EMPossible

Computational Science:
Computational Methods in Engineering

The Fast Fourier Transform

Fast Fourier Transform (FFT)

For a sequence of 7 points, the DFT is

$$F(0) = f(0) + f(1) + f(2) + f(3) + f(4) + f(5) + f(6) + f(7)$$
$$F(1) = f(0) + f(1)W_7 + f(2)W_7^2 + f(3)W_7^3 + f(4)W_7^4 + f(5)W_7^5 + f(6)W_7^6 + f(7)W_7^7$$
$$F(2) = f(0) + f(1)W_7^2 + f(2)W_7^4 + f(3)W_7^6 + f(4)W_7^8 + f(5)W_7^{10} + f(6)W_7^{12} + f(7)W_7^{14}$$
$$F(3) = f(0) + f(1)W_7^3 + f(2)W_7^6 + f(3)W_7^9 + f(4)W_7^{12} + f(5)W_7^{15} + f(6)W_7^{18} + f(7)W_7^{21}$$
$$F(4) = f(0) + f(1)W_7^4 + f(2)W_7^8 + f(3)W_7^{12} + f(4)W_7^{16} + f(5)W_7^{20} + f(6)W_7^{24} + f(7)W_7^{28}$$
$$F(5) = f(0) + f(1)W_7^5 + f(2)W_7^{10} + f(3)W_7^{15} + f(4)W_7^{20} + f(5)W_7^{25} + f(6)W_7^{30} + f(7)W_7^{35}$$
$$F(6) = f(0) + f(1)W_7^6 + f(2)W_7^{12} + f(3)W_7^{18} + f(4)W_7^{24} + f(5)W_7^{30} + f(6)W_7^{36} + f(7)W_7^{42}$$

Notice that there are MANY redundant calculations above.

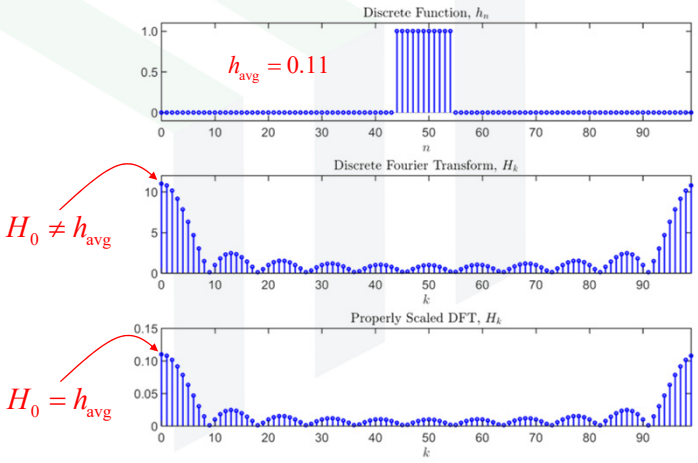
Perhaps there is a more efficient way of calculating a DFT that avoids redundant calculations. This is an FFT!

A *fast Fourier transform* (FFT) is just an efficient calculation of a DFT.

Scaling of the FFT

Most FFT algorithms scale the data so that the result must be divided by the number of points N to calculate correct DFT data.

$$H(k) = \frac{1}{N}\text{FFT}\{h(n)\}$$

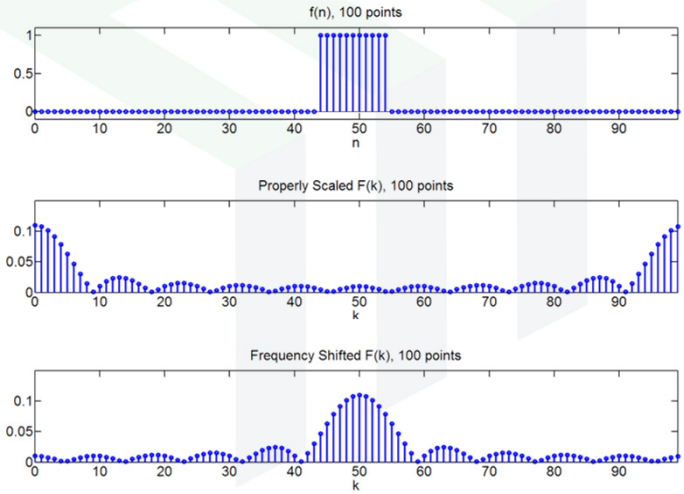


EMPossible

Slide 3

Frequency Shifting of the FFT

Also, most FFT algorithms output a shifted spectrum. Use `fftshift()` to correct this.



EMPossible

Slide 4