The Pseudospectral Method

Heiner Igel

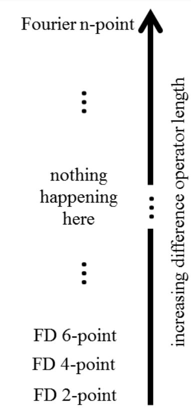
Department of Earth and Environmental Sciences
Ludwig-Maximilians-University Munich

Infinite Order Finite Differences

Infinite Order Finite Differences

- Any finite-difference operation can be decribed as a convolution operation implying that the specific finite-difference operator has a spectral representation that can be compared with the exact $-ik$ operator
- Using Taylor Series for accuracy improvement (= length of operator)
- Using Fourier concepts for calculating exact derivatives

$\Rightarrow$ Fourier method can be interpreted as an infinite order finite-difference scheme



Infinite Order Finite Differences

Let us restate the previous result of the partial derivative as an inverse Fourier transform defined as

$$\begin{aligned} \partial_x f(x) &= \frac{1}{\sqrt{2\pi}} \int_{\infty}^{\infty} \partial_x F(k) e^{-ikx} dk \\ &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} -ikF(k) e^{-ikx} dk \end{aligned}$$

Defining the factors in front of the complex amplitude spectrum $F(k)$ of function $f(x)$ as

$$\partial_x f(x) = \int_{-\infty}^{\infty} D(k) F(k) e^{-ikx} dk, \quad D(k) = -ik$$

Infinite Order Finite Differences

Two functions $d(x)$ and $f(x)$ with complex spectra $D(k)$ and $F(k)$, are thus linked by

$$
\begin{aligned}
D(k) &= \mathscr{F}[d] \\
F(k) &= \mathscr{F}[f] \\
d * f &= \mathscr{F}^{-1}[D(k)F(k)]
\end{aligned}
$$

where $\mathscr{F}$ represents the Fourier transform, and $*$ denotes convolution, defined in the continuous case as

$$(d * f)(x) := \int_{-\infty}^{\infty} d(x')f(x - x')dx'$$

and in the discrete case with vectors $d_i,\ i = 0, 1, \ldots, m$, and $f_j,\ j = 0, 1, \ldots, n$

$$(d * f)_k = \sum_{i=0}^{m} d_k f_{k-i}\ ,\ k = 0, 1, \ldots, m + n$$

Infinite Order Finite Differences

$D(k)$ in general is nothing else but a function defined in the spectral domain acting like a filter on the complex spectrum $F(k)$.
The convolution theorem implies that

$$\partial_x f(x) \; = \int_{-\infty}^{\infty} d(x - x') f(x') dx'$$

where d(x) is a real function, the spatial representation of spectrum $D(k)$, in other words

$$d(x) \; = \mathscr{F}^{-1}[D(k)] \; .$$

Infinite Order Finite Differences

Limiting the wavenumber domain to the Nyquist wavenumber $k_{max} = \pi/dx$. Thus $D(k)$ becomes

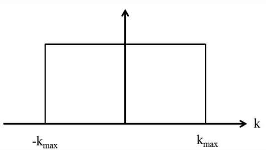
$$D(k) = ik[H(k + k_{max}) - H(k_{kmax})]$$

where $H()$ denotes the Heaviside function, and to obtain $d(x)$ we simply have to inverse transform

$$d(x) = \mathscr{F}^{-1}[ik[H(k + k_{max}) - H(k - k_{kmax})]]$$

leading to

$$d(x) = \frac{1}{\pi x^2}[sin(k_{max}x) - k_{max}x\cos(k_{max}x)]$$



Infinite Order Finite Differences

The r.h.s. of the Fourier integral is a multiplication of two spectra:

- derivative operator ik
- boxcar function and its solution is the *sinc* function of the form $\sin(x)/x$

If space is discretized according to

$$x_n = n\,dx, \quad n = -N, \dots, 0, \dots, N$$

In this case the convolution integral becomes a convolution sum

$$\partial_x f(x) \approx \sum_{n=-N}^{n=N} d_n f(x - ndx)$$

where d_n is the difference operator.

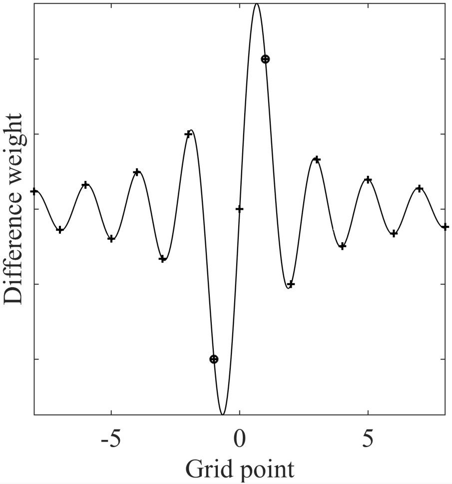
Infinite Order Finite Differences

Inserting the discretization into

$$
\begin{aligned}
d(x) &= \frac{1}{\pi x^2}[sin(k_{max}x) \\
&\quad - k_{max}x\cos(k_{max}x)]
\end{aligned}
$$

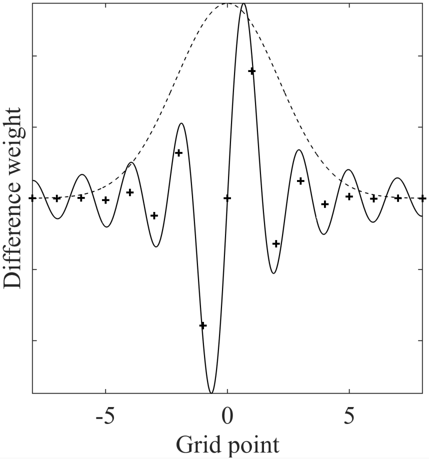
we obtain analytically the discrete difference operator

$$
d_n = \begin{cases} 0 & \text{for } n = 0 \\ \frac{(-1)^n}{ndx} & \text{for } n \neq 0 \,. \end{cases}
$$

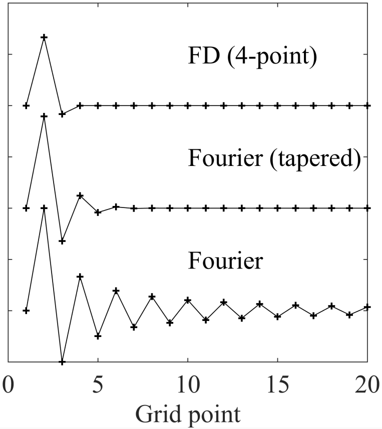


Infinite Order Finite Differences

Truncated Fourier operator



Convolutional difference operators



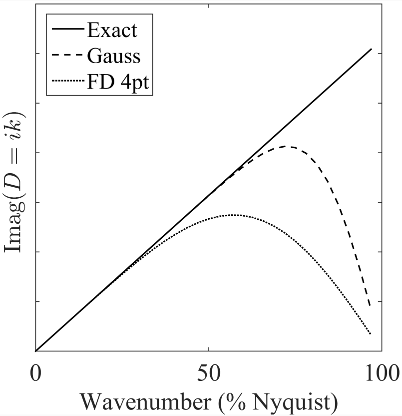
Infinite Order Finite Differences

How can we conveniently compare the accuracy of such operators?

$\Longrightarrow$ The space representation of the exact difference operator $D(k) = -ik$ in the wavenumber domain

Thus, for a finite-difference operator d_n^{FD} we will obtain

$$D^{FD}(k) = i\,\tilde{k}_\nu(k) = \mathscr{F}[d_n^{FD}]$$



Chebyshev Pseudospectral Method

Chebyshev Polynomials

Let us start with the trigonometric relation

$$\cos\left[(n+1)\phi\right] + \cos\left[(n-1)\phi\right] \;=\; 2\cos(\phi)\cos(n\phi) \qquad n \in \mathbb{N}\,.$$

Inserting $n = 0$ leads to a trivial statement. However, for $n \geq 1$ we obtain statements like

$$\begin{aligned}
\cos(2\phi) &= 2\cos^2(\phi) - 1 \\
\cos(3\phi) &= 4\cos^3(\phi) - 3\cos(\phi) \\
\cos(4\phi) &= 8\cos^4(\phi) - 8\cos^2(\phi) + 1 \\
&\;\;\vdots
\end{aligned}$$

Chebyshev Polynomials

$$\cos(n\phi) \;=:\; T_n(\cos(\phi)) \;=\; T_n(x)$$

with

$$x \;=\; \cos(\phi) \qquad x \in [-1, 1]\ , \qquad n \in \mathbb{N}_0$$

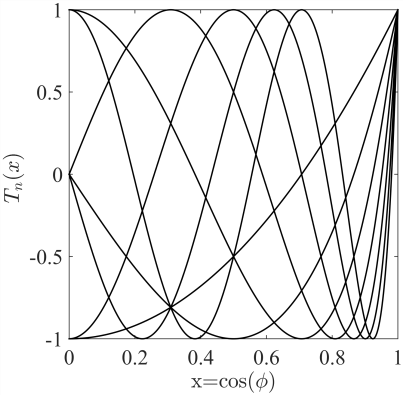
T_n being the n-th order Chebyshev polynomial. Furthermore

$$|T_n(x)| \leqslant 1 \text{ for } [-1, 1]\,, \qquad n \in \mathbb{N}_0$$

Chebyshev Polynomials

Finally, we can write down the first polynomials in $x \in [-1, 1]$

$$
\begin{aligned}
T_0(x) &= 1 \\
T_1(x) &= x \\
T_2(x) &= 2x^2 - 1 \\
T_3(x) &= 4x^3 - 3x \\
T_4(x) &= 8x^4 - 8x^2 - 1 \\
&\vdots
\end{aligned}
$$



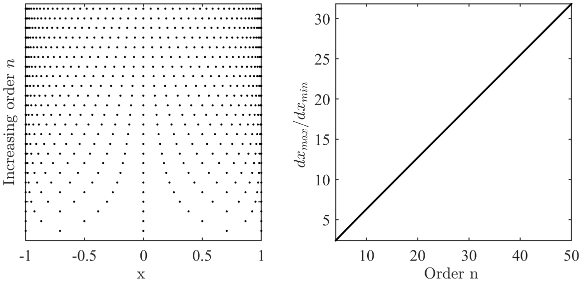
Chebyshev Polynomials

A generating function calculates the Chebyshev polynomials of any order n

$$T_{n+1}(x) \;=\; 2xT_n(x) \;-\; T_{n-1}(x), \quad n \geq 1$$

The extremal values $x_k^{(e)}$ of these polynomials have a very simple form

$$x_k^{(e)} \;=\; \cos(\frac{k\pi}{n}) \qquad k \;= 0, 1, 2, \ldots, n$$



Chebyshev Polynomials

The Chebyshev polynomials form an orthogonal set with respect to the weighting function $w(x) = 1/\sqrt{1-x^2}$.
$\Rightarrow$ Using them as a basis for function interpolation

$$f(x) \;\approx\; g_n(x) \;=\; \frac{1}{2}c_0 T_0(x) + \sum_{k=1}^{n} c_k T_k(x)$$

where f(x) is an arbitrary function in the interval $[-1, 1]$, $T_n(x)$ are the Chebyshev polynomials, and c_k are real coefficients.

Chebyshev Polynomials

By minimizing the least-squares misfit between $f(x)$ and $g_n(x)$, the coefficients c_k can be found

$$c_k = \frac{2}{\pi} \int_{-1}^{1} f(x) T_k(x) \frac{\mathrm{d}x}{\sqrt{1-x^2}} \qquad k = 0, 1, \ldots, n$$

which - after substituting $x = \cos(\phi)$ - can be written as

$$c_k = \frac{1}{\pi} \int_{-\pi}^{\pi} f(\cos(\phi)) \cos(k\phi) \mathrm{d}\phi \qquad k = 0, 1, \ldots, n$$

These coefficients turn out to be the Fourier coefficients for the even $2\pi-$periodic function $f(\cos(\phi))$ with $x = \cos(\phi)$.

Chebyshev Polynomials

The points we need are the extrema of the Chebyshev polynomials (Chebyshev-) Gauss-Lobatto points defined as

$$x_i \ = \ \cos(\frac{\pi}{N} i) \qquad i = 0, 1, \ldots, n \, .$$

With these unevenly distributed grid points we can define the discrete Chebyshev transform as follows. The approximating function is

$$g_n^*(x) \ = \ \frac{1}{2} c_0^* T_0 + \sum_{k=1}^{n-1} c_k^* T_k(x) + \frac{1}{2} c_n^* T_n$$

Chebyshev Polynomials

with the coefficients defined as

$$c_k^* = \frac{2}{m}\left[\frac{1}{2}(f(1) + (-1)^k f(-1)) + \sum_{j=1}^{m-1} f_j \cos(\frac{kj\pi}{m})\right]$$

$$k = 0, 1, \ldots, n, \qquad n = m$$

where $f(1)$ and $f(-1)$ are the function values at the interval boundaries and f_j are the values at the collocation points $f(x = \cos(j\pi/m))$. The fundamental property is

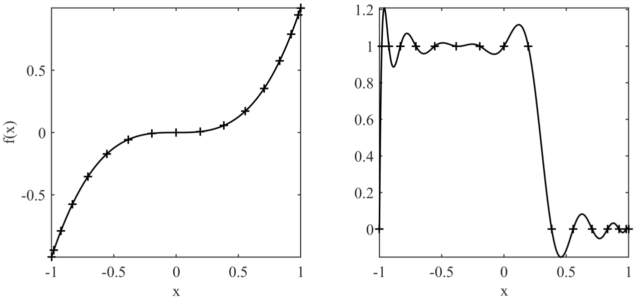
$$g_m^*(x_i) = f(x_i)$$

where x_i are the collocation points.

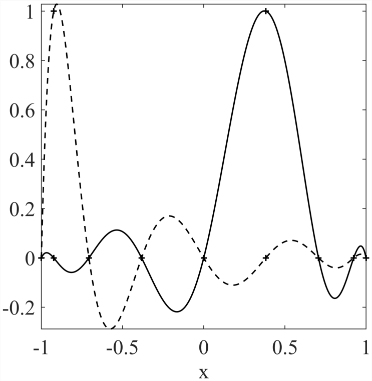
Example

When we have a function $f(x) = x^3$ in the interval $[-1, 1]$ using the Chebyshev transform, the function $f(x)$

1. can be exactly interpolated at the collocation points
2. converges very rapidly with just a few polynomials



Chebyshev polynomials



Two cardinal functions with Chebyshev polynomials for grid points $i = n/2$ (solid line) and $i = n$ (dashed line) are shown for $n = 8$

Chebyshev Derivatives, Differentiation matrices

Chebyshev Derivatives, Differentiation matrices

A convolution operation can be formulated as a matrix-vector product involving Toeplitz matrices. Defining a derivative matrix D_{ij}

$$D_{ij} = \begin{cases} -\frac{2N^2+1}{6} & \text{for } i = j = N \\ -\frac{1}{2}\frac{x_i}{1-x_i^2} & \text{for } i = j = 1,2,\ldots,N-1 \\ \frac{c_i}{c_j}\frac{(-1)^{i+j}}{x_i-x_j} & \text{for } i \neq j = 0,1,\ldots,N \end{cases}$$

where N+1 is the number of Chebyshev collocation points $x_i = \cos(i\pi/N)$, $i = 0, \ldots, N$ and the c_i are given as

$$c_i = \begin{cases} 2 & \text{for } i=0 \text{ or } N \\ 1 & \text{otherwise} \end{cases}$$

Chebyshev Derivatives, Differentiation matrices

This differentiation matrix allows us to write the derivative of function $u_i = u(x_i)$ simply as

$$\partial_x u_i \;=\; D_{ij}\, u_j$$

where the right-hand side is a matrix-vector product, and the Einstein summation convention applies.

Chebyshev Derivatives, Differentiation matrices

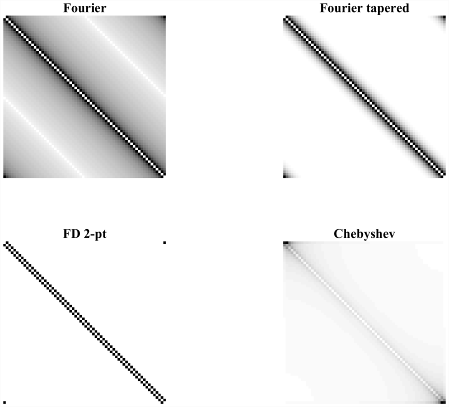
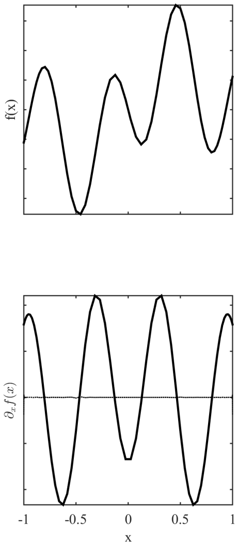


Illustration of differentiation matrices (n=64). **Top left:** Exact Fourier differentiation matrix for regular grid (full). **Top right:** Exact Chebyshev differentiation matrix for Chebyshev collocation points. Increasing weights at the corners overshadows interior values. **Bottom Left:** Standard 2-point finite difference operator (banded). **Bottom Right:** Tapered Fourier operator (12-point). Matrix is banded. For illustration purposes the square root of the absolute values are shown.

Chebyshev Derivatives, Differentiation matrices



By testing the differentiation, we define a function to seismic wavefield calculations as

$$f(x_i) = \sin(2x_i) - \sin(3x_i) + \sin(4x_i) - \sin(10x_i)$$

in the interval $x_i \in [-1, 1]$, where the discrete points are the Chebyshev collocation points $x_i = \cos(\pi i/n),\ i = 0, \ldots, n$ given for $n = 63$.

Elastic 1D with Chebyshev Method

Elastic 1D with Chebyshev Method

Elastic 1D wave equation using the standard 3-point operator

$$\rho_i \frac{u_i^{j+1} - 2u_i^j + u_i^{j-1}}{\mathrm{d}t^2} = \left(\partial_x \left[\mu(x)\partial_x u(x,t)\right]\right)_i^j + f_i^j$$

where the lower index i corresponds to the spatial discretization and the upper index j to the discrete time levels.
The displacement field as well as the geophysical parameters like density ρ_i and shear modulus μ_i are defined on the irregular Chebyshev collocation points.

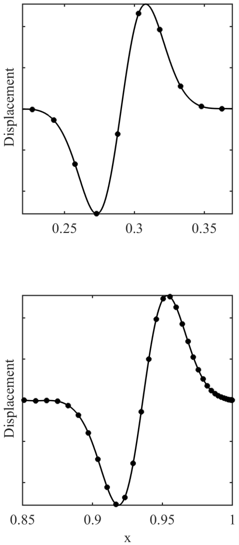
Example

- Distance between grid points is 80 times smaller at the boundaries
- The time step for a stable simulation requires $cdt/dx \leq \epsilon$

$\Rightarrow$ Grid distance near boundary is responsible for the global simulation time step

Parameter	Value
nx	200
c	3000 m/s
ρ	2500 kg/m^3
dt	6×10^{-8} s
dx_{min}	1.2×10^{-4} m
dx_{max}	0.015 m
f_0	100 kHz
ϵ	1.4

Result




```python
# ---------------------------------------------------------------
# Time extrapolation
# ---------------------------------------------------------------
# Differentiation matrix
D = get_cheby_matrix(nx)
for it in range(nt):
    # Space derivatives
    dp = np.dot(D, p.T)
    dp = mu/rho * dp
    dp = D @ dp

    # Time extrapolation
    pnew = 2*p - pold + np.transpose(dp) * dt**2

    # Source injection
    pnew = pnew + sg*src[it]*dt**2/rho

    # Remapping
    pold, p = p, pnew
    p[0] = 0; p[nx] = 0 # set boundaries pressure free
```

Result

- To obtain a stable solution we need a very small time step that is only needed at the boundaries. Mathematically the time step scales with $O(N^{-2})$.
- In principle we can stretch the spatial grid such that the grid points close to the boundaries are further apart while the grid point distances at the centre remain basically unchanged. If that stretching function is $\xi(x)$ then the derivative of a function $f(x)$ on the stretched grid is defined as

$$\partial_x f(x) = \frac{\partial f}{\partial \xi}\frac{d\xi}{dx} .$$

Summary

- Pseudospectral methods are based on discrete function approximations that allow exact interpolation at so-called collocation points. The most prominent examples are the Fourier method based on trigonometric basis functions and the Chebyshev method based on Chebyshev polynomials.
- The Fourier method can be interpreted as an application of discrete Fourier series on a regular-spaced grid. The space derivatives can be obtained exactly (except for rounding errors). Derivatives can be efficiently calculated with the discrete Fourier transform requiring $n \log n$ operations.
- The Fourier method implicitly assumes periodic behavior. Boundary conditions like the free surface or absorbing behaviour are difficult to implement.
- The Chebyshev method is based on the description of spatial fields using Chebyshev polynomials defined in the interval $[-1, 1]$ (easily generalized to arbitrary domain sizes). Exact interpolation is possible when the discrete fields are defined at the Chebyshev collocation points given by $x_i = \cos(\pi i/n), \; i = 0, \ldots, n$. Therefore, the derivatives can also be evaluated exactly and errors accumulate only due to the finite-difference time extrapolation.

Summary

- Because of the grid densification at the boundaries of the Chebyshev collocation points very small time steps are required for stable simulations when n is large. This can be avoided by stretching the grids by a coordinate transformation.
- A main advantage of the Chebyshev method is an elegant formulation of boundary conditions (free surface or absorbing) through the definition of so-called characteristic variables.
- Pseudospectral methods have isotropic errors. Therefore they lend themselves to the study of physical anisotropy.
- The derivative operations of pseudospectral methods are of a global nature. That means every point on a spatial grid contributes to the gradient. While this is the basis for the high precision, it creates problems when implementing pseudospectral algorithms on parallel computers with distributed memory architectures. As communication is usually the bottleneck, efficient and scalable parallelization of pseudospectral methods is difficult.