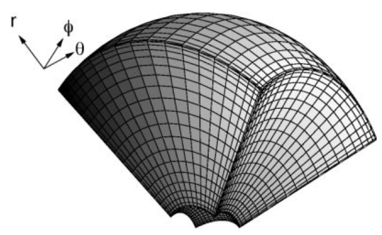
The Pseudospectral Method

Heiner Igel

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

Introduction

Motivation



1. Orthogonal basis functions, special case of FD
2. Spectral accuracy of space derivatives
3. High memory efficiency
4. Explicit method
5. No requirement of grid staggering
6. Problems with strongly heterogeneous media

History

- Coining as transform methods as their implementation was based on the Fourier transform (Gazdag, 1981; Kossloff and Bayssal, 1982)
- Initial applications to the acoustic wave equation were extended to the elastic case (Kossloff et al., 1984), and to 3D (Reshef et al., 1988)
- Developing efficient time integration schemes (Tal-Ezer et al., 1987) that allowed large times steps to be used in the extrapolation procedure
- Replacing harmonic functions as bases for the function interpolation by Chebyshev poly-nomials (Kosloff et al., 1990)
- To improve the accurate modelling of curved internal interfaces and surface topography grid stretching as coordinate transforms was introduced and applied (Tessmer et al., 1992; Komatitsch et al., 1996)
- By mixing finite-difference operators and pseudospectral operators in the different spatial directions, the method was used for interesting seismological problems (Furumura et al., 1998b; Furumura and Kennett, 2005)

History

- Coining as transform methods as their implementation was based on the Fourier transform (Gazdag, 1981; Kossloff and Bayssal, 1982)
- Initial applications to the acoustic wave equation were extended to the elastic case (Kossloff et al., 1984), and to 3D (Reshef et al., 1988)
- Developing efficient time integration schemes (Tal-Ezer et al., 1987) that allowed large times steps to be used in the extrapolation procedure
- Replacing harmonic functions as bases for the function interpolation by Chebyshev poly-nomials (Kosloff et al., 1990)
- To improve the accurate modelling of curved internal interfaces and surface topography grid stretching as coordinate transforms was introduced and applied (Tessmer et al., 1992; Komatitsch et al., 1996)
- By mixing finite-difference operators and pseudospectral operators in the different spatial directions, the method was used for interesting seismological problems (Furumura et al., 1998b; Furumura and Kennett, 2005)

History

- Coining as transform methods as their implementation was based on the Fourier transform (Gazdag, 1981; Kossloff and Bayssal, 1982)
- Initial applications to the acoustic wave equation were extended to the elastic case (Kossloff et al., 1984), and to 3D (Reshef et al., 1988)
- Developing efficient time integration schemes (Tal-Ezer et al., 1987) that allowed large times steps to be used in the extrapolation procedure
- Replacing harmonic functions as bases for the function interpolation by Chebyshev poly-nomials (Kosloff et al., 1990)
- To improve the accurate modelling of curved internal interfaces and surface topography grid stretching as coordinate transforms was introduced and applied (Tessmer et al., 1992; Komatitsch et al., 1996)
- By mixing finite-difference operators and pseudospectral operators in the different spatial directions, the method was used for interesting seismological problems (Furumura et al., 1998b; Furumura and Kennett, 2005)

History

- Coining as transform methods as their implementation was based on the Fourier transform (Gazdag, 1981; Kossloff and Bayssal, 1982)
- Initial applications to the acoustic wave equation were extended to the elastic case (Kossloff et al., 1984), and to 3D (Reshef et al., 1988)
- Developing efficient time integration schemes (Tal-Ezer et al., 1987) that allowed large times steps to be used in the extrapolation procedure
- Replacing harmonic functions as bases for the function interpolation by Chebyshev poly-nomials (Kosloff et al., 1990)
- To improve the accurate modelling of curved internal interfaces and surface topography grid stretching as coordinate transforms was introduced and applied (Tessmer et al., 1992; Komatitsch et al., 1996)
- By mixing finite-difference operators and pseudospectral operators in the different spatial directions, the method was used for interesting seismological problems (Furumura et al., 1998b; Furumura and Kennett, 2005)

History

- Coining as transform methods as their implementation was based on the Fourier transform (Gazdag, 1981; Kossloff and Bayssal, 1982)
- Initial applications to the acoustic wave equation were extended to the elastic case (Kossloff et al., 1984), and to 3D (Reshef et al., 1988)
- Developing efficient time integration schemes (Tal-Ezer et al., 1987) that allowed large times steps to be used in the extrapolation procedure
- Replacing harmonic functions as bases for the function interpolation by Chebyshev poly-nomials (Kosloff et al., 1990)
- To improve the accurate modelling of curved internal interfaces and surface topography grid stretching as coordinate transforms was introduced and applied (Tessmer et al., 1992; Komatitsch et al., 1996)
- By mixing finite-difference operators and pseudospectral operators in the different spatial directions, the method was used for interesting seismological problems (Furumura et al., 1998b; Furumura and Kennett, 2005)

History

- Coining as transform methods as their implementation was based on the Fourier transform (Gazdag, 1981; Kossloff and Bayssal, 1982)
- Initial applications to the acoustic wave equation were extended to the elastic case (Kossloff et al., 1984), and to 3D (Reshef et al., 1988)
- Developing efficient time integration schemes (Tal-Ezer et al., 1987) that allowed large times steps to be used in the extrapolation procedure
- Replacing harmonic functions as bases for the function interpolation by Chebyshev poly-nomials (Kosloff et al., 1990)
- To improve the accurate modelling of curved internal interfaces and surface topography grid stretching as coordinate transforms was introduced and applied (Tessmer et al., 1992; Komatitsch et al., 1996)
- By mixing finite-difference operators and pseudospectral operators in the different spatial directions, the method was used for interesting seismological problems (Furumura et al., 1998b; Furumura and Kennett, 2005)

The Pseudospectral Method in a Nutshell

The Pseudospectral method is:

- a grid point method
- a series expansion method (Fourier or Chebyshev)

Looking at the acoustic wave equation using finite-difference method leaves us with

$$\frac{p(x, t + \mathrm{d}t) - 2p(x, t) + p(x, t - \mathrm{d}t)}{\mathrm{d}t^2} = c(x)^2 \partial_x^2 p(x, t) + s(x, t)$$

The Pseudospectral Method in a Nutshell

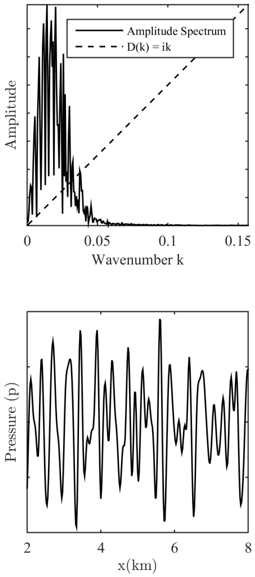
The remaining task is to calculate the space derivative on the r.h.s.

$$\partial_x^{(n)} p(x,t) \;=\; \mathscr{F}^{-1}[(ik)^n P(k,t)]$$

where i is the imaginary unit, $\mathscr{F}^{-1}$ is the inverse Fourier transform, and $P(k,t)$ is the spatial Fourier transform of the pressure field $p(x,t)$, k being the wavenumber.

Using discrete Fourier transform of functions defined on a regular grid, we obtain exact derivatives up to the Nyquist wavenumber $k_N = \pi/dx$.

The Pseudospectral Method in a Nutshell



Principle of the pseudospectral method based on the Fourier series

- Use of sine and cosine functions for the expansions implies periodicity
- Using Chebyshev polynomials similar accuracy of common boundary conditions (free surface, absorbing) can be achieved

The Pseudospectral Method: Ingredients

In many situations we either...

- seek to approximate a known analytic function by a polynomial representation
- know a function only at a discrete set of points and we would like to interpolate in between those points

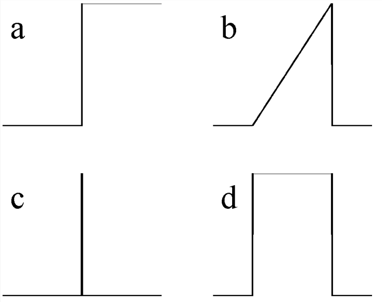
Let us start with the first problem such that our known function is approximated by a finite sum over some N basis functions Φ_i

$$f(x) \approx g_N(x) = \sum_{i=1}^{N} a_i \Phi_i(x)$$

and assume that the basis functions form an orthogonal set

Orthogonal Functions, Interpolation, Derivative

Why would one want to replace a known function by something else?



Dynamic phenomena are mostly expressed by PDEs

- Either nature is not smooth and differentiable
- mathematical functions are non-differentiable

Orthogonal Functions, Interpolation, Derivative

With the right choice of differentiable basis functions Φ_i the calculation becomes

$$\partial_x f(x) \approx \partial_x g_N(x) = \sum_{i=1}^{N} a_i \partial_x \Phi_i(x)$$

Consider the set of (trigonometric) basis functions

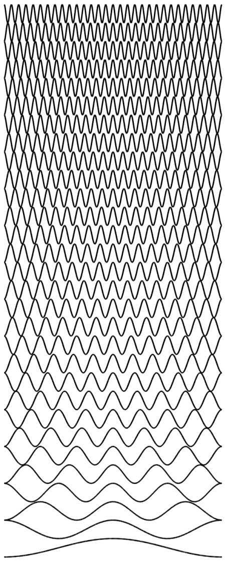
$$\begin{aligned} \cos(nx) \qquad n &= 0, 1, \ldots, \infty \\ \sin(nx) \qquad n &= 1, 2, \ldots, \infty \end{aligned}$$

with

$$1, \cos(x), \cos(2x), \cos(3x), \ldots$$
$$\sin(x), \sin(2x), \sin(3x), \ldots$$

in the interval $[-\pi, \pi]$

Orthogonal Functions, Interpolation, Derivative



Checking whether these functions are orthogonal by evaluating integrals with all possible combinations

$$\int_{-\pi}^{\pi} \cos(jx)\cos(kx)\mathrm{d}x = \begin{cases} 0 & \text{for } j \neq k \\ 2\pi & \text{for } j = k = 0 \\ \pi & \text{for } j = k > 0 \end{cases}$$

$$\int_{-\pi}^{\pi} \sin(jx)\sin(kx)\mathrm{d}x = \begin{cases} 0 & \text{for } j \neq k\,;\ j,k > 0 \\ \pi & \text{for } j = k > 0 \end{cases}$$

$$\int_{-\pi}^{\pi} \cos(jx)\sin(kx)\mathrm{d}x = 0 \text{ for } j \geqslant 0\,,\ k > 0$$

Orthogonal Functions, Interpolation, Derivative

The approximate function $g_N(x)$ can be stated as

$$f(x) \approx g_N(x) = \sum_{k=0}^{N} a_k \cos(kx) + b_k \sin(kx)$$

By minimizing the difference between approximation $g_N(x)$ and the original function $f(x)$, the so-called l_2-norm, the coefficients a_k, b_k can be found

$$\|f(x) - g_N(x)\|_{l_2} = \left[\int_a^b \{f(x) - g_N(x)\}^2 \,\mathrm{d}x \right]^{\frac{1}{2}} = \text{Min}$$

$\Longrightarrow$ independent of the choice of basis functions

The most important concept of this section will consist of the properties of Fourier series on regular grids.

The approximate function $g_N(x)$ has the following form

$$g_N(x) \;=\; \frac{1}{2}a_0 + \sum_{k=1}^{n} a_k \cos(kx) + b_k \sin(kx)$$

and leads to the coefficients

$$\begin{aligned} a_k &= \frac{1}{\pi}\int_{-\pi}^{\pi} f(x)\cos(kx)\mathrm{d}x \qquad & k &= 0, 1, \ldots, n \\ b_k &= \frac{1}{\pi}\int_{-\pi}^{\pi} f(x)\sin(kx)\mathrm{d}x \qquad & k &= 1, 2, \ldots, n \;. \end{aligned}$$

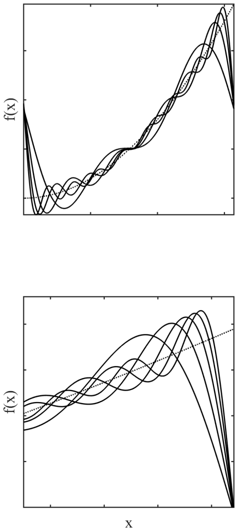
Using Euler's formulae, yields

$$g_N(x) = \sum_{k=-n}^{k=n} c_k e^{ikx}$$

with complex coefficients c_k given by

$$\begin{aligned} c_k &= \frac{1}{2}\left(a_k - ib_k\right) \\ c_{-k} &= \frac{1}{2}\left(a_k + ib_k\right) \quad k > 0 \\ c_0 &= \frac{1}{2}a_0 \ . \end{aligned}$$

Fourier Series and Transforms



Finding the interpolating trigonometric polynomial for the periodic function

$$f(x + 2\pi x) \;=\; f(x) \;=\; x^2 \qquad x \in [0, 2\pi]$$

The approximation $g_N(x)$ can be obtained with

$$g_N(x) \;=\; \frac{4\pi^2}{3} + \sum_{k=1}^{N} \left\{ \frac{4}{k^2} \cos(kx) - \frac{4\pi}{k} \sin(kx) \right\}$$

Fourier Series and Transforms

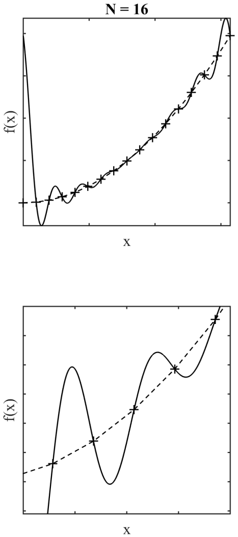
We assume that we know our function f(x) at a discrete set of points x_i given by

$$x_i = \frac{2\pi}{N} i \quad i = 0, \ldots, N \ .$$

Using the "trapezoidal rule" for the integration of a definite integral we obtain for the Fourier coefficients

$$\begin{aligned}
a_k^* &= \frac{2}{N} \sum_{j=1}^{N} f(x_j) \cos(k x_j) & k &= 0, 1, \ldots, n \\
b_k^* &= \frac{2}{N} \sum_{j=1}^{N} f(x_j) \sin(k x_j) & k &= 1, 2, \ldots, n
\end{aligned}$$

Fourier Series and Transforms



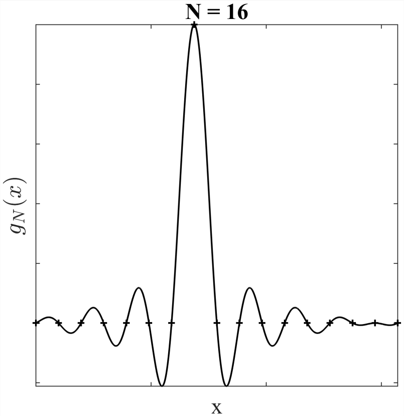
We thus obtain the specific Fourier polynomial with $N = 2n$

$$g_n^* := \frac{1}{2}a_0^* + \sum_{k=1}^{n-1}\{a_k^* \cos(kx) - b_k^* \sin(kx)\} + \frac{1}{2}a_n^* cos(nx)$$

with the tremendously important property

$$g_n^*(x_i) = f(x_i) .$$

Cardinal functions



Approximate unity value at grid point x_i and zero at all other points

Analytical solution is a *sinc*-function

This is called a **Cardinal function**

Fourier Series and Transforms

Forward Transform

$$F(k) = \mathscr{F}[f(x)] = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(x) e^{-ikx} dx$$

Inverse Transform

$$f(x) = \mathscr{F}^{-1}[F(k)] = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} F(k) e^{ikx} dk$$

Fourier Series and Transforms

Taking the formulation of the inverse transform to obtain the derivative of function f(x)

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

with $D(k) = ik$

We can extend this formulation to the calculation of the $n-th$ derivative of $f(x)$ to achieve

$$F^{(n)}(k) = D(k)^n\ F(k) = (ik)^n\ F(k)$$

Fourier Series and Transforms

Thus using the condense Fourier transform operator $\mathscr{F}$ we can obtain an exact $n - th$ derivative using

$$\begin{aligned} f^{(n)}(x) &= \mathscr{F}^{-1}[(ik)^n \, F(k)] \\ &= \mathscr{F}^{-1}[(ik)^n \, \mathscr{F}[f(x)]] \, . \end{aligned}$$

Adopting the complex notation of the forward transform we get

$$F_k = \sum_{j=0}^{N-1} f_j \, e^{-i \, 2\pi jk/N} \quad k = 0, \ldots, N$$

and the inverse transform

$$f_j = \frac{1}{N} \sum_{k=0}^{N-1} F_k \, e^{i \, 2\pi jk/N} \quad j = 0, \ldots, N$$

Fourier Series and Transforms

We are able to get exact $n-th$ derivatives on our regular grid by performing the following operations on vector f_j defined at grid points x_j

$$\partial_x^{(n)} f_j \; = \mathscr{F}^{-1}[(ik)^n \; F_k]$$

where

$$F_k = \mathscr{F}[f_j]$$

Example

We initialize a 2π-periodic Gauss-function in the interval $x \in [0, 2\pi]$ as

$$f(x) \;=\; e^{-1/\sigma^2\,(x-x_0)^2}$$

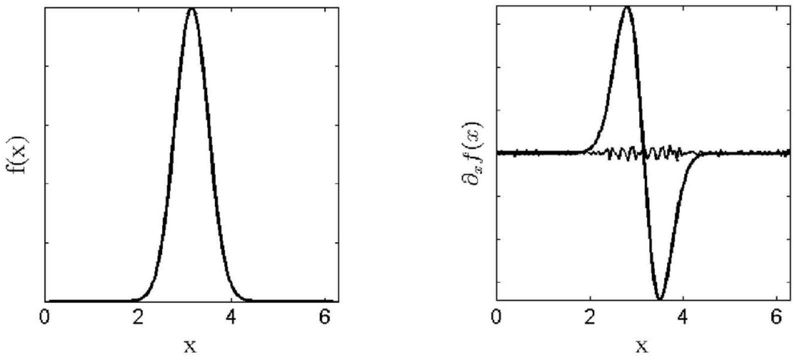
with $x_0 = \pi$ and the derivative

$$f'(x) \;=\; -2\frac{(x-x_0)}{\sigma^2}\,e^{-1/\sigma^2\,(x-x_0)^2}$$

Grid spacing of $dx = \frac{2\pi}{N}$ with $N = 127$ and $x_j = j\frac{2\pi}{N},\ j = 0, \ldots, N$.

```python
# [...]
# Basic parameters
nx = 128
x0 = pi
def fourier_derivative(f, dx):
    # Length of vector f
    nx = f.size
    # Initialize k vector up to Nyquist wavenumber
    kmax = pi/dx
    dk = kmax/(nx/2)
    k = arange(float(nx))
    k[:nx/2] = k[:nx/2] * dk
    k[nx/2:] = k[:nx/2] - kmax
    # Fourier derivative
    ff = 1j*k*fft(f)
    df = ifft(ff).real
    return df
# [...]
# Main program
# Initialize space and Gauss function (also return dx)
x = linspace(2*pi/nx, 2*pi, nx)
dx = x[1]-x[0]
sigma = 0.5
f = exp(-1/sigma**2 * (x - x0)**2)
# Calculate derivative of vector f
df = fourier_derivative(f, dx)
# [...]
```

Result



The error on the right figure was multiplied by a factor 10^{13} !

Constant-density acoustic wave equation in 1D

$$\ddot{p} \;=\; c^2 \partial_x^2 p + s$$

The time-dependent part is solved using a standard 3-point finite-difference operator leading to

$$\frac{p_j^{n+1} - 2p_j^n + p_j^{n-1}}{\mathrm{d}t^2} \;=\; c_j^2 \partial_x^2 p_j^n + s_j^n$$

where upper indices represent time and lower indices space.

Calculating the 2nd derivatives using the Fourier transform

$$\begin{aligned}\partial_x^2 p_j^n &= \mathscr{F}^{-1}[(ik)^2\ P_\nu^n] \\ &= \mathscr{F}^{-1}[-k^2\ P_\nu^n]\end{aligned}$$

where P_ν^n is the discrete complex wavenumber spectrum at time n leading to an exact derivative with only numerical rounding errors.

```python
# [...]
# Fourier derivative
def fourier_derivative_2nd(f, dx):
    # [...]
    # Fourier derivative
    ff = (1j * k)**2 * fft(f)
    df = ifft(ff).real
    return df
# [...]
# Time extrapolation
for it in range(nt):
    # 2nd space derivative
    d2p = fourier_derivative_2nd(p, dx)
    # Extrapolation
    pnew = 2 * p - pold + c**2 * dt**2 * d2p
    # Add sources
    pnew = pnew + sg * src[it] * dt**2
    # Remap pressure field
    pold, p = p, pnew
# [...]
```

Example

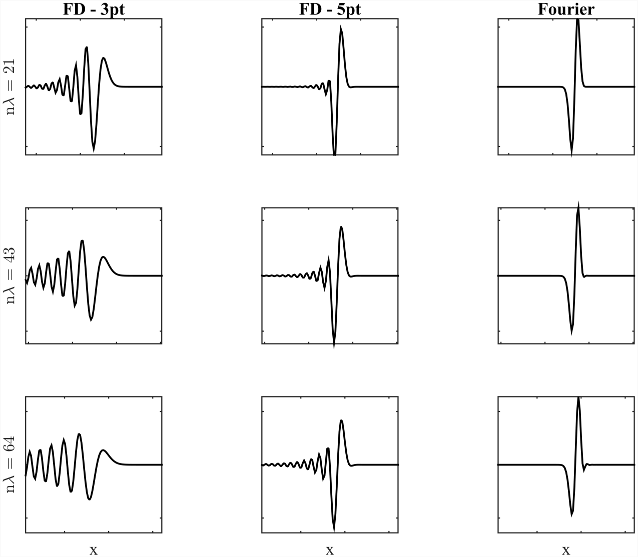
In FD method possible to initiate a point-like source at one grid point

In PS method not possible because Fourier transform of a spike-like function creates oscillations

$\Longrightarrow$ Defining a space-dependent part of the source using a Gaussian function $e^{-1/\sigma^2(x-x_0)^2}$ with $\sigma = 2dx$, dx being the grid interval and x_0 the source location

Parameter	Value
x_{max}	1250 m
nx	2048
c	343 m/s
dt	0.00036 s
dx	0.62 m
f_0	60 Hz
ϵ	0.2

Result



Stability, Convergence, Dispersion

To understand the behaviour of numerical approximations using discrete plane waves of the form

$$\begin{aligned} p_j^n &= e^{i(kj\mathrm{d}x-\omega n\mathrm{d}t)} \\ \partial_x^2 p_j^n &= -k^2 e^{i(kj\mathrm{d}x-\omega n\mathrm{d}t)} \end{aligned}$$

The time-dependent part can be expressed as

$$\partial_t^2 p_j^n = -\frac{4}{dt^2}\sin^2(\frac{\omega dt}{2})\, e^{i(kj\mathrm{d}x-\omega n\mathrm{d}t)}$$

where we made use of Euler's formula and that $2\sin^2 x = 1/2(1-\cos 2x)$

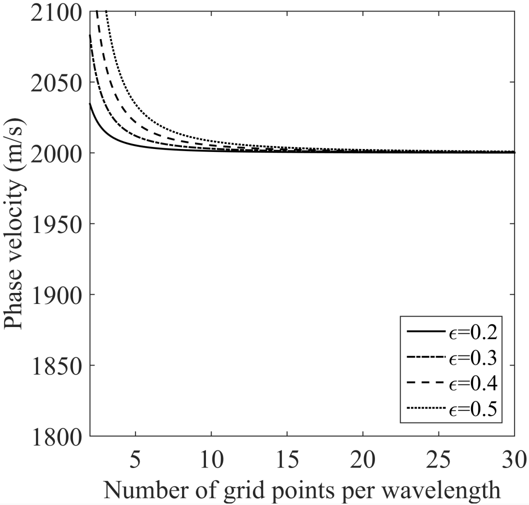
Stability, Convergence, Dispersion

Phase velocity c(k)

$$c(k) = \frac{\omega}{k} = \frac{2}{kdt} \sin^{-1}(\frac{kcdt}{2}) \; .$$

- When dt becomes small $\sin^{-1}(kcdt/2) \approx kcdt/2$
- dx does not appear in this equation
- The inverse sine must be smaller than one the stability limit requires $k_{max}(cdt/2) \leq 1$. As $k_{max} = \pi/dx$ the stability criterion for the 1D case is $\epsilon = cdt/dx = 2/\pi \approx 0.64$

Stability, Convergence, Dispersion



Acoustic 2D

Acoustic wave equation in 2D

$$\ddot{p} \;=\; c^2(\partial_x^2 p + \partial_z^2 p) \;+ s$$

The time-dependent part is replaced by a standard 3-point finite-difference approximation

$$\frac{p_{j,k}^{n+1} - 2p_{j,k}^{n} + p_{j,k}^{n-1}}{\mathrm{d}t^2} \;=\; c_{j,k}^2(\partial_x^2 p + \partial_z^2 p)_{j,k} \;+ s_{j,k}^n$$

Using Fourier approach for approximating 2^{nd} partial derivatives

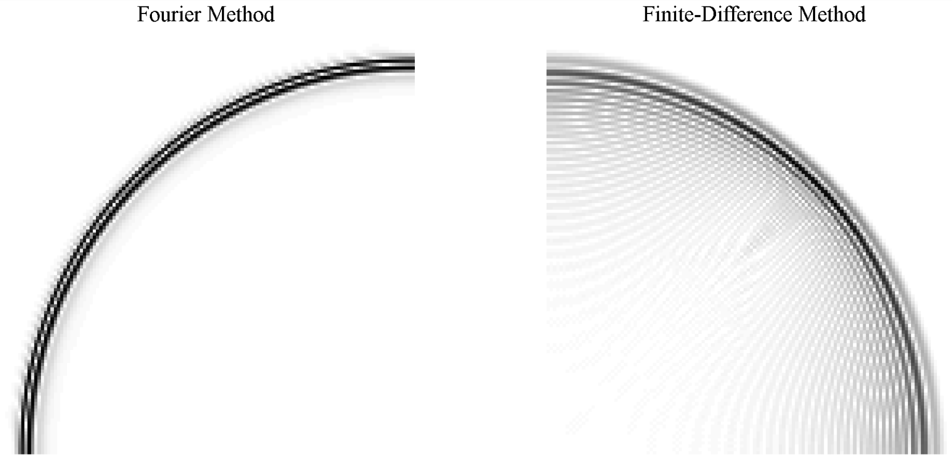
$$\partial_x^2 p + \partial_z^2 p \;= \mathscr{F}^{-1}[-k_x^2 \; \mathscr{F}[p]] \;+\; \mathscr{F}^{-1}[-k_z^2 \; \mathscr{F}[p]]$$

Acoustic 2D

Parameter	Value
x_{max}	200 m
nx	256
c	343 m/s
dt	0.00046 s
dx	0.78 m
f_0	200 Hz
ϵ	0.2

```
# [...]
# second space derivatives
for j in range(nz):
    d2px[:,j] = fourier_derivative_2nd(p[:,j].T, dx)
for i in range(nx):
    d2pz[i,:] = fourier_derivative_2nd(p[i,:], dx)
# Extrapolation
pnew = 2 * p - pold + c**2 * dt**2 * (d2px + d2pz)
# [...]
```

Acoustic 2D



Numerical anisotropy

Investigating the dispersion behaviour by finding solutions to monochromatic plane waves propagating in the direction $\mathbf{k} = (k_x, k_z)$

$$p^n_{j,k} = e^{i(k_x j \mathrm{d}x + k_z k \mathrm{d}x - \omega n \mathrm{d}t)}$$

With Fourier method the derivatives can be calculated by

$$\partial_x^2 p^n_{j,k} = -k_x^2 \, e^{i(k_x j \mathrm{d}x + k_z k \mathrm{d}x - \omega n \mathrm{d}t)}$$
$$\partial_z^2 p^n_{j,k} = -k_z^2 \, e^{i(k_x j \mathrm{d}x + k_z k \mathrm{d}x - \omega n \mathrm{d}t)}$$

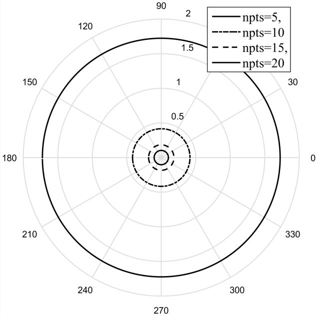
Combining this with the 3-point-operator for the time derivative

$$\partial_t^2 p^n_{j,k} = -\frac{4}{dt^2} \sin^2(\frac{\omega dt}{2}) \, e^{i(k_x j \mathrm{d}x + k_z k \mathrm{d}x - \omega n \mathrm{d}t)}$$

Numerical anisotropy

We obtain the numerical dispersion relation in 2D for arbitrary wave number vectors (i.e., propagation directions) **k** as

$$c(\mathbf{k}) = \frac{\omega}{|\mathbf{k}|} = \frac{2}{|\mathbf{k}|dt} \sin^{-1}\left(\frac{cdt\sqrt{k_x^2 + k_z^2}}{2}\right) .$$



Elastic 1D

1D Elastic wave equation

$$\rho(x)\ddot{u}(x,t) \;=\; \partial_x\left[\mu(x)\partial_x u(x,t)\right] \;+\; f(x,t)$$

u	displacement field
μ	space-dependent shear modulus

The finite-difference approximation of the extrapolation part leads to

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

with space derivatives to be calculated using the Fourier method.

Elastic 1D

The sequence of operations required to obtain the r.h.s. reads

$$u_i^j \;\rightarrow \mathscr{F}[u_i^j] \rightarrow \; U_\nu^j \rightarrow ikU_\nu^j \rightarrow \mathscr{F}^{-1}[ikU_\nu^j] \rightarrow \partial_x u_i^j$$
$$\partial_x u_i^j \rightarrow \mathscr{F}[\mu_i \partial_x u_i^j] \rightarrow \tilde{U}_\nu^j \rightarrow \mathscr{F}^{-1}[ik\tilde{U}_\nu^j] \rightarrow \partial_x \left[\mu(x)\partial_x u(x,t)\right]$$

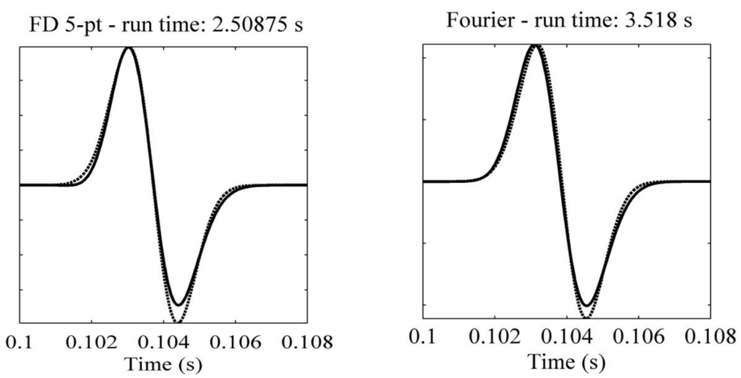
where capital letters denote fields in the spectral domain, lower indices with Greek letters indicate discrete frquencies, and $\tilde{U}_\nu^j = \mu_i \partial_x u_i^j$ was introduced as an intermediate result to facilitate notation.

Elastic 1D

Finding a setup for a classic staggered-grid finite-difference solution to the elastic 1D problem, leads to an energy misfit to the analytical solution u_a of 1%. The energy misfit is simply calculated by $(u_{FD} - u_a)^2 / u_a^2$

	FD	PS
nx	3000	1000
nt	2699	3211
c	3000 m/s	3000 m/s
dx	0.33 m	1.0 m
dt	5.5e-5 s	4.7e-5 s
f_0	260 Hz	260 Hz
ϵ	0.5	0.14
n/λ	34	11

Elastic 1D



Comparing memory requirements and computation speed between the Fourier method (**right**) and a 4th-order finite-difference scheme (**left**). In both cases the relative error compared to the analytical solution (misfit energy calculated by $\frac{u_{FD}-u_a}{u_a^2}$) is approximately 1%. The big difference is the number of grid points along the x dimension. The ratio is 3:1 (FD:Fourier)

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.