The Quick Fourier Transform: An FFT Based on Symmetries

Haitao Guo, Gary A. Sitton, C. Sydney Burns

Shruti Kolachana: 2020102053

Arshini Govindu: 2020102009

Abstract

- Creating an algorithm that involves the least number of operations in calculating DFT.
- DFT with only a subset of input or output points and reduces operations by half when real data is given.

Applications

- QFT is one of the most important existing algorithms and it is used for performing large calculations efficiently.
- Defines DCT and DST which are used in image compression.
- For real data, the operations required are significantly lesser.

Motivation of Work

- Efficient QFT for calculating DFT by removing unnecessary calculations.
- Removes the repeated calculations in basic QFT.
- No need to calculate DFT for the whole input data as we have to in a normal FFT.

Method

- The input signal is decomposed into its real and imaginary parts and further into the odd and even parts.
- The sum over an integral number of periods of an odd function is zero, and the sum of an even function over half of the period is one half the sum over the whole period.
- Symmetric relations of the sine and cosine are used.

$$\cos\left(\frac{2\pi(N-n)k}{N}\right) = \cos\left(\frac{2\pi nk}{N}\right)$$

$$\sin\left(\frac{2\pi(N-n)k}{N}\right) = -\sin\left(\frac{2\pi nk}{N}\right)$$

- We get these important equations from the DFT and the above relations. This method is for the 2^M QFT.

$$DCT(k, N+1, x) = \sum_{n=0}^{N} x(n)cos(\frac{\pi nk}{N}) \qquad k = 0, \dots, N$$

$$DST(k, N-1, x) = \sum_{n=1}^{N-1} x(n)sin(\frac{\pi nk}{N}) \qquad k = 1, \dots, N-1$$

- The number of operations reduces significantly.
- After calculating the number of multiplications and additions for real data, we observe that the number is double for complex data.
- We use recursive decomposition of DCT and DST to derive the DFT.
- In a mixed radix QFT, a recursive radix-2 FFT with a prime length QFT is used ending with a DCT/DST of half-length $N = p/2$.
- We cannot efficiently and directly calculate radix-q FFT for $n > 2$ as even-odd symmetries come in pairs.

Any relevant discussion

The discrete Fourier Transform is of O(N^2) complexity while the QFT is executed with O($NlogN$).

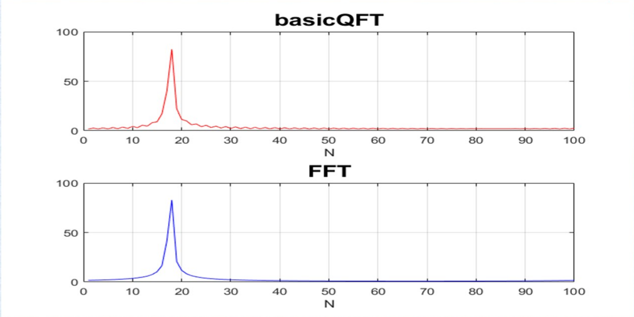
$$F(k) = \sum_{n=0}^{N-1} x(n)e^{\frac{-j2\pi nk}{N}}$$

These two equations calculate the output values two at a time thus reducing the number of operations

$$C(k) = 2\sum_{n=0}^{N/2-1} \begin{matrix}[u_e(n)cos\theta_{nk} + v_o(n)sin\theta_{nk}] \\ +j[v_e(n)cos\theta_{nk} - u_o(n)sin\theta_{nk}]\end{matrix}$$

$$C(N-k) = 2\sum_{n=0}^{N/2-1} \begin{matrix}[u_e(n)cos\theta_{nk} - v_o(n)sin\theta_{nk}] \\ +j[v_e(n)cos\theta_{nk} + u_o(n)sin\theta_{nk}]\end{matrix}$$

Results/ MATLAB plots



Algorithm	Real multi	Real add
FFT	$4N^2$	$4N^2$
Basic QFT	N^2	$N^2 + 4N$

Conclusion

- The basic QFT algorithm is more efficient than the direct methods.
- The length-2^M QFT is a lot faster than the radix-2 FFT as we had hoped to prove. It is much more suitable for real data as it involves nearly half the operations.
- Errors mostly occur due to inverse trigonometry multiplications

References

[1] A. V. Oppenheim and R. W. Schafer, Discrete-Time Signal Processing. Englewood Cliffs, NJ: Prentice-Hall, 1989.
[2] H. Guo, G. A. Sitton, and C. S. Burrus, "The quick discrete Fourier transform," in Proc. IEEE Int. Conf. Acoust., Speech, Signal Process., Adelaide, Australia, Apr. 19–22, 1994, pp. III:445–448.