Signal Processing Algorithms & Architecture

Dr. Anirban Dasgupta
Department of Electronics & Electrical Engineering
Indian Institute of Technology Guwahati

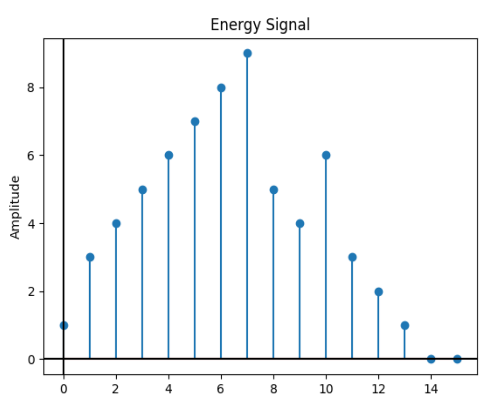
Lec 12: Coherence Analysis

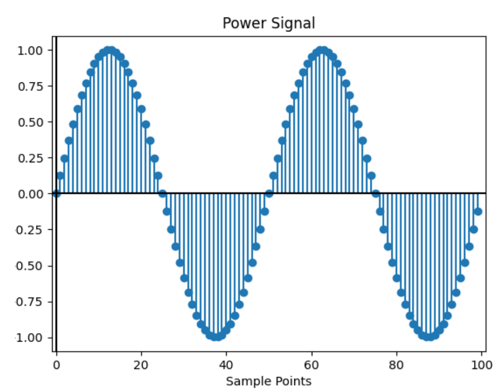
Hello everyone, welcome to a fresh new lecture on the topic of coherence analysis for the course on signal processing algorithms and architecture. This is Dr. Anirban Dasgupta and let us get started. So, we will recap something from module 1, time domain signal processing, where we differentiated power and energy signals. So, we define signal energy as the sum of the squared values of the signal from negative infinity to positive infinity, and if this energy is finite, the signal is called an energy signal. So aperiodic signals of finite length are typically energy signals, this is an example.

$$\text{Signal Energy, } E = \sum_{n=-\infty}^{\infty} |x[n]|^2$$

$$\text{Signal Energy, } P = \lim_{N\to\infty} \frac{1}{2N+1} \sum_{n=-N}^{N} |x[n]|^2$$

Similarly, for periodic signals of infinite duration, we do not define the energy because it will be infinite. So, we define something called power. Power is defined by this relation, where this is the energy thing, but here we are computing over a value n, and then what we are doing is using this limit. So, if we see that the power is finite, then the signal is called a power signal, and take a sine wave, this is just a part of the sine wave, extending to infinity and also starting from negative infinity, so this is a power signal.





The power of an energy signal is 0, and the energy of a power signal is ∞. Now we will introduce a new term, which is energy spectral density. So, energy spectral density is how the energy of a signal is distributed across the different frequency components. So how do we obtain it? So let us say we have an analog signal x(t); of course, this is for the theory. I am taking this analog signal for computational purposes, we will again go back to the discrete-time counterparts.

So, this is a graph where your $\Psi_{xx}(f)$ is your energy spectral density, and this is computed in this manner. So this is kind of based on Parseval's theorem, which states that energy is conserved in time and frequency. So I will take the Fourier transform of the signal and square it.

$$\Psi_{xx}(f) = | X_T(f) |^2$$

And why do I come from the Fourier? Because I want to get one for each frequency component. So, power signals have infinite energy as already discussed.

So, the Fourier transform, and your energy spectral density may not exist for a power signal. May also exist, but the problem is how do we get such information? So, for this, we need an alternate spectral density definition with similar properties, such as the energy spectral density, and here comes the power spectral density to the rescue. So, power spectral density describes how the power of the signal is distributed across different frequencies. So, if you have a signal x(t), the PSD is usually estimated using different methods, such as the periodogram or the Welch method. We have already discussed the FFT-based PSD estimation.

So, how do we define the PSD? So the PSD is defined by the formula

$$P_{xx}(f) = \lim_{T \to \infty} \frac{1}{T} E[| X_T(f) |^2]$$

Now, why is this expectation operator used? Now typically this will account for stochastic signals as well. If the signal is deterministic, then that realization is the same as the expected realization. But if the signal is stochastic, there can be several realizations, so we take the expectation. Typically, what happens is, as I mentioned, that the power spectral is for periodic signals or power signals, and practically, you cannot capture the entire power signal because it goes from $-\infty$ to $+\infty$, and we do not know what $-\infty$ and $+\infty$ are.

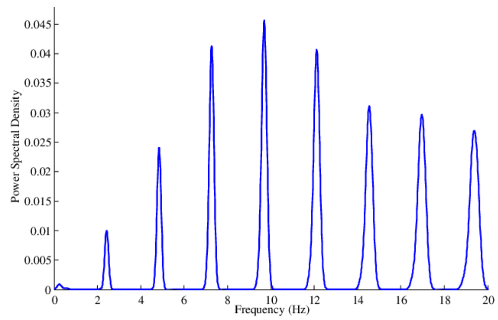
We will take a time-limited version, and hence, because of windowing and other factors, we get a noisy spectrum. So, this is typically corrected or estimated using Welch's method and what is this power spectral density related to the signals? It is the Fourier transform of the autocorrelation function. So, you can find the autocorrelation function that we have already discussed in the time domain. You take the Fourier transform; what you get is the power spectral density.

And autocorrelation again, so for a stochastic signal, you can get it using the formula

$$P_{xx}(f) = F\{R_{xx}(\tau)\}$$

$$R_{xx}(\tau) = E[x(t)x(t+\tau)]$$

where τ is the lag between the two points in the signal in the time domain. Well, this is an example of a power spectral density,



which says that these are the dominant frequencies. Presenting these frequencies, some have very high power, while others have no component or power, and this is usually obtained using the Welch method. So that is why we get a clean, smooth estimate of the PSD, and this PSD is also useful in finding the dominant frequencies in the signal. That is the frequency estimation problem.

So well, like correlation had two forms: in the time domain, autocorrelation and cross-correlation. Here, we also have the auto power spectral density (APSD) and cross power spectral density (CPSD).

APSD: $$P_{XX}\,(f) = \lim_{T\to\infty}\left|\int_{-\frac{T}{2}}^{\frac{T}{2}} x(t)e^{-j2\pi ft}\,dt\right|^2$$

CPSD: $$P_{XY}\,(f) = \lim_{T\to\infty}\left|\int_{-\frac{T}{2}}^{\frac{T}{2}} x(t)e^{-j2\pi ft}\,dt\right|.\left|\int_{-\frac{T}{2}}^{\frac{T}{2}} y(t)e^{-j2\pi ft}\,dt\right|^*$$

So, autocorrelation, if you see this, is analogous to the APSD, and this represents how the signal is distributed across different frequency components, right? What about the cross-power spectral density? Now here, what we are doing is like in autocorrelation; we are calculating the power of the same signal multiplied by itself twice. But in the cross, we are using two signals, and one signal we are finally conjugating. So now we will come to the term "coherence".

So, coherence is a measure of correlation between two signals, x(t) and y(t), in the frequency domain. So, this is kind of a normalized version of the cross-power spectral density between two signals. Just as we have a normalized version of the Pearson correlation coefficient in the time domain, coherence is somewhat analogous to this concept. So this is the formula for calculating coherence.

$$C_{xy}(f) = \frac{|P_{xy}(f)|^2}{P_{xx}(f)P_{yy}(f)} \;;\; 0 \leq C_{xy}(f) \leq 1$$

So $C_{xy}(f)$ is the ratio of the square of the cross power spectral density divided by the product of the auto power spectral densities of signal x and signal y.

This value will be between the range of 0 and 1, where 1 indicates high coherence and 0 indicates low coherence. Let us see an example.

$$x(t)=\sin(2\pi 40t)$$

$$y(t)=\sin(2\pi 50t)$$

So if we have a signal of frequency 40 hertz and another signal of a 50 hertz frequency. So, if we see the power spectral density, $P_{XX}(f)$will have a peak at 40 and $P_{yy}(f)$will have a peak at 50. But if we look at the cross power spectral density, no frequency components are common. So, the coherence will be zero. So, this is an indication of how my common frequencies are linearly related to each other. So, this is what I just showed or told you in the previous example. So, we have a sine wave, and of course, this is a windowed version of a sine wave; again, you cannot technically have the full sine wave. So, this is a 40 hertz sine wave, and this is the 50 hertz sine wave, and this is the cross-correlation, so this is in the time domain.

So, what happens in the frequency domain is that the cross coherence ideally should yield a value that is nearly 0, but because of this windowing effect, you see that there are small fluctuations in the frequency spectrum or the coherent spectrum. So, what is the intuition and meaning of coherence? So, this is essentially a frequency domain correlation, and it will show how the phase and amplitude of two signals are related at different frequencies. So, in correlation, we see how they are related in time; this is a relation in frequencies. So, if you have a high coherence at a certain frequency, that indicates that at this frequency there is a strong linear relationship between the two signals. and which technically means that one signal can be predicted from the other signal at that specific frequency.

On the other hand, low or zero coherence at a specific frequency means that the signals are uncorrelated at that frequency, or changes in one signal are not linearly related to the changes in the other signal at that specific frequency. So, if there is high coherence, for example, if we are recording brain signals and have recorded brain waves or EEG signals from two regions, and we are interested in a specific band, say the alpha band, which is typically 8 to 12 Hertz. Now, those who are into EEG processing might appreciate this factor; otherwise, understand that the alpha band is a specific frequency band in this frequency range in the EEG and if there is a high coherence between two regions, such as the frontal region and the occipital region, it means there is functional connectivity between those regions. Low coherence: where is it useful? So, suppose we have a communication system with low coherence between the transmitted and received signals; then it can be an indication that there is noise or interference that has affected certain frequency ranges.

So, there are several applications of coherence, like correlation. So, neuroscience is one example I just gave where you can see the different brain regions and see for which activities, like if you are watching a movie and watching a lecture, the brain will react similarly. So, this coherence can give you a lot of indications regarding this, or if you are thinking, and if you are meditating, and if you are focusing or concentrating on someone, how the coherence between the different signals will talk about the mental state. Then vibration analysis and structural health monitoring coherence are often used in mechanical systems. So, as I said, this low coherence might indicate that there is some noise in the input and output, or the input and output are affected; if the input and output have low coherence, that means there is some non-linearity and noise introduced in the system.

Now, how do we estimate, as I said, this APSD and CPSD? These are power spectral densities, and because of this windowing, we also have a lot of noise in practical scenarios. So direct calculation using the method that I mentioned may give you a very noisy estimate. So, what the Welch method suggests is that it will divide the signal into overlapping segments, and then you apply a window function to reduce spectral leakage; this window may not be a rectangular window, as you can have different forms of windows, and then the periodograms will be averaged across the windows. So, this will result in a smoother and more reliable estimate of the PSD. This is one example; see, this is an estimate of PSD, which is a standard periodogram, and it denotes two dominant frequencies somewhere around 50 and 150, but there are noisy components that do not really mean anything but noise.

So, Welch method we get a smoother estimate of the PSD. So, you are not sure whether this is noise or if some frequencies are really present here; it is very confusing. But Welch's method will average them out, and you get a smoother estimate. So, this is the algorithm for computing the coherence. So, we start with two discrete time sequences, and now we are back to the discrete time domain, and then we will use Welch's method.

So first we will estimate $P_{XX}(f)$and $P_{YY}(f)$just as mentioned earlier, but this will use the Welch method to get a smoother variant. So, we will use the Fourier transform and the Welch method to compute the APSD. Similarly, we will calculate the CPSD by taking the cross-spectral density, and this is the final formula to obtain the coherence. But practically, what are the main things for which coherence analysis is challenging, and there is still research going on to get a very good estimate? So one is windowing and spectral leakage. So, what is spectral leakage? Spectral leakage is like to explain in a nutshell.

Say that if I have a sine wave, and when I say sine, I mean from minus infinity to plus infinity, I should get a spectrum at, say, if this is a 50 hertz sine wave, I should get a spectrum at 50, 50, and minus 50. I say, just saying, seeing the positive half, I should get a peek at 50. But in real terms in our digital hardware, we get a lobe like this around 50. which means that the peak, which should be only at 50, is spread out around this region of 50. And why does this happen? So, this is called the windowing effect.

But why does this specific spread happen? This happens because we are taking a window of this; a window means a small region of this sine wave, not the full sine wave. So, if the sine wave is x(t), this window is basically multiplying with a rectangular function that has 1 within this range and 0 outside. So technically we are multiplying x(t) with r(t), which is the window function, a rectangular window, and this estimate is the Fourier transform or the spectrum of the combined signal or the product of x(t) and r(t). Now, when we have multiplication in the time domain, we should have convolution in the frequency domain. So, $X(\omega)$ is convolved with my $R(\omega)$, and $X(\omega)$ is the one that has only one peak at 50, or maybe two peaks at 50 and minus 50.

And what is $R(\omega)$? $R(\omega)$ is the Fourier transform of this rectangular window, which is of a sink nature. So this is the Fourier transform amplitude; the magnitude spectra will be something like this. So, if you convolve this with that, you should be getting something like this. To minimize this, instead of a rectangular window, better windows are available, such as the Hamming window and the Hann window, which are applied to segments of the data. But this is also a trade-off regarding how you will have good frequency resolution and what size of window you should select.

Then averaging and smoothing, as I said, this Welch method is smoothing out noise, but it may also smooth out some frequency information. Say there is a fault and that fault has a small frequency component, but the Welch method can totally remove that. So, this is again a big challenge of which window or how much you should smooth; this is a real challenge. The length of the data and frequency resolution is also a challenge or practical consideration that you should keep in mind: should I take very long data or very short data? Also, if I have longer data, I will have a finer frequency resolution. But you might be required to do a lot of computation to get stable estimates.

And in multivariate systems, like, say, multichannel EEG. So, I gave you an example of only two channels, but there can be many channels. So in such a case, how do you find the coherence? So, in such cases, partial coherence is also observed, which will give you the linear relationship between two signals while accounting for the influence of other signals. It is very much synonymous with the concept of mutual inductance, where you have a lot of circuits and how each one will mutually influence the inductance. A lot of inductors are there in the circuit.

So, this is again very much analogous to the analysis of superposition where you remove all components and keep two kinds of things. So, you can remove all effects of other signals, and then you can calculate the coherence. So that is all for this lecture. Thank you all. We will meet again.