1. Convergence of Fourier Series

When we say “let $X_n$ be a Fourier series for $f(x)$ on $[a,b]$ satisfying the boundary conditions” we mean one of the following:

- $[a,b] = [0,L]$, $f(0) = f(L) = 0$, $X_n(x) = \sin \frac{\pi nx}{L}$, $n = 1,2,\ldots$
- $[a,b] = [0,L]$, $f'(0) = f'(L) = 0$, $X_n(x) = \cos \frac{\pi nx}{L}$, $n = 0,1,2,\ldots$
- $[a,b] = [0,L]$, $f(0) = f'(L) = 0$, $X_n(x) = \sin \frac{\pi (2n+1)x}{2L}$, $n = 0,1,2,\ldots$
- $[a,b] = [0,L]$, $f'(0) = f'(L) = 0$, $X_n(x) = \cos \frac{\pi (2n+1)x}{2L}$, $n = 0,1,2,\ldots$

The last one is the classic Fourier series of sines and cosines. Results regarding the convergence of Fourier series are the following:

**Theorem 1.** The Fourier series $\sum c_n X_n(x)$ converges to $f(x)$ uniformly on $[a,b]$ provided that

1. $f$, $f'$, and $f''$ exist and are continuous on $[a,b]$;
2. $f(x)$ satisfies the boundary conditions.

**Theorem 2.** The Fourier series $\sum c_n X_n(x)$ converges to $f(x)$ on $[a,b]$ in $L_2$ provided that

$$\int_a^b |f(x)|^2 dx < \infty.$$ 

**Theorem 3.** Let $f$ and $f'$ be piecewise continuous on $[a,b]$. Then the Fourier series converges pointwise to

$$\frac{1}{2}(f(x + 0) + f(x - 0)).$$
**Example** Consider the “sawtooth function”: \( f(x) = x, \ x \in [-\pi, \pi] \). Since \( f(x) \) is odd, the Fourier Series is the sine series.

\[
b_n = \frac{1}{\pi} \int_{-\pi}^{\pi} x \sin(nx) \, dx = (-1)^{n+1} \frac{2}{n}.
\]

The coefficients decay as \( O\left(\frac{1}{n}\right) \). The series converges pointwise to \( f(x) \) on \((-\pi, \pi)\). However, \( \sup_{x \in (-\pi, \pi)} |f(x) - S_N(x)| \) does not decay as \( N \to \infty \). This exemplifies the Gibbs phenomenon.

**Example** Consider the “half-wave rectifier function”: \( f(x) = \sin x, \ x \in [-\pi, 0], \ f(x) = 0, \ x \in (0, \pi] \). This function is continuous with a piecewise continuous derivative. Its Fourier coefficients are

\[
a_0 = \frac{1}{\pi}, \ a_{2n} = -\frac{2}{\pi(4n^2 - 1)}, \ n > 0, \ a_{2n+1} = 0, \ n \geq 0,
\]

\[
b_1 = \frac{1}{2}, \ b_n = 0 \ n > 1.
\]

The coefficients decay as \( O(n^{-2}) \) for large \( n \). The series converges pointwise to \( f(x) \) on \([-\pi, \pi]\).
Example Consider the function called “symmetric, imbricated Lorentzian”:

\[ f(x) = \frac{1 - p^2}{1 + p^2 - 2p \cos x}, \quad 0 < p < 1. \]

this function is infinitely differentiable. Its Fourier series is given by

\[ f(x) = 1 + 2 \sum_{n=1}^{\infty} p^n \cos(nx). \]

One can check it as follows.

\[
\begin{align*}
  f(x) &= 1 + 2 \sum_{n=1}^{\infty} p^n \cos(nx) \\
  &= 1 + 2 \sum_{n=1}^{\infty} p^n \left(e^{inx} + e^{-inx}\right) \\
  &= 1 - p e^{ix} + 1 - p^{-ix} - 1 + p^2 + 2p \cos x \\
  &= 1 - p^2 + 1 - p^2 + 2p \cos x \\
  &= 1 + p^2 - 2p \cos x.
\end{align*}
\]

The Fourier coefficients decay faster than any power of \( n \). In this case we say that the coefficients decay exponentially or spectrally.
**Definition 1.** The algebraic index of convergence \( k \) is the largest number for which
\[
\lim_{n \to \infty} |c_n| n^k < \infty, \ n \gg 1,
\]
where \( c_n \) are the coefficients of the series.

Alternative definition: if
\[
c_n \sim O \left( \frac{1}{n^k} \right), \ n \gg 1,
\]
then \( k \) is the algebraic index of convergence.

The first form gives an unambiguous definition for the case where \( c_n \sim O \left( \frac{\log n}{n} \right) \).

**Definition 2.** If the algebraic index of convergence is unbounded than the series convergence exponentially or spectrally.

**Definition 3.** The exponential index of convergence \( r \) is given by
\[
r = \limsup_{n \to \infty} \frac{\log \| \log |c_n| \|}{\log n}.
\]
Equivalently, if
\[
c_n \sim O(s \exp[-qn^r]), \ n \gg 1,
\]
then the exponential index of convergence is the exponent \( r \).

**Example** For the symmetric, imbricated Lorentzian, \( c_n = 2p^n = 2e^{-n(-\log p)} \), (note, \( \log p < 0 \) as \( 0 < p < 1 \)), hence \( r = 1, \ q = -\log p, \ s = 2 \).

The coefficients of the complex-exponential form of a Fourier series are:
\[
c_n = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(x) e^{-inx} dx.
\]

Note that if \( f(x) \) is real-valued, then
(1) \[
f(x) = \sum_{-\infty}^{\infty} c_n e^{inx} = \overline{\overline{f(x)}} = \sum_{-\infty}^{\infty} \overline{c_n} e^{inx} = \sum_{-\infty}^{\infty} \overline{c_n} e^{-inx}.
\]

Therefore, for a real-valued \( f(x) \),
\[
c_n = \overline{c_{-n}}.
\]

If we integrate Eq. (1) by parts, repeatedly integrating the exponent and differentiating \( f \), then after \( J + 1 \) steps we obtain
\[
c_n = \frac{1}{2\pi} \sum_{j=0}^{J} (-1)^{j+1} \left( \frac{i}{n} \right)^{j+1} \left\{ f^{(j)}(\pi) - f^{(j)}(-\pi) \right\}
+ \frac{1}{2\pi} \left( \frac{i}{n} \right)^{J+1} \int_{-\pi}^{\pi} f^{(J+1)}(x) e^{-inx} dx.
\]
Exercise Show that for the classic Fourier series $a_n = 2 \text{Re}(c_n)$ and $b_n = -2 \text{Im}(c_n)$ and
\[
a_n \sim \frac{1}{\pi} \sum_{j=0}^{J} (-1)^{n+j} \left( \frac{f(2j+1)(\pi) - f(2j+1)(-\pi)}{n^{2j+2}} \right) + O(n^{-(2J+4)}), \ n \to \infty, \ J \text{ fixed},
\]
\[
b_n \sim \frac{1}{\pi} \sum_{j=0}^{J} (-1)^{n+j+1} \left( \frac{f(2j)(\pi) - f(2j)(-\pi)}{n^{2j+1}} \right) + O(n^{-(2J+3)}), \ n \to \infty, \ J \text{ fixed}.
\]
Therefore the following theorem takes place.

Theorem 4. If
\begin{align*}
(1) & \quad f(-\pi) = f(\pi), \ f'(\pi) = f'(-\pi), \ldots, \ f^{(k-2)}(-\pi) = f^{(k-2)}(\pi), \\
(2) & \quad f^{(k)} \text{ is integrable},
\end{align*}
then
\[
|a_n| \leq \frac{F}{n^k}, \quad |b_n| \leq \frac{F}{n^k},
\]
where $F$ is a constant that does not depend on $n$.

The next is the Fourier truncation error theorem.

Theorem 5. The error of approximating $f(x)$ by the sum of the first $N$ terms of its Fourier series is bounded by the sum of the absolute values of all neglected coefficients.

Example The “sawtooth function” that is obtained from $f(x) = x$, $x \in (-\pi, \pi)$ by the periodic extension to the real line is not continuous (Fig. 3). Its integral is continuous. Its derivative can be expressed in terms of the Dirac delta-function as follows
\[
f'(x) = 1 - 2\pi \sum_{n=-\infty}^{\infty} \delta(x - \pi(2n + 1)).
\]
You can check it by verifying the identity
\[
f(x) = \int_{0}^{x} f'(x')dx'.
\]
Hence $f'(x)$ is integrable on $[-\pi, \pi]$. Therefore, we need to take $k = 1$ to apply Theorem 4 and get the bound $|b_n| \leq \frac{F}{n}$. This bound is correct as we have seen in Example 1.

2. Pseudospectral errors

In this Section, we estimate the errors of approximating a function by a truncated Fourier interpolation series.

Theorem 6. Let the interpolation (or collocation) points be defined by
\[
x_k = -\pi + \frac{2\pi k}{N}, \ k = 1, 2, \ldots, N.
\]
Let a function $f(x)$ have the exact, infinite Fourier series representation

$$f(x) = \frac{a_0}{2} + \sum_{n=1}^{\infty} \alpha_n \cos(nx) + \sum_{n=1}^{\infty} \beta_n \sin(nx).$$

Let the trigonometric polynomial which interpolates to $f(x)$ at the $N$ collocation points be

$$S_N = \frac{a_0}{2} + \sum_{n=1}^{N/2-1} a_n \cos(nx) + \sum_{n=1}^{N/2-1} b_n \sin(nx) + \frac{1}{2} a_{N/2} \cos((N/2)x),$$

where

$$S_N(x_k) = f(x_k), \quad k = 1, 2, \ldots, N.$$

Then the coefficients of interpolant can be computed without error by the trapezoidal rule

$$a_n = \frac{2}{N} \sum_{k=1}^{N} f(x_k) \cos(nx_k),$$

$$b_n = \frac{2}{N} \sum_{k=1}^{N} f(x_k) \sin(nx_k).$$

and these coefficients of the interpolant are given by infinite series of the exact Fourier coefficients:

$$a_n = \alpha_n + \sum_{j=1}^{\infty} (\alpha_{n+jN} + \alpha_{-n+jN}), \quad n = 0, 1, 2, \ldots, \frac{N}{2},$$

$$b_n = \beta_n + \sum_{j=1}^{\infty} (\beta_{n+jN} - \beta_{-n+jN}), \quad n = 1, 2, \ldots, \frac{N}{2} - 1.$$
Furthermore, if \( f_N \) is the sum of the first \( N \) terms of the exact Fourier series, we have:

\[
|f(x) - f_N(x)| \leq \left\{ \beta_{N/2} + \sum_{n=1+N/2}^{\infty} (|\alpha_n| + |\beta_n|) \right\},
\]

while for the trigonometric interpolation the bound is

\[
|f(x) - S_N(x)| \leq 2 \left\{ \beta_{N/2} + \sum_{n=1+N/2}^{\infty} (|\alpha_n| + |\beta_n|) \right\}.
\]

That is to say that the error is bounded by twice the sum of the absolute values of all the neglected coefficients.

### 3. Discrete Fourier Transform in MATLAB

In MATLAB 2015b, the discrete Fourier transform is defined as follows. Let \( x \) be a vector with \( N \) entries. The functions \texttt{fft} and \texttt{ifft} implement the discrete Fourier transform and the inverse discrete Fourier transform. If \( y = \texttt{fft}(x) \) and \( x = \texttt{ifft}(y) \) then

(2) \quad y(k) = \sum_{j=1}^{N} x(j) \omega_N^{(j-1)(k-1)}

and

(3) \quad x(j) = \frac{1}{N} \sum_{k=1}^{N} y(k) \omega_N^{-(j-1)(k-1)},

where

(4) \quad \omega_N = e^{-\frac{2\pi i}{N}}.

**Exercise** Check that the functions \texttt{fft} and \texttt{ifft} defined by Eqs. (2), (3) and (4) are mutually inverse.

Now we discuss how to use the functions \texttt{fft} and \texttt{ifft} for solving PDEs.

**Example** Consider a linear dispersive PDE \( u_t + u_{xxx} = 0 \) on the interval \(-\pi \leq x \leq \pi\), \( t \geq 0 \), with periodic boundary conditions and the initial condition \( u(x,0) = u_0(x) \). The periodic boundary conditions suggest the use of trigonometric Fourier series. First we solve it using the Fourier series. Write

\[
u(x,t) = \sum_{k=-\infty}^{\infty} c_k(t) e^{ikx}, \quad \text{where} \quad c_k = \frac{1}{2\pi} \int_{-\pi}^{\pi} u(x,t) e^{-ikx} dx.
\]

Then

\[
u_t = \sum_{k=-\infty}^{\infty} c'_k(t) e^{ikx}, \quad u_{xxx} = \sum_{k=-\infty}^{\infty} c_k(t)(ik)^3 e^{ikx}.
\]
Therefore, all harmonics evolve in time independently. For \( k \)th harmonic we have
\[
c'_k(t) = -(ik)^3c_k(t) = ik^3c_k(t).
\]
Hence \( c_k(t) = c_k(0)e^{ik^3t} \).

Finally,
\[
u(x, t) = \sum_{k=-\infty}^{\infty} c_k(0)e^{ik^3t}e^{ikx}, \quad \text{where} \quad c_k(0) = \frac{1}{2\pi} \int_{-\pi}^{\pi} u_0(x)e^{-ikx}dx.
\]

Then the exact solution is given by
\[
u_0(x) = \cos(x)\sin(25x).
\]

Exercise  Solve the IVP above using the Fourier transform in the Fourier space. Find phase and group velocities. Verify that the exact solution is given by Eq.(6).

The script `linear_dispersion.m` computes the solution using the discrete Fourier transform and superimposes it with the exact solution (see Fig. 4).

```matlab
% Solves u_t + u_{xxx} = 0 using exact time integration and discrete % Fourier Transform
N = 1024; % the number of collocation points
nt = 10000; % the number of time steps
tmax = 0.1; % the time till that the solution is computed
t = linspace(0,tmax,nt);
x = linspace(-pi,pi,N);
u0 = cos(x).*sin(25*x); % the initial condition
f0 = fft(u0); % Fourier coefficients at time 0
k = [0:N/2-1,-N/2:-1]; % frequencies

fig = figure;
grid;
hold on;
he = plot(x,u0,'Linewidth',1,'color','r');
h = plot(x,u0,'Linewidth',2,'color','b');
axis([-pi,pi,-1,1]);
drawnow;

for j = 1 : nt
    ft = f0.*exp(1i*k.*3*t(j)); % Fourier coefficients at time t(j)
    ut = ifft(ft); % solution at time t(j)
    set(h,'xdata',x,'ydata',real(ut));
    set(he,'xdata',x,'ydata',cos(x+1876*t(j)).*sin(25*x+15700*t(j)));
end
```
axis([-pi,pi,-1,1]);
drawnow;
pause(0.1)
end

![Figure 4.](image)

**Figure 4.** This blue curve: the numerical solution at time $t = 0.1$ of $u_t + u_{xxx} = 0$ with the initial condition (5) obtained using the discrete Fourier transform and exact time integration. The thin red curve: the exact solution at $t = 0.1$ given by Eq. (6).

If we are solving a PDE with periodic boundary conditions on the interval $[0, L]$ then it is very important to scale the frequencies properly. Suppose we are using $N$-point DFT. Then $x_j = Lj/N$, $j = 0, 1, \ldots, N - 1$.

$$\text{[fft}(u)\text{]}_k = \sum_{j=0}^{N-1} f(x_j)e^{-\frac{i2\pi jk}{N}} = \sum_{j=0}^{N-1} f(x_j)e^{-\frac{i2\pi x_j k}{L}}.$$ 

Therefore, the $k$-th frequency $f_k = \frac{2\pi k}{L}$, and the vector of frequencies should be set to

$$\text{freq} = \left[\frac{2\pi}{L}[0, 1, \ldots, N/2 - 1, -N/2, \ldots, -1]\right].$$

4. **Aliasing and Nyquist sampling theorem**

Have you notices that sometimes in the movies it seems like a car’s wheels rotate in the wrong direction? The reason is aliasing. The movie shooting speed is 24 frames per second. If a wheel rotates by an angle $\pi < \alpha < 2\pi$ between the frames your brain will process it as if it rotates in the wrong direction with the angular velocity $\frac{(\alpha - 2\pi)}{24}$.

A similar error due to the discrete sampling occurs when we restore a function from its Fourier transform. Suppose a function $f(x)$, $x \in [-\pi, \pi]$ can be written as an infinite series

$$f(x) = \sum_{k=-\infty}^{\infty} \alpha_k e^{ikx}.$$
The term $\alpha_m e^{imx}$ is the $m$-th Fourier component of $f$. Now we consider $f_m(x) \equiv \alpha_m e^{imx}$ on the interval $[0, 2\pi]$. Let us sample it at the points $x_j := \frac{2\pi j}{N}$ and apply the $N$-point discrete Fourier transform to it. Then for the $\text{fft}(f_m)$ we have

$$[\text{fft}(f)]_k = \sum_{j=0}^{N-1} \alpha_m e^{i2\pi jm/N} e^{-i2\pi jk/N} = \sum_{j=0}^{N-1} \alpha_m e^{i2\pi j(m-k)/N}.$$

Let $m = Nq + r$, where $r \in \{0, 1, \ldots, N - 1\}$. Then

$$[\text{fft}(f)]_k = \sum_{j=0}^{N-1} \alpha_m e^{i2\pi j(m-k)/N} = \sum_{j=0}^{N-1} \alpha_m e^{i2\pi j(Nq+r-k)} = \sum_{j=0}^{N-1} \alpha_m e^{i2\pi j(r-k)} = \alpha_m \delta_{rk}.$$

Therefore, these frequencies will be aliased to lower frequencies that correspond to the wave numbers equal to the residual after division by $N$.

Another important moment to have in mind is that the largest frequency that can be resolved without aliasing by the Fourier transform is $N/2$, not $N$. The reason is that the frequencies $N/2 < k < N$ are aliased to the negative frequencies between $-N/2$ and 0.

**Example** Suppose a 25-point discrete Fourier transform is applied to the function $f(x) = \sin(32x)$ (the blue curve). Then the inverse discrete Fourier transform is applied to the result. The wave numbers 32 and $-32$ are aliased to the wave numbers $k_1$ and $k_2$ in the range $-12 \leq k_1, k_2 \leq 12$. It is easy to check that $k_1 = 7$ and $k_2 = -7$. Equivalently, $k_2 = 18$, if we need to convert them to the range $0 \leq k_1, k_2 \leq 24$.

**Figure 5.** A 25-point discrete Fourier transform was applied to the function $f(x) = \sin(32x)$ (the blue curve). Then the result was restored using the inverse discrete Fourier transform. The resulting function is $g(x) = \sin(7x)$ (the red markers and the dashed curve).

**Example** Similarly, $\sin(16x)$ aliases to $-\sin(9x)$ if 25-point DFT is used.
Exercise Suppose you apply an 32-point Fourier transform to the function \( \sin(50x) \) and then restore it with the discrete inverse Fourier transform. What should be the result?

As we have seen, in order to avoid aliasing, a signal of frequency \( f_c \) should be sampled with the sampling frequency \( f_s \geq 2f_c \) in order to avoid aliasing. This fact is known as the Nyquist sampling theorem. The critical frequency \( 2f_c \) is called the Nyquist frequency.

A note on aliasing can be found in \[ \text{http://redwood.berkeley.edu/bruno/npb261/aliasing.pdf} \].

5. Fast Fourier Transform

Performing a discrete Fourier transform is equivalent to matrix multiplication:

\[
X = \Omega x, \quad \Omega_{kj} = e^{-\frac{2\pi}{N}kj}, \quad k, j = 0, 1, \ldots, N - 1.
\]

The computational cost of the multiplication of a complex \( N \times N \) matrix by a complex vector \( N \times 1 \) is

\[
\sim 8N^2 \text{ real floating point operations (or flops)}.
\]

This cost can be dramatically reduced if \( N \) is a power of 2. The Fourier transform performed using the trick explained below is called the Fast Fourier Transform. Its cost is

\[
\sim 5N \log_2 N \text{ flops}.
\]

Consider the discrete Fourier transform with \( N = 2M \). Then we can rewrite it as

\[
X(k) = \sum_{j=0}^{N-1} x(j)\omega_N^{jk} = \sum_{j=0}^{N/2-1} \left[ x(2j)\omega_N^{k(2j)} + x(2j + 1)\omega_N^{k(2j+1)} \right] = Y_k + \omega^k Z_k, \quad k = 0, 1, \ldots, N/2 - 1.
\]

Note that \( Y_k \) and \( Z_k, 0 \leq k \leq N/2 - 1 \), are the discrete Fourier transforms of the data sets \([x_0, x_2, \ldots, x_{N/2-2}]\) and \([x_1, x_3, \ldots, x_{N/2-1}]\). The only problem is that \( Y_k \) and \( Z_k \) are defined only for \( k = 0, 1, \ldots, N/2 - 1 \). To define then for \( N/2 \leq k < N \) we observe that

\[
\omega^{k+N/2} = e^{-i\frac{2\pi(k+N/2)}{N}} = e^{-i\frac{2\pi k}{N} - i\pi} = -e^{i\frac{2\pi k}{N}} = -\omega^k.
\]

Hence,

\[
X(k + N/2) = Y_k - \omega^k Z_k.
\]
Example Let \( N = 2^3 = 8 \). Let \( X = \text{fft}(x) \). Then we have

\[
\begin{bmatrix}
X_0 \\
X_1 \\
X_2 \\
X_3 \\
X_4 \\
X_5 \\
X_6 \\
X_7
\end{bmatrix}
= 
\begin{bmatrix}
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\
1 & \omega & \omega^2 & \omega^3 & \omega^4 & \omega^5 & \omega^6 & \omega^7 \\
1 & \omega^2 & \omega^4 & \omega^6 & \omega^8 & \omega^{10} & \omega^{12} & \omega^{14} \\
1 & \omega^3 & \omega^6 & \omega^9 & \omega^{12} & \omega^{15} & \omega^{18} & \omega^{21} \\
1 & \omega^4 & \omega^8 & \omega^{12} & \omega^{16} & \omega^{20} & \omega^{24} & \omega^{28} \\
1 & \omega^5 & \omega^{10} & \omega^{15} & \omega^{20} & \omega^{25} & \omega^{30} & \omega^{35} \\
1 & \omega^6 & \omega^{12} & \omega^{18} & \omega^{24} & \omega^{30} & \omega^{36} & \omega^{42} \\
1 & \omega^7 & \omega^{14} & \omega^{21} & \omega^{28} & \omega^{35} & \omega^{42} & \omega^{49}
\end{bmatrix}
= 
\begin{bmatrix}
x_0 \\
x_1 \\
x_2 \\
x_3 \\
x_4 \\
x_5 \\
x_6 \\
x_7
\end{bmatrix}
\equiv 
\begin{bmatrix}
Y_0 + Z_0 \\
Y_1 + \omega Z_1 \\
Y_2 + \omega^2 Z_2 \\
Y_3 + \omega^3 Z_3 \\
Y_0 - Z_0 \\
Y_1 - \omega Z_1 \\
Y_2 - \omega^2 Z_2 \\
Y_3 - \omega^3 Z_3
\end{bmatrix},
\]

where \( Y_k \)'s and \( Z_k \)'s are the Fourier transform of the even and odd components of \( x \), i.e.,

\[
Y = \text{fft}([x_0 \ x_2 \ x_4 \ x_6]), \quad Z = \text{fft}([x_1 \ x_3 \ x_5 \ x_7]).
\]

More exactly,

\[
Y_k = \sum_{j=0}^{3} x_{2j} \omega^{j(2k)}, \quad Z_k = \sum_{j=0}^{3} x_{2j+1} \omega^{j(2k)}, \quad k = 0, 1, 2, 3.
\]

To obtain the last equality in Eq. (7) we have used the following identities:

\[
\omega^{(4+l)} \equiv e^{-\frac{2\pi i(4+l)}{8}} = e^{-\frac{\pi i}{4}} e^{-\frac{2\pi i l}{8}} = -e^{-\frac{2\pi i l}{8}} \equiv -\omega^l, \quad l = 0, 1, 2, 3,
\]

\[
\omega^{2k(4+l)} \equiv e^{-\frac{2\pi i 2k(4+l)}{8}} = e^{-\frac{8(2k+1)\pi i}{8}} e^{-\frac{2\pi i l(2k)}{8}} = e^{-\frac{4\pi i l(2k)}{8}} \equiv \omega^{l(2k)}, \quad l = 0, 1, 2, 3.
\]

Therefore, instead of doing a single 8-point Fourier transform, we can perform 2 4-point Fourier transforms and combine the results of the 8-point Fourier transform out of them.

We proceed further recursively.

In turn,

\[
[Y_0, Y_1] = [P_0 + Q_0, P_1 + \omega^2 Q_1], \quad [Y_2, Y_3] = [P_0 - Q_0, P_1 - \omega^2 Q_1],
\]

\[
[Z_0, Z_1] = [S_0 + T_0, S_1 + \omega^2 T_1], \quad [Z_2, Z_3] = [S_0 - T_0, S_1 - \omega^2 T_1],
\]

where

\[
[P_0, P_1] = \text{fft}([x_0, x_4]), \quad [Q_0, Q_1] = \text{fft}([x_2, x_6]),
\]

\[
[S_0, S_1] = \text{fft}([x_1, x_5]), \quad [T_0, T_1] = \text{fft}([x_3, x_7]).
\]

For the two-point discrete Fourier transform we have:

\[
P_0 = x_0 e^{-i 2\pi 0/2} + x_4 e^{-i 2\pi 0/2} = x_0 + x_4, \quad P_1 = x_0 e^{-i 2\pi 0/2} + x_4 e^{-i 2\pi /2} = x_0 - x_4.
\]

Hence,

\[
P_0 = x_0 + x_4, \quad P_1 = x_0 - x_4, \quad Q_0 = x_2 + x_6, \quad Q_1 = x_2 - x_6,
\]

\[
S_0 = x_1 + x_5, \quad S_1 = x_1 - x_5, \quad T_0 = x_3 + x_7, \quad T_1 = x_3 - x_7.
\]
Now we compute the number of flops in the Fast Fourier Transform. Let $N \equiv 2^m$. Let $F(2^m)$ be the number of flops to be found. Complex addition and subtraction require 2 flops. Complex multiplication requires 6 flops. (Check this!). Therefore,

$$F(2^m) = 2F(2^{m-1}) + 2 \cdot 2^m + 6 \cdot 2^{m-1},$$

i.e., the cost $2^m$-point Fast Fourier Transform is equal to two costs of the Fast Fourier Transform with twice as few points plus the cost of $2^m$ additions and subtractions for $Y_k \pm \omega^k Z_k$ plus the cost of $2^{m-1}$ complex multiplications $\omega^k Z_k$. Continuing, we obtain

$$F(2^m) = 2F(2^{m-1}) + 2 \cdot 2^m + 6 \cdot 2^{m-1}$$
$$= 2(2F(2^{m-2}) + 2 \cdot 2^{m-1} + 6 \cdot 2^{m-2}) + 2 \cdot 2^m + 6 \cdot 2^{m-1}$$
$$= 2^2 F(2^{(m-2)}) + 2 \cdot (2 \cdot 2^m + 6 \cdot 2^{m-1}) \ldots$$
$$= 2^m F(0) + m(2 \cdot 2^m + 6 \cdot 2^{m-1}) = 2^m F(0) + m(2 + 3)2^m$$
$$= 5N \log_2 N.$$

Here we have taken into account that $F(0) = 0$ as the one-point DFT, i.e., \texttt{fft} of a number $a$ is \texttt{fft}(a) = a, which requires no floating point operations.

References
