Fourier Transform: Applications

- Seismograms
- Eigenmodes of the Earth
- Time derivatives of seismograms
- The pseudo-spectral method for acoustic wave propagation

Fourier: Space and Time

Space	
x	space variable
L	spatial wavelength
$k=2\pi/\lambda$	spatial wavenumber
F(k)	wavenumber spectrum

Time	
t	Time variable
T	period
f	frequency
$\omega=2\pi f$	angular frequency

Fourier integrals

With the complex representation of sinusoidal functions e^{ikx} (or ($e^{i\omega t}$) the Fourier transformation can be written as:

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

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

The Fourier Transform
discrete vs. continuous

Whatever we do on the computer with data will be based on the discrete Fourier transform

continuous

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

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

discrete

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

$$f_k = \sum_{j=0}^{N-1} F_j e^{2\pi ikj/N}, k = 0,1,\ldots, N-1$$

The Fast Fourier Transform

... 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

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

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

The Fast Fourier Transform

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

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

.. where ...

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

The Fast Fourier Transform

... 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).

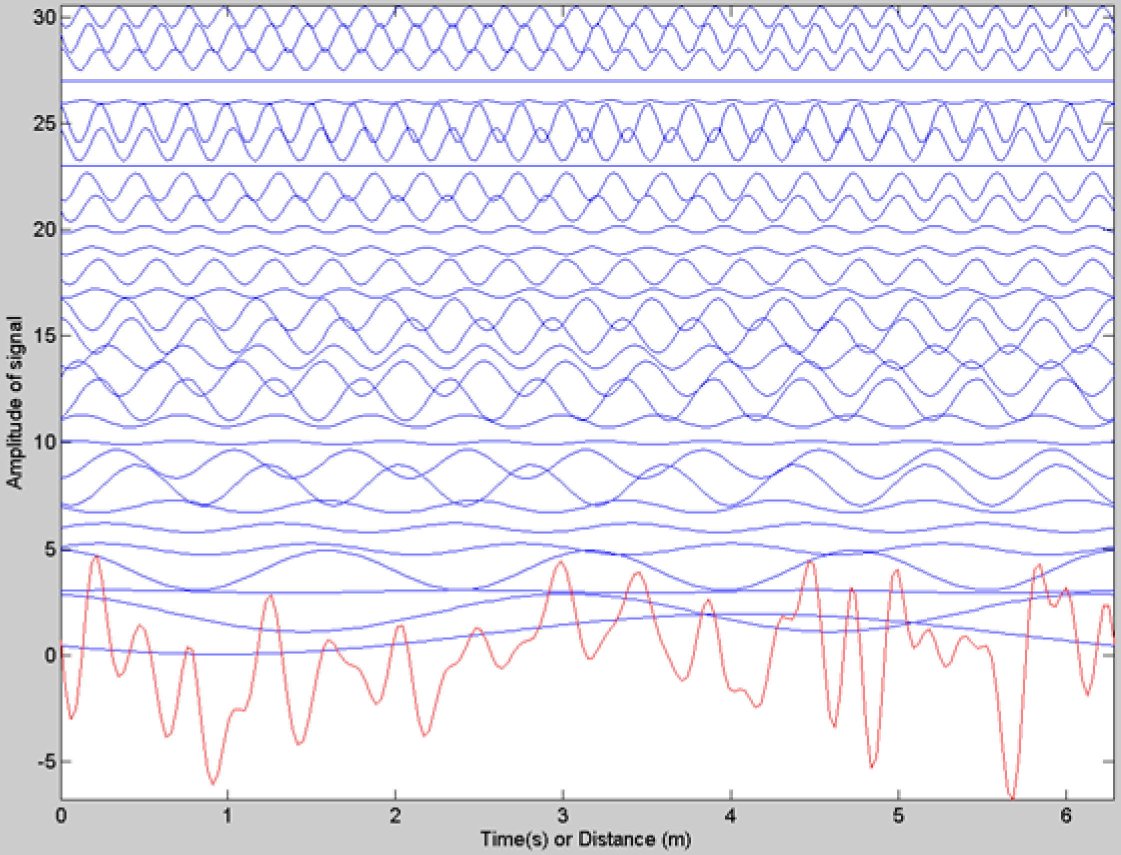
The Fast Fourier Transform

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.

Spectral synthesis



The red trace is the sum of all blue traces!

Phase and amplitude spectrum

The spectrum consists of two real-valued functions of angular frequency, the amplitude spectrum mod ($F(\omega)$) and the phase spectrum $\phi(\omega)$

$$F(\omega) = \left| F(\omega) \right| e^{i\Phi(\omega)}$$

In many cases the amplitude spectrum is the most important part to be considered. However there are cases where the phase spectrum plays an important role (-> resonance, seismometer)

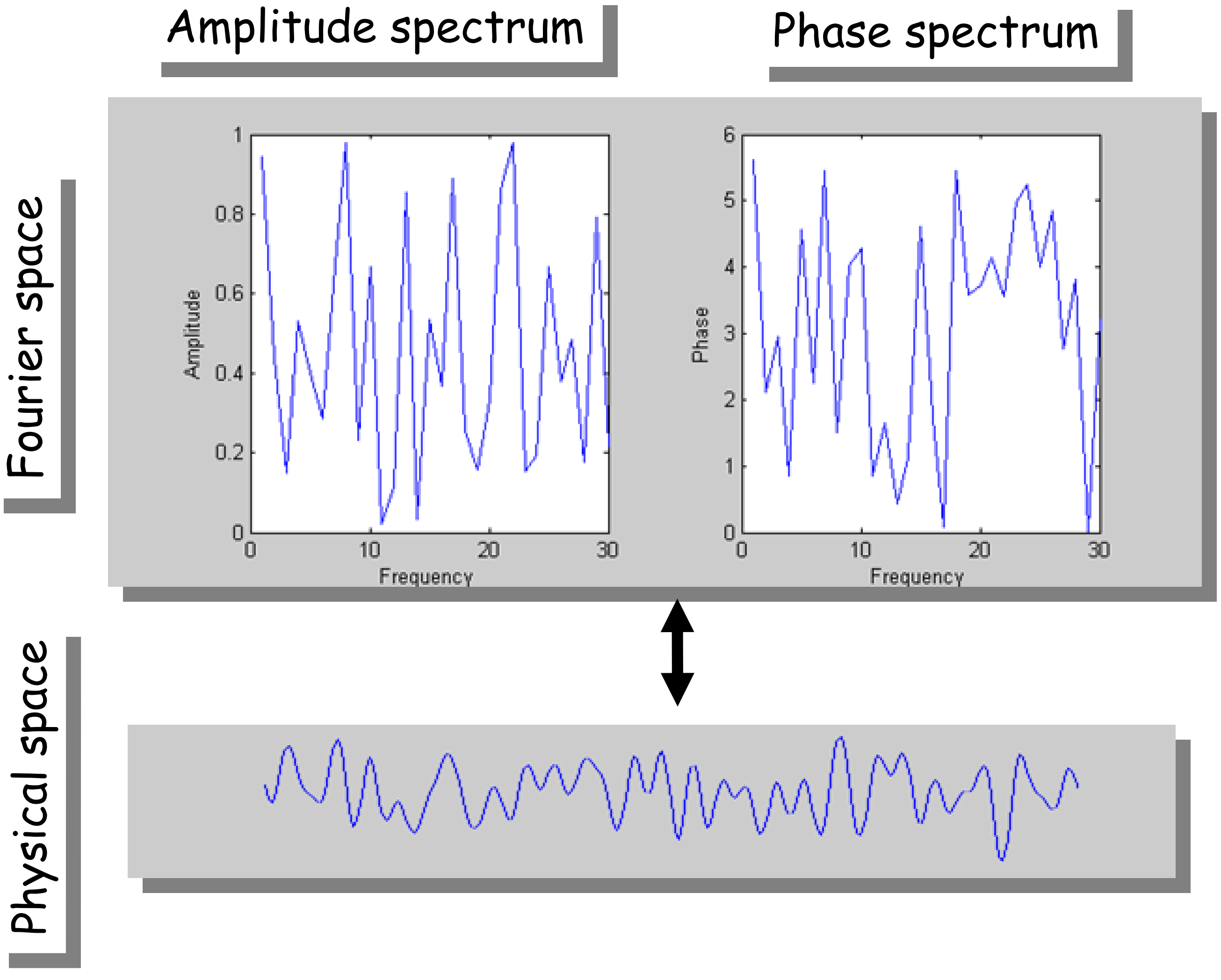
… remember …

$$z^* = a - ib = r(\cos\phi - i\sin\phi)$$

$$= r\cos{-\phi} - ri\sin(-\phi) = r^{-i\phi}$$

$$\left|z^2\right| = zz^* = (a+ib)(a-ib) = r^2$$

The spectrum



The Fast Fourier Transform (FFT)

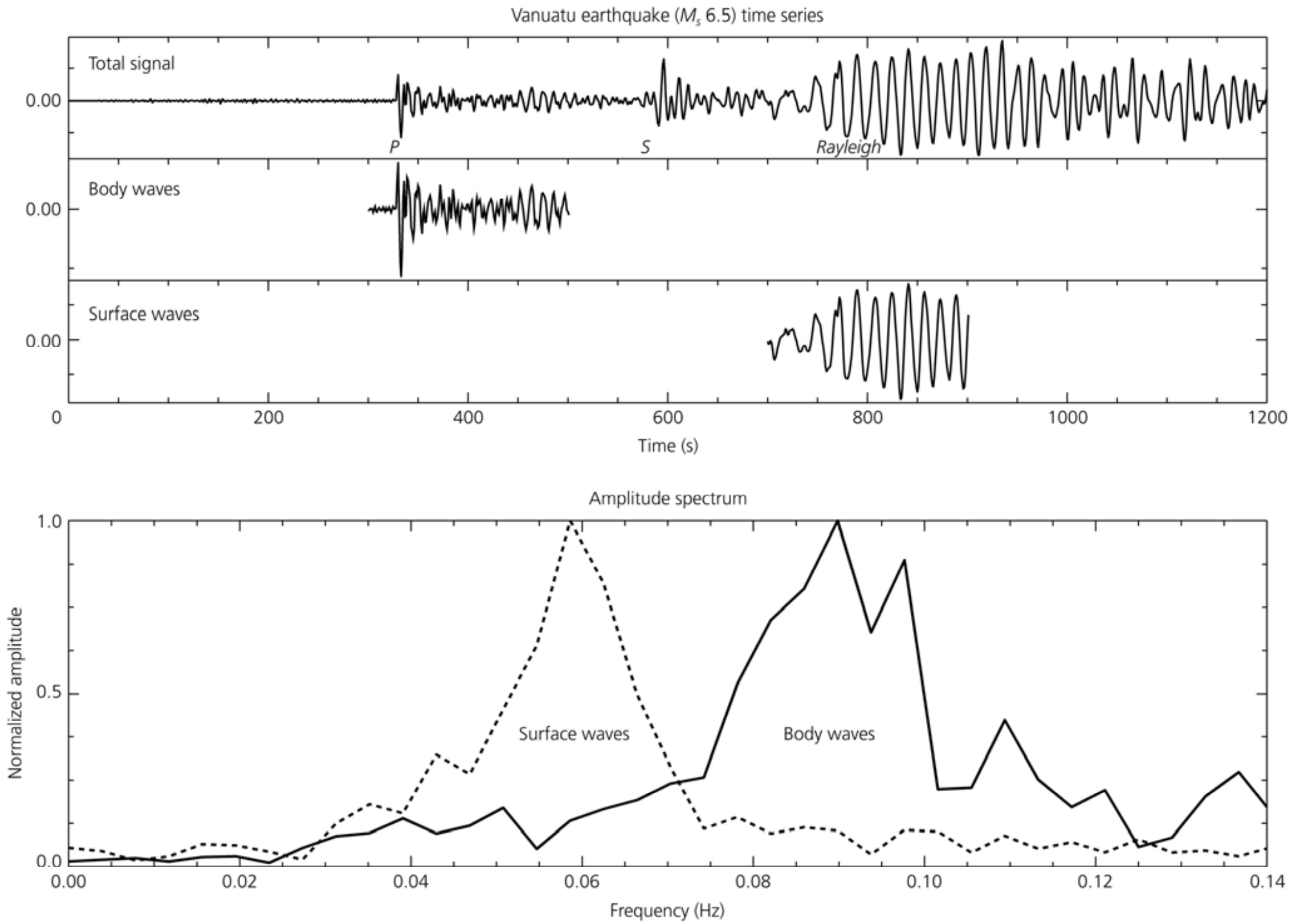
Most processing tools (e.g. octave, Matlab, Mathematica, Fortran, etc) have intrinsic functions for FFTs

Matlab FFT

```
>> help fft

 FFT Discrete Fourier transform.
  FFT(X) is the discrete Fourier transform (DFT) of vector X.  For
  matrices, the FFT operation is applied to each column. For N-D
  arrays, the FFT operation operates on the first non-singleton
  dimension.

  FFT(X,N) is the N-point FFT, padded with zeros if X has less
  than N points and truncated if it has more.

  FFT(X,[],DIM) or FFT(X,N,DIM) applies the FFT operation across the
  dimension DIM.

  For length N input vector x, the DFT is a length N vector X,
  with elements
                   N
     X(k) =       sum  x(n)*exp(-j*2*pi*(k-1)*(n-1)/N), 1 <= k <= N.
                  n=1
  The inverse DFT (computed by IFFT) is given by
                   N
     x(n) = (1/N) sum  X(k)*exp( j*2*pi*(k-1)*(n-1)/N), 1 <= n <= N.
                  k=1

  See also IFFT, FFT2, IFFT2, FFTSHIFT.
```

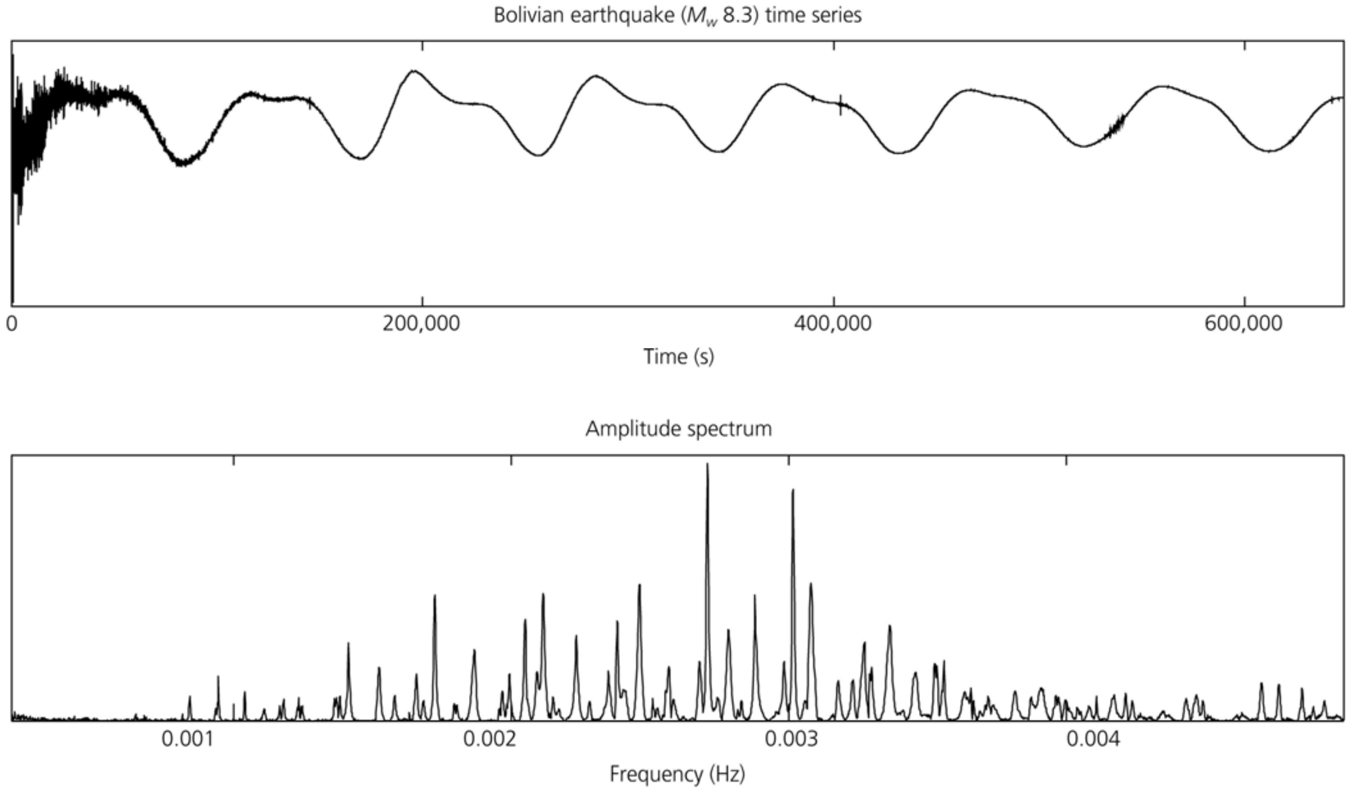
Frequencies in seismograms

Figure 6.2-3: Amplitude spectra for the body and surface wave segments from a large earthquake.

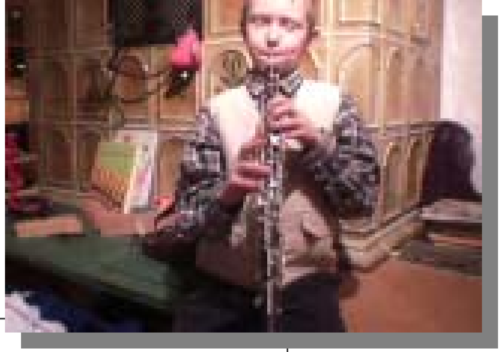


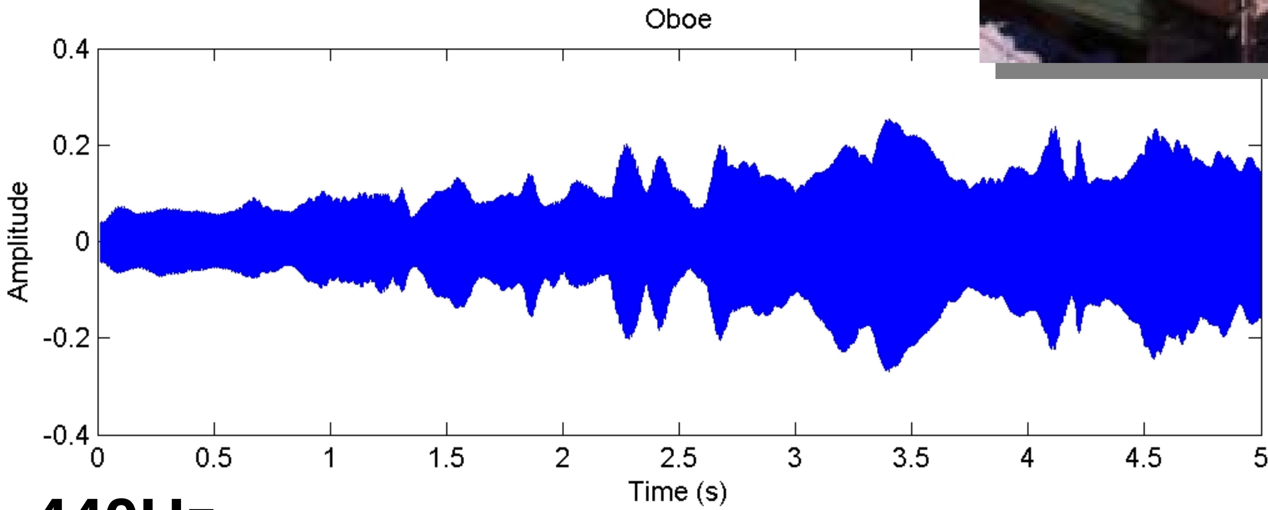
Amplitude spectrum
Eigenfrequencies

Figure 6.2-4: Amplitude spectra of a vertical-component seismogram from the great 1994 Bolivian earthquake.

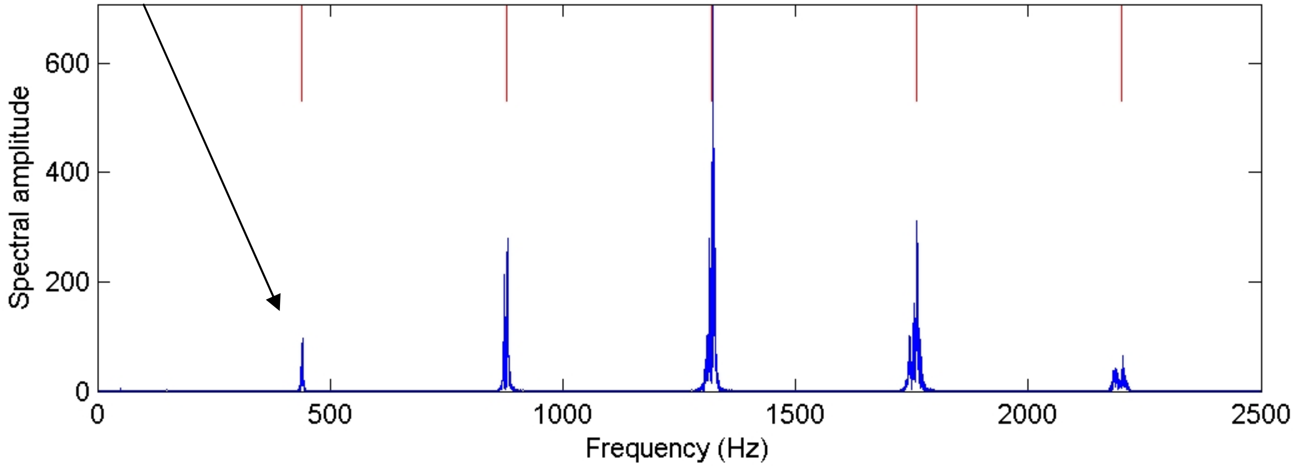


Sound of an instrument

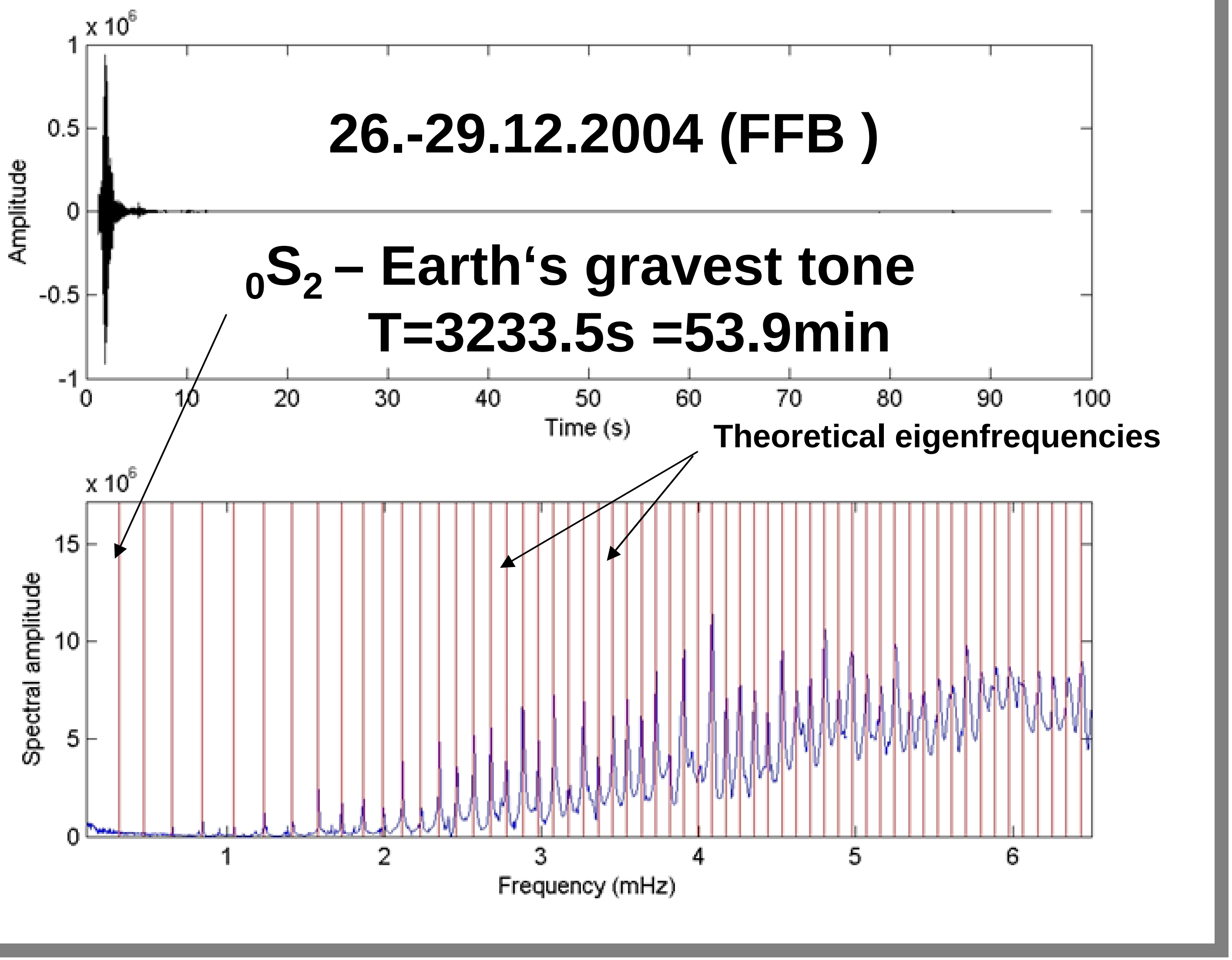




a' - 440Hz

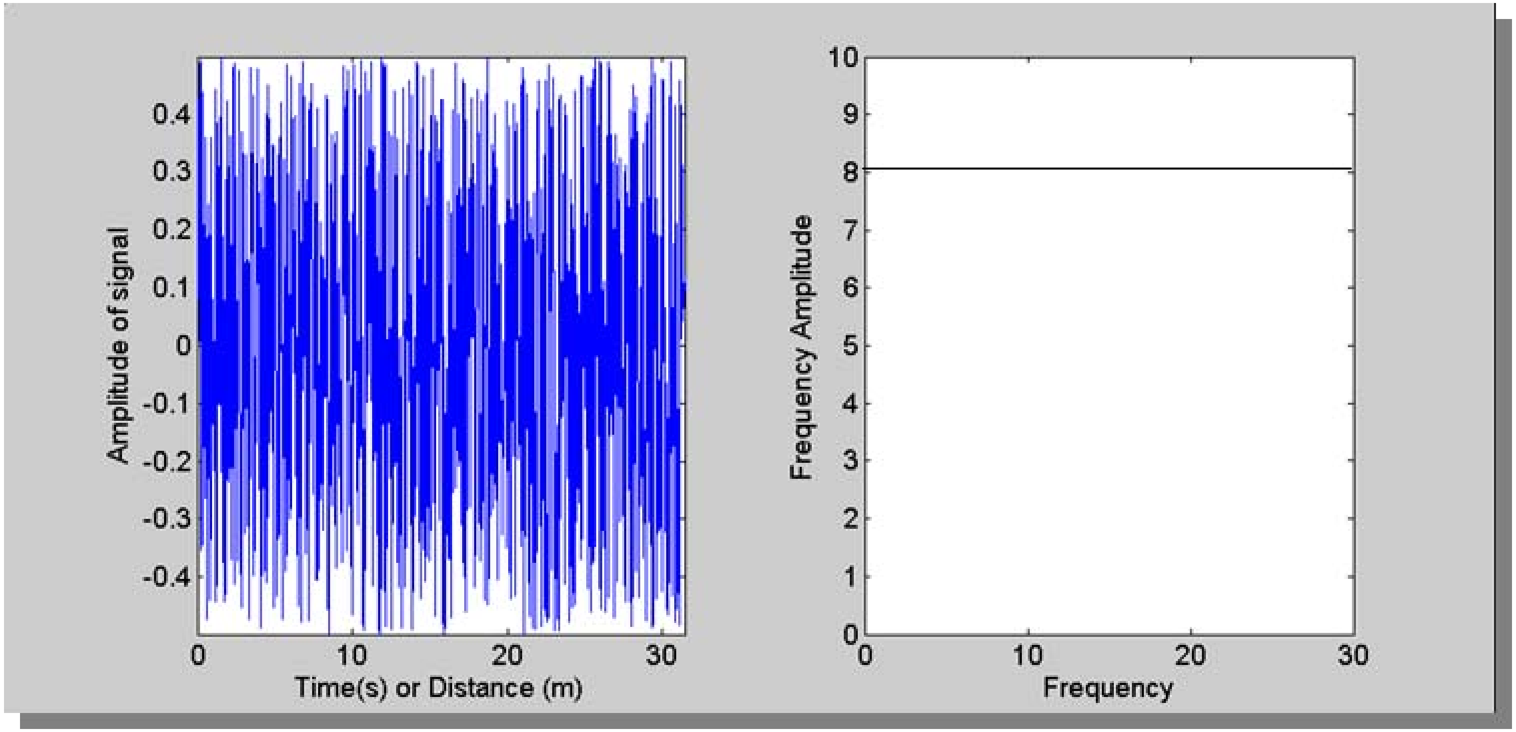


Instrument Earth



Fourier Spectra: Main Cases

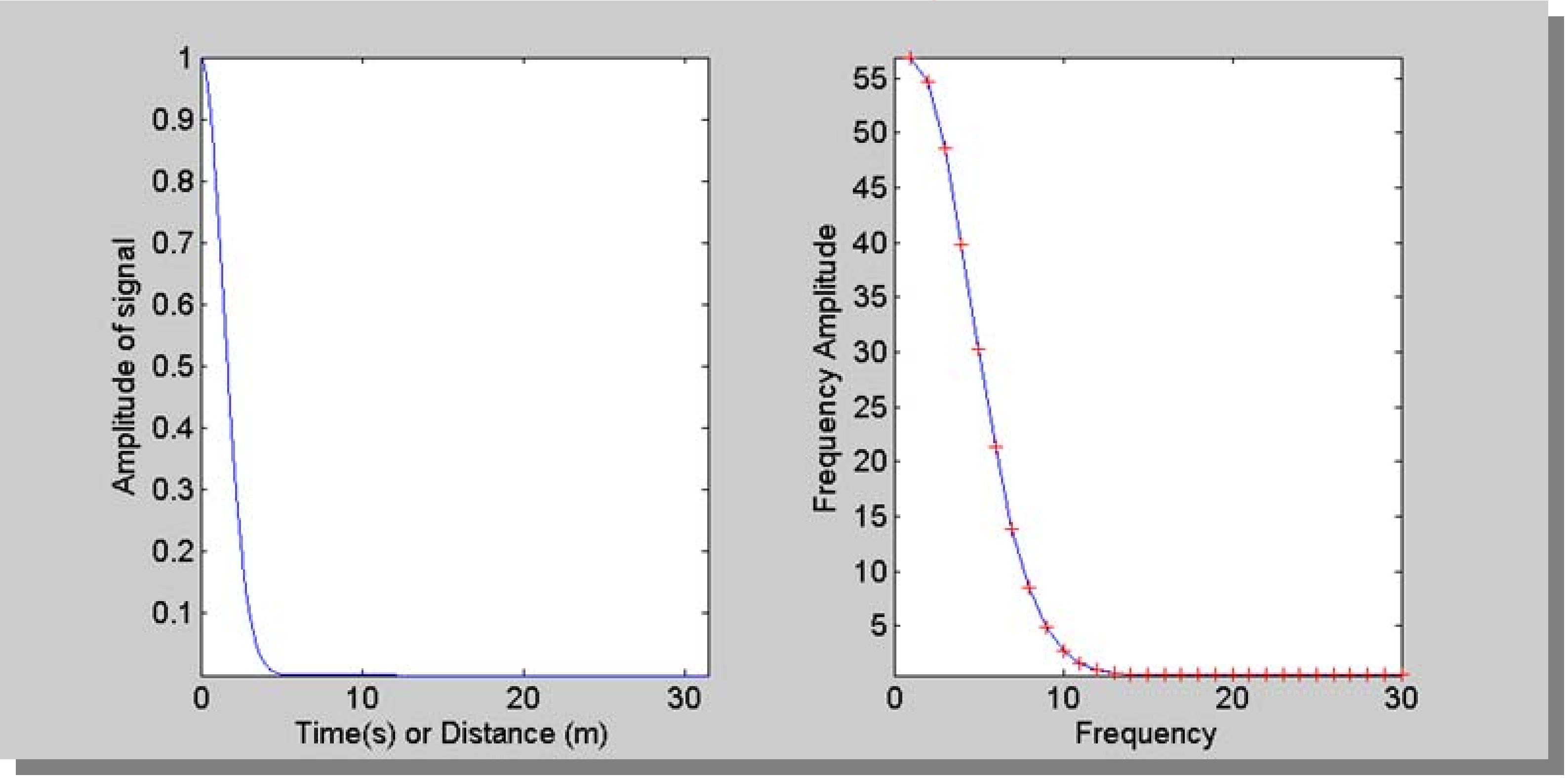
random signals



Random signals may contain all frequencies. A spectrum with constant contribution of all frequencies is called a white spectrum

Fourier Spectra: Main Cases

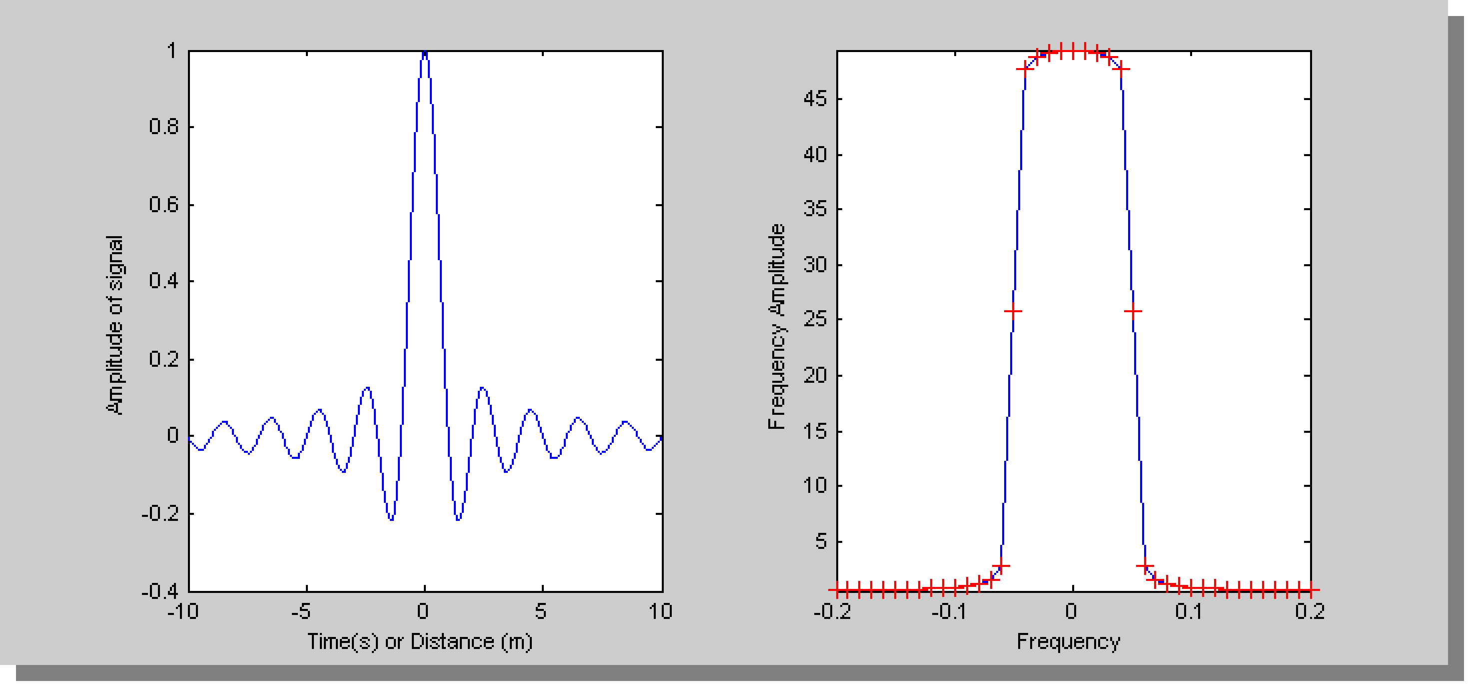
Gaussian signals



The spectrum of a Gaussian function will itself be a Gaussian function. How does the spectrum change, if I make the Gaussian narrower and narrower?

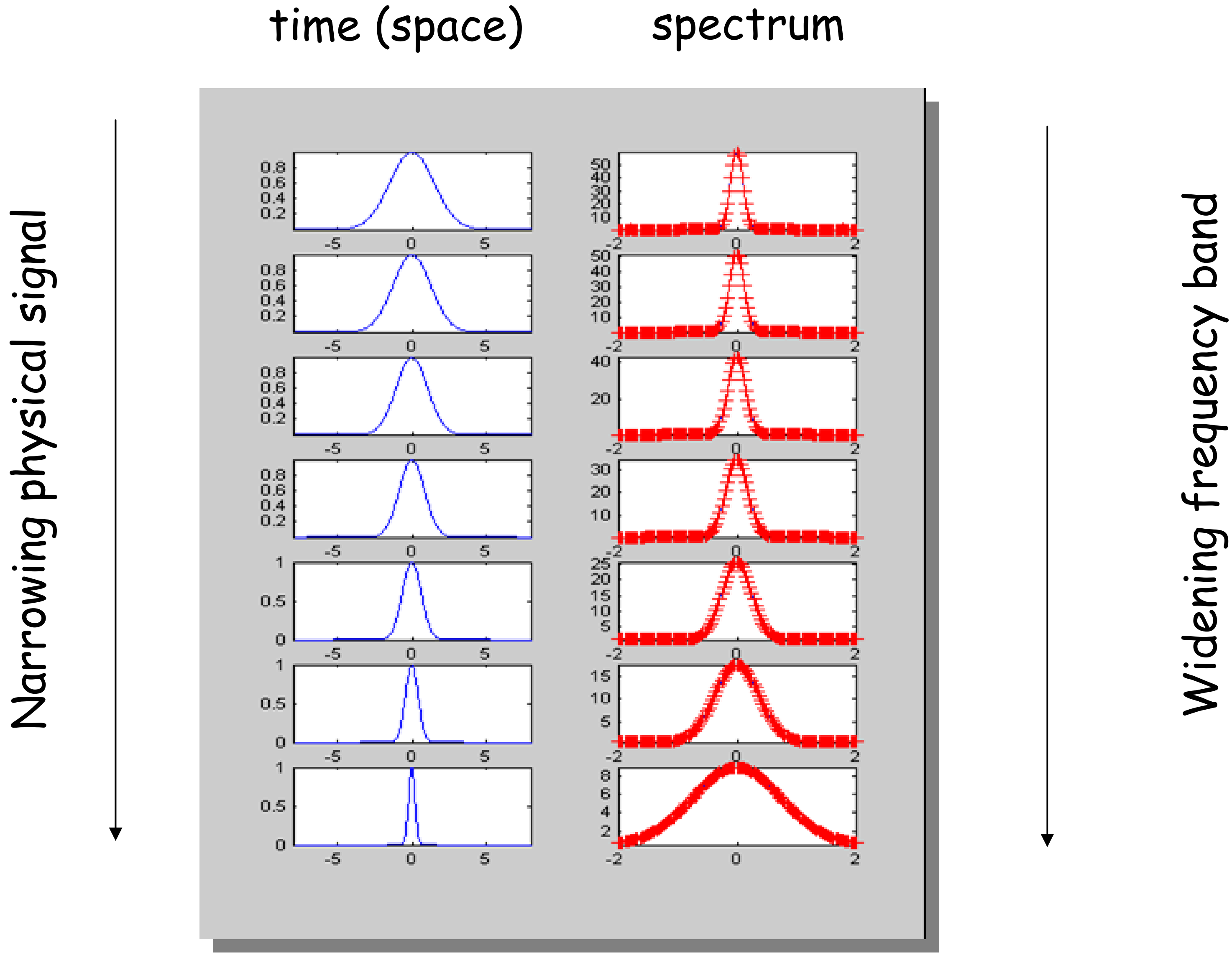
Fourier Spectra: Main Cases

Transient waveform

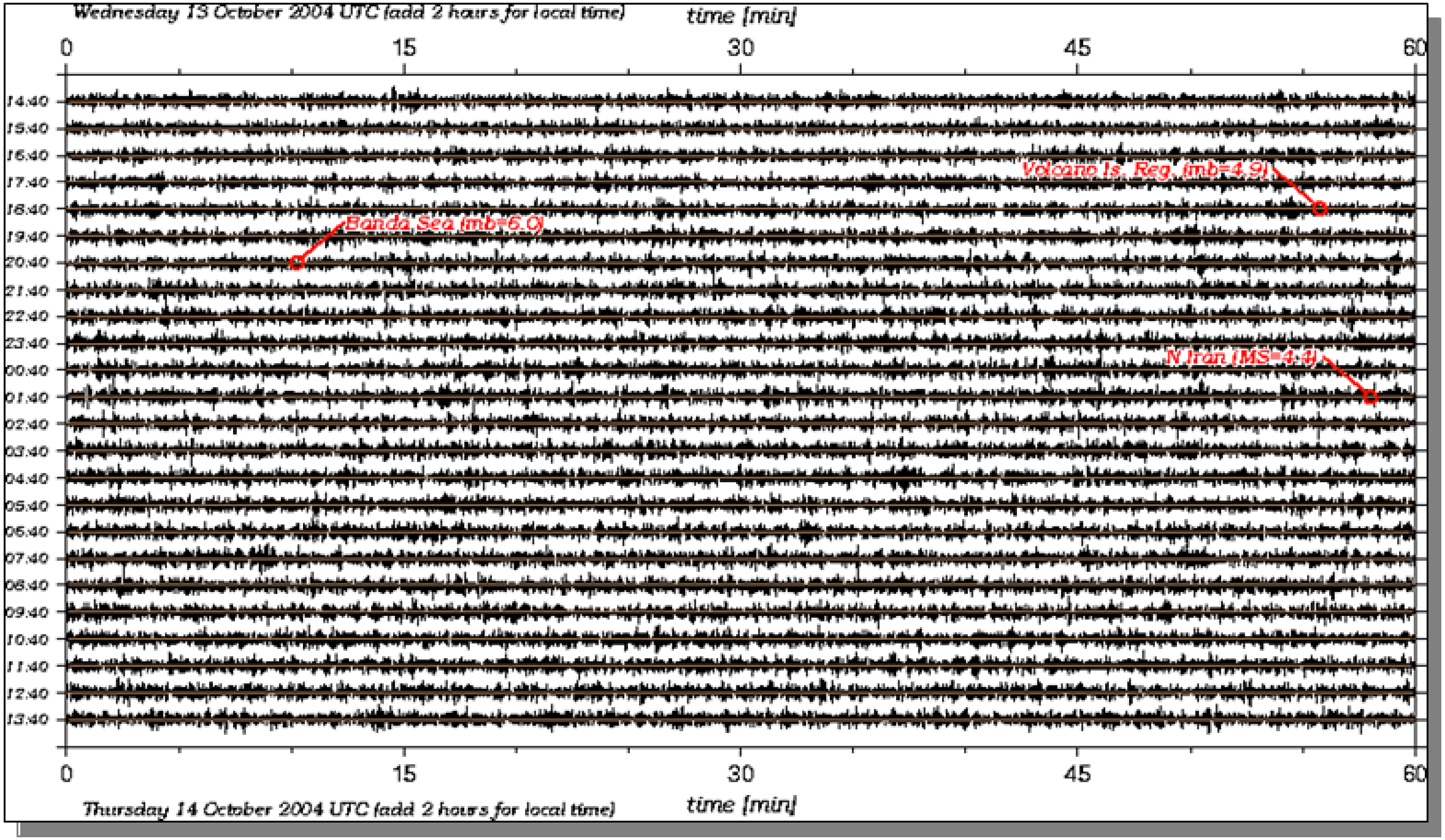


A transient wave form is a wave form limited in time (or space) in comparison with a harmonic wave form that is infinite

Puls-width and Frequency Bandwidth

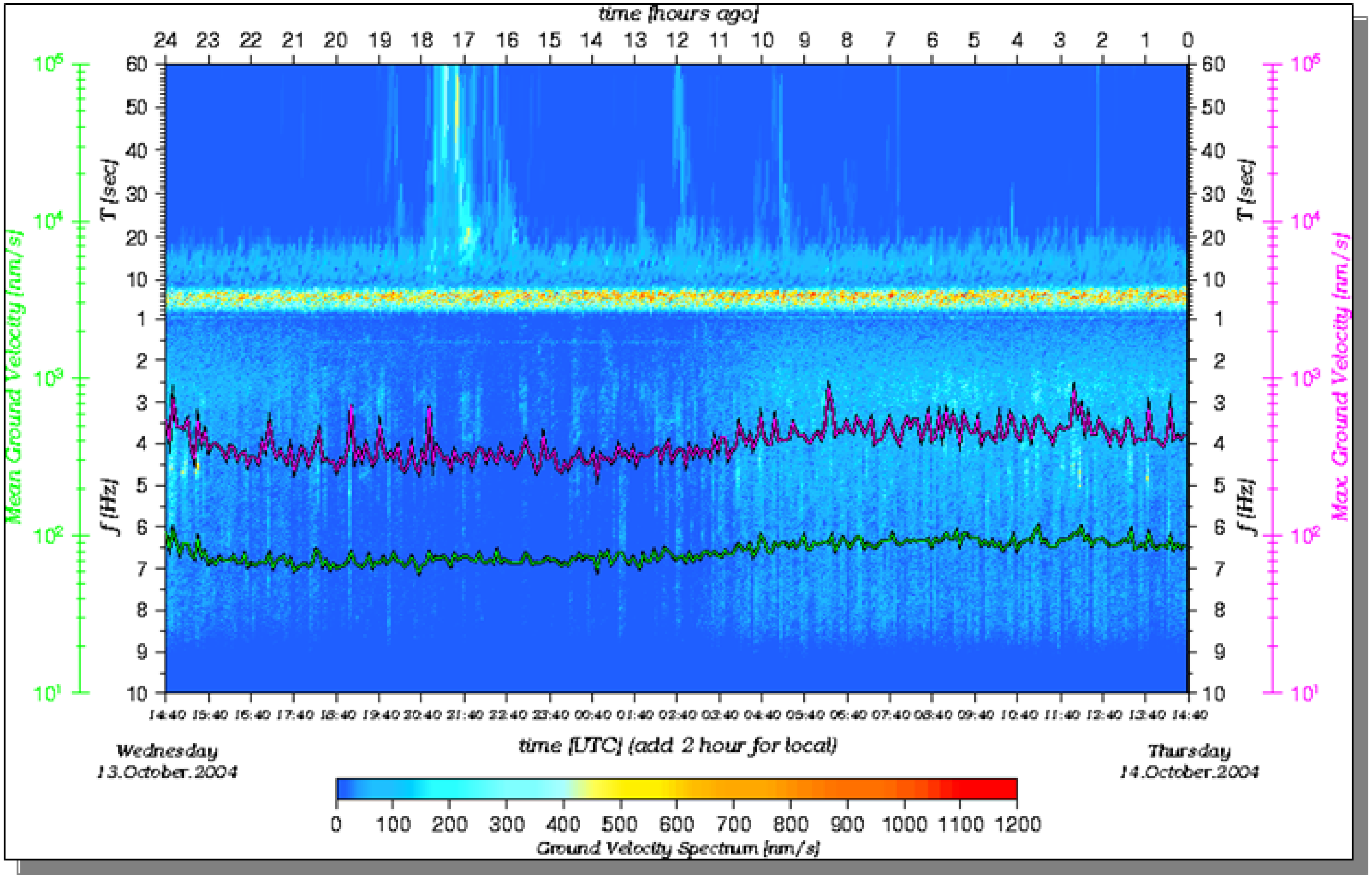


Spectral analysis: an Example



24 hour ground motion, do you see any signal?

Seismo-Weather



Running spectrum of the same data

Some properties of FT

- FT is linear

  signals can be treated as the sum of several signals, the transform will be the sum of their transforms

- FT of a real signals

  has symmetry properties

$$F(-\omega) = F^*(\omega)$$

  the negative frequencies can be obtained from symmetry properties

- Shifting corresponds to changing the phase (shift theorem)

$$f(t-a) \rightarrow e^{-i\omega a} F(\omega)$$

$$F(\omega - a) \rightarrow e^{-i\omega a} f(t)$$

- Derivative

$$\frac{d^n}{dt} f(t) \rightarrow (i\omega)^n F(\omega)$$

Fourier Derivatives

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

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

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

... 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

Acoustic Wave Equation - Fourier Method

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 ...

Acoustic Wave Equation - Fourier Method

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

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

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

multiply by 1/ρ

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

... then extrapolate ...

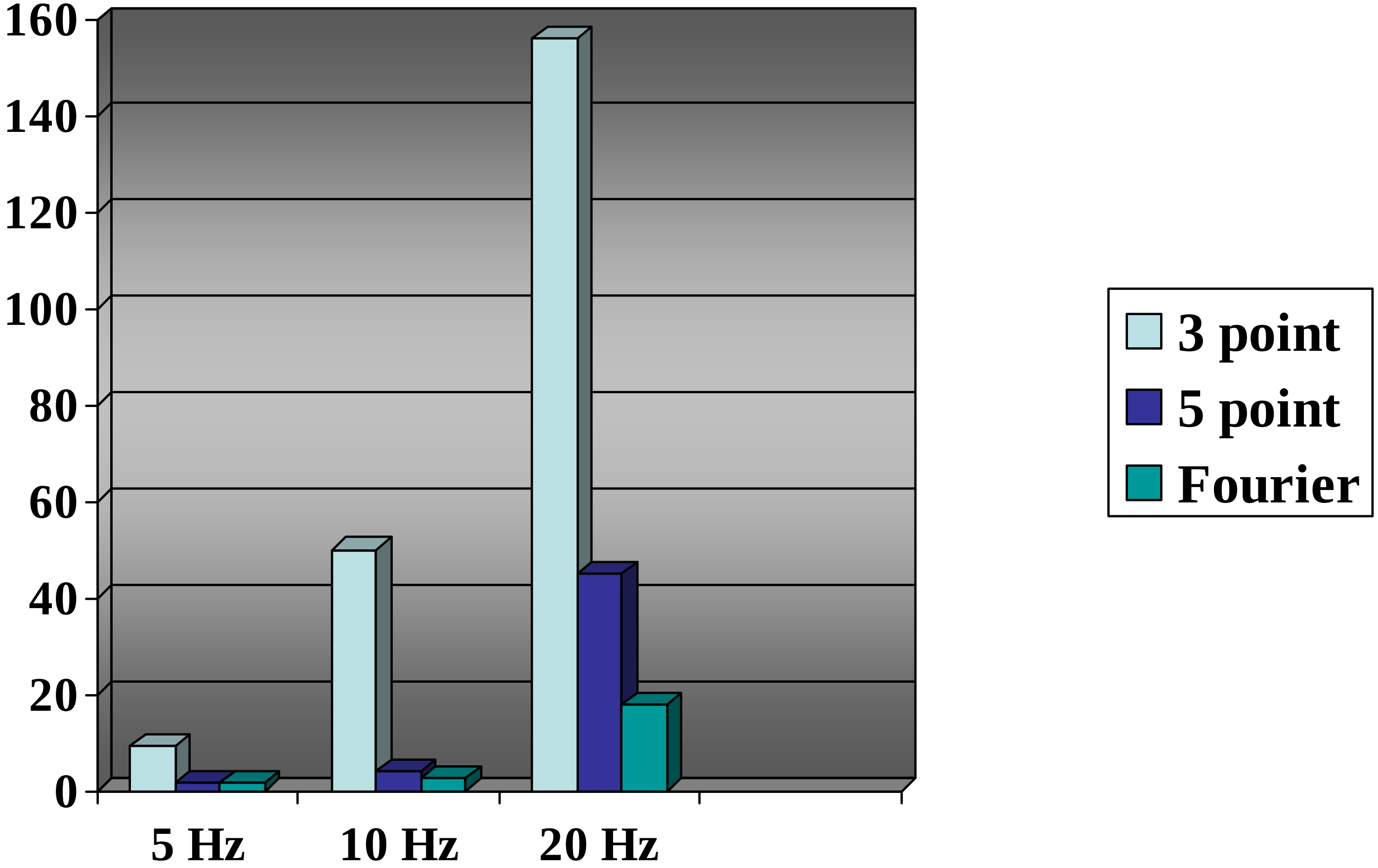
... and the first derivative using FFTs ...

```
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 ...

Fourier Method - Comparison with FD - Table

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



Simulation time
5.4s
7.8s
33.0s

Numerical solutions and Green's Functions