

Notes on DFT and Spectrograms

Marta Campi

July 30, 2019

1 Discrete Fourier Transform

Intuition:

The DFT is a mathematical process which analyses the time-domain signal $x(n)$ in order to extract that signal's frequency content or frequency domain information denoted as $X(k)$. It operates by comparing or correlating the signal being analysed $x(n)$ against a signal known as the sinusoidal basis function or sinusoidal waveform. The comparison is achieved using a mathematical process called correlation. Correlation can be thought as being a measure of similarity between signals or as a measure of how strongly present one signal is in another signal. In this example, we have a time-domain signal $x(n)$ being compared against a DFT analysis basis functions which is a cosine waveform which has exactly one cycle over N samples, where N represents the number of samples in the signal being analysed. In this case the output of the correlation is producing a numerical value of 1.6 and the magnitude of this correlation measurement gives an indication of how strongly present in our signal the analysis basis function is in the time-domain signal being analysed. So the greater the magnitude of the correlation measurement, the most strongly present the analysis basis function is in the time-domain signal being analysed. The analysis basis functions used have the same length as the original signal. They are also limited to having an integer number of cycles over the duration of the signal being analysed. The sinusoidal basis functions are both sine waveforms and cosine waveforms.

The result of the comparison of the signal being analysed with the sine wave basis function is stored as a complex number in the correlation measurement. The overall correlation result from both the cosine waveform and the sine waveform is then stored as a complex value in DFT bin one denoted as $X(1)$ (usually this will be $X(k)$). The magnitude of this complex number or the correlation result that is then used to produce the magnitude spectrum.

This comparison continues on for sinusoidal basis functions with different frequencies. So $x(n)$ will be compared against basis functions which have 2 cycles over N samples giving $X(2)$, 3 cycles over N samples giving $X(3)$ and so on. The comparison with the cosine basis function will be always stored as a real value in $X(k)$, while the comparison with the sine basis function will be always stored as an imaginary value in $X(k)$. The overall complex result is stored as a DFT value and is denoted by $X(k)$. The index k representing the bin number of the frequency represents the number of cycles associated with the basis function. The definition of the DFT is given as follows:

$$X(k) = \sum_{n=0}^{N-1} \left(x(n) e^{-j \frac{2\pi nk}{N}} \right) \quad (1)$$

The DFT transforms N discrete-time sample to the same number of discrete frequency samples. The inverse transformation is denoted as IDFT. Properties:

- Periodicity
- Circular shift
- Time Reversal
- Complex Conjugate
- Circular Convolution Property
- Multiplication Property
- Parseval's Theorem
- (Symmetry)

2 Spectrum analysis

2.1 Discrete-time Fourier Transform (DTFT)

$$X(\omega) = \sum_{n=-\infty}^{\infty} \left(x(n) e^{-j\omega n} \right) \quad (2)$$

Relationship between DFT and DTFT: The DFT gives the discrete-time Fourier series coefficients of a periodic sequence $x(n) = x(n + N)$ of period N samples, or

$$X(\omega) = \frac{2\pi}{N} \sum \left(X(k) \delta \left(\omega - \frac{2\pi k}{N} \right) \right) \quad (3)$$

as can easily be confirmed by computing the inverse DTFT of the corresponding line spectrum:

$$\begin{aligned} x(n) &= \frac{1}{2\pi} \int_{-\pi}^{\pi} \left(\frac{2\pi}{N} \sum X(k) \delta \left(\omega - \frac{2\pi k}{N} \right) \right) e^{j\omega n} d\omega \\ &= \frac{1}{N} \sum_{k=0}^{N-1} \left(X(k) e^{+j \frac{2\pi n k}{N}} \right) \\ &= IDFT(X(k)) \\ &= x(n) \end{aligned} \quad (4)$$

The DFT can thus be used to exactly compute the relative values of the N line spectral components of the DTFT of any periodic discrete-time sequence with an integer-length period.

2.2 DFT and DTFT of finite-length data

When a discrete-time sequence happens to equal 0 for all samples except for those between 0 and $N - 1$, the infinite sum in the DTFT equation becomes the same as the finite sum from 0 to $N - 1$ in the DFT equation. By matching the terms in the exponential terms, we observe that the DFT values exactly equal the DTFT for specific DTFT frequencies $\omega_k = \frac{2\pi k}{N}$. That is, the DFT computes exact sample of the DTFT at N equally spaced frequencies $\omega_k = \frac{2\pi k}{N}$, or:

$$X\left(\omega_k = \frac{2\pi k}{N}\right) = \sum_{n=-\infty}^{\infty} \left(x(n)e^{-j\omega_k n}\right) = \sum_{n=0}^{N-1} \left(x(n)e^{-j\frac{2\pi k n}{N}}\right) = X(k) \quad (5)$$

2.3 DFT as DTFT approximation

In most cases, the signal is neither exactly periodic nor truly of finite length; in such cases, the DFT of a finite block of N consecutive discrete-time samples does not exactly equal samples of the DTFT at specific frequencies. Instead, the DFT gives frequency samples windowed (truncated) DTFT

$$\hat{X}\left(\omega_k = \frac{2\pi k}{N}\right) = \sum_{n=0}^{N-1} \left(x(n)e^{-j\omega_k n}\right) = \sum_{n=-\infty}^{\infty} \left(x(n)w(n)e^{-j\omega_k n}\right) = X(k) \quad (6)$$

where

$$w(n) = \begin{cases} 1 & \text{if } 0 \leq n \leq N \\ 0 & \text{otherwise} \end{cases} \quad (7)$$

$X(k)$ exactly equals $X(\omega_k)$ a DTFT frequency sample only when $x(n) = 0, n \notin [0, N - 1]$.

2.4 Relationship between continuous-time FT and DFT

The goal of spectrum analysis is often to determine the frequency content of an analog (continuous-time) signal; very often, as in most modern spectrum analyzers, this is actually accomplished by sampling the analog signal, windowing (truncating) the data, and computing and plotting the magnitude of its DFT. It is thus essential to relate the DFT frequency samples back to the original analog frequency. Assuming that the analog signal is bandlimited and the sampling frequency exceeds twice the limit so that no frequency aliasing occurs, the relationship between the continuous-time Fourier frequency Ω (in radians) and the DTFT frequency ω imposed by sampling is $\omega = \Omega T$ where T is the sampling period. Through the relationship $\omega_k = \frac{2\pi k}{N}$ between the DTFT frequency ω and the DFT frequency index k , the correspondence between the DFT frequency index and the original analog frequency can be found:

$$\Omega = \frac{2\pi k}{NT} \quad (8)$$

or in terms of analog frequency f in Hertz (cycles per second rather than radians)

$$f = \frac{k}{NT} \quad (9)$$

for k in the range k between 0 and $\frac{N}{2}$. It is important to note that $k \in [\frac{N}{2} + 1, N - 1]$ correspond to negative frequencies due to periodicity of the DTFT and the DFT.

2.5 Zero-padding

If more than N equally spaced frequency samples of a length- N signal are desired, they can be easily be obtained by zero-padding the discrete-time signal and computing a DFT of the longer length. In particular, if LN DTFT samples are desired of a length- N sequence, one can compute the length- LN DFT of a length- LN zero-padded sequence

$$z(n) = \begin{cases} x(n) & \text{if } 0 \leq n \leq N-1 \\ 0 & \text{if } N \leq n \leq LN-1 \end{cases} \quad (10)$$

$$X\left(w_k = \frac{2\pi k}{LN}\right) = \sum_{n=0}^{N-1} \left(x(n)e^{-j\frac{2\pi kn}{LN}}\right) = \sum_{n=0}^{LN-1} \left(z(n)e^{-j\frac{2\pi kn}{LN}}\right) = DFT_{LN}[z[n]] \quad (11)$$

Note that zero-padding interpolates the spectrum. One should always zero-pad (by about at least a factor of 4) when using the DFT to approximate the DTFT to get a clear picture of the DFTF. While performing computation on zeros may seem inefficient, using FFT algorithms, which generally expect the same number of input and output samples, actually makes this approach very efficient.

2.6 Effects of windowing

Applying the DTFT multiplication property

$$X(\hat{\omega}_k) = \sum_{n=-\infty}^{\infty} \left(x(n)w(n)e^{-j\omega_k n}\right) = \frac{1}{2\pi} X(\omega_k) * W(\omega_k) \quad (12)$$

we find that the DFT of the windowed (truncated) signal produces samples not of the true (desired) DFTF spectrum $X(\omega)$, but of a smoothed version $X(\omega) * W(\omega)$. We want to resemble $X(\omega)$ as closely as possible, so $W(\omega)$ should be as close to an impulse as possible. The window $w(n)$ need not be a simple truncation (or rectangle or boxcar) window; other shapes can also be used as long as they limit the sequence to at most N consecutive non-zero samples. All good windows are impulse-like, and represent various tradeoffs between three criteria: (1) main lobe width: (limits resolution of closely-spaced peaks of equal height); (2) height of the first sidelobe: (limits ability to see a small peak near a big peak); (3) slope of sidelobe drop-off: (limits ability to see small peaks further away from a big peak).

3 Classical Statistical Spectral Estimation

Many signals are either or partly or wholly stochastic, or random. Such signals may have a distinct average spectral structure that reveals important information. Spectrum analysis of any single block of data using window-based deterministic spectrum analysis, however, produces a random spectrum that may difficult to interpret. For such situation, the classical statistical spectrum estimation methods described in this module can be used.

The goal in classical statistical spectrum analysis is to estimate $E[|X(\omega)|^2]$, the power spectral

density (PSD) across frequency of the stochastic signal. That is, the goal is to find the expected (mean, or average) energy density of the signal as a function of frequency. Since the spectrum of each block of signal samples is itself random, we must average the squared spectral magnitudes over a number of blocks of data to and the mean. There are two main classical approaches, the periodogram and auto-correlation method.

3.1 Periodogram

The periodogram method divides the signal into a number of shorter (and often overlapped) blocks of data, computes the squared magnitude of the windowed DFT, $X_i(\omega_k)$, of each block, and averages them to estimate the power spectral density. The squared magnitudes of the DFTs of L possibly overlapped length- N windowed blocks of signal are averaged to estimate the PSD:

$$\hat{X}(\omega_k) = \frac{1}{L} \sum_{i=1}^L \left(|X_i(\omega_k)|^2 \right) \quad (13)$$

For a fixed total number of samples, this introduces a tradeoff: larger individual data blocks provides better frequency resolution due to the use of a longer window, but it means there are less blocks to average, so the estimate has higher variance and appears more noisy. The best tradeoff depends on the application. Overlapping blocks by a factor of two to four increases the number of averages and reduces the variance, but since the same data is being reused, still more overlapping does not further reduce the variance. **IMPORTANT:** the periodogram produces an estimate of the windowed spectrum that is $\hat{X}(\omega_k) = E \left[(|X(\omega) * W_M|)^2 \right]$ not $E \left[(|X(\omega)|)^2 \right]$.

3.2 Auto - Correlation based approach

The averaging necessary to estimate a power spectral density can be performed in the discrete-time domain, rather than in frequency, using the auto-correlation method. The squared magnitude of the frequency response, from the DTFT multiplication and conjugation properties, corresponds in the discrete-time domain to the signal convolved with the time-reverse of itself,

$$\left((|X(\omega)|)^2 \right) = X(\omega)X^*(\omega) \iff (x(n), x^*(-n)) = r(n) \quad (14)$$

or its auto-correlation function

$$r(n) = \sum x(k)x^*(n+k) \quad (15)$$

We can thus compute the squared magnitude of the spectrum of a signal by computing the DFT of its auto-correlation. For stochastic signals, the power spectral density is an expectation, or average, and by linearity of expectation can be found by transforming the average of the auto-correlation. For a finite block of N signal samples, the average of the autocorrelation values, $r(n)$, is:

$$r(n) = \frac{1}{N-n} \sum_{k=0}^{N-(1-n)} (x(k)x^*(n+k)) \quad (16)$$

Note that with increasing lag, n , fewer values are averaged, so they introduce more noise into the estimated power spectrum. By windowing the auto-correlation before transforming it to the

frequency domain, a less noisy power spectrum is obtained, at the expense of less resolution. The multiplication property of the DTFT shows that the windowing smooths the resulting power spectrum via convolution with the DTFT of the window:

$$\hat{X}(\omega) = \sum_{n=-M}^M (r(n)w(n)e^{-j\omega n}) = \left(E \left[(|X(\omega)|)^2 \right] \right) * W(n) \quad (17)$$

This yields another important interpretation of how the auto-correlation method works: it estimates the power spectral density by averaging the power spectrum over nearby frequencies, through convolution with the window function's transform, to reduce variance. Just as with the periodogram approach, there is always a variance vs. resolution tradeoff. The periodogram and the auto-correlation method give similar results for a similar amount of averaging; the user should simply note that in the periodogram case, the window introduces smoothing of the spectrum via frequency convolution before squaring the magnitude, whereas the periodogram convolves the squared magnitude with $W(n)$.

3.3 Short Time Fourier Transform

The Fourier transforms (FT, DTFT, DFT, etc.) do not clearly indicate how the frequency content of a signal changes over time. That information is hidden in the phase - it is not revealed by the plot of the magnitude of the spectrum. Note: To see how the frequency content of a signal changes over time, we can cut the signal into blocks and compute the spectrum of each block. To improve the result

1. blocks are overlapping
2. each block is multiplied by a window that is tapered at its endpoints

Several parameters must be chosen:

- Block length, R .
- The type of window.
- Amount of overlap between blocks. (Figure 2.15 (STFT: Overlap Parameter))
- Amount of zero padding, if any.

The short-time Fourier transform is defined as

$$\begin{aligned} X(\omega, m) &= (STFT(x(n))) := DTFT(x(n-m)w(n)) \\ &= \sum_{n=-\infty}^{\infty} (x(n-m)w(n)e^{-j\omega n}) \\ &= \sum_{n=0}^{R-1} (x(n-m)w(n)e^{-j\omega n}) \end{aligned} \quad (18)$$

where $w(n)$ is the window function of length R .

1. The STFT of a signal $x(n)$ is a function of two variables: time and frequency.
2. The block length is determined by the support of the window function $w(n)$.
3. A graphical display of the magnitude of the STFT, $|X(\omega, m)|$, is called the spectrogram of the signal. It is often used in speech processing.
4. The STFT of a signal is invertible.
5. One can choose the block length. A long block length will provide higher frequency resolution (because the main-lobe of the window function will be narrow). A short block length will provide higher time resolution because less averaging across samples is performed for each STFT value.
6. A narrow-band spectrogram is one computed using a relatively long block length R , (long window function).
7. A wide-band spectrogram is one computed using a relatively short block length R , (short window function).

- Short Time Fourier Transform

To numerically evaluate the STFT, we sample the frequency axis ω in N equally spaced samples from $\omega = 0$ to $\omega = 2\pi$

$$\omega_k = \frac{2\pi}{N}k, \quad 0 \leq k \leq N-1 \quad (19)$$

We then have the discrete STFT

$$\begin{aligned} \left(X^d(k, m) := X\left(\frac{2\pi}{N}k, m\right) \right) &= \sum_{n=0}^{R-1} \left(x(n-m)w(n)e^{j\omega_k n} \right) \\ &= \sum_{n=0}^{R-1} \left(x(n-m)w(n)W_N^{-(kn)} \right) \\ &= DFT_N \left(x(n-m)w(n) \Big|_{n=0}^{R-1}, 0 \dots, 0 \right) \end{aligned} \quad (20)$$

where $0, \dots, 0$ is $N-R$. In this definition, the overlap between adjacent blocks is $R-1$. The signal is shifted along the window one sample at a time. That generates more points than is usually needed, so we also sample the STFT along the time direction. That means we usually evaluate

$$X^d(k, Lm) \quad (21)$$

where L is the time-skip. The relation between the time-skip, the number of overlapping samples, and the block length is:

$$Overlap = R - L \quad (22)$$