TECH 350: DSP

Class IX: The Fourier Transform and its Digital Implementations

LTI Systems Review (Concepts we've Covered)

System Properties

Signal Flow Diagrams

Difference Equations

Sine-Wave Analysis

Frequency Response

Filter Parameters

Filter Topologies

Comb + Allpass Filters

Digital Algorithm Reverb Basics

Impulse Response (FIR vs. IIR)

z-Transform

The Convolution Theorem

Transfer Functions

Zeroes + Poles

Complex Numbers

What is *z*, though?

z is a ***complex number***, a topic that I've been avoiding, but that we necessarily need to dig into a bit before we move on to an actual definition of the Fourier Transform.

Since this class is targeted at electronic musicians, I'm going to introduce complex numbers in bite-size pieces, although some of you already eat them for breakfast.

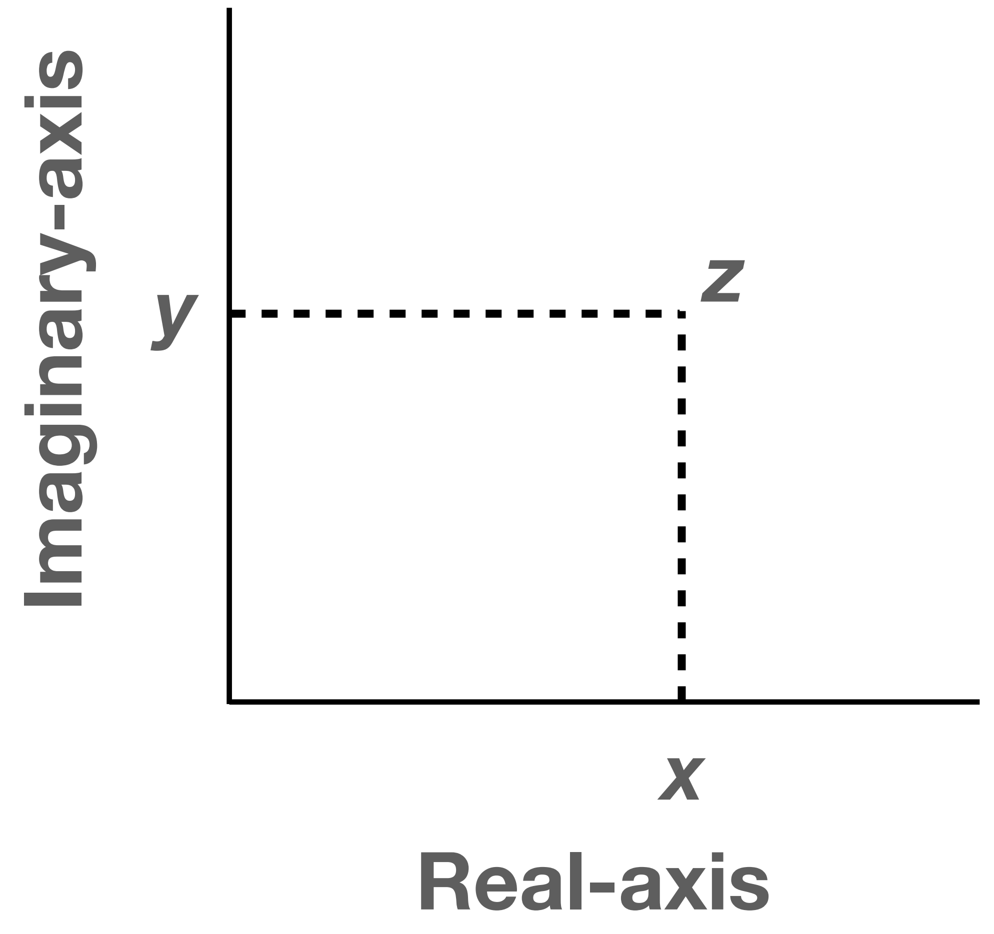
A complex number (in its ***rectangular form***) is written as:

$z = x + yj,$

where x is the ***real*** part of the number and y is the ***imaginary*** part

j, which is equivalent to $\sqrt{-1}$, is called the ***unit imaginary number***

(NOTE: i is used outside of an engineering context)



A Taste of Complex Numbers

Converting the rectangular form ($z = x + yj$) to the polar, we get the following:

$z = r\cos(\theta) + j\,r\sin(\theta)$,

where θ is an angle and $r = |z| = \sqrt{x^2 + y^2}$

(i.e., r is the distance from the origin of the complex number)

This is made more useful by taking advantage of Euler's identity,

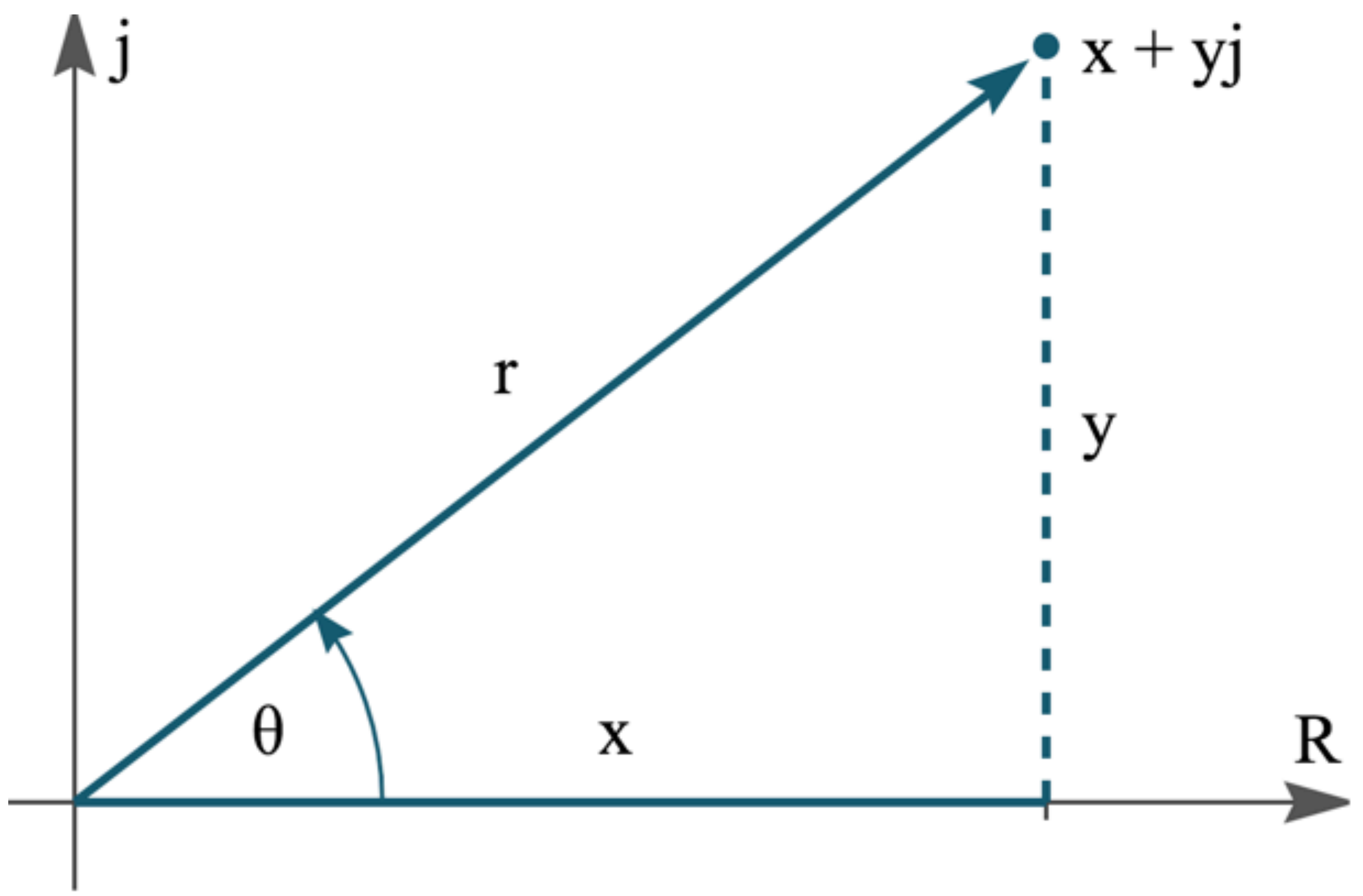
$e^{j\theta} = \cos(\theta) + j\sin(\theta)$,

where e is Euler's number, the base of the natural logarithm,

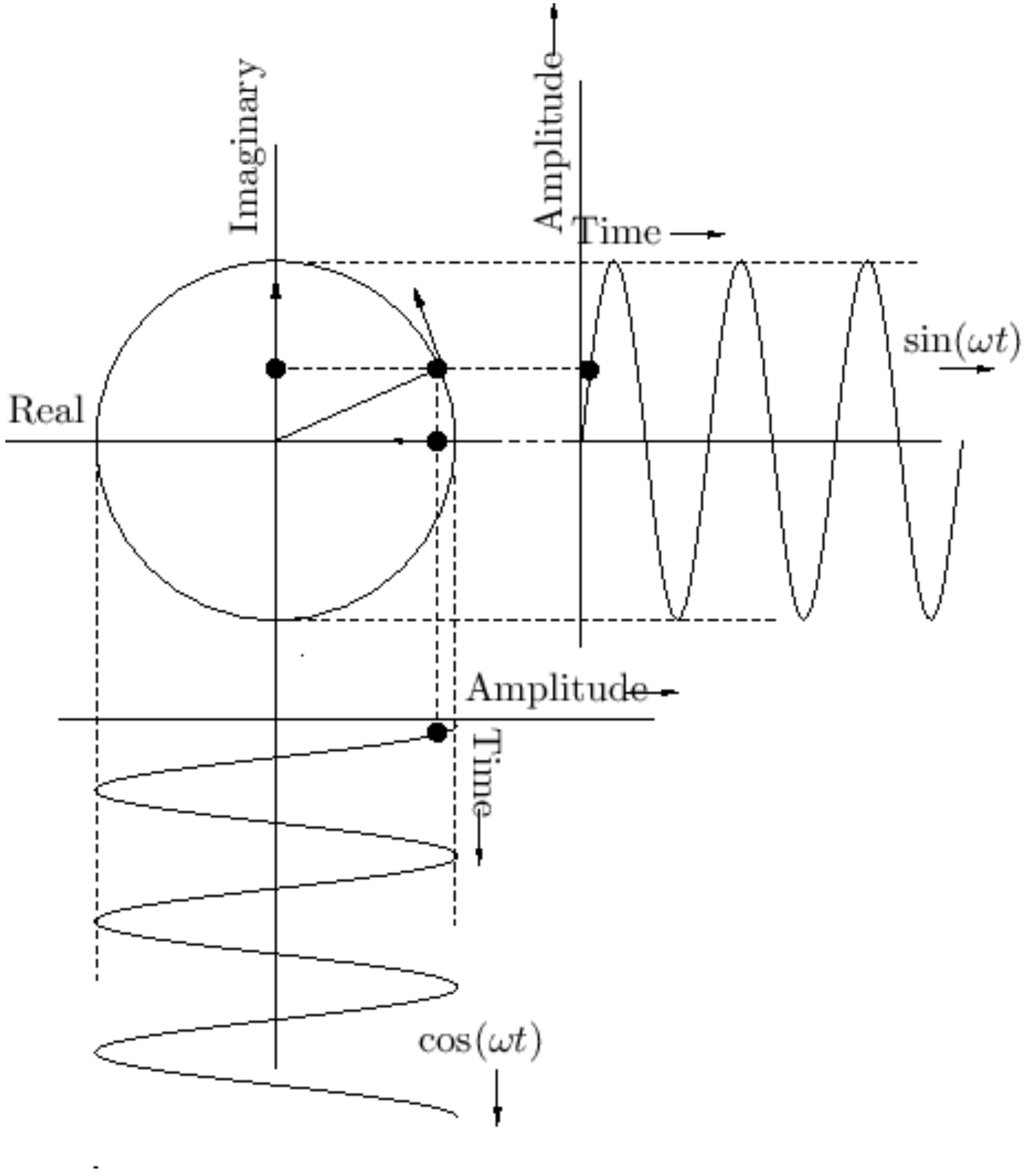
to create the form we'll use to explore the Fourier Transform:

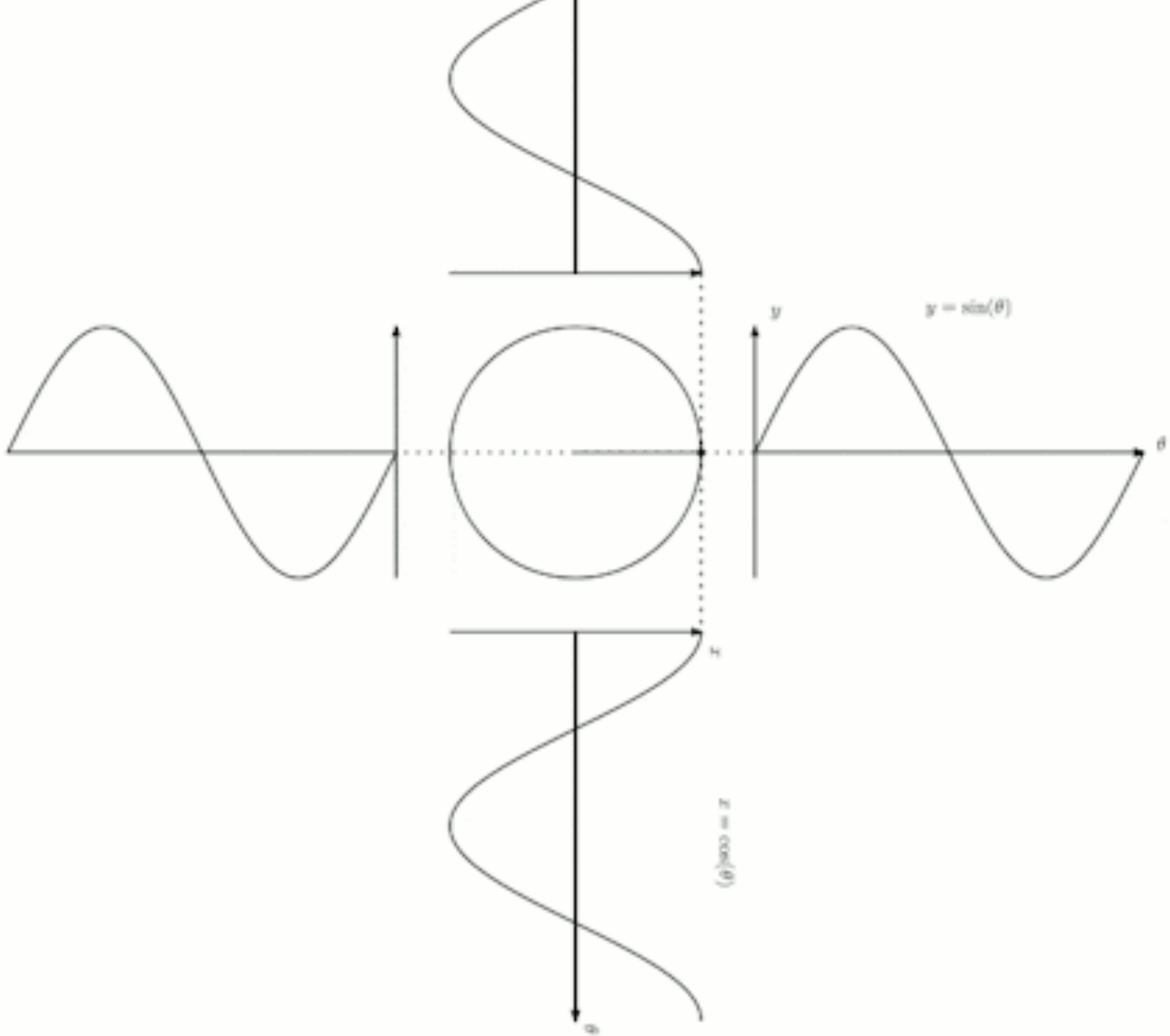
$z = re^{j\theta}$,

the ***polar form (AKA exponential form)*** of the complex number.



Complex Sinusoids

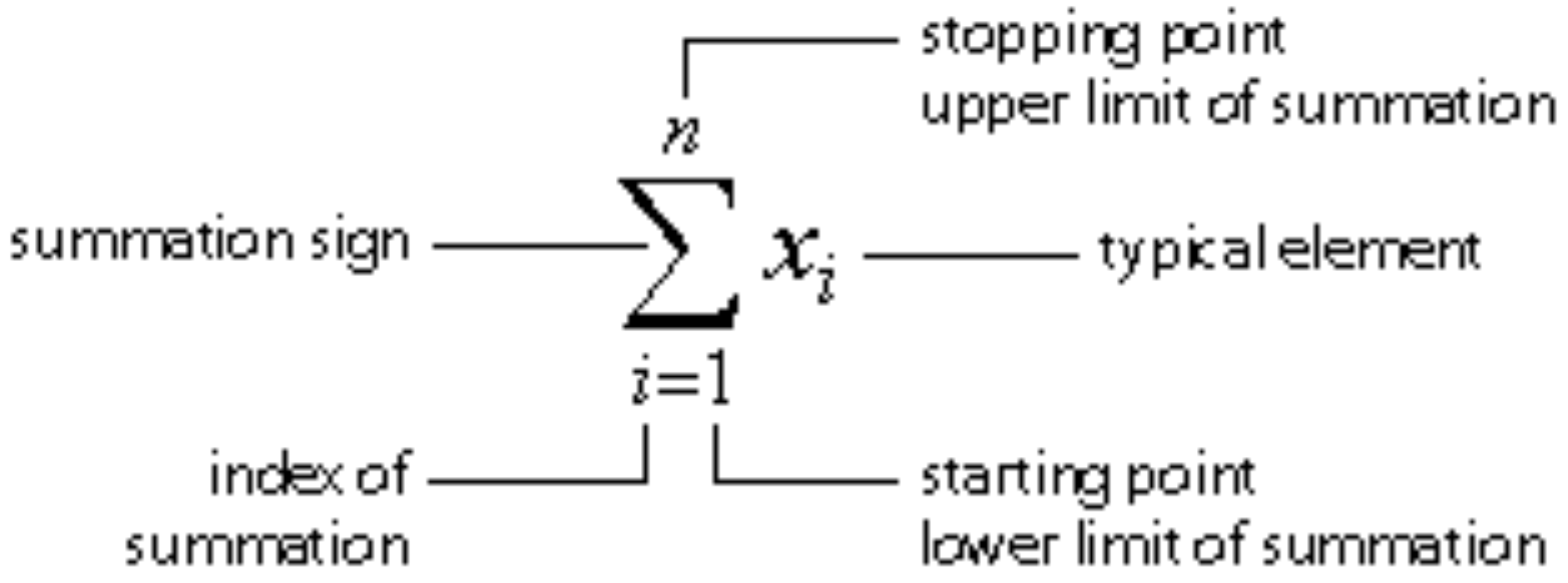




Mathematical Symbology: Summation

Summing: a lot of the procedures we've been doing (convolution, calculating the output of a filter) involve summing terms.

As shorthand for summation (adding together) of terms, we use the following (Big Sigma):



For example,

$$\sum_{n=1}^{5} n^2 = 1^2 + 2^2 + 3^2 + 4^2 + 5^2 = 1 + 4 + 9 + 16 + 25 = 55$$

The Fourier Transform: Take 0

Jean Baptiste Joseph Fourier

Late 1700s researcher on periodic waves and their analysis

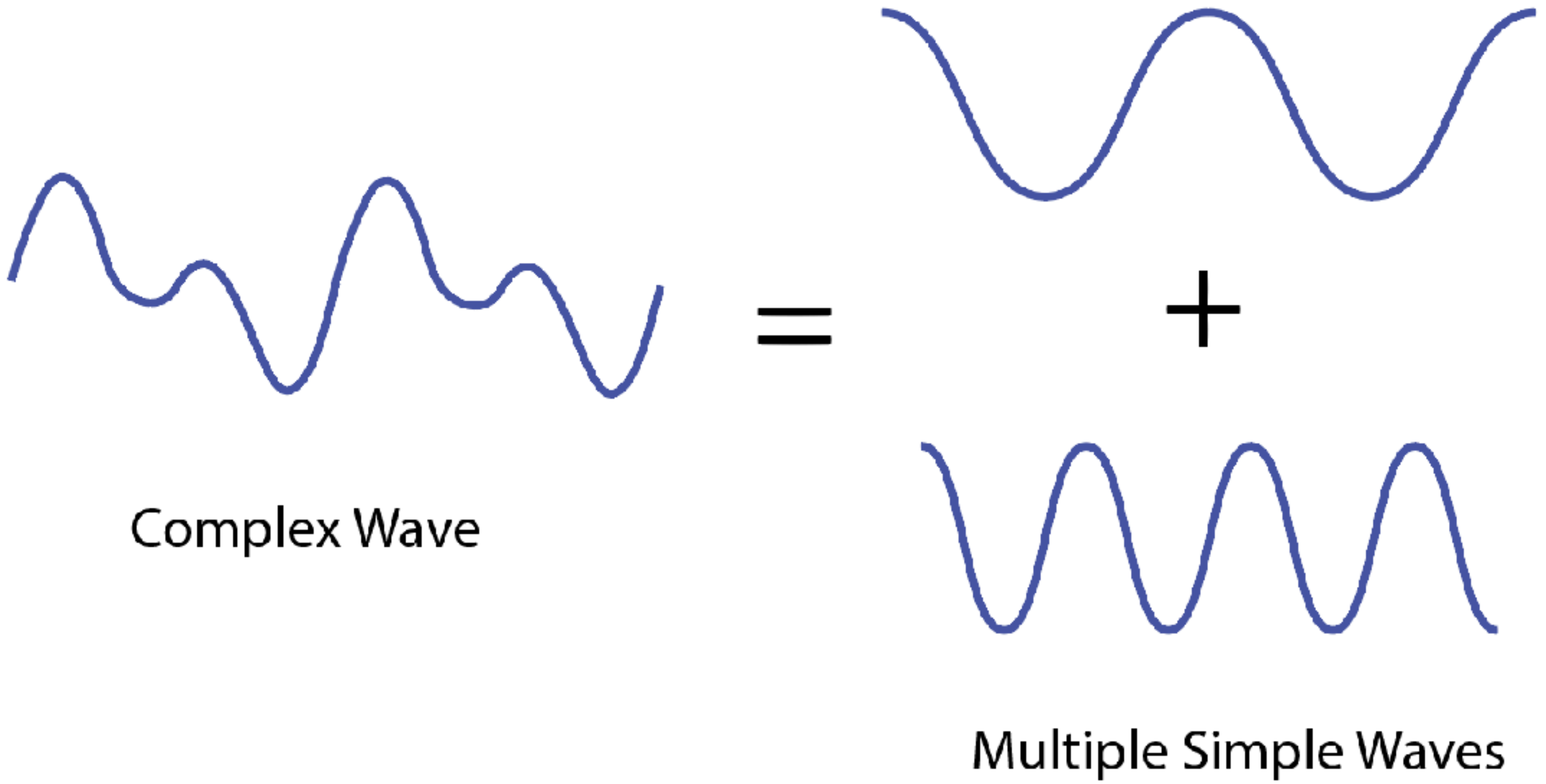
Studied the conductive diffusion of heat

Scientific advisor to Napoleon

Fourier's Theorem:

A complex periodic sound can be decomposed into a set of simple waves

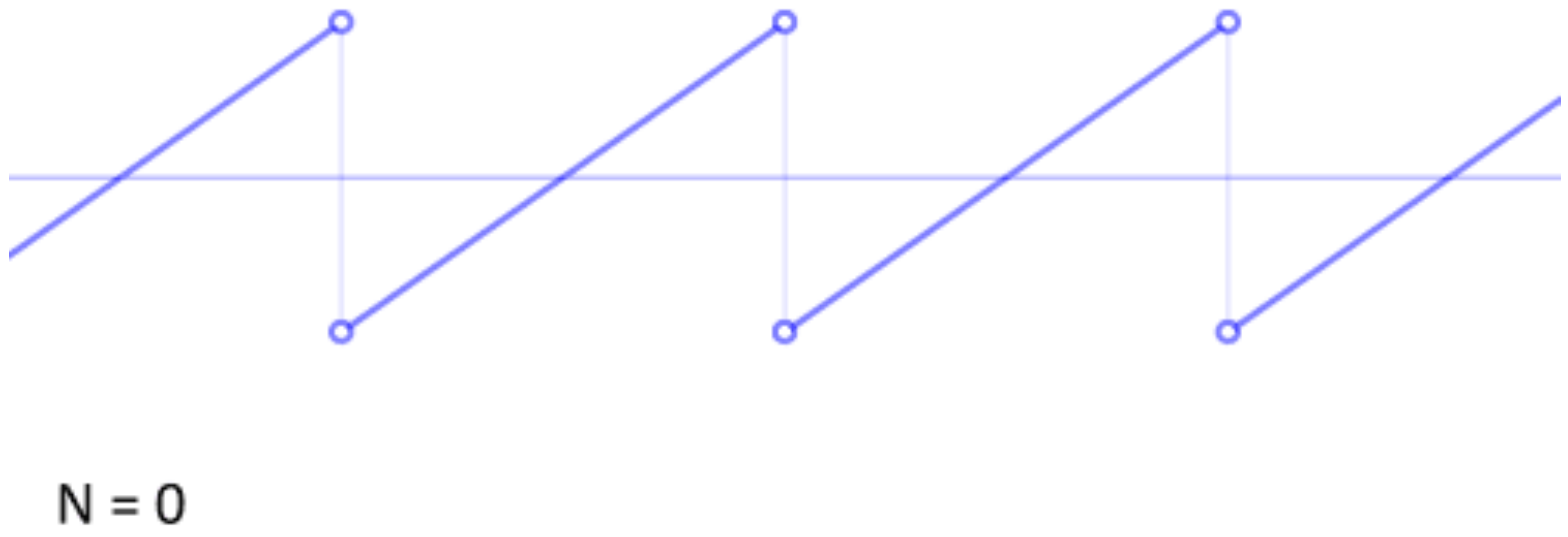
Deconstructing a complex wave (Fourier analysis)



Fourier's Theorem:

A complex periodic sound can be decomposed into a set of simple waves

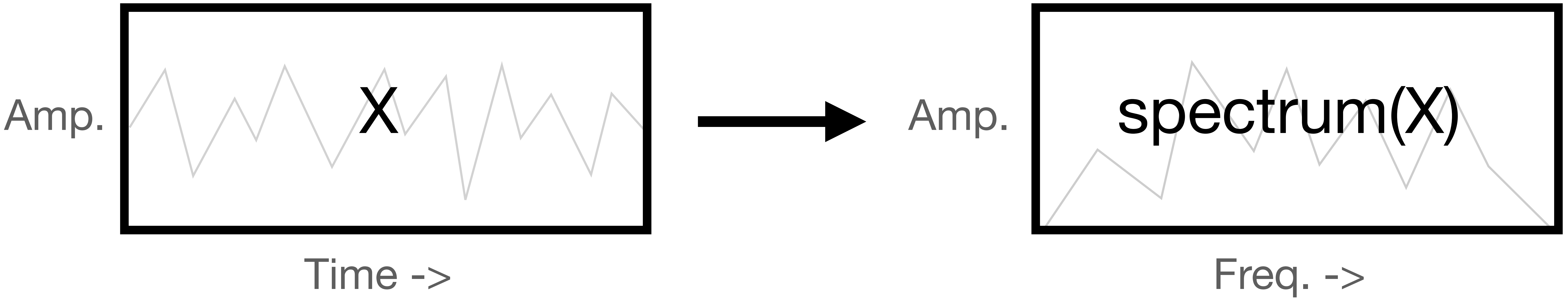
Summing simple waves (additive synthesis)



Approximating a sawtooth wave using multiple simple (sine) waves

The Fourier Transform: Take I

Goal: *Given some signal X, we want to decompose X into its constituent frequencies, that is, we want the spectrum of X.*



The result (the Fourier Transform of X) will give us two types of information:

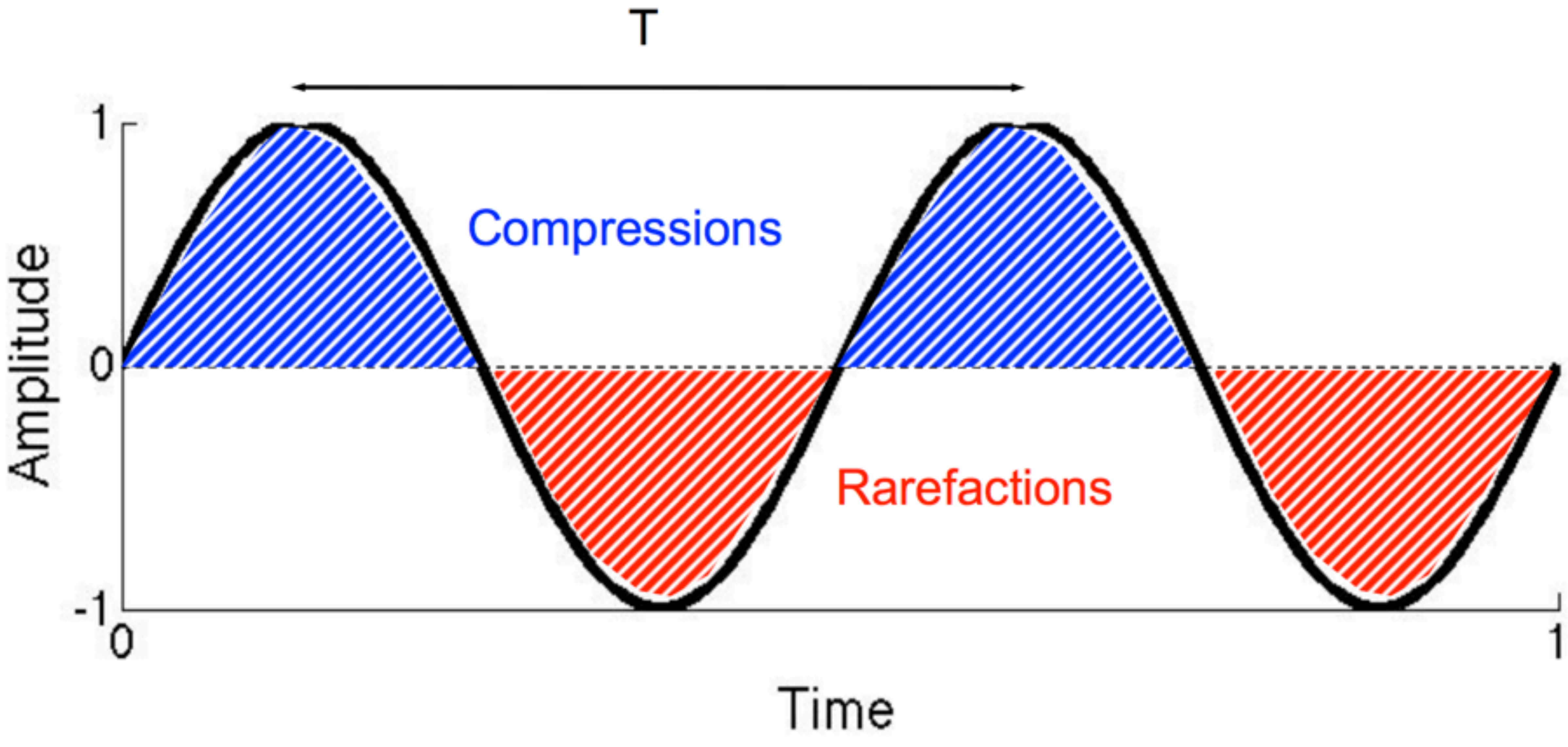
1. How much a particular frequency sinusoid is present (its magnitude) and, additionally,
2. Where in the cycle of the sinusoid it begins (its phase offset)

Remember that we get these two different (but intimately related) types of information!

The Fourier Transform

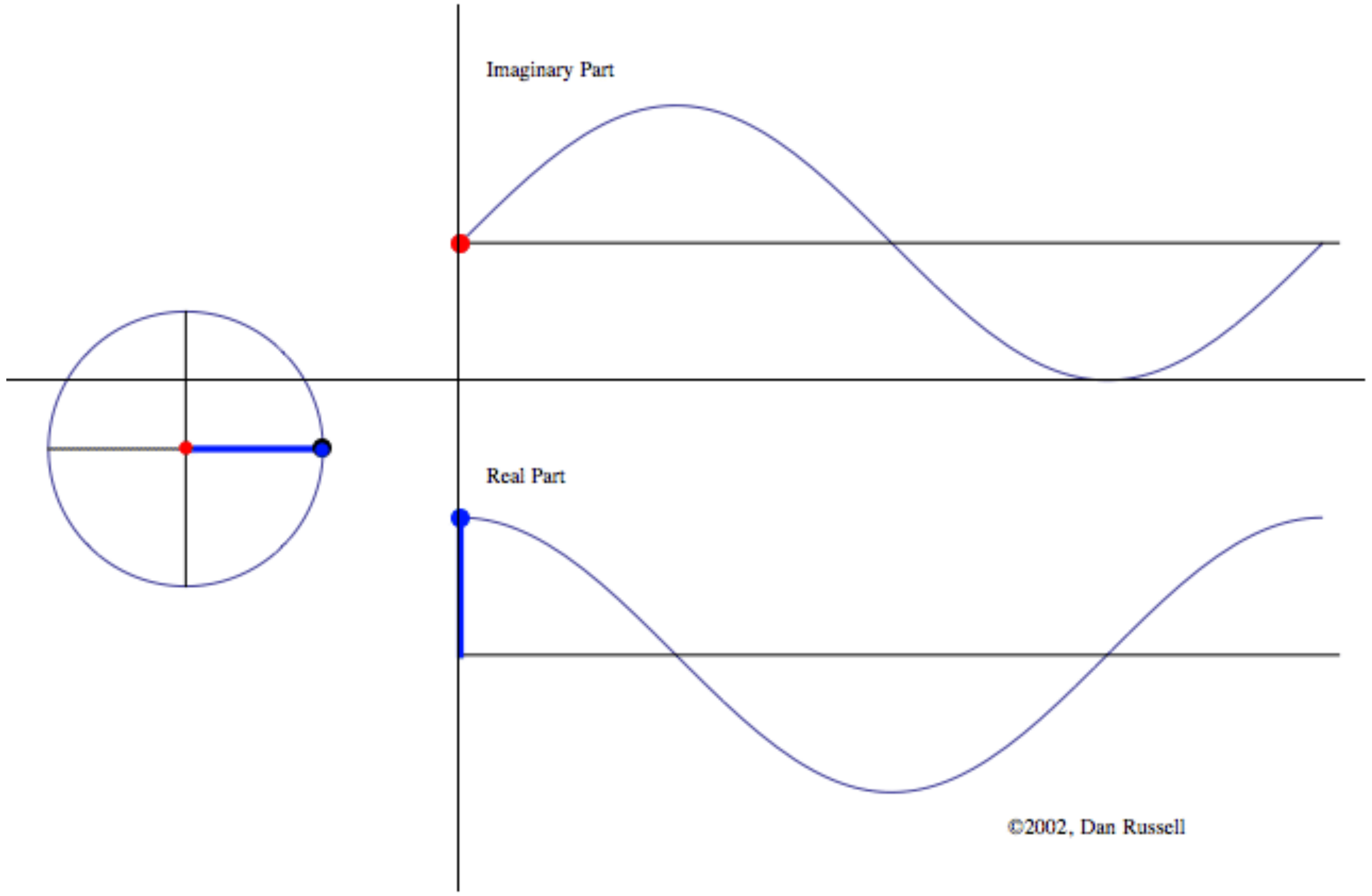
Sinusoids, which represent a single frequency in the frequency domain, are represented in the time domain in one dimension with the following equation:

$$x(t) = A \cdot \sin(2\pi ft + \theta)$$



The Fourier Transform

If, instead we have a sine in the x-Axis and a cosine in the y-Axis (each being fed the same angle), we get a 2-dimensional tracing of the circumference of a circle.



It is here where the polar (or exponential form) of the complex number,

$\boldsymbol{z = re^{j\theta}}$

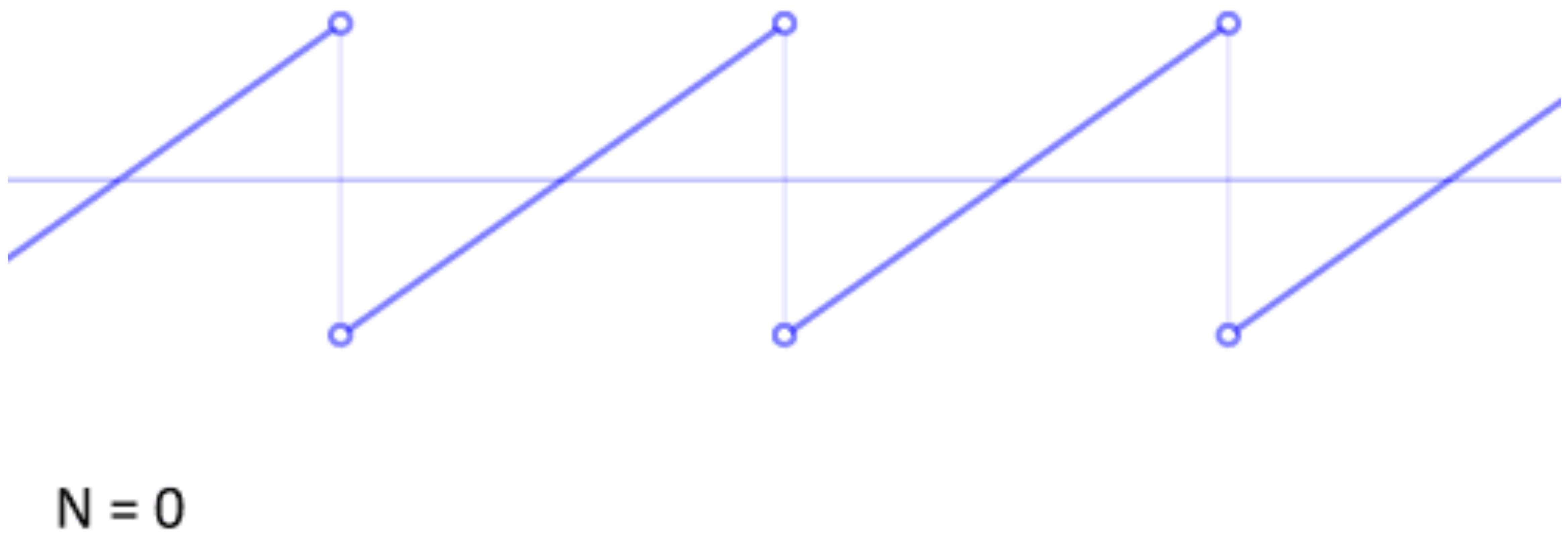
comes in handy, as it allows us to easily specify r (amplitude) and θ (phase, the rate of change of which will dictate frequency), and get two domains (real and imaginary) for the price of one!

This is called a ***phasor*** or ***complex sinusoid.***

The Fourier Transform

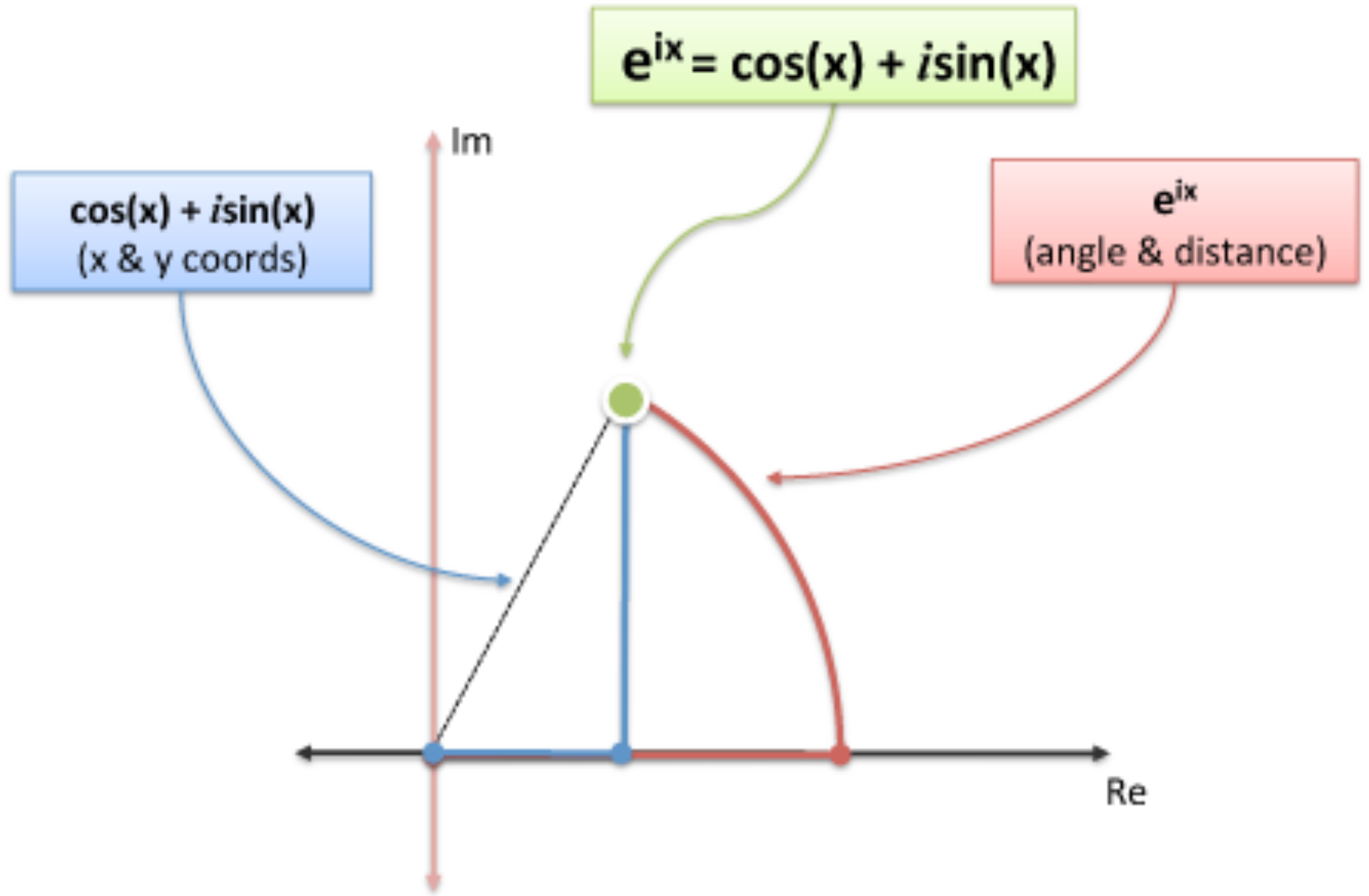
***Fourier Series*:** a group (really a function) of *harmonically related* sinusoids of different weights.

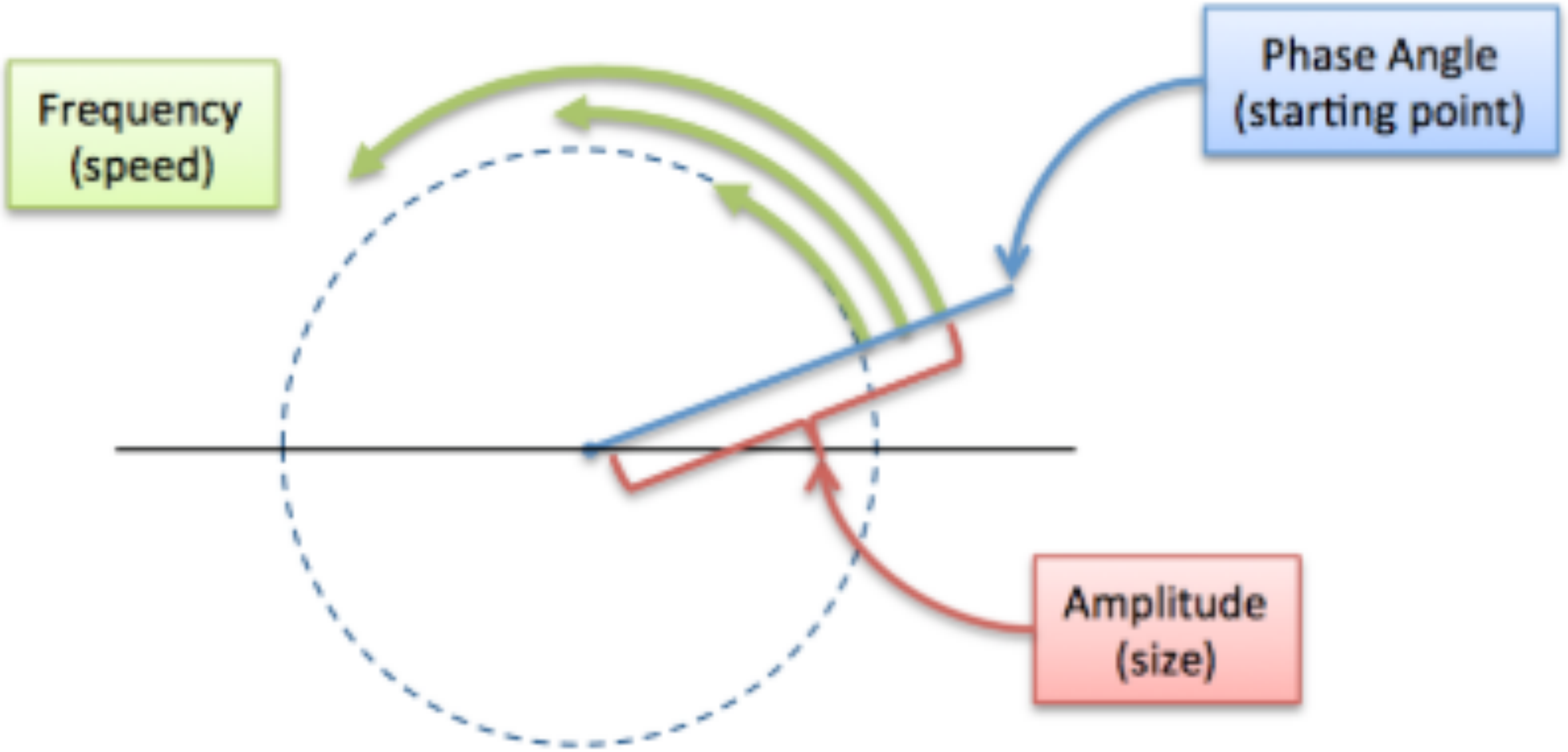
Using such a series, it is proven that we can ***synthesize*** a single period of (nearly any) periodic waveform. These series might (and often are) infinite. If we move to the discrete-time domain (e.g. digital domain), and get away from infinitesimally small (or infinite) time-scales, things get a little easier (perhaps)…



Approximating a sawtooth wave using the first fifty partials of the Fourier Series

The Fourier Transform





We can describe each complex sinusoid in our series via its amplitude (how present (loud) it needs to be to resynthesize the input) and phase (where it starts tracing the sinusoid at the beginning of our signal).

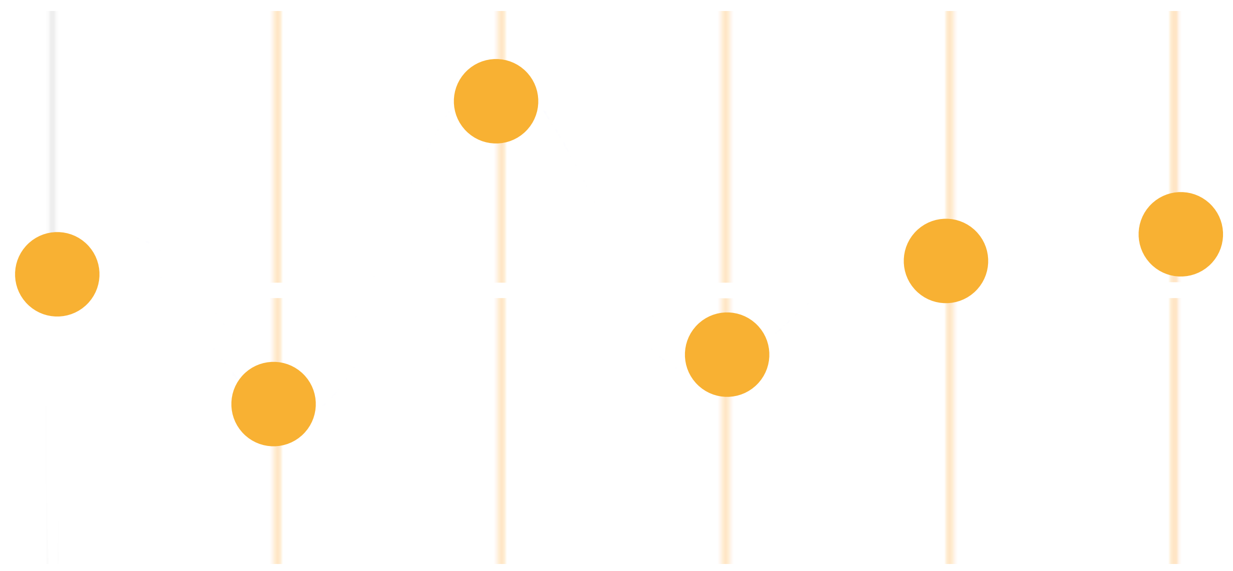
Frequency is fixed at the harmonics, based on a chosen fundamental (e.g. 10Hz) and the partials above it (20Hz, 30Hz, 40Hz, …).

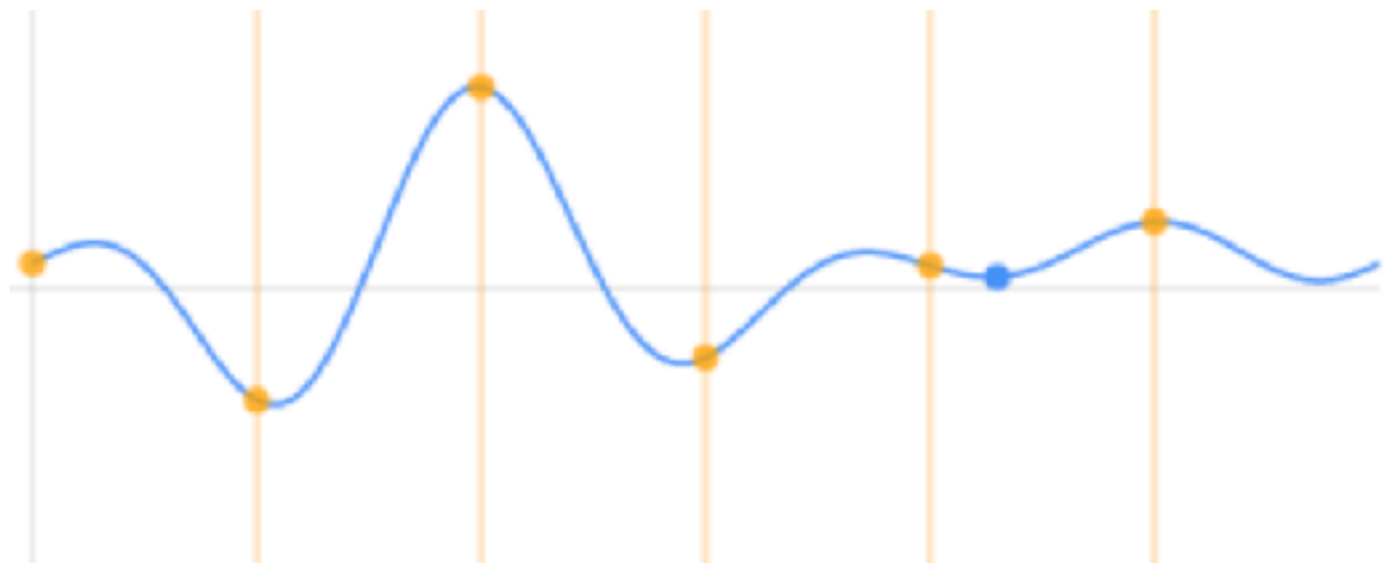
Additionally 0Hz (or DC offset, for electronic musicians) is included.

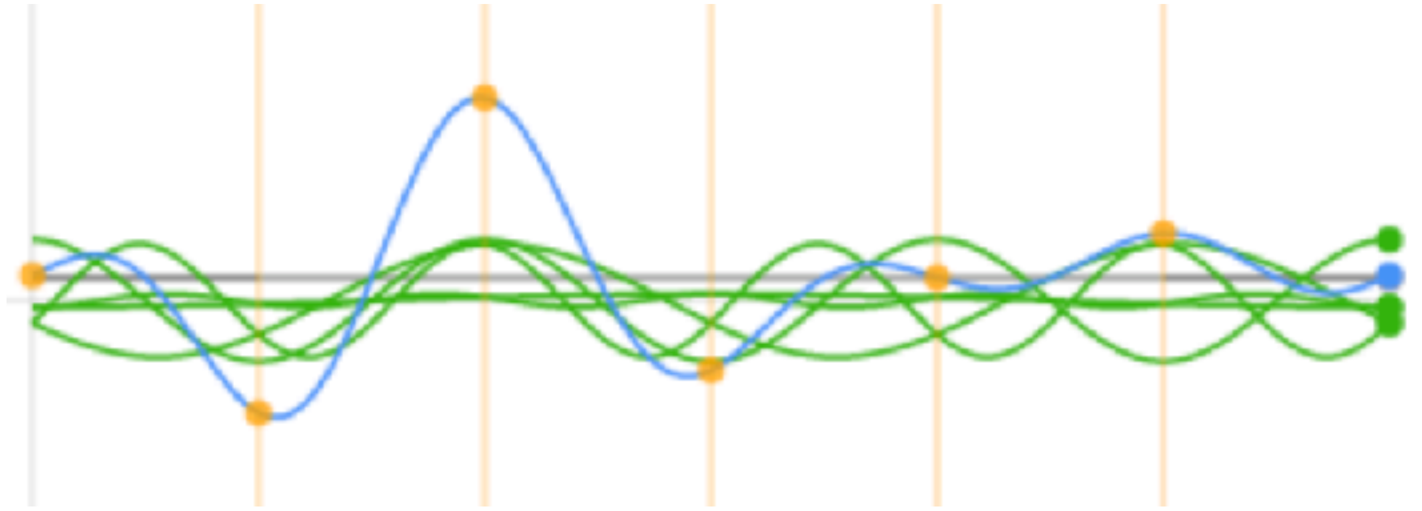
For our purposes, we'll start with 1Hz, 2Hz, etc.

The Fourier Transform

In discrete time, we need a sum of a series of sine waves to line up with each sample value in our signal, that is:







A time-domain digital signal → A curve that hits each of the digital signal's sample values → By the FT: the individual sinusoids (with diff. phases and amps.) within the series that constitute the curve

A discrete time-domain sampling of that curve ← The sum of the component sinusoids as a curve ← A frequency-domain spectrum, represented as a particular set of sinusoid harmonics

Red = DFT

Blue = Inverse DFT

The real part of the sinusoids series (amplitudes) is the frequency domain representation of the signal (spectrum)!

Interactive Example
(courtesy of Kalid Azad)

https://betterexplained.com/articles/an-interactive-guide-to-the-fourier-transform/