Fourier transform – Fast FT and its application

- <u>summary</u> of Fourier transform and its properties
  - for an arbitrary $u \in L^2(\mathbb{R})$, the <u>Fourier transform</u> is defined as

$$\hat{u}(\xi) = (\mathcal{F}u)(\xi) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} u(x) e^{-i\xi x} dx \quad \text{for all } \xi \in \mathbb{R}$$

  - the image $\hat{u}(\xi)$ is also from $L^2(\mathbb{R})$ and its inverse FT is

$$u(x) = (\mathcal{F}\hat{u})(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \hat{u}(\xi) e^{i\xi x} d\xi, \quad x \in \mathbb{R}$$

  - generalization to more dimensions is straightforward
- Note: there are other definitions which differ in factors

  in general we can have

$$\hat{u}(\xi) = \frac{a}{\sqrt{2\pi}} \int_{-\infty}^{\infty} u(x) e^{-ic\xi x} dx \iff u(x) = \frac{b}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \hat{u}(\xi) e^{ic\xi x} d\xi$$

  but constants a, b, c must satisfy the condition

$$\frac{ab}{|c|} = 1 \quad \text{which follows from} \quad \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{ic\xi(x-x')} d\xi = \frac{1}{|c|} \delta(x-x')$$

- Parseval identity in general

$$\int_{-\infty}^{\infty} f(x) g^*(x) dx = \frac{|c|}{|a|^2} \int_{-\infty}^{\infty} \hat{f}(\xi) \hat{g}^*(\xi) d\xi \quad \text{or} \quad \|f\|_2 = \sqrt{\frac{|c|}{|a|^2}} \|\hat{f}\|_2$$

  and in our notation $(a=1, c=1)$ simply $\|f\|_2 = \|\hat{f}\|_2$

- Fourier transform of a convolution

$$(u * v)(x) = \int_{-\infty}^{\infty} u(x-y) v(y) dy = \int_{-\infty}^{\infty} u(y) v(x-y) dy$$

  is (for $u \in L^2, v \in L^1$ or the other way around)

$$\widehat{u * v}(\xi) = \hat{u}(\xi) \hat{v}(\xi)$$

- other properties
  - linearity: $\widehat{u+v}(\xi) = \hat{u}(\xi) + \hat{v}(\xi)$, $\widehat{cu}(\xi) = c\hat{u}(\xi)$
  - translation: for $x_0 \in \mathbb{R}$: $\widehat{u(x+x_0)}(\xi) = e^{i\xi x_0} \hat{u}(\xi)$
  - modulation: for $\xi_0 \in \mathbb{R}$: $\widehat{e^{i\xi_0 x} u(x)}(\xi) = \hat{u}(\xi - \xi_0)$
  - dilatation: for $c \in \mathbb{R}, c \neq 0$: $\widehat{u(cx)}(\xi) = \hat{u}(\xi/c)/|c|$
  - conjugation: $\widehat{u^*}(\xi) = \hat{u}(-\xi)^*$, - derivative: $\widehat{u_x}(\xi) = i\xi \hat{u}(\xi)$

• Discrete Fourier transform (DFT)

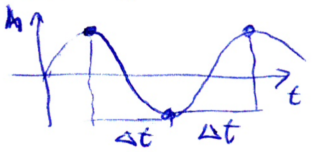
- motivation: let us consider a function $h(t)$, but evaluate it only with a certain sample frequency, i.e. at points

$$t_n = n \cdot \Delta t \quad , \text{ where } \quad n = \ldots -3, -2, -1, 0, 1, 2, 3 \ldots$$

(sampling rate is $\frac{1}{\Delta t}$ = number of sampling points per 1 s)

- the highest frequency which can be resolved from this sampling is Nyquist critical frequency

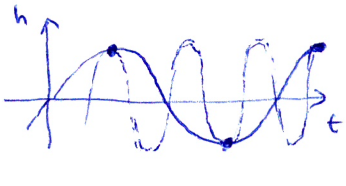
$$f_c = \frac{1}{2\Delta t}$$

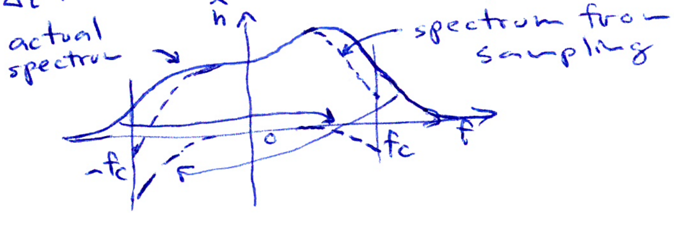


when we have two points per one period, which is the minimum to catch such a frequency, i.e. of the wave $\sin 2\pi f_c t$

- our goal is to use such sample points to get Fourier transform of $h(t)$ at least for frequencies in $\langle -f_c, f_c \rangle$

- but there is the aliasing problem
  - if the Fourier image of $h(t)$ has significant contribution for higher frequencies than f_c, it appears at lower frequencies of other sign
    - this phenomenon is called aliasing
  - the reason is that sampling with a time step Δt cannot distinguish frequencies which differ by a multiple of $\frac{1}{\Delta t}$, i.e. band width of $\langle -f_c, f_c \rangle$





- possible solutions:
  - sample according to the highest frequency contained in the signal
  - or use some filter to get rid of higher frequencies

- infinite series of samples would give a continuous image of $\langle -f_c, f_c \rangle$, but we usually work only with a finite samples at points $t_k = k\Delta t$, $k = 0, 1, \dots, N-1$ which provides only Fourier image at N frequencies between $-f_c$ and f_c; we use

$$f_n = \frac{n}{N\Delta t}, \quad n = \underbrace{-\frac{N}{2}}_{-f_c}, \dots, \underbrace{\frac{N}{2}}_{+f_c}$$

here we have actually $N+1$ frequencies but values at $-f_c$ and f_c will be the same

- using t_k and f_n, we get (for DFT, it is usually used $a = \sqrt{2\pi}$, $c = -2\pi$)

$$H(f_n) = \int_{-\infty}^{\infty} h(t)\, e^{2\pi i f_n t}\, dt \cong \sum_{k=0}^{N-1} h_k e^{2\pi i f_n t_k} \Delta t =$$

this is just approximation of $\hat{h}(f)$

$$= \Delta t \underbrace{\sum_{k=0}^{N-1} h_k e^{2\pi i k n/N}}_{H_n} = \Delta t H_n$$

this is called the discrete Fourier transform of the sequence h_k (note: it is not dependent on Δt!)

- even though we used $n = -\frac{N}{2}, \dots, \frac{N}{2}$, H_n is not changed if we add N to n, i.e.

$$H_{-n} = H_{N-n}$$

therefore, in codes, it is obtained H_n for $n = 0, \dots, N-1$ (as for t_k) (at least usually)

- but it is necessary to be careful, which frequencies correspond to values H_n, usually H_0 is for zero frequency

$H_1, \dots, H_{N/2-1}$ corresponds to positive frequencies $0 < f < f_c$

$H_{N/2+1}, \dots, H_{N-1}$ to negative frequencies $-f_c < f < 0$

and $H_{N/2}$ to both f_c and $-f_c$

- inverse discrete Fourier transform is then

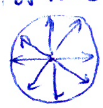
$$h_k = \frac{1}{N} \sum_{n=0}^{N-1} H_n e^{-2\pi i k n/N}$$

this follows from

$$\sum_{n=0}^{N-1} e^{2\pi i \frac{kn}{N}} = N\delta_{k0}$$

for $k = 0, \dots, N-1$

For $N=8$



which differs from DFT only in sign in the exponents and by factor $\frac{1}{N}$, thus it can be evaluated almost in the same way

• Fast Fourier transform (FFT)

- if we would evaluate DFT directly, we would need $\mathcal{O}(N^2)$ operations but there are ways how to do that using $\mathcal{O}(N \log_2 N)$ operations
  - such algorithms were developed by Cooley and Tukey in 1965 or by Danielson and Lanczos in 1942, which we apply here
- for $N = 2^x$, we can write

$$H_n = \sum_{k=0}^{N-1} e^{2\pi i k n / N} h_k =$$

$$= \sum_{k=0}^{N/2-1} e^{2\pi i n (2k)/N} h_{2k} + \sum_{k=0}^{N/2-1} e^{2\pi i n (2k+1)/N} h_{2k+1} =$$

$$= \sum_{k=0}^{N/2-1} e^{2\pi i n k/(N/2)} h_{2k} + W^n \sum_{k=0}^{N/2-1} e^{2\pi i n k/(\frac{N}{2})} h_{2k+1} =$$

$$= \underbrace{H_n^e}_{\text{DFT of even samples}} + W^n \underbrace{H_n^o}_{\text{DFT of odd samples in } h_k}$$

- if we apply this principle recursively on H_n^e and H_n^o we finally end up with a single value of the form

$$H_n^{eoee\ldots oe} = h_k \quad \text{for a certain } k \text{ corresponding to the odd-even pattern}$$

- it can be shown (just think about it for a while) that the correct index k can be expressed in the binary form as $eoee\ldots oe = 0100\ldots 110$ but in the reverse order i.e. $k = 011\ldots 0010_2$

. thus, it is necessary to reorder h_k's using <u>bit reversal algorithm</u>

example

$$\begin{array}{|c|}\hline 00 \\ \hline 01 \\ \hline 10 \\ \hline 11 \\ \hline\end{array} \quad \begin{array}{c} \to \\ \times \\ \\ \to \end{array} \quad \begin{array}{|c|}\hline 00 \\ \hline 01 \\ \hline 10 \\ \hline 11 \\ \hline\end{array}$$

it is possible to do this reordering in $\mathcal{O}(N \log_2 N)$ operations

- see notebook in Mathematica with the algorithm taken from Numerical Recipes

- there are also versions for $N \neq 2^x$ also of order $\mathcal{O}(N \log_2 N)$ but we will not discuss them here

Evaluation of the Fourier transform by FFT

- let us have a routine ~~that~~ returns

$$H_n = \sum_{k=0}^{N-1} h_k e^{2\pi i k n/N} \quad , \quad n = 0, \dots, N-1$$

i.e. discrete Fourier transform of h by the FFT algorithm

and we want to evaluate numerically the integral FT

$$I(\omega_m) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{i\omega_m t} h(t)\, dt$$

for a discrete set of frequencies which is actually given by the grid we use $t_k = t_{min} + k\Delta t$, $k = 0, \dots, N-1$

for which frequencies will be $\omega_m = \dfrac{2\pi m}{N\Delta t}$, $m = -\frac{N}{2}, \dots, \frac{N}{2}$

- using the trapezoidal rule we get

$$I(\omega_m) \simeq \frac{1}{\sqrt{2\pi}} \sum_{r=0}^{N-1} \overbrace{h(t_r)}^{h_r} e^{i\omega_m t_r} \Delta t = \quad t_r = t_{min} + r\Delta t$$

$$= \underbrace{\frac{\Delta t}{\sqrt{2\pi}} e^{i\omega_m t_{min}}} \underbrace{\sum_{r=0}^{N-1} h_r e^{2\pi i r m/N}}_{\tilde{H}_m} \quad \text{for } m = -\tfrac{N}{2}, \dots, \tfrac{N}{2}$$

$$\tilde{H}_m = \begin{cases} H_{m+N} & \text{for } m<0 \\ H_m & \text{for } m \geq 0 \end{cases}$$

this factor is cancelled when doing FT to reciprocal space and back

be careful with position of negative frequencies

! but be careful when evaluating the convolution $c(y) = \int_{-\infty}^{\infty} f(x)\, g(y-x)\, dx$ using FT because by doing FT of f and g we should use this factor for both, but then when going back using inverse FT we cancell only one factor, the second should be still there!

Example of the use of FT to solve time evolution in QM

- time evolution in quantum mechanics is described by the evolution operator $U = e^{-iHt}$ ($\hbar = 1$ in atomic units)

  where the Hamiltonian of the system is usually in the form

  $H = T + V$ with T being the kinetic-energy operator and V being the potential-energy operator

- T and V do not commute, thus we cannot write $e^{-iHt} = e^{-iTt} e^{-iVt}$ but for small time steps Δt, this expression is at least approximate
- it is better to use approximation of order $\mathcal{O}(\Delta t^2)$ called the <u>split-operator technique</u> which applies operators symmetrically

$$e^{-iH\Delta t} = e^{-\frac{iV\Delta t}{2}} \underbrace{e^{-iT\Delta t}} e^{-\frac{iV\Delta t}{2}} + \mathcal{O}(\Delta t^3)$$

in the p-representation it can be expressed as $e^{-i\frac{p^2}{2m}\Delta t}$ or similarly

and thus it is advantegous to apply $e^{-i\frac{V\Delta t}{2}}$ in the x-representation, but $e^{-iT\Delta t}$ in the p-representation

$\Rightarrow$ use of the Fourier transform to go from $\psi^x(x)$ to $\psi^p(p)$ and back

- if we use DFT (FFT) we have to be careful which p we use in $e^{-i\frac{p^2}{2m}\Delta t}$

  (the constant factors are not important if we apply both direct and inverse FT)

  the DFT image should be multiplied by $e^{-i\frac{p_j^2}{2m}\Delta t}$

  where $p_j = \dfrac{2\pi j}{N\Delta x}$ for $j = 0, \ldots, \frac{N}{2} - 1$

  and $p_j = \dfrac{2\pi}{\Delta x}(j - N)$ for $j = \frac{N}{2}, \ldots, N-1$

- there are generalizations to get methods of higher orders

  - see e.g. J. Chem. Phys. 150 (2019) 204113