The Fourier Method

- What is a *pseudo*-spectral Method?
- Fourier Derivatives
- The Fast Fourier Transform (FFT)
- The Acoustic Wave Equation with the Fourier Method
- Comparison with the Finite-Difference Method
- Dispersion and Stability of Fourier Solutions

Spectral solutions to time-dependent PDEs are formulated in the frequency-wavenumber domain and solutions are obtained in terms of spectra (e.g. seismograms). This technique is particularly interesting for geometries where partial solutions in the ω-k domain can be obtained analytically (e.g. for layered models).

In the pseudo-spectral approach - in a finite-difference like manner - the PDEs are solved pointwise in physical space (x-t). However, the space derivatives are calculated using orthogonal functions (e.g. Fourier Integrals, Chebyshev polynomials). They are either evaluated using matrix-matrix multiplications or the fast Fourier transform (FFT).

.. let us recall the definition of the derivative using Fourier integrals ...

$$\partial_x f(x) = \partial_x \left(\int_{-\infty}^{\infty} F(k) e^{-ikx} dk \right)$$

$$= -\int_{-\infty}^{\infty} ikF(k) e^{-ikx} dk$$

... we could either ...

1) perform this calculation in the space domain by convolution

2) actually transform the function f(x) in the k-domain and back

... the latter approach became interesting with the introduction of the Fast Fourier Transform (FFT). What's so fast about it ?

The FFT originates from a paper by Cooley and Tukey (1965, Math. Comp. vol 19 297-301) which revolutionised all fields where Fourier transforms where essential to progress.

The discrete Fourier Transform can be written as

$$\hat{u}_k = \frac{1}{N} \sum_{j=0}^{N-1} u_j e^{-2\pi ikj/N}, k = 0,1,..., N-1$$

$$u_k = \sum_{j=0}^{N-1} \hat{u}_j e^{2\pi ikj/N}, k = 0,1,..., N-1$$

... this can be written as matrix-vector products ...
for example the inverse transform yields ...

$$\begin{bmatrix} 1 & 1 & 1 & 1 & \dots & 1 \\ 1 & \omega & \omega^2 & \omega^3 & \dots & \omega^{N-1} \\ 1 & \omega^2 & \omega^4 & \omega^6 & \dots & \omega^{2N-2} \\ \vdots & \vdots & & & & \vdots \\ \vdots & \vdots & & & & \vdots \\ 1 & \omega^{N-1} & \dots & \dots & \dots & \omega^{(N-1)^2} \end{bmatrix} \begin{bmatrix} \hat{u}_0 \\ \hat{u}_1 \\ \hat{u}_2 \\ \vdots \\ \vdots \\ \hat{u}_{N-1} \end{bmatrix} = \begin{bmatrix} u_0 \\ u_1 \\ u_2 \\ \vdots \\ \vdots \\ u_{N-1} \end{bmatrix}$$

.. where ...

$$\omega = e^{2\pi i / N}$$

... the FAST bit is recognising that the full matrix - vector multiplication can be written as a few sparse matrix - vector multiplications (for details see for example Bracewell, the Fourier Transform and its applications, MacGraw-Hill) with the effect that:

Number of multiplications

full matrix	FFT
N^2	$2N\log_2 N$

this has enormous implications for large scale problems.
Note: the factorisation becomes particularly simple and effective when N is a highly composite number (power of 2).

Number of multiplications

Problem	full matrix	FFT	Ratio full/FFT
1D (nx=512)	2.6×10^5	9.2×10^3	28.4
1D (nx=2096)			94.98
1D (nx=8384)			312.6

.. the right column can be regarded as the speedup of an algorithm when the FFT is used instead of the full system.

let us take the acoustic wave equation with variable density

$$\frac{1}{\rho c^2}\partial_t^2 p = \partial_x\left(\frac{1}{\rho}\partial_x p\right)$$

the left hand side will be expressed with our standard centered finite-difference approach

$$\frac{1}{\rho c^2 dt^2}\left[p(t+dt) - 2p(t) + p(t-dt)\right] = \partial_x\left(\frac{1}{\rho}\partial_x p\right)$$

... leading to the extrapolation scheme ...

$$p(t+dt) = \rho c^2 dt^2 \boxed{\partial_x \left(\frac{1}{\rho} \partial_x p \right)} + 2p(t) - p(t-dt)$$

where the space derivatives will be calculated using the Fourier Method. The highlighted term will be calculated as follows:

$$P_j^n \to \mathrm{FFT} \to \hat{P}_\upsilon^n \to ik_\upsilon \hat{P}_\upsilon^n \to \mathrm{FFT}^{-1} \to \partial_x P_j^n$$

multiply by $1/\rho$

$$\frac{1}{\rho}\partial_x P_j^n \to \mathrm{FFT} \to \left(\frac{1}{\rho}\partial_x \hat{P}\right)_\upsilon^n \to ik_\upsilon \left(\frac{1}{\rho}\partial_x \hat{P}\right)_\upsilon^n \to \mathrm{FFT}^{-1} \to \partial_x \left(\frac{1}{\rho}\partial_x P_j^n\right)$$

... then extrapolate ...

Acoustic Wave Equation - 3D

$$
\begin{aligned}
p(t+dt) = {} & \\
& \rho c^2 dt^2 \boxed{\left(\partial_x \left(\frac{1}{\rho} \partial_x p \right) + \partial_y \left(\frac{1}{\rho} \partial_y p \right) + \partial_z \left(\frac{1}{\rho} \partial_z p \right) \right)} \\
& + 2p(t) - p(t-dt)
\end{aligned}
$$

.. where the following algorithm applies to each space dimension ...

$$
P_j^n \rightarrow \text{FFT} \rightarrow \hat{P}_\upsilon^n \rightarrow ik_\upsilon \hat{P}_\upsilon^n \rightarrow \text{FFT}^{-1} \rightarrow \partial_x P_j^n
$$

$$
\frac{1}{\rho} \partial_x P_j^n \rightarrow \text{FFT} \rightarrow \left(\frac{1}{\rho} \partial_x \hat{P} \right)_\upsilon^n \rightarrow ik_\upsilon \left(\frac{1}{\rho} \partial_x \hat{P} \right)_\upsilon^n \rightarrow \text{FFT}^{-1} \rightarrow \partial_x \left(\frac{1}{\rho} \partial_x P_j^n \right)
$$

let us compare the core of the algorithm - the calculation of the derivative (Matlab code)

```
function df=fder1d(f,dx,nop)
% fDER1D(f,dx,nop) finite difference
% second derivative

nx=max(size(f));

n2=(nop-1)/2;

if nop==3; d=[1 -2 1]/dx^2; end
if nop==5; d=[-1/12 4/3 -5/2 4/3 -1/12]/dx^2; end

df=[1:nx]*0;

for i=1:nop;
df=df+d(i).*cshift1d(f,-n2+(i-1));
end
```

... and the first derivative using FFTs ...

```
function df=sder1d(f,dx)
% SDER1D(f,dx) spectral derivative of vector
nx=max(size(f));


% initialize k
kmax=pi/dx;
dk=kmax/(nx/2);
for i=1:nx/2, k(i)=(i)*dk; k(nx/2+i)=-kmax+(i)*dk; end
k=sqrt(-1)*k;

% FFT and IFFT
ff=fft(f); ff=k.*ff; df=real(ifft(ff));
```

.. simple and elegant ...

... with the usual *Ansatz*

$$p_j^n = e^{i(kjdx - n\omega dt)}$$

we obtain

$$\partial_x^2 p_j^n = -k^2 e^{i(kjdx - \omega ndt)}$$

$$\partial_t^2 p_j^n = -\frac{4}{dt^2} \sin^2 \frac{\omega dt}{2} e^{i(kjdx - \omega ndt)}$$

... leading to

$$k = \frac{2}{cdt} \sin \frac{\omega dt}{2}$$

$$k = \frac{2}{cdt} \sin \frac{\omega dt}{2}$$

$$\omega = \frac{2}{dt} \sin^{-1}\left(\frac{kcdt}{2}\right)$$

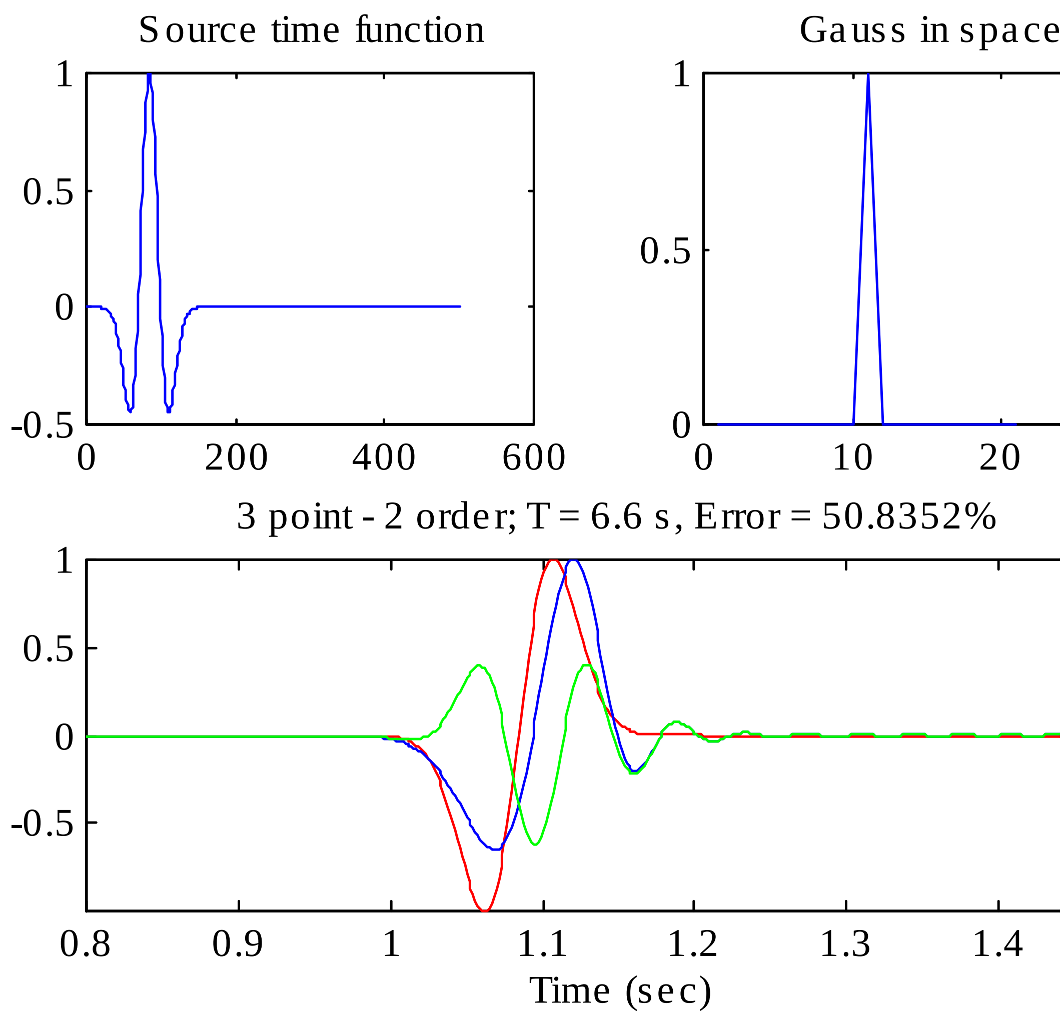
What are the consequences?

a) when dt << 1, $\sin^{-1}$ (kcdt/2) ≈kcdt/2 and w/k=c
-> practically no dispersion
b) the argument of asin must be smaller than one.

$$\frac{k_{\max} cdt}{2} \leq 1$$

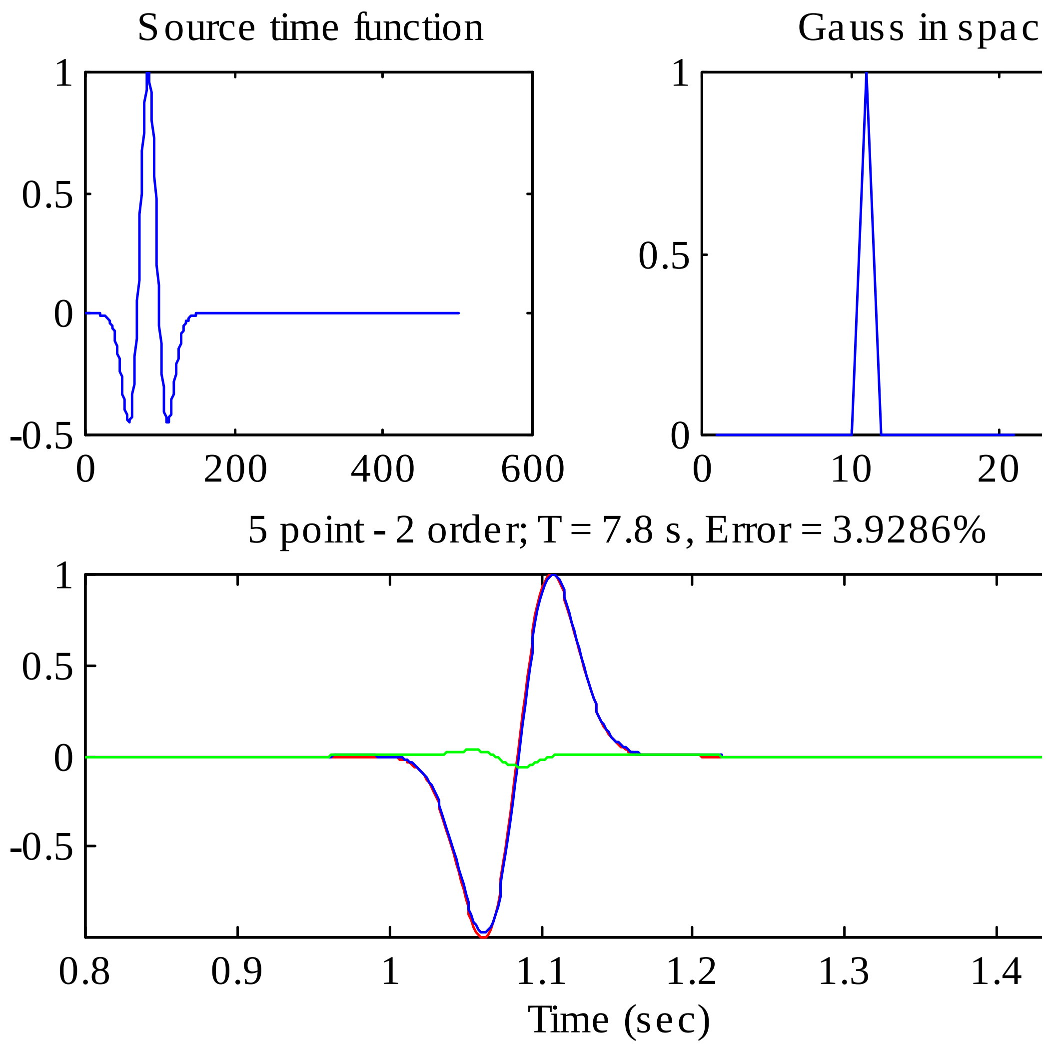
$$cdt / dx \leq 2 / \pi \approx 0.636$$

Fourier Method - Comparison with FD - 10Hz

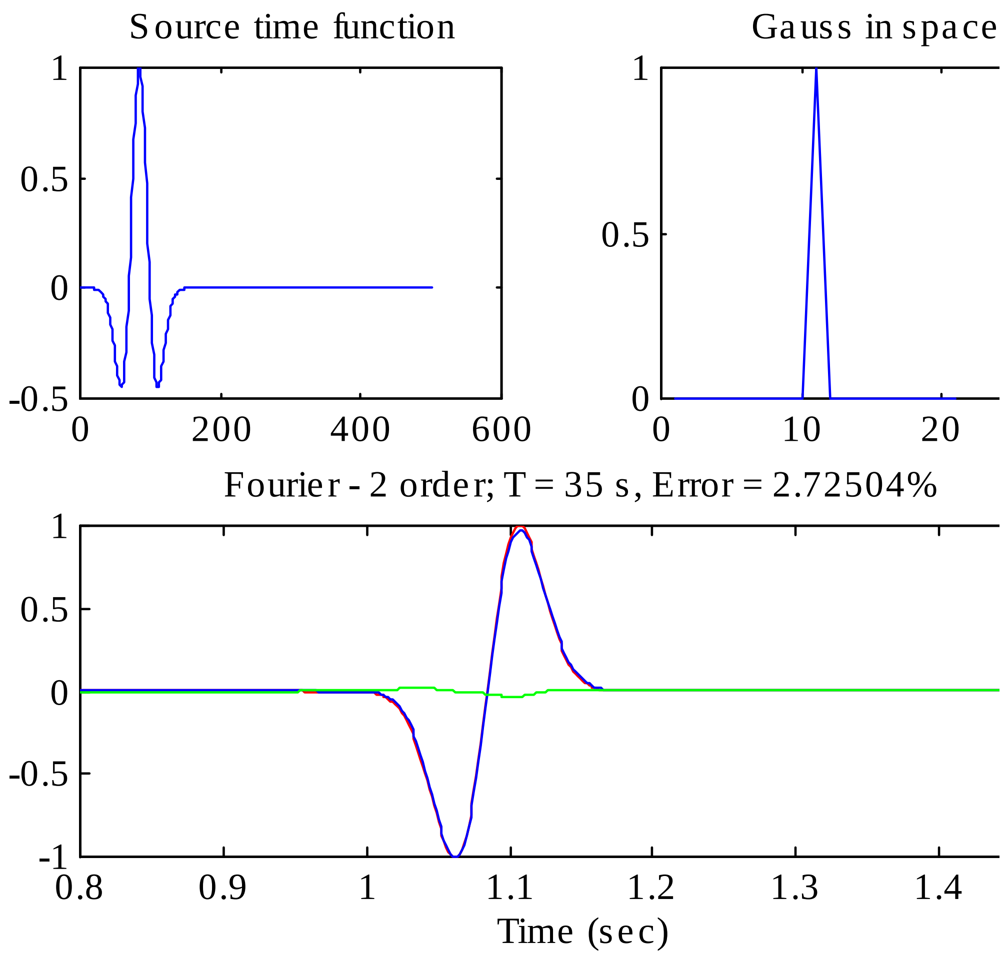


Example of acoustic 1D wave simulation.
FD 3 -point operator
red-analytic; blue-numerical; green-difference

Fourier Method - Comparison with FD - 10Hz

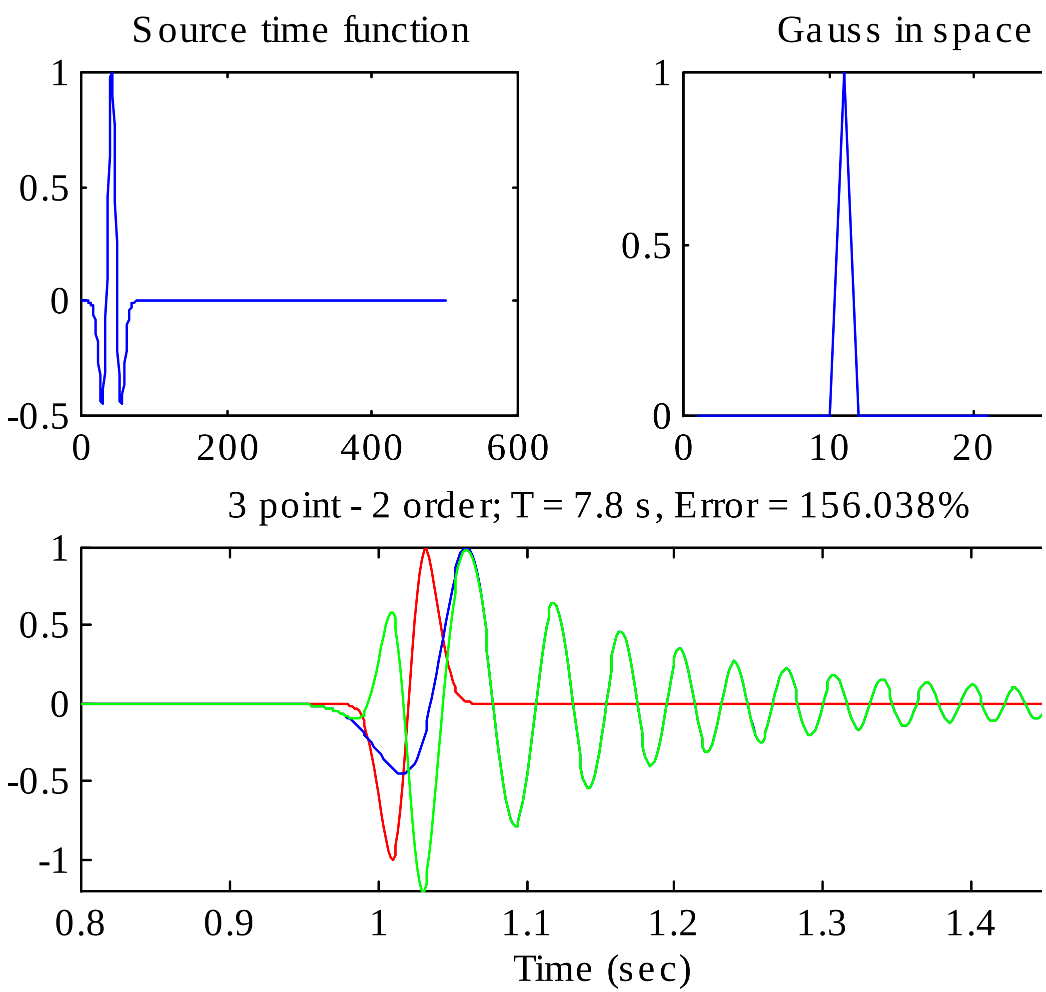


Example of acoustic 1D wave simulation.
FD 5 -point operator
red-analytic; blue-numerical; green-difference

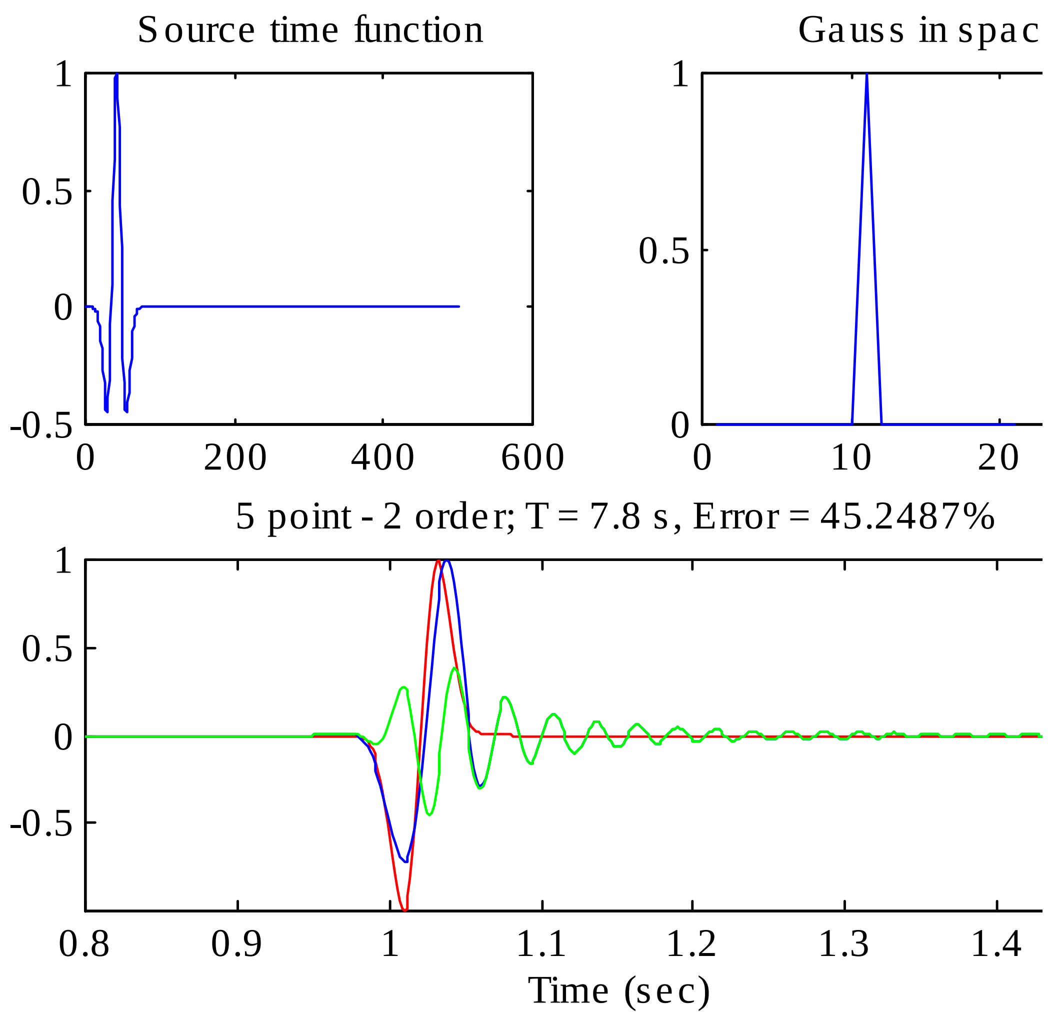


Example of acoustic 1D wave simulation.
Fourier operator
red-analytic; blue-numerical; green-difference

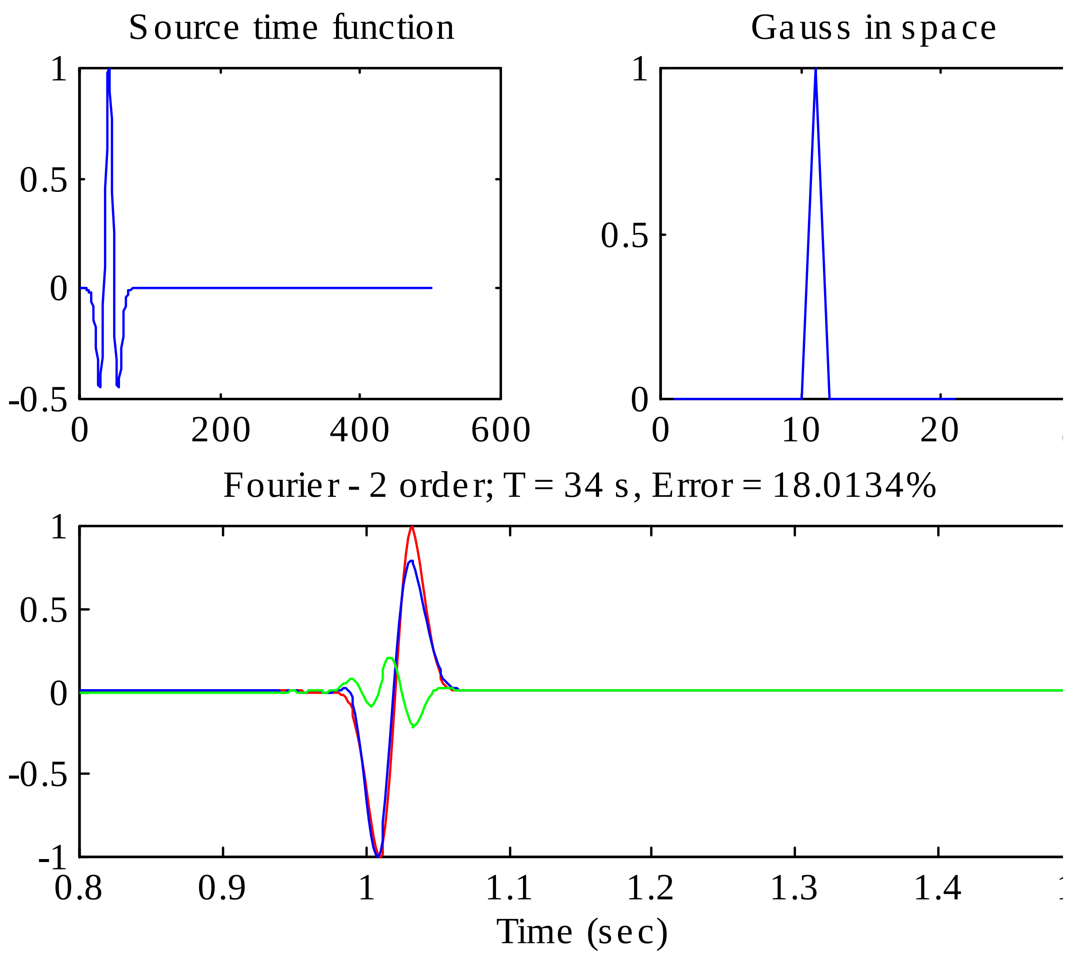
Fourier Method - Comparison with FD - 20Hz



Example of acoustic 1D wave simulation.
FD 3 -point operator
red-analytic; blue-numerical; green-difference

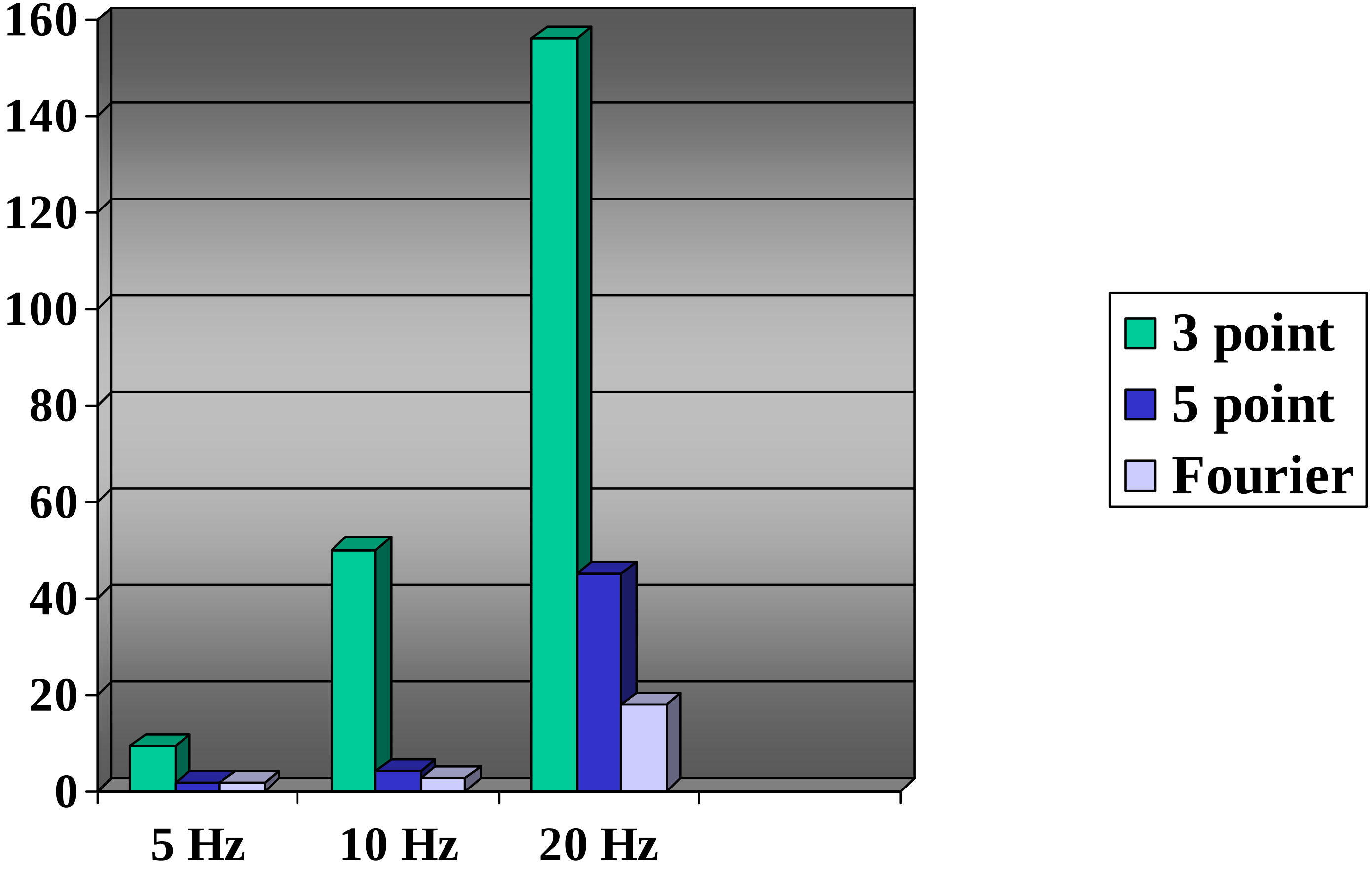


Example of acoustic 1D wave simulation.
FD 5 -point operator
red-analytic; blue-numerical; green-difference



Example of acoustic 1D wave simulation.
Fourier operator
red-analytic; blue-numerical; green-difference

Difference (%) between numerical and analytical solution as a function of propagating frequency



Simulation time
5.4s
7.8s
33.0s

The concept of Green's Functions (impulse responses) plays an important role in the solution of partial differential equations. It is also useful for numerical solutions. Let us recall the acoustic wave equation

$$\partial_t^2 p = c^2 \Delta p$$

with Δ being the Laplace operator. We now introduce a delta source in space and time

$$\partial_t^2 p = \delta(\underline{x})\delta(t) + c^2 \Delta p$$

the formal solution to this equation is

$$p(\underline{x}, t) = \frac{1}{4\pi c^2} \frac{\delta(t - |\underline{x}|/c)}{|\underline{x}|}$$

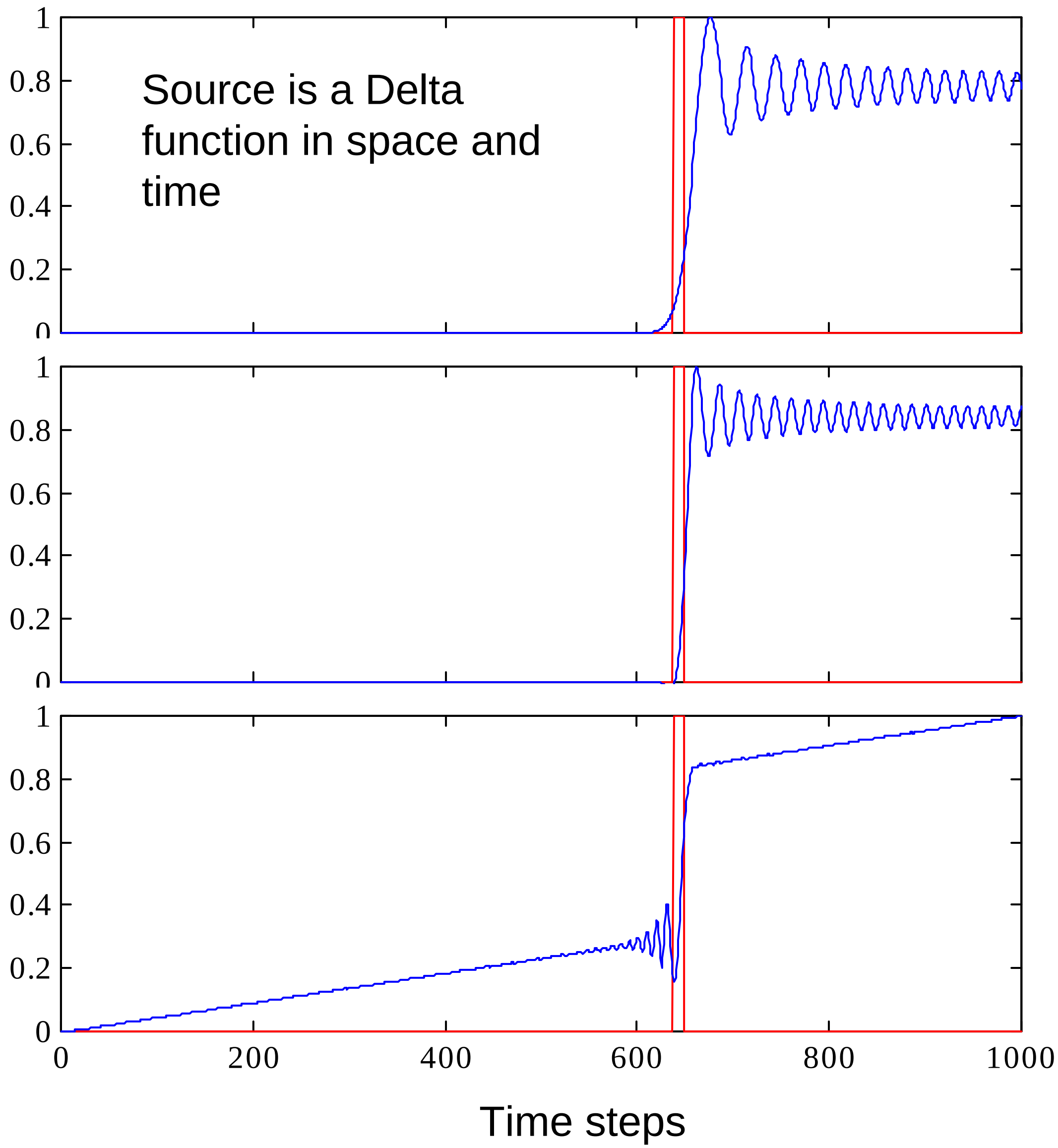
(Full proof given in Aki and Richards, Quantitative Seismology, Freeman+Co, 1981, p. 65)

$$p(\underline{x},t) = \frac{1}{4\pi c^2} \frac{\delta(t - |\underline{x}|/c)}{|\underline{x}|}$$

In words this means (in 1D and 3D but not in 2D, why?) , that in homogeneous media the same source time function which is input at the source location will be recorded at a distance r, but with amplitude proportional to 1/r.

An arbitrary source can evidently be constructed by summing up different delta - solutions. Can we use this property in our numerical simulations?

What happens if we solve our numerical system with delta functions as sources?



3 point operator

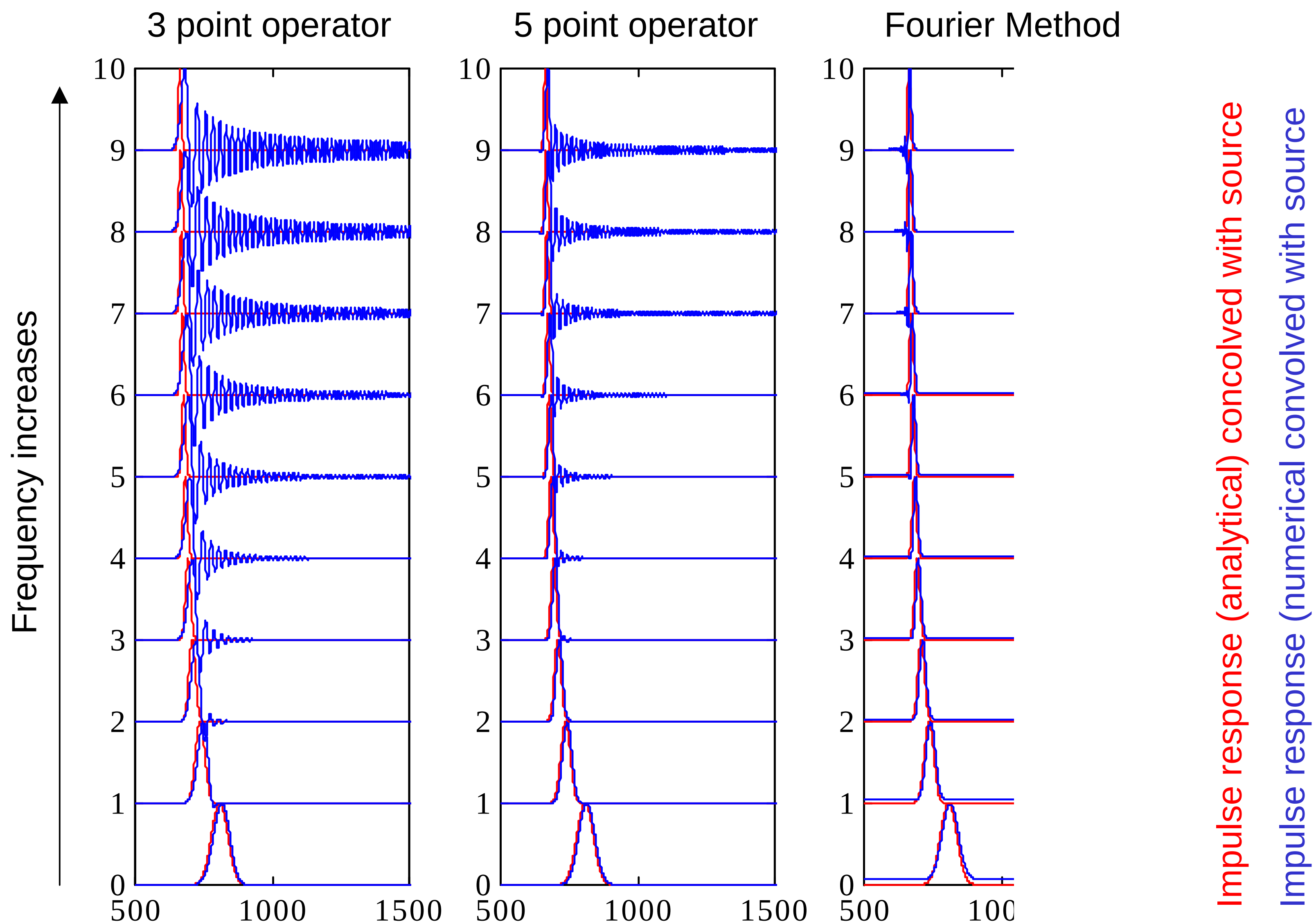
5 point operator

Fourier Method

Impulse response (analytical)

Impulse response (numerical

Numerical solutions and Green's Functions



The Fourier Method can be considered as the limit of the finite-difference method as the length of the operator tends to the number of points along a particular dimension.

The space derivatives are calculated in the wavenumber domain by multiplication of the spectrum with *ik.* The inverse Fourier transform results in an exact space derivative up to the Nyquist frequency.

The use of Fourier transform imposes some constraints on the smoothness of the functions to be differentiated. Discontinuities lead to Gibb's phenomenon.

As the Fourier transform requires periodicity this technique is particular useful where the physical problems are periodical (e.g. angular derivatives in cylindrical problems).