1.6.4 FAST FOURIER TRANSFORM (FFT) ALGORITHM

The FFT is an *algorithm* for efficient computation of the DFT.

It is *not* a new transform.

James William Cooley

Born September 18, 1926; with John Tukey, creator of the fast Fourier transform.

Education: BA, arts, Manhattan College, 1949; MA, mathematics, Columbia University, 1951; PhD, applied mathematics, Columbia University, 1961.

Professional Experience: programmer, Institute for Advanced Study, Princeton University, 1953-1956; research assistant, mathematics, Courant Institute, New York University, 1956-1962; research staff, IBM Watson Research Center, 1962-1991; professor, electrical engineering, University of Rhode Island, 1991-present.

Honors and Awards: Contribution Award, Audio and Acoustics Society, 1976; Meritorious Service Award, ASSP Society, 1980; Society Award, Acoustics Speech and Signal Processing, 1984; IEEE Centennial Award, 1984; fellow, IEEE

James W. Cooley started his career in applied mathematics and computing when he worked and studied under Professor F.J. Murray at Columbia University. He then became a programmer in the numerical weather prediction group at John von Neumann's computer project at the Institute for Advanced Study in Princeton, New Jersey. [See the biography of Jule Charney.] In 1956, he started working as a research assistant at the Courant Institute at New York University, New York. Here he worked on numerical methods and programming of quantum mechanical calculations (Cooley 1961). This led to his thesis for his PhD degree from Columbia University.

In 1962 he obtained a position as a research staff member at the IBM Watson Research Center in Yorktown Heights, New York. Here he worked on numerical methods for solving ordinary and partial differential equations, solutions of hole-electron diffusion equations for semiconductors, and numerous other research projects. He collaborated with Fred Dodge, a neurophysiologist, in research in neurophysiology including modeling of electrical activity in nerve membranes and in heart muscle (Cooley and Dodge 1966).

With John Tukey, he wrote the fast Fourier transform (FFT) paper (Cooley and Tukey 1965) that has been credited with introducing the algorithm to the digital signal processing and scientific community in general.

Cooley spent the academic year 1973-1974 on a sabbatical at the Royal Institute of Technology, Stockholm, Sweden. He gave courses on the FFT and its applications there and in several other locations in Europe and worked on new versions of the FFT and on number theoretic Fourier transforms.

In 1974 Cooley started collaboration with S. Winograd and R. Agarwal on applications of computational complexity theory to convolution and Fourier transform algorithms (Agarwal and Cooley 1977).

Around 1985 he worked with a group that programmed the elementary functions for the new IBM 3090 Vector Facility. He and the same group also produced the Digital Signal Processing subroutines for the Engineering and Scientific Subroutine Library (ESSL) for the IBM 3090 Vector Facility and, later, for the new IBM RS6000 computer (Agarwal and Cooley 1987).

Cooley retired from IBM in 1991 and joined the faculty of the Electrical Engineering Department of the University of Rhode Island as director of the computer engineering program.

BIBLIOGRAPHY

Significant Publications

Agarwal, R.C., and J.W. Cooley, "New Algorithms for Digital Convolution," *IEEE Trans. Acoust., speech, Signal Processing*, Vol. ASSP-25, No. 5, Oct. 1977, pp. 392-410.

Agarwal, R.C., and J.W. Cooley, "Vectorized Mixed Radix Fourier Transform Algorithms," *IEEE Proc.*, Sept. 1987, pp. 1283-1292.

Cooley, J.W., "An Improved Eigenvalue Corrector Formula for Solving the Schrodinger Equation for Central Fields," *Mathematics of Computation*, Vol. 15, No. 7, Oct. 1961, pp. 363-374.

Cooley, J.W., and F.A. Dodge, Jr., "Digital Computer Solutions for Excitation and Propagation of the Nerve Impulse," *Biophysical J.*, Vol. 6, No. 5, 1966.

Cooley, J.W., and J.W. Tukey, "An Algorithm for the Machine Calculation of Complex Fourier Series," *Mathematics of Computation*, Vol. 19, Apr. 1965, p. 297.

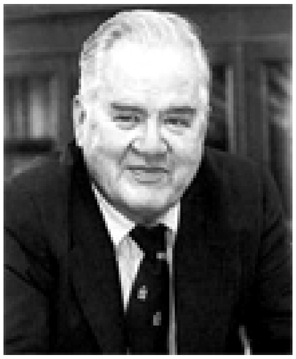
UPDATES

Portrait changed (MRW, 2012)

PDF version

John Wilder Tukey

Born June 16, 1915, New Bedford, Mass.; with James Cooley, the creator of the fast Fourier transform (FFT) and possible creator of the word ‚bit.[1]



Education: ScB, chemistry, Brown University, 1936; ScM, Brown University, 1937; MA, Princeton University, 1938; PhD, mathematics, Princeton University, 1939.

Professional Experience: Princeton University: instructor, mathematics, 1939-1941, research associate, Fire Control Research Office, 1941-1945, professor, statistics, 1965-present, Donner Professor of Science, 1976-present; Bell Telephone Laboratories: member, technical staff, 1945-1948, assistant director, Research Communication Principles, 1958-1961, associate executive director, Research Communication Principles Division, 1961-present.

Honors and Awards: DSc (Hon.), Case Institute of Technology, 1962; DSc (Hon.), Brown University, 1965; S.S. Wilks Medal, American Statistical Association, 1965; DSc (Hon.), Yale University, 1968; DSc (Hon.), University of Chicago, 1969; National Medal of Science, 1973; IEEE Medal of Honor, 1982; member, National Academy of Sciences; member, American Academy of Arts and Sciences; honorary member, Royal Statistical Society.

BIBLIOGRAPHY

Biographical

Tropp, H.S., "Origin of the Term Bit," *Ann. Hist. Comp.*, Vol. 6, No. 2, Apr. 1984, pp. 152-155.

Significant Publications

Cooley, J.W., and J.W. Tukey, "An Algorithm for the Computation of Complex Fourier Series," *Math. of Computation*, Apr. 1965.

UPDATES

John Tukey died July 26, 2000 (MRW, 2013).

---

[1] See Tropp 1984.

John Tukey

From Wikipedia, the free encyclopedia

John Tukey	
Born	June 16, 1915 New Bedford, Massachusetts, USA
Died	July 26, 2000 (aged 85) New Brunswick, NJ, USA
Residence	USA
Nationality	American
Field	Mathematician
Institutions	Bell Labs Princeton University
Alma mater	Brown University Princeton University
Academic advisor	Solomon Lefschetz
Notable students	Frederick Mosteller
Known for	FFT algorithm Box plot
Notable prizes	IEEE Medal of Honor (1982)

John Wilder Tukey (June 16, 1915 - July 26, 2000) was a statistician born in New Bedford, Massachusetts.

Tukey obtained a B.A. in 1936 and M.Sc. in 1937, in Chemistry, from Brown University, before moving to Princeton University where he received his Ph.D. in mathematics. During World War II, Tukey worked at the Fire Control Research Office and collaborated with Samuel Wilks and William Cochran. After the war, he returned to Princeton, dividing his time between the university and AT&T Bell Laboratories. He was awarded the IEEE Medal of Honor in 1982 "For his contributions to the spectral analysis of random processes and the fast Fourier transform (FFT) algorithm."

Contents

- 1 Scientific contributions
- 2 Quotes
- 3 Bibliography
- 4 External links
- 5 References

Scientific contributions

His statistical interests were many and varied. He is particularly remembered for his development with James Cooley of the Cooley-Tukey FFT algorithm. In 1970, he contributed significantly to what is today known as the jackknife estimation—also termed Quenouille-Tukey jackknife. He introduced the box plot in his 1977 book, *Exploratory Data Analysis*.

He also contributed to statistical practice and articulated the important distinction between exploratory data analysis and confirmatory data analysis, believing that much statistical methodology placed too great an emphasis on the latter. Though he believed in the utility of separating the two types of analysis, he pointed out that sometimes, especially in natural science, this was problematic and termed such situations uncomfortable science.

He wrote four papers with his fifth cousin Paul Tukey, who was an undergraduate at Princeton when they met.

Among many contributions to civil society, Tukey served on a committee of the American Statistical

Association that produced a report challenging the conclusions of the Kinsey Report, *Statistical Problems of the Kinsey Report on Sexual Behavior in the Human Male*.

Tukey coined many statistical terms that have become part of common usage, but the two most famous coinages attributed to him were related to computer science. While working with John von Neumann on early computer designs, Tukey introduced the word "bit" as a contraction of binary digit[1]. The term bit was first used in an article by Claude Shannon in 1948. Tukey used the term "software" in a computing context in a 1958 article for American Mathematical Monthly, the first published use of the term.[2]

A D Gordon offered the following summary of Tukey's principles for statistical practice:

> ... the usefulness and limitation of mathematical statistics; the importance of having methods of statistical analysis that are robust to violations of the assumptions underlying their use; the need to amass experience of the behaviour of specific methods of analysis in order to provide guidance on their use; the importance of allowing the possibility of data's influencing the choice of method by which they are analysed; the need for statisticians to reject the role of 'guardian of proven truth', and to resist attempts to provide once-for-all solutions and tidy over-unifications of the subject; the iterative nature of data analysis; implications of the increasing power, availability and cheapness of computing facilities; the training of statisticians.

He is also the creator of several little-known methods such as the trimean and Median-Median line, an easier alternative to linear regression.

Tukey retired in 1985. In 2000, he died in New Brunswick, New Jersey.

Quotes

- "Far better an approximate answer to the right question, which is often vague, than the exact answer to the wrong question, which can always be made precise." J. W. Tukey (1962), "The future of data analysis". *Annals of Mathematical Statistics* 33(1), pp. 1-67.
- "The combination of some data and an aching desire for an answer does not ensure that a reasonable answer can be extracted from a given body of data." J. W. Tukey (1986), "Sunset salvo". *The American Statistician* 40(1). Online at http://www.jstor.org/view/00031305/di020589/02p0102y/0

Bibliography

- Hoaglin, D C; Mosteller, F & Tukey, John Wilder (Eds) (1985). *Exploring Data Tables, Trends and Shapes*. ISBN 0-471-09776-4.
- Hoaglin, D C; Mosteller, F & Tukey, John Wilder (Eds) (1983). *Understanding Robust and Exploratory Data Analysis*. ISBN 0-471-09777-2.
- Tukey, John Wilder (1977). *Exploratory Data Analysis*. ISBN 0-201-07616-0.

External links

- (published in the Annals of Statistics) *John W. Tukey: His Life and Professional Contributions* (http://www.stat.berkeley.edu/~brill/Papers/life.pdf)
- Memories of John Tukey (http://cm.bell-labs.com/cm/ms/departments/sia/tukey/index.html)

- Short biography (http://stat.bell-labs.com/who/tukey/bio.html) by Mary Bittrich
- Obituary (http://stat.bell-labs.com/who/tukey/nytimes.html)
- "John W. Tukey - The Man and His Many Achievements" (http://www.youtube.com/watch?v=isDlJvb1VN4) Video Biography
- "Remembering John W. Tukey" (http://projecteuclid.org/Dienst/UI/1.0/Summarize/euclid.ss/1076102421) , special issue of *Statistical Science*
- John Tukey (http://www.genealogy.ams.org/html/id.phtml?id=15860) at the Mathematics Genealogy Project

References

- O'Connor, John J; Edmund F. Robertson "John Tukey". *MacTutor History of Mathematics archive*.
- Interview of John Tukey about his experience at Princeton (http://infoshare1.princeton.edu/libraries/firestone/rbsc/finding_aids/mathoral/pmc41.htm)

Footnoted references

1. ^ The origin of the 'bit' (http://www.linfo.org/bit.html)
2. ^ John Tukey, 85, Statistician; Coined the Word 'Software', New York Times, Obituaries, July 28, 2000

Awards		
Preceded by **Sidney Darlington**	**IEEE Medal of Honor** 1982	Succeeded by **Nicolaas Bloembergen**

Retrieved from "http://en.wikipedia.org/wiki/John_Tukey"

Categories: 1915 births | 2000 deaths | Brown University alumni | Princeton University alumni | Princeton University faculty | American statisticians | National Medal of Science laureates | IEEE Medal of Honor recipients | Data analysis | Presidents of the Institute of Mathematical Statistics

---

An Algorithm for the Machine Calculation of Complex Fourier Series

By James W. Cooley and John W. Tukey

An efficient method for the calculation of the interactions of a 2^m factorial experiment was introduced by Yates and is widely known by his name. The generalization to 3^m was given by Box et al. [1]. Good [2] generalized these methods and gave elegant algorithms for which one class of applications is the calculation of Fourier series. In their full generality, Good's methods are applicable to certain problems in which one must multiply an N-vector by an $N \times N$ matrix which can be factored into m sparse matrices, where m is proportional to $\log N$. This results in a procedure requiring a number of operations proportional to $N \log N$ rather than N^2. These methods are applied here to the calculation of complex Fourier series. They are useful in situations where the number of data points is, or can be chosen to be, a highly composite number. The algorithm is here derived and presented in a rather different form. Attention is given to the choice of N. It is also shown how special advantage can be obtained in the use of a binary computer with $N = 2^m$ and how the entire calculation can be performed within the array of N data storage locations used for the given Fourier coefficients.

Consider the problem of calculating the complex Fourier series

$$(1) \qquad X(j) = \sum_{k=0}^{N-1} A(k) \cdot W^{jk}, \qquad j = 0, 1, \cdots, N-1,$$

where the given Fourier coefficients $A(k)$ are complex and W is the principal Nth root of unity,

$$(2) \qquad W = e^{2\pi i/N}.$$

A straightforward calculation using (1) would require N^2 operations where "operation" means, as it will throughout this note, a complex multiplication followed by a complex addition.

The algorithm described here iterates on the array of given complex Fourier amplitudes and yields the result in less than $2N \log_2 N$ operations without requiring more data storage than is required for the given array A. To derive the algorithm, suppose N is a composite, i.e., $N = r_1 \cdot r_2$. Then let the indices in (1) be expressed

$$(3) \qquad \begin{aligned} j &= j_1 r_1 + j_0, \qquad j_0 = 0, 1, \cdots, r_1 - 1, \qquad j_1 = 0, 1, \cdots, r_2 - 1, \\ k &= k_1 r_2 + k_0, \qquad k_0 = 0, 1, \cdots, r_2 - 1, \qquad k_1 = 0, 1, \cdots, r_1 - 1. \end{aligned}$$

Then, one can write

$$(4) \qquad X(j_1, j_0) = \sum_{k_0} \sum_{k_1} A(k_1, k_0) \cdot W^{jk_1 r_2} W^{jk_0}.$$

Received August 17, 1964. Research in part at Princeton University under the sponsorship of the Army Research Office (Durham). The authors wish to thank Richard Garwin for his essential role in communication and encouragement.

Recall

$$X^{(N)}(k) = \sum_{n=0}^{N-1} x(n)\, e^{-j2\pi kn/N}\ , \qquad k = 0, 1, \ldots, N-1$$

Superscript (N) is to show length of DFT. For each value of k, computation of X(k) requires:

N	complex multiplications
N−1	complex additions

Define a *complex operation* (CO) as 1 complex multiplication and 1 complex addition.

Computation of length N DFT then requires approximately N^2 CO's.

To derive an efficient algorithm for computation of the DFT, we employ a divide-and-conquer strategy.

Assume N is even.

$$X^{(N)}(k) = \sum_{n=0}^{N-1} x(n)e^{-j2\pi kn/N}$$

$$X^{(N)}(k) = \sum_{\substack{n=0 \\ n \text{ even}}}^{N-1} x(n)e^{-j2\pi kn/N}$$

$$+ \sum_{\substack{n=0 \\ n \text{ odd}}}^{N-1} x(n)e^{-j2\pi kn/N}$$

$$= \sum_{m=0}^{N/2-1} x(2m)e^{-j2\pi k(2m)/N}$$

$$+ \sum_{m=0}^{N/2-1} x(2m+1)e^{-j2\pi k(2m+1)/N}$$

$$X^{(N)}(k) = \sum_{m=0}^{N/2-1} x(2m)e^{-j2\pi km/(N/2)}$$

$$+ e^{-j2\pi k/N} \sum_{m=0}^{N/2-1} x(2m+1)e^{-j2\pi km/(N/2)}$$

$\boxed{m}$

Let $x_0(n) = x(2m), \qquad m = 0,...,N/2-1$

$x_1(n) = x(2m+1), \qquad m = 0,...,N/2-1$

Now have

$$X^{(N)}(k) = X_0^{(N/2)}(k) + e^{-j2\pi k/N}\, X_1^{(N/2)}(k)\ ,$$

$$k = 0, \ldots, N-1$$

Note that $X_0^{(N/2)}(k)$ and $X_1^{(N/2)}(k)$ are both periodic with period $N/2$, while $e^{-j2\pi k/N}$ is periodic with period N.

To summarize

We have an N-point signal, where N is even

$$x^{(N)}[n],\ 0 \le n \le N-1. \tag{1}$$

We want to compute the N-point DFT

$$x^{(N)}[n] \overset{\text{DFT}}{\leftrightarrow} X^{(N)}[k],\ 0 \le k \le N-1. \tag{2}$$

We break the signal into two signals, each having $N/2$ points

$$\begin{aligned} x_0^{(N/2)}[n] &= x^{(N)}[2n],\ 0 \le n \le N/2-1 \\ x_1^{(N/2)}[n] &= x^{(N)}[2n+1],\ 0 \le n \le N/2-1 \end{aligned}. \tag{3}$$

We compute the $N/2$-point DFT of each of these signals

$$\begin{aligned} x_0^{N/2}[n] &\overset{\text{DFT}}{\leftrightarrow} X_0^{(N/2)}[k],\ 0 \le k \le N/2-1 \\ x_1^{N/2}[n] &\overset{\text{DFT}}{\leftrightarrow} X_1^{(N/2)}[k],\ 0 \le k \le N/2-1 \end{aligned}. \tag{4}$$

We then combine these $N/2$-point DFTs with a twiddle factor to form the desired N-point DFT

$$X^{(N)}[k] = X_0^{(N/2)}[k] + e^{-j2\pi k/N} X_1^{(N/2)}[k],\ 0 \le k \le N-1. \tag{5}$$

Note that all signals and DFTs are viewed as being periodic with period indicated by the superscript. Thus, in particular, the two $N/2$-point DFTs $X_0^{(N/2)}[k]$ and $X_1^{(N/2)}[k]$ are periodic with period $N/2$. But the twiddle factor $e^{-j2\pi k/N}$ has period N. So the N-point DFT $X^{(N)}[k]$ has period N.

Computation

1. Direct

$$X^{(N)}(k)\ , \quad k = 0,\ldots,N-1 \qquad N^2 \text{ CO's}$$

2. Decimation by factor of 2

$$X_0^{(N/2)}(k)\ , \quad k = 0,\ldots,N/2-1 \qquad N^2/4 \text{ CO's}$$
$$X_1^{(N/2)}(k)\ , \quad k = 0,\ldots,N/2-1 \qquad N^2/4 \text{ CO's}$$

$$X^{(N)}(k) = X_0^{(N/2)}(k) + e^{-j2\pi k/N}\, X_1^{(N/2)}(k)\ ,$$

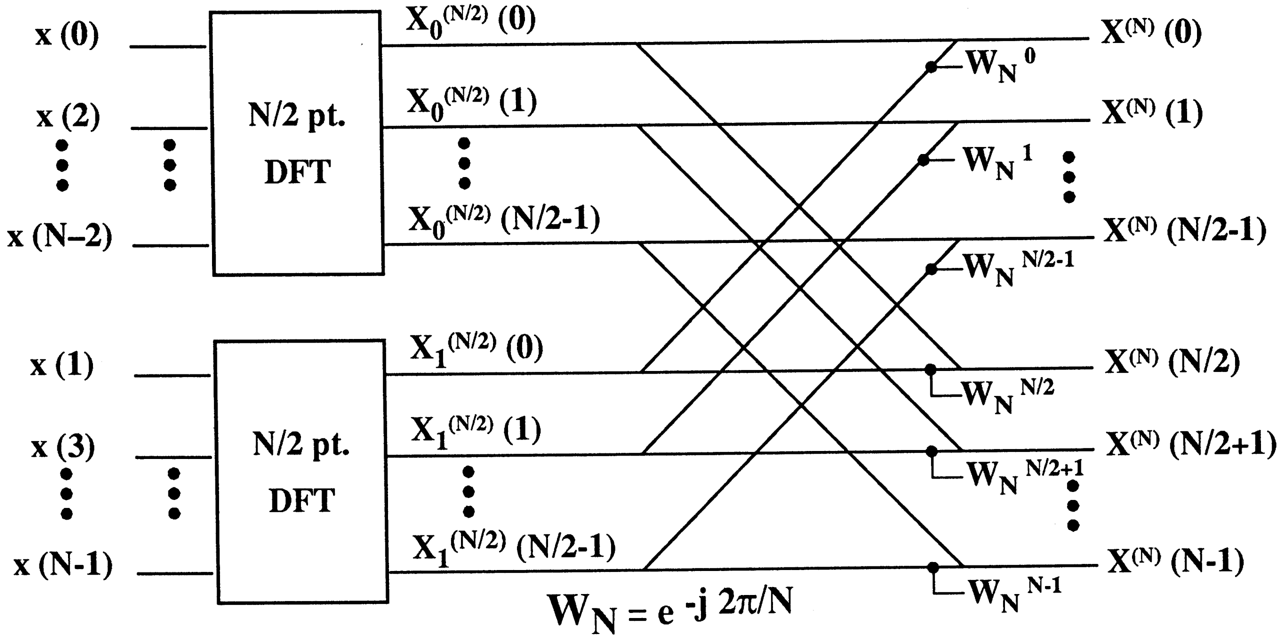
$$k = 0, \ldots, N-1$$

$$N\ \ CO's$$

Total $\qquad N^2/2 + N\ \ CO's$

For large N, we have nearly halved computation.

Consider a signal flow diagram of what we have done so far:



$$x_0^{(N/2)}[n] = x^{(N)}[2n],\ 0 \le n \le N/2-1$$

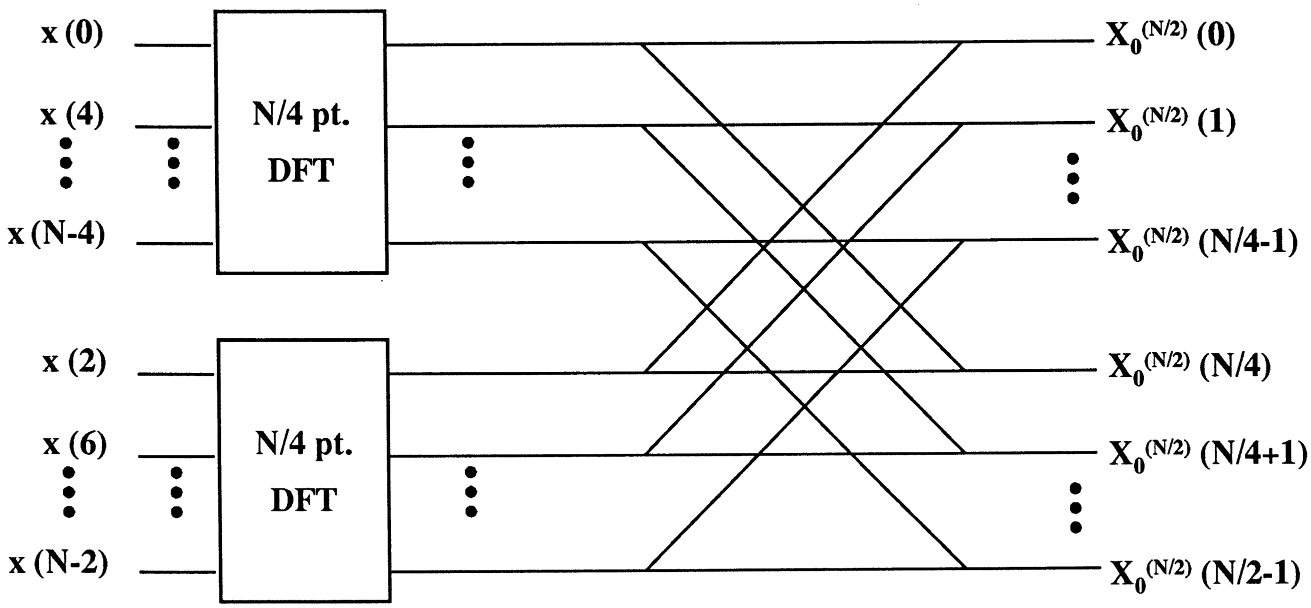
$$x_1^{(N/2)}[n] = x^{(N)}[2n+1],\ 0 \le n \le N/2-1$$

$$x_0^{N/2}[n] \overset{\text{DFT}}{\leftrightarrow} X_0^{(N/2)}[k],\ 0 \le k \le N/2-1$$

$$x_1^{N/2}[n] \overset{\text{DFT}}{\leftrightarrow} X_1^{(N/2)}[k],\ 0 \le k \le N/2-1$$

$$X^{(N)}[k] = X_0^{(N/2)}[k] + e^{-j2\pi k/N} X_1^{(N/2)}[k],\ 0 \le k \le N-1$$

If N is even, we can repeat the idea with each N/2 pt. DFT:



If $N = 2^M$, we repeat the process M times resulting in M stages.

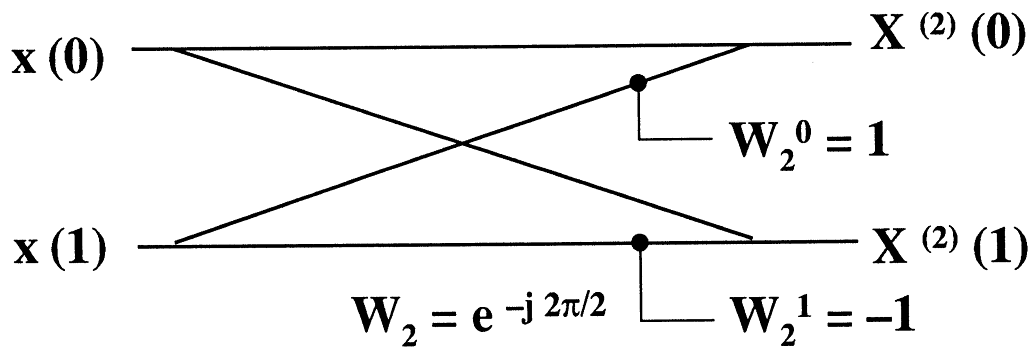
The first stage consists of 2 pt. DFT's

$$X^{(2)}(k) = \sum_{n=0}^{1} x(n)\, e^{-j2\pi kn/2}$$

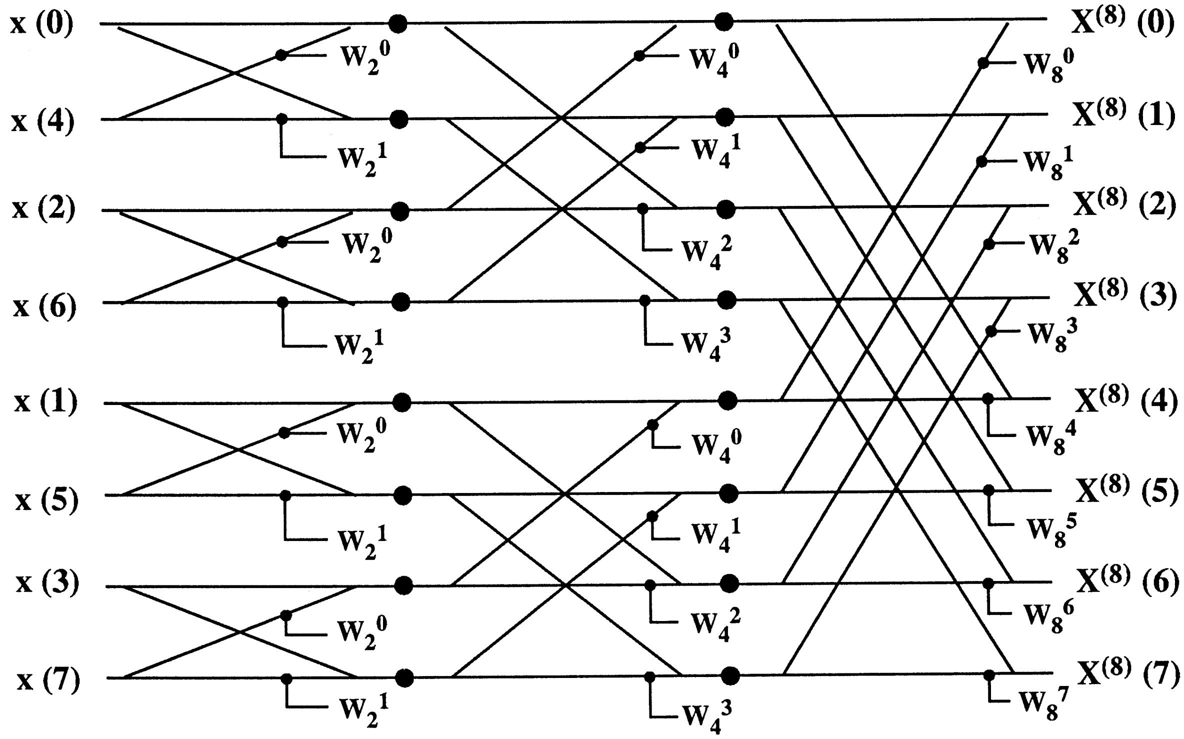
$$X^{(2)}(0) = x(0) + x(1)$$

$$X^{(2)}(1) = x(0) - x(1)$$

Flow diagram of 2 pt. DFT



Full Example for N = 8 (M = 3)



Ordering of Input Data

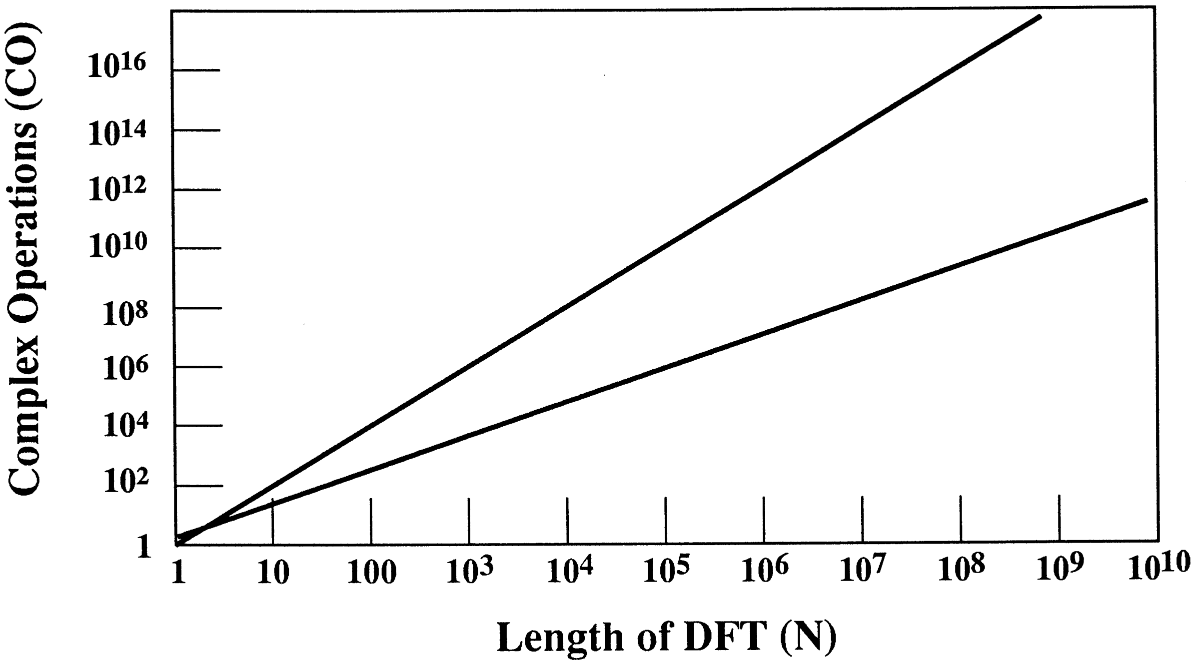
Normal Order		Bit Reversed Order	
Decimal	Binary	Binary	Decimal
0	0 0 0	0 0 0	0
1	0 0 1	1 0 0	4
2	0 1 0	0 1 0	2
3	0 1 1	1 1 0	6
4	1 0 0	0 0 1	1
5	1 0 1	1 0 1	5
6	1 1 0	0 1 1	3
7	1 1 1	1 1 1	7

Computation $(\mathbf{N = 2^M})$

$M = \log_2 N$ stages

N CO's/stage

Total: $N \log_2 N$ CO's


Complex Operations (CO)
10^{16}
10^{14}
10^{12}
10^{10}
10^{8}
10^{6}
10^{4}
10^{2}
1
1
10
100
10^{3}
10^{4}
10^{5}
10^{6}
10^{7}
10^{8}
10^{9}
10^{10}
Length of DFT (N)

Comments

- The algorithm we derived is the decimation-in-time radix 2 FFT.
  - input in bit-reversed order
  - in-place computation
  - output in normal order
- The dual of it is the decimation-in-frequency radix 2 FFT.
  - input in normal order
  - in-place computation
  - output in normal order

- The same approaches may be used to derive either decimation-in-time or decimation-in-frequency mixed radix FFT algorithms for any N which is a composite number.

- Another class of FFT algorithms is based on fast techniques for performing small convolutions (Winograd).

- All FFT algorithms are based on composite N and require O(N log N) computation.

- The DFT of a length N real signal can be expressed in terms of a length N/2 DFT.