Check for updates

SIGNAL PROCESSING

Revisiting signal analysis in the big data era

A fast and accurate time–frequency analysis is challenging for many applications, especially in the current big data era. A recent work introduces a fast continuous wavelet transform that effectively boosts the analysis speed without sacrificing the resolution of the result.

Madhur Srivastava

In the first decade of the 1900s, when Albert Einstein was making breakthroughs in physics, there was a silent revolution happening in the yet-to-be-born field of signal processing: the invention of varying time–frequency analysis that later became popular as wavelet transforms[1]. Wavelet transforms simultaneously probe signal data based on its index location and its variation in magnitude, and allow multiresolution analysis of the signal by customizing time and frequency resolutions (that is, by providing flexibility in zooming in and out). With the advent of computers and digital applications, the multiresolution property of wavelet transforms has seen applications in different fields, such as medicine, image and video processing, telecommunication, industrial processing, and fundamental science. This wavelet transform revolution was spearheaded by the discrete wavelet transform[2] (DWT), a relatively fast transform to compute as it reduces the number of data points to a necessary minimum to reconstruct the original signal, which was a boon to computers in the 1980s and 1990s owing to their limited computing and data-storage capabilities. Moreover, fewer data points were necessary because signals needed to be compressed, stored and transmitted with a small bandwidth. However, DWT has low resolution, which limits its application in advanced signal analysis and processing that require high accuracy. Continuous wavelet transform[3] (CWT) was developed as an alternative that improves resolution when compared with DWT, but at the same time substantially increases the computational cost. This trade-off between computational cost and resolution limits the application of wavelet transforms for real-time and accurate signal processing. Writing for *Nature Computational Science*, Arts and van den Broek overcome this problem by introducing fast CWT (fCWT)[4].

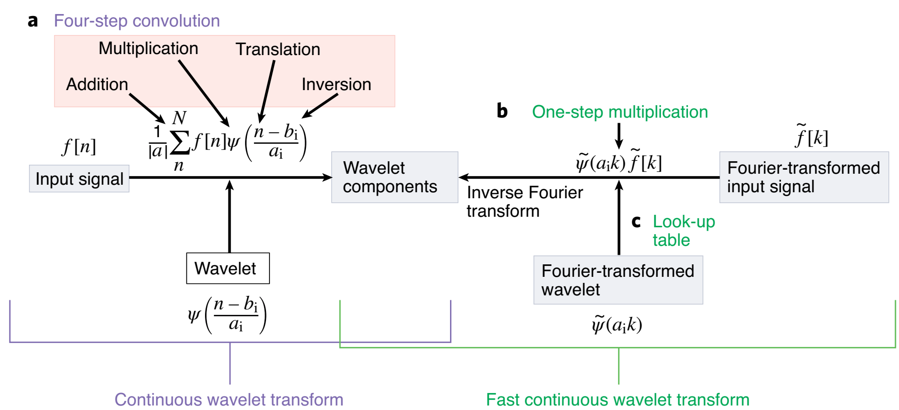

Fig. 1 | The fast continuous wavelet transform process and its comparison with the standard continuous wavelet transform. a, In CWT, the wavelet is used to perform the time–frequency of the input signal using parameters ‘a’ and ‘b’, where ‘a’ represents the time resolution and wavelet window length, and ‘b’ is used to select the frequency resolution. **b**, A representation of the fCWT process. While CWT is performed in the time domain via the process of convolution, which is computationally expensive with its four-step process of inversion, translation, multiplication and addition, fCWT overcomes this problem by carrying out operations in the Fourier domain. The four-step process of convolution in the time domain translates into a single step of multiplication in the Fourier domain. **c**, The wavelet function in the Fourier domain can be pre-calculated and obtained from the look-up table. *n* and *k* are the index numbers for the time and frequency domain signal, respectively.

A CWT yields multiple wavelet-transformed signals called wavelet components, each of which represent a different time–frequency analysis of an original signal. In other words, wavelet components are analogous to color bands of white light, where both composition as well as intensity associated with each color are represented. In time–frequency analysis, each wavelet component represents a frequency resolution, and coefficients in the wavelet component represent a time resolution. However, unlike color bands that can be obtained using a single glass prism, each wavelet component must be independently calculated, and thus the calculation of the components naturally becomes computationally expensive. For calculating a wavelet component, a wavelet preset to a desired time–frequency is used to process the input signal (Fig. 1, a_i and b_i parameters). The wavelet is then updated for the next iteration of time–frequency analysis of the signal for obtaining another wavelet component. For each iteration of finding a wavelet component, the time complexity is of the order O(N), which includes the computing time associated with processing the input signal of length N and with maintaining the required time resolution. With M iterations representing different frequency resolutions, the computational complexity dramatically increases to the polynomial order of O(MN). To reduce computation time, one may either reduce M, which compromises frequency resolution, or decrease N, which compromises time resolution.

In their most recent work[4], Arts and van den Broek devised fCWT to reduce the computation time without compromising frequency and time resolutions. They achieve this by reducing the number of operations and by utilizing pre-calculated look-up tables. A CWT obtains the wavelet transform in the time domain using the process of convolution (that is, inversion, translation, multiplication and addition) between the input signal and the wavelet (Fig. 1a). A fCWT reduces this four-step process into a single step of multiplication (Fig. 1b) between the input signal and the wavelet by carrying out the process in the Fourier domain, which saves a substantial number of operations. Further, the look-up table (Fig. 1c) is used to store the Fourier-transformed wavelet coefficients, avoiding the need for a calculation to obtain each wavelet component. The Fourier domain data is then converted into wavelet components by using inverse fast Fourier transform (Fig. 1b). Because the computational speed-up does not require the removal of any data point in the signal, the resolution of the wavelet-transformed signal is unaffected.

Arts and van den Broek demonstrated the efficacy of fCWT by implementing it on electroencephalography (EEG) and in in vivo electrophysiology signals[4]. Electroencephalography has an essential role in measuring brain activity, but its signals can be influenced by noise. Averaging multiple traces reduces noise, but at the same time, it reduces temporal resolution, defeating the purpose of EEG of measuring time-dependent brain activity. With real-time fast computations, fCWT yields high resolution for a single trace, allowing for the observation of time-dependent fluctuations in brain activity. Similarly, local field potential electrophysiology signals are a potent tool for studying neuron function, where low and high signal frequencies are associated with different tasks from long-distance communication to local neural processing. To study connection and association between these tasks, it becomes essential to have real-time high-resolution analysis at all frequency ranges, which has been demonstrated by using fCWT. It is worth noting that fCWT can be potentially applied to other signals, such as those obtained from biophysical microscopy and spectroscopy for advanced data analysis.

In the era of big data, high-resolution signal analysis becomes increasingly essential, and fCWT is a useful tool for embracing this new era. The gains in computation increase asymptotically, meaning that the larger the data length, the higher the gains. Fast CWT enables the analysis of signals for many applications that desire high resolution, but that are limited by the high computation cost of processing a large data set. For example, with a reduced computational cost, fCWT will enable real-time signal de-noising of biophysical signals that currently use the slow process of signal averaging, and further avoid other side effects originating from long acquisition time, such as biological sample breakdown.

While fCWT provides an excellent platform for taking wavelet transforms, the following questions remain untouched. Should the wavelet components, contain non-overlapping time–frequency information of a signal? What time–frequency resolution for a given signal is appropriate? How many wavelet components, that is, *M*, is desirable for a signal? These are application-specific questions and the next steps will involve addressing them. One would certainly want to avoid obtaining more wavelet components than is needed, which inherently defeats the purpose of faster, real-time computation with fCWT. ❐

Madhur Srivastava [1,2] ✉
[1]*National Biomedical Center for Advanced ESR Technology, Cornell University, Ithaca, NY, USA.*
[2]*Department of Chemistry and Chemical Biology, Ithaca, NY, USA.*
✉*e-mail: madhur.srivastava@cornell.edu*

Published online: 24 February 2022
https://doi.org/10.1038/s43588-022-00210-7

References

1. Daubechies, I. *Ten Lectures on Wavelets* (USA Society for Industrial and Applied Mathematics, 1992).
2. Shena, M. J. *IEEE Trans. Signal Process.* **40**, 2464–2482 (1992).
3. Addison, P. S. *Phil. Trans. R. Soc. A* **376**, 20170258 (2018).
4. Arts, L. P. A. & van den Broek, E. L. *Nat. Comput. Sci.* https://doi.org/10.1038/s43588-021-00183-z (2022).

Competing interests

The author declares no competing interests.