4] addresses this issue by applying local decomposition filters to a signal on multiple scales. However, the self-similarity constraint among the wavelet basis functions destroys the phase information, so the coefficients supply only locally referenced phase information. Furthermore, even though the term "scale" can be approximately interpreted as "frequency," there is no straightforward way to transform the scale information into proper frequency information.

The Stockwell transform (ST, sometimes called the S-transform) [6, 8, 9, 16] is a time-frequency decomposition that offers absolutely referenced phase information (i.e., the phase information is referenced to time $t = 0$). Hence, the accumulation (sum) of the coefficients for a fixed frequency yields the exact Fourier coefficient for that frequency. However, a huge amount of time and storage are needed for a signal of moderate size because the ST is highly redundant. For a signal of length N, the discrete ST generates N^2 coefficients. To combat this redundancy, the time-frequency domain can be partitioned into N regions, and each region is represented by one coefficient; this is the strategy adopted by the discrete orthonormal Stockwell transform (DOST) [14], making the DOST more convenient than other time-frequency trans-

*Received by the editors October 3, 2008; accepted for publication (in revised form) June 22, 2009; published electronically November 11, 2009.

http://www.siam.org/journals/sisc/31-5/73711.html

†Department of Applied Mathematics, 200 University Avenue West, University of Waterloo, Waterloo, Ontario N2L 3G1, Canada (y72wang@math.uwaterloo.ca).

‡David R. Cheriton School of Computer Science, 200 University Avenue West, University of Waterloo, Waterloo, Ontario N2L 3G1, Canada (jorchard@cs.uwaterloo.ca).

forms, such as the Gabor transform [13]. The DOST coefficients can be computed by taking the vector dot-product of the input signal with a set of N basis vectors, which makes computing the DOST (and its inverse) $\mathcal{O}(N^2)$. While this is an improvement over the full ST, the DOST is still cumbersome for applications that have large signals, such as audio processing, remote sensing, and medical imaging. A fast algorithm to compute the DOST is presented in this paper as the fast DOST (FDOST).

When ones applies the FT to a real-valued input signal, the resulting Fourier coefficients exhibit conjugate symmetry. This property is advantageous for many applications. For instance, when manipulating the transform coefficients (e.g., for filtering), it is trivial to adjust them in such as way that the resulting signal is still real-valued. The original derivation of the DOST [14] has offered the possibility to fulfill this requirement of symmetry. In the later part of this paper, suitable positive- and negative-frequency indexes and parameters have been given explicitly to show the conjugate symmetry of the DOST. We also present a fully symmetric version of the DOST [18] by adjusting the index parameters. The fast computation method also works nicely on this fully symmetric DOST.

2. Review of the ST and the DOST.

2. Review of the ST and the DOST. The ST [16, 9, 8, 6] gives a full time-frequency decomposition of a signal. Consider a one-dimensional (1-D) function $h(t)$. The ST of $h(t)$ is defined as the FT of the product between $h(t)$ and a Gaussian window function

$$S(\tau, f) = \int_{-\infty}^{\infty} h(t) \frac{|f|}{\sqrt{2\pi}} e^{-\frac{(\tau-t)^2 f^2}{2}} e^{-i2\pi f t} dt, \tag{2.1}$$

where f is the frequency and t and τ are time variables. The ST decomposes a signal into temporal (τ) and frequency (f) components.

By the integral properties of the Gaussian function, the relation between $S(\tau, f)$ and $H(f)$ (the FT of $h(t)$) is

$$\int_{-\infty}^{\infty} S(\tau, f) d\tau = H(f). \tag{2.2}$$

That is, the accumulation of the Stockwell coefficients over the time axis yields the FT of the signal, highlighting a special feature of the ST. Hence, the original function $h(t)$ can be recovered by calculating the inverse FT of $H(f)$

$$h(t) = \int_{-\infty}^{\infty} \left\{ \int_{-\infty}^{\infty} S(\tau, f) d\tau \right\} e^{i2\pi f t} df. \tag{2.3}$$

Using the equivalent frequency-domain definition of the ST, the discrete ST (DST) [16] can be written

$$S[j, n] = \sum_{m=0}^{N-1} H[m+n] e^{-2\pi^2 m^2/n^2} e^{i2\pi m j/N} \tag{2.4}$$

for $n \neq 0$, where $H[\cdot]$ is the DFT of $h[\cdot]$. For the $n = 0$ voice, define

$$S[j, 0] = \frac{1}{N} \sum_{m=0}^{N-1} h[m], \tag{2.5}$$

analogous to the DC value of the FT. The DST has been used in various fields. For example, in geophysics it is used for analyzing internal atmospheric wave packets [15], atmospheric studies [11], characterization of seismic signals, and global sea surface temperature analysis [8]. It is used in electrical engineering [3], mechanical engineering [10], and digital signal processing [12]. It has also been applied in the medical field for human brain mapping [1], cardiovascular studies [17], magnetic resonance imaging (MRI) analysis [7], and to study the physiological effects of drugs [2].

From (2.4), it is obvious that the ST is an overcomplete representation. For a signal of length N, there are N^2 Stockwell coefficients, and each one takes $\mathcal{O}(N)$ to compute. Hence, computing all N^2 coefficients of the ST has computational complexity $\mathcal{O}(N^3)$. The ST gets exponentially more expensive for higher-dimensional data. A more efficient mathematical and computational framework is needed to pursue this time-frequency decomposition.

The DOST is a pared-down version of the fully redundant ST [14]. Since lower frequencies have longer periods, it stands to reason that lower frequencies can cope with lower sampling rates. Hence, the DOST subsamples the low frequencies. Similarly, high frequencies have higher sampling rates. The DOST takes advantage of this sample spacing paradigm and distributes its coefficients accordingly. It does so by constructing a set of N orthogonal unit-length basis vectors, each of which targets a particular region in the time-frequency domain. The regions are described by a set of parameters: ν specifies the center of each frequency band (voice), β is the width of that band, and τ specifies the location in time. Using these parameters, the kth basis vector is defined as

$$D[k]_{[\nu,\beta,\tau]} = \frac{1}{\sqrt{\beta}} \sum_{f=\nu-\beta/2}^{\nu+\beta/2-1} \exp\left(-i2\pi \frac{k}{N} f\right) \exp\left(i2\pi \frac{\tau}{\beta} f\right) \exp\left(-i\pi\tau\right) \tag{2.6}$$

for $k = 0, \ldots, N-1$, which can be summed analytically to

$$D[k]_{[\nu,\beta,\tau]} = ie^{-i\pi\tau} \frac{e^{-i2\alpha(\nu-\beta/2-1/2)} - e^{-i2\alpha(\nu+\beta/2-1/2)}}{2\sqrt{\beta}\sin\alpha}, \tag{2.7}$$

where $\alpha = \pi(k/N - \tau/\beta)$ is the center of the temporal window. For the singular point on $\alpha = 0$, we can either directly get the value of $D[k]$ from (2.6), which is

$$D[k]_{[\alpha=0]} = -\sqrt{\beta} ie^{-i\pi\tau}, \tag{2.8}$$

or pursue a calculation of the limit using some Calculus techniques. So the continuity of the basis functions will follow.

To make the family of basis vectors in (2.7) orthogonal, the parameters ν, β, and τ have to be chosen suitably. Letting the variable p index the frequency bands, Stockwell defines the DOST basis vectors of the positive frequency for each p on page 5 in [14], using

- if $p = 0$, $D[k]_{[\nu,\beta,\tau]} = 1$ (only one basis vector);
- if $p = 1$, $D[k]_{[\nu,\beta,\tau]} = \exp(-i2k\pi/N)$ (only one basis vector);

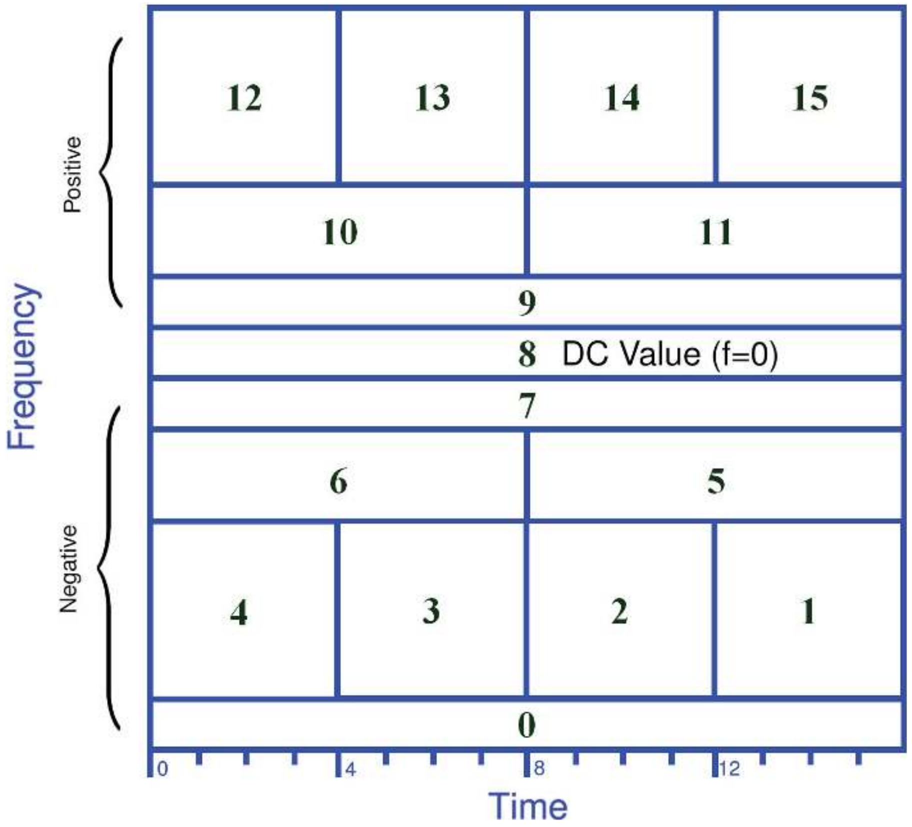


Fig. 2.1. *The order of the* 2*-D DOST coefficients into a* 1*-D* N*-vector.*

- for $p = 2, 3 \ldots, \log_2 N - 1$, pick
  (2.9)
  $$\left.\begin{array}{l} \nu = 2^{(p-1)} + 2^{(p-2)} \\ \beta = 2^{(p-1)} \\ \tau = 0, \ldots, \beta - 1 \end{array}\right\} \text{ defines } 2^{p-1} \text{ basis vectors for each frequency band.}$$

Combining these basis vectors with the basis vectors for the negative frequencies (described in the next section), we can prove that these parameter choices generate a basis of N orthogonal unit vectors; hence N DOST coefficients. For real applications, it is helpful to order these N coefficients into a 1-D vector. The ordering we use is shown in Figure 2.1 for a signal of length 16 (see Figure 4.1 for more details). By convention, our time index (τ) traverses the time axis in the negative direction for negative frequencies. Doing so creates a symmetric correspondence between the positive- and negative-frequency coefficients in the 1-D representation. That is, for a given coefficient with index i in the 1-D DOST vector, its negative-frequency analogue is at index $N - i$. This indexing convention will help later to gain symmetry of the DOST.

Figure 2.2 shows the logarithm of the magnitude of the two-dimensional (2-D) DOST coefficients for one of the most popular example images, Lena. The ST and the DOST are both separable transforms, so they can be applied in higher dimensions trivially. As we can see, the coefficients decay very quickly, which makes the DOST a powerful tool for image compression and other applications.

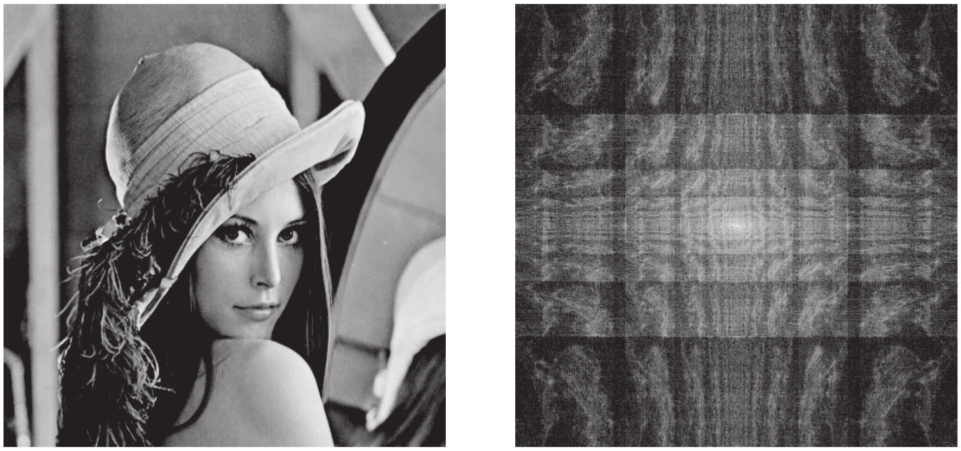


FIG. 2.2. *Lena and the logarithm of its DOST coefficients.*

3. Conjugate symmetry on the DOST. If we pick the parameters (ν, β, and τ) suitably, a real-valued input signal yields a set of conjugate symmetric DOST coefficients.

More explicitly, if we use the negative integers p to index the negative-frequency bands and let $q = -p$, then we can choose the parameters, using

- if $q = 1$, $D[k]_{[\nu,\beta,\tau]} = \exp(i2k\pi/N)$ (only one basis vector);
- for $q = 2, 3 \ldots, \log_2 N - 1$, pick

$$\left.\begin{aligned} &\nu = -2^{(q-1)} - 2^{(q-2)} + 1 \\ &\beta = 2^{(q-1)} \\ &\tau = 0, \ldots, \beta - 1 \end{aligned}\right\} \quad 2^{q-1} \text{ basis vectors for each band;} \tag{3.1}$$

- if $q = \log_2 N$, $D[k]_{[\nu,\beta,\tau]} = e^{-ik\pi}$ (only one basis vector).

THEOREM 3.1. *The DOST basis functions under the parameters ν, β, and τ according to the rules of* (2.9) *and* (3.1) *form an orthonormal basis of an N-dimensional vector space. Moreover, for a real-valued input signal, the DOST coefficients are conjugate symmetric about the DC value* ($p = 0$).

Proof. The orthogonality has been implied by the definition (2.6). We will focus on the conjugate symmetry here.

For an arbitrarily given band index $|p|$ ($p \neq 0, \log_2 N$), we have two groups of basic functions: one group corresponding to the positive frequencies, and the other group corresponding to the negative frequencies. Notice here the values of β are the same, and the values of τ have the same range $0, 1, \ldots, \beta - 1$. We will distinguish the positive-frequency parameters from the negative-frequency parameters using a superscripted positive sign or negative sign. Then

$$\begin{aligned} D[k]^*_{[\nu^+,\beta,\tau]} &= \left(ie^{-i\pi\tau} \frac{e^{-i2\pi(k/N-\tau/\beta)(\nu^+-\beta/2-1/2)} - e^{-i2\pi(k/N-\tau/\beta)(\nu^++\beta/2-1/2)}}{2\sqrt{\beta}\sin\pi(k/N-\tau/\beta)} \right)^* \\ &= -ie^{i\pi\tau} \frac{e^{i2\pi(k/N-\tau/\beta)(\nu^+-\beta/2-1/2)} - e^{i2\pi(k/N-\tau/\beta)(\nu^++\beta/2-1/2)}}{2\sqrt{\beta}\sin\pi(k/N-\tau/\beta)}, \end{aligned} \tag{3.2}$$

where * denotes complex conjugation. Note that, for the corresponding negative index

$-p$, we have $\nu^+ = -(\nu^- - 1)$. Thus, (3.2) can be written

$$\begin{aligned} D[k]^*_{[\nu^+,\beta,\tau]} &= -ie^{i\pi\tau}\frac{e^{i2\pi(k/N-\tau/\beta)(-\nu^-+1-\beta/2-1/2)} - e^{i2\pi(k/N-\tau/\beta)(-\nu^-+1+\beta/2-1/2)}}{2\sqrt{\beta}\sin\pi(k/N-\tau/\beta)} \\ &= ie^{i\pi\tau}\frac{e^{-i2\pi(k/N-\tau/\beta)(\nu^--\beta/2-1/2)} - e^{-i2\pi(k/N-\tau/\beta)(\nu^-+\beta/2-1/2)}}{2\sqrt{\beta}\sin\pi(k/N-\tau/\beta)}, \qquad (3.3) \end{aligned}$$

where we have swapped the terms in the numerator in (3.3). Since τ is an integer, $e^{i\pi\tau}$ is always real. Thus $e^{i\pi\tau} = e^{-i\pi\tau}$. Making these substitutions, we can write (3.3) as

$$D[k]^*_{[\nu^+,\beta,\tau]} = ie^{-i\pi\tau}\frac{e^{-i2\pi(k/N-\tau/\beta)(\nu^--\beta/2-1/2)} - e^{-i2\pi(k/N-\tau/\beta)(\nu^-+\beta/2-1/2)}}{2\sqrt{\beta}\sin\pi(k/N-\tau/\beta)}, \tag{3.4}$$

which means, if the same τ values have been picked, the basis vectors for the positive-frequency band p are conjugate symmetric to the corresponding basis vectors for the negative-frequency band $-p$. Hence, the corresponding DOST coefficients will exhibit conjugate symmetry when the input is real-valued. □

4. Alternative symmetric DOST.

Figure 4.1(a) shows how the parameters ν, β, and τ partition the time-frequency domain and how the DOST gives conjugate symmetry when the input is real. Motivated by the Fourier shift theorem, we can modify the definitions of the parameters and define an alternative fully symmetric DOST.

In [18], we previously proposed a symmetric DOST where the basis vectors were altered so that the resulting coefficients were conjugate symmetric for real-valued input. The new basis is also orthonormal. Here, we repeat the derivation.

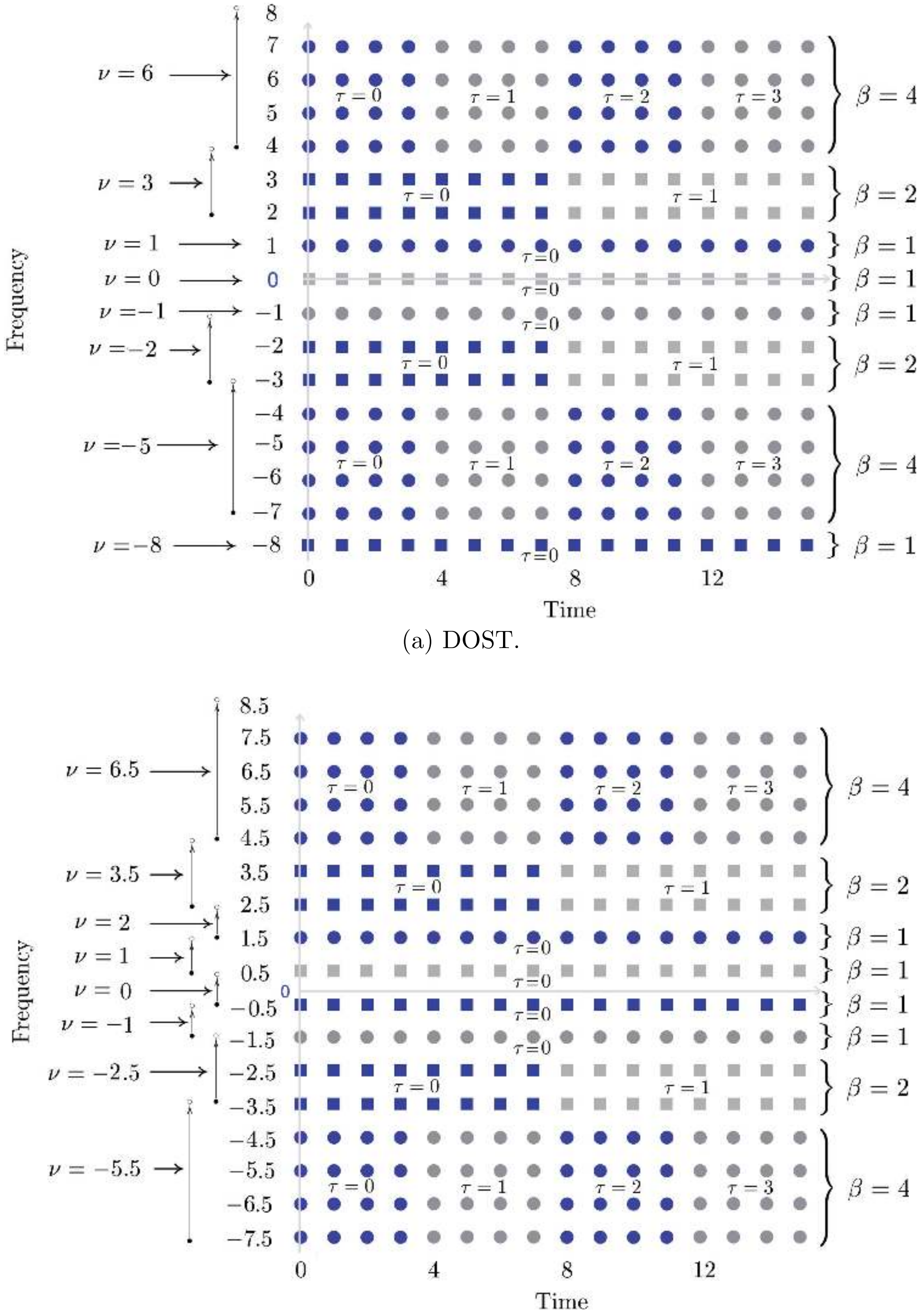
Imposing the conjugate symmetry requirement on the DOST coefficients [18] gives us

$$S_{[\nu,\beta,\tau]} = \left(S_{[-\nu,\beta,\tau]}\right)^*. \tag{4.1}$$

This symmetry constraint is satisfied for all nonzero ν if we simply shift all the samples away from the zero frequency by 1/2. Then we are left with a gap from -1 to 1 containing one coefficient whose band extends from $-1/2$ to $1/2$. Instead, we split the gap into two coefficients: one with a band from 0 to 1 and the other with a band from -1 to 0. This alternative partition can be implemented by simply replacing ν with $(\nu + 1/2)$ in (2.7) for the positive frequencies; a partition that is mirrored for the negative frequencies. Then, the basis vectors for this symmetric DOST, denoted $\tilde{D}$, can be written

$$\tilde{D}[k]_{[\nu,\beta,\tau]} = ie^{-i\pi\tau}\frac{e^{-i2\alpha(\nu-\beta/2)} - e^{-i2\alpha(\nu+\beta/2)}}{2\sqrt{\beta}\sin\alpha}. \tag{4.2}$$

The orthogonality property still holds for this family of basis vectors. Figure 4.1(b) shows the partition over the time-frequency domain of this symmetric DOST and how it differs from that of the original DOST.



(a) DOST.

(b) Alternative Symmetric DOST

FIG. 4.1. *Partition diagram of the time-frequency domain. Each rectangle block area with the same shape and color corresponds to one DOST coefficient. In* (b), *the partitions for the symmetric DOST have been shifted along the frequency axis according to the description in section* 4.

5. FDOST and fast symmetric DOST. We state above that the matrix-vector implementation of the DOST has computational complexity of $\mathcal{O}(N^2)$. However, the DOST can be calculated in a faster manner by taking advantage of the FFT. While this fact was mentioned in [14], we developed our method independently, and we supply a rigorous proof of its computational complexity class here.

Consider the inner product between $D[k]_{[\nu,\beta,\tau]}$, as shown in (2.6), and the input signal $h[k]$ (of length N). The resulting expression is the DOST coefficient S for the region corresponding to the choice of $[\nu, \beta, \tau]$ and can be expressed as

$$\begin{aligned} S_{[\nu,\beta,\tau]} &= \langle D[k]_{[\nu,\beta,\tau]}, h[k] \rangle \\ &= \frac{1}{\sqrt{\beta}} \sum_{k=0}^{N-1} \sum_{f=\nu-\beta/2}^{\nu+\beta/2-1} \exp\left(-i2\pi \frac{k}{N} f\right) \exp\left(i2\pi \frac{\tau}{\beta} f\right) \exp\left(-i\pi\tau\right) h[k]. \end{aligned} \tag{5.1}$$

In the above summation, the order of the sums can be switched and the common factors taken out. Then (5.1) becomes

$$\frac{1}{\sqrt{\beta}} \sum_{f=\nu-\beta/2}^{\nu+\beta/2-1} \exp\left(-i\pi\tau\right) \exp\left(i2\pi \frac{\tau}{\beta} f\right) \left[\sum_{k=0}^{N-1} \exp\left(-i2\pi \frac{k}{N} f\right) h[k] \right]. \tag{5.2}$$

The part in the square brackets is $H[f]$, the discrete Fourier coefficient of our signal, evaluated at the frequency index f. Hence, we have

$$S_{[\nu,\beta,\tau]} = \frac{1}{\sqrt{\beta}} \sum_{f=\nu-\beta/2}^{\nu+\beta/2-1} \exp\left(-i\pi\tau\right) \exp\left(i2\pi \frac{\tau}{\beta} f\right) H[f], \tag{5.3}$$

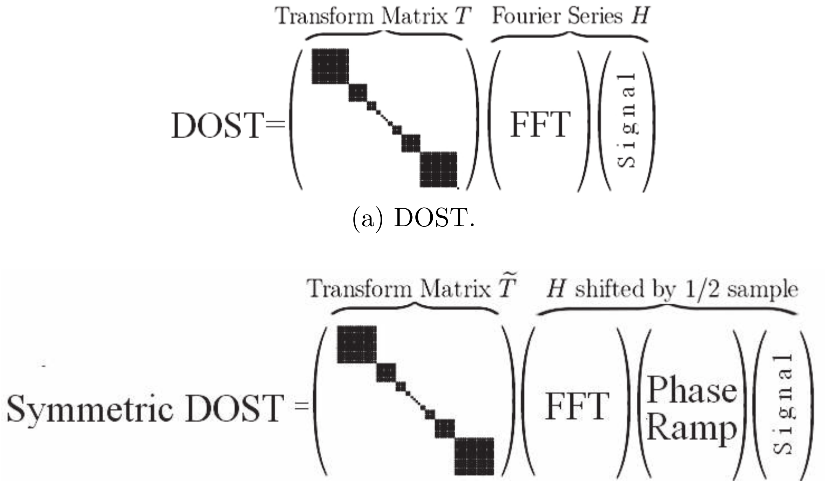
where the value of f is summed only on a certain band (depending on ν and β). Hence, this summation can be represented by the inner product between a row in a sparse matrix and the vector of the Fourier coefficients H.

This strategy can be summarized as in Figure 5.1(a). The block-diagonal nature of the transform matrix T offers the opportunity to calculate the DOST coefficients in a blockwise fashion. Hence, this sparse matrix allows for more efficient matrix multiplication.

The alternative symmetric DOST can be represented in a similar way (as shown in Figure 5.1(b)) by first multiplying the signal by a phase ramp. Despite the fact that the symmetric DOST corresponds to a 1/2-sample shift along the frequency axis, there is no loss of information due to resampling because the phase ramp that precedes the FFT implements the shift by the Fourier shift theorem. Note that the transform matrix is slightly different for the symmetric DOST. However, these transform matrices have essentially the same structure and are block-diagonal in both cases.

Not only is T sparse, but each block of T has a special structure that facilitates efficient matrix multiplication. To see this, consider the top-left block, labeled T_1, for an example where $N = 16$:

$$\frac{1}{\sqrt{\beta}} \begin{pmatrix} e^{-\pi i \tau_0} e^{2\pi i \frac{\tau_0}{\beta}(A)} & e^{-\pi i \tau_0} e^{2\pi i \frac{\tau_0}{\beta}(A+1)} & e^{-\pi i \tau_0} e^{2\pi i \frac{\tau_0}{\beta}(A+2)} & e^{-\pi i \tau_0} e^{2\pi \frac{\tau_0}{\beta}(A+3)} \\ e^{-\pi i \tau_1} e^{2\pi i \frac{\tau_1}{\beta}(A)} & e^{-\pi i \tau_1} e^{2\pi i \frac{\tau_1}{\beta}(A+1)} & e^{-\pi i \tau_1} e^{2\pi i \frac{\tau_1}{\beta}(A+2)} & e^{-\pi i \tau_1} e^{2\pi i \frac{\tau_1}{\beta}(A+3)} \\ e^{-\pi i \tau_2} e^{2\pi i \frac{\tau_2}{\beta}(A)} & e^{-\pi i \tau_2} e^{2\pi i \frac{\tau_2}{\beta}(A+1)} & e^{-\pi i \tau_2} e^{2\pi i \frac{\tau_2}{\beta}(A+2)} & e^{-\pi i \tau_2} e^{2\pi i \frac{\tau_2}{\beta}(A+3)} \\ e^{-\pi i \tau_3} e^{2\pi i \frac{\tau_3}{\beta}(A)} & e^{-\pi i \tau_3} e^{2\pi i \frac{\tau_3}{\beta}(A+1)} & e^{-\pi i \tau_3} e^{2\pi i \frac{\tau_3}{\beta}(A+2)} & e^{-\pi i \tau_3} e^{2\pi i \frac{\tau_3}{\beta}(A+3)} \end{pmatrix},$$



(a) DOST.

(b) Alternative Symmetric DOST

FIG. 5.1. *Calculation strategies of the DOST and the alternative symmetric DOST. The symmetric DOST is equivalent to the shifted version of the DOST with a different transform matrix.*

where we have replaced $(\nu-\beta/2)$ with A for notational simplicity. Noting that $\tau_k = k$, if we index the rows with k and the columns with j (where $j, k = 0, \ldots, \beta-1$), then the (j, k) element of T_1 is

$$\begin{aligned} \beta^{-\frac{1}{2}}\, e^{-\pi i\tau_k}\, e^{2\pi i\frac{\tau_k}{\beta}(A+j)} &= \beta^{-\frac{1}{2}}\, e^{-\pi i\tau_k\left(1-2\frac{A}{\beta}\right)}\, e^{2\pi i\frac{\tau_k}{\beta}j} \\ &= \beta^{-\frac{1}{2}}\, e^{-\pi ik\left(1-2\frac{A}{\beta}\right)}\, e^{2\pi i\frac{k}{\beta}j}. \end{aligned} \tag{5.4}$$

From (5.4), we can see that T_1 can be factored into a product of two matrices

$$T_1 = R_1 V_1, \tag{5.5}$$

where R_1 is a diagonal ramp matrix with entries $r_k = \beta^{-1/2}e^{-\pi ik(1-2A/\beta)}$ and V_1 is the inverse Fourier matrix (of size $\beta = 4$ in our example).

Therefore, the process of multiplying by T_1 can be broken into two parts: applying V_1 which takes $\mathcal{O}(\beta\log\beta)$, and applying R_1 which takes $\mathcal{O}(\beta)$. Accumulating the operation counts over all the blocks in T (i.e., for $\beta = N/4, N/8, \ldots, 1, \ldots, N/8, N/4, 1$), the complexity to modify the Fourier coefficients to get the DOST coefficients is $\mathcal{O}(N\log N)$. A formal and detailed proof of the computational complexity of this technique will be given in next section. Since the initial FFT in Figure 5.1(a) also has a complexity of $\mathcal{O}(N\log N)$, the total complexity for calculating the DOST coefficients is $\mathcal{O}(N\log N)$.

By studying the entry of the ramp matrix in our algorithm, it turns out (taking into consideration how the parameters have been chosen) that

$$r_k = e^{-2\pi i\frac{k}{\beta}(\beta-\nu)} = e^{-2\pi i\frac{k}{\beta}\frac{\beta}{2}}, \tag{5.6}$$

which means the slope is $\beta/2$ in the algorithm we presented here. According to the Fourier shift theorem, that slope is equivalent to a shift over the input sequence before the inverse fast Fourier transform (IFFT) is taken, which makes our algorithm

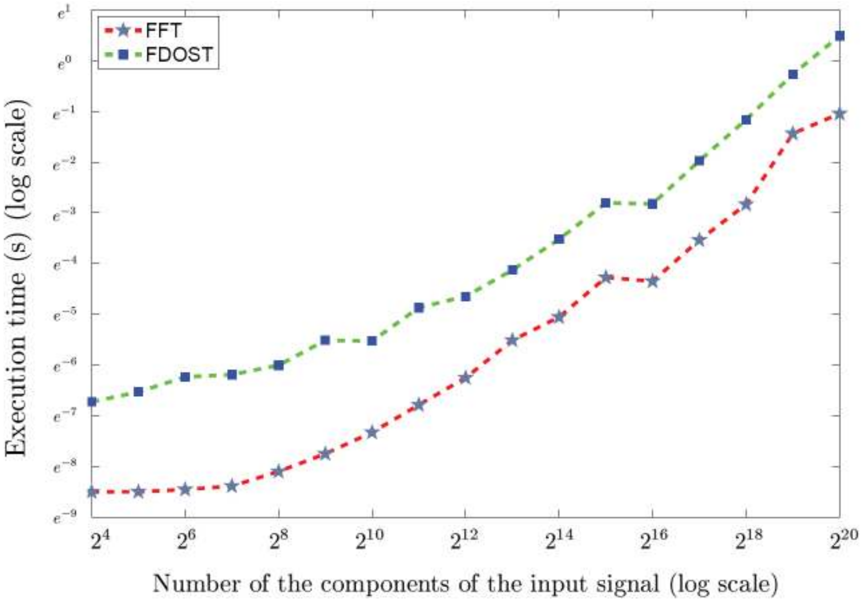


FIG. 5.2. *The comparison of time between the FDOST and FFT for various sizes of input signals.*

equivalent to the one described in [5], where the shift of $-N_y/2$ is taken before the IFFT.

Let us now consider the operation of reconstruction, the inverse DOST. All the blocks of T are unitary matrices, so T is a unitary matrix. Hence the inverse of T is the adjoint (conjugate transpose) of T. The adjoint of T has the same structure as T and can still be decomposed into a diagonal matrix and a Fourier matrix and therefore applied with computational complexity $\mathcal{O}(N \log N)$. The other matrix factors shown in Figure 5.1(a) are all trivially invertible and applied with the same computational complexity as the forward operators. Thus, the inverse DOST can also be computed in $\mathcal{O}(N \log N)$.

Moreover, during the decomposition and reconstruction, at no point does a matrix need to be explicitly stored. The FT matrices are implemented by the FFT, and the other matrices are all diagonal.

Besides the computational advantages, the matrix decomposition helps to elucidate the nature of the DOST decomposition. In the series of calculations to get the DOST, the input signal is transformed into pure frequency information first. Next, an inverse FT is applied to a narrow frequency band, yielding time-domain coefficients specific to that frequency band. Thus the final coefficients will carry both frequency and temporal information. This explanation is similar to the rationale given in [14] and [5].

Figure 5.2 plots the logarithm of the execution time for computing the FFT and FDOST. Both curves show the same growth trend, although the FDOST appears to be slower by a constant factor. Between 2^{14} and 2^{16}, both lines plateau slightly due to memory caching.

Since the FDOST method is in a different computational complexity class than the brute-force DOST computation (using vector dot-products), we did not embark on a formal study to compare the execution times between the two methods. However, we include here a realistic example to give an impression of the speed difference.

On a signal of length 1024, it took 2.285 seconds to compute the DOST using vector dot-products, but only 0.0086 seconds using our FDOST method. It is worth noting, however, that these timings were run in Matlab. Although every effort was made to implement the two methods on a "level playing field" (using Matlab's vectorization wherever possible), the timings ultimately depend on the particular Matlab implementation.

The alternative symmetric DOST has a slightly different transform matrix $\widetilde{T}$ as well as a different ramp matrix (e.g., R_1 in 5.5). However, both matrices have the same structure as their regular-DOST counterparts, so the symmetric FDOST algorithm also has complexity $\mathcal{O}(N \log N)$. Moreover, if the input signal is real-valued, the symmetry property allows one to compute only half of the coefficients.

6. Computational complexity.

THEOREM 6.1. *The computational complexity of the fast DOST and fast inverse DOST algorithms, as described in section* 5, *is* $\mathcal{O}(N \log N)$. *The fast algorithms for the alternative symmetric DOST are also* $\mathcal{O}(N \log N)$.

Proof. Assume we have an input series h of size N. As well known, the computational complexity of taking the FFT on h is $\mathcal{O}(N \log N)$. Assume that the actual number of floating-point operations of the FFT (and IFFT) algorithm is $\alpha N(\log N)$.

Without loss of generality, first assume $N = 2^n$, where n is a positive integer larger than three. The total accumulation of the DOST operations has been divided into two stages.

Stage 1. In this stage, we take the global FT using the FFT, i.e., the right-most matrix multiplication in Figure 5.1(a). The operation count for this stage is

$$S_1 = \alpha N \log N. \tag{6.1}$$

Stage 2. In this stage, we perform the blockwise matrix multiplication of the Fourier coefficients (from stage 1) with T, i.e., the matrix multiplication on the left in Figure 5.1(a).

Based on the partition strategy, in the left-most matrix of Figure 5.1(a) we have a series of matrices of size $\{2^{n-2}, 2^{n-3}, \ldots, 2, 1, 1, 2^{n-3}, 2^{n-2}, 1\}$. Recall from (5.5) that the matrix block can be factored into a diagonal matrix (R) and a Fourier matrix (V). For a block of size 2^m, the number of floating-point operations required to perform the IFFT and diagonal matrix multiplication is

$$\alpha 2^m \log 2^m + 2^m = \alpha m 2^m \log 2 + 2^m. \tag{6.2}$$

So the total operations needed in this stage will be

$$\begin{aligned} S_2 &= 2\sum_{m=0}^{n-2} (\alpha m 2^m \log 2 + 2^m) + 2 * 2^0 \\ &= 2\alpha \log 2 \sum_{m=1}^{n-2} m 2^m + 2 \sum_{m=0}^{n-2} 2^m + 2. \end{aligned} \tag{6.3}$$

Now we need to evaluate the sum of an arithmetic-geometric sequence $m2^m, m = 1, \ldots, n-2$. Letting

$$U = \sum_{m=1}^{n-2} m 2^m, \tag{6.4}$$

multiply by 2 on both sizes

$$2U = \sum_{m=1}^{n-2} m2^{m+1} = \sum_{m=2}^{n-1} (m-1)2^m. \tag{6.5}$$

Subtracting (6.4) from (6.5), we get

$$U = (n-2)2^{n-1} - \sum_{m=2}^{n-2} 2^m - 1. \tag{6.6}$$

Using the fact $n = \log N / \log 2$,

$$\begin{aligned} S_2 &= 2\alpha \log 2 \left((n-2)2^{n-1} - \sum_{m=2}^{n-2} 2^m - 1 \right) + 2 \sum_{m=0}^{n-2} 2^m + 2 \\ &= \alpha(n-2)2^n \log 2 - \alpha 2^n \log 2 + 2^n + 8\alpha \log 2 \\ &= \alpha N \log N - (3\alpha \log 2 + 1)N + 8\alpha \log 2. \end{aligned} \tag{6.7}$$

Thus, the total number of floating-point operations required to calculate the DOST coefficients is

$$\begin{aligned} S &= S_1 + S_2 \\ &= 2\alpha N \log N - (3\alpha \log 2 + 1)N + 8\alpha \log 2 \\ &= \mathcal{O}(N \log N). \end{aligned} \tag{6.8}$$

In general, for a series of size between 2^{n-1} and 2^n, the total operations to calculate the DOST coefficients would still be $\mathcal{O}(N \log N)$, but with a different factor.

The computational complexity for the reconstruction and the alternative symmetric version can be proven in a similar fashion, which completes this proof. □

7. Conclusions and future work.

In this paper, we have shown that the DOST and its inverse can be computed in $\mathcal{O}(N \log N)$, comparable to the FFT algorithm. This accomplishment is made possible by a simple factorization of the DOST transformation matrix. These stunning speed gains make the DOST feasible for a vastly larger set of problems than was previously thought. Due to the fact that the DOST is separable over different dimensions, the computational complexity of computing the FDOST of a k-dimensional signal is $\mathcal{O}(N^k \log N)$.

Until now, the ST and DOST were not typically used on large images or on three-dimensional (3-D) volume data sets. The efficient computation and the analysis of frequency data is extremely important in signal processing, medical imaging, remote sensing, and other related fields. We believe that the FDOST algorithm will become more and more widely used in problem domains such as image compression, restoration, filtering, and registration. We plan to wield this efficient technique to investigate the uses of the ST in realms where it was previously too cumbersome to apply.

REFERENCES

[1] S. Andino, R. Menendez, C. Lantz, O. Blank, C. Michel, and T. Landis, *Non-stationary distributed source approximation: An alternative to improve localization procedures*, Human Brain Mapping, 14 (2001), pp. 81–95.

[2] S. Assous, A. Humeau, M. Tartas, P. Abraham, and J. L'Huillier, *Physiological effects of indomethacin and celecobix: An S-transform laser Doppler flowmetry signal analysis*, Phys. Med. Biol., 50 (2005), pp. 1951–1959.

[3] P. Dash, B. Panigrahi, and G. Panda, *Power quality analysis using S-transform*, IEEE Trans. Power Delivery, 18 (2003), pp. 406–411.

[4] I. Daubechies, *Ten Lectures on Wavelets*, CBMS-NSF Regional Conf. Ser. in Appl. Math. 61, SIAM, Philadelphia, 1992.

[5] S. Drabycz, R. G. Stockwell, and J. R. Mitchell, *Image texture characterization using the discrete orthonormal S-transform*, J. Digit. Imaging, (2008).

[6] M. Eramian, R. A. Schincariol, R. G. Stockwell, R. P. Lowe, and L. Mansinha, *Review of applications of* 1*D and* 2*D S-transforms*, Proc. SPIE, 3078 (1997), pp. 558–568.

[7] B. G. Goodyear, H. Zhu, R. A. Brown, and J. R. Mitchell, *Removal of phase artifacts from fMRI data using a Stockwell transform filter improves brain activity detection*, Magnetic Resonance in Med., 51 (2004), pp. 16–21.

[8] L. Mansinha, R. G. Stockwell, R. Lowe, M. Eramian, and R. A. Schincariol, *Local S-spectrum analysis of* 1*-D and* 2*-D data*, Phys. Earth Planet. Inter., 103 (1997), p. 329–336.

[9] L. Mansinha, R. G. Stockwell, and R. P. Lowe, *Pattern analysis with two-dimensional spectral localisation: Application of two-dimentional S transforms*, Phys. A, 239 (1997), pp. 286–295.

[10] P. McFadden, J. Cook, and L. Forster, *Decomposition of gear vibration signals by the generalised S-transform*, Mech. Systems Signal Process., 13 (1999), pp. 691–707.

[11] E. G. Merzlyakov, Y. I. Portnyagin, C. Jacobi, N. J. Mitchell, H. G. Muller, A. H. Manson, A. N. Fachrutdinova, W. Singer, and P. Hoffmann, *On the longitudinal structure of the transient day-to-day variation of the semidiurnal tide in the mid-latitude lower thermosphere—I. Winter season*, Ann. Geophys., 19 (2001), pp. 545–562.

[12] C. R. Pinnegar and L. Mansinha, *The Bi-Gaussian S-transform*, SIAM J. Sci. Comput., 24 (2003), pp. 1678–1692.

[13] S. Qian and D. Chen, *Discrete Gabor transform*, IEEE Trans. Signal Process., 41 (1993), pp. 2429–2438.

[14] R. G. Stockwell, *A basis for efficient representation of the S-transform*, Digit. Signal Process., 17 (2007), pp. 371–393.

[15] R. G. Stockwell and R. Lowe, *Airglow imaging of gravity waves—*1*. Results from a small network of OH nightglow scanning imagers*, J. Geophys. Res., 106 (2001), pp. 185–203.

[16] R. G. Stockwell, L. Mansinha, and R. P. Lowe, *Localization of complex spectrum: The S transform*, IEEE Trans. Signal Process., 144 (1996), pp. 998–1001.

[17] M. Varanini, G. Paolis, M. Emdin, A. Macerata, S. Pola, M. Cipriani, and C. Marchesi, *Spectral analysis of cardiovascular time series by the S-transform*, in Computers in Cardiology, 1997, pp. 383–386.

[18] Y. Wang and J. Orchard, *Symmetric discrete orthonormal Stockwell transform*, American Istitute of Physics (AIP) Conference Proceedings, 1048 (2008), pp. 585–588.