Spectral analysis: Foundations

- Orthogonal functions
- Fourier Series
- Discrete Fourier Series
- Fourier Transform: properties
- Chebyshev polynomials
- Convolution
- DFT and FFT

Scope: Understanding where the Fourier Transform comes from. Moving from the continuous to the discrete world. (Almost) everything we need to understand for filtering.

Fourier Series: one way to derive them

The Problem

we are trying to approximate a function f(x) by another function $g_n(x)$ which consists of a sum over N *orthogonal* functions $\Phi(x)$ weighted by some coefficients a_n.

$$f(x) \approx g_N(x) = \sum_{i=0}^{N} a_i \Phi_i(x)$$

The Problem

... and we are looking for optimal functions in a least squares (l_2) sense ...

$$\left\| f(x) - g_N(x) \right\|_{l2} = \left[\int_a^b \left\{ f(x) - g_N(x) \right\}^2 dx \right]^{1/2} = \text{Min!}$$

... a good choice for the basis functions Φ(x) are *orthogonal* functions. *What are orthogonal functions?* Two functions f and g are said to be orthogonal in the interval [a,b] if

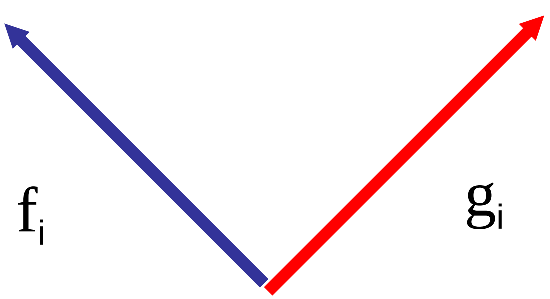
$$\int_a^b f(x) g(x) dx = 0$$

How is this related to the more conceivable concept of orthogonal vectors? Let us look at the original definition of integrals:

Orthogonal Functions

$$\int_a^b f(x)g(x)dx = \lim_{N\to\infty}\left(\sum_{i=1}^{N} f_i(x)g_i(x)\Delta x\right)$$

... where $x_0=a$ and $x_N=b$, and $x_i-x_{i-1}=\Delta x$ ...
If we interpret $f(x_i)$ and $g(x_i)$ as the ith components of an N component vector, then this sum corresponds directly to a scalar product of vectors. The vanishing of the scalar product is the condition for *orthogonality* of vectors (or functions).

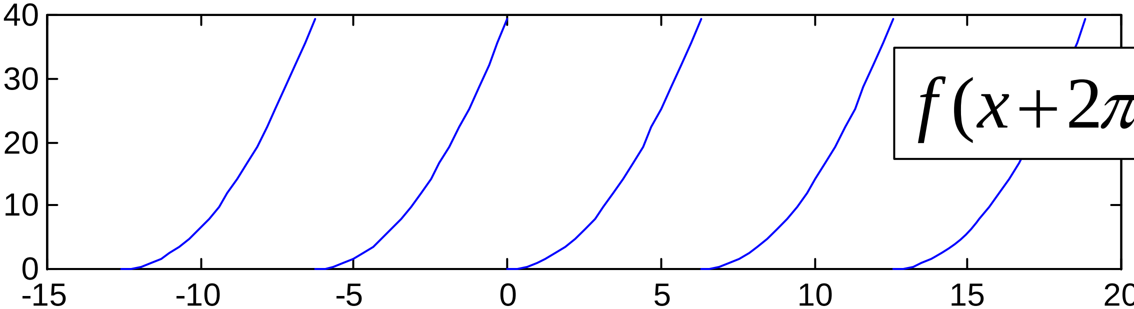


$$f_i \bullet g_i = \sum_i f_i g_i = 0$$

Periodic functions

Let us assume we have a piecewise continuous function of the form

$$f(x+2\pi) = f(x)$$



$$f(x+2\pi) = f(x) = x^2$$

... we want to approximate this function with a linear combination of 2π periodic functions:

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

$$\Rightarrow f(x) \approx g_N(x) = \frac{1}{2}a_0 + \sum_{k=1}^{N}\{a_k\cos(kx) + b_k\sin(kx)\}$$

Orthogonality

... are these functions orthogonal ?

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

$$\int_{-\pi}^{\pi} \sin(jx)\sin(kx)dx = \begin{cases} 0 & j \neq k,\ j,k > 0 \\ \pi & j = k > 0 \end{cases}$$

$$\int_{-\pi}^{\pi} \cos(jx)\sin(kx)dx = 0 \qquad j \geq 0, k > 0$$

... YES, and these relations are valid for any interval of length 2π.
Now we know that this is an orthogonal basis, but how can we obtain the coefficients for the basis functions?

$\Rightarrow$ from minimising f(x)-g(x)

Fourier coefficients

optimal functions g(x) are given if

$$\left\| g_n(x) - f(x) \right\|_2 = \text{Min}! \qquad or \qquad \frac{\partial}{\partial a_k}\left\{ \left\| g_n(x) - f(x) \right\|_2 \right\} = 0$$

... with the definition of g(x) we get ...

$$\frac{\partial}{\partial a_k}\left\| g_n(x) - f(x) \right\|_2^2 = \frac{\partial}{\partial a_k}\left[\int_{-\pi}^{\pi} \left[\frac{1}{2}a_0 + \sum_{k=1}^{N}\left\{ a_k \cos(kx) + b_k \sin(kx) \right\} - f(x) \right]^2 dx \right]$$

leading to

$$\boxed{\begin{aligned} & g_N(x) = \frac{1}{2}a_0 + \sum_{k=1}^{N}\left\{ a_k \cos(kx) + b_k \sin(kx) \right\} \qquad \text{with} \\ & a_k = \frac{1}{\pi}\int_{-\pi}^{\pi} f(x)\cos(kx)\,dx, \qquad k = 0,1,\ldots, N \\ & b_k = \frac{1}{\pi}\int_{-\pi}^{\pi} f(x)\sin(kx)\,dx, \qquad k = 1,2,\ldots, N \end{aligned}}$$

Fourier approximation of |x|

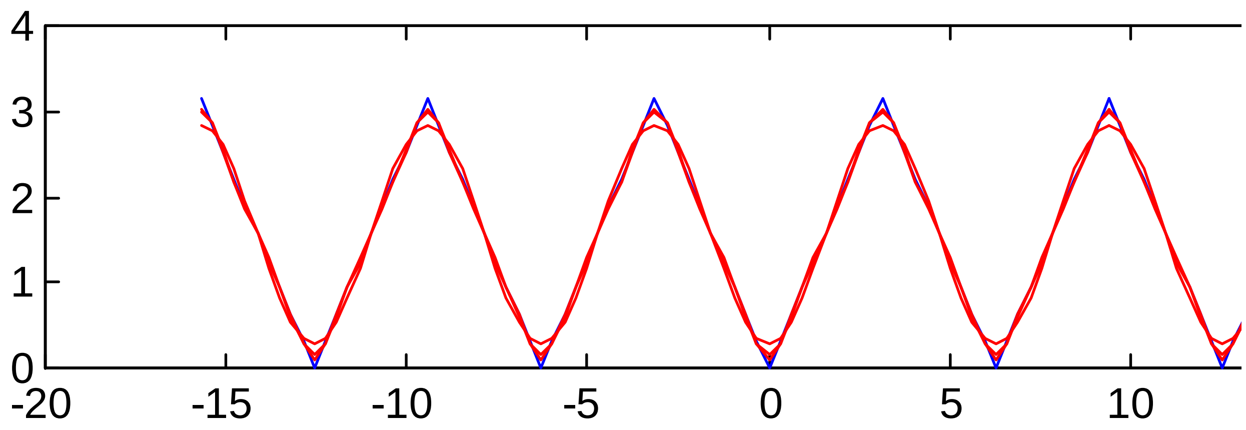
... Example ...

$$f(x) = |x|, \qquad -\pi \leq x \leq \pi$$

leads to the Fourier Serie

$$g(x) = \frac{1}{2}\pi - \frac{4}{\pi}\left\{ \frac{\cos(x)}{1^2} + \frac{\cos(3x)}{3^2} + \frac{\cos(5x)}{5^2} + \ldots \right\}$$

.. and for n<4 g(x) looks like



Fourier approximation of x^2

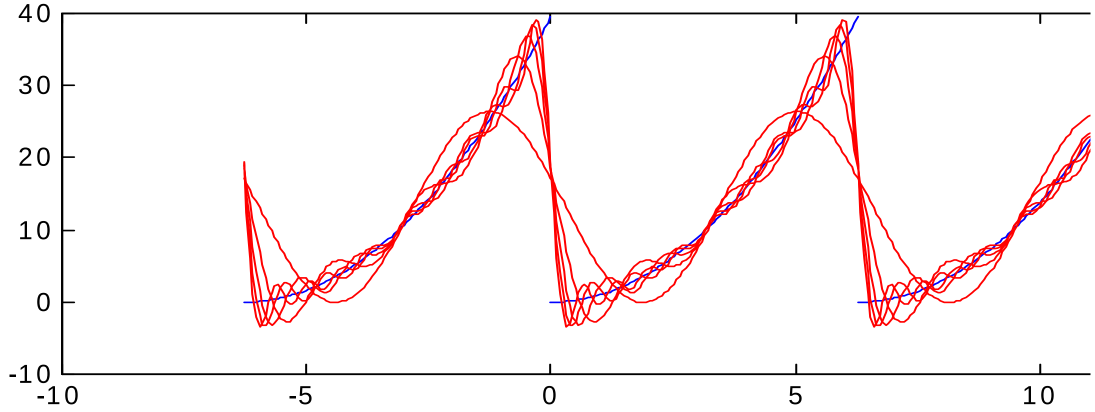
... another Example ...

$$f(x) = x^2, \qquad 0 < x < 2\pi$$

leads to the Fourier Serie

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

.. and for N<11, g(x) looks like



Fourier - discrete functions

... what happens if we know our function f(x) only at the points

$$x_i = \frac{2\pi}{N} i$$

it turns out that in this *particular* case the coefficients are given by

$$a_k^* = \frac{2}{N} \sum_{j=1}^{N} f(x_j) \cos(kx_j), \qquad k = 0,1,2,...$$

$$b_k^* = \frac{2}{N} \sum_{j=1}^{N} f(x_j) \sin(kx_j), \qquad k = 1,2,3,...$$

.. the so-defined Fourier polynomial is the unique interpolating function to the function $f(x_j)$ with N=2m

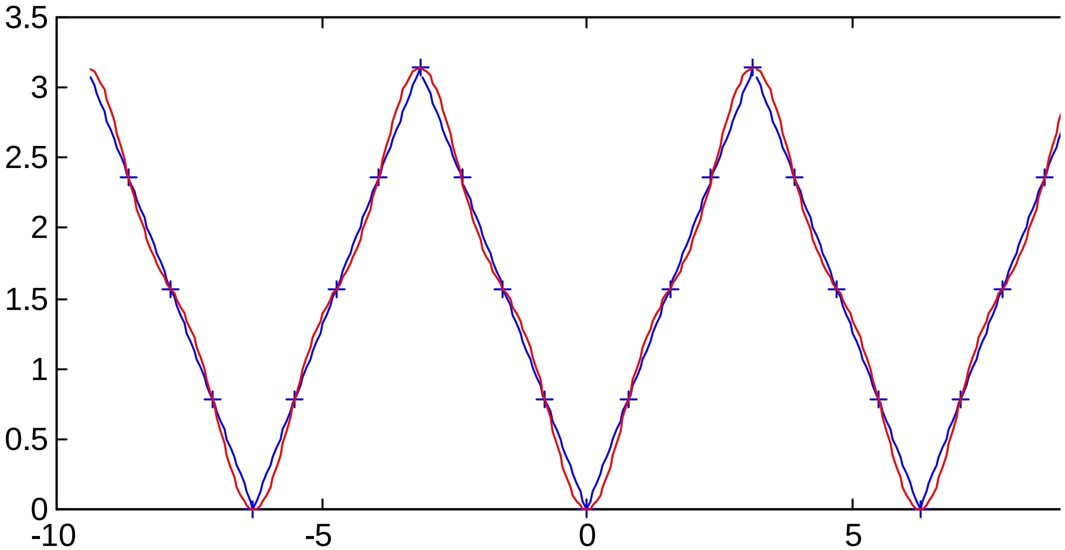
$$g_m^*(x) = \frac{1}{2} a_0^* + \sum_{k=1}^{m-1} \left\{ a_k^* \cos(kx) + b_k^* \sin(kx) \right\} + \frac{1}{2} a_m^* \cos(kx)$$

Fourier - collocation points

... with the important property that ...

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

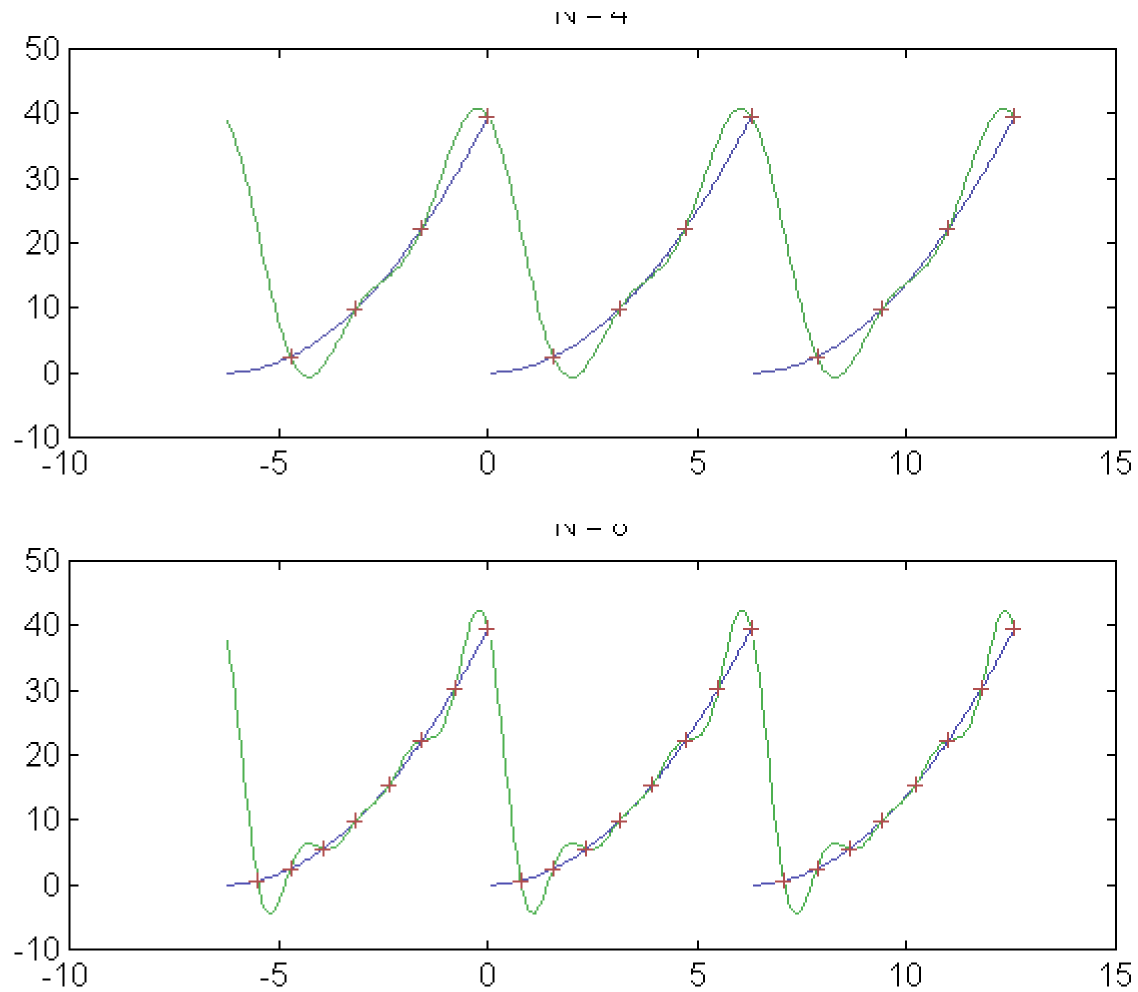
... in our previous examples ...



f(x)=|x| => f(x) - blue ; g(x) - red; x_i - ‘+’

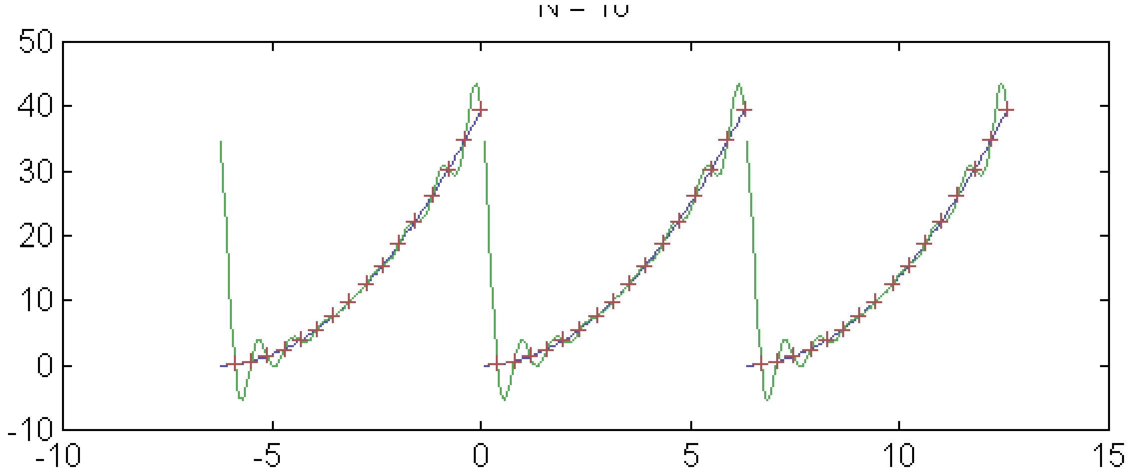
Fourier series - convergence

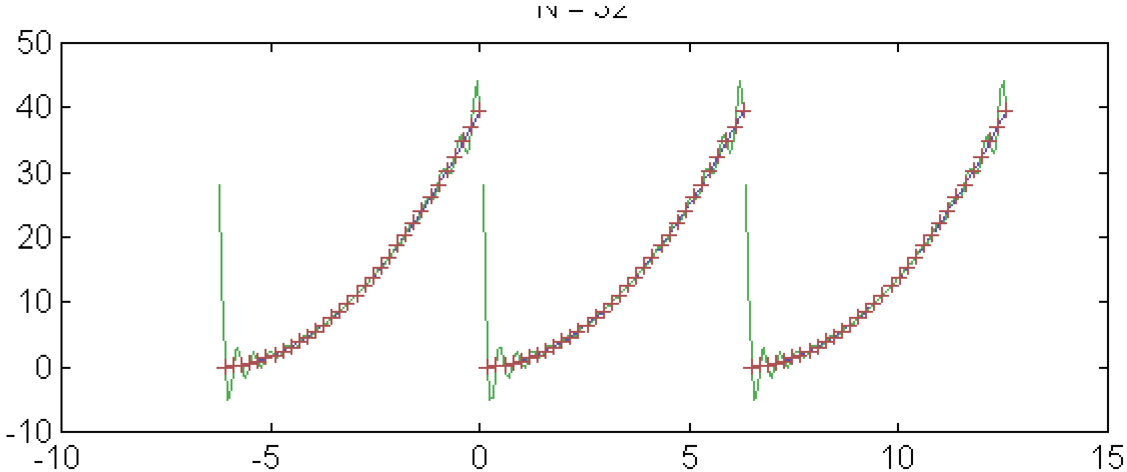
$f(x)=x^2$ => $f(x)$ - blue ; $g(x)$ - red; x_i - '+'



Fourier series - convergence

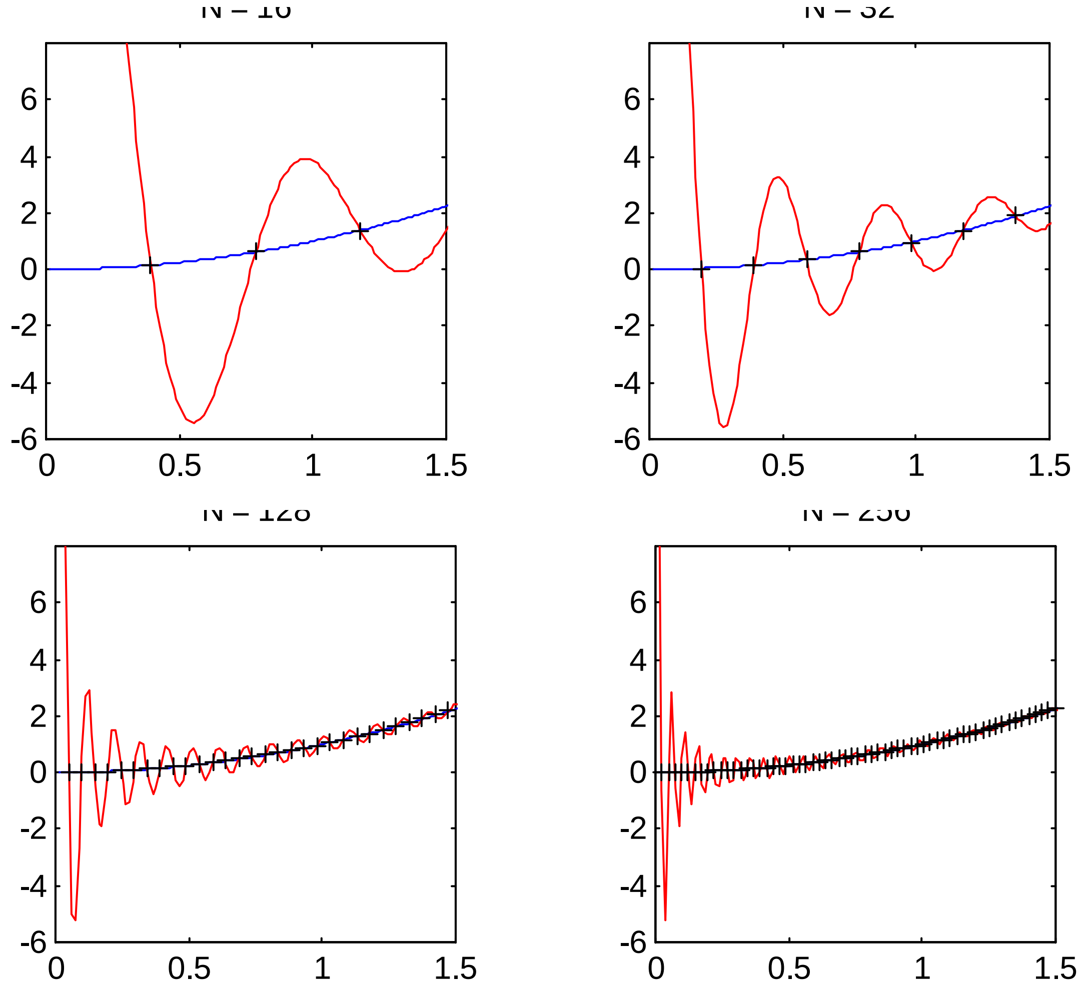
$f(x)=x^2$ => f(x) - blue ; g(x) - red; x_i - '+'





Gibb's phenomenon

$f(x)=x^2$ => f(x) - blue ; g(x) - red; x_i - '+'



The overshoot for equi-spaced Fourier interpolations is ≈14% of the step height.

Chebyshev polynomials

We have seen that Fourier series are excellent for interpolating (and differentiating) periodic functions defined on a regularly spaced grid. In many circumstances physical phenomena which are not periodic (in space) and occur in a *limited area.* This quest leads to the use of **Chebyshev polynomials**.

We depart by observing that $\cos(n\varphi)$ can be expressed by a polynomial in $\cos(\varphi)$:

$$\cos(2\varphi) = 2\cos^2\varphi - 1$$

$$\cos(3\varphi) = 4\cos^3\varphi - 3\cos\varphi$$

$$\cos(4\varphi) = 8\cos^4\varphi - 8\cos^2\varphi + 1$$

... which leads us to the definition:

Chebyshev polynomials - definition

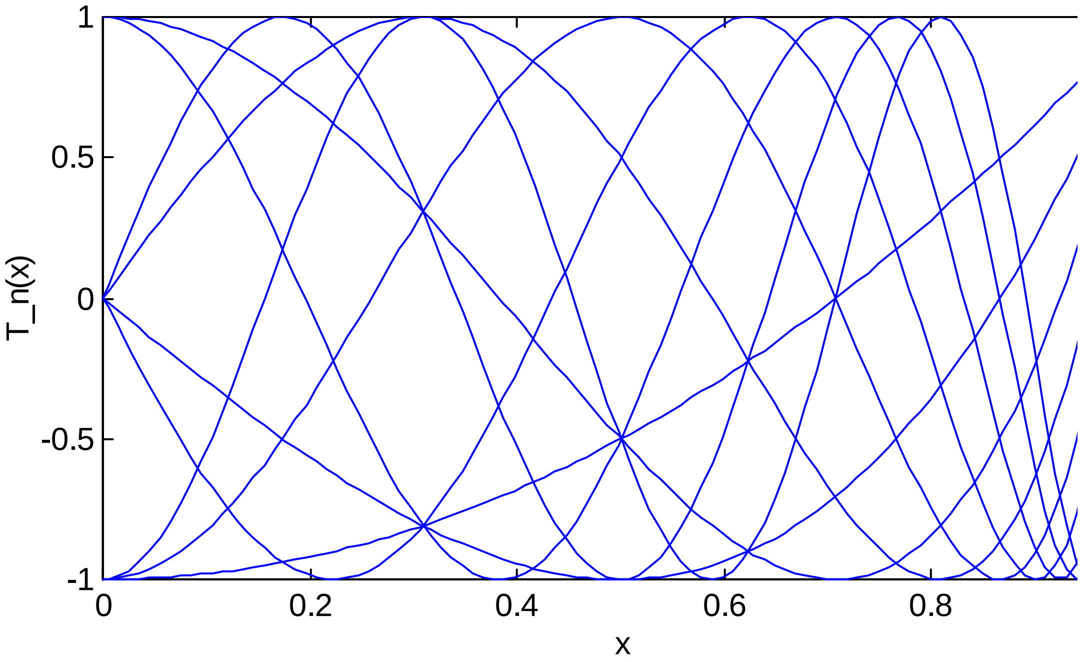
$$\cos(n\varphi) = T_n(\cos(\varphi)) = T_n(x), \qquad x = \cos(\varphi), \qquad x \in [-1,1], \qquad n \in N$$

... for the Chebyshev polynomials $T_n(x)$. Note that because of x=cos(φ) they are defined in the interval [-1,1] (which - however - can be extended to $\Re$). The first polynomials are

$$T_0(x) = 1$$
$$T_1(x) = x$$
$$T_2(x) = 2x^2 - 1$$
$$T_3(x) = 4x^3 - 3x$$
$$T_4(x) = 8x^4 - 8x^2 + 1 \qquad \text{where}$$
$$|T_n(x)| \le 1 \quad \text{for} \quad x \in [-1,1] \quad \text{and} \quad n \in N_0$$

Chebyshev polynomials - Graphical

The first ten polynomials look like [0, -1]

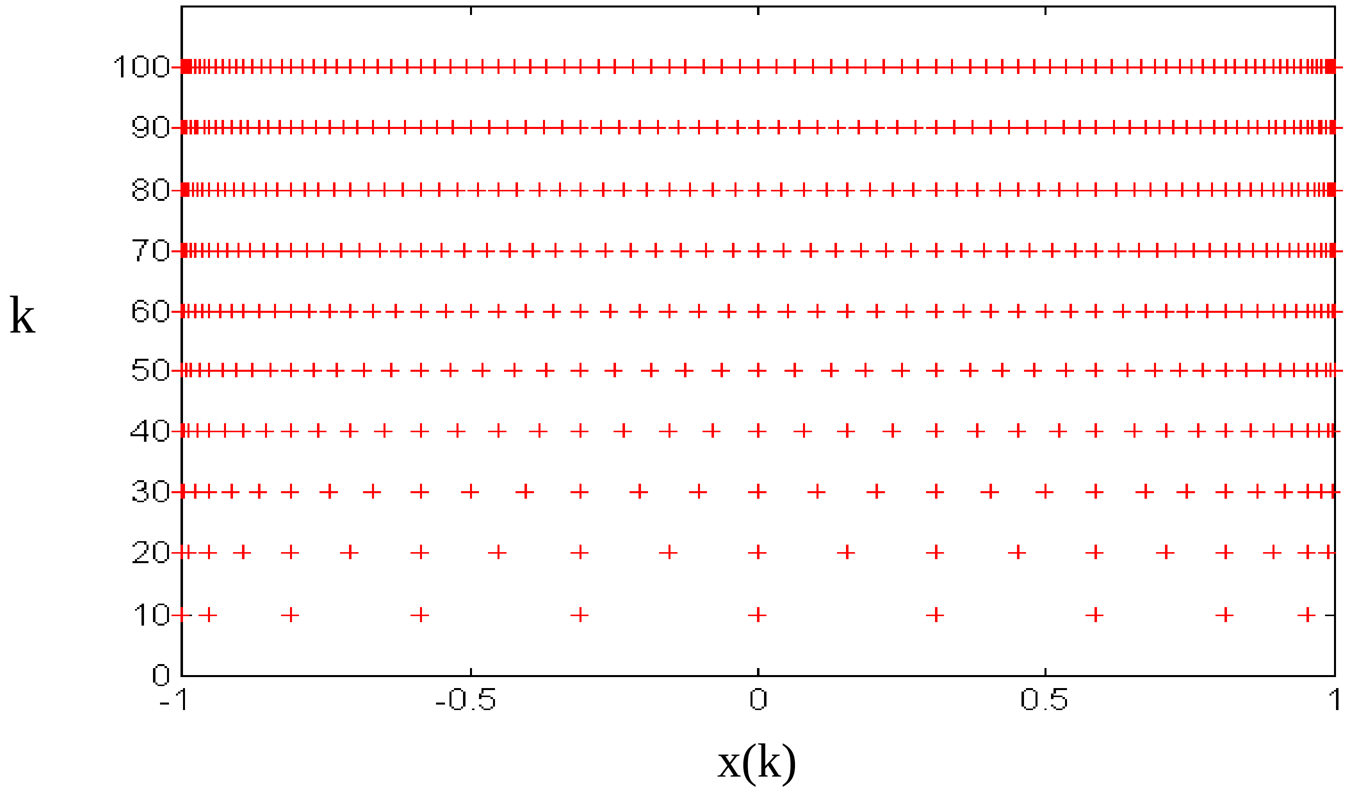


The n-th polynomial has extrema with values 1 or -1 at

$$x_k^{(ext)} = \cos(\frac{k\pi}{n}), \qquad k = 0,1,2,3,\ldots, n$$

Chebyshev collocation points

These extrema are not equidistant (like the Fourier extrema)



$$x_k^{(ext)} = \cos\left(\frac{k\pi}{n}\right), \qquad k = 0,1,2,3,\ldots, n$$

Chebyshev polynomials - orthogonality

... are the Chebyshev polynomials orthogonal?

Chebyshev polynomials are an orthogonal set of functions in the interval [-1,1] with respect to the weight function $1/\sqrt{1-x^2}$ *such that*

$$\int_{-1}^{1} T_k(x) T_j(x) \frac{dx}{\sqrt{1-x^2}} = \left\{ \begin{array}{ccc} 0 & for & k \neq j \\ \pi/2 & for & k = j > 0 \\ \pi & for & k = j = 0 \end{array} \right\}, \qquad k, j \in N_0$$

... this can be easily verified noting that

$$x = \cos\varphi, \qquad dx = -\sin\varphi d\varphi$$

$$T_k(x) = \cos(k\varphi), \quad T_j(x) = \cos(j\varphi)$$

Chebyshev polynomials - interpolation

... we are now faced with the same problem as with the Fourier series. We want to approximate a function f(x), this time not a periodical function but a function which is defined between [-1,1]. We are looking for $g_n(x)$

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

... and we are faced with the problem, how we can determine the coefficients c_k. Again we obtain this by finding the extremum (minimum)

$$\frac{\partial}{\partial c_k} \left[\int_{-1}^{1} \left\{ g_n(x) - f(x) \right\}^2 \frac{dx}{\sqrt{1-x^2}} \right] = 0$$

Chebyshev polynomials - interpolation

... to obtain ...

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

... surprisingly these coefficients can be calculated with FFT techniques, noting that

$$c_k = \frac{2}{\pi}\int_{0}^{\pi} f(\cos\varphi)\cos k\varphi d\varphi, \qquad k = 0,1,2,..., n$$

... and the fact that f(cosφ) is a 2π-periodic function ...

$$c_k = \frac{1}{\pi}\int_{-\pi}^{\pi} f(\cos\varphi)\cos k\varphi d\varphi, \qquad k = 0,1,2,..., n$$

... which means that the coefficients c_k are the Fourier coefficients a_k of the periodic function F(φ)=f(cos φ)!

Chebyshev - discrete functions

... what happens if we know our function f(x) only at the points

$$x_i = \cos \frac{\pi}{N} i$$

in this *particular* case the coefficients are given by

$$c_k^* = \frac{2}{N} \sum_{j=1}^{N} f(\cos \varphi_j) \cos(k\varphi_j), \qquad k = 0,1,2,\ldots N/2$$

... leading to the polynomial ...

$$g_m^*(x) = \frac{1}{2} c_0^* T_0 + \sum_{k=1}^{m} c_k^* T_k(x)$$

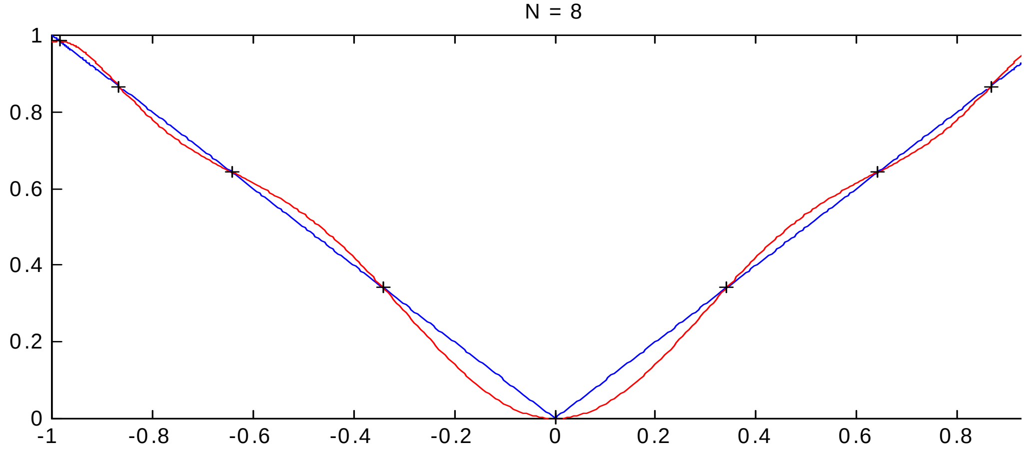
... with the property

$$g_m^*(x) = f(x) \quad \text{at} \quad x_j = \cos(\pi j/N) \qquad j = 0,1,2,\ldots, \quad N$$

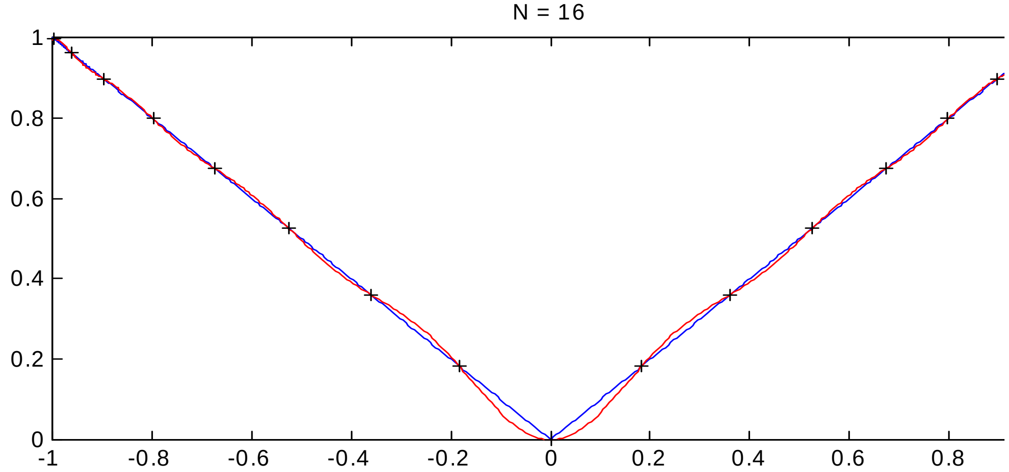
Chebyshev - collocation points - |x|

f(x)=|x| => f(x) - blue ; g_n(x) - red; x_i - '+'

8 points



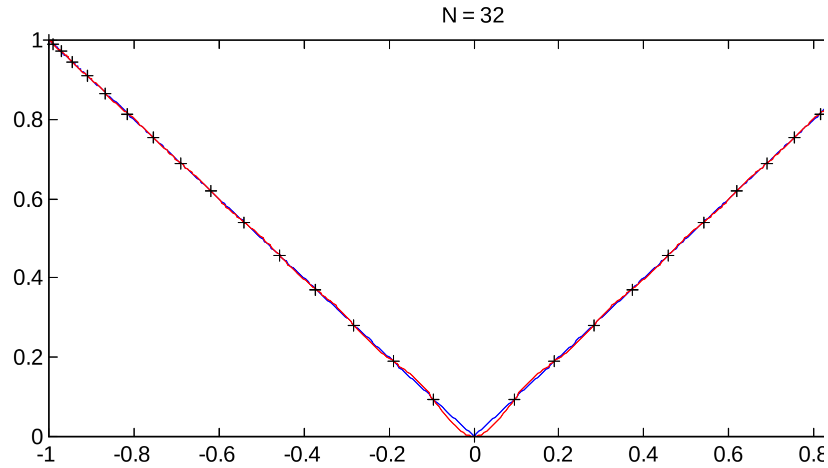
16 points



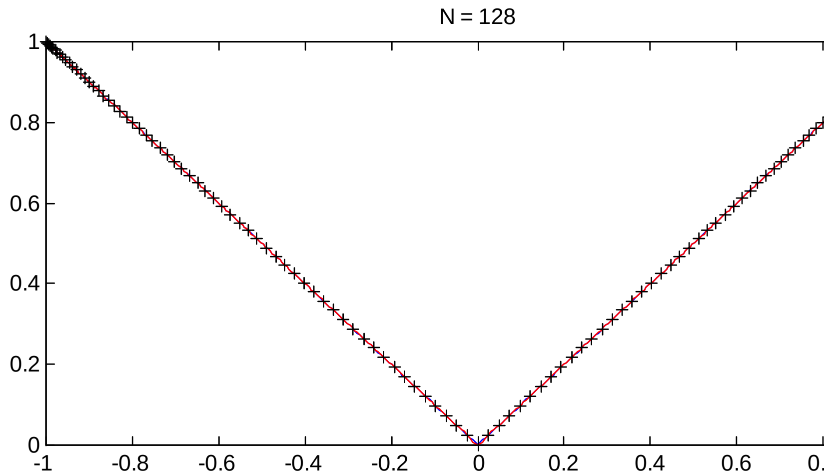
Chebyshev - collocation points - |x|

f(x)=|x| => f(x) - blue ; $g_n(x)$ - red; x_i - '+'

32 points



128 points

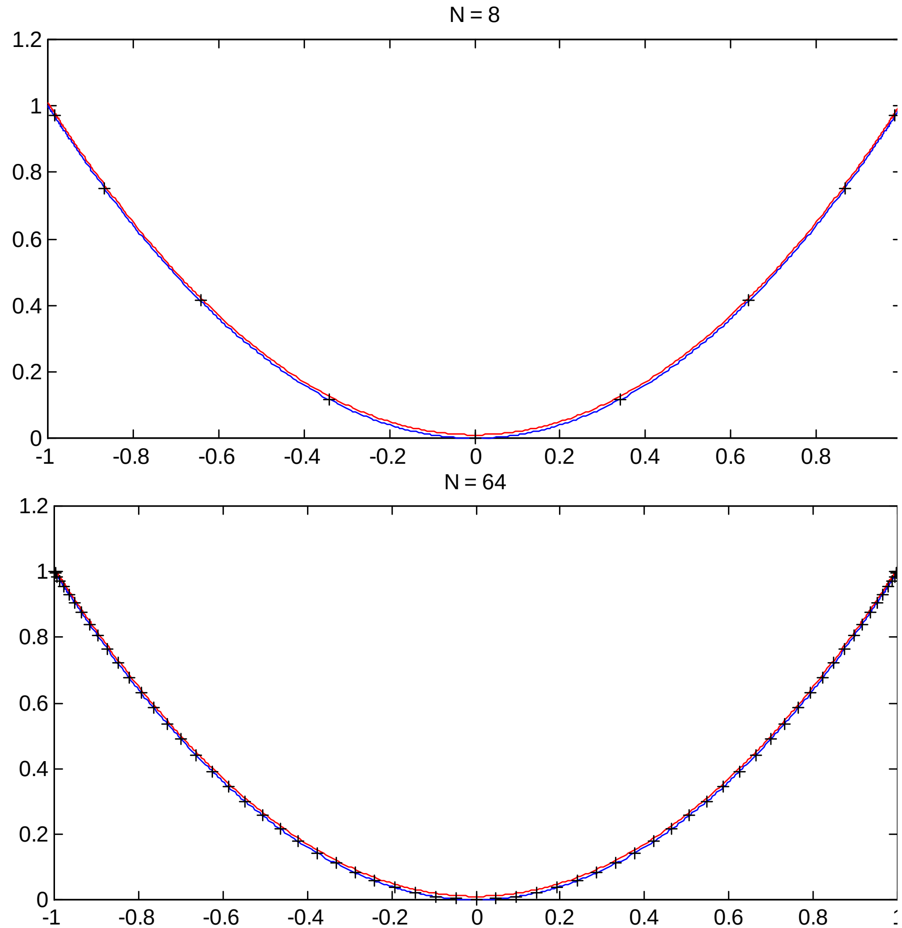


Chebyshev - collocation points - x^2

$f(x)=x^2$ => $f(x)$ - blue ; $g_n(x)$ - red; x_i - '+'

8 points

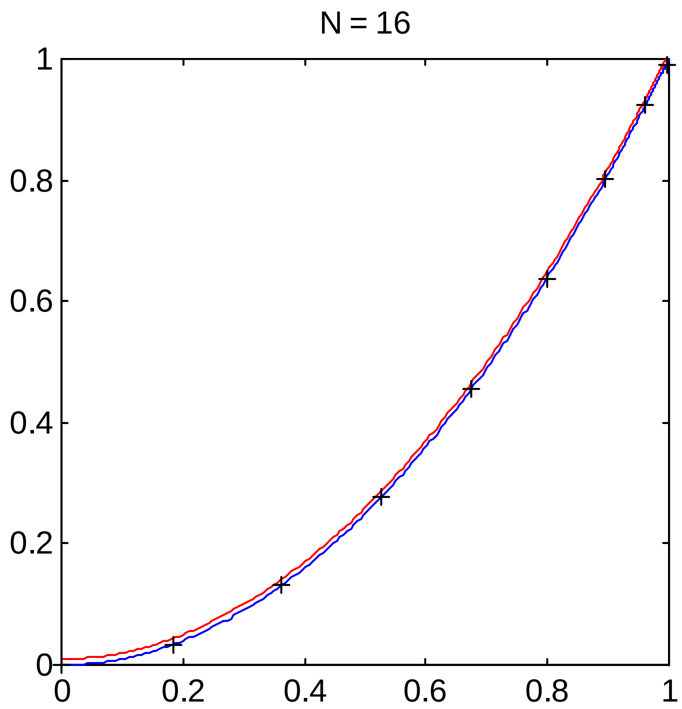
64 points



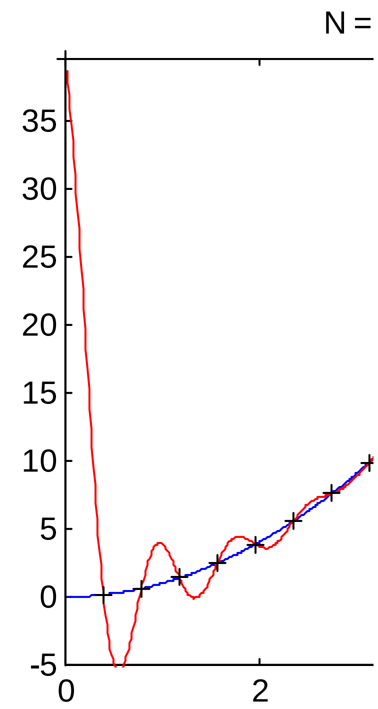
The interpolating function $g_n(x)$ was shifted by a small amount to be visible at all!

Chebyshev vs. Fourier - numerical

Chebyshev



Fourier

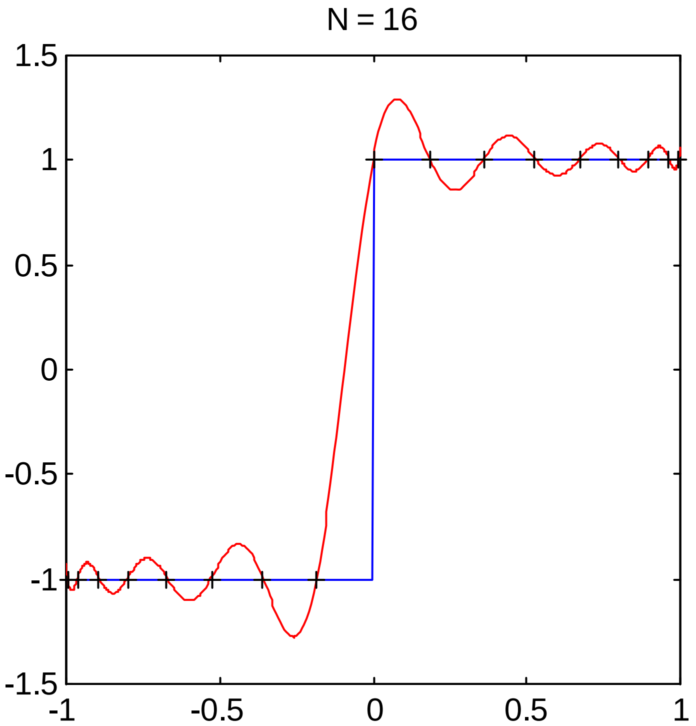


$f(x)=x^2$ => $f(x)$ - blue ; $g_N(x)$ - red; x_i - '+'

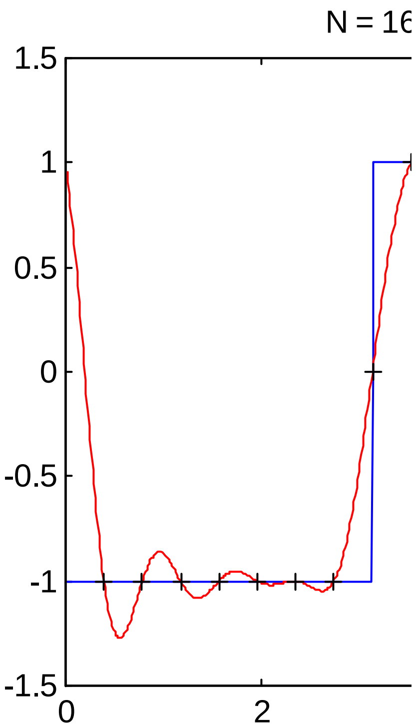
This graph speaks for itself ! Gibb's phenomenon with Chebyshev?

Chebyshev vs. Fourier - Gibb's

Chebyshev



Fourier

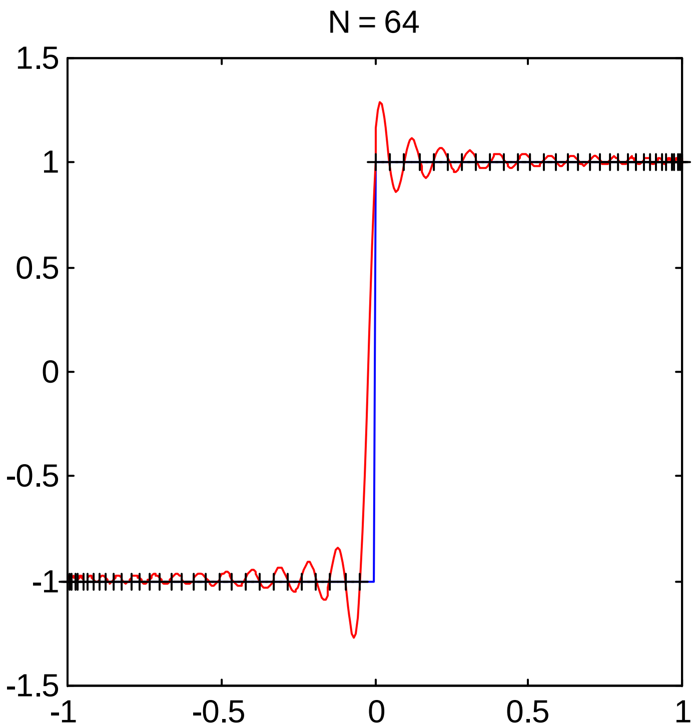


$f(x)=\mathrm{sign}(x-\pi)$ => f(x) - blue ; $g_N(x)$ - red; x_i - '+'

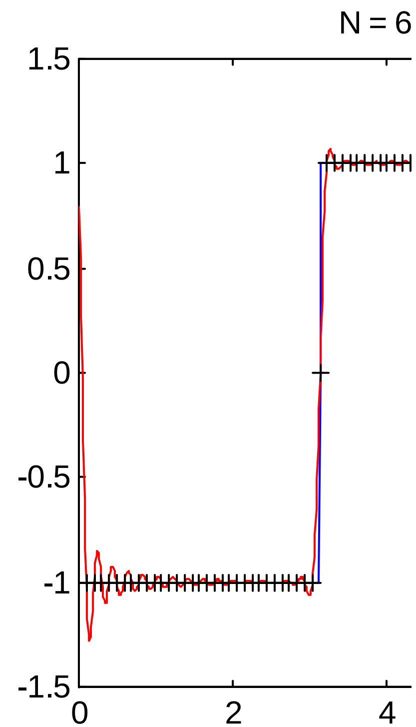
Gibb's phenomenon with Chebyshev? YES!

Chebyshev vs. Fourier - Gibb’s

Chebyshev



Fourier



$f(x)=\mathrm{sign}(x-\pi)$ => $f(x)$ - blue ; $g_N(x)$ - red; x_i - ‘+’

Fourier vs. Chebyshev

Fourier		Chebyshev
$x_i = \frac{2\pi}{N} i$	*collocation points*	$x_i = \cos \frac{\pi}{N} i$
periodic functions	*domain*	limited area [-1,1]
$\cos(nx), \sin(nx)$	*basis functions*	$T_n(x) = \cos(n\varphi),$ $x = \cos\varphi$
$g_m^*(x) = \frac{1}{2}a_0^* + \sum_{k=1}^{m-1}\left\{a_k^* \cos(kx) + b_k^* \sin(kx)\right\} + \frac{1}{2}a_m^* \cos(kx)$	*interpolating function*	$g_m^*(x) = \frac{1}{2}c_0^* T_0 + \sum_{k=1}^{m} c_k^* T_k(x)$

Fourier vs. Chebyshev (cont'd)

Fourier

$$a_k^* = \frac{2}{N}\sum_{j=1}^{N} f(x_j)\cos(kx_j)$$

$$b_k^* = \frac{2}{N}\sum_{j=1}^{N} f(x_j)\sin(kx_j)$$

- Gibb's phenomenon for discontinuous functions
- Efficient calculation via FFT
- infinite domain through periodicity

coefficients

some properties

Chebyshev

$$c_k^* = \frac{2}{N}\sum_{j=1}^{N} f(\cos\varphi_j)\cos(k\varphi_j)$$

- limited area calculations
- grid densification at boundaries
- coefficients via FFT
- excellent convergence at boundaries
- Gibb's phenomenon

The Fourier Transform Pair

$$F(\omega) = \frac{1}{2\pi} \int_{-\infty}^{\infty} f(t) e^{i\omega x} dt$$

Forward transform

$$f(t) = \int_{-\infty}^{\infty} F(\omega) e^{-i\omega x} dt$$

Inverse transform

Note the conventions concerning the sign of the exponents and the factor.

The Fourier Transform Pair

$$F(\omega) = R(\omega) + iI(\omega) = A(\omega)e^{i\Phi(\omega)}$$

$$A(\omega) = \left|F(\omega)\right| = \sqrt{R^2(\omega) + I^2(\omega)}$$

$$\Phi(\omega) = \arg F(\omega) = \arctan\frac{I(\omega)}{R(\omega)}$$

$A(\omega)$	**Amplitude spectrum**
$\Phi(\omega)$	Phase spectrum

In most application it is the amplitude (or the power) spectrum that is of interest.

The Fourier Transform: when does it work?

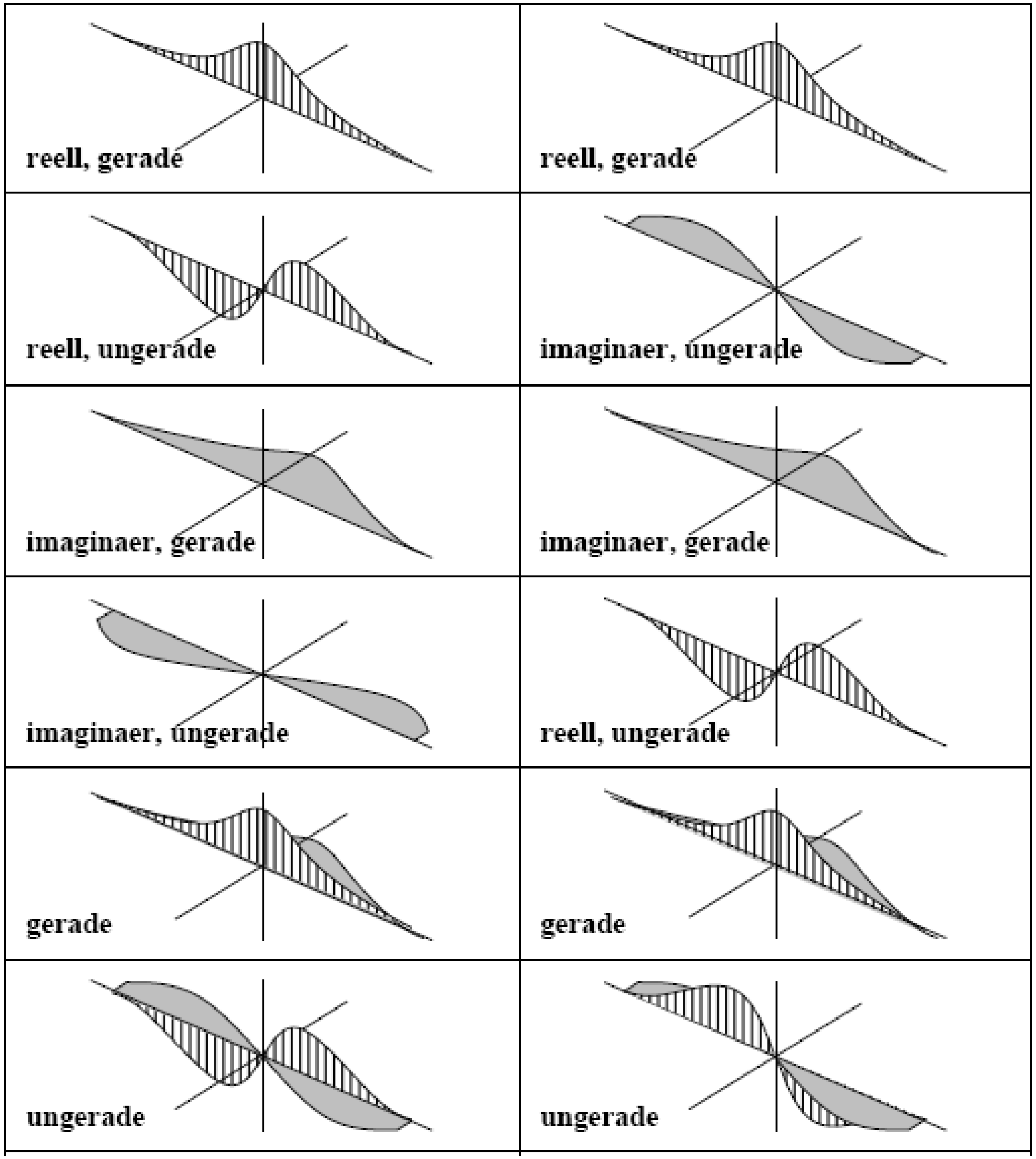
Conditions that the integral transforms work:

- f(t) has a finite number of jumps and the limits exist from both sides
- f(t) is integrable, i.e. $\int_{-\infty}^{\infty} |f(t)| dt = G < \infty$

Properties of the Fourier transform for special functions:

Function f(t)	Fouriertransform F(ω)
even	even
odd	odd
real	hermitian
imaginary	antihermitian
hermitian	real

… graphically …



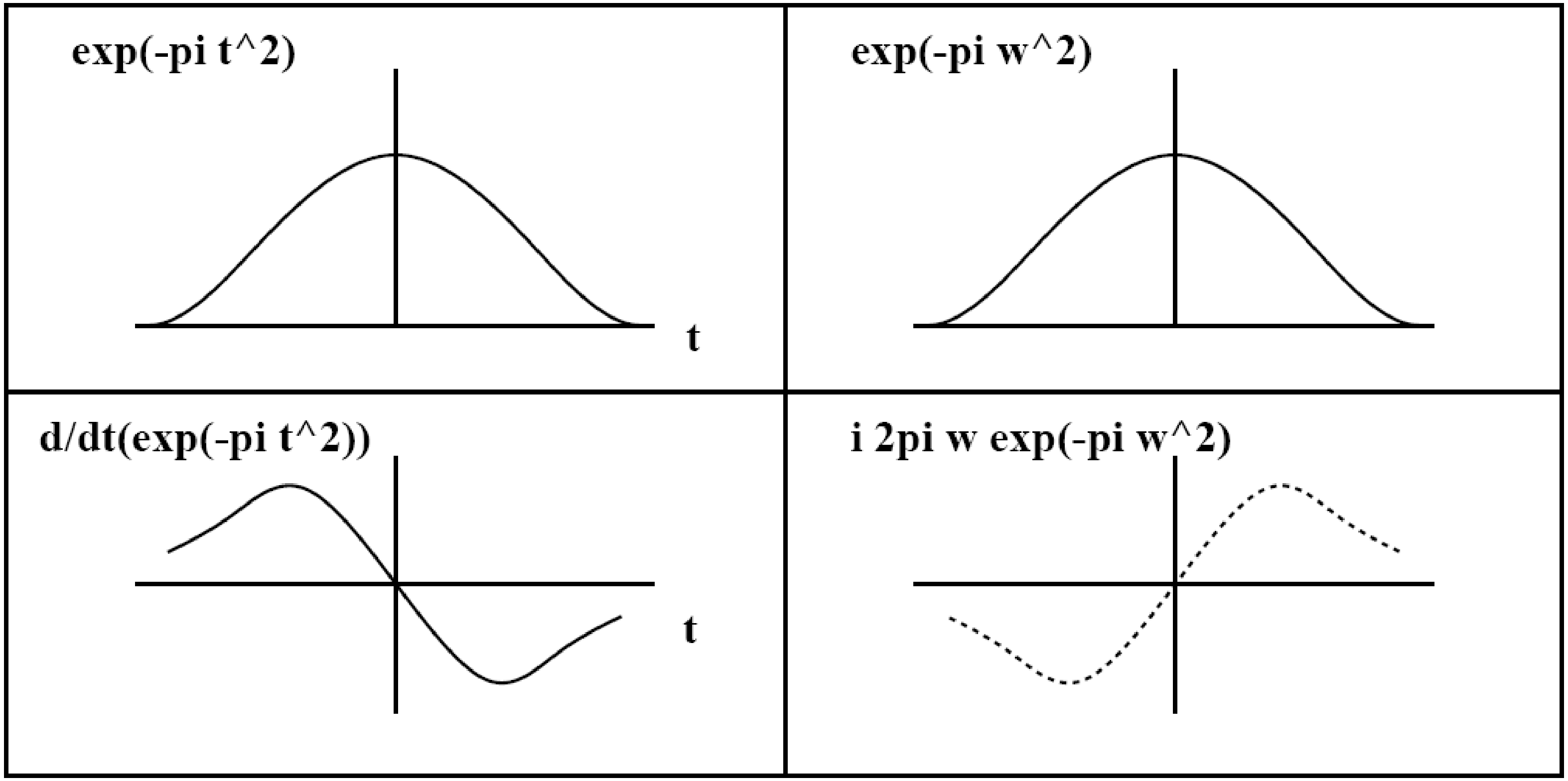
Some properties of the Fourier Transform

Defining as the FT: $f(t) \Rightarrow F(\omega)$

- Linearity $$af_1(t) + bf_2(t) \Rightarrow aF_1(\omega) + bF_2(\omega)$$
- Symmetry $$f(-t) \Rightarrow 2\pi F(-\omega)$$
- Time shifting $$f(t + \Delta t) \Rightarrow e^{i\omega\Delta t} F(\omega)$$
- Time differentiation $$\frac{\partial^n f(t)}{\partial t^n} \Rightarrow (-i\omega)^n F(\omega)$$

Differentiation theorem

- Time differentiation $$\frac{\partial^n f(t)}{\partial t^n} \Rightarrow (-i\omega)^n F(\omega)$$



Convolution

The **convolution** operation is at the heart of **linear systems.**

Definition:

$$f(t) * g(t) = \int_{-\infty}^{\infty} f(t')g(t-t')dt' = \int_{-\infty}^{\infty} f(t-t')g(t')dt'$$

Properties:

$$f(t) * g(t) = g(t) * f(t)$$

$$f(t) * \partial(t) = f(t)$$

$$f(t) * H(t) = \int f(t)dt$$

H(t) is the Heaviside function:

The convolution theorem

A convolution in the time domain corresponds to a multiplication in the frequency domain.

… and vice versa …

a convolution in the frequency domain corresponds to a multiplication in the time domain

$$f(t) * g(t) \Rightarrow F(\omega)G(\omega)$$

$$f(t)g(t) \Rightarrow F(\omega) * G(\omega)$$

The first relation is of tremendous practical implication!

The convolution theorem

From Bracewell (Fourier transforms)

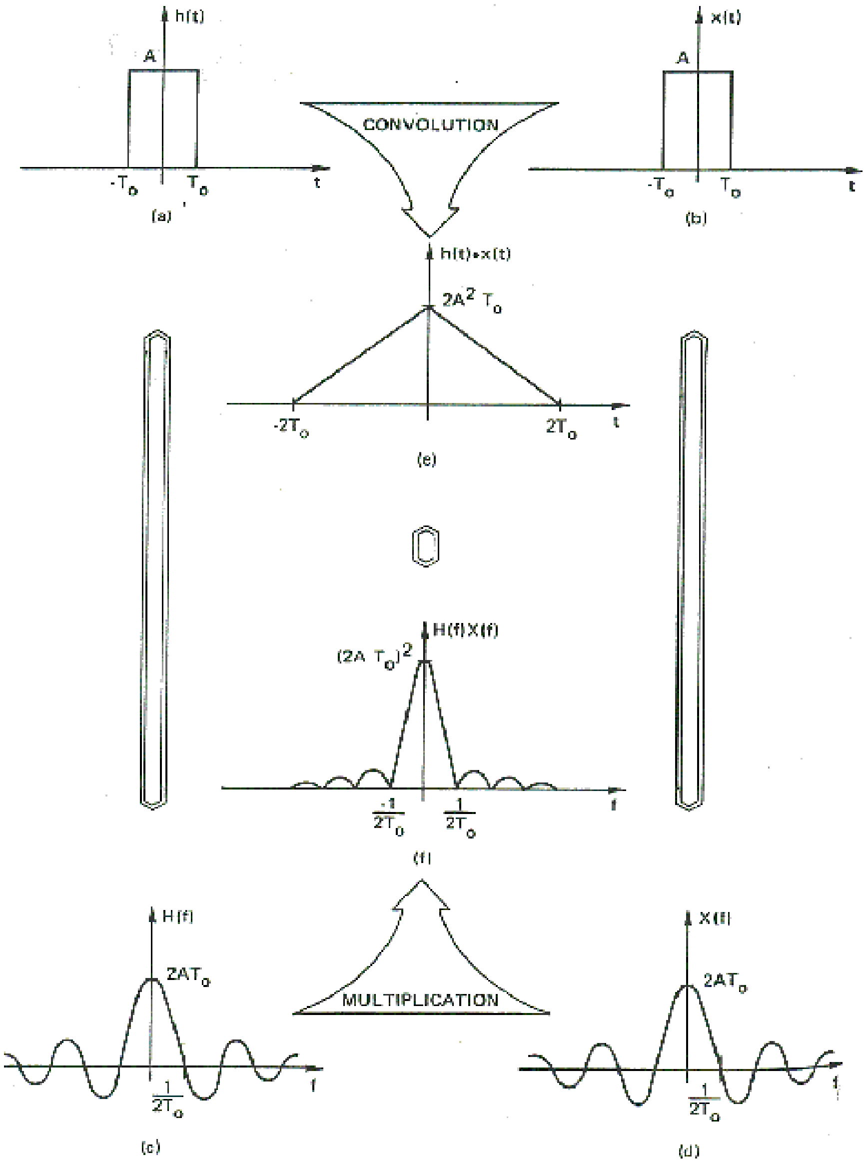


Figure 4.7: Graphical example of the convolution theorem

Discrete Convolution

Convolution is the mathematical description of the change of waveform shape after passage through a filter (system).

There is a special mathematical symbol for convolution (*):

$$y(t) = g(t) * f(t)$$

Here the impulse response function g is convolved with the input signal f. g is also named the „Green's function"

$$y_k = \sum_{i=0}^{m} g_i f_{k-i}$$

$$k = 0,1,2,\ldots,m+n$$

$$g_i \qquad i = 0,1,2,\ldots,m$$

$$f_j \qquad j = 0,1,2,\ldots,n$$

Convolution Example(Matlab)

```
>> x
x =
    0    0    1    0
>> y
y =
    1    2    1
>> conv(x,y)
ans =
    0    0    1    2    1    0
```

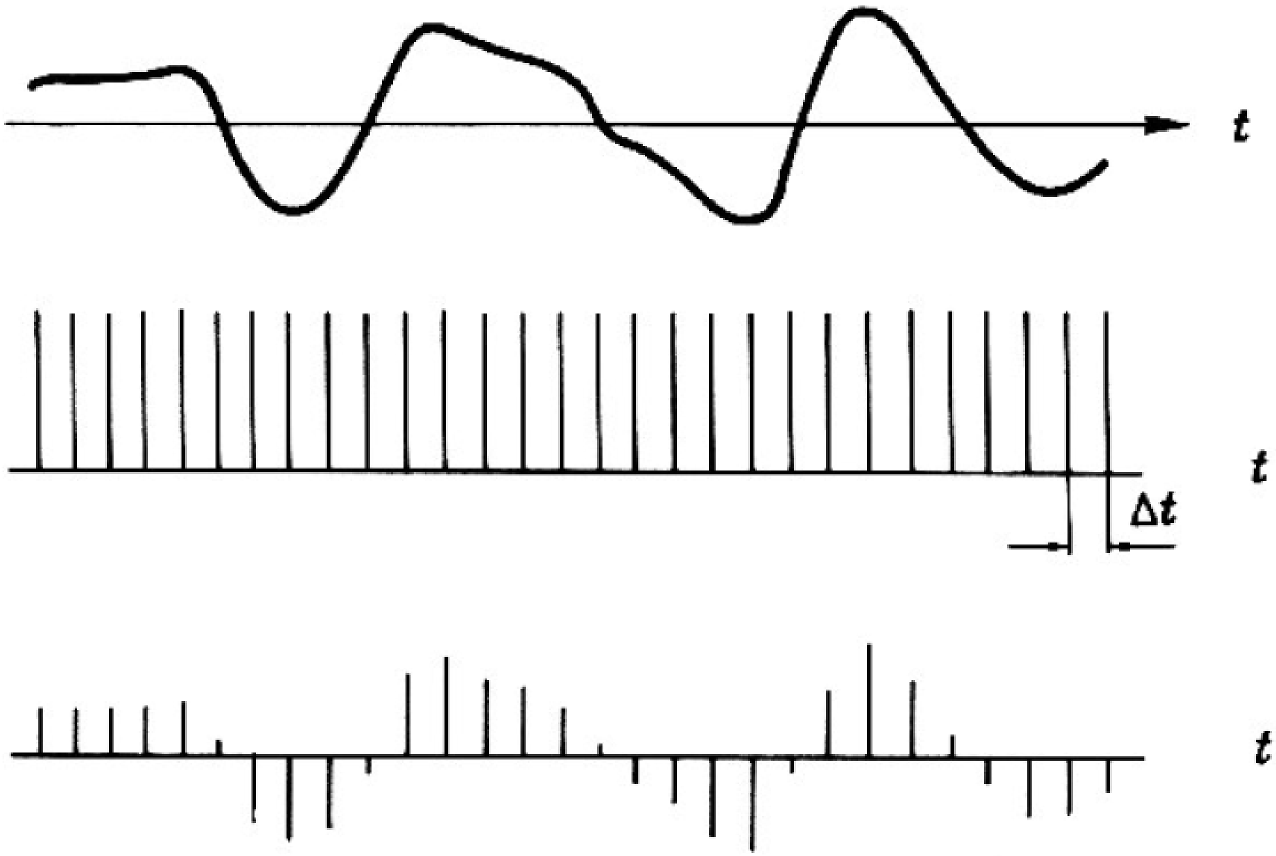
Impulse response

System input

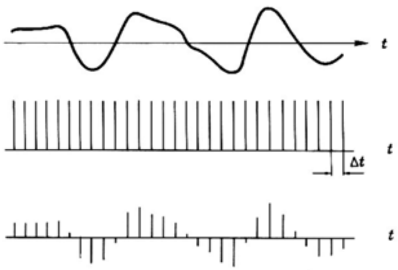
System output

Convolution Example (pictorial)

The digital world



The digital world



$$g_s(t) = g(t) \sum_{j=-\infty}^{\infty} \delta(t - jdt)$$

g_s is the digitized version of g and the sum is called the *comb function.* Defining the Nyquist frequency f_{Ny} as

$$f_{Ny} = \frac{1}{2dt}$$

after a few operations the spectrum can be written as

$$G_s(f) = \frac{1}{dt}\left\{G(f) + \sum_{n=1}^{\infty}\left[G(f - 2nf_{Ny}) + G(f + 2nf_{Ny})\right]\right\}$$

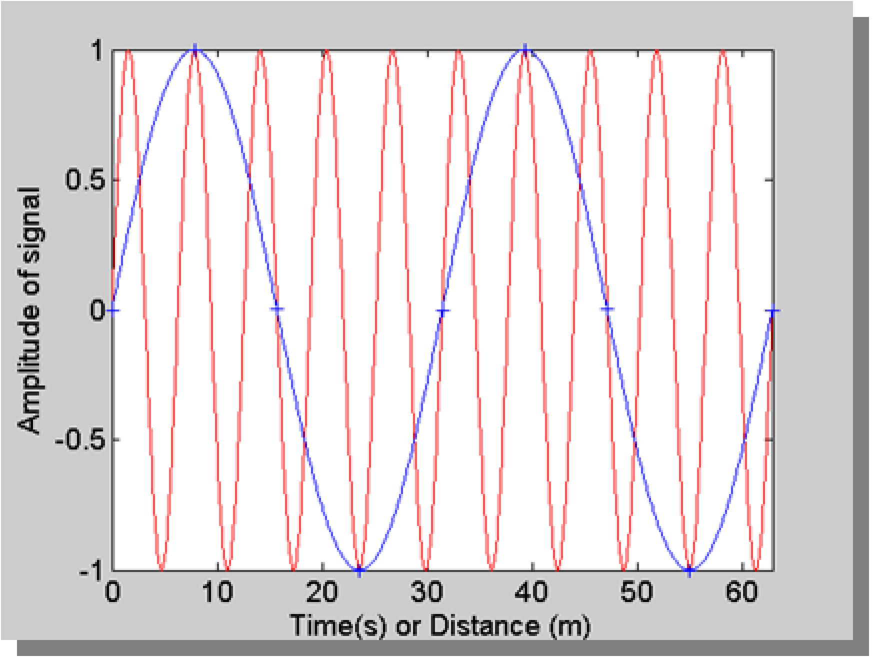
… with very important consequences …

The sampling theorem

The implications are that for the calculation of the spectrum at frequency f there are also contributions of frequencies $f \pm 2nf_{Ny}$, n=1,2,3,…

$$f_{Ny} = \frac{1}{2dt}$$

That means dt has to be chosen such that f_N is the largest frequency contained in the signal.



The Fast Fourier Transform FFT

... spectral analysis became interesting for computing 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

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

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

The Fast Fourier Transform FFT

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

$$\begin{bmatrix} 1 & 1 & 1 & 1 & \dots & 1 \\ 1 & \omega & \omega^2 & \omega^3 & \dots & \omega^{N-1} \\ 1 & \omega^2 & \omega^4 & \omega^6 & \dots & \omega^{2N-2} \\ \vdots & \vdots & & & & \vdots \\ \vdots & \vdots & & & & \vdots \\ 1 & \omega^{N-1} & \cdots & \cdots & \cdots & \omega^{(N-1)^2} \end{bmatrix} \begin{bmatrix} F_0 \\ F_1 \\ F_2 \\ \vdots \\ \vdots \\ F_{N-1} \end{bmatrix} = \begin{bmatrix} f_0 \\ f_1 \\ f_2 \\ \vdots \\ \vdots \\ f_{N-1} \end{bmatrix}$$

.. where ...

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

FFT

... 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).

FFT

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.

Summary

- The Fourier Transform can be derived from the problem of approximating an arbitrary function.
- A regular set of points allows exact interpolation (or derivation) of arbitrary functions
- There are other basis functions (e.g., Chebyshev polynomials) with similar properties
- The discretization of signals has tremendous impact on the estimation of spectra: aliasing effect
- The FFT is at the heart of spectral analysis