What is Fast Fourier Transform?

Technical Report · August 2022

DOI: 10.13140/RG.2.2.28731.49444

1 author:

Amin Gasmi
Société Francophone de Nutrithérapie et de Nutrigénétique Appliquée

What is Fast Fourier Transform?

Amin Gasmi

What is Fast Fourier Transform?

Amin Gasmi

1er Août 2022

Historical background

The history of the Fast Fourier Transform (FFT) is of an interesting nature. It began in the year 1805 when Carl Friedrich Gauss tried to calculate and determine the orbit of certain asteroids from sample locations which led to the development of Discrete Fourier Transform [1]. In calculating the DFT, he invented an algorithm which was similar and equivalent to that which was developed by Cooley and Tukey. However, this method was more general than that of Cooley and Tukey [1].

Jean-Baptiste-Joseph, Baron Fourier (1768-1830) a French mathematician is known for developing various mathematical concepts which now find their application in mathematics and engineering through his concept of "The Analytical Theory of Heat" where he was able to show that conduction of heat in solids can be analyzed in terms of infinite mathematical series now called the Fourier series. In the second part of the 19th century and the 20th-century scientist and mathematicians improved the idea of Fourier to become what it is today where the idea has been expanded and therefore find its application in almost all the engineering field [2]. Signal processing is one of such fields where Fourier Transform is applied.

Introduction

A signal can be defined as a physical variable that changes with time and contains information. There are two basic forms of representing a signal, either in analogue (continuous) or discrete (digital) form [3]. In engineering, most variables of interest are of analogue form however, modern digital data acquisition equipment facilitates digital representation of these analogue signals. Digital representation of analogue signal usually affect the characteristics of the signal, therefore, it is important for one to understand the underlying principles in signal processing [4].

Signal processing and analysis are not only an important aspect of electrical and electronic engineering, and telecommunication engineering but also all other engineering sectors and physiological processes. Fourier transform is the key tool in the analysis of the signal in the time domain and frequency domain [5]. The algorithmic application of the Fourier Transform is the best description of the Fast Fourier Transform since it make the use of discrete samples of a waveform in time domain. The captured signal spectrum frequency domain representation is provided by the time domain data after undergoing Fourier Transform [6]. FFT provides an algorithm for the computation of the Fourier coefficient faster and with the use of less effort. It produced important changes in the methods used in the computation of digital spectral analysis, filter simulation, and related field [2].

It has been shown that the fast Fourier transform (FFT) provides an excellent mechanism for the computation of the Discrete Fourier transform (DFT) of a time series, that is, discrete data samples. The interest in this technique is in the rise due to its substantial and economic way of obtaining the solution to complex spectral problems [7]. The Discrete Fourier transform is a powerful reversible mapping operation usually applied in the time series. Its mathematical properties are similar and analogous to Fourier integral transform where it primarily defines a spectrum of a time series; multiplication of the transform of two-time series parallels to convolving the time series [2]. In the computation of DFT of a time series, the fast Fourier transform (FFT) is highly recommended since it takes the advantage of the fact that coefficients of the DFT can be calculated iteratively and thus leads to a substantial reduction of time used in the computation [2].

There are various know applications where a substantial reduction in time of computation is important and has been achieved. These areas include; filter simulation, decomposing of convolved functions, computation of the power spectra and other autocorrelation functions of sampled data, computation of cepstra, bispectra, cross-covariance functions, and related functions, and pattern recognition by using a two-dimensional form of the DFT [8]. The similarity in these applications is that the fast Fourier transform (FFT) has been incorporated to reduce the computation time.

General Description of the Fast Fourier Transform

When dealing with FFT, it is essential to note that in addition to reducing the computation time, Fast Fourier Transform substantially reduces systematic errors (round-off errors) associated with these computations. According to [4] mentions that both computation time and round-off error are usually reduced by a factor of (log2N)/N where N represents the number of data samples in the time series.

According to [2], if $N = 1024 = 2^{10}$ then, $N \bullet \log_2 N = 10240$.

In the conventional methods for computing for $N = 1024$ would require an effort which is proportional to $N^2 = 1048576$, which is more than 50 times that required with the Fast Fourier Transform (FFT).

Fast Fourier Transform is a technique which sequentially combines progressively larger weighted sums of data samples thereby producing the required DFT coefficients. The FFT technique can be interpreted in terms of combining the DFTs of the individual data samples which may be taken as the number of occurrences of the samples.

Consider the definition of DFT and its inverse below

$$A_r = \sum_{k=0}^{N-1} (X_k) W^{rk} \qquad \text{r = 0, …, N – 1;}$$

Where

$$W = \exp(-2\pi j / \mathrm{N})$$

And its inverse

$$X_l = \frac{1}{N} \sum_{k=0}^{N-1} (\mathrm{A}_r) W^{-rl} \qquad \text{l = 0, 1, …, N – 1;}$$

It is clear that the DFT and its inverse are of the same form and therefore a subroutine which is used to calculate one can be employed to compute the other by simply exchanging the role of Xk and Ar, and making appropriate scale-factor and sign changes [2]. This, therefore, means that the two basic forms of the FFT are similar and equivalent. The forms are the decimation in time (Time Domain Analysis) and exchanging the role of Xk and Ar gives decimation in frequency (Frequency Domain Analysis).

Time Domain Analysis of FFT

We consider a time series having N samples such as Xk which can be divided into two functions i.e. Yk and Zk with each having half of the total points i.e. (N/2). We take the function Yk to be composed of even-numbered points (X_0, X_2, X_4 …), and we also take Zk to be composed of the odd-numbered points (X_1, X_3, X_5 …). The functions of Yk and Zk can be written formally as;

$$\mathrm{Y_k} = \mathrm{X_{2k}}$$

$$\mathrm{Z_k} = \mathrm{X_{2k+1}}$$

Where

k = 0, 1, 2, …, N/2 - 1

But Y_k and Z_k are sequences of N/2 points each, therefore they have a discrete Fourier transform defined by;

$$B_r = \sum_{k=0}^{(N/2)-1} Y_k \exp(-4\pi jrk / N)$$

$$C_r = \sum_{k=0}^{(N/2)-1} Z_k \exp(-4\pi jrk / N)$$

Where

$$r = 0, 1, 2, \ldots, N/2 - 1$$

The DFT we expect is A_r. A_r can be written in terms of odd and even numbered points as;

$$A_r = \sum_{k=0}^{(N/2)-1} \left\{ Y_k \exp(-4\pi jrk / N) + Z_k \exp\left(-\frac{2\pi jr}{N} \left[2k+1 \right] \right) \right\}$$

Where; r = 0, 1, 2, …, N - 1

Which can be written in the form;

$$A_r = B_r + \exp\left(-\frac{2\pi jr}{N}\right)C_r \qquad 0 \leq r < N/2$$

For values of ***r*** greater than N/2***,*** the discrete Fourier transforms ***Br*** and ***Cr,*** repeat periodically the values taken on when r is less than N/2***.*** Therefore, substituting (r + N/2) for r in the previous equation, we obtain;

$$A_{r+N/2} = B_r + \exp\left(-\frac{2\pi j\left[r+\frac{N}{2}\right]}{N}\right)C_r \quad 0 \leq r < N/2$$

$$= B_r - \exp\left(-\frac{2\pi jr}{N}\right)C_r \qquad 0 \leq r < N/2$$

Which can be written as;

$$A_r = B_r + W^r C_r \quad 0 \leq r < N/2$$

$$A_{r+N/2} = B_r - W^r C_r \qquad 0 \leq r < N/2$$

From the two equations above, both the first and the last N/2 points of the DFT of X_k can easily be found from the DFT of Yk and Zk, both sequence of N/2 samples.

When we assume that we have a method which computes discrete Fourier transforms in a time domain which varies linearly to the square of the number of samples, we can use this algorithm to calculate the transforms of Yk and Zk, requiring a time varying linearly with $2(N/2)^2$ and the last two equations to find Ar with additional N operations.

And since to calculate the Discrete Fourier Transform of ***N*** samples, we can reduce such calculation by considering Discrete Fourier Transforms with two $N/2$ samples sequences, similarly, the calculation of Bk (or Ck***)*** can be reduced to the calculation of order of N/2 samples. If each sample is divisible by 2 i.e. N = 2^n. Thus generally, N.$\log_2$N complex addition and, at most, 1/2.N. log2N complex multiplications must be obtained for calculation of the DFT of an ***N*** point order, where ***N*** is a power of ***2.*** From these illustrations, we see that a complex DFT has been simplified using FFT.

Frequency Domain Analysis of FFT

FFT has been considered as the most important modern tools when it comes to signal processing. FFT analysis with the use of zoom is applied in frequency domain where it provides high resolution. Zoom is also used in vibration analysis of some frequency band [9].

Assuming a time series Xk have a DFT Ar. Both the series and the DFT contain N terms. We divide Xk into two sequences having N/2 points each. Contrary to the time domain, the first sequence, Yk, is composed of the first N/2 points in Xk and Zk is composed of the last N/2 points in Xk [10]. Then;

$$\mathrm{Y_k} = \mathrm{X_k}$$

$$\mathrm{Z_k} = \mathrm{X}_{\mathrm{k}+N/2}$$

Where k = 0, 1, 2, ..., N/2 – 1

The N point DFT of Xk may therefore, be written in terms of Yk and Zk as

$$A_r = \sum_{k=0}^{(N/2)-1} \left\{ Y_k \exp(-2\pi jrk/N) + Z_k \exp\left(-\frac{2\pi jr\left[k + \frac{N}{2}\right]}{N} \right) \right\}$$

$$A_r = \sum_{k=0}^{(N/2)-1} \left\{ Y_k + \left[\exp(-\pi jr)\right] Z_k \right\} \exp\left(-\frac{2\pi jrk}{N} \right)$$

Considering the odd-numbered and even-numbered points of the transform separately, and letting even-numbered points be Rr and odd-numbered points be Sr where;

$$R_r = A_{2r}$$

$$S_r = A_{2r+1}$$

Where $0 \leq r < N/2$

This step may be referred to as decimation in frequency [1].

Computing the spectrum points which are even-numbered yields;

$$R_r = A_{2r} = \sum_{k=0}^{(N/2)-1} \{Y_k + Z_k\} \exp\left(-\frac{2\pi jrk}{(N/2)}\right)$$

A close check on the above equation reveals that it is the N/2 point Discrete Fourier Transform of the function (Y_k+Z_k), which is the summation of the first and the last N/2 time samples.

The odd-numbered spectrum points similarly become;

$$S_r = A_{2r+1} = \sum_{k=0}^{(N/2)-1} \{Y_k + Z_k \exp(-\pi j)[2r+1]\} \bullet \exp(-2\pi j[2r+1]k/\mathrm{N})$$

$$= \sum_{k=0}^{(N/2)-1} \{Y_k - Z_k\} \bullet \exp\left(-\frac{2\pi jk}{N}\right) \bullet \exp\left(-\frac{2\pi jk}{(N/2)}\right)$$

Which a close cross-check reveals to be the N/2 point Discrete Fourier Transform of the function (Yk-Zk) exp (-2πjk/N).

It therefore, follows that both methods of spectral analyses (time and frequency) require N/2.logN complex additions, subtractions, and multiplication [2]. Therefore from the two analyses, it is clear that Fast Fourier Transform reduces the complexity of computing Discrete Fourier Transform.

Use of FFT in Signal Processing

When implementing Fast Fourier Transform, many applications require that N be of the form N^x where x be equal to 2. If the sample I not of the power 2, the signal is "pad" using a virtual sample of zero value at the end in a process termed "zero padding" [11]. For instance, given 1000 samples, padding of the signal can be achieved with the use 24 zeros which attains to $2^{10} = 1024$ samples.

The mathematical concept and formulation of FFT and Discrete Fourier Transform relies on the assumption of the repetitive nature of the signal after a given number of samples. This implies that FFT and DFT may contain a vital constituent in the lowest non-zero frequency, usually represented as $\left(f_s / N\right)$, even though such frequency may be absent in the initial signal [12]. From this property, the following can be inferred:

1. When the original signal does not have periodic properties (i.e. is non-periodic), isolation of the lowest potion in the repeated signal having nonzero frequency from $\left(f_s / N\right)$ can be achieved through zero padding [11].
2. Supposing that the initial signal is periodic in nature and has specific frequency *f*, the *N* number samples of this signal should be made up of many complete oscillations. This implies that $\left(f_s / N\right)$ frequency is too small compared to the actual frequency *f*.
3. Whenever the original signal is considered periodic and the signal possesses a specific frequency *f*, the number of cycles for such a signal samples should be an integer. In such a scenario, $\left(f_s / N\right)$frequency should have very low modulus [11].

Software and Means of FFT Calculation

With the advancement in technology, there are a number of computer software which has been programmed for the spectral analysis [13]. Most of these software applies Fast Fourier Transform since it requires smaller memory and storage compared to following the long procedure of Discrete Fourier Transform.

Some of the software packages which make use of Fast Fourier Transform available in the market include [14]:

1. MATLAB by MathWorks Inc.
2. SpectraPLUS by Pioneer Hill Software LLC.
3. ME'Scope by Vibrant Technology
4. Spectrum Analyser by Keuwlsoft
5. SciLab by the Scilab team within ESI Group

References

[1] T. H. Michael, H. J. Don and B. C. Sideny, "Gauss and the History of the Fast Fourier Transform," *Archive for History of Exact Sciences (Springer),* vol. 34, no. 3, p. 265{277, 1985.

[2] W. Cochran, J. Cooley, D. Favin, H. Helms, R. Kaenel, W. Lang, G. Maling, D. Nelson, C. Rader and P. Welch, "What is the fast Fourier transform?," *Proceedings of the IEEE,* vol. 55, no. 10, p. 1664—1674, 1967.

[3] A. Kaiser H, Digital Signal Processing using the Fast Fourier Transform (FFT), Loughborough University, 1997.

[4] K. Ivan P., L. Tingye and E. W. Alan, Optical Fiber Telecommunications Systems and Networks, 6 ed., Academic Press, 2013.

[5] H. R. Taylor, "Book Review: The Fast Fourier Transform and its Applications," *International Journal of Electrical Engineering Education,* vol. 27, no. 3, 1990.

[6] E. Chu, Discrete and Continuous Fourier Transforms: Analysis, Applications and Fast Algorithms, Chapman and Hall/CRC., 2008.

[7] D. E. Smith, E. J. Powers and G. S. Caldwell, "Fast-Fourier-Transform Spectral-Analysis Techniques as a Plasma Fluctuation Diagnostic Tool," *IEEE Transactions on Plasma Science,* vol. 2, no. 4, 1974.

[8] J. W. H. Wong, C. Durante and H. M. Cartwright, "Application of Fast Fourier Transform Cross-Correlation for the Alignment of Large Chromatographic and Spectral Datasets," *Analytical Chemistry,* vol. 77, no. 17, p. 5655—5661, 2005.

[9] H. J. Nussbaumer, Fast Fourier transform and Convolution Algorithms, vol. 2, Springer Science & Business Media, 2012.

[10] H. Sorensen, D. Jones, M. Heideman and C. Burrus, "Corrections to "Real-valued fast Fourier transform algorithms," *IEEE Transactions on Acoustics Speech and Signal Processing,* vol. 35, no. 9, 1987.

[11] L. Kevin M., M. Nicholas and E. Matthew L., Embedded Computing and Mechatronics with the PIC32 Microcontroller, Newnes, 2015.

[12] L. Yaguo, Intelligent Fault Diagnosis and Remaining Useful Life Prediction of Rotating Machinery, Butterworth-Heinemann, 2017.

[13] D. F. Elliott, Handbook of Digital Signal Processing: Engineering Applications, Elsevier, 2013.

[14] Digiducer Inc, "Digiducer - Software," 2019. [Online]. Available: https://digiducer.com/pages/software. [Accessed 28 May 2019].