Karlsruher Institut für Technologie

Department of Mathematics
Institute for Applied and
Numerical Mathematics

JProf. Dr. Katharina Schratz
Dipl-Math. techn. Patrick Krämer

Aspects of Numerical Time Integration — Exercise Sheet 04 July 20, 2017

The aim of this exercise sheet is to get familiar with **spectral space discretization** schemes. We will see that this schemes work very well for periodic problems. However, we have to be careful with their application to non-periodic problems which are made periodic artificially for practical implementation issues.

Motivation and definition: (cf. literature in footnote) Let $N \in \mathbb{N}$ be even and let

$$h = (2\pi)/N \qquad \text{and} \qquad x_j = jh,\ j = -N/2, \dots, N/2$$

be a discretization of the interval $[-\pi, \pi]$. Let $u\colon [-\pi, \pi] \to \mathbb{C}$ smooth and 2π-periodic on $[-\pi, \pi]$, i.e.

$$u(-\pi) = u(\pi) \qquad \text{and in particular} \qquad u(x_{-N/2}) = u(x_{N/2}).$$

Then the Fourier expansion of u reads $u(x) = \sum_{k\in\mathbb{Z}} \hat{u}_k e^{ikx}$.

The idea is now to use a **trigonometric interpolation polynomial** $t_N(x)$ to approximate $u(x)$ with interpolation property in the grid points x_j, i.e. $t_N(x_j) = u(x_j)$, $j = -N/2, \dots, N/2$.

Let $U \in \mathbb{C}^N$ with $U_j = u(x_j)$, $j = -N/2+1, \dots, N/2$. We define the following trigonometric polynomial

$$t_N(x) = \frac{1}{2N}\left(\hat{U}_{-N/2} e^{-ixN/2} + \hat{U}_{N/2} e^{ixN/2}\right) + \frac{1}{N}\sum_{k=-N/2+1}^{N/2-1} \hat{U}_k e^{ikx} \quad \text{for all } x \in [-\pi, \pi]$$

where $\hat{U} = \mathcal{F}_N U = \left(\hat{U}_k\right)_{k=-N/2+1}^{N/2}$ is the **discrete Fourier transform** of U, defined via

$$\hat{U}_k := \sum_{j=-N/2+1}^{N/2} U_j e^{-ijx_k} \quad \text{for all } k = -N/2+1, \dots, N/2.$$

<u>Exercise 6:</u>

a) Show that for all $k = -N/2, \dots, N/2$ we have that

$$\hat{U}_{-N/2} e^{-ix_k N/2} = \hat{U}_{N/2} e^{ix_k N/2}.$$

b) Show that t_N satisfies the interpolation property, i.e. show that for all $j = -N/2, \dots, N/2$

$$t_N(x_j) = u(x_j).$$

Hint: Exploit part a).

In particular Exercise 6 shows that

$$t_N(x_j) = \frac{1}{N}\sum_{k=-N/2+1}^{N/2} \hat{U}_k e^{ikx_j} =: \left(\mathcal{F}_N^{-1}\hat{U}\right)_j, \qquad j = -N/2+1, \dots, N/2$$

defines the **inverse discrete Fourier transform** $\mathcal{F}_N^{-1}\hat{U}$ of $\hat{U}$.

please turn over →

On the basis of the trigonometric polynomial t_N and the discrete Fourier transform above we now define the following **spectral differentiation methods**:

We numerically approximate the m-th spatial derivative of u, i.e. $\partial_x^m u(x_j)$, in the grid points $x_j, j = -N/2+1, \dots, N/2$ by

$$\partial_x^m t_N(x_j) = t_N^{(m)}(x_j) = \mathcal{F}_N^{-1}\left((i\tilde{k})^m \cdot \mathcal{F}_N U\right)_j, \quad j = -N/2+1, \dots, N/2, \tag{1a}$$

where (see remark in the footnote on the ordering in MATLAB)

$$\tilde{k} = [-N/2+1, \dots, N/2-1, \chi], \quad \chi = \begin{cases} N/2, & m \text{ even}, \\ 0, & m \text{ odd}. \end{cases} \tag{1b}$$

One big advantage of **spectral methods** compared to finite difference schemes (see Exercise Sheet 3) is that the numerical approximation error to spatial derivatives of u depends on the **regularity of the function** u, i.e. if $u \in H^r([-\pi, \pi])$ with $r > m$ then (in the ℓ_0^∞ norm)

$$\|\partial_x^m u(\cdot) - \partial_x^m t_N(\cdot)\|_\infty = \max_{j=-N/2+1,N/2} |\partial_x^m u(x_j) - \partial_x^m t_N(x_j)| \le CN^{-(r-m)}.$$

Similar results are obtained in the lecture in the ℓ_s^1 and ℓ_s^2 norms. Note that the finite difference scheme (see Exercise Sheet 3) for approximating $\partial_x^2 u$ only yields error bounds of order $\mathcal{O}(N^{-2})$ even if $u \in C^\infty$.

Another advantage is that if we set $N = 2^R, R \in \mathbb{N}$ the MATLAB built-in functions `fft` and `ifft` allow us to reduce the computational cost of numerically approximating the m-th derivative of u from $\mathcal{O}(N^2)$ operations (finite differences) to $\mathcal{O}(N \log N)$ operations (**spectral schemes**).

Note that the theory on spectral derivatives also works for periodic functions on arbitrary intervals $[a, b] \subset \mathbb{R}$ by transforming the interval $[-\pi, \pi]$ to the interval $[a, b]$.

Which **additional factor** do we then need in the coefficients $\tilde{k}$?

Our aim is now to numerically compare spectral differentiation methods with the method of finite differences.

Exercise 7:

Consider $f(x) := \exp(\sin(x))$ on the interval $x \in [-\pi, \pi]$ and consider $N = 16$ grid points at first.

a) In MATLAB implement the spectral space discretization scheme to compute an approximation to the second derivative $f''(x)$. Plot the numerical result together with the exact derivative.

b) Add a finite difference approximation to f'' to your plot. What do you observe?

c) Create an order plot for the spatial accuracy of the spectral and the finite difference method using $N_l = 8 \cdot 2^l,\ l = 0, \dots, 9$ grid points. Compute the corresponding errors in the approximate L^2 norm, i.e.

$$err_l = \sqrt{h_l}\,\|f''_{num} - f''_{exact}\|.$$

d) Repeat all the steps for the function $g(x) = 1/\cosh(x)$ on the interval $x \in [-\pi, \pi]$. What can you observe?

How do your results change if we consider g on $x \in [-4\pi, 4\pi]$ instead? Can you give an explanation?

Hint: $\cosh(x) = (e^x + e^{-x})/2, \quad \frac{d}{dx}\cosh(x) = \sinh(x), \quad \cosh(x)^2 = 1 + \sinh(x)^2.$

Discussion in the problem class thursday 3:45 pm, in room 3.061 in the Kollegiengebäude Mathematik 20.30.

Note that Matlab orders the Fourier numbers differently as follows $\tilde{k} = [0, \dots, N/2-1, \chi, -N/2+1, \dots, -1]$.

Literature: L. Trefethen - Spectral Methods in Matlab (2000)