

Chapter 9: Waveforms and Fourier's Theorem

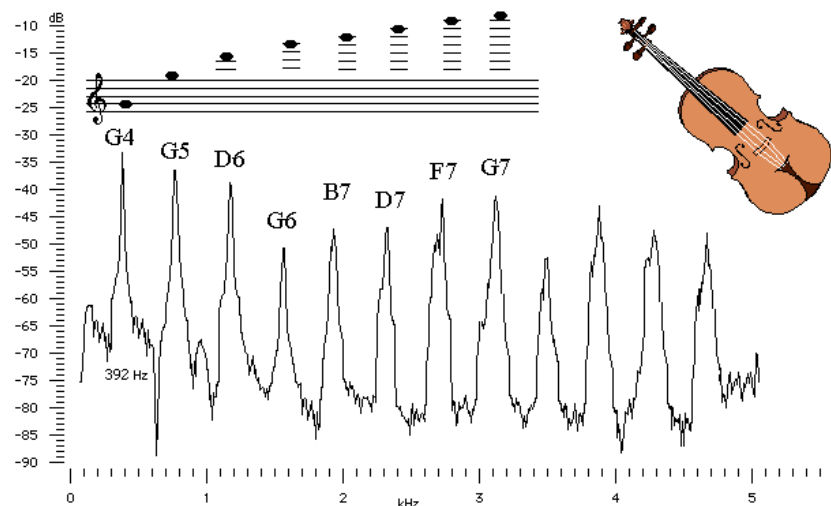
In this chapter, I will discuss how different *waveforms* shape *timbre* (roughly pronounced “tam-ber”), which is, roughly speaking, the “tonal color” that makes one signal sound different from another or one instrument sound different from another. I could play you a sine tone versus a square wave or sawtooth wave, and you could describe them as “buzzier,” “rougher,” or even “having more high frequency parts,” but here I seek to be more specific.

The discussion will center on Fourier's Theorem and its applications. Fourier's Theorem states that any complex repeating waveform can be constructed by a sum of sine waves.

Overtones and partials

The best way to hear a pure sine tone is either to use a signal generator to form the sine wave and play it through a speaker, or to use a tuning fork. The tines of a tuning fork move back and forth in (close to) simple harmonic motion, so it produces a pure tone.

Musical instruments are interesting precisely because they *don't* produce pure sine tones. Each instrument in an orchestra (and even different instruments played by different people in the same section) has a different *timbre*, or tone color. The reason is that each instrument produces a different mix of frequencies. At the risk of jumping ahead of Fourier, note what the frequency spectrum of a violin playing a G natural looks like:

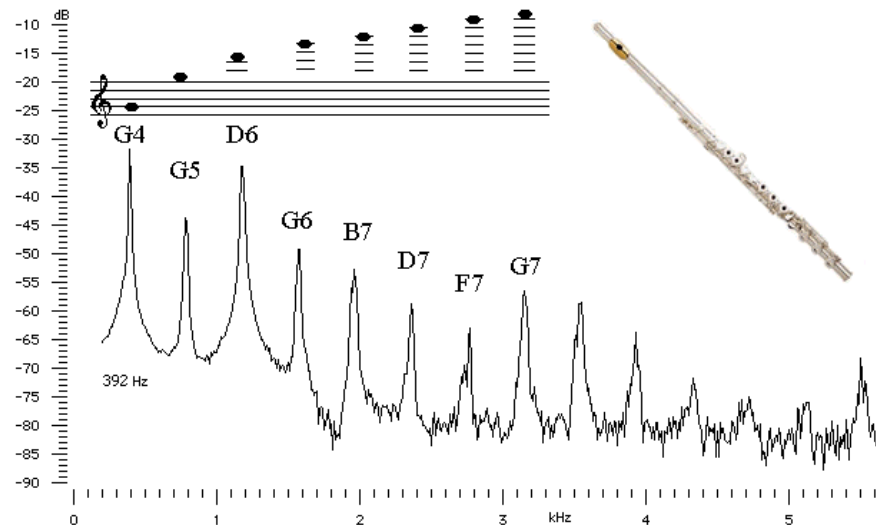


Violin frequency spectrum as it plays a 392 Hz note ([source](#))

This graph shows how much intensity the violin puts out at different frequencies. Relative level is shown on the y-axis, and frequency is shown on the x-axis. The highest spike to the left represents the 392 Hz

fundamental frequency, and we expect that to be there. What is more surprising, perhaps, is the fact that there are several other spikes. The spikes occur at regular intervals, and all those frequencies are multiples of 392 Hz, the fundamental frequency.

If we look at another instrument, you'll see the mix of spikes is slightly different (I choose a flute playing the same fundamental frequency as the violin above):



Flute frequency spectrum as it plays a 392 Hz note ([source](#))

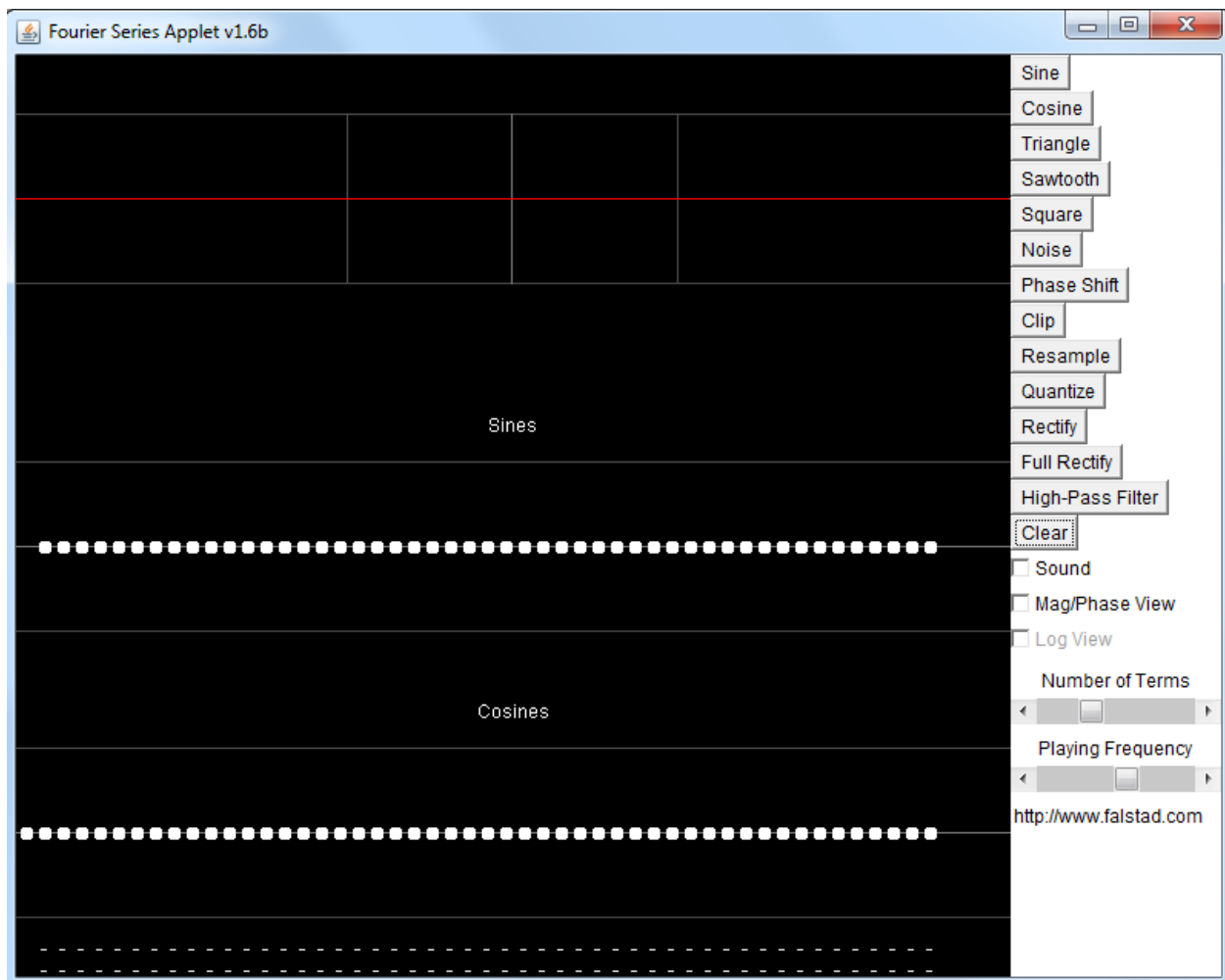
The size of the spikes is different, and that contributes to the flute's different timbre.

These spikes or frequency components are called *partials*, and any partial of a higher frequency than the fundamental is called an *overtone*. Overtones are, as mentioned before, integer multiples of the fundamental frequency. The overtones that are two, four, eight, etc. times larger than the fundamental correspond to octaves of the fundamental.

However, there are also often overtones that don't correspond to octaves, for example, the third, fifth, and sixth harmonics above. These overtones DO, however, correspond to notes we are familiar with – recall the construction of the Pythagorean or Just temperaments! Consider the second and third spikes above. If the second spike is double the fundamental frequency and the third is triple the fundamental frequency, the ratio between the two spike frequencies is $2(392) : 3(392)$, or 2:3. This gives a fifth interval, which is the interval between a G and a D. The notes that correspond to the next few spikes are marked above each one (higher numbers mean higher octave).

Constructing complex waveforms from simpler ones

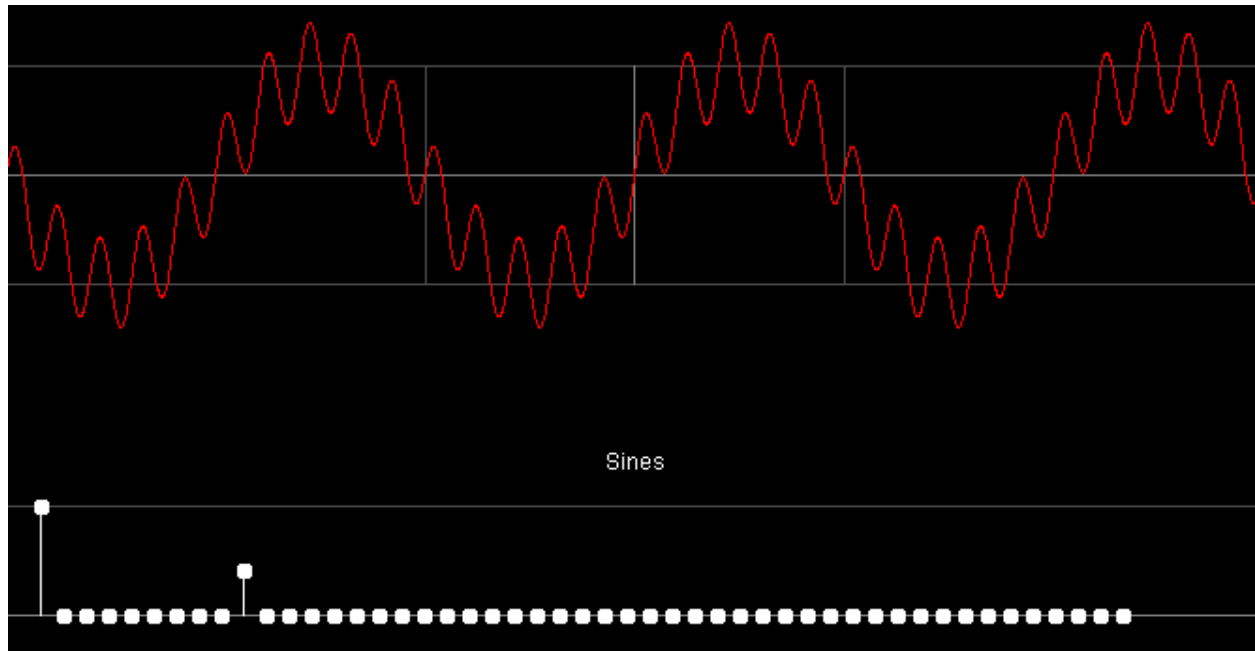
For a lot of this section, the best way to understand this is to play with an applet on the computer. One that I've really come to like is found [HERE](#). You can click on the buttons to the right to load in some common waveforms. Really, the best thing to do is play with it a while.



The Falstad [Fourier series applet](http://www.falstad.com/fourier/)

You can create triangle waves, sawtooths, square waves, and many others simply by clicking the buttons on the right. One of the more powerful features, though, is your direct control of sine and cosine waves. Note the white dots under “sines” and “cosines.” Each represents some multiple of a fundamental frequency that the program sets to about 220 Hz by default. You can slide them up and down to add them to the total waveform, shown in red at the top of the screen.

Let me show you one example. If I turn up the fundamental sine tone all the way up and add a little bit of the tenth harmonic, you can see evidence of both waves. If you mouse over either of the sliders you’ve turned up, you will see in yellow the shape of the partial in question.



The fundamental and a bit of the tenth harmonic. Hopefully you can see influence from each.

It takes some background in calculus to know exactly how to create a complex waveform from sine tones. However, there are some simple results I can share – the method to make a sawtooth, square wave, and triangle wave.

Making sawtooth, square, and triangle waves from sine tones

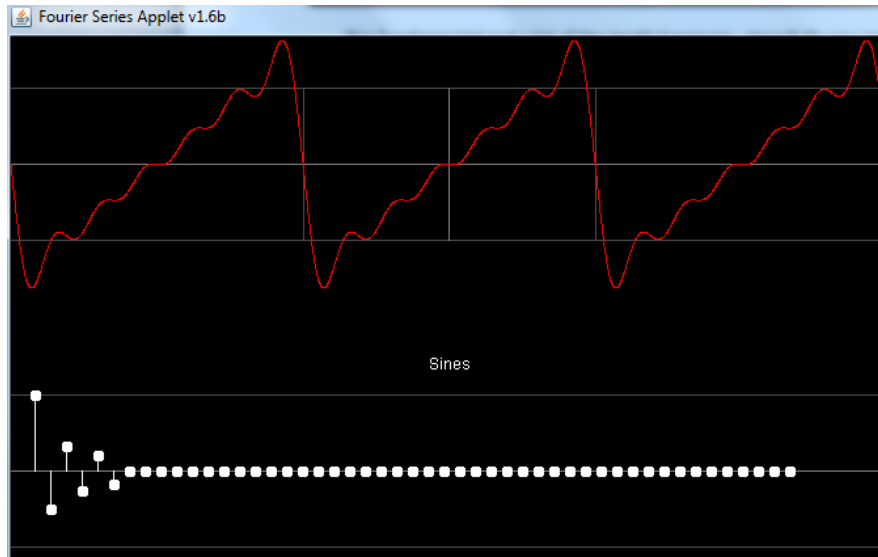
Before we can make these waveforms, there's one other feature of the applet you should know about if you haven't discovered it already. (What? You haven't loaded it yet? Do it!) The sliders can move up OR down. If you slide one of the partials downward, you get a sine wave that is out of phase with the one you get if you slide upward. Partial's do not all need to be in phase with each other – our freedom to choose the phase of each is part of what allows us to make any form we want.

Sawtooth wave: To make a sawtooth, pick an amplitude for your fundamental frequency. Then, add in a second harmonic with half the amplitude of the fundamental AND make it out of phase. Next, add in the third harmonic with the same phase as the fundamental but with $1/3$ the amplitude. Continue this way, making sure the even numbered harmonics are flipped "negative."

If the amplitude of the fundamental frequency is 1, the amplitudes of each harmonic are, respectively:

$$1, -1/2, 1/3, -1/4, 1/5, -1/6, \text{ etc.}$$

It is difficult to use the sliders to get exact amplitudes, but when I try to get as close as possible using the Fourier applet, this is the result:



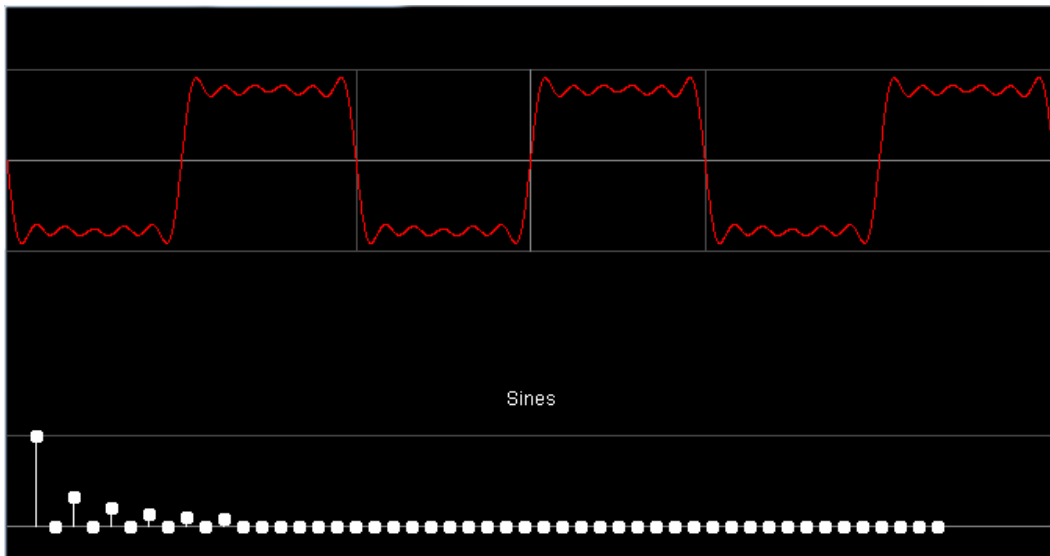
The first six harmonics making up a sawtooth wave

Fourier's Theorem says that the more terms you add to this series of harmonics, the closer to the desired sawtooth you will get. The edges of each tooth are trickier, but I will cover that later.

Square wave: To make the square wave, follow the exact same procedure as the sawtooth, but REMOVE the even numbered harmonics. If the amplitude of your fundamental wave is 1, the pattern for the harmonics goes as follows:

1, 0, 1/3, 0, 1/5, 0, etc.

I will add a few more terms to this square wave so the pattern is more apparent.

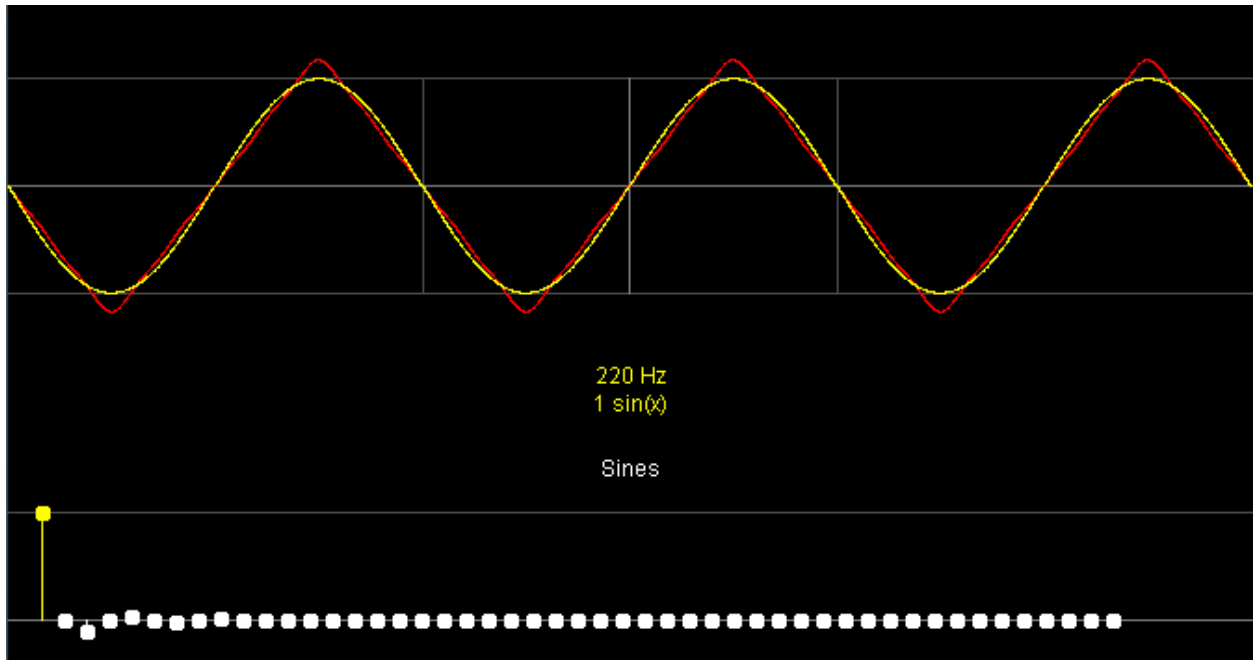


The first few harmonics that make up a square wave

Triangle wave: To get a triangle wave, again we use only the odd harmonics, but every other odd harmonic is flipped out of phase. Also, the amplitudes drop as the square of the harmonic number, so again, if the amplitude of the fundamental is 1, the pattern goes like this for the others:

$$1, 0, -1/3^2, 0, 1/5^2, 0, -1/7^2, \text{ etc.}$$

Clearly, for this wave, the fundamental dominates. You can see this very clearly if you highlight the fundamental part at the same time the whole triangle wave is visible:

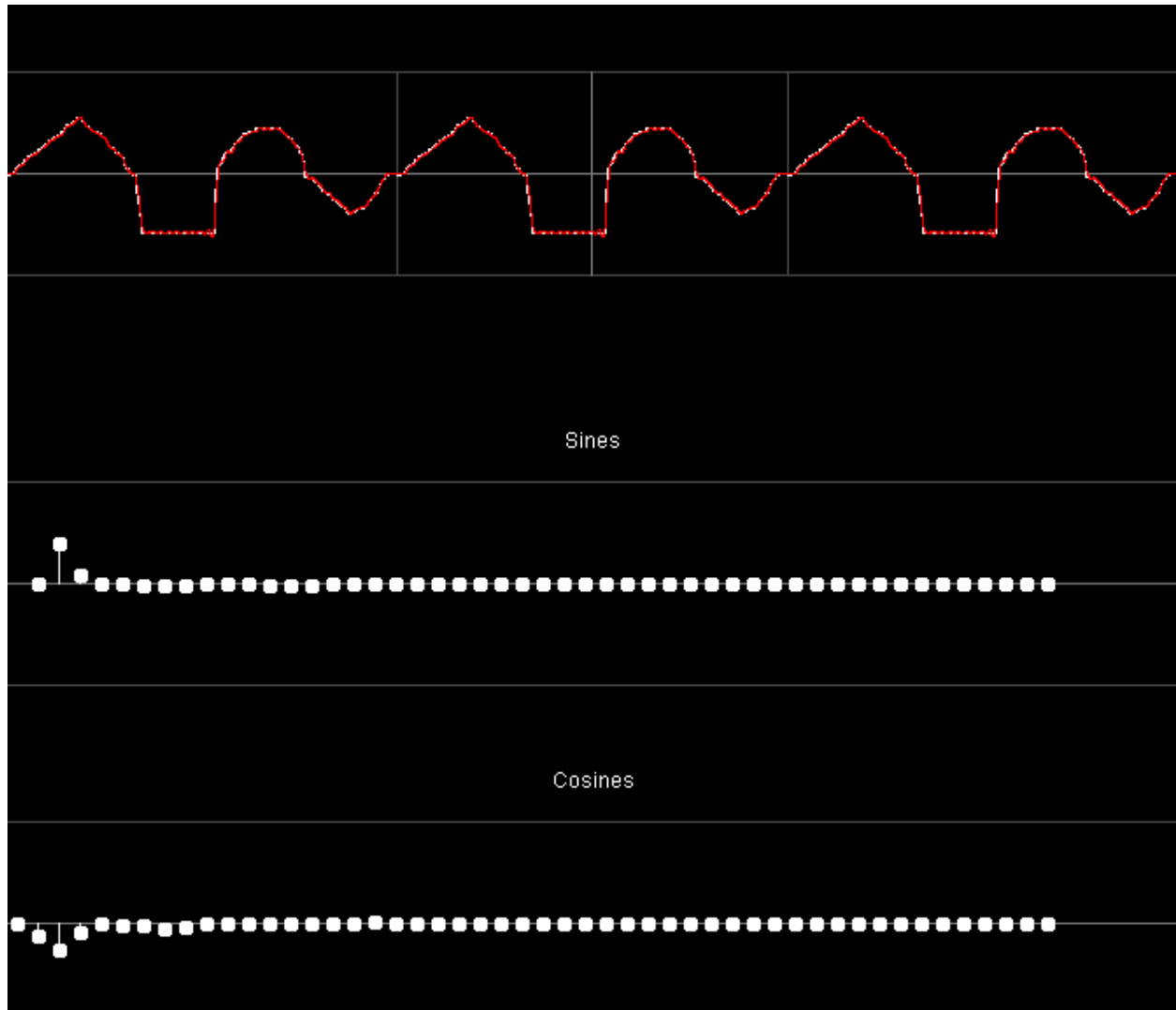


A triangle wave (red) with its fundamental partial highlighted (yellow)

Fourier analysis

Making a complex waveform from simple sine tones is called *Fourier synthesis*. The reverse process, that is, figuring out what simple sine tones make up a complex tone is called *Fourier analysis*. I skipped the mathematical steps for sawtooth, square, and triangle waves for now (but see the extra section below on how that math is done). Computers can quickly do the math and give a frequency spectrum for any given signal. The Java applet I linked to can do this automatically. Any repeating function I draw can be analyzed and broken down into a sum of sine and cosine waves.

Try it, and be whimsical if you want – I will try drawing a few shapes (some up, some down). The program will cause my drawing to repeat, then it will show the sine and cosine components.



The computer did Fourier analysis on my weird drawing and shows the sine and cosine pieces

Applications of Fourier analysis

Fourier analysis helps create the frequency spectrum graphs from the violin and flute I showed earlier. In addition to many purely mathematical applications, there are many audio-related applications of Fourier analysis. If you can break a complex sound into its component parts, you are better able to control those parts. A good place to do this is in the case where AC hum is present – if you can pick out the frequency of the hum, you can eliminate it.

Equalization is also possible using Fourier analysis. Equalization is the process of adjusting the levels of different frequency ranges in a recording. Many playback devices have equalizers that allow you to boost or lower bass, treble, or midrange frequencies.

Also, telephone buttons use Fourier's idea – each button press plays two frequencies, and knowing the exact frequency breakdown can help the electronics understand which button was pressed.

Gibbs phenomenon

One effect you may notice on the simulation is that whenever you try to simulate a wave with sudden jumps in the waveform (like the square or sawtooth), there are small spikes or horns at the jumps. This is known as the *Gibbs phenomenon*, which states that if you use a finite number of sine or cosine waves to simulate a function with sudden jumps, there will be an overshoot of about 9%.

This is to some degree unavoidable in Fourier synthesis, since using an infinite number of sine waves is technically impossible. The complication is that clipping may result.

The mathematics of Fourier series

I stated some recipes for Fourier coefficients without proof above. With a little bit of trigonometry and calculus knowledge, we can figure out a way to come up with the proper coefficients for the sine and cosine functions of a repeating waveform.

For simplicity, I will assume the function we want has a period 2π , like the sine and cosine functions we are about to use. (If you've read the appendix and note that a function like $y = (\sin 3t)$ has a period of $2\pi/3$, that's true – but it repeats completely after 2π as well.) If our goal is to write a function $F(t)$ as a series of sines and cosines, it might look like this:

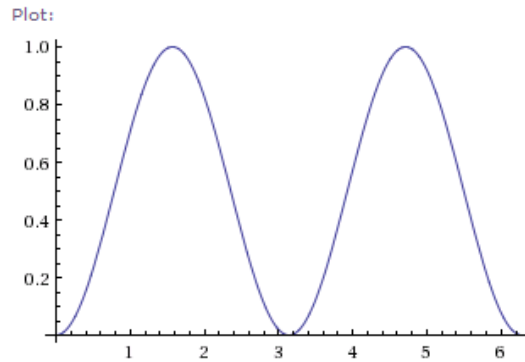
$$F(t) = A_0 + A_1 \cos(t) + A_2 \cos(2t) + A_3 \cos(3t) \dots + B_1 \sin(t) + B_2 \sin(2t) + B_3 \sin(3t) + \dots$$

The term A_0 is a vertical shift that goes with the term " $\cos 0t$," which is equal to one.

Suppose you wanted to find one of the sine wave coefficients – say, B_1 . Here's the mathematical trick we can use to get it. If you multiply every term here by $\sin(t)$, you get

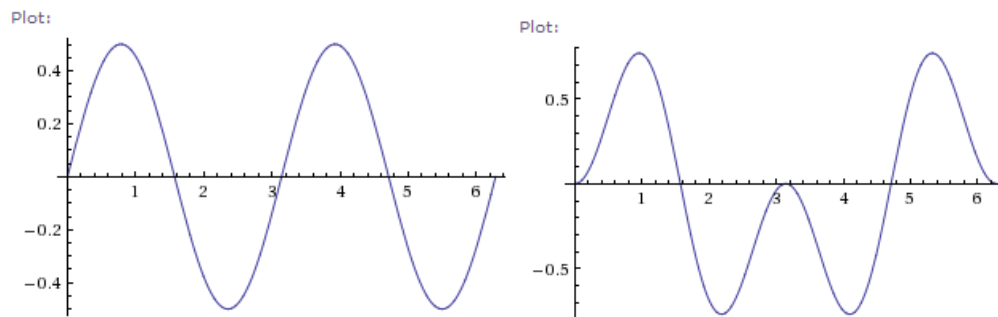
$$F(t)\sin(t) = A_0\sin(t) + A_1 \cos(t) \sin(t) + A_2 \cos(2t) \sin(t) + A_3 \cos(3t) \sin(t) \dots \\ + B_1 \sin(t) \sin(t) + B_2 \sin(2t) \sin(t) + B_3 \sin(3t) \sin(t) + \dots$$

This has seemingly created a big mess. But if we look at graphs of some of these trigonometric products, we see something interesting. [Observe](#) the $\sin(t) \sin(t)$ part next to the B_1 coefficient we want.



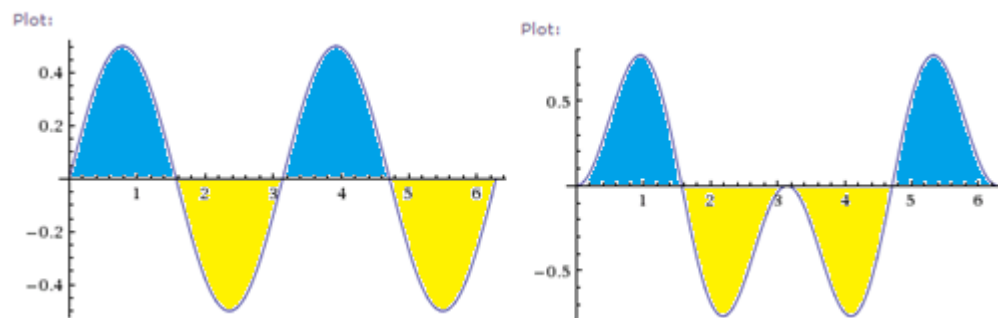
$\sin(t)$ multiplied by $\sin(t)$

And now, note these graphs of some of the others – for example, [sin\(t\) cos\(t\)](#) or [sin\(t\) sin\(2t\)](#)



Two other plots: $\sin(t) \cos(t)$ and $\sin(t) \cos(2t)$

The key difference is that the $\sin(t) \sin(t)$ function on top is positive everywhere. The two sample functions I graphed below it spend equal time below and above the t-axis. Notice also that though these two functions look different, both have equal areas above and below that axis – that is, the blue and yellow areas below are the same:



Positive and negative areas colored in

Since these positive and negative areas cancel out, we say the area *under* these graphs is zero. And what's more – this property holds for ALL those trig function products (and the A_0 term) in the messy Fourier equation above EXCEPT for $\sin(t) \sin(t)$!

This helps us simplify. If, in the messy equation above, we take the AREA under the graphs, we get:

$$\begin{aligned} \text{Area under } (F(t) \sin(t)) \text{ from } 0 \text{ to } 2\pi \\ = \text{Area under } (A_0 \sin(t) + A_1 \cos(t) \sin(t) + A_2 \cos(2t) \sin(t) + A_3 \cos(3t) \sin(t) \dots \\ + B_1 \sin(t) \sin(t) + B_2 \sin(2t) \sin(t) + B_3 \sin(3t) \sin(t) + \dots) \end{aligned}$$

All but one area on the right is zero, giving:

$$\text{Area under } (F(t) \sin(t)) \text{ from } 0 \text{ to } 2\pi = \text{Area under } (B_1 \sin(t) \sin(t))$$

But B_1 is a constant, so:

$$\text{Area under } (F(t) \sin(t)) \text{ from } 0 \text{ to } 2\pi = B_1 \times \text{Area under } (\sin(t) \sin(t))$$

The entire plot area of the $\sin(t) \sin(t)$ graph is 2π , and the area under the curve is exactly half, or π . Therefore, we have an expression for B_1 :

$$B_1 = \frac{1}{\pi} \times [\text{Area under } (F(t) \sin(t)) \text{ from } 0 \text{ to } 2\pi]$$

Or, in the integral language of calculus:

$$B_1 = \frac{1}{\pi} \int_0^{2\pi} [F(t) \sin(t)] dt$$

In general, the coefficients A_n and B_n can be found with:

$$A_n = \frac{1}{\pi} \int_0^{2\pi} [F(t) \cos(nt)] dt$$

$$B_n = \frac{1}{\pi} \int_0^{2\pi} [F(t) \sin(nt)] dt$$

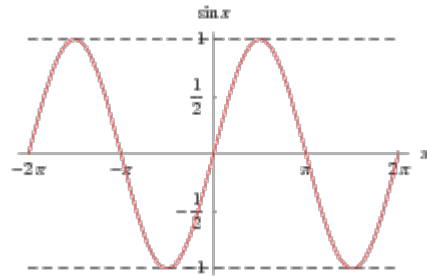
Carrying out these calculations for forms like the sawtooth, square, and triangle prove the “recipes” I gave earlier. Recall also that I assumed a period 2π for the periodic function $F(t)$. A more general formula for Fourier coefficients of a function of a different period can be found [here](#).

Odd and even properties

We see in both the formulas and linked applets above that Fourier coefficients tell us how much to dial in each harmonic of the sines and cosines. Some waveforms allow us to eliminate all the sines or all the cosines depending on their *even-* or *oddness*.

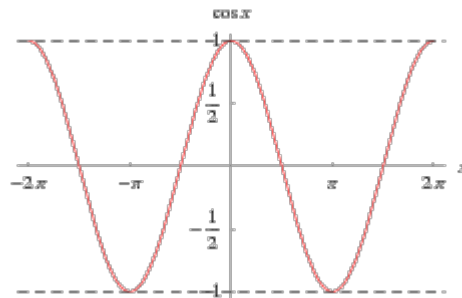
Odd functions are called symmetric about the origin – for every point (x, y) in a graph, the point $(-x, -y)$ is also in the graph. The name *odd* comes from the fact that the graphs of the functions $y = x$, $y = x^3$, $y = x^5$, etc., (note the pattern of odd exponents) as well as all sums of these functions have this property. The

sine function is called an *odd* function because $\sin(-x) = -\sin(x)$. The Appendix shows how we can approximate sine waves by some combination of odd powers of x .



The sine function is odd (picture from Wolfram MathWorld)

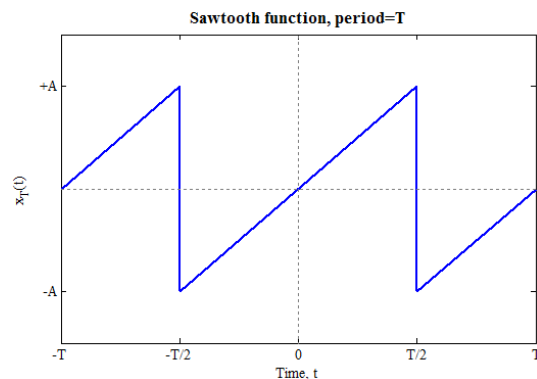
The cosine function has the property $\cos(-x) = \cos(x)$. Even functions are symmetric about the vertical axis. The functions $y = x^2$, $y = x^4$, etc. (as well as sums of these) have this property and give even functions their name. The Appendix shows how we can approximate cosine waves by some combination of even powers of x . “Constant” functions like $y = 5$ are graphed as horizontal lines and also have the “even” property.



The cosine function is even (picture from Wolfram MathWorld)

A function is either even, odd, or neither. If a waveform is even, it can only be made up of even cosine functions. If a waveform is odd, it can only be made up of odd sine functions. If a waveform is neither even nor odd and you want to build it up with sines and cosines, you have to use some of both.

For example, look at this sawtooth:



A sawtooth wave (from [here](#))

This function is odd – for every point, look for the mirror image point across the origin. That means if I want to find the Fourier series breakdown of this, I will only use sine functions and the coefficients of the cosines will all be zero.

Other complex waveforms and musical effects

In an orchestra or choir, there is an advantage of having large sections of instruments or voices that goes beyond pure loudness. If multiple instruments play the same note at the same time, the different instruments may be subtly different in pitch or phase. When this happens, a richer sound often results, and this phenomenon is called the *chorus effect*.

Even acoustic instruments take advantage of this effect. Two pairs of strings on a 12-string guitar are tuned to the same note, so you'll get a small chorus effect with those. The remaining four pairs are tuned octaves apart, so the difference in those pairs is more striking. Mandolins are 8-stringed instruments with four identical string pairs. Also, piano keys often strike multiple strings; for most notes, there are three strings struck. These are tuned very close to each other in pitch, but are not always a perfect match.

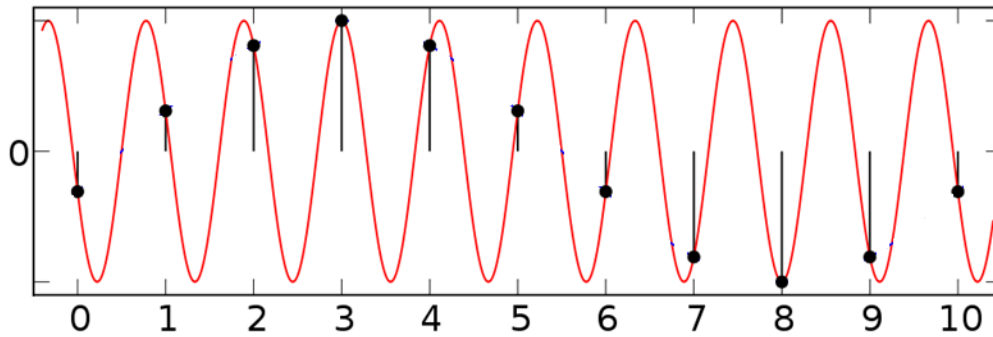
Electronically, you can get a chorus effect by copying a signal and applying a slight delay or pitch shift to some of the copies before recombining everything again. This gives the illusion of multiple instruments playing at once. The effect called *flanging* is a variant of this – the delay between a signal and its copy is precisely varied to give a complex-sounding sum.

A recording danger: aliasing

I'd like to digress slightly from the topic of Fourier's theorem to discuss a potential pitfall in recording: *aliasing*. An alias is a low frequency artifact that can occur when digitally sampling a wave. Many recording applications contain filters that automatically deal with this problem, but you should be aware of it.

When digitally recording a signal, an analog input is converted to a string of numbers. The method this is done is *sampling*: it is impossible to record the state of an analog signal at EVERY possible moment, so a snapshot is taken at regular intervals. The rate at which a signal is sampled is called, not surprisingly, the *sampling rate*.

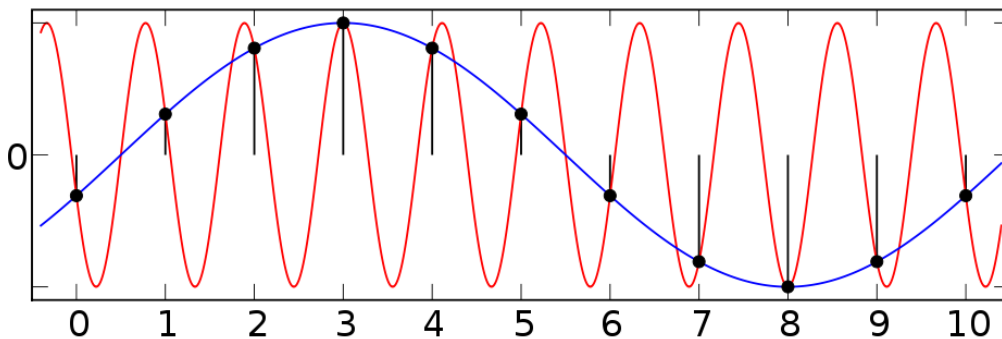
Most audio material recorded is filtered so anything above about 20 kHz is removed. The standard sampling rate for compact discs is 44.1 kHz. Let's look at why this is. Consider the following wave and determine its frequency if the x axis represents time in seconds.



A wave to be sampled. Find the frequency!

Many people say it's about 1 Hz, but be careful! Count the number of complete cycles between the time just after zero seconds and the time just past ten seconds. There are ten dips, but nine complete cycles. Nine cycles in ten seconds means the frequency is $9/10$ or 0.9 Hz.

Now, let's see what happens when we *sample* the 0.9 Hz frequency every second. I've already done this in the picture above. The black dots represent the samples, and those occur every second – they're lined up exactly with the tick marks! If you were to play the signal the black dots represent (picture connecting the dots), you get this:



Thanks to the relatively low sample rate, the [blue signal](#) is what's captured.

If I just showed you the blue wave, you'd have no way to tell if it was a good reproduction of a 0.1 Hz wave (check this!) or a poorly sampled version of a 1 Hz wave. The 0.1 Hz wave shown here is a low-frequency artifact called an *alias*. Of course, both 1 Hz and 0.1 Hz are below our hearing, so we couldn't hear either of these signals, but the sampling principle holds for higher frequencies too.

To avoid this problem, we should have sampled more often so the 1 Hz wave is more accurately represented. The *Nyquist criterion* says that a signal can be accurately digitized with a minimum of aliasing if the sampling rate is double the highest frequency component. Here's the connection to previous material in the chapter: we see that many instruments and signals have high frequency partials. These must be filtered out if to avoid aliasing. As a rule, the frequency of possible aliases is given by:

$$f_{alias} = |f_{original} - N \times f_{sample\ rate}|$$

In my previous example, $N = 1$ (and that's usually the N for the most problematic alias). So practically, if you filter everything above 20 kHz and sample at 44.1 kHz, audible aliases will be minimized.

One can see visible aliasing as well. In an earlier chapter, we showed what happens with standing waves on a string. If you shine a strobe light on such a setup where the strobe frequency is equal or close to the string frequency, you can see the string appear to stop or move in slow motion, as shown [here](#).

Summary

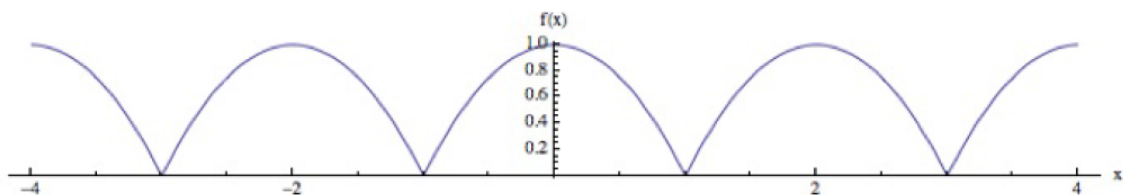
- Musical instruments play more than just the intended *fundamental* frequency. They also produce multiples of that frequency called *overtones*.
- Overtones are sometimes octaves of the fundamental frequency. When they're not, they represent some Pythagorean interval. The fifth often appears prominently.
- Fourier's Theorem says that any repeating wave can be created with a series of sine and cosine waves of the proper amplitude.
- Sawtooth waves use all the harmonics of a fundamental frequency, properly adjusted for phase and amplitude.
- Square and triangle waves use a different mix of harmonics; they don't use the even harmonics at all.
- Odd waveforms are made up of only sine wave Fourier pieces (because sine functions are odd). Even waveforms are made up of cosines and constant functions (the A_0 terms).
- High frequency overtones may lead to aliases, which are low frequency artifacts that appear in signal processing when the sampling rate is not high enough.

Homework problems

1. Go to either the PhET website or Falstad applet listed in the Powerpoint slides online. Make a complex wave where all harmonics have the same amplitude. Don't make the harmonic amplitudes too big, or the whole shape won't fit on screen. Sketch and describe the shape you get. Or remember what you saw when I did it (if I did.)

For 2-4, give the amplitude you would use (amplitudes are always positive!) for the first seven harmonics of these waves. If a certain harmonic is **not** used, mark the amplitude as zero. You shouldn't need the applets for these – just indicate what amplitudes you **would** use for such a program.

2. A sawtooth wave where the fundamental frequency has amplitude 4.
3. A triangle wave where the fundamental frequency has amplitude 1.
4. A square wave where the fundamental frequency has amplitude 6.
5. (a) A tone has a harmonic sounding at 39000 Hz. If the sampling rate is 42000 Hz, where would there be an alias frequency? (b) What is the frequency (called the Nyquist frequency) we should use as a filter cutoff to prevent lower frequency aliases?
6. Here's a decibel question for quiz review. If one clarinet playing quietly has intensity of 61 dB, how many dB would a group of 11 of them playing at that intensity produce?
7. Consider the function $f(x)$ shown here:

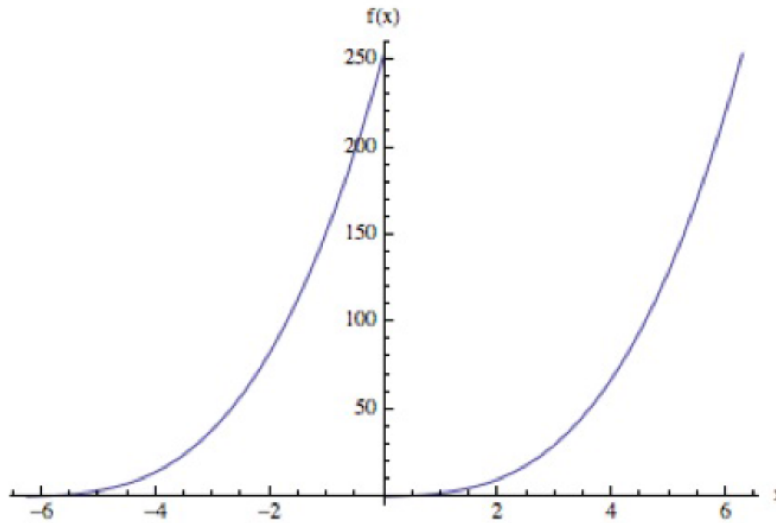


- (a) What is the period of this function?
- (b) One way to write the Fourier series for a function $f(x)$ and period L is:

$$f(x) = \frac{a_0}{2} + \sum_{n=1}^{\infty} \left[a_n \cos\left(\frac{n\pi x}{L}\right) + b_n \sin\left(\frac{n\pi x}{L}\right) \right]$$

Which of these coefficients (a_0 , or the a_n cosine coefficients or b_n sine coefficients) can we say for sure will be zero or non-zero? Why?

8. Consider the function $f(x) = x + x^3$ but repeated to have a period equal to 2π .



Which of the Fourier coefficients (a_0 , or the a_n cosine coefficients or b_n sine coefficients) can we say for sure will be zero or non-zero? Why?

9. Consider the function $f(x) = 5$. Is this function even or odd? Can we write this as a Fourier series at all?

10. Consider the function $f(x) = |x|$ for $-1 \leq x \leq 1$. This would be something like a triangle wave if you repeat this periodically. The Fourier series for this function is given by

$$f(x) = \frac{1}{2} - \frac{4}{\pi^2} \left(\cos(\pi x) + \frac{\cos(3\pi x)}{9} + \frac{\cos(5\pi x)}{25} + \dots \right)$$

Looks like a handful, but it has properties we expect it to have: note that the absolute value function is even, so that's why there are only cosines in it (aside from the $\frac{1}{2}$ term in front that shifts it up), and it's a triangle wave, so odd harmonics appear (and those denominators get larger in the way we expect for these waves. **Here's the question:** How would you write out the Fourier series for a function $g(x)$ if $g(x) = 4f(x) - 2$? This would give a triangle wave bigger in amplitude and shifted slightly downward.