Discrete Fourier Transforms and Approximate Solutions of PDEs
While we have discussed many solution techniques for partial differential equations that use fourier methods, this requires that we have ways of effectively obtaining the fourier coeffients and for reconstructing functions from their fourier representations. For many problems we can not readily determine analytically the fourier coefficients. This may arise since the functional forms are complicated, integrals are not easily analytically expressable, or because the input functions are only known empirically, such as tabulated data from experimental measurements. While we could in principle approximate numerically the integrals that appear in the transforms, in many situations it would be useful to further control these approximations so that when we perform multiple transformations these serve as exact inverses of each other.

For this purpose, we consider periodic functions $u(x)$ sampled as $u_m = u(x_m)$ at lattice locations $x_m = mL/n$ on the interval $[0, L]$. We define the following fourier transforms for this discrete data.
Discrete Fourier Transform (DFT). The function is transformed to a frequency space representation using

$$\hat{u}_k = \mathcal{F}_k[\{u_m\}] = \frac{1}{n}\sum_{m=0}^{n-1} u_m e^{-i2\pi km/n}.$$

Inverse Discrete Fourier Transform (IDFT). The function is reconstructed at the lattice sites x_m using the inverse transform (IDFT) given by

$$u_m = \mathcal{F}_m^{-1}[\{\hat{u}_k\}] = \sum_{k=0}^{n-1} \hat{u}_k e^{i2\pi km/n}.$$

We can also express this concisely by using vector notation $\hat{\mathbf{u}} = \{\hat{u}_k\}_{k=0}^{n-1}$ and $\mathbf{u} = \{u_m\}_{m=0}^{n-1}$. The DFT and IDFT are then seen to be linear transforms that are inverses of each other with

$$\hat{\mathbf{u}} = \mathcal{F}[\mathbf{u}], \quad \mathbf{u} = \mathcal{F}^{-1}[\hat{\mathbf{u}}].$$

Aliasing. There is an important relationship between the discrete fourier transforms and the continuous fourier transforms. Consider the continuous fourier transform for periodic functions on $[0, L]$,

$$u(x) = \sum_{k=-\infty}^{\infty} \hat{\mathbb{U}}_k e^{i2\pi kx/L}, \quad \hat{\mathbb{U}}_k = \frac{1}{L}\int_0^L u(x)e^{-i2\pi kx/L}dx.$$

When we restrict evaluations to the lattice points $x_m = mL/n$, we can not distinguish between the fourier modes $e^{i2\pi kx_m/L} = e^{i2\pi(k+\alpha n)x_m/L}$ for any $\alpha \in \mathbb{Z}$. As a consequence we have

$$\begin{aligned} u_m &= \sum_{k=0}^{n-1} \hat{u}_k e^{i2\pi km/n} \\ u(x_m) &= \sum_{k=-\infty}^{\infty} \hat{\mathbb{U}}_k e^{i2\pi km/n} = \sum_{k=0}^{n-1}\sum_{\alpha=-\infty}^{\infty} \hat{\mathbb{U}}_{k+\alpha n} e^{i2\pi km/n}. \end{aligned}$$

Since $u_m = u(x_m)$ we have

$$\hat{u}_k = \sum_{\alpha=-\infty}^{\infty} \hat{\mathbb{U}}_{k+\alpha n}.$$

This shows that the discrete fourier transform coefficients $\hat{u}_k$ are exactly the sum of the continuous fourier transform coefficients $\{\hat{\mathbb{U}}_{k'}\}_{k' \in \mathcal{K}(k)}$, where $k' = k + \alpha n$ for some $\alpha \in \mathbb{Z}$. This gives a set of n equivalence classes for the fourier modes with respect to the discrete cases, $k = 0, 1, \ldots, n-1$. This equivalence is referred to as "aliasing" of the continuous fourier modes when they are restricted to being evaluated only on the lattice x_m. This aliasing phenomena associated with discrete samplings has many consequences in signal processing, computer graphics (anti-aliasing), numerical analysis, and other applications.

Interpolation of Sampled Functions. For functions sampled on the lattice x_m, there are many possible ways to perform interpolation. We shall utilize the connection between the complex-exponential series representation and the real-valued sine-cosine series. The first interpolation one might consider is

$$\tilde{u}(x) = \sum_{k=0}^{n-1} \hat{u}_k e^{i2\pi kx/L}.$$

However, this does not provide a good interpolation since for some real-valued functions sampled at x_m we can obtain complex values when evaluated off the lattice ($x \neq x_m$ for any m). For example at location $x = \frac{1}{2}(x_1 + x_0)$. We would like an interpolation that remains real-valued between the lattice locations. This will require some more careful considerations.

We saw that for a real-valued function we need to have that the discrete fourier coefficients satisfy

$$\overline{\hat{u}_k} = \hat{u}_{n-k}.$$

This can be verified directly or also seen as following from the continuous coefficients since $\overline{\hat{\mathbb{U}}_k} = \hat{\mathbb{U}}_{-k}$ using the aliasing formula above. From the real-valued series expansion we have

$$\begin{aligned} u(x) &= \frac{1}{2}A_0 + \sum_{k=1}^{\infty} A_k \cos(2\pi kx/L) + B_k \sin(2\pi kx/L) \\ &= \frac{1}{2}A_0 + \sum_{k=1}^{\infty} \frac{1}{2}\left(A_k - iB_k\right)\exp(2\pi kx/L) + \frac{1}{2}\left(A_k + iB_k\right)\exp(-2\pi kx/L) \\ &= \sum_{k=-\infty}^{\infty} c_k \exp(2\pi kx/L). \end{aligned}$$

We used here Euler's Identity $e^{i\theta} = \cos(\theta) + i\sin(\theta)$ to express $\cos(\theta) = \frac{1}{2}\left(e^{i\theta} + e^{-i\theta}\right)$ and $\sin(\theta) = \frac{1}{2i}\left(e^{i\theta} - e^{-i\theta}\right)$.

We consider the case when n is odd and let $N = n - 1$. We can use the aliasing formula to obtain for evaluations at the lattice sites

$$\begin{aligned} u(x_m) &= \frac{1}{2}\sum_{\alpha=-\infty}^{\infty} A_{0+\alpha n} + \sum_{k=1}^{N/2}\sum_{\alpha=-\infty}^{\infty} A_{k+\alpha n}\cos(2\pi k x_m/L) + B_{k+\alpha n}\sin(2\pi k x_m/L) \\ &= \frac{1}{2}\sum_{\alpha=-\infty}^{\infty} A_{0+\alpha n} + \sum_{k=1}^{N/2}\sum_{\alpha=-\infty}^{\infty} \frac{1}{2}\left(A_{k+\alpha n} - iB_{k+\alpha n}\right)\exp(2\pi k x_m/L) \\ &\quad + \sum_{k=1}^{N/2}\sum_{\alpha=-\infty}^{\infty} \frac{1}{2}\left(A_{k+\alpha n} + iB_{k+\alpha n}\right)\exp(-2\pi k x_m/L). \end{aligned}$$

We can now extend this off-lattice which yields real-values since

$$\begin{aligned} u(x) &= \frac{1}{2}\sum_{\alpha=-\infty}^{\infty} A_{0+\alpha n} + \sum_{k=1}^{N/2}\sum_{\alpha=-\infty}^{\infty} A_{k+\alpha n}\cos(2\pi k x/L) + B_{k+\alpha n}\sin(2\pi k x/L) \\ &= \frac{1}{2}\sum_{\alpha=-\infty}^{\infty} A_{0+\alpha n} + \sum_{k=1}^{N/2}\sum_{\alpha=-\infty}^{\infty} \frac{1}{2}\left(A_{k+\alpha n} - iB_{k+\alpha n}\right)\exp(2\pi k x/L) \\ &\quad + \sum_{k=1}^{N/2}\sum_{\alpha=-\infty}^{\infty} \frac{1}{2}\left(A_{k+\alpha n} + iB_{k+\alpha n}\right)\exp(-2\pi k x/L) \\ &= \sum_{k=-N/2}^{N/2}\sum_{\alpha=-\infty}^{\infty} c_{k+\alpha n}\exp(2\pi k x/L) \\ &= \sum_{k=-N/2}^{N/2} \hat{u}_k \exp(2\pi k x/L). \end{aligned}$$

We remark an alternative more abstract derivation also could have been performed by using $\overline{c_k} = c_{-k}$ and conjugacy conditions of the complex exponentials. From the considerations above, we obtain a real-valued interpolation for approximating the function between the lattice sites by using

$$\tilde{u}(x) = \sum_{k=-N/2}^{N/2} \hat{u}_k e^{i2\pi k x/L},$$

where $N = n - 1$. It will also be useful sometimes to express this as

$$\tilde{u}(x) = \mathcal{I}[\{u_m\}](x) = \hat{\mathcal{I}}[\{\hat{u}_k\}](x) = \sum_{k=-N/2}^{N/2} \hat{u}_k e^{i2\pi k x/L}.$$

The derivation above also can be further extended to obtain a real-valued interpolation when n is even. We remark that to obtain a real-valued interpolation we needed to center the expansion between $-(n-1)/2$ to $(n-1)/2$ to balance the complex terms. Here, we leveraged the aliasing to accomplish this balance to obtain a real-valued interpolation. We will refer to $\mathcal{I}[\{u_m\}](x)$ and $\hat{\mathcal{I}}[\{\hat{u}_k\}](x)$ as the *fourier interpolation* for the function sampled at the lattice locations x_m with values u_m. The only difference between $\mathcal{I}$ and $\hat{\mathcal{I}}$ is the notation for if we are thinking of the interpolation as being obtained from the lattice values $\{u_m\}$ directly or the fourier modes $\{\hat{u}_k\}$, but ultimately they yield the same final function $\tilde{u}$.

Approximating Solutions of Poisson PDE. Consider the Poisson Partial Differential Equation (PDE) on the interval $[0, L]$,

$$\Delta u = -f, \;\; u(0) = u(L), \;\; \int_0^L u(x)dx = 0.$$

To approximate the solution, consider the case when the function can be represented exactly as

$$\tilde{u}(x) = \sum_{k=-N/2}^{N/2} \hat{u}_k e^{i2\pi kx/L}.$$

The Laplacian of such a function would then be given by

$$\begin{aligned}\Delta\tilde{u}(x) &= \sum_{k=-N/2}^{N/2} \left(-\frac{4\pi^2 k^2}{L^2}\right) \hat{u}_k e^{i2\pi kx/L} \\ &= \sum_{k=-N/2}^{N/2} \hat{\mathcal{L}}_k \hat{u}_k e^{i2\pi kx/L}.\end{aligned}$$

The fourier symbol of the Laplacian is $\hat{\mathcal{L}}_k = \left(-\frac{4\pi^2 k^2}{L^2}\right)$. Similarly, consider the case when f is of the form

$$\tilde{f}(x) = \sum_{k=-N/2}^{N/2} \hat{f}_k e^{i2\pi kx/L}.$$

By setting $\Delta\tilde{u} = -\tilde{f}$ we obtain

$$\hat{u}_k = \frac{-f_k}{\hat{\mathcal{L}}_k}.$$

This holds for $k \neq 0$. In the case $k = 0$, we have $\hat{u}_k = 0$ by the integral condition. This provides the solution, since these are the fourier coefficients $\{\hat{u}_k\}_{k=0}^{n-1}$ needed to construct $\tilde{u} = \hat{\mathcal{I}}[\{\hat{u}_k\}_{k=0}^{n-1}]$. The assumptions about the form of u and f were that we could represent it exactly using the finite expansion. In general this will not be the case. Instead, we can think about how functions sampled on the lattice would be projected to functions of this form. From the aliasing formula above, we then have some insight into the errors of this approximation.

To obtain an approximate solution to the Poisson PDE, we proceed as follows. Sample the function f at the lattice locations x_m and compute the discrete fourier transform (DFT) to obtain $\hat{f}_k = \mathcal{F}_k[f]$. Compute the fourier coefficients $\hat{u}_k = -f_k/\hat{\mathcal{L}}_k$ as above. Now compute the inverse discrete fourier transform to obtain the solution u at the lattice locations x_m. This solution also can be interpolated off the lattice using the fourier coefficients $\hat{u}_k$ to obtain the approximate solution function $\tilde{u}(x) = \hat{\mathcal{I}}[\{\hat{u}_k\}](x)$. The approach above provides for smooth functions f typically good approximations of the solution of the PDE. The approaches above also can be readily extended to higher dimensions for rectangles, cubes, and other geometries.

Summary: The discrete fourier transform provides some versatile ways to obtain approximate solutions to the Poisson PDE. Since these approximation methods make use of properties of the spectrum of the differential operator, eigenvalues and eigenfunctions, they are referred to as *spectral numerical methods*. We emphasize that the above procedures did not require knowledge of the functional form of f or computing any integrals, only that we can sample the function at the lattice locations x_m. The efficiency of the methods will depend on how one computes the DFT and IDFT. In practice, these can be computed in $O(n\log(n))$ time by using *fast fourier transforms*. This provides some general ways to solve elliptic partial differential equations and also works for other geometries. Much of the efficiency then depends on the approximation of fourier expansions or development of alternative fourier-like representations. The spectral methods also tend to work best for functions which are smooth. When functions have more localized features there are also alternative approaches that are widely used such as *finite difference methods* and *finite element methods* to approximate the solutions of PDEs. We also remark that the discrete fourier transforms (DFTs) and (IDFTs) are also widely used in other related settings for approximating soutions of PDEs, signal processing, statistical analysis, and other applications.