

Mathematical Tripos Part II: Michaelmas Term 2025

Numerical Analysis – Lecture 12

Method 3.8 (The algebra of Fourier expansions) Let \mathcal{A} be the set of all functions $f : [-1, 1] \rightarrow \mathbb{C}$, which are analytic in $[-1, 1]$, periodic with period 2, and that can be extended analytically into the complex plane. Then \mathcal{A} is a linear space, i.e., $f, g \in \mathcal{A}$ and $\alpha \in \mathbb{C}$ then $f + g \in \mathcal{A}$ and $\alpha f \in \mathcal{A}$. In particular, with f and g expressed in its Fourier series, i.e.,

$$f(x) = \sum_{n=-\infty}^{\infty} \hat{f}_n e^{i\pi n x}, \quad g(x) = \sum_{n=-\infty}^{\infty} \hat{g}_n e^{i\pi n x}$$

we have

$$f(x) + g(x) = \sum_{n=-\infty}^{\infty} (\hat{f}_n + \hat{g}_n) e^{i\pi n x}, \quad \alpha f(x) = \sum_{n=-\infty}^{\infty} \alpha \hat{f}_n e^{i\pi n x} \quad (3.3)$$

and

$$f(x) \cdot g(x) = \sum_{n=-\infty}^{\infty} \left(\sum_{m=-\infty}^{\infty} \hat{f}_{n-m} \hat{g}_m \right) e^{i\pi n x} = \sum_{n=-\infty}^{\infty} (\hat{f} * \hat{g})_n e^{i\pi n x}, \quad (3.4)$$

where $*$ denotes the convolution operator, hence $(\widehat{f \cdot g})_n = (\hat{f} * \hat{g})_n$. Moreover, if $f \in \mathcal{A}$ then $f' \in \mathcal{A}$ and

$$f'(x) = i\pi \sum_{n=-\infty}^{\infty} n \cdot \hat{f}_n e^{i\pi n x}. \quad (3.5)$$

Since $\{\hat{f}_n\}$ decays faster than $\mathcal{O}(n^{-p})$ for any $p \in \mathbb{N}$, this provides that all derivatives of f have rapidly convergent Fourier expansions.

Example 3.9 (Application to differential equations) Consider the two-point boundary value problem: $y = y(x)$, $-1 \leq x \leq 1$, solves

$$y'' + a(x)y' + b(x)y = f(x), \quad y(-1) = y(1), \quad (3.6)$$

where $a, b, f \in \mathcal{A}$ and we seek a periodic solution $y \in \mathcal{A}$ for (3.6). Substituting y, a, b and f by their Fourier series and using (3.3)-(3.5) we obtain an infinite dimensional system of linear equations for the Fourier coefficients \hat{y}_n :

$$-\pi^2 n^2 \hat{y}_n + i\pi \sum_{m=-\infty}^{\infty} m \hat{a}_{n-m} \hat{y}_m + \sum_{m=-\infty}^{\infty} \hat{b}_{n-m} \hat{y}_m = \hat{f}_n, \quad n \in \mathbb{Z}. \quad (3.7)$$

Since $a, b, f \in \mathcal{A}$, their Fourier coefficients decrease rapidly, like $\mathcal{O}(n^{-p})$ for every $p \in \mathbb{N}$. Hence, we can truncate (3.7) into the N -dimensional system

$$-\pi^2 n^2 \hat{y}_n + i\pi \sum_{m=-N/2+1}^{N/2} m \hat{a}_{n-m} \hat{y}_m + \sum_{m=-N/2+1}^{N/2} \hat{b}_{n-m} \hat{y}_m = \hat{f}_n, \quad n = -N/2+1, \dots, N/2. \quad (3.8)$$

Remark 3.10 The matrix of (3.8) is in general dense, but our theory predicts that fairly small values of N , hence very small matrices, are sufficient for high accuracy. For instance: choosing $a(x) = f(x) = \cos \pi x$, $b(x) = \sin 2\pi x$ (which incidentally even leads to a sparse matrix) we get

$N = 16$	error of size 10^{-10}
$N = 22$	error of size 10^{-15} (which is already hitting the accuracy of computer arithmetic)

Method 3.11 (Computation of Fourier coefficients (DFT)) We have to compute

$$\widehat{f}_n = \frac{1}{2} \int_{-1}^1 f(t) e^{-i\pi n t} dt, \quad n \in \mathbb{Z}. \quad (3.9)$$

For this, suppose we wish to compute the integral on $[-1, 1]$ of a function $h \in \mathcal{A}$ by means of the Riemann sums on the uniform partition

$$\int_{-1}^1 h(t) dt \approx \frac{2}{N} \sum_{k=-N/2+1}^{N/2} h\left(\frac{2k}{N}\right). \quad (3.10)$$

This is known as a *rectangle rule*. We want to know how good this approximation is. As in Definition 1.18, let $\omega_N = e^{2\pi i/N}$. Then we have

$$\frac{2}{N} \sum_{k=-N/2+1}^{N/2} h\left(\frac{2k}{N}\right) = \frac{2}{N} \sum_{k=-N/2+1}^{N/2} \sum_{n=-\infty}^{\infty} \widehat{h}_n e^{2\pi i n k / N} = \frac{2}{N} \sum_{n=-\infty}^{\infty} \widehat{h}_n \sum_{k=-N/2+1}^{N/2} \omega_N^{nk}.$$

Since $\omega_N^N = 1$ we have

$$\sum_{k=-N/2+1}^{N/2} \omega_N^{nk} = \omega_N^{-n(N/2-1)} \sum_{k=0}^{N-1} \omega_N^{nk} = \begin{cases} N, & n \equiv 0 \pmod{N}, \\ 0, & n \not\equiv 0 \pmod{N}, \end{cases}$$

and we deduce that

$$\frac{2}{N} \sum_{k=-N/2+1}^{N/2} h\left(\frac{2k}{N}\right) = 2 \sum_{r=-\infty}^{\infty} \widehat{h}_{Nr}.$$

Hence, the error committed by the Riemann approximation is

$$\begin{aligned} e_N(h) &:= \frac{2}{N} \sum_{k=-N/2+1}^{N/2} h\left(\frac{2k}{N}\right) - \int_{-1}^1 h(t) dt = 2 \sum_{r=-\infty}^{\infty} \widehat{h}_{Nr} - 2\widehat{h}_0 \\ &= 2 \sum_{r=1}^{\infty} (\widehat{h}_{Nr} + \widehat{h}_{-Nr}). \end{aligned}$$

Since $h \in \mathcal{A}$, its Fourier coefficients decay at spectral rate, namely $\widehat{h}_{Nr} = \mathcal{O}((Nr)^{-p})$, and hence the error of the Riemann sums approximation (3.10) decays spectrally as a function of N ,

$$e_N(h) = \mathcal{O}(N^{-p}) \quad \forall p \in \mathbb{N}.$$

Going back to the computation of the Fourier coefficients (3.9), we see that we may compute the integral of $h(x) = \frac{1}{2} f(x) e^{-i\pi n x}$ by means of the Riemann sums, and this gives a spectral method for calculating the Fourier coefficients of f :

$$\widehat{f}_n \approx \frac{1}{N} \sum_{k=-N/2+1}^{N/2} f\left(\frac{2k}{N}\right) \omega_N^{-nk}, \quad n = -N/2 + 1, \dots, N/2. \quad (3.11)$$

Remark 3.12 One can recognise that formula (3.11) is the *discrete Fourier transform (DFT)* of the sequence $(y_k) = (f(\frac{2k}{N}))$, see previous definition, hence not only have we a spectral rate of convergence, but also a fast algorithm (FFT) of computing the Fourier coefficients.

Revision 3.13 (The fast Fourier transform (FFT)) The *fast Fourier transform (FFT)* is a computational algorithm, which computes the leading N Fourier coefficients of a function in just $\mathcal{O}(N \log_2 N)$ operations (cf. Algorithm 1.19). We assume that N is a power of 2, i.e. $N = 2^p$, and for $\mathbf{y} \in \Pi_{2^p}$, denote by

$$\mathbf{y}^{(E)} = \{y_{2j}\}_{j \in \mathbb{Z}} \quad \text{and} \quad \mathbf{y}^{(O)} = \{y_{2j+1}\}_{j \in \mathbb{Z}}$$

the even and odd portions of \mathbf{y} , respectively. Note that $\mathbf{y}^{(\text{E})}, \mathbf{y}^{(\text{O})} \in \Pi_m$. To execute FFT, we start from vectors of unit length and in each s -th stage, $s = 1 \dots p$, assemble 2^{p-s} vectors of length 2^s from vectors of length 2^{s-1} with

$$x_\ell = x_\ell^{(\text{E})} + \omega_{2^s}^\ell x_\ell^{(\text{O})}, \quad \ell = 0, \dots, 2^{s-1} - 1. \quad (3.12)$$

Therefore, it costs just s products to evaluate the first half of \mathbf{x} , provided that $\mathbf{x}^{(\text{E})}$ and $\mathbf{x}^{(\text{O})}$ are known. It actually costs nothing to evaluate the second half, since

$$x_{2^{s-1}+\ell} = x_\ell^{(\text{E})} - \omega_{2^s}^\ell x_\ell^{(\text{O})}, \quad \ell = 0, \dots, 2^{s-1} - 1.$$

Altogether, the cost of FFT is $p2^{p-1} = \frac{1}{2}N \log_2 N$ products.