

Explained: The Discrete Fourier Transform

November 25 2009, by Larry Hardesty



(PhysOrg.com) -- In 1811, Joseph Fourier, the 43-year-old prefect of the French district of Isère, entered a competition in heat research sponsored by the French Academy of Sciences. The paper he submitted described a novel analytical technique that we today call the Fourier transform, and it won the competition; but the prize jury declined to publish it, criticizing the sloppiness of Fourier's reasoning. According to Jean-Pierre Kahane, a French mathematician and current member of the academy, as late as the early 1970s, Fourier's name still didn't turn up in the major French encyclopedia the Encyclopædia Universalis.

Now, however, his name is everywhere. The Fourier transform is a way to decompose a signal into its constituent frequencies, and versions of it are used to generate and filter cell-phone and Wi-Fi transmissions, to compress audio, image, and video files so that they take up less bandwidth, and to solve differential equations, among other things. It's

so ubiquitous that “you don’t really study the Fourier transform for what it is,” says Laurent Demanet, an assistant professor of applied mathematics at MIT. “You take a class in signal processing, and there it is. You don’t have any choice.”

The Fourier transform comes in three varieties: the plain old Fourier transform, the Fourier series, and the discrete Fourier transform. But it’s the discrete Fourier transform, or DFT, that accounts for the Fourier revival. In 1965, the computer scientists James Cooley and John Tukey described an algorithm called the fast Fourier transform, which made it much easier to calculate DFTs on a computer. All of a sudden, the DFT became a practical way to process [digital signals](#).

To get a sense of what the DFT does, consider an MP3 player plugged into a loudspeaker. The MP3 player sends the speaker audio information as fluctuations in the voltage of an electrical signal. Those fluctuations cause the speaker drum to vibrate, which in turn causes air particles to move, producing sound.

An audio signal’s fluctuations over time can be depicted as a graph: the x-axis is time, and the y-axis is the voltage of the electrical signal, or perhaps the movement of the speaker drum or air particles. Either way, the signal ends up looking like an erratic wavelike squiggle. But when you listen to the sound produced from that squiggle, you can clearly distinguish all the instruments in a symphony orchestra, playing discrete notes at the same time.

That’s because the erratic squiggle is, effectively, the sum of a number of much more regular squiggles, which represent different frequencies of sound. “[Frequency](#)” just means the rate at which air molecules go back and forth, or a voltage fluctuates, and it can be represented as the rate at which a regular squiggle goes up and down. When you add two frequencies together, the resulting squiggle goes up where both the

component frequencies go up, goes down where they both go down, and does something in between where they're going in different directions.

The DFT does mathematically what the human ear does physically: decompose a signal into its component frequencies. Unlike the analog signal from, say, a record player, the digital signal from an MP3 player is just a series of numbers, representing very short samples of a real-world sound: CD-quality digital audio recording, for instance, collects 44,100 samples a second. If you extract some number of consecutive values from a digital signal — 8, or 128, or 1,000 — the DFT represents them as the weighted sum of an equivalent number of frequencies. (“Weighted” just means that some of the frequencies count more than others toward the total.)

The application of the DFT to wireless technologies is fairly straightforward: the ability to break a signal into its constituent frequencies lets cell-phone towers, for instance, disentangle transmissions from different users, allowing more of them to share the air.

The application to data compression is less intuitive. But if you extract an eight-by-eight block of pixels from an image, each row or column is simply a sequence of eight numbers — like a digital signal with eight samples. The whole block can thus be represented as the weighted sum of 64 frequencies. If there's little variation in color across the block, the weights of most of those frequencies will be zero or near zero. Throwing out the frequencies with low weights allows the block to be represented with fewer bits but little loss of fidelity.

Demanet points out that the DFT has plenty of other applications, in areas like spectroscopy, magnetic resonance imaging, and quantum computing. But ultimately, he says, “It's hard to explain what sort of impact Fourier's had,” because the Fourier transform is such a

fundamental concept that by now, “it’s part of the language.”

Provided by Massachusetts Institute of Technology ([news](#) : [web](#))

Citation: Explained: The Discrete Fourier Transform (2009, November 25) retrieved 30 April 2024 from <https://phys.org/news/2009-11-discrete-fourier.html>

This document is subject to copyright. Apart from any fair dealing for the purpose of private study or research, no part may be reproduced without the written permission. The content is provided for information purposes only.