The Mathematics Behind Music

Anton Vlasev

April 2026

Every time you listen to a song, your brain solves a problem that stumped mathematicians for centuries. The sound arriving at your eardrum is a single, wildly complicated wave — a rapid oscillation in air pressure that encodes, simultaneously, the kick drum, the bass line, the vocals, the reverb, and every other element of the track collapsed into one signal. Your auditory system pulls all of those apart in real time, identifying individual frequencies and instruments as though they were never mixed together. The mathematics that describes exactly how this is possible, and how engineers have learned to replicate and exploit it, is the Fourier transform, and it is one of the most beautiful and far-reaching ideas in all of mathematics. It did not start with music. It started with heat. And the fact that it ended up inside every mp3 player, recording studio, and streaming service on the planet is the kind of unreasonable effectiveness that makes mathematics genuinely thrilling.

Joseph Fourier was trying to solve the heat equation in 1807 when he made the claim that broke mathematics. He asserted that any function — literally any function, however jagged or discontinuous — could be written as an infinite sum of sines and cosines. The mathematical community at the time, including Lagrange, rejected the idea almost immediately, because it seemed obviously wrong. A sine wave is smooth and periodic and well-behaved. A function with a jump discontinuity is none of those things. How could infinitely many smooth waves add up to something with a corner in it? The answer, it turned out, was: very carefully, and with some important caveats that took another century to fully sort out. But the core of Fourier's insight was correct, and it is the foundation of everything that follows.

The basic idea is this. A pure musical tone is a sine wave. If you play a tuning fork at concert A, the air pressure at your ear oscillates as

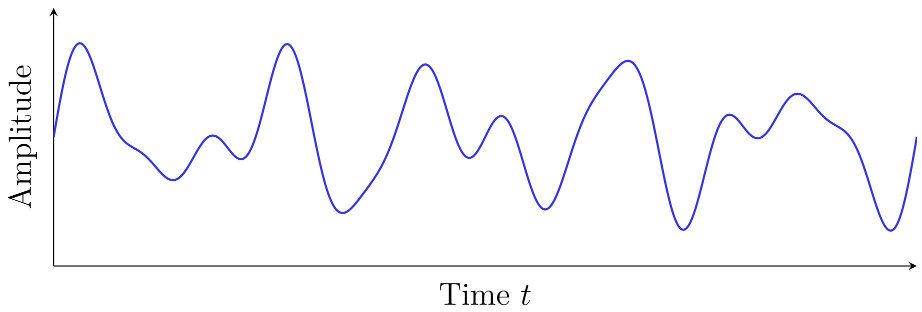
$$f(t) = A\sin(2\pi \cdot 440 \cdot t),$$

where 440 is the frequency in Hertz and A is the amplitude. That is the simplest possible sound. Every other sound — a violin, a human voice, an electric guitar — is more complicated, but Fourier's theorem says it is still just a sum of sine waves at different frequencies and amplitudes. The Fourier transform is the tool that takes a complicated signal $f(t)$ and tells you exactly which frequencies are present and how strongly. It is defined as

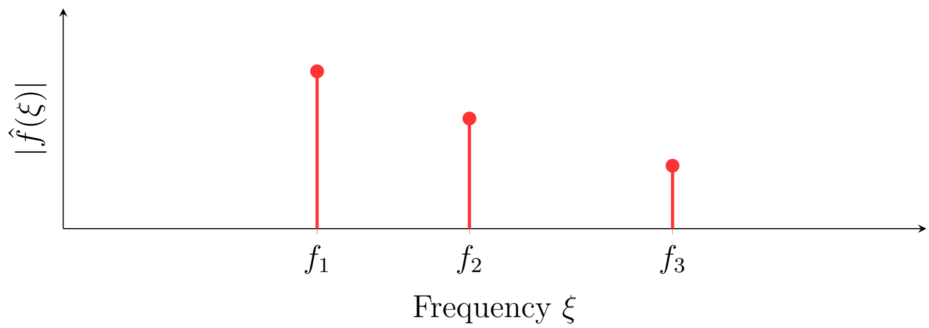
$$\hat{f}(\xi) = \int_{-\infty}^{\infty} f(t)\, e^{-2\pi i \xi t}\, dt,$$

where ξ is the frequency variable and $\hat{f}(\xi)$ is the complex amplitude of the component at frequency ξ. The magnitude $|\hat{f}(\xi)|$ tells you how much of frequency ξ is in the signal. The output of the Fourier transform is called the frequency spectrum, and it is, in a very precise sense, what your ears are computing when they hear music.

A chord: three sine waves summed into one signal



Its Fourier transform: the three frequencies revealed



The reason this works, the reason you can decompose a signal into frequencies and then reconstruct it perfectly, comes down to a remarkable orthogonality property of the complex exponentials $e^{2\pi i\xi t}$. Two functions are orthogonal if their inner product is zero, where the inner product on function space is defined by integration:

$$\langle f, g\rangle = \int_{-\infty}^{\infty} f(t)\overline{g(t)}\, dt.$$

The complex exponentials at different frequencies are orthogonal in exactly this sense:

$$\int_{-\infty}^{\infty} e^{2\pi i\xi t} e^{-2\pi i\xi' t}\, dt = \delta(\xi - \xi'),$$

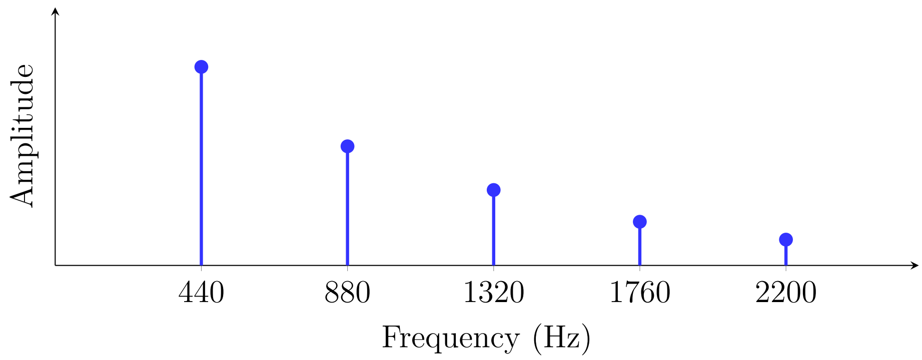
where δ is the Dirac delta function. This orthogonality is the reason the Fourier transform can isolate individual frequencies without contamination from others. Each frequency component lives in its own orthogonal dimension of function space, and the transform is simply the projection of the signal onto each of those dimensions. The analogy with vectors in ordinary space is exact: just as you can decompose a vector into its x, y, and z components by projecting onto three orthogonal axes, you decompose a signal into its frequency components by projecting onto infinitely many orthogonal exponentials. The Fourier transform is a change of basis in an infinite-dimensional vector space.

This is where the connection to music becomes concrete and, frankly, astonishing. When a violin plays the note A, the string vibrates not just at 440 Hz but at 880 Hz, 1320 Hz, 1760 Hz, and all subsequent integer multiples, each with its own amplitude. These are called harmonics or overtones, and they are predicted by the wave equation. The general solution to the wave equation on a string of length L fixed at both ends is

$$y(x,t) = \sum_{n=1}^{\infty} \sin\left(\frac{n\pi x}{L}\right)\left(A_n \cos\left(\frac{n\pi ct}{L}\right) + B_n \sin\left(\frac{n\pi ct}{L}\right)\right),$$

where c is the wave speed and the coefficients A_n, B_n are determined by the initial conditions. Each term in this sum is a harmonic. The first term, $n = 1$, is the fundamental frequency, the pitch you name when you say the note is A. The terms $n = 2, 3, 4, \ldots$ are the overtones, and their relative amplitudes are what give the violin its distinctive timbre. A flute and a violin playing the same note have the same fundamental frequency but completely different overtone structures, and the Fourier transform of their sounds will look completely different even though the pitch is identical. Timbre is Fourier structure. When you hear the difference between a violin and a flute, you are perceiving a difference in the shape of a frequency spectrum.

Harmonic series of a string at 440 Hz: overtones with decaying amplitude



Now comes the part that connects all of this to modern music technology. In practice, audio signals are not continuous functions but sequences of discrete samples: a digital recording is a list of numbers representing the amplitude of the sound at equally spaced instants in time. The CD standard samples at 44100 times per second, meaning every second of audio is stored as 44100 numbers. To analyse the frequency content of a discrete signal, you need the discrete Fourier transform, or DFT. Given a sequence of N samples $x_0, x_1, \ldots, x_{N-1}$, the DFT produces N complex numbers

$$X_k = \sum_{n=0}^{N-1} x_n \, e^{-2\pi i k n/N}, \qquad k = 0, 1, \ldots, N-1,$$

where X_k is the amplitude and phase of the frequency component k/N cycles per sample. This is the discrete analogue of the continuous Fourier transform, and it inherits all the same properties: orthogonality, invertibility, and the ability to reconstruct the original signal perfectly from its frequency components via the inverse DFT. The DFT is mathematically clean but computationally expensive: computed naively, it requires $O(N^2)$ multiplications, which for a one-second audio clip at CD quality means roughly two billion operations. For most of the history of digital signal processing, this was a genuine bottleneck.

In 1965, Cooley and Tukey published the Fast Fourier Transform, or FFT, an algorithm that computes the DFT in $O(N \log N)$ operations by exploiting the symmetry of the complex exponentials. The key insight is that the DFT of a sequence of length N can be expressed in terms of two DFTs of sequences of length

$N/2$, each of which can be expressed in terms of two DFTs of length $N/4$, and so on recursively. The reduction from N^2 to $N \log N$ is enormous in practice: for $N = 44100$, this is the difference between roughly two billion operations and roughly seven hundred thousand. The FFT made real-time audio processing possible, and it is not an exaggeration to say that modern music production, digital audio workstations, streaming compression, audio fingerprinting, and autotune all run, somewhere underneath, on the FFT.

MP3 compression is a particularly striking example. The MP3 format works by computing the Fourier spectrum of short windows of audio, identifying components that the human auditory system cannot perceive — either because they are too quiet, or because they are masked by louder nearby frequencies — and discarding them. The psychoacoustic model that determines what to discard is itself based on the frequency-selective nature of the cochlea, which performs a biological approximation of the Fourier transform by having different regions respond to different frequencies. The compression algorithm is exploiting the fact that the ear is a Fourier analyser, and it throws away precisely the frequency information that the ear would not have detected anyway. An MP3 file is a carefully chosen subset of a Fourier spectrum.

Autotune works by a related but distinct mechanism. The pitch detection algorithm computes the autocorrelation of the audio signal, which is intimately related to the Fourier transform by the Wiener–Khinchin theorem, to identify the fundamental frequency in real time. It then resamples the signal to shift that fundamental to the nearest desired pitch, a process that is equivalent to stretching or compressing specific regions of the frequency spectrum. The characteristic robotic sound of heavy autotune is an artefact of this resampling: when the pitch correction is large, the resampling introduces phase discontinuities that manifest as the metallic, stepped quality associated with artists like T-Pain or the vocoder sections of Daft Punk. The aesthetic that defined a genre of music is a side effect of a numerical error in a Fourier-based algorithm.

What I find most remarkable about all of this is that Fourier had none of it in mind. He was trying to model how heat spreads through a metal plate. The claim that any function can be decomposed into sinusoids was a tool for solving a partial differential equation, not a theory of sound or a blueprint for digital audio. The fact that the same mathematics describes the vibration of a violin string, the compression of a music file, the way your cochlea processes sound, and the robotic artifacts in a pop song is not a coincidence engineered by engineers who knew what they wanted. It is the discovery, made gradually over two centuries, that sinusoids are in some deep sense the natural basis for oscillatory phenomena. The universe seems to prefer them, and Fourier's theorem is the precise statement of why.

There is a version of the Fourier transform lurking in almost every corner of modern science and technology, from quantum mechanics to medical imaging to gravitational wave detection. But music is where it lives most vividly, because music is something everyone experiences and almost nobody analyses. The next time you put on a track and hear the bass separate cleanly from the treble, or notice the particular warmth of a piano versus the brightness of a guitar, you are perceiving Fourier structure directly. Your brain is doing functional analysis. It has been doing it your whole life. It just never told you the name.