CHAPTER 7

Short-Time Fourier Transform Analysis and Synthesis

7.1 Introduction

In analyzing speech signal variations with the discrete-time Fourier transform, we encounter the problem that a single Fourier transform cannot characterize changes in spectral content over time such as time-varying formants and harmonics. In contrast, the discrete-time *short-time Fourier transform* (STFT), introduced in our discussion of discrete-time spectrographic analysis in Chapter 3, consists of a separate Fourier transform for each instant in time. In particular, we associate with each instant the Fourier transform of the signal in the neighborhood of that instant, so that spectral evolution of the signal can be traced in time. For practical implementation, each Fourier transform in the STFT is replaced by the discrete Fourier transform (DFT). The resulting STFT is discrete in both time and frequency. We call this the *discrete STFT* to distinguish it from the discrete-time STFT, which is continuous in frequency. These two transforms, their properties and inter-relationships, and a glimpse into their use in speech processing analysis/synthesis applications are the primary focus of this chapter.

The discrete-time STFT and discrete STFT were implicitly used in the previous two chapters in two analysis/synthesis techniques, based on linear prediction and homomorphic filtering, in applying a short-time window to the waveform $x[n]$ to account for time variations in the underlying source and system. These two analysis/synthesis methods are *model-based* in the sense that they rely on a source/filter speech model and often on a further parameterization with rational (pole-zero) z-transforms to represent the vocal tract system function and glottal source. In this chapter, we develop a new set of analysis/synthesis techniques that largely do not require such strict source/filter speech models.

In Section 7.2 of this chapter, we give a formal introduction to the discrete-time and the discrete STFTs for analysis, as well as a brief look into their time-frequency resolution properties. We then consider, in Section 7.3, the problem of synthesizing a sequence from its STFT. While

this is straightforward for the discrete-time STFT, a number of important STFT concepts are introduced for addressing the more challenging task of synthesis from the discrete STFT. The basic theory part of the chapter is essentially concluded in Section 7.4 in treating the magnitude of the STFT as a transform in its own right. Next, in Section 7.5, we consider the important practical problem of estimating a signal from a modified STFT or STFT magnitude that does not satisfy the definitional constraints of the STFT, leading to many practical applications of the STFT and STFT magnitude in speech processing. The particular applications of time-scale modification and noise reduction are introduced in Section 7.6.

7.2 Short-Time Analysis

In this section, following the development in [13],[20], we explore two different views of the STFT: (1) the Fourier transform view and (2) the filter bank view. We begin with the first perspective, which draws on the STFT representation of a sequence being analogous to that of the Fourier transform.

7.2.1 Fourier Transform View

The expression for the discrete-time STFT at time n was given in Chapter 3 as[1]

$$X(n,\omega) = \sum_{m=-\infty}^{\infty} x[m]w[n-m]e^{-j\omega m}, \tag{7.1}$$

where $w[n]$ is assumed to be non-zero only in the interval $[0, N_w - 1]$ and is referred to as the *analysis window* or sometimes as the *analysis filter* for reasons that will become clear later in this chapter. The sequence $f_n[m] = x[m]w[n-m]$ is called a short-time section of $x[m]$ at time n. This sequence is obtained by time-reversing the analysis window, $w[m]$, shifting the result by n points, and multiplying it with $x[m]$. With the short-time section for time n, we can take its Fourier transform to obtain the frequency function $X(n,\omega)$. This series of operations is illustrated in Figure 7.1. To obtain $X(n+1,\omega)$, we slide the time-reversed analysis window one point from its previous position, multiply it with $x[m]$, and take the Fourier transform of the resulting short-time section. Continuing this way, we generate a set of discrete-time Fourier transforms that together constitute the discrete-time STFT. Typically, the analysis window is selected to have a much shorter duration than the signal $x[n]$ for which the STFT is computed; as we have seen for a speech waveform, the window duration is typically set at about 20–30 ms or a few pitch periods.

By analogy with the discrete Fourier transform (DFT), the discrete STFT is obtained from the discrete-time STFT through the following relation:

$$X(n,k) = X(n,\omega)|_{\omega=\frac{2\pi}{N}k}, \tag{7.2}$$

where we have sampled the discrete-time STFT with a *frequency sampling interval*[2] of $\frac{2\pi}{N}$ in order to obtain the discrete STFT. We refer to N as the *frequency sampling factor*. Substituting

[1] We have changed our notation slightly from that in Chapter 3, where we denoted the STFT by $X(\omega,\tau)$.

[2] More strictly, the sampling occurs over one period so that $X(n,k) = X(n,\omega)|_{\omega=\frac{2\pi}{N}k}$ for $k = 0, 1, \ldots N-1$, and zero elsewhere, but here we think of the DFT as periodic with period N [14].

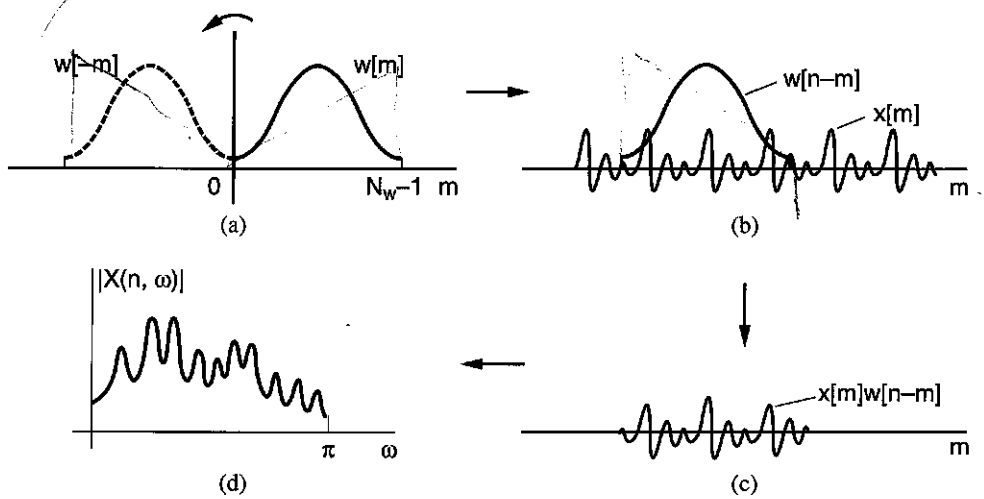

Figure 7.1 Series of operations required to compute a short-time section and STFT: (a) flip window; (b) slide the window sample-by-sample over the sequence (*Note:* $w[-(m-n)] = w[n-m]$); (c) multiply the sequence by the displaced window; (d) take the Fourier transform of the windowed segment.

Equation (7.1) into Equation (7.2), we obtain the following relation between the discrete STFT and its corresponding sequence $x[n]$:

$$X(n,k) = \sum_{m=-\infty}^{\infty} x[m]w[n-m]e^{-j\frac{2\pi}{N}km}. \tag{7.3}$$

The following example illustrates the distinction between the discrete STFT and the discrete-time STFT.

EXAMPLE 7.1 Consider the periodic unit sample sequence $x[n] = \sum_{l=-\infty}^{\infty} \delta[n - lP]$ and analysis window $w[n]$ a triangle of length P. The discrete-time STFT is given by

$$\begin{aligned} X(n,\omega) &= \sum_{m=-\infty}^{\infty} x[m]w[n-m]e^{-j\omega m} \\ &= \sum_{m=-\infty}^{\infty} \left(\sum_{l=-\infty}^{\infty} \delta[m - lP] \right) w[n-m]e^{-j\omega m} \\ &= \sum_{l=-\infty}^{\infty} w[n - lP]e^{-j\omega lP} \end{aligned}$$

where, in the last step, the delta function picks off the values of $m = lP$. The discrete-time STFT is a series of windows translated in time by lP samples and with linear phase $-\omega lP$. For each time n, because the translated windows are nonoverlapping, $X(n,\omega)$ has a constant magnitude and linear phase along the frequency dimension. Also, because the $w[n - lP]$ are non-overlapping, for each frequency ω, $X(n,\omega)$ has a magnitude that follows the replicated triangular window and a phase that is fixed in frequency, i.e., $\angle X(n,\omega) = -\omega lP$, along the time dimension.

Consider now the discrete STFT and suppose that the frequency sampling factor $N = P$. Then the discrete STFT is given by

$$\begin{aligned} X(n,k) &= \sum_{m=-\infty}^{\infty} x[m]w[n-m]e^{-j\frac{2\pi}{N}km} \\ &= \sum_{m=-\infty}^{\infty}\left(\sum_{l=-\infty}^{\infty}\delta[m-lP]\right)w[n-m]e^{-j\frac{2\pi}{P}km} \\ &= \sum_{l=-\infty}^{\infty} w[n-lP]e^{-j\frac{2\pi}{P}klP} \\ &= \sum_{l=-\infty}^{\infty} w[n-lP] \end{aligned}$$

and so consists of translated non-overlapping windows. For each discrete frequency k, the magnitude of $X(n,k)$ follows the translated windows and the phase is zero; because the frequency sampling interval equals the fundamental frequency of the periodic sequence $x[n]$, we see no phase change in time. ▲

Recall that the two-dimensional function $|X(n,\omega)|^2$ was called the *spectrogram* in Chapter 2. For a "short" window $w[n]$, as in Example 7.1 where the window duration is one pitch period, $|X(n,\omega)|^2$ is referred to as the *wideband spectrogram*, exhibiting periodic temporal structure in $x[n]$ as "vertical striations." In Example 7.1, these vertical striations correspond to the repeated, non-overlapping windows $w[n-lP]$. When the window $w[n]$ is "long" in duration, e.g., a few pitch periods in duration, we refer to $|X(n,\omega)|^2$ as the *narrowband spectrogram*, exhibiting the harmonic structure in $x[n]$ as "horizontal striations" (Exercise 7.2).

In many applications, the time variation (the n dimension) of $X(n,k)$ is decimated by a temporal decimation factor, L, to yield the function $X(nL,k)$. Just as the discrete-time STFT can be viewed as a set of Fourier transforms of the short-time sections $f_n[m] = x[m]w[n-m]$, the discrete STFT in Equation (7.3) is easily seen to be a set of DFTs of the short-time sections $f_n[m]$. When the time dimension of the discrete STFT is decimated, the corresponding short-time sections $f_{nL}[m]$ are a subset of $f_n[m]$ obtained by incrementing time by multiples of L. This notion is illustrated in Figure 7.2. How we chose sampling rates in time and frequency for a unique representation of $x[n]$ is a question that we will return to later.

In viewing $X(n,\omega)$ as a Fourier transform for each fixed n, we see that the frequency function $X(n,\omega)$ for each n has all the general properties of a Fourier transform [13]. For example, with respect to the frequency variable ω, $X(n,\omega)$ is periodic with period 2π and Hermetian symmetric for real sequences. Another property, analogous to that of the Fourier transform, is that a time shift in a sequence leads to a linear phase factor in the frequency domain. Suppose we shift $x[n]$ by n_o samples. Then with a change in variables $q = m - n_o$, we have

$$\begin{aligned} \tilde{X}(n,\omega) &= \sum_{m=-\infty}^{\infty} x[m-n_o]w[n-m]e^{-j\omega m} = \sum_{q=-\infty}^{\infty} x[q]w[n-n_o-q]e^{-j\omega(q+n_o)} \\ &= e^{-j\omega n_o}\sum_{q=-\infty}^{\infty} x[q]w[n-n_o-q]e^{-j\omega q} = e^{-j\omega n_o}X(n-n_o,\omega). \end{aligned}$$

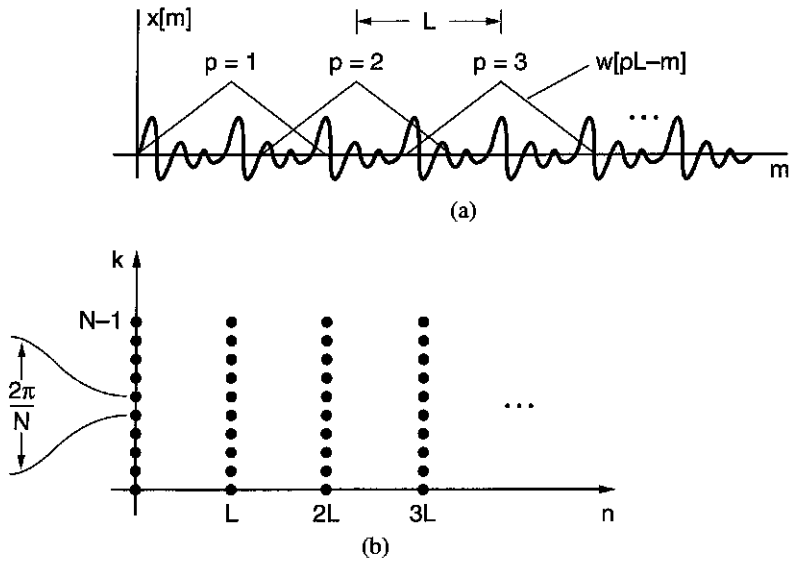

Figure 7.2 Time and frequency decimation used in computing the discrete STFT $X(nL, k)$: (a) analysis window positions; (b) time-frequency sampling.

SOURCE: S.H. Nawab and T.F. Quatieri, "Short-Time Fourier Transform" [13]. ©1987, Pearson Education, Inc. Used by permission.

Thus, a shift by n_o in the original time sequence introduces a linear phase, but also a shift in time, corresponding to a shift in each short-time section by n_o. Likewise, most of the properties of the discrete-time STFT have a straightforward extension to the discrete STFT [13]. For the above shifting property, however, it should be noted that if the discrete STFT is decimated in time by a factor L, when the shift is not an integer multiple of L, there is no general relationship between the discrete STFTs of $x[n]$ and $x[n-n_o]$. This happens because the short-time sections corresponding to $X(nL, k)$ cannot be expressed as shifted versions of the short-time sections corresponding to the discrete STFT of $x[n - n_o]$ [13].

In the implementation of the discrete STFT from the Fourier transform view, the FFT can be used to efficiently compute $X(n, k)$ by computing the time-aliased version of each short-time section and then applying the N-point FFT to each of those sections [13]. If N is greater than or equal to the analysis window length, the computation of the time-aliased version is eliminated.

7.2.2 Filtering View

The STFT can also be viewed as the output of a filtering operation where the analysis window $w[n]$ plays the role of the filter impulse response, and hence the alternative name *analysis filter* for $w[n]$. For the filtering view of the STFT, we fix the value of ω at ω_o (in the Fourier transform view, we had fixed the value of n), and rewrite Equation (7.1) as

$$X(n, \omega_o) = \sum_{m=-\infty}^{\infty} (x[m]e^{-j\omega_o m})w[n - m]. \tag{7.4}$$

We then recognize from the form of Equation (7.4) that the STFT represents the convolution of the sequence $x[n]e^{-j\omega_o n}$ with the sequence $w[n]$. Using convolution notation in Equation (7.4), we obtain

$$X(n, \omega_o) = (x[n]e^{-j\omega_o n}) * w[n]. \tag{7.5}$$

The product $x[n]e^{-j\omega_o n}$ can be interpreted as the modulation of $x[n]$ up to frequency ω_o. Thus, $X(n, \omega_o)$ for each ω_o is a sequence in n which is the output of the process illustrated in Figure 7.3a. The signal $x[n]$ is modulated with $e^{-j\omega_o n}$ and the result passed through a filter whose impulse response is the analysis window, $w[n]$. We can view this as a modulation of a band of frequencies in $x[n]$ around ω_o down to baseband, and then filtered by $w[n]$ (Figure 7.3b).

A slight variation on the filtering and modulation view of the STFT is obtained by manipulating Equation (7.5) into the following form (Exercise 7.1):

$$X(n, \omega_o) = e^{-j\omega_o n}(x[n] * w[n]e^{j\omega_o n}). \tag{7.6}$$

In this case, the sequence $x[n]$ is first passed through the same filter as in the previous case except for a linear phase factor. The filter output is then modulated by $e^{-j\omega_o n}$. This view of the time variation of the STFT for a fixed frequency is illustrated in Figure 7.4a. We can view this as filtering out a band of frequencies that is then demodulated down to baseband (Figure 7.4b).

The discrete STFT of Equation (7.3) can also be interpreted from the filtering viewpoint. In particular, having a finite number of frequencies allows us to view the discrete STFT as the output of the filter bank shown in Figure 7.5a, i.e.,

$$X(n, k) = e^{-j\frac{2\pi}{N}kn}(x[n] * w[n]e^{j\frac{2\pi}{N}kn}).$$

Observe that each filter is acting as a bandpass filter centered around its selected frequency. Thus, the discrete STFT can be viewed as a collection of sequences, each corresponding to the frequency components of $x[n]$ falling within a particular frequency band, as illustrated in the following example. This implementation is similar to that of the filter banks used in the early analog spectrograms [8].

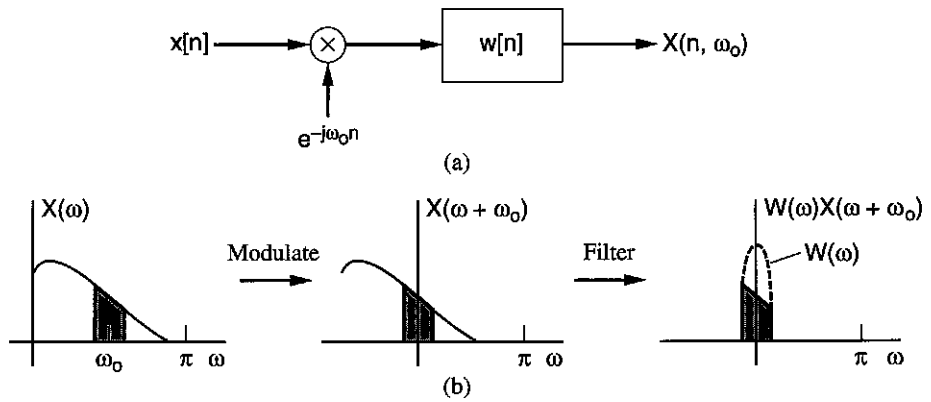

Figure 7.3 Filtering view of STFT analysis at frequency ω_o: (a) block diagram of complex exponential modulation followed by a lowpass filter; (b) operations in the frequency domain.

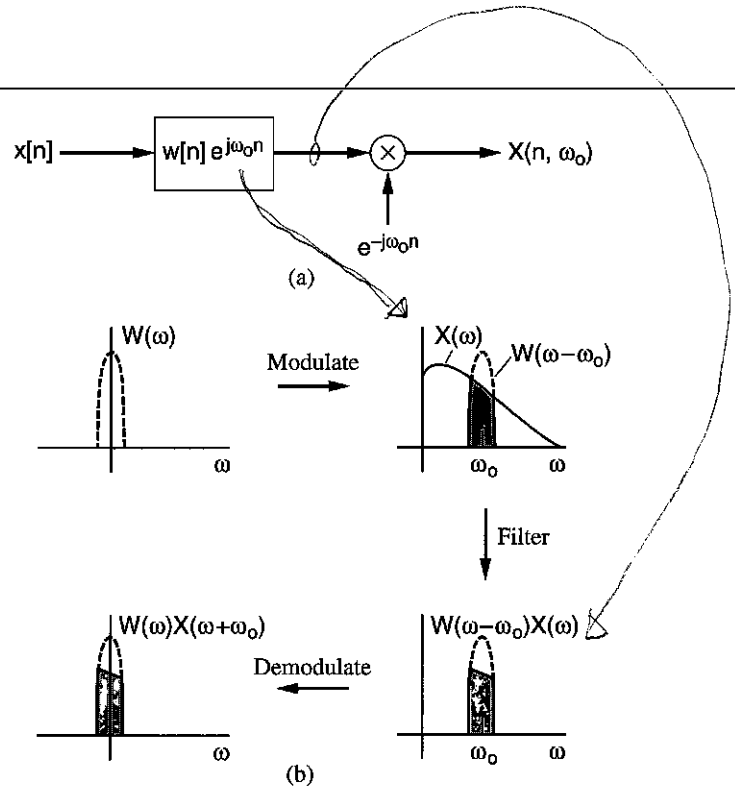

Figure 7.4 Alternative filtering view of STFT analysis at frequency ω_o: (a) block diagram of bandpass filtering followed by complex exponential modulation; (b) operations in the frequency domain.

EXAMPLE 7.2 Consider a window that is Gaussian in shape, i.e., of the form $w[n] = e^{a(n-n_o)^2}$, shown in Figure 7.6a (for $n_o = 0$). (For the STFT, the window is made causal and of finite length.) The discrete STFT with DFT length N, therefore, can be considered as a bank of filters with impulse responses

$$h_k[n] = e^{a(n-n_o)^2} e^{j\frac{2\pi}{N}kn}.$$

If the input sequence $x[n] = \delta[n]$, then the output of the kth bandpass filter is simply $h_k[n]$. Figure 7.6b,c,d shows the real component of the impulse response for three bandpass filters ($k = 5, 10, 15$) for a discrete STFT where $N = 50$ corresponds to bandpass filters spaced by 200 Hz for a sampling rate of 10000 samples/s. Each impulse response has an *envelope*, i.e., the Gaussian window, that multiplies the modulation function $e^{j\frac{2\pi kn}{N}}$. Observe that the output of filter $k = 0$ equals the Gaussian window of Figure 7.6a. ▲

As with the Fourier transform interpretation, the filtering view also allows us to easily deduce a number of STFT properties. In particular, we view $X(n, \omega)$ as a filter output for each fixed frequency. Therefore, the time variation of $X(n, \omega)$ for each ω has all the general properties of a filtered sequence. We list a few of these properties below [13].

P1: If $x[n]$ has length N and $w[n]$ has length M, then $X(n, \omega)$ has length $N + M - 1$ along n.

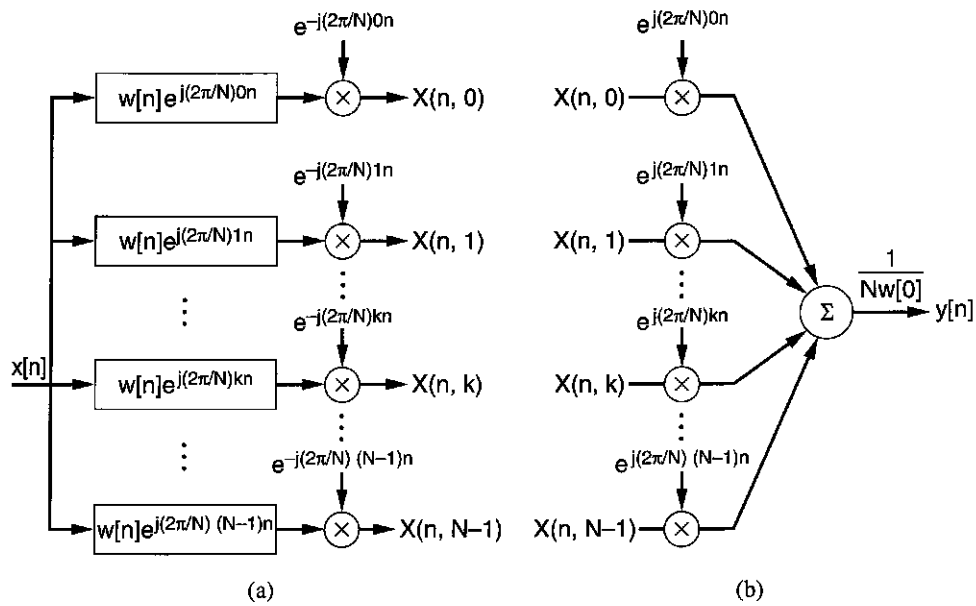

Figure 7.5 The filtering view of analysis and synthesis with the discrete STFT: (a) the discrete STFT (analysis) as the output of a filter bank consisting of bandpass filters; (b) filter bank summation procedure for signal synthesis from the discrete STFT.

SOURCE: S.H. Nawab and T.F. Quatieri, "Short-Time Fourier Transform" [13]. ©1987, Pearson Education, Inc. Used by permission.

P2: The bandwidth of the sequence (along n) $X(n, \omega_o)$ is less than or equal to the bandwidth of $w[n]$.

P3: The sequence $X(n, \omega_o)$ has the spectrum centered at the origin.

In the first property, we make use of a standard result for the length of a sequence obtained through the convolution of any two sequences of lengths N and M. For the second property, we note that $X(n, \omega)$ as a function of n is the output of a filter whose bandwidth is the bandwidth of the analysis window. The third property follows from the modulation steps used in obtaining $X(n, \omega_o)$ (Figures 7.3 and 7.4). The STFT properties from the filtering viewpoint remain the same for the discrete STFT since, for a fixed frequency, the time variation of the discrete STFT is the same as the time variation of the discrete-time STFT at that frequency. The next example illustrates properties **P2** and **P3** for the filter bank of Example 7.2.

EXAMPLE 7.3 Consider the filter bank of Example 7.2 that was designed with a Gaussian window of the form $w[n] = e^{a(n-n_o)^2}$, shown in Figure 7.6a. Figure 7.7 shows the Fourier transform magnitudes of the output of the four complex bandpass filters $h_k[n]$ for $k = 0, 5, 10,$ and15 of Figure 7.6. Each spectral bandwidth is identical to the bandwidth of the Gaussian window. Indeed, in each case the spectral magnitude equals that of the window transform, except for a frequency shift, and must itself be Gaussian in shape. (The reader should verify this property). After demodulation by $e^{-j\frac{2\pi}{N}kn}$ (as in Equation (7.6) and its discrete version), the resulting bandpass outputs have the same spectral shapes as in the figure, but centered at the origin. ▲

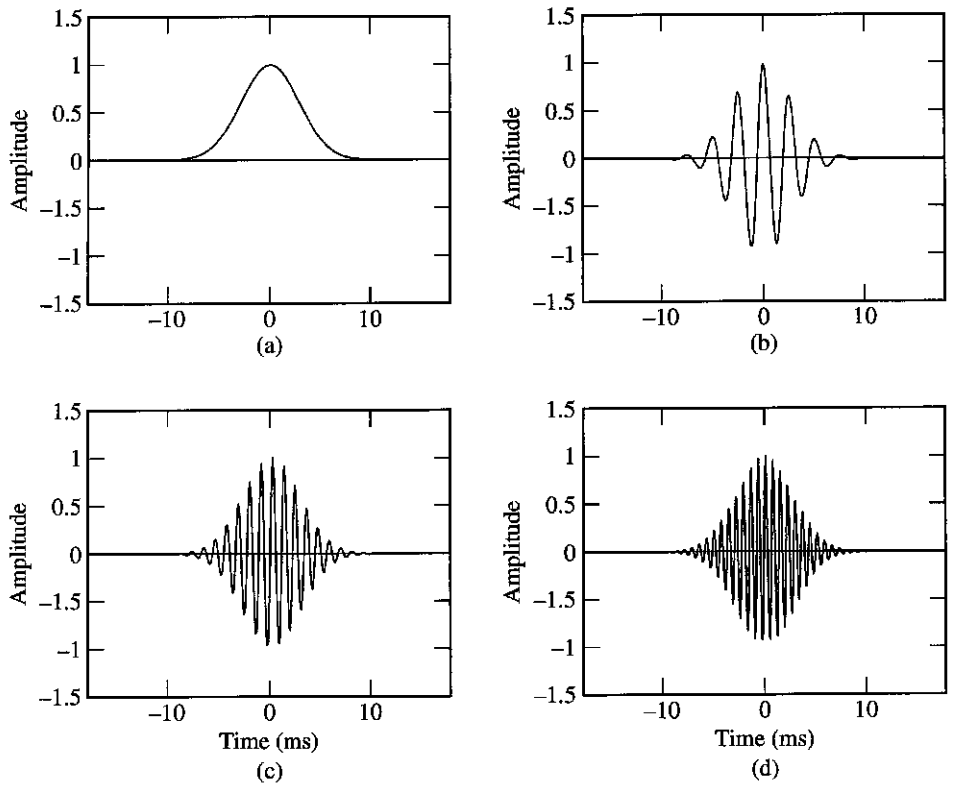

Figure 7.6 Real part of the bandpass filter outputs with unit sample input $\delta[n]$ for the discrete filter bank of Example 7.2 prior to demodulation: (a) Gaussian window $w[n]$ (also output of filter $k = 0$); (b) discrete frequency $k = 5$; (c) discrete frequency $k = 10$; (d) discrete frequency $k = 20$. Frequency of the output increases with increasing k, while the Gaussian envelope remains intact.

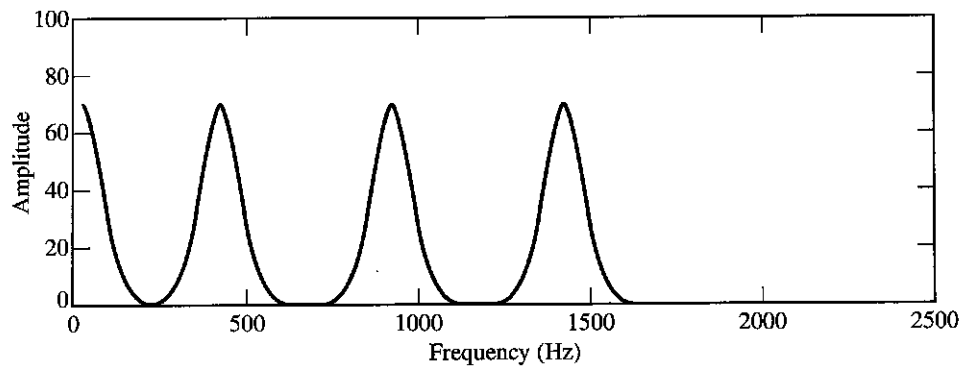

Figure 7.7 Superimposed spectra of bandpass filter outputs of Figure 7.6.

Finally, in the implementation of the discrete STFT from the filtering view, the signal $x[n]$ is passed through a bank of filters, shown in Figure 7.5a, where the output of each filter is the time variation of the STFT at frequency ω_k. If the output of each filter is decimated in time by a factor L, then we obtain the discrete STFT $X(nL, k)$.

7.2.3 Time-Frequency Resolution Tradeoffs

We have seen in Chapter 3 that a basic issue in analysis window selection is the compromise required between a long window for showing signal detail in frequency and a short window for representing fine temporal structure. We first recall that the STFT, $X(n, \omega)$, is the Fourier transform of the short-time section $f_n[m] = x[m]w[n - m]$. From Fourier transform theory, we know that the Fourier transform of the product of two sequences is given by the convolution of their respective Fourier transforms. With $X(\omega)$ as the Fourier transform of $x[m]$ and $W(-\omega)e^{j\omega n}$ as the Fourier transform of $w[n - m]$ with respect to the variable m, we can write the STFT as [13],[17]

$$X(n, \omega) = \frac{1}{2\pi} \int_{-\pi}^{\pi} W(\theta)e^{j\theta n}X(\omega + \theta)d\theta. \tag{7.7}$$

Thus, the frequency variation of the STFT for any fixed time may be interpreted as a smoothed version of the Fourier transform of the underlying signal. Thus, for faithful reproduction of the properties of $X(\omega)$ in $X(n, \omega)$, the function $W(\omega)$ should appear as an impulse with respect to $X(\omega)$. The closer $W(\omega)$ is to an impulse (i.e., with narrow bandwidth), $X(n, \omega)$ is said to have better *frequency resolution*. However, for a given window, frequency resolution varies inversely with the effective length of the window. Thus, good frequency resolution requires long analysis windows, whereas the desire for short-time analysis, and thus good *time resolution*, requires short analysis windows, as illustrated in the following example.

EXAMPLE 7.4 An example of the time-frequency resolution tradeoff is given in Figure 7.8, which shows the Fourier transform magnitude of a section of a chirp signal whose frequency is a linear function of time [13]. The aim is to measure the instantaneous frequency at the center of the chirp. This is performed by using rectangular analysis windows of various lengths. The very short window of duration 5 ms gives good temporal resolution, but low frequency resolution because of spectrum smoothing by a wide window in frequency. The very long window of duration 20 ms yields a wide spectrum, reflecting the time-varying frequency of the chirp, giving neither good temporal nor spectral resolution. An intermediate window length of 10 ms provides a good tradeoff in time and frequency resolution. ▲

We see implied in the previous example that Equation (7.7) is not a valid smoothing interpretation when the sequence $x[n]$ is not stationary. One approach, however, to apply this interpretation in the speech context is to *pretend* that a short-time segment comes from a longer stationary signal. For example, we can pretend that a quasi-periodic speech segment comes from an infinitely long periodic sequence. With this viewpoint, we can use the smoothing interpretation of Equation (7.7) and deduce the time-frequency tradeoff that we described in Chapter 3 in relation to the narrowband and wideband spectrograms for voiced speech. We illustrate the particular case of harmonic spectra in Figure 7.9 for a long and a short window. The long window better represents the spectral harmonicity and results in harmonic amplitudes that better reflect the underlying vocal tract spectral envelope. The short window blurs the spectral harmonicity,

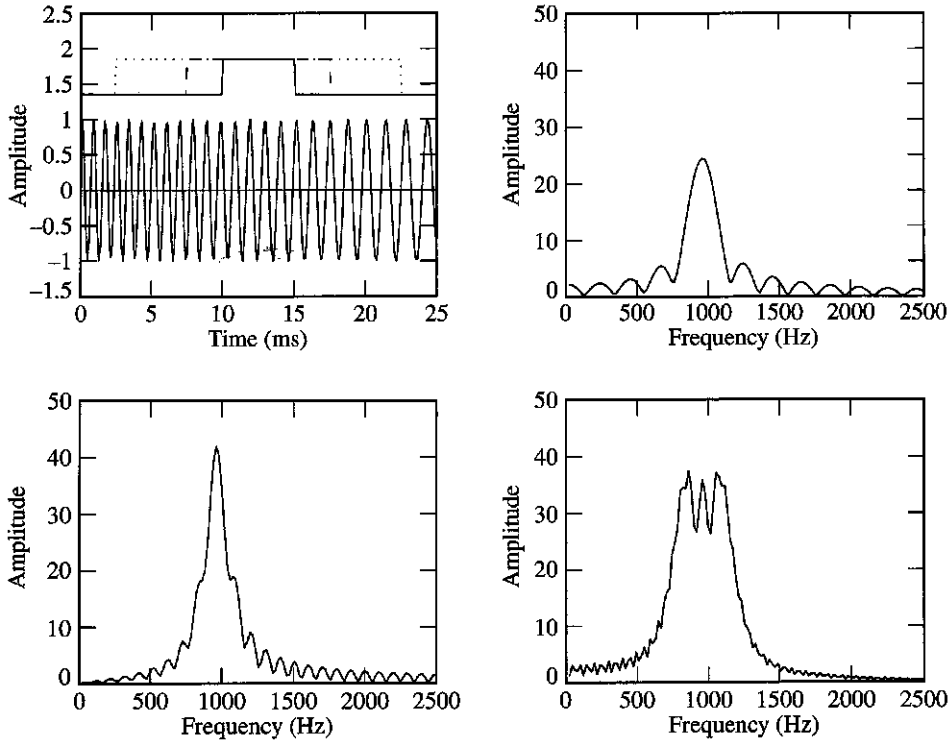

Figure 7.8 Effect of the length of the analysis window on the discrete Fourier transform of a linearly frequency-modulated sinusoid of length 25 ms whose frequency decreases from 1250 Hz to 625 Hz. The discrete Fourier transform uses a rectangular window centered at 12.5 ms, as illustrated in panel (a). Transforms are shown for three different window lengths: (b) 5 ms [solid in (a)]; (c) 10 ms [dashed in (a)]; (d) 20 ms [dotted in (a)].

SOURCE: S.H. Nawab and T.F. Quatieri, "Short-Time Fourier Transform" [13]. ©1987, Pearson Education, Inc. Used by permission.

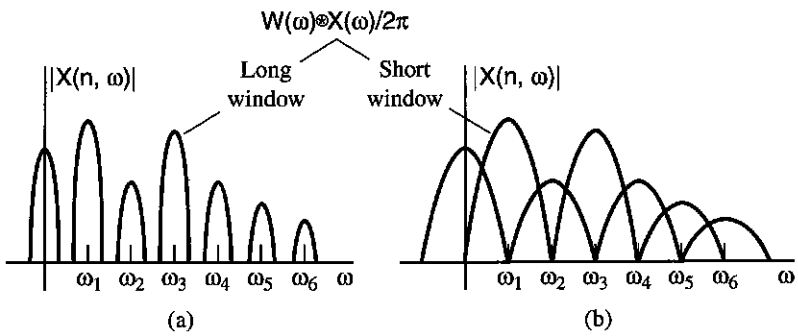

Figure 7.9 Schematic of convolutional view of time-frequency resolution tradeoff with long and short analysis windows for harmonic spectra: (a) long window; (b) short window.

and degrades the harmonic amplitudes, but better captures changes in the harmonicity and the spectral envelope. More generally, a similar view holds for unvoiced sounds (e.g., plosives and fricatives). Nevertheless, we are still faced with the problem that some sounds are stationary over only very short time spans, or are never stationary, as we saw in Chapter 3 with diphthongs and with rapid conversational speech. A similar problem arises when sounds of very different spectral and temporal character are closely spaced, as with a voiced plosive preceding a vowel. In these cases, poor frequency resolution with the STFT may be our only option when good time resolution is desired. Chapter 11 will address these tradeoffs in more detail, as well as alternative time-frequency distributions that provide resolution tradeoffs different from that with the spectrogram.

7.3 Short-Time Synthesis

In this section, we first consider the problem of obtaining a sequence back from its discrete-time STFT. The inversion is represented mathematically by a *synthesis equation* which expresses a sequence in terms of its discrete-time STFT. Whereas the discrete-time STFT is always invertible in this sense, the discrete STFT requires certain constraints on its sampling rate for invertibility. A common approach to developing synthesis methods for the discrete STFT has been to start from one of the many synthesis equations [13],[20]. A discretized version of such an equation is then considered as the basis for a candidate synthesis method and conditions are derived under which such a method can synthesize a sequence from its discrete STFT. We describe two classical methods, the Filter Bank Summation (FBS) method and the Overlap-Add (OLA) method.

7.3.1 Formulation

The invertibility of the discrete-time STFT is easily seen through the Fourier transform interpretation of the STFT, where the discrete-time STFT is viewed for each value of n as a function of frequency obtained by taking the Fourier transform of the short-time section $f_n[m] = x[m]w[n-m]$. It follows that if, for each n, we take the inverse Fourier transform of the corresponding function of frequency, then we obtain the sequence $f_n[m]$. If we evaluate this short-time section at $m = n$, we obtain the value $x[n]w[0]$. Assuming that $w[0]$ is non-zero, we can divide by $w[0]$ to recover $x[n]$. The process of taking the inverse Fourier transform of $X(n, \omega)$ for a specific n and then dividing by $w[0]$ is represented by the following relation:

$$x[n] = \frac{1}{2\pi w[0]} \int_{-\pi}^{\pi} X(n, \omega) e^{j\omega n} d\omega. \tag{7.8}$$

This equation represents a *synthesis equation* for the discrete-time STFT. In fact, there are numerous synthesis equations that map $X(n, \omega)$ uniquely back to $x[n]$. Observe that there is much redundancy in the STFT where the analysis window slides one sample at a time; therefore, the use of one inverse Fourier transform per time sample in Equation (7.8) is inefficient.

In contrast to the discrete-time STFT, the discrete STFT $X(n, k)$ is not always invertible. For example, consider the case when $w[n]$ is bandlimited with bandwidth of B. Figure 7.10 shows the filter regions used to obtain $X(n, k)$ for the case when the sampling interval $\frac{2\pi}{N}$ is greater than B. Note that in this case there are frequency components of $x[n]$ which do not pass through any of the filter regions of the discrete STFT. Those frequency components can have arbitrary values and yet we would have the same discrete STFT. Thus, in these cases, the

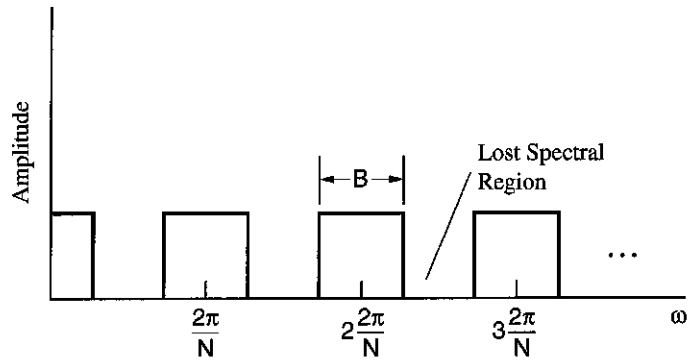

Figure 7.10 Undersampled STFT when the frequency sampling interval $\frac{2\pi}{N}$ is greater than the analysis-filter bandwidth B.

SOURCE: S.H. Nawab and T.F. Quatieri, "Short-Time Fourier Transform" [13]. ©1987, Pearson Education, Inc. Used by permission.

discrete STFT is not a unique representation of $x[n]$ and therefore cannot be invertible. The invertibility problem is also of interest when the discrete STFT has been decimated in time. For example, consider the case when the analysis window $w[n]$ is non-zero over its length N_w. When temporal decimation factor L is greater than N_w, there are samples of $x[n]$ which are not included in any short-time section of the discrete STFT. These samples can have arbitrary values and yet we would have the same time-decimated discrete STFT. Thus, in such cases the discrete STFT is again not a unique representation of $x[n]$ and therefore cannot be invertible.

By selecting appropriate constraints on the frequency-sampling and time-decimation rates, the discrete STFT becomes invertible. As an example, we consider the case of a finite-length analysis window. We have already seen that in such cases the discrete STFT is not invertible if the temporal decimation factor L is greater than the analysis window length N_w. We now see that if the temporal decimation factor is less than or equal to the analysis window length, then the discrete STFT is invertible, provided we impose constraints on the frequency sampling interval. Suppose that the temporal decimation factor is equal to the analysis window length. The discrete STFT in this case consists of the DFTs of adjacent but non-overlapping short-time sections. Thus, to reconstruct $x[n]$ from its discrete STFT, we must require that each N_w-point short-time section be recoverable from its DFT. However, the DFT of an N_w point sequence is invertible, provided its frequency sampling interval is less than or equal to $\frac{2\pi}{N_w}$ [14]. It follows that the discrete STFT is invertible when the analysis window is non-zero over its finite-length N_w, the temporal decimation factor $L \leq N_w$, and the frequency sampling interval $\frac{2\pi}{N} \leq \frac{2\pi}{N_w}$. We will see in the following sections that more relaxed bounds can be derived in the sense that the above conditions are sufficient but not necessary for invertibility.

7.3.2 Filter Bank Summation (FBS) Method

In this section we present a traditional short-time synthesis method that is commonly referred to as the Filter Bank Summation (FBS) method [1],[13],[20]. This method is best described in terms of the filtering interpretation of the discrete STFT. In this interpretation, the discrete STFT is considered to be the set of outputs of a bank of filters. In the FBS method, the output of each filter is modulated with a complex exponential, and these modulated filter outputs are summed at

each instant of time to obtain the corresponding time sample of the original sequence, as shown in Figure 7.5b. For dealing with temporal decimation, the traditional strategy is to perform a temporal interpolation filtering on the discrete STFT in order to restore the temporal decimation factor to unity. The FBS method is then performed on the interpolated output.

The FBS method is motivated by the relation between a sequence and its discrete-time STFT given in Equation (7.8), derived in the previous section for showing the invertibility of the discrete-time STFT. The FBS method carries out a discretized version of the operations suggested by this relation. That is, given a discrete STFT, $X(n, k)$, the FBS method synthesizes a sequence $y[n]$ satisfying the following equation:

$$y[n] = \frac{1}{Nw[0]} \sum_{k=0}^{N-1} X(n, k) e^{j\frac{2\pi}{N}nk}. \tag{7.9}$$

We are interested in deriving conditions for the FBS method such that the sequence $y[n]$ in Equation (7.9) is the same as the sequence $x[n]$. We can conjecture conditions by viewing the discretized equation from the filter bank viewpoint of Figure 7.5. We first see that the modulation resulting from the complex exponentials in analysis is undone by the demodulation of the complex exponentials at the filter bank outputs in synthesis. Therefore, for $y[n] = x[n]$ the sum of the filter bank outputs must add to unity. We now show this condition more formally.

Substituting $X(n, k)$ in Equation (7.9) for the FBS method, we obtain

$$y[n] = \frac{1}{Nw[0]} \sum_{k=0}^{N-1} \underbrace{\left[\sum_{m=-\infty}^{\infty} x[m]w[n-m]e^{-j2\pi km/N} \right]}_{X(n,k)} e^{j\frac{2\pi}{N}nk}. \tag{7.10}$$

Using the linear filtering interpretation of the STFT [the discrete version of Equation (7.6)], this equation reduces to

$$y[n] = \frac{1}{Nw[0]} x[n] * \sum_{k=0}^{N-1} w[n] e^{j\frac{2\pi}{N}nk}.$$

Taking $w[n]$ out of the summation and noting that the finite sum over the complex exponentials reduces to an impulse train with period N, we obtain

$$y[n] = \frac{1}{w[0]} x[n] * w[n] \sum_{r=-\infty}^{\infty} \delta[n - rN].$$

In the above expression, we note that $y[n]$ is obtained by convolving $x[n]$ with a sequence that is the product of the analysis window with a periodic impulse train. It follows that if we desire $y[n] = x[n]$, then the product of $w[n]$ and the periodic impulse train must reduce to $w[0]\delta[n]$. That is

$$w[n] \sum_{r=-\infty}^{\infty} \delta[n - rN] = w[0]\delta[n]. \tag{7.11}$$

This is satisfied for any causal analysis window whose length N_w is less than or equal to the number of analysis filters N, i.e., *any* finite-length analysis window can be used in the FBS

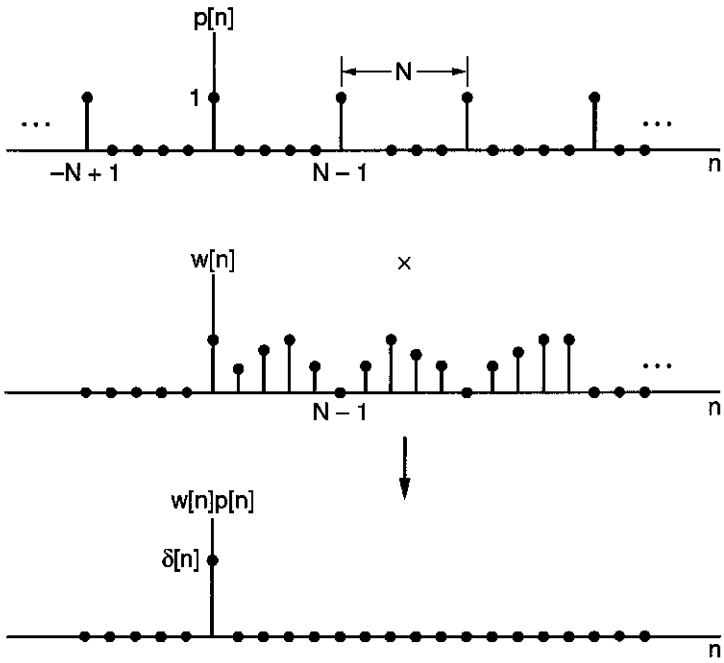

Figure 7.11 Example of an analysis window and how it satisfies the FBS constraint. The analysis-window length is longer than the frequency sampling factor. The sequence $p[n] = \sum_{r=-\infty}^{\infty} \delta[n, n-r]$.

SOURCE: S.H. Nawab and T.F. Quatieri, "Short-Time Fourier Transform" [13]. ©1987, Pearson Education, Inc. Used by permission.

method provided the length of the window is less than the frequency sampling factor N. We can even have $N_w > N$, provided $w[n]$ is chosen such that every Nth sample is zero, i.e.,

$$w[rN] = 0 \quad \text{for } r = -1, 1, -2, 2, -3, 3, \cdots \tag{7.12}$$

as illustrated in Figure 7.11.

Equation (7.12) is known as the *FBS constraint* because this is the requirement on the analysis window that ensures exact signal synthesis with the FBS method. This constraint is more commonly expressed in the frequency domain. Taking the Fourier transform of both sides of Equation (7.11), we obtain (Exercise 7.3)

$$\sum_{k=0}^{N-1} W\left(\omega - \frac{2\pi}{N}k\right) = Nw[0]. \tag{7.13}$$

This constraint essentially states that the frequency responses of the analysis filters should sum to a constant across the entire bandwidth. We have already seen that any finite-length analysis window whose length is less than or equal to the frequency sampling factor N satisfies this constraint. We conclude that a filter bank with N filters, based on an analysis filter of a length less than or equal to N, is *always* an all-pass system. This is not surprising because, from

the synthesis equation $x[n] = \frac{1}{2\pi w[0]} \int_{-\pi}^{\pi} X(n, \omega) e^{j\omega n} d\omega$, at time n the inverse DFT of $X(n, k)/Nw[0]$ must give $x[n]$ when the DFT length is longer than the sequence.

A generalization of the FBS method is motivated by the following relation between a sequence and its discrete-time STFT:

$$x[n] = \frac{1}{2\pi} \int_{-\pi}^{\pi} \left[\sum_{r=-\infty}^{\infty} f[n, n - r] X(r, \omega) \right] e^{j\omega n} d\omega, \tag{7.14}$$

where the "smoothing" function $f[n, m]$ is referred to as the time-varying *synthesis filter*. It can be shown that any $f[n, m]$ that satisfies

$$\sum_{m=-\infty}^{\infty} f[n, -m] w[m] = 1 \tag{7.15}$$

makes Equation (7.14) a valid synthesis equation (Exercise 7.6). It should be noted that the motivating equation for the basic FBS method can be obtained by setting the synthesis filter to be a non-smoothing filter, i.e., $f[n, m] = \delta[m]$ in Equation (7.14) (Exercise 7.6).

The *generalized FBS* method carries out a discretized version of the operations suggested by Equation (7.14). That is, given a discrete STFT which is decimated in time by a factor L, the generalized FBS method synthesizes a sequence $y[n]$ satisfying the following equation:

$$y[n] = \frac{L}{N} \sum_{r=-\infty}^{\infty} \sum_{k=0}^{N-1} f[n, n - rL] X(rL, k) e^{-j\frac{2\pi}{N}nk}. \tag{7.16}$$

Although Equation (7.16) contains the time-varying synthesis filter $f[n, n - rL]$, we consider here the time-invariant case, $f[n, n - rL] = f[n - rL]$, because of its practical importance in interpolating a time-decimated STFT. For a time-invariant synthesis filter, Equation (7.16) reduces to

$$y[n] = \frac{L}{N} \sum_{r=-\infty}^{\infty} \sum_{k=0}^{N-1} f[n - rL] X(rL, k) e^{-j\frac{2\pi}{N}nk}. \tag{7.17}$$

This equation holds when the following constraint is satisfied by the analysis and synthesis filters as well as the temporal decimation and frequency sampling factors [13]:

$$L \sum_{r=-\infty}^{\infty} f[n - rL] w[rL - n + pN] = \delta[p], \quad \text{for all } n. \tag{7.18}$$

The constraint in Equation (7.18) reduces to the constraint Equation (7.13) for the basic FBS method when the synthesis filter for the generalized FBS method is $f[n, m] = \delta[m]$ and $L = 1$ [13]. Finally, it should also be noted that if $L \neq 1$, and if we let $f[n]$ be the interpolating filter preceding the FBS synthesis, then the synthesis of Equation (7.17) with the synthesis filter $f[n]$ is equivalent to the entire process of interpolation and filter bank summation. Moreover, there exist fast computational methods based on this generalized FBS formulation. Some of these methods such as *helical interpolation* introduced by Portnoff and the *weighted overlap-add method* introduced by Crochiere are described in [4],[13],[18],[20].

7.3.3 Overlap-Add (OLA) Method

Just as the FBS method was motivated from the filtering view of the STFT, the OLA method is motivated from the Fourier transform view of the STFT [1],[13],[20]. The simplest method obtainable from the Fourier transform view is, in fact, not the OLA method. It is instead a method known as the Inverse Discrete Fourier Transform (IDFT) method. In this method, for each fixed time, we take the inverse DFT of the corresponding frequency function and divide the result by the analysis window. This method is generally not favored in practical applications because a slight perturbation in the STFT can result in a synthesized signal very different from the original.[3] For example, consider the case where the STFT is multiplied by a linear phase factor of the form $e^{j\omega n_o}$ with n_o unknown. Then the IDFT for each fixed time results in a shifted version of the corresponding short-time section. Since the shift n_o is unknown, dividing by the analysis window without taking the shift into account, introduces a distortion in the resulting synthesized signal. In contrast, the OLA method, which we describe in this section, results in a shifted version of the original signal without distortion.

In the OLA method, we take the inverse DFT for each fixed time in the discrete STFT. However, instead of dividing out the analysis window from each of the resulting short-time sections, we perform an overlap and add operation between the short-time sections. This method works provided the analysis window is designed such that the overlap and add operation effectively eliminates the analysis window from the synthesized sequence. The intuition here is that the redundancy within overlapping segments and the averaging of the redundant samples remove the effect of windowing.

The OLA method is motivated by the following relation between a sequence and its discrete-time STFT:

$$x[n] = \frac{1}{2\pi W(0)} \int_{-\pi}^{\pi} \sum_{p=-\infty}^{\infty} X(p,\omega) e^{j\omega p} d\omega, \tag{7.19}$$

where

$$W(0) = \sum_{n=-\infty}^{\infty} w[n].$$

This synthesis equation can be thought of as the synthesis equation in Equation (7.8), averaged over many short-time segments and normalized by $W(0)$. For the derivation of this synthesis equation, the reader is referred to [13]. The OLA method carries out a discretized version of the operations suggested on the right of Equation (7.19). That is, given a discrete STFT $X(n,k)$, the OLA method synthesizes a sequence $y[n]$ given by

$$y[n] = \frac{1}{W(0)} \sum_{p=-\infty}^{\infty} \left[\frac{1}{N} \sum_{k=0}^{N-1} X(p,k) e^{j\frac{2\pi}{N}kn} \right].$$

The term inside the rectangular brackets is an inverse DFT which for each p gives us

$$f_p[n] = x[n]w[p-n],$$

[3] One might consider averaging out this effect by summing many inverted DFTs, with the analysis window divided out, at each time n. This is in fact the strategy of the OLA method, but without the need to divide out each IDFT by the window shape.

provided that the DFT length N is longer than the window length N_w, i.e., there is no aliasing from the DFT inversion [14]. [Observe also that the inverse DFT is implicitly zero outside the interval $[0, N)$.] The expression for $y[n]$ therefore becomes

$$y[n] = \frac{1}{W(0)} \sum_{p=-\infty}^{\infty} x[n]w[p - n],$$

which can be rewritten as

$$y[n] = x[n]\left(\frac{1}{W(0)}\right) \sum_{p=-\infty}^{\infty} w[p - n].$$

In the above expression, we note that $y[n]$ is equal to $x[n]$ provided

$$\sum_{p=-\infty}^{\infty} w[p - n] = W(0),$$

which we observe is *always true because the sum of values of a sequence must always equal the first value of its Fourier transform.* Furthermore, if the discrete STFT had been decimated in time by a factor L, it can be similarly shown that if the analysis window satisfies (Exercise 7.4)

$$\sum_{p=-\infty}^{\infty} w[pL - n] = \frac{W(0)}{L}, \tag{7.20}$$

then $x[n]$ can be synthesized using the following relation:

$$x[n] = \frac{L}{W(0)} \sum_{p=-\infty}^{\infty} \left[\frac{1}{N} \sum_{k=0}^{N-1} X(pL, k)e^{j\frac{2\pi}{N}kn}\right]. \tag{7.21}$$

Equation (7.20) is the general constraint imposed by the OLA method on the analysis window. Unlike the case for $L = 1$, this constraint does not hold for any arbitrary window. It requires the sum of all the analysis windows (obtained by sliding $w[n]$ with L-point increments) to add up to a constant, as shown in Figure 7.12. It is interesting to note the duality between this constraint and the FBS constraint in Equation (7.13), where the shifted versions of the Fourier transform of the analysis window were required to add up to a constant. For the FBS method, we also saw that all finite-length analysis windows whose length N_w is less than the number of analysis filters N satisfy the FBS constraint. Analogously, we can show that the OLA constraint in Equation (7.20) is satisfied by all finite-bandwidth analysis windows whose maximum frequency is less than $\frac{2\pi}{L}$, where L is the temporal decimation factor. In addition, this finite-bandwidth constraint can be relaxed by allowing the shifted window transform replicas to take on value zero at the frequency origin $\omega = 0$. In particular, if $W(\omega - \frac{2\pi}{L}k)$ has zeros at $\omega = \frac{2\pi}{L}k$, then the OLA constraint holds. This is remarkably analogous to the relaxation of the FBS constraint $N_w < N$ by allowing the window $w[n]$ to take on value zero at $n = \pm N, \pm 2N, \ldots$. A proof of this frequency-domain view of the OLA constraint is given in [13].

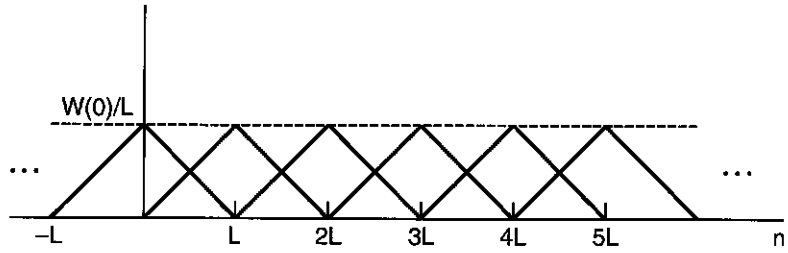

Figure 7.12 The OLA constraint visualized in the time domain.

SOURCE: S.H. Nawab and T.F. Quatieri, "Short-Time Fourier Transform" [13]. ©1987, Pearson Education, Inc. Used by permission.

The duality[4] between the FBS and OLA constraints is summarized in Figure 7.13. The FBS method depends on a sampling relation in frequency while the OLA method depends on a sampling relation in time. With the FBS method, each time sample is obtained by summing filter outputs, while with the OLA method each time sample is obtained by summing different short-time sections. The OLA constraint with no time decimation always holds, provided, of course, that the DFT length is longer than the window length. This is true because the sum of the window's samples is equal to its first Fourier transform value. However, with time decimation, the OLA constraint does not always hold. We saw one situation where it does hold:

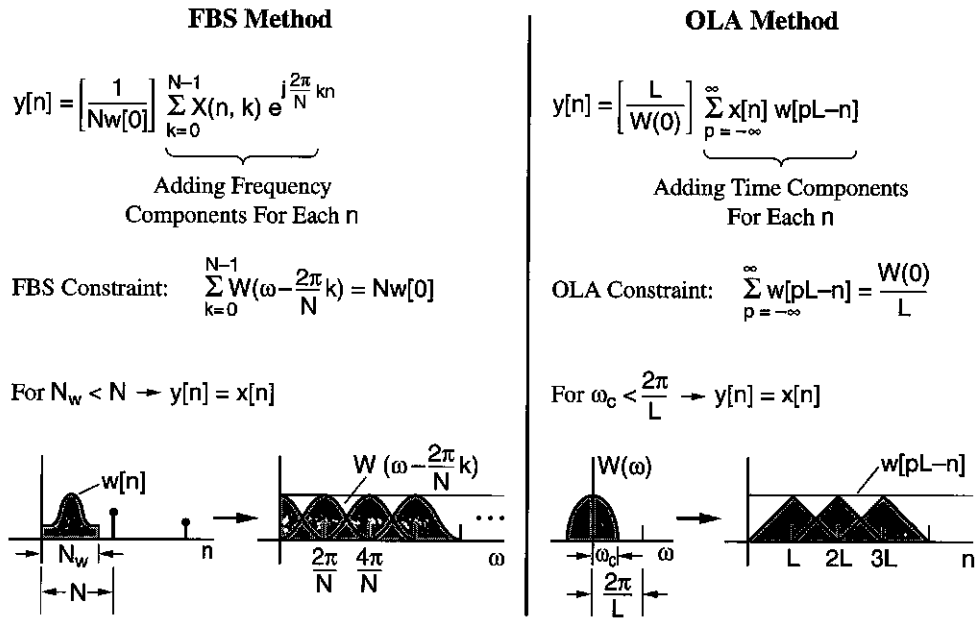

Figure 7.13 Duality between the FBS and OLA constraints. Relaxation of constraints to allow zero crossings in time and in frequency is not shown.

[4] Observe that the duality is not exact because a finite bandwidth window implies an infinite window duration. Nevertheless, we assume the effective duration is finite.

The window bandwidth is less than $\frac{2\pi}{L}$. We can relax this constraint if the shifted window transforms pass through zero at the frequency origin; in particular, if the window transform equals zero at $\omega = \frac{2\pi}{L}k$, the constraint holds. This is analogous to relaxing the FBS constraint that the window length N_w is less than the frequency sampling factor N by letting the analysis window pass through zero at $n = kN$.

7.3.4 Time-Frequency Sampling

We now give a different qualitative discussion of the above time-frequency sampling concepts for the OLA and FBS constraints from the perspective of classical time- and frequency-domain aliasing [20]. This discussion also serves to further summarize the sampling issues for these methods, and gives motivation for our earlier statement that sufficient but not necessary conditions for invertibility of the discrete STFT are that the analysis window is non-zero over its finite length N_w, the temporal decimation factor $L \leq N_w$, and the frequency sampling interval $\frac{2\pi}{N} \leq \frac{2\pi}{N_w}$.

Consider a short-time segment $f_n[m] = w[m]x[n-m]$ and its Fourier transform $X(n, \omega)$ with the analysis window of duration N_w. From the Fourier transform view, recovering time sequence $f_n[m]$ (with respect to the variable m) by an inverse DFT of the discrete STFT $X(n, k)$ requires a frequency sampling interval of $\frac{2\pi}{N_w}$ or finer to avoid time-domain aliasing of the time segment $f_n[m]$. Consider now a time decimation factor L. From the filtering view of the STFT, recovering the time sequence $X(n, k)$ (with respect to the time variable n) requires that this time sampling interval L meets the Nyquist criterion based on the bandwidth of the "filter" $w[n]$. This implies that we sample $X(n, k)$ at a time interval $L \leq \frac{2\pi}{\omega_c}$, where ω_c is the filter bandwidth (i.e., $W(\omega)$ is zero outside $[-\omega_c, \omega_c]$ within the interval $[-\pi, \pi]$), to avoid frequency-domain aliasing of the time sequence $X(n, \omega)$. This time-frequency sampling is illustrated in Figure 7.14. Selecting a Hamming window, typically used in speech analysis, and defining the bandwidth with respect to the 3 dB attenuation points, we find that the above sampling requirements, over all N filters in our filter bank, imply four times the number of samples in the original representation of the sequence $x[n]$ [20].

The above time-frequency sampling constraints, derived from simple aliasing considerations, are consistent with the OLA constraint (filter bandwidth $\omega_c \leq \frac{2\pi}{L}$, L being the time decimation factor) and the FBS constraint (window duration $N_w \leq N$, $\frac{2\pi}{N}$ being the frequency sampling interval). These window length and bandwidth constraints can, however, be relaxed, as we have seen in the OLA and FBA constraints, by allowing zeros in the window or its transform at the appropriate time or frequency points, respectively. This implies that we can avoid the four-fold increase in sampling requirements in the above example with a Hamming window analysis. We return to this issue later in Chapter 12 in our discussion of time-frequency analysis with application to speech coding.

Integrating our discussion of aliasing with the OLA and FBS methods, we summarize the following time-frequency constraint considerations from the perspective of each method.

OLA Method

1. For a window length N_w and with a DFT length chosen to give sufficient frequency sampling, i.e., a frequency sampling factor less than $\frac{2\pi}{N_w}$, then each short-time segment can be recovered because there is no time-domain aliasing.

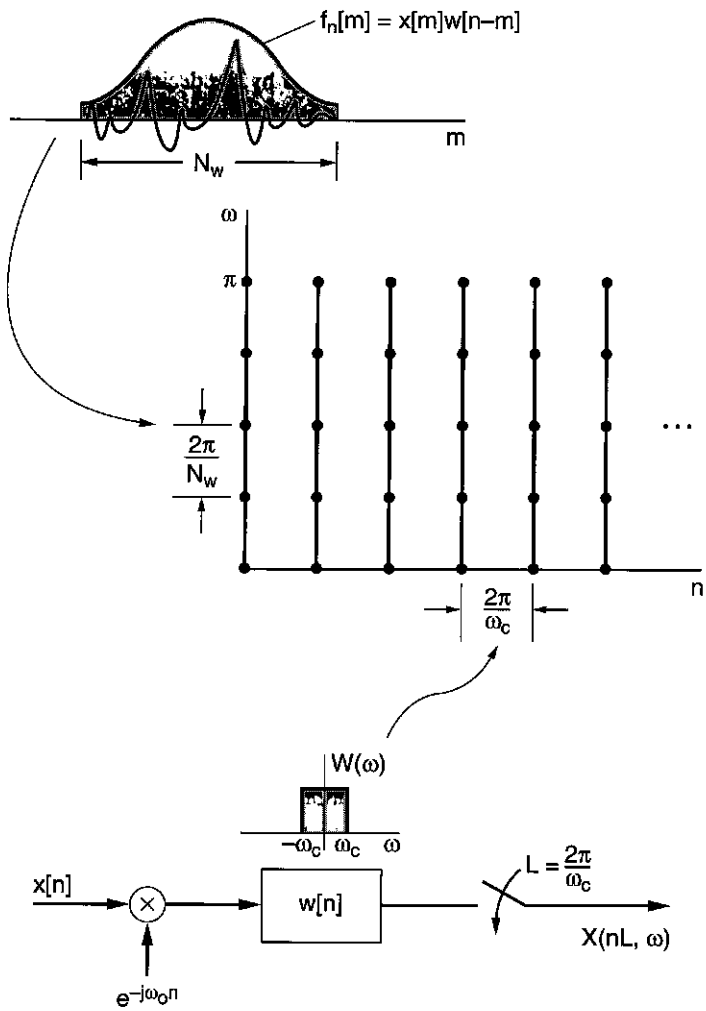

Figure 7.14 Time-frequency sampling constraints from the perspective of classical time- and frequency-domain aliasing. The time sampling must satisfy the Nyquist criterion to avoid aliasing in frequency (but the OLA constraint allows relaxing the finite filter bandwidth constraint), while the frequency sampling must be fine enough to avoid aliasing in time (but the FBS constraint allows relaxing the finite window duration condition).

2. The time decimation factor L is fine enough if the filter bandwidth $\omega_c \leq \frac{2\pi}{L}$. This results in no frequency-domain distortion from the window transform and is the strict form of the OLA constraint.

3. We can relax (2) if we allow zeros in the window transform. In this case, we can undersample in time and still recover $x[n]$.

FBS Method

1. With time decimation of filter outputs by the decimation factor L, we assume that the time sampling meets the Nyquist criterion to recover each filter output, i.e., $L \leq \frac{2\pi}{\omega_c}$, so that frequency-domain aliasing does not occur.
2. The frequency sampling is sufficient if $\frac{2\pi}{N_w} \geq \frac{2\pi}{N}$, i.e., $N_w \leq N$, giving no time-domain distortion from the window (as indicated in Equation (7.11)). This is the strict form of the FBS constraint.
3. We can relax (2) if we allow zeros in the window. In this case, we can undersample in frequency and still recover $x[n]$.

7.4 Short-Time Fourier Transform Magnitude

In speech applications, the spectrogram, which is the squared STFT magnitude (STFTM), has played a major role. For example, visual cues in the spectrogram have been related to parameters important for speech perception. In fact, it has been suggested [2] that the human ear extracts perceptual information strictly from a spectrogram-like representation of speech. Alternatively, experienced speech researchers have trained themselves to "read" the spectrogram itself [23], which indicates that, at least on the phonetic level, the speech signal is largely retained in the spectrogram. One might question, then, how much "information" actually has been lost in the spectrogram, and whether a signal can be recovered from this time-frequency magnitude representation.

By removing a (possibly) unnecessary phase function (the STFT is a complex-valued function), a magnitude-only representation may have uses in a variety of applications, such as time-scale modification and enhancement of speech, where estimation of the phase of the STFT is difficult [16]. For example, as we will see in Chapter 13, the estimation of the phase of a signal's frequency response in the presence of noise is more difficult than the estimation of its magnitude. Indeed, a number of techniques have been developed that obtain a STFT phase estimate from a STFT magnitude estimate, thus circumventing the more difficult phase estimation problem. In this section we introduce the magnitude of the STFT as an alternative time-frequency signal representation. We shall see that many signals can be uniquely represented by the real-valued and non-negative STFTM. Furthermore, we develop analysis and synthesis techniques for the STFTM just as we did for the STFT and show that, while STFTM analysis is similar to STFT analysis, short-time synthesis is very different for the two transforms.

The STFTM is related to another function, the short-time autocorrelation, $r[n, m]$, through the following Fourier transform relationship:[5]

$$r[n, m] = \frac{1}{2\pi} \int_{-\pi}^{\pi} |X(n, \omega)|^2 e^{j\omega m} d\omega$$

$$|X(n, \omega)|^2 = \sum_{m=-\infty}^{\infty} r[n, m] e^{-j\omega m}$$

[5] We use here a slightly different notation than that in Chapter 5, where the short-time autocorrelation was written as $r_n[\tau]$.

where m is the autocorrelation "lag" introduced in Chapter 5. The autocorrelation $r[n, m]$ is given by the convolution of the short-time section $f_n[m] = x[m]w[m - n]$ with its time-reversed version, i.e.,

$$r[n, m] = f_n[m] * f_n[-m]$$

with "*" denoting convolution. Generally, the short-time section $f_n[m] = x[m]w[m - n]$ cannot be obtained from its short-time autocorrelation function [14]. Because $f_n[m]$ is of finite length, its z-transform consists of zeros that can lie inside or outside the unit circle. A conjugate reciprocal flip of a zero preserves the sequence's Fourier transform magnitude and thus its autocorrelation. That is, we saw in our review in Chapter 2 that applying the all-pass function

$$H_{ap}(z) = \frac{z^{-1} - a^*}{1 - az^{-1}}$$

flips the zero at $z = \frac{1}{a^*}$ to its conjugate reciprocal location $z = a$ without a change in the magnitude of the frequency response because $|H_{ap}(z)| = 1$. However, as we shall see shortly, the autocorrelations of short-time sections that have partial overlap in time can be used jointly to solve for the underlying short-time sections, thus removing this inherent ambiguity. This will enable us, under certain conditions, to use the STFTM as a unique representation of the underlying signal.

7.4.1 Signal Representation

We now consider the problem of determining when the discrete-time STFTM can be used to represent a sequence uniquely. After demonstrating uniqueness, we then proceed to determine an algorithm for sequence recovery from the STFTM. That the STFTM is not a unique representation in all cases is easily seen from the simple observation that $x[n]$ and its negative, $-x[n]$, have the same STFTM. We will also demonstrate that there are other kinds of situations where the STFTM is not a unique signal representation. We will then proceed to show that, by imposing certain mild restrictions on the analysis window and the signal, unique signal representation is indeed possible with the discrete-time STFTM [13].

To develop insight into the kinds of situations where a sequence cannot be represented uniquely by its discrete-time STFTM, let us consider the case of a sequence $x[n]$ with a gap of zero samples between two non-zero portions. That is, suppose $x[n]$ is the sum of two signals, $x_1[n]$ and $x_2[n]$, occupying different regions of the n-axis, as depicted in Figure 7.15 (upper). Suppose further that the gap of zeros between $x_1[n]$ and $x_2[n]$ is large enough so that there is no analysis window position for which the corresponding short-time section includes non-zero contribution from $x_1[n]$ as well as $x_2[n]$. Clearly, in such a situation the STFTM of $x[n]$ is the sum of the STFT magnitudes of $x_1[n]$ and $x_2[n]$. However, we have previously observed that a signal and its negative have the same STFTM. It follows that $x[n]$ has the same STFTM as the signals obtained from the differences $x_1[n] - x_2[n]$ and $x_2[n] - x_1[n]$, shown in Figure 7.15 (middle and bottom). We conclude that if there is a large enough gap of zero samples, there will be sign ambiguities on either side of the gap. Consequently, any uniqueness conditions must include a restriction on the length of the zero gaps between non-zero portions of the signal $x[n]$. In particular, the sufficient uniqueness conditions we show are the following:

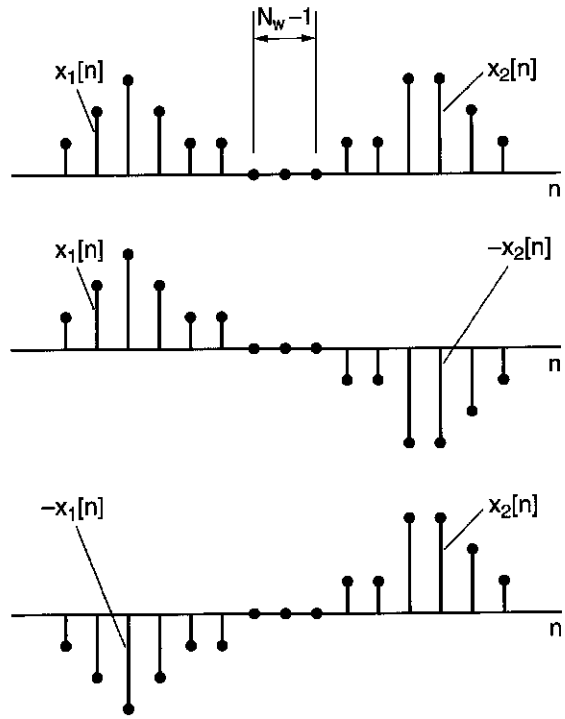

Figure 7.15 Three sequences with the same STFTM.

SOURCE: S.H. Nawab and T.F. Quatieri, "Short-Time Fourier Transform" [13]. ©1987, Pearson Education, Inc. Used by permission.

1. The analysis window $w[n]$ is a known sequence of finite length N_w, with no zeros over its duration.
2. The sequence $x[n]$ is one-sided with at most $N_w - 2$ consecutive zero samples, and the sign of its first non-zero value is known.

The key to showing the uniqueness of $x[n]$ under the above conditions is the observation that $|X(n, \omega)|$ has additional information about the short-time sections of $x[n]$ besides their spectral magnitudes. This information is contained in the overlap of the analysis window positions, i.e., there is much redundancy between adjacent segments. If the short-time section at time n is known, then the signal corresponding to the spectral magnitude of the adjacent section at time $n + 1$ must be *consistent* in the region of overlap with the known short-time section. By consistent, we mean that if the analysis window were non-zero and of length N_w, then after dividing out the analysis window, the first $N_w - 1$ samples of the segment at time $n + 1$ must equal the last $N_w - 1$ samples of the segment at time n, as illustrated in Figure 7.16. Therefore, if we could *extrapolate* the last sample of a segment from its first $N_w - 1$ values, we could repeat this process to obtain the entire signal $x[n]$.

To develop the procedure for extrapolating the next sample of a sequence using its STFTM, assume that the sequence $x[n]$ has been obtained up to some time $n - 1$. Thus, the first $N_w - 1$ samples under the analysis window positioned at time n are known. The goal is to compute the

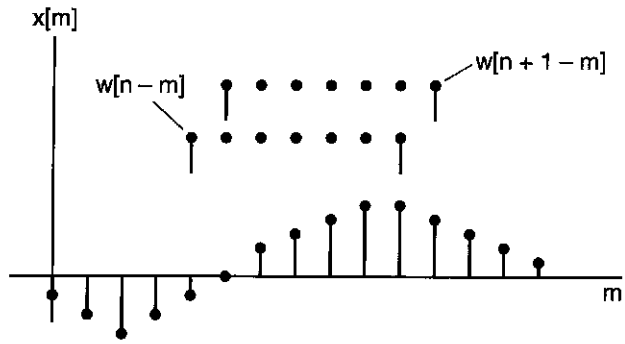

Figure 7.16 Illustration of the consistency required among adjacent short-time sections. Note the samples that are common to the adjacent sections. A rectangular analysis window is assumed.

SOURCE: S.H. Nawab and T.F. Quatieri, "Short-Time Fourier Transform" [13]. ©1987, Pearson Education, Inc. Used by permission.

sample $x[n]$ from these initial samples and the STFT magnitude $|X(n, \omega)|$ or, equivalently, $r[n, m]$. Note that the last value of the short-time autocorrelation function, $r[n, N_w - 1]$, is given by the product of the first and last value of the segment:

$$r[n, N_w - 1] = (w[0]x[n])(w[N_w - 1]x[n - (N_w - 1)])$$

as illustrated in Figure 7.17. Therefore, $x[n]$ is given by

$$x[n] = \frac{r[n, N_w - 1]}{w[0]w[N_w - 1]x[n - (N_w - 1)]}. \tag{7.22}$$

If the first value of the short-time section, i.e., $x[n - (N_w - 1)]$, happens to equal zero, we must then find the first non-zero value within the section and again use the product relation given by Equation (7.22). We can always find such a sample because we have assumed at most

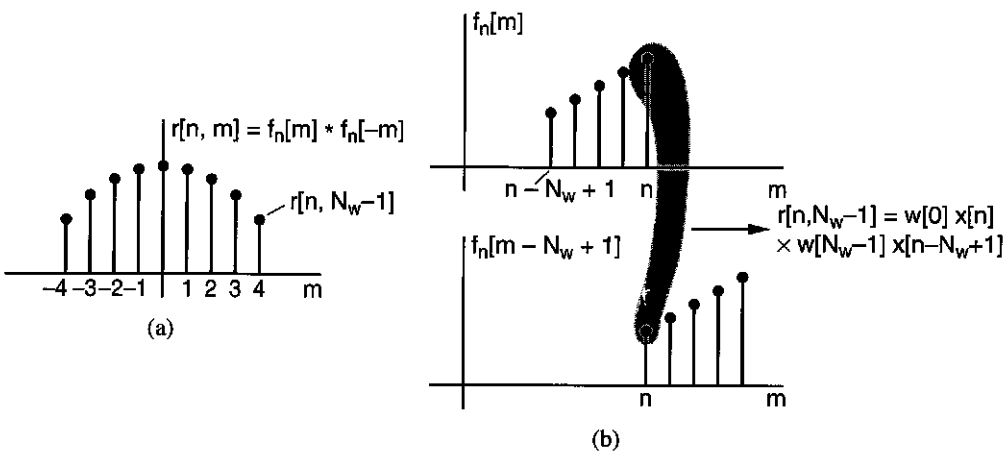

Figure 7.17 Computation of the last non-zero autocorrelation sample of a five-point sequence: (a) autocorrelation function; (b) product of first and last sequence values.

$N_w - 2$ consecutive zero samples between any two non-zero samples of $x[n]$. This completes our argument for the sufficiency of our two conditions for uniquely representing $x[n]$ with $|X(n, \omega)|$. We will see in the next section that the uniqueness conditions still hold with at least N_w uniformly spaced samples of $|X(n, \omega)|$ in frequency over $[0, \pi]$, i.e., if a $(2N_w - 1)$-point (or greater) DFT is computed. The above argument for uniqueness, therefore, leads to the reconstruction algorithm:

S1: Initialize with $x[0]$.

S2: Update time n.

S3: Compute $r[n, N_w - 1]$ from the inverse DFT of $|X(n, k)|^2$.

S4: Compute $x[n] = \frac{r[n, N_w-1]}{w[0]w[N_w-1]x[n-(N_w-1)]}$.

S5: Return to Step (S2) and repeat.

We refer to this as a *sequential extrapolation* algorithm.

7.4.2 Reconstruction from Time-Frequency Samples

In order to carry out STFTM analysis on a digital computer, we need to introduce the *discrete* STFTM. By sampling the frequency dimension of the STFTM, $|X(n, \omega)|$, we obtain the discrete STFTM, which is defined as $|X(n, k)|$, the magnitude of the discrete STFT. In the last section, we saw that, under certain conditions, the discrete-time STFTM is a unique signal representation. We noted that the theory can be easily extended to the discrete STFTM. In particular, the uniqueness conditions of the previous section relied on using the short-time autocorrelation functions of adjacent short-time sections which are overlapping in time. These autocorrelation functions can be obtained even if the STFTM is sampled in frequency. That is, if the analysis window is N_w points long, then each short-time autocorrelation function is, at most, $2N_w - 1$ points long and thus can be obtained (without aliasing) from $2N_w - 1$ frequency samples of the STFTM (Exercise 7.7). Therefore, the uniqueness conditions of the discrete-time STFTM extend without change to the discrete STFTM with adequate frequency sampling.

To consider the effects of temporal decimation with factor L, we note that adjacent short-time sections now have an overlap of $N_w - L$ instead of $N_w - 1$. The successive extrapolation procedure discussed in the previous section can be extended to this case by requiring the extrapolation of the L last samples of a short-time section, using the first $N_w - L$ samples and the short-time autocorrelation function of that section (Exercise 7.7). This can be accomplished provided the overlap between adjacent short-time sections is greater than $\frac{N_w}{2}$ and there are no zero-gaps of length greater than $N_w - 2L$. In addition, to initialize the extrapolation procedure, L initial samples of the underlying sequence must be known. We summarize the sufficient uniqueness conditions for the partial overlap case as follows:

1. The analysis window $w[n]$ is a known sequence of finite length N_w, with no zeros over its duration.
2. The sequence $x[n]$ is one-sided with, at most, $N_w - 2L$ consecutive zero samples. L consecutive samples of $x[n]$ (from the first non-zero sample) are known. This is a sufficient but not a necessary condition.

As with the $L = 1$ case, a sequential extrapolation algorithm can be implemented for the more general case. For synthesizing a sequence $x[n]$ from $|X(nL, k)|$ under the above

conditions, we assume that the first non-zero sample of $x[n]$ falls at $n = 0$ and that the L samples of $x[n]$ for $0 \leq n < L$ are known. The L known samples of $x[n]$ completely determine the short-time section corresponding to $|X(nL, k)|$ for $n = 1$. The short-time section corresponding to $|X(nL, k)|$ for $n = 2$ can then be extrapolated from its DFT magnitude and its known samples in the region of overlap with the previously determined short-time section. This process continues as the complete extrapolation of each new short-time section makes possible the extrapolation of the next overlapping short-time section. Although valid in theory, this algorithm has computational difficulty as L increases. For L greater than about four samples, accuracy of the reconstruction degrades rapidly in time because of a computational round-off error as with, for example, quantization error in the FFT; the sequential nature of the algorithm causes the effect of this error to accumulate over time [12]. Thus, more robust reconstruction algorithms are needed [6],[12]. We discuss one such algorithm in Section 7.5.2.

There are also a variety of other uniqueness conditions that express the tradeoff between time decimation and frequency sampling [7],[12],[19]. For example, for a time-decimation factor of $L = 1$, it can be shown that, under certain mild conditions, at each time instant two samples of $|X(n, \omega)|$ in frequency, but not necessarily at the same samples, ω_1 and ω_2, are sufficient to recover $x[n]$ [19]. More generally, $2L$ frequency samples are required for an L-sample time decimation [7]. A number of these conditions are derived in Exercises 7.8 and 7.10.

7.5 Signal Estimation from the Modified STFT or STFTM

In many applications, it is desired to synthesize a signal from a time-frequency function consisting of a modified STFT or STFTM. Such modifications may arise due to quantization errors (e.g., from speech coding) or due to desired time-varying filtering in signal processing applications such as speech enhancement (e.g., noise reduction) and speech modification (e.g., change in articulation rate). In noise reduction, each frequency slice of the STFT is reduced in certain regions, while in changing articulation rate, we modify the STFT by discarding or adding spectral slices. We study such STFT modifications in later sections of this chapter and throughout the text.

An arbitrary function of time and frequency, however, does not necessarily represent the STFT or STFTM of a signal. This is because the definitions of these transforms impose a structure on their time and frequency variations. In particular, because of the overlap between short-time sections, adjacent short-time segments cannot have arbitrary variations. A necessary but not sufficient condition on these variations is that the short-time section corresponding to each time instant must lie within the duration of the corresponding analysis window. For example, the short-time section corresponding to $X(0, \omega)$ is given by $f_0[n] = x[n]w[-n]$ and, therefore, it must lie within the duration of $w[-n]$, as illustrated in the following example.

EXAMPLE 7.5 Consider modifying a valid $X(n, \omega)$ by inserting a zero gap where there is known to lie an unwanted interfering sinewave component. This modification is illustrated in Figure 7.18 with multiplication by the filter $H(n, \omega)$. Then the resulting $Y(n, \omega) = H(n, \omega) \times X(n, \omega)$, when inversed transformed at a specific time n to obtain the modified short-time sequence, denoted by $g_n[m]$, is non-zero beyond the extent of the original short-time segment $f_n[m] = x[m]w[n - m]$. ▲

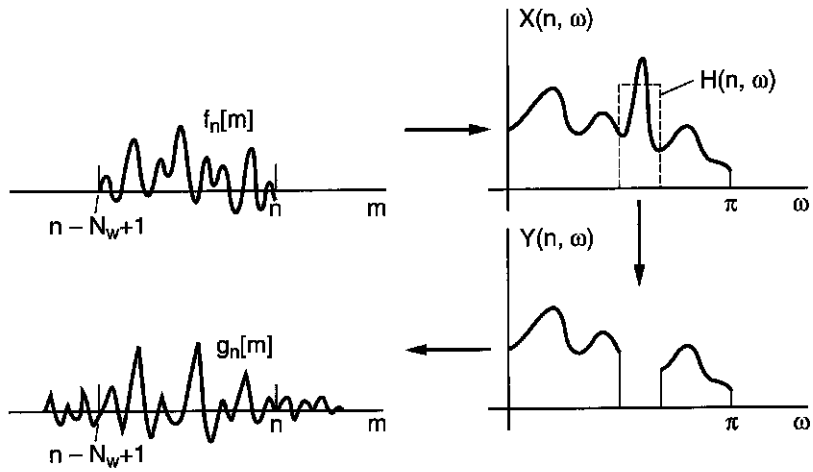

Figure 7.18 Schematic of violation of STFT duration constraint after modification. The original STFT $X(n, \omega)$ is modified by a filter $H(n, \omega)$ that removes an interfering sinewave component.

Another condition that the STFT or STFTM must satisfy is that adjacent short-time sections must be consistent in their region of overlap. When the STFT or STFTM of a signal is modified, the resulting time-frequency function does not generally satisfy such constraints. That is, if the short-time segment $f_n[m] = x[m]w[n - m]$ corresponds to $X(n, \omega)$, then, after modification, the inverse Fourier transform of the resulting STFT $Y(n, \omega)$, denoted by $g_n[m]$, is not always consistent with its adjacent short-segments, e.g., $g_{n+1}[m]$, because after dividing out the respective analysis window functions, the resulting sequences are not necessarily equal in the region of window overlap. Consider the following example.

EXAMPLE 7.6 Suppose a time-decimated STFT, $X(nL, \omega)$, is multiplied by a linear phase factor $e^{j\omega n_o}$ at time n to obtain $Y(nL, \omega) = X(nL, \omega)e^{jn_o\omega}$. Likewise, the following spectral slice $X((n + 1)L, \omega)$ is multiplied by the negative of this linear phase factor, i.e., $e^{-j\omega n_o}$, to obtain $Y((n + 1)L, \omega) = X((n + 1)L, \omega)e^{-jn_o\omega}$. Then, as illustrated in Figure 7.19, the inverse Fourier transforms, denoted by $g_{nL}[m]$ and $g_{(n+1)L}[m]$, respectively, are not consistent in their region of overlap. ▲

The synthesis methods we discussed in Sections 7.3 and 7.4 were derived with the assumption that the time-frequency functions to which they are applied satisfy the constraints in the definitions of the STFT or STFTM, i.e., the definitions of the STFT and STFTM, as we have just seen, impose structure on the time-frequency functions. Given a function which does not satisfy those constraints, the synthesis methods have no theoretical validity for their application. However, under certain conditions, those methods can be shown to yield "reasonable" results in the presence of modifications. For example, in Section 7.5.1 we illustrate conditions under which the FBS and OLA methods yield intuitively satisfying results when the STFT has been modified with a multiplicative factor. We can think of these methods as "heuristic" in that they blindly use the FBS, OLA, and signal extrapolation synthesis methods. In Section 7.5.2, on the other hand, we discuss a "rigorous" theoretically-based approach to signal synthesis from the modified STFT that invokes least-squared-error approximation methods. A similar approach is then discussed in Section 7.5.3 for signal estimation from the modified STFTM.

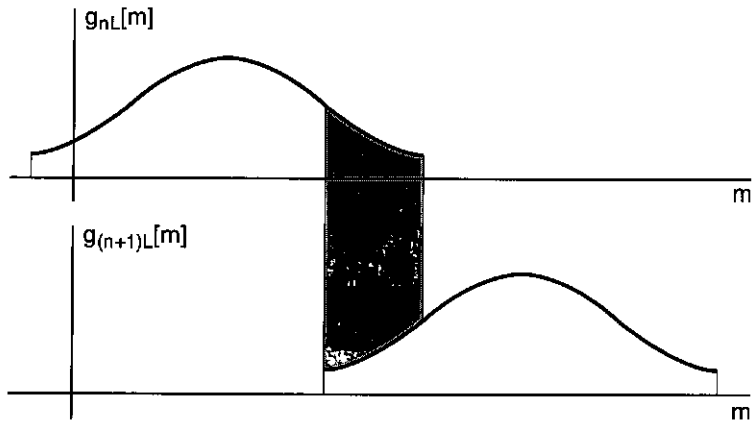

Figure 7.19 Consistency must be satisfied in adjacent short-time segments after modification for a valid STFT. This figure illustrates the violation of the consistency constraint with linear phase modification. After dividing out the window, the resulting sequences are not equal in their region of overlap.

7.5.1 Heuristic Application of STFT Synthesis Methods

Historically, signal estimation from the modified STFT has been performed by applying the FBS and OLA synthesis methods of Section 7.3 on time-frequency functions which are not valid STFT functions. Even though a modified STFT, $Y(n, \omega)$, is not a valid STFT, it is desirable that applying a synthesis method to $Y(n, \omega)$ should yield a "reasonable" result. Such heuristic use of the synthesis methods has been of practical importance in many signal processing applications. Since the synthesis methods have no theoretical basis for their application in such situations, it is common to analyze the effects that the methods have on the synthesized signal [1]. In this section, we contrast the FBS and OLA synthesis methods when they are applied to the STFT which has been modified through multiplication with another time-frequency function. For both the methods, the resulting synthesized signal can be shown to be a time-varying convolution between $x[n]$ and a function $\hat{h}[n, m]$.

Let us assume that the STFT $X(n, \omega)$ has been modified by a function $H(n, \omega)$ to give $Y(n, \omega)$, i.e.,

$$Y(n, \omega) = X(n, \omega)H(n, \omega).$$

This corresponds to a new short-time segment $g_n[m] = f_n[m] * h[n, m]$, where $h[n, m]$ can be thought of as a time-varying system impulse response (introduced in Chapter 2). We first investigate the use of the FBS synthesis method. In the FBS method, we discretize frequency to obtain

$$Y(n, k) = Y(n, \omega)|_{\omega=\frac{2\pi}{N}k} = X(n, k)H(n, k)$$

where $\frac{2\pi}{N}$ is the frequency sampling interval. Letting $\tilde{h}[n, m]$, for each n, represent the N-point inverse DFT of $H(n, k)$, we have $\tilde{h}[n, m] = \sum_{l=-\infty}^{\infty} h[n, m - lN]$. Keep in mind that although we have written $\tilde{h}[n, m]$ as periodic, we take only inverse DFT samples in the interval $[0, N - 1]$ [14]. For the FBS method, it can be shown that the resulting sequence

$y[n]$ can be written as

$$y[n] = \sum_{m=-\infty}^{\infty} x[n-m]\hat{h}[n,m] \tag{7.23}$$

where

$$\hat{h}[n,m] = w[m] \sum_{l=-\infty}^{\infty} h[n, m-lN]$$

which is equivalent to filtering $x[n]$ with the time-varying impulse response $\hat{h}[n,m]$. We see that the time-varying impulse response $\hat{h}[n,m]$ is obtained by multiplying $\tilde{h}[n,m]$, for each n, by the window $w[m]$ (Appendix 7.A and Exercise 7.11). Note that periodic replicas of the sequence $h[n,m]$ are used because of the discretized frequencies. We thus window $\sum_{l=-\infty}^{\infty} h[n, m-lN]$ and then convolve the resulting sequence $\hat{h}[n,m]$ at each time instant, n, with respect to the variable m.

Using the OLA synthesis method, it can be similarly shown that $y[n]$ can be obtained by convolving $x[n]$ with a time-varying impulse response as in Equation (7.23). For the OLA method, however, the time-varying impulse response $\hat{h}[n,m]$ is given by (Exercise 7.11)

$$\hat{h}[n,m] = w[n] * \sum_{l=-\infty}^{\infty} h[n, m-lN] \tag{7.24}$$

which, for the mth value of impulse response $\hat{h}[n,m]$, is the convolution of $h[n,m]$ with $w[n]$ with respect to the variable n. Therefore, each coefficient of the time-varying response is *smoothed* (along the time variable n) by the window $w[n]$ that typically has a lowpass frequency response [20]. For the FBS method, on the other hand, each time-varying response $h[n,m]$ is multiplied by the window $w[m]$ (along the time variable m). An important implication is that the FBS method allows instantaneous modification, while the OLA method restricts the modification to the bandwidth of the window [20]. It is interesting to note that if $h[n,m]$ is independent of n, i.e., $h[n,m] = h[m]$, and so represents a time-invariant impulse response, then the FBS method results in a synthesized signal which is the convolution of $x[n]$ with $h[n]w[n]$. On the other hand, the time-invariant case for the OLA method results in a synthesized signal which is the convolution of $x[n]$ with $h[n]$. The following example illustrates the OLA approach for this case.

EXAMPLE 7.7 Suppose we want to deliberately introduce reverberation into a signal $x[n]$ by convolution with the filter

$$h[n] = \delta[n] + \alpha\delta[n - n_o]$$

which has as its Fourier transform

$$H(\omega) = 1 + \alpha e^{-j\omega n_o}.$$

(Reverberation is often used as a special effect in speech and music modification.) But we will introduce reverberation through time-invariant modification of the STFT of $x[n]$:

$$Y(n,\omega) = X(n,\omega)H(\omega)$$

where

$$X(n,\omega) = \sum_{m=-\infty}^{\infty} x[m]w[n-m]e^{-j\omega m}.$$

Suppose we use the OLA method, i.e.,

$$y[n] = \frac{1}{W(0)} \sum_{p=-\infty}^{\infty} \left[\frac{1}{N} \sum_{k=0}^{N-1} Y(p,k) e^{j\frac{2\pi}{N}kn} \right].$$

It is then possible to derive an expression for $y[n]$ in terms of the original sequence $x[n]$ and the filter $h[n]$ by writing

$$y[n] = \frac{1}{W(0)} \sum_{m=-\infty}^{\infty} x[m] \underbrace{\left[\frac{1}{N} \sum_{k=0}^{N-1} H(k) e^{j\frac{2\pi}{N}k(n-m)} \right]}_{\text{IDFT} \to \sum_{r=-\infty}^{\infty} h[n-m+rN]} \underbrace{\left[\sum_{p=-\infty}^{\infty} w[p-m] \right]}_{W(0)}$$

$$= \sum_{m=-\infty}^{\infty} x[m] \hat{h}[n-m]$$

where IDFT denotes the inverse DFT, and where for $h[n]$, we have the modified echo-generating function

$$\hat{h}[n] = \sum_{r=-\infty}^{\infty} h[n+rN]$$

$$= \sum_{r=-\infty}^{\infty} (\delta[n+rN] + \alpha\delta[n-n_0+rN])$$

from which we take values only in the interval $[0, N-1]$ because of the inverse DFT operation (represented in Figure 7.20 by the sequence $R[n]$ which is a rectangular function, non-zero in the interval $[0, N-1]$). Therefore, STFT-based filtering with OLA synthesis does indeed yield the desired reverberation when $n_o < N$, as seen in Figure 7.20. ▲

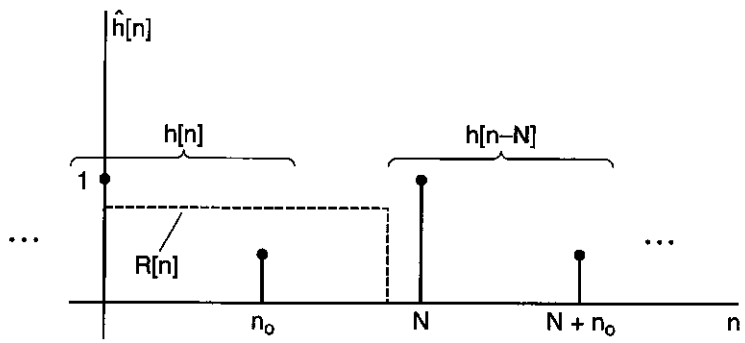

Figure 7.20 Illustration of the echo-generating function $\hat{h}[n]$ that results from the OLA method in Example 7.7. $R[n]$ is the rectangular function, non-zero in the interval $[0, N-1]$, invoked by the inverse DFT.

In this section we have seen how the effects of applying the FBS and OLA methods to the modified STFT may be analyzed for the case of multiplicative modifications. A similar analysis can also be carried out for situations where a time-frequency function has been added to a valid STFT [1].

7.5.2 Least-Squared-Error Signal Estimation from the Modified STFT

Rather than applying the FBS and OLA methods in a heuristic manner, we now consider a different approach that is specifically designed for signal estimation from the modified STFT. In this approach, we estimate a signal whose STFT is closest in some sense to the modified STFT (Figure 7.21)[6] [6]. More specifically, we want to minimize the mean-squared-error between the discrete-time STFT, $X_e(n, \omega)$, of the signal estimate, $x_e[n]$:

$$X_e(n, \omega) = \sum_{m=-\infty}^{\infty} w[n - m]x_e[m]e^{-j\omega m}$$

and the modified discrete-time STFT, which we denote by $Y(n, \omega)$. This optimization results in the following solution for the estimated signal $x_e[n]$ [6]:

$$x_e[n] = \frac{\sum_{m=-\infty}^{\infty} w[m - n]f_m[n]}{\sum_{m=-\infty}^{\infty} w^2[m - n]} \tag{7.25}$$

where $f_m[n]$ is the inverse Fourier transform of the short-time segment at time m, corresponding to the modified STFT, $Y(m, \omega)$. The specific distance measure used in the minimization is the squared error between $X_e(n, \omega)$ and $Y(n, \omega)$ integrated over all ω and summed over all n:

$$D[X_e(n, \omega), Y(m, \omega)] = \sum_{m=-\infty}^{\infty} \frac{1}{2\pi} \int_{-\pi}^{\pi} |X_e(m, \omega) - Y(m, \omega)|^2 d\omega. \tag{7.26}$$

Proof of Equation (7.25) can be made by using Parseval's theorem, making the error criterion a function of the desired signal, $x_e[n]$, and minimizing with respect to the desired signal for each time n (Exercise 7.15). Note that although the distance measure is defined over continuous frequency, the implementation of the solution for $x_e[n]$ that minimizes the distance measure

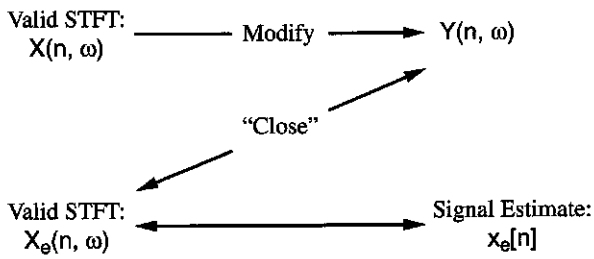

Figure 7.21 Approach of finding a sequence $x_e[n]$ whose (valid) STFT is "close" to a desired $Y(n, \omega)$ in some sense.

[6] This approach has also been used to recover a sequence from other time-frequency distributions such as the wavelet transform and Wigner distribution that we describe in Chapters 8 and 11.

involves frequency samples of $Y(n, \omega)$; therefore, it is required that the frequency sampling factor $\frac{2\pi}{N}$ be small enough so that unaliased versions of the short-time sections $f_m[n]$ are obtained.

The solution in Equation (7.25) extends in a simple manner to the case involving temporal decimation. Specifically, if L is the temporal decimation factor, then the solution in Equation (7.25) becomes:

$$x_e[n] = \frac{\sum_{m=-\infty}^{\infty} w[mL - n] f_{mL}[n]}{\sum_{m=-\infty}^{\infty} w^2[mL - n]} \tag{7.27}$$

which is illustrated graphically in Figure 7.22, where we see that for each time n a set of weighted short-time segments contributes to the solution. Finally, it is interesting to observe that when no modification is made, then the optimal solution is $x[n]$ as expected, provided that $\sum_{p=-\infty}^{\infty} w^2[pL - n] \neq 0$ (Exercise 7.15). We refer to Equation (7.27) as the least-squared-error (LSE) solution.

In general, the sum in the denominator of the right side of Equation (7.27) is a function of n. However, there exist analysis windows $w[n]$ such that the sum in the denominator is independent of n. It should be noted that the sum in the denominator has the same form as the sum in the constraint Equation (7.20) for the OLA method except that the analysis window is replaced by its square. That is, any window whose square satisfies the OLA constraint will make the

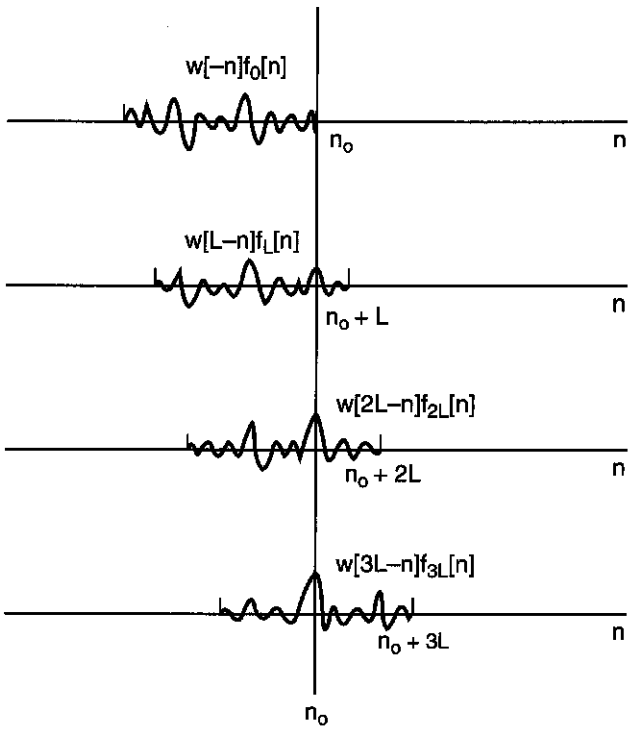

Figure 7.22 Least-squared-error (LSE) solution from the modified STFT with time decimation: $x_e[n] = \frac{\sum_{m=-\infty}^{\infty} w[mL-n] f_{mL}[n]}{\sum_{m=-\infty}^{\infty} w^2[mL-n]}$. Weighted short-time segment contributions are shown at time n_o.

Table 7.1 Comparison of the OLA and LSE solutions.

	OLA	LSE
Synthesis	$y[n] = \frac{L}{W(0)} \sum_{m=-\infty}^{\infty} \underbrace{x[n]w[mL-n]}_{f_{mL}[n]}$	$x_e[n] = \frac{\sum_{m=-\infty}^{\infty} w[mL-n] f_{mL}[n]}{\sum_{m=-\infty}^{\infty} w^2[mL-n]}$
Constraint	$\sum_{m=-\infty}^{\infty} w[mL-n] = \frac{W(0)}{L}$	$\sum_{m=-\infty}^{\infty} w^2[mL-n] \neq 0$

denominator sum in Equation (7.27) independent of n. If this happens, then Equation (7.27) can be simply interpreted as an overlap-add operation among the short-time sections corresponding to $Y(n, \omega)$, but with an additional weighting of each short-time section by the window. For the particular case of a rectangular analysis window that satisfies the OLA constraint, the LSE method reduces to the OLA method. A comparison of these relations is given in Table 7.1.

7.5.3 LSE Signal Estimation from Modified STFTM

The LSE approach can also be used for signal estimation from the modified STFTM which may or may not have come from a valid STFT. The approach is to estimate a signal $x_e[n]$ whose STFTM is "closest" in a least-squared-error sense to the modified STFTM [6]. More specifically, the resulting method estimates a sequence $x_e[n]$ from a desired time-frequency function $|Y(n, \omega)|$, which is a modified version of an original STFTM, $|X(n, \omega)|$, similar to that illustrated in Figure 7.21 but with STFT magnitudes replacing the STFTs. The method iteratively reduces the following distance measure between the STFTM, $|X_e(n, \omega)|$, of the signal estimate and the modified STFTM, $|Y(n, \omega)|$:

$$D[|X_e(n,\omega)|, |Y(n,\omega)|] = \sum_{m=-\infty}^{\infty} \frac{1}{2\pi} \int_{-\pi}^{\pi} \left[|X_e(m,\omega)| - |Y(m,\omega)|\right]^2 d\omega \quad (7.28)$$

where the minimization occurs with respect to the unknown signal $x_e[n]$ embedded within $X_e(n, \omega)$. The solution is found iteratively because as yet no closed-form solution has been discovered for $x_e[n]$ using the distance criterion in Equation (7.28). The iteration takes place as follows [6]. An arbitrary sequence (usually white noise) is selected as the first estimate $x_e^1[n]$ of $x_e[n]$. We then compute the STFT of $x_e^1[n]$ and modify it by replacing its magnitude by the desired magnitude $|Y(n, \omega)|$. From the resulting modified STFT, we can obtain a signal estimate (closest to this modified STFT) using the method based on Equation (7.25) in the previous section. This process then continues iteratively. In particular, the $(i + 1)$st estimate $x_e^{i+1}[n]$ is first obtained by computing the STFT, $X_e^i(n, \omega)$, of $x_e^i[n]$ and replacing its magnitude by $|Y(n, \omega)|$ to obtain $Y^i(n, \omega)$. The signal with the STFT closest to $Y^i(n, \omega)$ is found by using Equation (7.25). All steps in the iteration can be summarized in the following update equation:

$$x_e^{i+1}[n] = \frac{\sum_{m=-\infty}^{\infty} w[m-n] \frac{1}{2\pi} \int_{-\pi}^{\pi} Y^i(m,\omega) e^{j\omega n} d\omega}{\sum_{m=-\infty}^{\infty} w^2[m-n]}$$

where

$$Y^i(m,\omega) = |Y(m,\omega)| \frac{X_e^i(m,\omega)}{|X_e^i(m,\omega)|}.$$

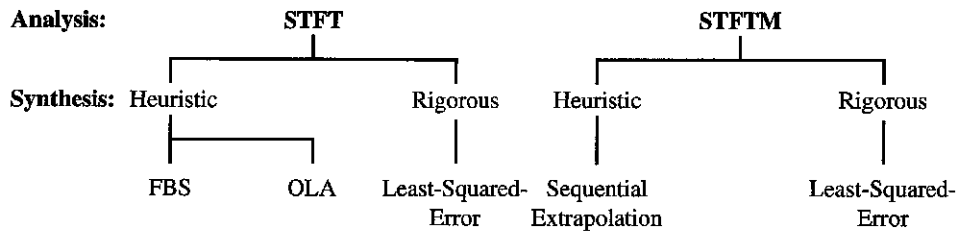

Figure 7.23 Overview of "heuristic" and "rigorous" synthesis methods corresponding to a modified STFT or SFTM analysis approach.

This algorithm is particularly important when phase is not available or difficult to measure or estimate, as demonstrated in the applications of the following section.

It has been shown [6] that this iterative procedure reduces the distance measure of Equation (7.28) on every iteration. Furthermore, the process converges to a local minimum, not necessarily the global minimum, of that distance measure. Although we restricted our above discussion to the discrete-time STFT, these results are easily extendable to the case where the STFT has been decimated in time. Furthermore, with discrete frequency the method iteratively reduces the distance measure in Equation (7.28), provided the frequency sampling factor is sufficiently large to avoid aliasing when determining the short-time sections corresponding to $Y^i(n, \omega)$.

Figure 7.23 summarizes our signal estimation methods from the modified STFT or STFTM. In each case, we have looked at both "heuristic" and "rigorous" methods of signal construction. For the STFT, the heuristic methods involved a brute-force application of the FBS and OLA synthesis methods, while the rigorous methods involved a closed-form solution to the LSE approach. For the STFTM, the heuristic methods involved a brute-force application of the sequential extrapolation method, while the rigorous method again involved the LSE approach, but using an iterative algorithm for solution.

7.6 Time-Scale Modification and Enhancement of Speech

The signal construction methods of this chapter can be applied in a variety of speech applications. In this section, we consider both the heuristic and rigorous synthesis methods from a modified STFT and STFTM in two applications: time-scale modification (i.e., changing the articulation rate of a speaker) and noise reduction. Time-scale modification is studied in a variety of other contexts throughout the text, while noise reduction is introduced more formally in a stochastic filtering framework in Chapter 13.

7.6.1 Time-Scale Modification

Model of Articulation Rate Change — Figure 7.24 shows a simple linear source/filter model for a change in the articulation rate of a speaker. The source input consists of impulses during voicing or plosives, and white noise during noisy speech. The glottal flow during voicing is embedded within the vocal tract response $h[n]$. In this model, the pitch changes at a faster or slower rate than the "nominal" (or normal) pitch so that the basic pitch *trajectory* is preserved; while the vocal tract moves faster or slower, the spectral *evolution* being preserved. In other

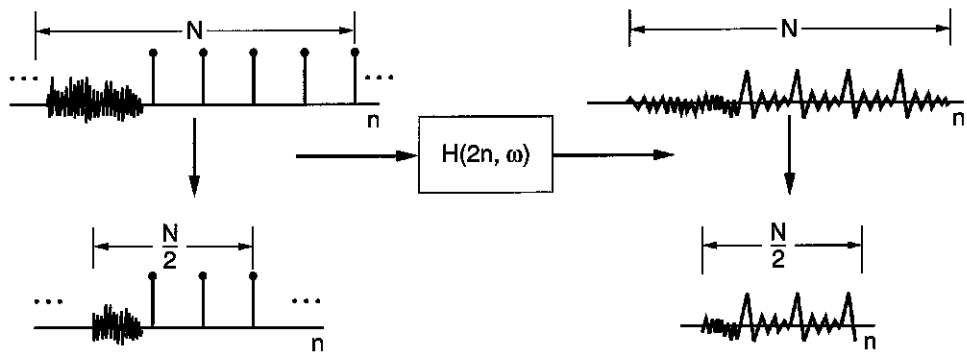

Figure 7.24 Model for uniform change in articulation rate. With a change in articulation rate, the source and system stay in (or move through) a certain state for longer or shorter time durations. In this illustration, the articulation rate is increased by a factor of two, so the source and speech waveforms are reduced to half their original length. In the source, the pitch trajectory is preserved during voicing, but the number of glottal cycles is reduced by a factor of two. The duration of the noise excitation is also reduced by a factor of two. The vocal tract moves twice as fast, but its frequency response is preserved, as indicated by the modified time-varying system function $H(2n, \omega)$.

words, the source and system pass through the same states as in nominal articulation, but they pass through these states for longer or shorter time durations. If the articulation rate is modified uniformly by a factor ρ, then, according to our model, the speech waveform is compressed or expanded in time by this factor ρ. The compression and expansion, however, do not occur as in speeding up or slowing down a tape recorder, because this would change pitch and spectral structure, but rather the waveform maintains its shape, simply decreasing or increasing the number of pitch periods during voicing and making unvoiced sounds last shorter or longer.

From our idealized convolutional model of speech production, the STFT of the speech waveform $X(n, \omega)$ can be written as a product of source and system components. In particular, for a sufficiently long window $w[n]$ (but intended to not be long enough to violate stationarity), we can think of each spectral slice of $X(n, \omega) \approx U(n, \omega)H(n, \omega)$, where $U(n, \omega)$ and $H(n, \omega)$ are the STFT of the source and the time-varying vocal tract system function, respectively. The duration of the window $w[n]$ is typically selected as two or three times the average pitch period, so that $U(n, \omega)$ is characterized by harmonic structure during voicing and by random fluctuations about the power spectrum during unvoiced regions (excluding an idealized impulsive source for a plosive sound category). In the frequency domain, the goal of time-scale modification is to modify the rate at which $U(n, \omega)$ and $H(n, \omega)$ vary with time and, therefore, how fast $X(n, \omega)$ varies with time. This coincides with how a change in articulation rate influences the speech spectrogram $|X(nL, \omega)|$. Suppose that a person speaks twice as slowly as his/her normal articulation rate. Then the spectral evolution of the speech would stretch over roughly twice the time of a normally spoken passage. This implies that the spectrogram is stretched out like an accordion. A similar argument can be made in speeding up articulation rate.

The model of articulation rate change is approximate; it is not known how articulation rate change takes place precisely, but it is clear that it does not occur uniformly, as assumed by our simple model. This nonuniform rate change was illustrated in Figure 3.30 of Chapter 3, which showed that voiced sounds tend to be altered more than fricative sounds or plosives.

For example, one does not expect a plosive to expand without limit because such expansion would alter the essence of the sound. We consider this more complex model in Chapter 9. Furthermore, the mechanism is probably more complicated than involving a simple temporal scaling of pitch and spectral trajectories. Time-scale modification may also entail changes in glottal volume velocity at the source and vocal tract shape which are not predictable by our idealized model. For example, in our model we have embedded the glottal flow during voicing in the vocal tract impulse response. With change in articulation rate, however, the flow may alter the spectral structure of this composite system response. For example, when speaking faster it may be less likely that we completely close the vocal folds (than in a nominal speaking state) because all articulators are moving faster; this incomplete closure affects the trend of the speech spectrum.

STFT Synthesis — One of the first time-scale modification systems uses the "cut and paste" method of Fairbanks [5].[7] To do time-compression by, for example, a factor of two, we extract successive short-time segments of the speech waveform and then discard every other time slice. Likewise, to expand the waveform, we repeat successive time slices. The modified temporal slices are then overlapped and added (Figure 7.25). In the frequency domain with time-decimation factor L, this corresponds to first forming the STFT, $X(nL, \omega)$, and then discarding or repeating spectral slices to form a new STFT $Y(nL, \omega)$. For example, with time-scale compression by a factor of two, $Y(nL, \omega) = X(2nL, \omega)$ (Figure 7.26). As we argued earlier, for a sufficiently long window $w[n]$, we can think of each spectral slice $X(nL, \omega) \approx U(nL, \omega)H(nL, \omega)$, where $U(nL, \omega)$ and $H(nL, \omega)$ are the STFTs of the source and the time-varying vocal tract spectrum, respectively. Thus, the source (pitch and voicing state) and system (vocal tract spectrum) characterization are roughly preserved in the modified STFT $Y(nL, \omega)$.

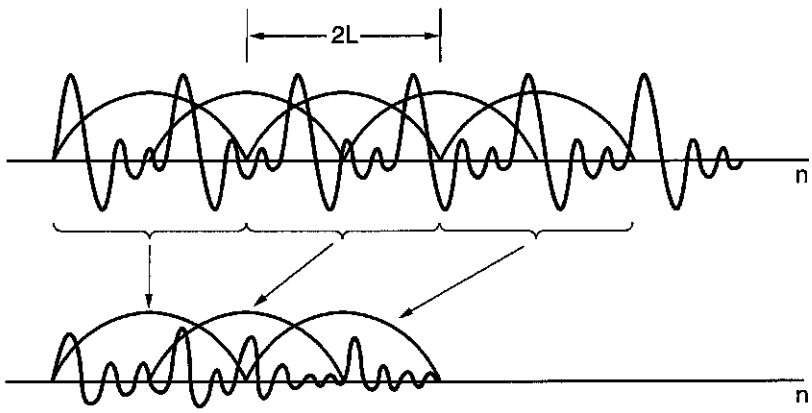

Figure 7.25 Time-scale compression by the Fairbanks technique. Alternating short-time segments are discarded and the remaining ones are overlapped and added.

[7] Observe that the methods in this section, including the Fairbanks and STFT- or STFTM-based approaches, are largely "non-model based," in contrast to "model-based" approaches. Examples of model-based approaches to time-scale modification use linear prediction and homomorphic analysis/synthesis (Exercise 7.16).

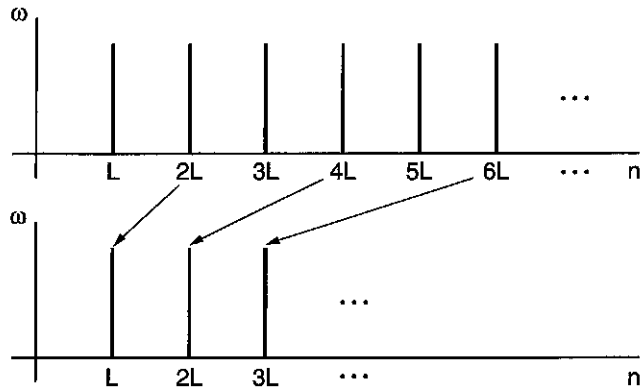

Figure 7.26 A STFT view of the Fairbanks technique, discarding every other frame for time-scale compression by a factor of two.

One can inverse Fourier transform the modified STFT, $Y(nL, \omega)$, and then, according to the (heuristic use of) OLA method, synthesize a modified sequence by overlapping and adding the resulting time slices. Although this transformation to the frequency domain and back is clearly superfluous, it gives us a different perspective on Fairbank's technique from the view of the OLA method. Observe that this synthesis is similar to the more rigorous LSE approach of Equation (7.27), except that, in the LSE approach, the inverted short-time segments are weighted by the analysis window, and the sum of weighted short-time segments is normalized by the sum of the squared overlapping windows. There is a problem, however, with both the heuristic and rigorous approaches: voiced segments are not "pitch synchronized," reflecting that the desired $Y(nL, \omega)$ is not a valid STFT in the sense that corresponding consecutive short-time segments are not consistent. For example, simply discarding every other frame gives a set of time slices that are not pitch synchronized when overlapped and added, as illustrated in Figure 7.27, producing a waveform construction with an erratic pitch perceived as "hoarseness" (Figure 7.25). A similar problem arises in time expansion where time slices are repeated to give a modified STFT. Alternatively, one can align the window on each frame at the same relative location within a pitch period, such as at a glottal pulse time, the time of glottal opening or closing, or a time instant at which there is no linear phase in the system impulse response (as we did in Section 6 of Chapter 6 in the context of homomorphic synthesis). You are asked to design such a system in Exercise 7.22.

This pitch-synchronized approach was introduced by Scott and Gerber [22], who used a glottal pulse time estimate to align pitch periods prior to overlapping and adding short-time segments, thus improving the quality of the time-scaled waveform. Successive frames are shifted so that they align at glottal pulse times. Rather than explicitly measuring glottal pulse times (or other times characteristic of the glottal flow), measures of "waveform similarity" are also used to select splicing points. In one such method, *synchronized overlap add* (SOLA), successive frames to be overlapped are cross-correlated. The peak of the cross-correlation function gives the time shift to make the two overlapping frames synchronize and thus add coherently[8] [21]. Other extensions of this approach have also been developed [9],[11].

[8] Conceptually, the measurement and application of this time difference is similar to the post-alignment process used in the mixed-phase homomorphic synthesis in Section 6 of Chapter 6.

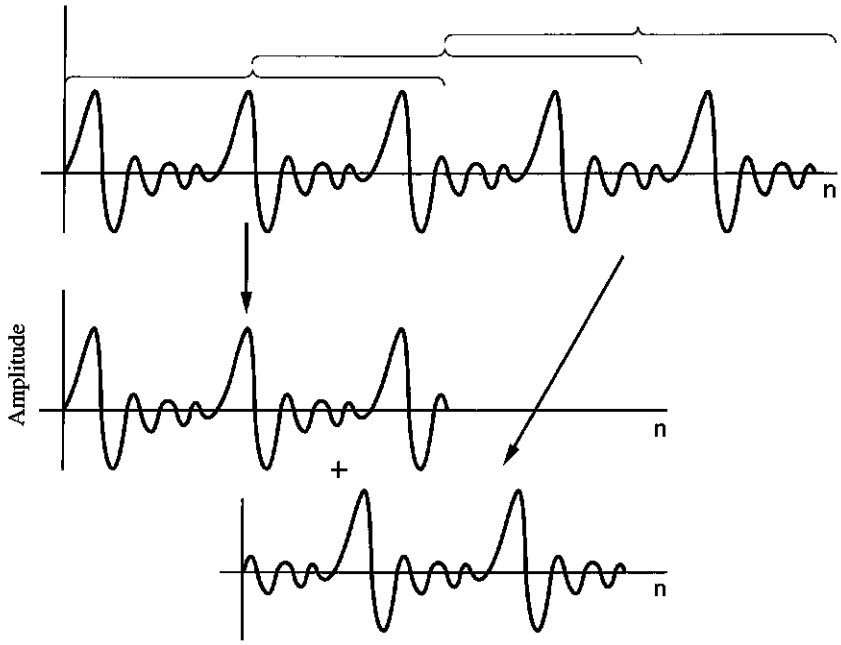

Figure 7.27 The problem of "pitch synchrony" in the OLA and LSE synthesis approaches. Pitch periods in the modified short-time segments are not synchronized.

The problems we have been describing here with pitch synchrony relate directly to the STFT phase function and, in particular, to the linear phase component of the STFT, because a time shift maps to a linear phase change. We now take an alternative approach to synthesis for time-scale modification, invoking the magnitude only of the STFT.

STFTM Synthesis — One way to avoid the need of pitch synchrony is to use only the magnitude of the STFT (STFTM) in time-scale modification. This approach is also motivated, in part, by the observation that when a person talks more rapidly or more slowly, the spectrogram is compressed or expanded in time like an accordion. As we argued above, with a sufficiently long window, $X(nL, \omega)$ captures essential source and system properties for articulation rate change. As with the Fairbanks technique, we discard or add frames, but now perform waveform synthesis from a modified STFTM, $|Y(nL, \omega)|$, using the iterative LSE method described in the previous section [6]. For time-compression by a factor of two, for example, we have the desired $|Y(nL, \omega)| = |X(2nL, \omega)|$, which was illustrated in Figure 7.26. Alternatively, to avoid the discarding of time slices, and thus the loss of pieces of the spectrogram, we propose the following algorithm for time-scale modification by a factor $\rho = \frac{M}{L}$ illustrated with $\rho = \frac{1}{2}$:

S1: Compute $|X(nL, \omega)|$ at an appropriate frame interval (e.g., $L = 128$ at 10000 samples/s).

S2: Pretend $|X(nL, \omega)|$ was determined with time decimation $M = \frac{L}{2}$ and form the desired STFTM (Figure 7.28):

$$|Y(nM, \omega)| = |X(nL, \omega)|.$$

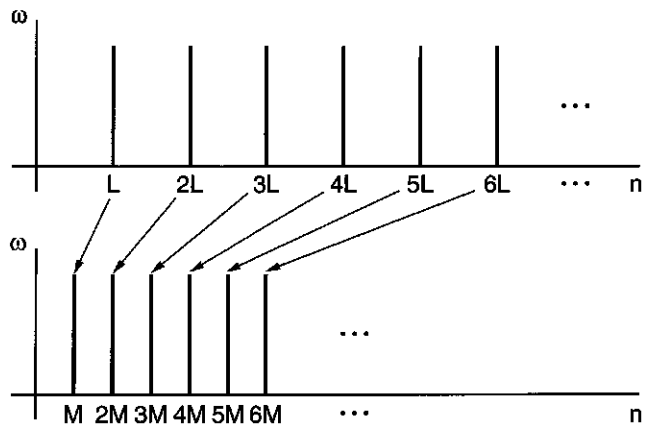

Figure 7.28 Alternative modified STFT for time-scale modification where no frames are discarded. In the example, $M = \frac{L}{2}$.

S3: Apply the LSE iterative algorithm with the desired $|Y(nM, \omega)|$.

EXAMPLE 7.8 Figure 7.29 shows an example with an original spectrogram [panel (a)] computed with $L = 128$ and the desired spectrogram [panel (b)] formed by compressing the original so that $M = \frac{L}{2} = 64$ [6]. The Fourier transforms were implemented with 512-point FFTs and the LSE algorithm ran for 100 iterations. Panel (c) in the figure gives the spectrogram of the resulting compressed waveform. The initial waveform estimate is Gaussian white noise. The distance measure $D[|X_e(n, \omega)|, |Y(n, \omega)|]$ of Equation (7.28) for this example, as a function of iteration number, does not approach zero, but does decrease monotonically. ▲

In time-scale modification of voiced speech with this method, although judged to be generally of good quality, a reverberant characteristic is occasionally perceived due to lack of STFT phase control. In other words, the STFT phase of the modified speech can differ from that of the original speech because we have constrained only the STFT magnitude. This phase change corresponds to a dispersed waveform that, visually, may bear little resemblance to the original speech signal (Exercise 7.17). In addition, in slowing down noise-like unvoiced speech (e.g., fricatives), a metallic quality is perceived. This effect can be understood by recalling that the periodogram of a random process fluctuates about an underlying power spectral density. With time-scale expansion using the LSE method, random peaks in the periodogram are stretched and thus held over time, resulting in unnatural "tones" in the synthesis, not characteristic of the original random process. Another important property of the LSE method is that the synthesis is highly dependent on the length of the analysis window $w[n]$. A long window, e.g., 2–3 pitch periods in duration, and thus a narrowband spectrogram are necessary to capture harmonic structure; this ensures a time-scale modified waveform for which harmonics are stretched or compressed in time. (Recall that we want $X(n, \omega) \approx U(n, \omega)H(n, \omega)$, where for voiced speech $U(n, \omega)$ is harmonic.) In contrast, with a short window, e.g., about a pitch period in duration, the method modifies the pitch of the speaker and not the time scale (Exercise 7.18).

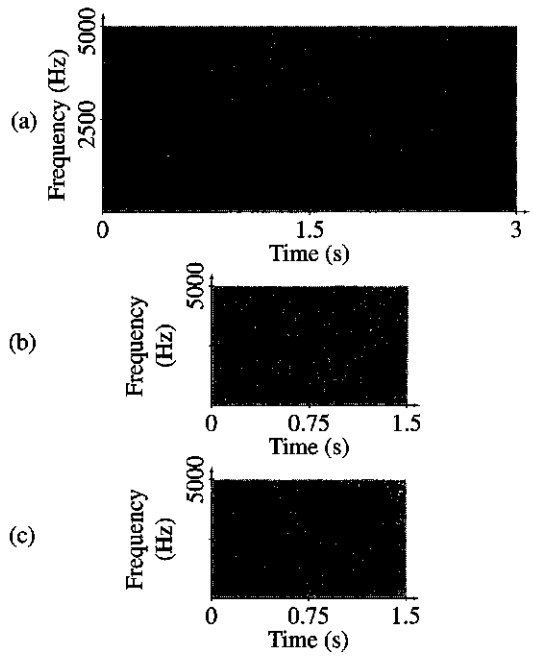

Figure 7.29 Time-scale modification with iterative LSE estimation: (a) original STFTM with $L = 128$; (b) modified STFTM with $L = 64$; (c) STFTM of LSE estimate. Speech utterance is, "Line up at the screen door."

SOURCE: D. Griffin and J.S. Lim, "Signal Estimation from Modified Short-Time Fourier Transform" [6]. ©1984, IEEE. Used by permission.

7.6.2 Noise Reduction

A number of STFT processing techniques have been developed for the reduction of additive noise in speech signals. The noise corrupted signal $y[n]$ is given by

$$y[n] = x[n] + b[n]$$

where $x[n]$ is the speech and $b[n]$ is the noise sequence. The approach is to modify each spectral slice of the STFT of $y[n]$ to remove noise and, from the resulting modified STFT, construct an enhanced waveform. As with time-scale modification, both STFT- and STFTM-based synthesis methods can be applied.

STFT Synthesis — In one of the first approaches to noise reduction, the magnitude of the STFT is modified by subtracting off an approximate noise power spectrum $\hat{S}_b(\omega)$ from $|Y(nL, \omega)|^2$ and retaining the noisy phase [3], i.e., the desired modified STFT is formed as

$$\hat{X}(nL, \omega) = [|Y(nL, \omega)|^2 - \alpha \hat{S}_b(\omega)]^{\frac{1}{2}} e^{j \angle Y(nL, \omega)}$$

where if $|Y(nL, \omega)|^2 - \alpha \hat{S}_b(\omega) < 0$, then the difference is set to zero, and where the phase is retained because of difficulty in its estimation. In this *spectral subtraction* method, the parameter

α serves as a control for the degree of noise reduction. Alternatively, in another early approach to noise reduction, the magnitude of the STFT is modified by a time-varying "optimal filter" $H(nL, \omega)$. This filter reduces the measured spectral slice when the noise spectrum is high relative to the time-varying speech spectrum, and preserves the spectral measurement when the noise spectrum is low relative to the speech spectrum, but again keeps the measured phase, i.e.,

$$\hat{X}(nL, \omega) = [|Y(nL, \omega)|H(nL, \omega)]e^{j\angle Y(nL,\omega)}.$$

In either case, the OLA or LSE method can then be applied to obtain an enhanced waveform construction. Such processing has been performed with the above modifications with a resulting drop in noise level. The primary processing artifacts with these basic approaches are the presence of short tone-bursts of varying frequency in the noise, sometimes referred to as "musicality," and some distortion of the speech spectrum [10]. These noise reduction methods are described in greater detail in Chapter 13.

STFTM Synthesis — Alternatively, we can avoid retaining the noisy measured phase and obtain a signal estimate from the modified STFT magnitude, using either the sequential extrapolation algorithm or the LSE estimation method. In one example, spectral subtraction was applied to the STFTM of a noisy speech waveform using a temporal decimation factor $L = 64$ and Hamming window of length $N_w = 128$. An enhanced speech waveform was then obtained using a variation of the sequential extrapolation algorithm described in Section 7.4.2. With a variety of signal-to-noise ratios (SNR) between 0 to 20 dB, for SNR above about 10 dB the signal estimates from the modified STFTM had a reduced noise level and retained natural speech quality and speaker identifiability [12]. As with the use of the STFT-based synthesis above, the primary artifacts are "musicality" and some distortion of the speech spectrum, typical of short-time spectral subtraction. A potential advantage of the STFTM-based approach over the STFT-based synthesis methods is that a STFT phase estimate is obtained (indirectly) rather than retaining the noisy phase function. Formal listening tests comparing the STFT- and STFTM-based synthesis, however, have yet to be performed using spectral subtraction or other, more refined methods that modify the spectral magnitude.

7.7 Summary

In this chapter, we presented both a Fourier transform and a filtering perspective of the discrete-time and the discrete STFT in signal analysis. Corresponding to these two analysis viewpoints are the overlap-add (OLA) and filter bank summation (FBS) methods of signal synthesis from the discrete STFT. Likewise, we considered signal analysis from the STFT magnitude (STFTM) for which we developed a sequential extrapolation approach to synthesis. Analysis and synthesis with the STFTM is useful when the STFT phase is not appropriate or difficult to measure. We also considered the important practical problem of estimating a signal from a processed STFT or STFTM which does not satisfy the definitional constraints of the STFT. Under this condition, blind use of the OLA, FBS, and sequential extrapolation methods can be applied. A more "rigorous" approach, however, was also developed whereby a signal estimate is obtained that has an STFT or STFTM close to the desired modified time-frequency function in a least-squared-error sense. Finally, we touched upon two particular applications of STFT and STFTM analysis and synthesis: time-scale modification and noise reduction. These applications illustrate how the STFT and STFTM analysis/synthesis techniques can provide a framework for speech signal processing. We will encounter this framework in many other speech processing areas throughout the text.

Appendix 7.A: FBS Method with Multiplicative Modification

Let the discrete STFT be modified by a function $H(n, k)$, i.e.,

$$y[n] = \frac{1}{Nw[0]} \sum_{k=0}^{N-1} X(n, k)H(n, k)e^{j\frac{2\pi}{N}nk}$$

and consider the time-invariant case

$$H(n, k) = H(k).$$

Letting $w[0] = 1$, and substituting the discrete STFT, $X(n, k)$:

$$\begin{aligned} y[n] &= \frac{1}{N} \sum_{k=0}^{N-1} \left[\sum_{m=-\infty}^{\infty} w[n - m]x[m]e^{-j\frac{2\pi}{N}km} \right] H(k)e^{j\frac{2\pi}{N}nk} \\ &= \sum_{m=-\infty}^{\infty} w[n - m]x[m] \left[\frac{1}{N} \sum_{k=0}^{N-1} H(k)e^{j\frac{2\pi}{N}k(n-m)} \right] \end{aligned}$$

where the bracketed term is the inverse DFT of $H(k)$ evaluated at $n - m$. Therefore,

$$\begin{aligned} y[n] &= \sum_{m=-\infty}^{\infty} w[n - m]x[m] \sum_{l=-\infty}^{\infty} h[n - m - lN] \\ &= \sum_{m=-\infty}^{\infty} x[m] \left(w[n - m] \sum_{l=-\infty}^{\infty} h[n - m - lN] \right). \end{aligned}$$

Because the bracketed term is a function of $n - m$, $y[n]$ can be written:

$$y[n] = x[n] * \hat{h}[n]$$

where

$$\hat{h}[n] = w[n] \sum_{l=-\infty}^{\infty} h[n - lN]$$

and where we take only values in the interval $[0, N - 1]$ due to the inverse DFT operation. But if we assume the FBS constraint $N_w < N$ is satisfied, we see in Figure 7.30 that there is no aliasing of the $h[n]$ replicas and $\hat{h}[n]$ reduces to $\hat{h}[n] = w[n]h[n]$. We then need to select a window $w[n]$ to achieve some desired fixed filter impulse response. In particular, we see that a rectangular window does not result in a change in the original impulse response $h[n]$.

More generally, it can be shown for a time-varying multiplicative modification $H(n, k)$:

$$y[n] = \sum_{m=-\infty}^{\infty} x[n - m]\hat{h}[n, m] \tag{7.29}$$

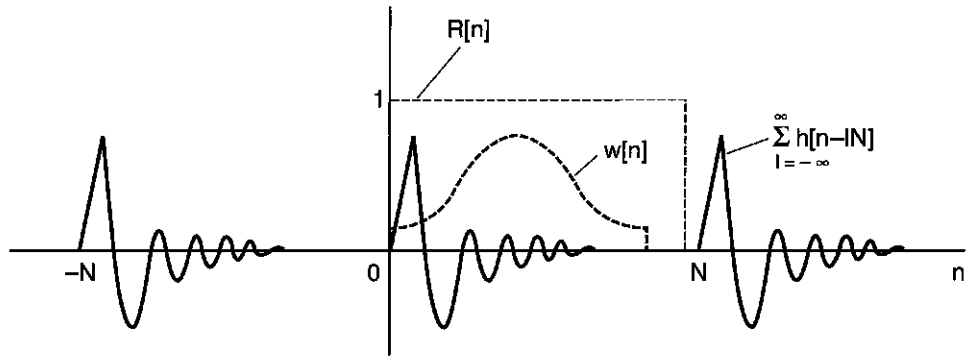

Figure 7.30 Schematic of relation of periodic sequence $\sum_{l=-\infty}^{\infty} h[n-lN]$ and the window $w[n]$ for the FBS method with a multiplicative modification. $R[n]$ is the rectangular function, non-zero in the interval $[0, N-1]$, invoked by the inverse DFT.

where

$$\hat{h}[n, m] = w[m] \sum_{l=-\infty}^{\infty} h[n, m - lN]$$

which is equivalent to filtering $x[n]$ with a time-varying impulse response $\hat{h}[n, m]$. We window $\sum_{l=-\infty}^{\infty} h[n, m-lN]$ and then convolve at each time instant, n, to obtain a single value $y[n]$. When aliasing of the response replicas does not occur, then $\hat{h}[n, m]$ reduces to the windowed time-varying filter response.

In summary, we see that for a time-varying $h[n, m]$, at each time n we invoke a different filter in the variable m. In essence, in Equation (7.29), we are filtering $x[n]$ with a time-varying impulse response. This operation was first introduced in Chapter 2 in Equations (2.26) and (2.27), where we described the time-varying unit sample response $h[n, m]$ and its related Green's function $g[n, m]$. This connection leads to other interesting relations between time-varying filtering and STFT analysis/synthesis derived by Portnoff [17], as well as its application to time-scale modification of speech [16].

EXERCISES

7.1 Show that the STFT expression in Equation (7.4) can be rewritten as

$$X(n, \omega_o) = e^{-j\omega_o n}(x[n] * w[n]e^{j\omega_o n}).$$

7.2 Repeat Example 7.1 using a window of length two pitch periods, corresponding to a narrowband spectrogram case. Compare your result with the wideband spectrogram case of Example 7.1.

7.3 Derive the FBS constraint Equation (7.13) in the frequency domain from Equation (7.11).

7.4 Show that for the OLA synthesis method, under OLA constraint Equation (7.20), the reconstruction Equation (7.21) follows.

7.5 Explain qualitatively the *duality principle* in both the time and frequency domains, which is associated with the FBS constraint Equation (7.13) and the OLA constraint Equation (7.20).

7.6 We stated in Section 7.3 that a generalized synthesis equation for recovering a sequence $x[n]$ from its STFT $X(n, \omega)$ is given by

$$x[n] = \frac{1}{2\pi} \int_{-\pi}^{\pi} \left[\sum_{r=-\infty}^{\infty} f[n, n-r] X(r, \omega) \right] e^{j\omega n} d\omega \tag{7.30}$$

where

$$X(r, \omega) = \sum_{m=-\infty}^{\infty} x[m] w[r-m] e^{-j\omega m}$$

and where the function $f[n, m]$ is referred to as the synthesis filter.

(a) Show that a sequence $x[n]$ can be recovered from Equation (7.30) under the constraint:

$$\sum_{m=-\infty}^{\infty} f[n, -m] w[m] = 1 \tag{7.31}$$

thus ensuring that Equation (7.30) is a valid synthesis equation.

(b) Show that with $f[n, m] = \delta[n]$, Equation (7.30) yields the synthesis equation for the basic FBS method.

(c) Show that with $f[n, m] = \frac{1}{W(0)}$, Equation (7.30) yields the synthesis equation for the OLA method with time decimation factor $L = 1$.

7.7 Consider a sequence $x[n]$ and an analysis window $w[n]$, which is non-zero over its length, N_w (assumed even). Suppose the first L samples of $x[n]$ are known. In this problem, you consider the representation of $x[n]$ by time and frequency samples of its STFT magnitude.

(a) Show that each short-time autocorrelation function $r[n, m]$ of $x[m]w[m-n]$ is at most $2N_w - 1$ points long and thus can be obtained (without aliasing) from $2N_w - 1$ frequency samples of the STFT magnitude.

(b) Consider the effects of temporal decimation with factor L (i.e., $r[nL, m]$) for which adjacent short-time sections have an overlap of $N_w - L$ samples. Prove that the sequence $x[n]$ can be uniquely recovered from the STFT magnitude, provided that:

1. Overlap between short-time sections is greater than $N_w/2$.
2. There exist no zero gaps in $x[n]$ of length greater than $N_w - 2L$.

7.8 The purpose of this problem is to show that, under certain conditions, only one frequency sample of the STFT magnitude is needed for unique representation of a non-negative sequence $x[n]$ for a time-decimation factor of unity. The STFT for a time-decimation factor of unity is defined by

$$X(n, \omega) = \sum_{m=-\infty}^{\infty} x[m] w[n-m] e^{-j\omega m}$$

where $w[n]$ is the analysis window of length N_w. Suppose that the following conditions on $x[n]$ and $w[n]$ hold (Figure 7.31):

1. $x[n]$ is a non-negative (i.e., $x[n] \geq 0$) right-sided sequence whose first non-zero value falls at $n = n_0$.
2. The analysis window $w[n]$ is N_w points long and positive (i.e., $w[n] > 0$) over the interval $0 \leq n \leq N_w - 1$.

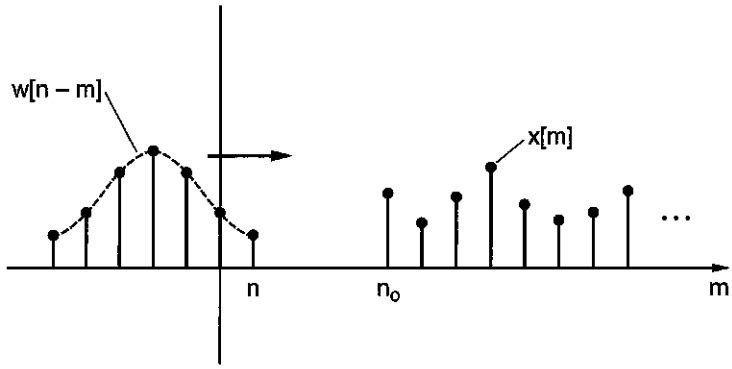

Figure 7.31 Relation between the analysis window and the sequence in generating the STFT for Example 7.8.

You are now asked to show that the sequence $x[n]$ is specified by one appropriately chosen frequency sample of $|X(n, \omega)|$.

(a) Consider the smallest value of n, namely n_o, such that $x[n]$ is non-zero. Show that $x[n_o]$ can be determined by the expression (Figure 7.31):

$$x[n_o] = \frac{|X(n_o, 0)|}{w[0]}.$$

(b) Suppose now that $x[n]$ is known up to time $n - 1$ and we want to compute the sample $x[n]$ from the previous values and $|X(n, \omega)|$. Show that

$$X(n, \omega) = Y(n, \omega) + x[n]w[0]e^{-j\omega n}$$

where

$$Y(n, \omega) = \sum_{m=n-N+1}^{n-1} x[m]w[n - m]e^{-j\omega m}.$$

$Y(n, \omega)$ is known, since it is a function of samples prior to time n.

(c) Under the above two conditions, show that $x[n]$ can be evaluated as

$$x[n] = \frac{1}{w[0]}(|X(n, 0)| - \sum_{m=n-N+1}^{n-1} x[m]w[n - m]).$$

Argue then that $x[n]$ can be recovered recursively for $n > n_o$ from only the DC value, i.e., $|X(n, 0)|$, of the STFT magnitude. It is interesting to note that this result can be generalized to require two frequency samples when the non-negative restriction is taken off of $x[n]$.

7.9 We saw in Section 7.6.1 a time-scale modification method based on a modified STFT magnitude, using the least-squared-error estimation method. This problem asks you to consider time-scale modification using the STFT.

(a) Suppose the STFT of $x[n]$ is computed with a time-decimation factor $L = 128$. Propose and describe a method based on least-squared-error estimation, from a modified STFT, to time-scale compress $x[n]$ by a factor of two. Briefly describe the steps of your method.

(b) Why might the least-squared-error estimation method be flawed? Carefully explain your reasoning. *Hint:* Consider the STFT phase.

7.10 Show that for a frame interval of L samples (i.e., time decimation of L samples), $2L$ samples of $|X(nL, k)|$ over $[0, \pi]$ in frequency are required to uniquely recover a sequence $x[n]$. Assume the window length $N_w \geq L/2$. For simplicity, assume N_w and L are even.

7.11 We derived in Section 7.5.1 (Appendix 7.A) the signal $y[n]$ resulting when the discrete STFT $X(n, k)$ of a signal $x[n]$ is modified by a time-invariant multiplicative modifier $H(n, k) = H(k)$. This was done for the FBS method. Derive the case of a time-varying multiplicative modifier $H(n, k)$ for the FBS method. Then derive the signal synthesis from a multiplicatively modified STFT using the OLA method with first the time-invariant and then the time-varying case. We will then have derived two cases for each method: (1) A time-invariant modification $H(k)$ and (2) A time-varying modification $H(n, k)$, for both FBS and OLA synthesis.

7.12 Consider modifying the STFT to obtain

$$Y(n, \omega) = X(n, k\omega)H(\omega)$$

where the modifying function is given by

$$H(\omega) = e^{jn_0\omega}$$

i.e., a linear-phase modification. Suppose a sequence $y[n]$ is computed by the filter bank summation (FBS) synthesis method:

$$y[n] = \left[\frac{1}{NW(0)}\right] \sum_{k=0}^{N-1} Y(n, k) e^{j\frac{2\pi}{N}kn}.$$

Derive an expression for $y[n]$ in terms of the original sequence $x[n]$ and the window $w[n]$. Consider two different cases: (1) The length of $w[n]$ less than N, and (2) The length of $w[n]$ greater than or equal to N. Give $y[n]$ for each case.

7.13 Figure 7.32 illustrates an idealization of the spectrogram, i.e., the short-time Fourier transform magnitude (STFTM), of the word "ate," consisting of the voiced phoneme /e/, the unvoiced phoneme /t/,

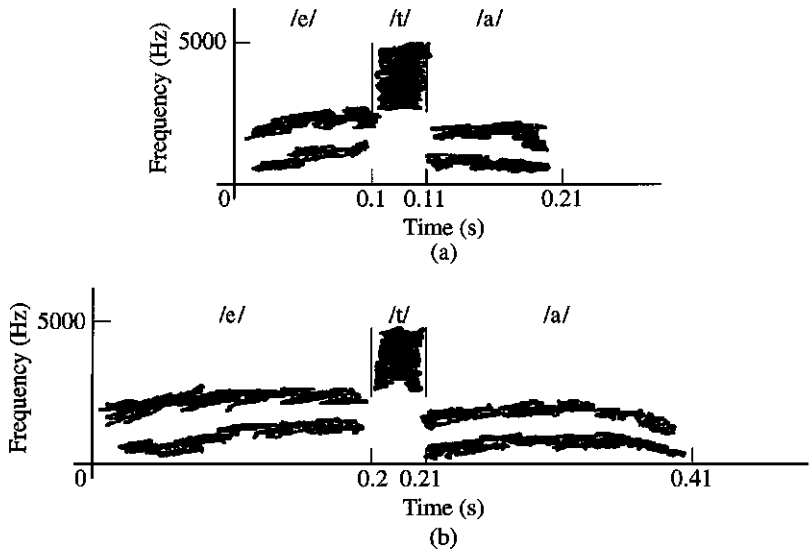

Figure 7.32 Spectrograms of the word "ate": (a) case I; (b) case II.

followed by the voiced phoneme /a/. The two different spectrograms correspond to two different articulation rates.

(a) Describe the differences in the articulation rate of the speaker with respect to the two spectrograms.

(b) Suppose you are given only the spectrogram of Figure 7.32(a) and the time-decimation factor $L = 10$ samples with a speech sampling rate of 10000 samples/s. In a few steps, describe a method to construct a waveform whose spectrogram approximately equals that of Figure 7.32(b). Assume a 200-sample Hamming analysis window. Are there any special considerations at the voiced/unvoiced and unvoiced/voiced transitions?

7.14 Consider a discrete-time signal $x[n]$ passed through a bank of filters $h_k[n]$ where each filter is given by a modulated version of a baseband prototype filter $h[n]$, i.e.,

$$h_k[n] = h[n] \exp[j(2\pi/N)kn]$$

where $h[n]$, a Hamming window, is assumed causal and lies over a duration $0 \leq n < N_w$, and $2\pi/N$ is the frequency sampling factor. In this problem, you are asked to time-scale expand some simple input signals by time-scale expanding the filter bank outputs.

(a) State the constraint (with respect to the values N_w and N) such that the input $x[n]$ is recovered when the filter bank outputs are summed.

(b) If the input to the filter bank is the unit sample $\delta[n]$, then the output of each filter is a complex exponential with "envelope" $a_k[n] = h[n]$ and phase $\theta_k[n] = (2\pi/N)kn$. Suppose each complex exponential output is time-expanded by two by interpolation of its envelope and phase (as in Chapter 2, Exercise 2.19). Derive a new constraint (with respect to values N_w and N), so that the summed filter bank outputs equal $\delta[n]$.

(c) Suppose now that the filter bank input equals

$$x[n] = \delta[n] + \delta[n - n_o],$$

and that the filter bank outputs are time-expanded as in part (b). Derive a sufficient condition on N_w, N, and n_o so that the summed filter bank output is given by

$$y[n] = \delta[n] + \delta[n - 2n_o],$$

i.e., the unit samples are separated by $2n_o$ samples rather than n_o samples.

7.15 In Section 7.5.2, we considered minimizing the mean-squared error between a modified STFT $Y(mL, \omega)$ and a valid STFT $X_e(mL, \omega)$. In this problem, you investigate the solution to this optimization problem and some related properties.

(a) Derive Equation (7.27), i.e.,

$$x_e[n] = \frac{\sum_{m=-\infty}^{\infty} w[mL - n] f_{mL}[n]}{\sum_{m=-\infty}^{\infty} w^2[mL - n]}$$

by minimization of Equation (7.26), generalized with the time decimation factor L. $f_{mL}[n]$ is the inverse Fourier transform of the modified STFT, $Y(mL, \omega)$, and $w[n]$ is the analysis window. *Hint:* Use Parseval's Theorem, make the error criterion a function of the desired signal $x_e[n]$, and minimize with respect to the desired signal for each time n.

(b) Suppose that $f_{mL}[n]$ is obtained from the unmodified STFT of a signal $x[n]$, i.e., $Y(mL, \omega) = X(mL, \omega)$. Show that the solution of part (a) recovers $x[n]$, i.e., $x_e[n] = x[n]$, provided

$$\sum_{m=-\infty}^{\infty} w^2[mL - n] \neq 0.$$

(c) Now suppose that $f_{mL}[n]$ is obtained from the following modified STFT

$$Y(mL, \omega) = X(mL, \omega)e^{-j\omega n_o}.$$

Under what condition on $w[n]$ and L does $x_e[n] = x[n]$ within a shift n_o and constant scale factor, i.e., $x_e[n] = cx[n - n_o]$?

(d) Suppose in part (c) that $w[n]$ is unity over a duration $N_w = L$. Derive a simplified expression for $x_e[n]$, starting with your solution in part (c).

(e) Discuss the similarities and differences between the OLA synthesis and the least-squared-error synthesis (part (a)). Give one set of conditions under which they give the same solution. Consider window constraints for each method. Discuss the relative benefits of the least-squared error method against the OLA method.

7.16 We have seen the application of short-time Fourier transform analysis/synthesis of the speech waveform to time-scale modification. This problem asks you to consider the application of homomorphic analysis/synthesis to time-scale modification.

(a) Consider the voiced speech case in the context of homomorphic analysis/synthesis. Suppose that pitch, voicing state, and a minimum-phase impulse response, $h[n]$ have been estimated every 10 ms. How would one modify the synthesis structure so that a voiced 10-ms segment is mapped to a 20-ms duration without change in pitch or vocal tract spectral characteristics?

(b) Now consider the unvoiced speech case, again with a 10-ms analysis interval. How would one modify the synthesis structure so that a 10-ms unvoiced segment is mapped to a 20-ms unvoiced segment without a change in the noise-like excitation or vocal tract spectral characteristics?

7.17 This problem addresses the least-squared-error approach to time-scale modification from a modified STFT.

(a) Suppose that the discrete STFT magnitude $|X(nL, k)|$ (time-decimated) of a speech signal is computed with a time-decimation factor $L = 32$ samples. Give a modified STFT magnitude from which time-scale expansion by a factor of two can be performed. Propose an algorithm for estimating the time-scale expanded signal whose STFT magnitude is a least-squared-error estimate of the modified STFT magnitude.

(b) What can you conclude about the phase of the STFT of the resulting time-scale expanded signal from part (a)? What are the implications of your answer for the shape of the modified (expanded) waveform?

7.18 Suppose you are given a wideband spectrogram created with a short analysis window of duration less than the average pitch period for the speaker being analyzed. In this problem, you study the use of the least-squared-error (LSE) signal estimation (from spectral magnitude-only) method of Section 7.5.3 to perform time-scale modification.

(a) Explain why the LSE signal estimation method, using the above wideband spectrogram, results in pitch modification and not the intended time-scale modification.

(b) How small a time decimation (i.e., the value of L) of the wideband spectrogram does one need to sufficiently track energy variations within a pitch period and thus, using the above LSE method, avoid a resulting irregular pitch change and thus a hoarse quality to the speech synthesis.

7.19 This problem considers changes in the spectrogram of an utterance with time-scale and pitch modification. The utterance is spoken by a female.

(a) In Figure 7.33, select the spectrograms that are narrowband spectrograms (generated with a wide time-domain analysis window) and the spectrograms that are wideband spectrogram (generated with a narrow time-domain analysis window). Briefly explain your reasoning.

(b) In Figure 7.33, select the spectrogram that is the time-scaled compressed version of the spectrogram in panel (a). Which spectrogram is the time-scaled expanded version of the spectrogram in panel (a)? What speech transformations were invoked to obtain the spectrograms you have not selected? Briefly explain your reasoning.

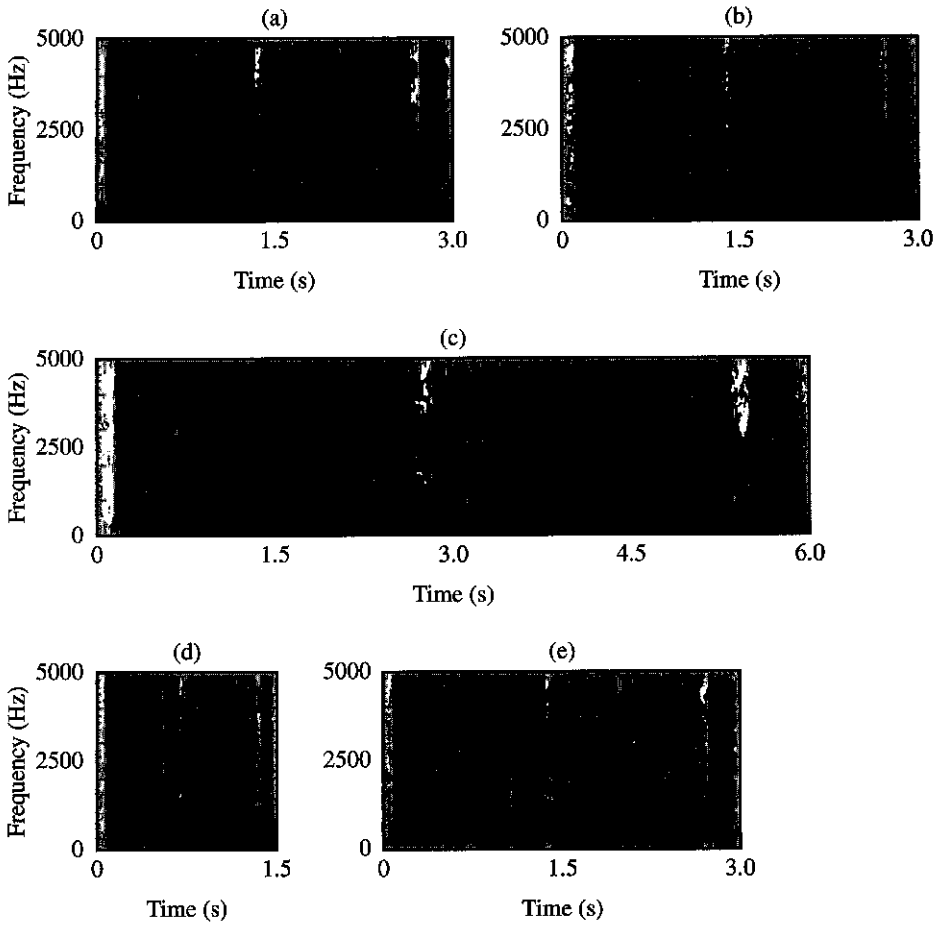

Figure 7.33 Narrowband and wideband spectrograms: (a) spectrogram of original passage; (b)–(e) spectrograms of modified passage.

(c) Suppose the time-scaled modified spectrograms in Figure 7.33 represent the results using STFT magnitude (STFTM) analysis/synthesis. In doing time-scale expansion by a factor of two with STFTM analysis/synthesis, suppose the synthesis interval in the synthesis stage for modification is 7 ms. What was the analysis frame duration in the analysis stage? In answering this question, use your result from Exercise 7.17.

7.20 We have seen that in idealized time-scale modification of a steady-state vowel, the pitch is maintained and the number of pitch periods changes. Suppose that the combined glottal flow (over one glottal cycle) and the vocal tract impulse response is denoted by $h[n] = g[n] * v[n]$. Then the steady-state vowel of duration N samples is expressed by

$$\begin{aligned} x[n] &= r[n](p[n] * h[n]) \\ &= r[n] \sum_{k=-\infty}^{\infty} h[n - kP] \end{aligned}$$

where $r[n]$ is an N-sample rectangular window, i.e., $r[n] = 1$, for $0 \leq n \leq N - 1$, and zero otherwise, and $p[n]$ is a periodic impulse train, $p[n] = \sum_{k=-\infty}^{\infty} \delta[n - kP]$.

(a) Write an expression for the steady-state vowel, time-scaled expanded by a factor of two. Sketch your result in contrast to the original waveform. Assume that the lumped response does not change in the modification, implying that neither the vocal tract impulse response nor the glottal flow shape changes.

(b) In reality, the glottal flow shape may change with a change in articulation rate. Suppose that the glottal flow during normal articulation is maximum-phase and given by

$$g[n] = \alpha^{-n}u[-n] * \alpha^{-n}u[-n]$$

i.e., a time-reversed decaying exponential convolved with itself. And suppose that when the articulation rate is slowed by a factor of two, then the glottal flow function widens as

$$g'[n] = g[n/2] = \alpha^{-n/2}u[-n] * \alpha^{-n/2}u[-n].$$

Assume now you are given the discrete STFT magnitude of $x[n]$, i.e., $|X(nL, k)|$, computed at a 10-sample frame interval (i.e., for nL where $L = 10$ is the time decimation factor) and with a 1024-point DFT. Assume also that

$$|X(nL, k)| \approx |P(nL, k)||G(k)||V(k)|$$

where $|P(nL, k)|$ is the STFT magnitude of $p[n]$, and assume you are given $|G(\omega)|$ (you can obtain this approximately from the complex cepstrum).

Propose an iterative algorithm from a modified STFT magnitude that yields a steady-state vowel, time-scale expanded by a factor of two, but having approximately a glottal flow with spectral magnitude $|G'(\omega)|$. You must first define the desired STFT magnitude and determine a way to find it. Can the resulting waveform be expressed as in your result of part (a)? (*Hint:* Consider the STFT phase of the resulting waveform, and whether your algorithm can discern a difference between $g'[n]$ and $g'[-n]$, or between $v[n]$ and $v[-n]$.)

7.21 (MATLAB) In this problem you use the speech waveform *speech2_10k* in the workspace *ex7M1.mat* located in companion website directory Chap_exercises/chapter7. This problem helps you to develop an understanding of the limitations of the STFT in achieving good time-frequency resolution. The speech was sampled at 10000 samples/s.

(a) Plot the speech signal *speech2_10k*, an unvoiced/voiced speech transition, and triangular windows of durations 30 ms and 5 ms, respectively, created using MATLAB function *triang.m*.

(b) Plot the first eight STFT log-magnitudes of *speech2_10k* by sliding the 30-ms triangular window in 15-ms intervals. Use a 1024-point FFT and display only the first 512 points of the STFT log-magnitude. Use the command *subplot (221)* so you can display two sets of four functions. Also, make a matrix of your eight STFT log-magnitudes (512 points each), and then use the *mesh.m* MATLAB function to plot the 2-D time-frequency function.

(c) Repeat part (b) with the 5-ms triangular analysis window.

(d) Repeat parts (b) and (c), but display spectrograms using the MATLAB function *spectrogram.m* or *specgram_ex7p21.m* in Chap_exercises/chapter7, rather than a mesh plot.

(e) Comment on the time-frequency resolution tradeoffs in using the long and short triangular windows.

7.22 (MATLAB) In this problem you use the voiced speech waveform *speech2_10k* in the workspace *ex7M1.mat* located in companion website directory Chap_exercises/chapter7. You are asked to design time-scale modification systems based on the OLA and LSE synthesis methods. The speech was sampled at 10000 samples/s.

(a) Write a MATLAB function to compute the STFT of the sequence *speech2_10k* using a 30-ms triangular analysis window, created using MATLAB function *triang.m*, at a 15-ms frame interval. Then reconstruct the original waveform from the STFT using the OLA approach to synthesis. Ignore tapering end effects of the first and last frames.

(b) Design in MATLAB an OLA-based synthesis method to time-scale expand the speech signal by a factor of two by repeating every frame.

(c) Repeat parts (a) and (b) using the LSE-based synthesis approach. Compare the time-scaled signals from each approach and comment on the differences.

(d) Repeat parts (b) and (c) using "pitch-synchronized" OLA- and LSE-based synthesis approaches. This will require that you estimate a consistent time instant (e.g., the waveform peak or the glottal pulse time) within a glottal cycle in order to synchronize consecutive frames. How does this approach improve your synthesis from parts (b) and (c)?

(e) Repeat parts (a)–(d) with a speech utterance from your own voice recording. For part (d), consider manually marking voiced and unvoiced regions and invoke pitch synchrony during the voiced regions.

7.23 (MATLAB) In this MATLAB exercise, use the workspace *ex7M2.mat*, as well as the function *uniform_bank.m*, both located in companion website directory Chap_exercises/chapter7. This exercise explores the conditions for recovery of a sequence using the filter bank summation (FBS) method.

(a) There are four different analysis windows (or "analysis filters") in the workspace *ex7M2.mat*: *filter1*, *filter2*, *filter3*, and *filter4*. Using the MATLAB command *subplot*, plot all four filters. Note that *filter4* is simply a shifted version of *filter3*.

(b) The function *uniform_bank.m* creates a filter bank $h_k[n]$. The output of the function is a 2-D array of modulated filter impulse responses, but without the demodulation term $e^{-j\frac{2\pi}{N}kn}$ shown in Figure 7.5. Also, the impulse responses are real because the complex conjugate response pairs have been combined. Note that the first and last filters do not correspond to complex conjugate pairs. Explain why.

(c) Do a *help* command on *uniform_bank.m* and run the function with a 250-Hz spacing between filters, the analysis filter *filter1*, and a plot factor of 100 (scaling the output for a good display). The function plots the frequency response magnitude of the filters using a 1024-point DFT. Assume the time sampling rate is 10000 samples/s so that the 512th frequency bin corresponds to 5000 Hz. Having the impulse responses $h_k[n]$ of your filter bank, write a MATLAB function to filter the unit sample *impulse* in *ex7M2.mat* with your filter bank, and then combine all impulse responses to create the composite (summed) filter bank impulse response. Explain the observed composite impulse response using the FBS constraint. Finally, plot the impulse response to the second and fifteenth bandpass filters and explain the difference in time structure of the two signals [recall that the filter bank is missing the STFT demodulation term $e^{-j\frac{2\pi}{N}kn}$ (Figure 7.5)].

(d) Repeat part (c) with analysis filter *filter2*. Using the FBS constraint, explain the reason for the deviation in the composite response from an impulse.

(e) Repeat part (c) with *filter3* (using a plot factor of 2). Superimpose the composite impulse response on *filter3* (plotting the first 400 samples) and again explain your observation using the FBS constraint.

(f) Repeat part (c) with *filter4*, which is *filter3* shifted by fifty samples to the right. Superimpose the composite impulse response on *filter4* (plotting the first 450 samples) and again explain your observation using the FBS constraint. Also comment on the difficulty in creating an impulsive composite response by simply shifting the analysis filter.

BIBLIOGRAPHY

[1] J.B. Allen and L.R. Rabiner, "A Unified Theory of Short-Time Spectrum Analysis and Synthesis," *Proc. IEEE*, vol. 65, no. 11, pp. 1558–1564, Nov. 1977.

[2] J.C. Anderson, *Speech Analysis/Synthesis Based on Perception*, Ph.D. Thesis, Massachusetts Institute of Technology, Dept. of Electrical Engineering and Computer Science, Cambridge, MA, Sept. 1984.

[3] S.F. Boll, "Suppression of Acoustic Noise in Speech Using Spectral Subtraction," *IEEE Trans. Acoustics, Speech, and Signal Processing*, vol. ASSP–27, no. 2, pp. 113–120, April 1979.

[4] R.E. Crochiere, "A Weighted Overlap-Add Method of Short-Time Fourier Analysis/Synthesis," *IEEE Trans. Acoustics, Speech and Signal Processing*, vol. ASSP–28, no. 1, pp. 99–102, Feb. 1980.

[5] G. Fairbanks, W.L. Everitt, and R.P. Jaeger, "Method for Time or Frequency Compression-Expansion of Speech," *IEEE Trans. Audio and Electroacoustics*, vol. AU–2, pp. 7–12, Jan. 1954.

[6] D. Griffin and J.S. Lim, "Signal Estimation from Modified Short-Time Fourier Transform," *IEEE Trans. Acoustics, Speech, and Signal Processing*, vol. ASSP–32, no. 2, pp. 236–243, April 1984.

[7] D. Israelevitz, "Some Results on the Time-Frequency Sampling of the Short-Time Fourier Transform Magnitude," *IEEE Trans. Acoustics, Speech, and Signal Processing*, vol. 33, no. 6, pp. 1611–1613, Dec. 1985.

[8] W. Koening, H.K. Dunn, and L.Y. Lacey, "The Sound Spectrogram," *J. Acoustical Society of America*, vol. 18, pp. 19–49, Feb. 1946.

[9] J. Laroche, "Time and Pitch Scale Modification of Audio Signals," chapter in *Applications of Digital Signal Processing to Audio and Acoustics*, M. Kahrs and K. Brandenburg, eds., Kluwer Academic Publishers, Boston, MA, 1998.

[10] J.S. Lim and A.V. Oppenheim, "Enhancement and Bandwidth Compression of Noisy Speech," *Proc. IEEE*, vol. 67, no. 12, pp. 1586–1604, Dec. 1979.

[11] E. Moulines and F. Charpentier, "Pitch-Synchronous Waveform Processing Techniques for Text-to-Speech Synthesis Using Diphones," *Speech Communication*, vol. 9, no. 5–6, pp. 453–467, 1990.

[12] S.H. Nawab, T.F. Quatieri, and J.S. Lim, "Signal Reconstruction from Short-Time Fourier Transform Magnitude," *IEEE Trans. Acoustics, Speech, and Signal Processing*, vol. ASSP–31, no. 4, pp. 986–998, Aug. 1983.

[13] S.H. Nawab and T.F. Quatieri, "Short-Time Fourier Transform," Chapter in *Advanced Topics in Signal Processing*, J.S. Lim and A.V. Oppenheim, eds., Prentice Hall, Englewood Cliffs, NJ, Oct. 1987.

[14] A.V. Oppenheim and R.W. Schafer, *Discrete–Time Signal Processing*, Prentice Hall, Englewood Cliffs, NJ, 1989.

[15] R.K. Potter, G.A. Kopp, and H.G. Kopp, *Visible Speech*, Dover Publications, Inc., New York, 1966.

[16] M.R. Portnoff, "Time-Scale Modification of Speech Based on Short-Time Fourier Analysis," *IEEE Trans. Acoustics, Speech,and Signal Processing*, vol. ASSP–30, no. 3, pp. 374–390, June 1981.

[17] M.R. Portnoff, "Time-Frequency Representation of Digital Signals and Systems Based on Short-Time Fourier Analysis," *IEEE Trans. Acoustics, Speech, and Signal Processing*, vol. ASSP–28, no. 1, pp. 55–69, Feb. 1980.

[18] M.R. Portnoff, "Implementation of the Digital Phase Vocoder Using the Fast Fourier Transform," *IEEE Trans. Acoustics, Speech, and Signal Processing*, vol. ASSP–24, no. 3, pp. 243–248, Feb. 1976.

[19] T.F. Quatieri, S.H. Nawab, and J.S. Lim, "Frequency Sampling of the Short-Time Fourier Transform Magnitude for Signal Reconstruction," *J. Optical Society of America*, Special Issue on Signal Recovery, vol. 73, pp. 1523–1526, Nov. 1983.

[20] L.R. Rabiner and R.W. Schafer, *Digital Processing of Speech Signals*, Prentice Hall, Englewood Cliffs, NJ, 1978.

[21] S. Roucos and A.M. Wilgus, "High-Quality Time-Scale Modification of Speech," *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Tampa, FL, pp. 493–496, April 1985.

[22] R. Scott and S. Gerber, "Pitch-Synchronous Time-Compression of Speech," *Proc. Conf. Speech Communications Processing*, pp. 63–65, April 1972.

[23] V.W. Zue, "Acoustic-Phonetic Knowledge Representation: Implications from Spectrogram Reading Experiments," *Proc. 1981 NATO Advanced Summer Institute on Automatic Speech Analysis and Recognition*, Bonas, France, 1981.