Math 6620: Analysis of Numerical Methods, II
Interpolation with Fourier Series

See Hesthaven, S. Gottlieb, and D. Gottlieb 2007, Chapters 2-3,
Canuto et al. 2011, Chapter 2.1,

Shen, Tang, and Wang 2011, Chapter 2

Akil Narayan[1]

[1] **Department of Mathematics, and Scientific Computing and Imaging (SCI) Institute**
University of Utah

We have established that Fourier series approximations u_N,

$$u_N(x) = \sum_{|k| \leqslant N} \widehat{u}_k \phi_k(x), \qquad \phi_k(x) = \frac{1}{\sqrt{2\pi}} e^{ikx}, \qquad \widehat{u}_k = \langle u, \phi_k \rangle,$$

have orders of convergence that depend on the smoothness of u:

$$u \in H_p^s \quad \Longrightarrow \quad \|u - u_N\|_{L^2} \leqslant N^{-s} \|u\|_{H_p^s}.$$

I.e.,

$$\text{Smoothness} \Longrightarrow \text{Compressibility}$$

Fourier Series approximations

We have established that Fourier series approximations u_N,

$$u_N(x) = \sum_{|k| \leqslant N} \widehat{u}_k \phi_k(x), \qquad \phi_k(x) = \frac{1}{\sqrt{2\pi}} e^{ikx}, \qquad \widehat{u}_k = \langle u, \phi_k \rangle,$$

have orders of convergence that depend on the smoothness of u:

$$u \in H_p^s \quad \Longrightarrow \quad \|u - u_N\|_{L^2} \leqslant N^{-s} \|u\|_{H_p^s}.$$

I.e.,

$$\text{Smoothness} \Longrightarrow \text{Compressibility}$$

One major outstanding question is *how* we actually compute $\widehat{u}_k$ in practice.

Quadrature

The expansion coefficients require computing an integral,

$$\widehat{u}_k = \frac{1}{\sqrt{2\pi}} \int_0^{2\pi} u(x) e^{-ikx} \mathrm{d}x,$$

which can be intractable, even if an explicit formula for u is available.

Quadrature

The expansion coefficients require computing an integral,

$$\widehat{u}_k = \frac{1}{\sqrt{2\pi}} \int_0^{2\pi} u(x) e^{-ikx} \mathrm{d}x,$$

which can be intractable, even if an explicit formula for u is available.

A standard recourse is to approximate the integral with quadrature:

$$\frac{1}{\sqrt{2\pi}} \int_0^{2\pi} u(x) e^{-ikx} \mathrm{d}x \approx \sum_{j=1}^{M} w_{k,j} u(x_j), \qquad w_{k,j} = \frac{\sqrt{2\pi}}{M} e^{-ikx_j}, \qquad x_j = \frac{2\pi(j-1)}{M},$$

where we have made particular *choices*:

- x_j are equispaced on $[0, 2\pi]$ for $j \in [M]$
- $w_{k,j}$ correspond to a uniform quadrature rule

The expansion coefficients require computing an integral,

$$\widehat{u}_k = \frac{1}{\sqrt{2\pi}} \int_0^{2\pi} u(x) e^{-ikx} \mathrm{d}x,$$

which can be intractable, even if an explicit formula for u is available.

A standard recourse is to approximate the integral with quadrature:

$$\frac{1}{\sqrt{2\pi}} \int_0^{2\pi} u(x) e^{-ikx} \mathrm{d}x \approx \sum_{j=1}^{M} w_{k,j} u(x_j), \qquad w_{k,j} = \frac{\sqrt{2\pi}}{M} e^{-ikx_j}, \qquad x_j = \frac{2\pi(j-1)}{M},$$

where we have made particular *choices*:

- x_j are equispaced on $[0, 2\pi]$ for $j \in [M]$
- $w_{k,j}$ correspond to a uniform quadrature rule
- We'll also assume that $M = 2N + 1$. (Quadrature nodes = number of expansion coefficients)

Note that this is just the trapezoid rule on $[0, 2\pi]$ with periodic boundary conditions.

One can make other choices, but these choices are most convenient for discussing the major concepts surrounding theory and computation.

Quadrature as linear algebra

$$\widehat{u}_k \approx \widetilde{u}_k := \sum_{j=1}^{M} w_{k,j} u(x_j), \qquad w_{k,j} = \frac{\sqrt{2\pi}}{M} e^{-ikx_j}, \qquad x_j = \frac{2\pi(j-1)}{M}.$$

We compute these coefficients for all $|k| \leqslant N$, with $M = 2N + 1$.

$$\widehat{u}_k \approx \widetilde{u}_k := \sum_{j=1}^{M} w_{k,j} u(x_j), \qquad w_{k,j} = \frac{\sqrt{2\pi}}{M} e^{-ikx_j}, \qquad x_j = \frac{2\pi(j-1)}{M}.$$

We compute these coefficients for all $|k| \leqslant N$, with $M = 2N+1$.

A simple implementation quadrature of amounts to matrix-vector algebra:

$$\boldsymbol{u} := \begin{pmatrix} u(x_1) \\ u(x_2) \\ \vdots \\ u(x_M) \end{pmatrix}, \qquad \widetilde{\boldsymbol{u}} := \begin{pmatrix} \widetilde{u}_{-N} \\ \widetilde{u}_{-N+1} \\ \vdots \\ \widetilde{u}_N \end{pmatrix} \implies \widetilde{\boldsymbol{u}} = \widetilde{\boldsymbol{V}}^* \boldsymbol{u},$$

where $\widetilde{\boldsymbol{V}}^*$ is the conjugate transpose of $\widetilde{\boldsymbol{V}}$, which in turn is given by,

$$\widetilde{\boldsymbol{V}} = \sqrt{\frac{2\pi}{M}} \boldsymbol{V}, \qquad \boldsymbol{V} = \begin{pmatrix} | & | & & | \\ \boldsymbol{v}_{-N} & \boldsymbol{v}_{-N+1} & \cdots & \boldsymbol{v}_N \\ | & | & & | \end{pmatrix}, \qquad \boldsymbol{v}_k = \sqrt{\frac{2\pi}{M}} \phi_k(\boldsymbol{x}),$$

and $\boldsymbol{x} = (x_1, x_2, \ldots, x_M)^T$.

$$\widetilde{\boldsymbol{V}} = \sqrt{\frac{2\pi}{M}}\boldsymbol{V}, \qquad \boldsymbol{V} = \begin{pmatrix} | & | & & | \\ \boldsymbol{v}_{-N} & \boldsymbol{v}_{-N+1} & \cdots & \boldsymbol{v}_{N} \\ | & | & & | \end{pmatrix}, \qquad \boldsymbol{v}_k = \sqrt{\frac{2\pi}{M}}\phi_k(\boldsymbol{x}),$$

A somewhat straightforward computation shows:

$$\langle \boldsymbol{v}_\ell, \boldsymbol{v}_k \rangle = \frac{1}{M}\sum_{j=1}^{M} e^{i(\ell-k)2\pi(j-1)/M} = \frac{1}{M}\sum_{j=0}^{M-1} \Big(\underbrace{e^{i(\ell-k)2\pi/M}}_{w^j}\Big)^j,$$

Duality of Fourier Series with quadrature

$$\widetilde{\boldsymbol{V}} = \sqrt{\frac{2\pi}{M}} \boldsymbol{V}, \qquad \boldsymbol{V} = \begin{pmatrix} | & | & & | \\ \boldsymbol{v}_{-N} & \boldsymbol{v}_{-N+1} & \cdots & \boldsymbol{v}_N \\ | & | & & | \end{pmatrix}, \qquad \boldsymbol{v}_k = \sqrt{\frac{2\pi}{M}} \phi_k(\boldsymbol{x}),$$

A somewhat straightforward computation shows:

$$\langle \boldsymbol{v}_\ell, \boldsymbol{v}_k \rangle = \frac{1}{M} \sum_{j=1}^{M} e^{i(\ell-k)2\pi(j-1)/M} = \frac{1}{M} \sum_{j=0}^{M-1} \left(e^{i(\ell-k)2\pi/M} \right)^j,$$

Thus, in particular if $\ell = k$ then $\langle \boldsymbol{v}_\ell, \boldsymbol{v}_k \rangle = 1$, and for $\ell \neq k$ and $|\ell - k| \leqslant M - 1$:

$$\langle \boldsymbol{v}_\ell, \boldsymbol{v}_k \rangle = \frac{1}{M} \frac{1 - \left(e^{i(\ell-k)2\pi/M}\right)^M}{1 - e^{i(\ell-k)2\pi/M}} = 0$$

I.e., $\{\boldsymbol{v}_k\}_{|k| \leqslant N}$ are *orthonormal vectors*.

$$\widetilde{\boldsymbol{V}} = \sqrt{\frac{2\pi}{M}}\boldsymbol{V}, \qquad \boldsymbol{V} = \begin{pmatrix} | & | & & | \\ \boldsymbol{v}_{-N} & \boldsymbol{v}_{-N+1} & \cdots & \boldsymbol{v}_N \\ | & | & & | \end{pmatrix}, \qquad \boldsymbol{v}_k = \sqrt{\frac{2\pi}{M}}\phi_k(\boldsymbol{x}),$$

A somewhat straightforward computation shows:

$$\langle \boldsymbol{v}_\ell, \boldsymbol{v}_k \rangle = \frac{1}{M}\sum_{j=1}^{M} e^{i(\ell-k)2\pi(j-1)/M} = \frac{1}{M}\sum_{j=0}^{M-1}\left(e^{i(\ell-k)2\pi/M}\right)^j,$$

Thus, in particular if $\ell = k$ then $\langle \boldsymbol{v}_\ell, \boldsymbol{v}_k \rangle = 1$, and for $\ell \neq k$ and $|\ell - k| \leqslant M - 1$:

$$\langle \boldsymbol{v}_\ell, \boldsymbol{v}_k \rangle = \frac{1}{M}\frac{1 - \left(e^{i(\ell-k)2\pi/M}\right)^M}{1 - e^{i(\ell-k)2\pi/M}} = 0$$

I.e., $\{\boldsymbol{v}\}_{|k|\leqslant N}$ are *orthonormal vectors*.

This shows the important property that $\boldsymbol{V}$ is a unitary matrix:

$$\boldsymbol{V}^*\boldsymbol{V} = \boldsymbol{I} \quad \Longrightarrow \quad \boldsymbol{V}^{-1} = \boldsymbol{V}^*.$$

Inverting the Fourier Series

Putting everything together:

$$\widetilde{\boldsymbol{u}} = \widetilde{\boldsymbol{V}}^* \boldsymbol{u}, \qquad \widetilde{\boldsymbol{V}} = \sqrt{\frac{2\pi}{M}} \boldsymbol{V}, \qquad \boldsymbol{V}^{-1} = \boldsymbol{V}^*.$$

This implies that:

$$\boldsymbol{u} = \left(\widetilde{\boldsymbol{V}}^*\right)^{-1} \widetilde{\boldsymbol{u}} = \sqrt{\frac{M}{2\pi}} \left(\boldsymbol{V}^*\right)^{-1} \widetilde{\boldsymbol{u}} = \sqrt{\frac{M}{2\pi}} \boldsymbol{V} \widetilde{\boldsymbol{u}} = \frac{M}{2\pi} \widetilde{\boldsymbol{V}} \widetilde{\boldsymbol{u}}.$$

Putting everything together:

$$\widetilde{\boldsymbol{u}} = \widetilde{\boldsymbol{V}}^* \boldsymbol{u}, \qquad \widetilde{\boldsymbol{V}} = \sqrt{\frac{2\pi}{M}} \boldsymbol{V}, \qquad \boldsymbol{V}^{-1} = \boldsymbol{V}^*.$$

This implies that:

$$\boldsymbol{u} = \left(\widetilde{\boldsymbol{V}}^*\right)^{-1} \widetilde{\boldsymbol{u}} = \sqrt{\frac{M}{2\pi}} \left(\boldsymbol{V}^*\right)^{-1} \widetilde{\boldsymbol{u}} = \sqrt{\frac{M}{2\pi}} \boldsymbol{V} \widetilde{\boldsymbol{u}} = \frac{M}{2\pi} \widetilde{\boldsymbol{V}} \widetilde{\boldsymbol{u}}.$$

I.e., the map between $\boldsymbol{u}$ and $\widetilde{\boldsymbol{u}}$ is invertible and quite explicit:

$$\widetilde{\boldsymbol{u}} = \widetilde{\boldsymbol{V}}^* \boldsymbol{u}, \qquad \boldsymbol{u} = \frac{M}{2\pi} \widetilde{\boldsymbol{V}} \widetilde{\boldsymbol{u}}.$$

This invertible map is called the Discrete Fourier Transform (DFT). As a consequence of $\boldsymbol{V}$ being unitary, we have also shown that the DFT is a (scaled) isometry,

$$\int_0^{2\pi} |u(x)|^2 \mathrm{d}x \approx \frac{2\pi}{M} \|\boldsymbol{u}\|_2^2 = \|\widetilde{\boldsymbol{u}}\|_2^2,$$

which is the discrete analogue of Parseval's identity.

The Fast Fourier Transform, I

The inverse/DFT is relatively expensive:

$$\boldsymbol{u} \xrightarrow{\mathcal{O}(M^2)} \widetilde{\boldsymbol{V}}^* \boldsymbol{u}, \qquad \widetilde{\boldsymbol{u}} \xrightarrow{\mathcal{O}(M^2)} \frac{M}{2\pi} \widetilde{\boldsymbol{V}} u$$

One of the most well-known algorithms is the *fast Fourier transform*, which is a fast algorithm for accomplishing the particular matrix-vector multiplication $\widetilde{\boldsymbol{V}}^* \boldsymbol{u}$.

It is simpler to explain the basic idea if M is even, in which case we have:

$$\begin{aligned} \frac{M}{\sqrt{2\pi}} \widetilde{u}_k &= \sum_{j=1}^{M} u(x_j) e^{-ikx_j} = \sum_{j=1}^{M} u(x_j) e^{-ik2\pi(j-1)/M} \\ &= \sum_{j=1}^{M/2} u(x_{2j}) e^{-ik2\pi 2(j-1)/M} + \sum_{j=1}^{M/2} u(x_{2j-1}) e^{-ik2\pi(2j-1)/M} \end{aligned}$$

The Fast Fourier Transform, I

The inverse/DFT is relatively expensive:

$$\boldsymbol{u} \xrightarrow{\mathcal{O}(M^2)} \widetilde{\boldsymbol{V}}^* \boldsymbol{u}, \qquad\qquad \widetilde{\boldsymbol{u}} \xrightarrow{\mathcal{O}(M^2)} \frac{M}{2\pi} \widetilde{\boldsymbol{V}} . u$$

One of the most well-known algorithms is the *fast Fourier transform*, which is a fast algorithm for accomplishing the particular matrix-vector multiplication $\widetilde{\boldsymbol{V}}^* \boldsymbol{u}$.

1 M^2 op $\sim M$
vs.
2 $(\frac{M}{2})^2$ op $\sim M^2/2$

It is simpler to explain the basic idea if M is even, in which case we have:

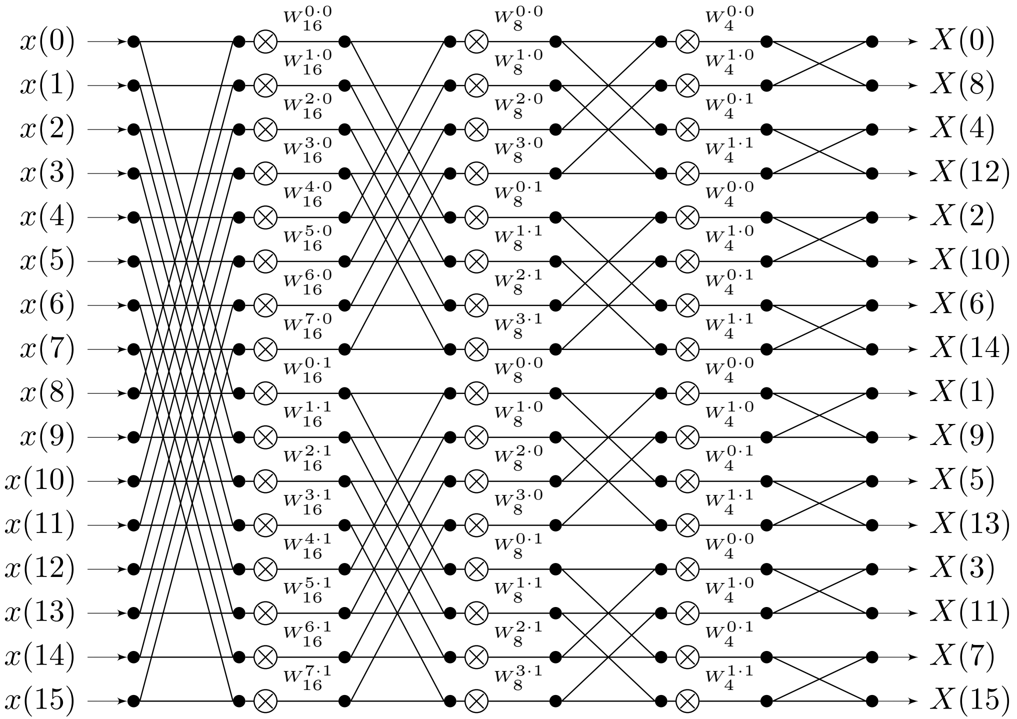
$$\begin{aligned}
\frac{M}{\sqrt{2\pi}} \widetilde{u}_k &= \sum_{j=1}^{M} u(x_j) e^{-ikx_j} = \sum_{j=1}^{M} u(x_j) e^{-ik2\pi(j-1)/M} \\
&= \sum_{j=1}^{M/2} u(x_{2j}) e^{-ik2\pi 2(j-1)/M} + \sum_{j=1}^{M/2} u(x_{2j-1}) e^{-ik2\pi(2j-1)/M} \\
&= \sum_{j=1}^{M/2} u(x_{2j}) e^{-ik2\pi 2(j-1)/M} + e^{ik2\pi/M} \sum_{j=1}^{M/2} u(x_{2j-1}) e^{-ik2\pi(2j-2)/M}.
\end{aligned}$$

Note that the last two sums are $M/2$-point DFT coefficients associated with half the data (either at x_{2j} or at x_{2j-1}).

I.e., with some book-keeping, we can compute the M-point DFT using 2 $M/2$-point DFT's.

The Fast Fourier Transform, II

This logic can be repeated, showing that actually we can compute the M-point DFT using J (M/J)-point DFT's, where J is a power of two. This yields the simplest, *radix 2 fast Fourier transform* (FFT) algorithm.



$J = M/2$

$J\left(\frac{M}{J}\right)^2 \sim \frac{M}{2}$

$+ \mathcal{O}(M)$ ops at every level

\# levels $= \log_2 M$

Through this divide-and-conquer strategy, an M-point DFT that naively requires $\mathcal{O}(M^2)$ complexity can be accomplished in $\mathcal{O}(M \log M)$ time.

$$\widetilde{\boldsymbol{u}} = \widetilde{\boldsymbol{V}}^* \boldsymbol{u}, \qquad \boldsymbol{u} = \frac{M}{2\pi} \widetilde{\boldsymbol{V}} \widetilde{\boldsymbol{u}}.$$

We have introduced the DFT via quadrature, but an alternative and illustrative viewpoint is *interpolation*.

Note that the coefficients $\widetilde{\boldsymbol{u}}$ are determined by the conditions,

$$\frac{M}{2\pi} \widetilde{\boldsymbol{V}} \widetilde{\boldsymbol{u}} = \boldsymbol{u} \quad \Longrightarrow \quad \begin{pmatrix} | & | & & | \\ \boldsymbol{\phi}_{-N} & \boldsymbol{\phi}_{-N+1} & \cdots & \boldsymbol{\phi}_{N} \\ | & | & & | \end{pmatrix} \widetilde{\boldsymbol{u}} = \boldsymbol{u}.$$

Note that these are "just" interpolation conditions for the $\widetilde{\boldsymbol{u}}$ at the data points x_j, $j \in [M]$.

$$\widetilde{\boldsymbol{u}} = \widetilde{\boldsymbol{V}}^* \boldsymbol{u}, \qquad \boldsymbol{u} = \frac{M}{2\pi} \widetilde{\boldsymbol{V}} \widetilde{\boldsymbol{u}}.$$

We have introduced the DFT via quadrature, but an alternative and illustrative viewpoint is *interpolation*.

Note that the coefficients $\widetilde{\boldsymbol{u}}$ are determined by the conditions,

$$\frac{M}{2\pi} \widetilde{\boldsymbol{V}} \widetilde{\boldsymbol{u}} = \boldsymbol{u} \quad \Longrightarrow \quad \begin{pmatrix} | & | & & | \\ \boldsymbol{\phi}_{-N} & \boldsymbol{\phi}_{-N+1} & \cdots & \boldsymbol{\phi}_N \\ | & | & & | \end{pmatrix} \widetilde{\boldsymbol{u}} = \boldsymbol{u}.$$

Note that these are "just" interpolation conditions for the $\widetilde{\boldsymbol{u}}$ at the data points x_j, $j \in [M]$.

Hence, $u_N(x) = \sum_{|k| \leqslant N} \widetilde{u}_k \phi_k(x)$ interpolates the data $\boldsymbol{u}$.

We have already concluded that this interpolation problem is *unisolvent*. Hence, there are *cardinal basis functions* $\ell_j(x)$, $j \in [M]$ such that,

$$u_N(x) = \sum_{j \in [M]} u(x_j) \ell_j(x), \qquad \ell_j(x_r) = \delta_{j,r}.$$

I.e., u_N has a Lagrange form.

Cardinal Lagrange basis

The cardinal Lagrange functions yield insight into the interpolation process.

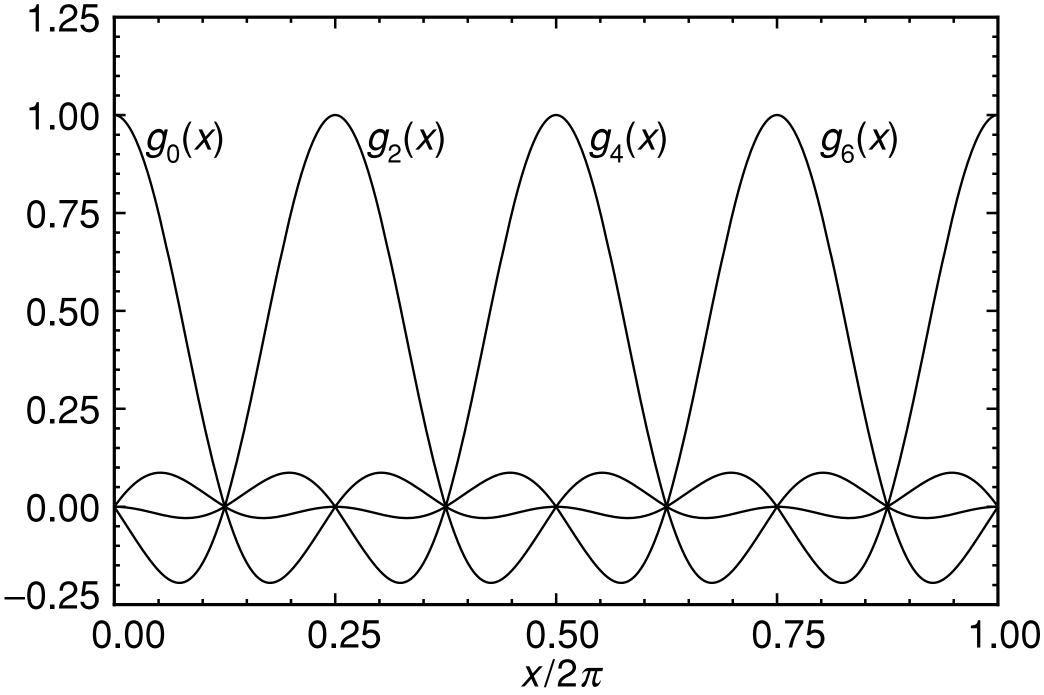


Figure 2.3 of Hesthaven, S. Gottlieb, and D. Gottlieb 2007

Note that interpolation implies

$$u(x) \in V_N := \operatorname{span}\left\{e^{ikx}\right\}_{|k| \leqslant N} \qquad \Longrightarrow \qquad I_N u := \sum_{|k| \leqslant N} \widetilde{u}_k \phi_k(x) = u(x).$$

Aliasing

The fact that our DFT is an interpolation process reveals a significant issue that we must be cognizant of: aliasing error.

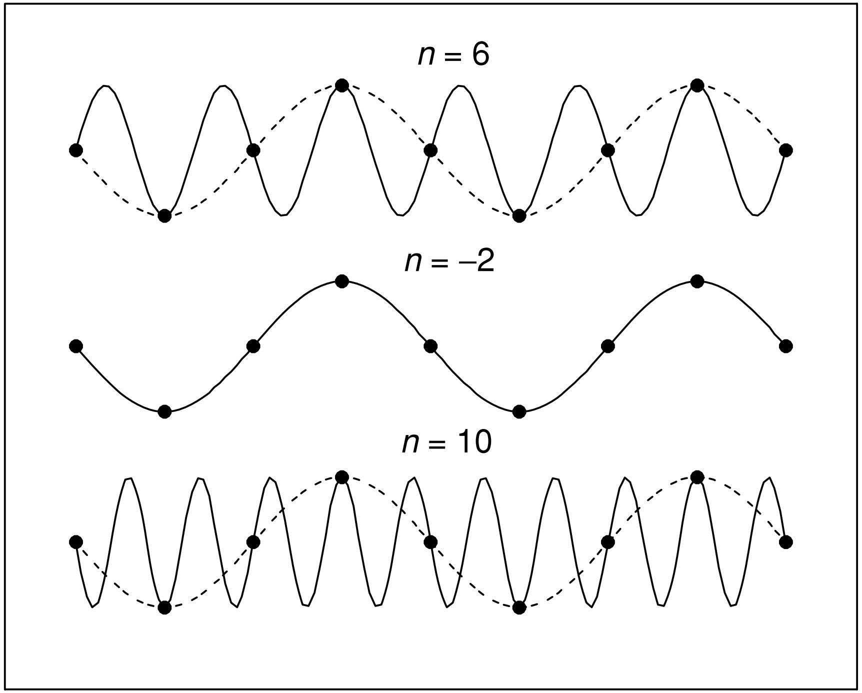


Figure 2.7 of Hesthaven, S. Gottlieb, and D. Gottlieb 2007

The fact that our DFT is an interpolation process reveals a significant issue that we must be cognizant of: aliasing error.

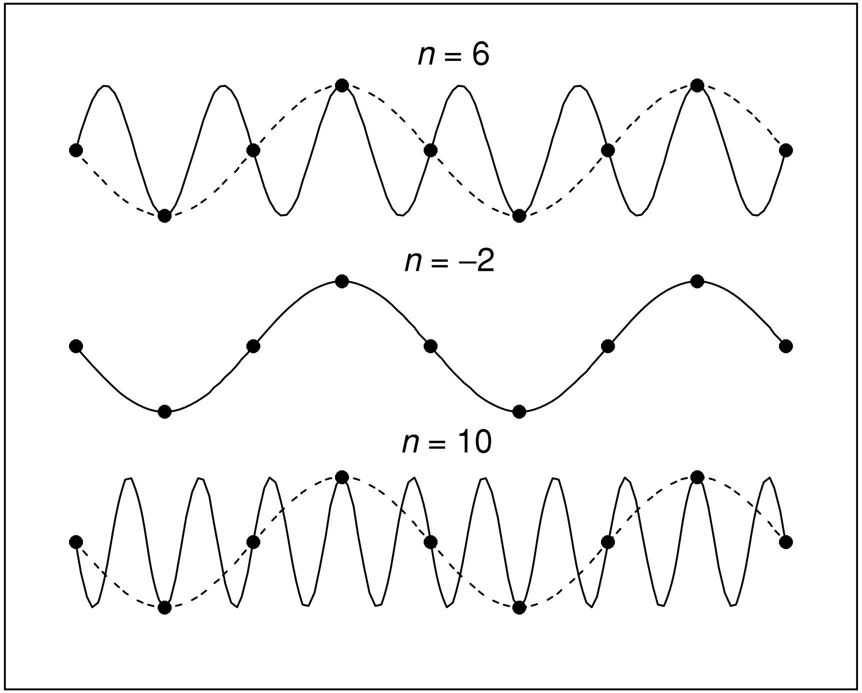


Figure 2.7 of Hesthaven, S. Gottlieb, and D. Gottlieb 2007

So, for example, even if $\left\langle e^{i\ell x}, \phi_k(x) \right\rangle = 0$ for $\ell > N$, it's possible that $I_N e^{i\ell x} \neq 0$.

I.e., the interpolation/DFT procedure *is* a projection operator, it's just an oblique one.

Aliasing error

Aliasing is not just an academic curiosity: with P_N the L^2-orthogonal projection operator onto

$$V_N = \text{span}\left\{e^{ikx}\right\}_{|k|\leqslant N},$$

recall that $u \in H_p^s$ implies that $\|u - P_N u\|_{L^2} \lesssim N^{-s}$.

Ok, but what about $I_N u$?

Aliasing error

Aliasing is not just an academic curiosity: with P_N the L^2-orthogonal projection operator onto

$$V_N = \operatorname{span}\left\{e^{ikx}\right\}_{|k|\leqslant N},$$

recall that $u \in H_p^s$ implies that $\|u - P_N u\|_{L^2} \lesssim N^{-s}$.

Ok, but what about $I_N u$?

The main strategy to understanding this is to estimate the aliasing error. Note that for the L^2 norm,

$$\begin{aligned}\|u - I_N u\| = \|\,(u - P_N u) + (P_N u - I_N u)\,\| &\leqslant \|u - P_N u\| + \|P_N u - I_N u\| \\ &=: \|u - P_N u\| + \|A_N u\|,\end{aligned}$$

where we have defined the aliasing error $A_N u$.

$$A_N u = P_N u - I_N u$$

The following observations are crucial:

- If $u \in V_N$, then $I_N u = P_N u = u$, so $A_N u = 0$. Therefore, $A_N u = A_N(I - P_N)u$.
  The aliasing error is only affected by truncation error.

Bounding the aliasing error

$$A_N u = P_N u - I_N u$$

The following observations are crucial:

- If $u \in V_N$, then $I_N u = P_N u = u$, so $A_N u = 0$. Therefore, $A_N u = A_N (I - P_N) u$.
  The aliasing error is only affected by truncation error.
- We know that $(I - P_N) u$ is small. Therefore, if A_N is "behaves well", then $A_N u$ will be small.
  The truncation error is small, but does A_N amplify small inputs?

Bounding the aliasing error

$$A_N u = P_N u - I_N u$$

The following observations are crucial:

- If $u \in V_N$, then $I_N u = P_N u = u$, so $A_N u = 0$. Therefore, $A_N u = A_N(I - P_N)u$.
  The aliasing error is only affected by truncation error.
- We know that $(I - P_N)u$ is small. Therefore, if A_N is "behaves well", then $A_N u$ will be small.
  The truncation error is small, but does A_N amplify small inputs?
- A_N is well-behaved: for $|k| \leqslant N$,

$$A_N e^{i(k+(2N+1))x} = e^{ikx},$$

and thus in particular,

$$u = \sum_{|k| \leqslant N} \widehat{u}_k \phi_k(x) \qquad \Longrightarrow \qquad \widetilde{u}_k = \sum_{\ell \in \mathbb{Z}} \widehat{u}_{k+\ell(2N+1)}.$$

$k \in \mathbb{Z}$

A_N does not amplify small inputs.

Bounding the aliasing error

$$A_N u = P_N u - I_N u$$

The following observations are crucial:

- If $u \in V_N$, then $I_N u = P_N u = u$, so $A_N u = 0$. Therefore, $A_N u = A_N (I - P_N) u$.
  The aliasing error is only affected by truncation error.
- We know that $(I - P_N)u$ is small. Therefore, if A_N is "behaves well", then $A_N u$ will be small.
  The truncation error is small, but does A_N amplify small inputs?
- A_N is well-behaved: for $|k| \leqslant N$,

  $$A_N e^{i(k+(2N+1))x} = e^{ikx},$$

  and thus in particular,

  $$u = \sum_{|k| \leqslant N} \widehat{u}_k \phi_k(x) \quad \Longrightarrow \quad \widetilde{u}_k = \sum_{\ell \in \mathbb{Z}} \widehat{u}_{k+\ell(2N+1)}.$$

  A_N does not amplify small inputs.

Therefore, if $\widehat{u}_{k+\ell(2N+1)}$ decays quickly for large $|\ell|$, then we can expect the aliased coefficients $\widetilde{u}_k$ to be "close" to $\widehat{u}_k$.

Interpolation estimates

$H_p^s \simeq u^{(s)} \in L^2$

While we have only discussed the high-level ideas, going through the details produces the following estimate:

Theorem

Assume $u \in H_p^s$ *with* $s > 1/2$. *Then*

$$\|u - I_N u\|_{L^2} \lesssim N^{-s}\|u\|_{H^s} \qquad .$$

$$\|u - I_N u\|_{H^r} \lesssim N^{-(s-r)}\|u\|_{H^s}, \qquad r < s.$$

Note that this is exactly the asymptotic behavior for the exact orthogonal projector P_N. Thus, one can expect the DFT to produce good results.

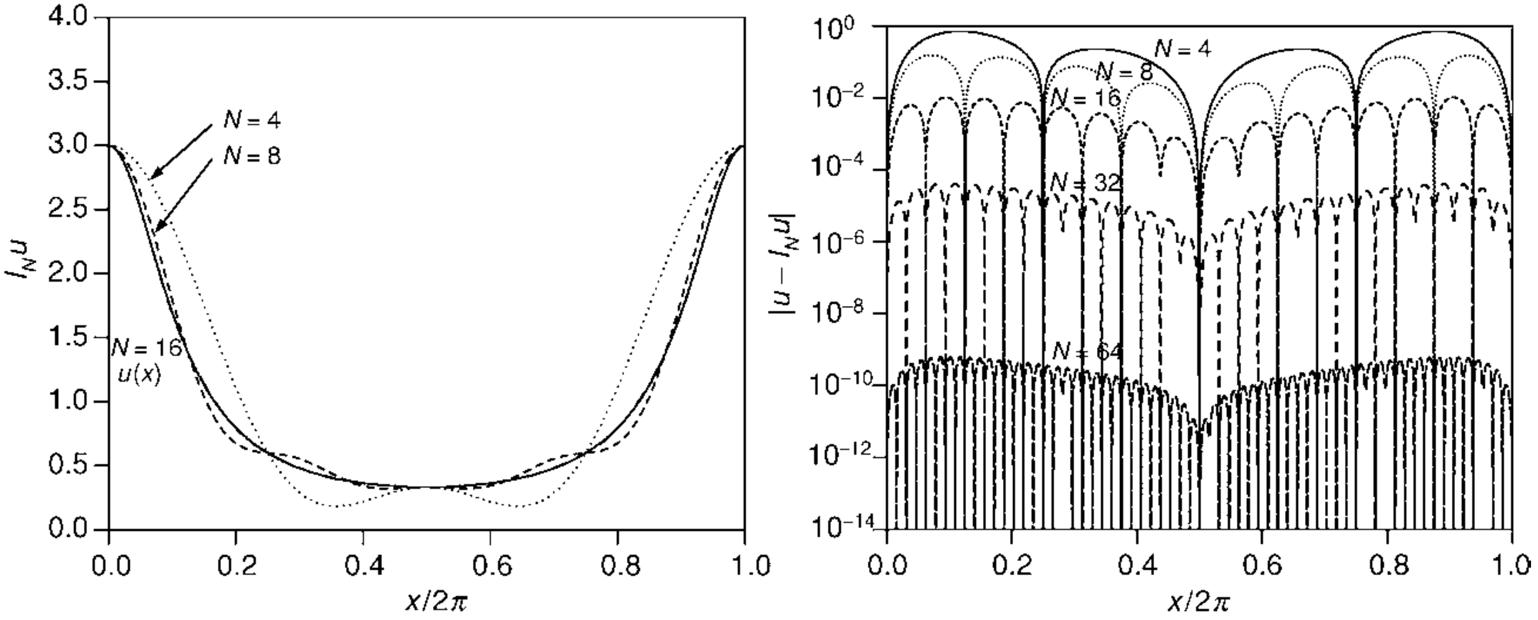


Figure 2.4 of Hesthaven, S. Gottlieb, and D. Gottlieb 2007

$$u(x) = \frac{3}{5 - 4\cos x}$$

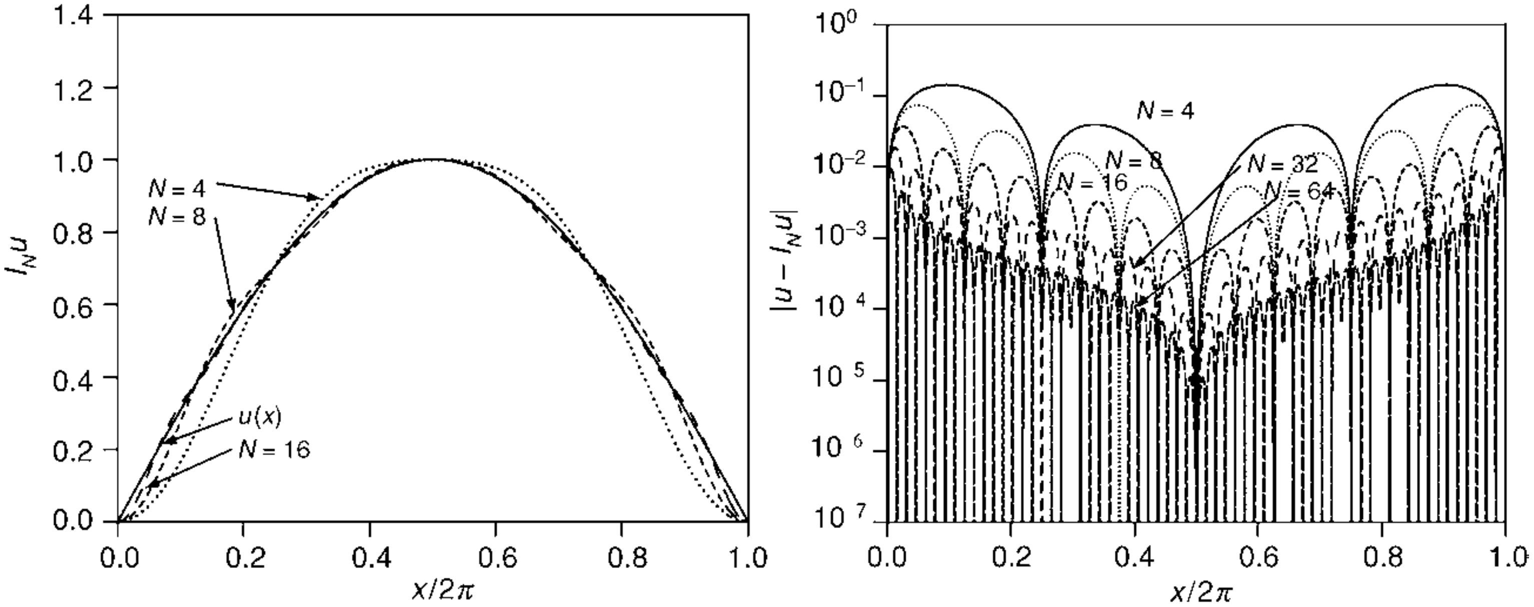


Figure 2.4 of Hesthaven, S. Gottlieb, and D. Gottlieb 2007

$$u(x) = \sin(x/2)$$

References I

- Canuto, Claudio et al. (2011). *Spectral Methods: Fundamentals in Single Domains*. 1st ed. 2006. Corr. 4th printing 2010 edition. Berlin ; New York: Springer. ISBN: 978-3-540-30725-9.
- Cooley, James W. and John W. Tukey (1965). "An Algorithm for the Machine Calculation of Complex Fourier Series". In: *Mathematics of Computation* 19.90, pp. 297–301. ISSN: 0025-5718, 1088-6842. DOI: 10.1090/S0025-5718-1965-0178586-1.
- Hesthaven, Jan S., Sigal Gottlieb, and David Gottlieb (2007). *Spectral Methods for Time-Dependent Problems*. Cambridge University Press. ISBN: 0-521-79211-8.
- Shen, Jie, Tao Tang, and Li-Lian Wang (2011). *Spectral Methods: Algorithms, Analysis and Applications*. Springer Science & Business Media. ISBN: 978-3-540-71041-7.