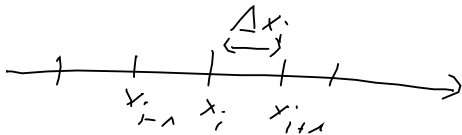


Fortsetzung Fourier Transformation

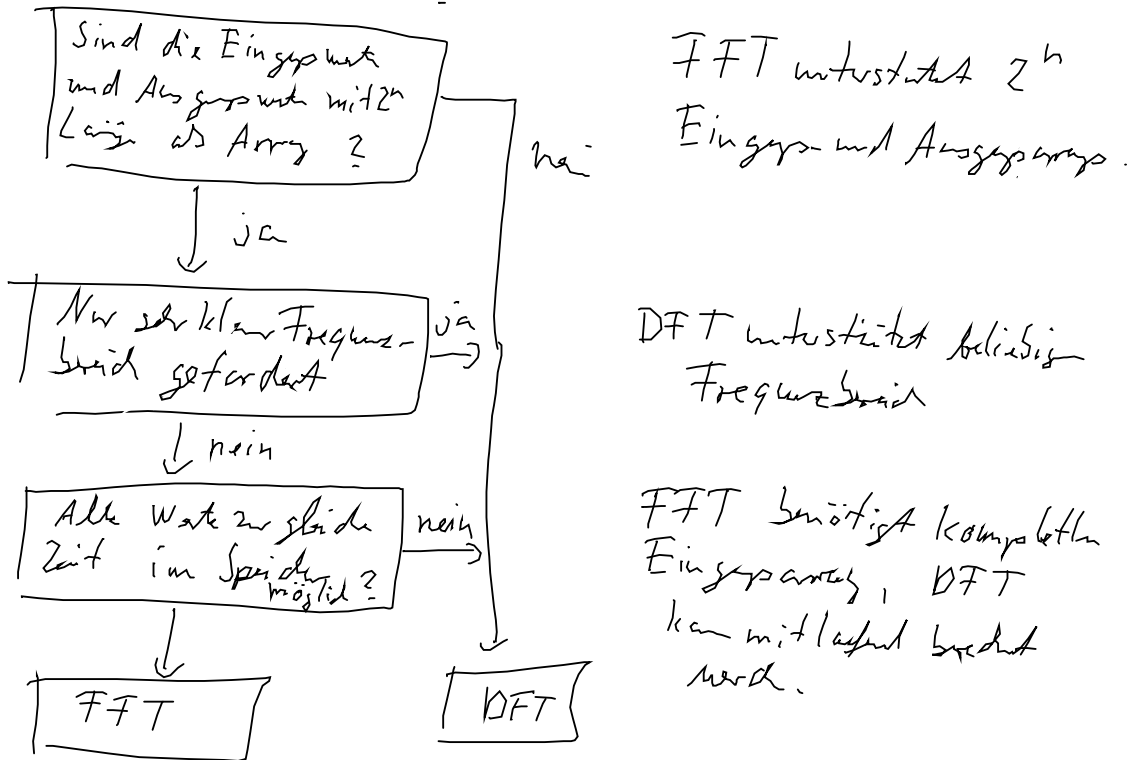
Erster Schritt zur Umsetzung diskretisieren:

$$f(k) = \int_{-\infty}^{\infty} dx e^{-ikx} f(x) \approx \sum \Delta x_i e^{-ik_j x_i} f(x_i)$$

\uparrow Intervalle \uparrow muss nicht äquidistant sein



FFT oder DFT?



FFT Algorithmen

Hauptidee: Bestimmung des Algorithmus der schon brendet Zwischenergebnisse.

Zunächst weitere diskretisierte Annahme $k_j = \Delta k \cdot j$ und $x_i = i \Delta x$ äquidistant, mit $\Delta k \cdot \Delta x = \frac{2\pi}{N}$ ← Anzahl der Punkte

$$f(k_j) = \sum_i \Delta x e^{-i \frac{2\pi}{N} i j} f(x_i) = \Delta x \sum_i \left(e^{-i \frac{2\pi}{N}} \right)^{i j} f(x_i)$$

$$= \Delta x \sum_{i=0}^{N-1} w_N^{i j} f(x_i) \quad \text{mit } w_N = e^{-i \frac{2\pi}{N}}$$

Man kann jetzt zeigen, dass die Fourier Transformierte von N Elementen auf zwei FT mit je $\frac{N}{2}$ Elementen zurückgeführt werden kann.

a) und zwar für gerade Elemente!

$$f(k_{2j}) = \Delta x \sum_{i=0}^{2n-1} w_N^{i \cdot 2j} f(x_i) = \Delta x \sum_{i=0}^{n-1} \left(w_N^{2ij} f(x_i) + w_N^{2(i+n)j} f(x_{i+n}) \right)$$

$$= \Delta x \sum_{i=0}^{n-1} \left(w_N^{2ij} (f(x_i) + f(x_{i+n})) \right)$$

Definiere $y_i = f(x_i) + f(x_{i+n})$

und beachte $w_N^2 = w_n$

Dann ist $f(k_{2j}) = \sum_{i=0}^{n-1} y_i w_n^{ij}$

b) und für ungerade Elemente

$$f(k_{2j+1}) = \Delta x \sum_{i=0}^{2n-1} f(x_i) w_N^{(2i+1)j} = \sum_{i=0}^{n-1} \Delta x \left[(f(x_i) - f(x_{i+n})) w_n^{ij} \right]$$

Definiere $y_{n+i} = (f(x_i) - f(x_{i+n})) w_n^{ij}$

$$f(k_{2j+1}) = \sum_{i=0}^{n-1} y_{n+i} w_n^{ij}$$

Dies Schritte a) und b) können sich falls N nie Potenz von 2 ist wiederholen!

Beweis $w_N^{2 \cdot 2n \cdot 2j} = w_N^{2n \cdot 2j} = w_n^{2j} = 1$

$$= w_N^{2j} = e^{-i \frac{2\pi}{N} 2j} = e^{-i \frac{2\pi}{n} j} = 1$$

Jeder Rechenschritt benötigt $\frac{N}{2}$ komplexe Multiplikationen
oder Rechenaufwand ($\frac{N}{2} \log_2 N$).

\Rightarrow Rekursive Implementate der FFT

Schnelle Implementate für FFT auf CPU ist FFTW
fastest Fourier Transform of the West (de facto Standard)

FFTWSS1 (parallelisiert)

DFT Algorithmus (Tipps zur Implementierung in der

Praxis): die Ausgangsliste muss nicht gespeichert werden
(Mitteln lassen des Algorithmus)

• Spezieller Spektralbereich interessiert.

z. B. $[w_{\text{start}}, w_{\text{end}}]$ z. B. mit M Punkten

Also $w_{\text{start}} + l \Delta w$

Dann hilft der folgende Trick (da $\exp T(k t + t_0)$!)

$$\Omega_{\text{start}}[l] = \exp(i(w_{\text{start}} + l \Delta w)t_0)$$

$$\text{z. B. } \Omega_{\text{start}}[0] = \exp(i w_{\text{start}} t_0)$$

$$\Omega_{\text{start}}[1] = \exp(i \Delta w t_0)$$

Verwendet: $\Omega_{\text{start}}[l+1] = \Omega_{\text{start}}[l] \times \Omega_{\text{start}}[1]$

Analysiere bare in Array:

$$\Omega_{\text{step}}[l] = \exp(i(w_{\text{start}} + l \Delta w) \Delta t)$$

Sowie in Array für die aktuelle Zeit

$$\Omega_{\text{start}}[l] = \Omega_{\text{step}}[l] \text{ initialisiert.}$$

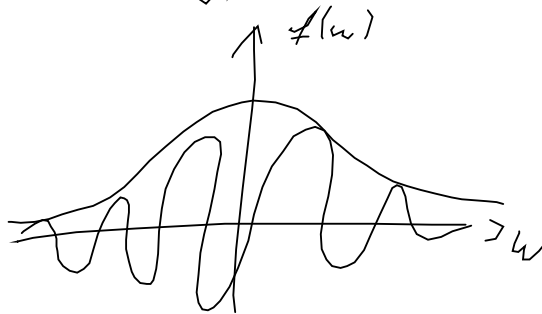
Von $t \rightarrow t + \Delta t$:

$$\Omega_{\text{start}}[l] *= \Omega_{\text{step}}[l]$$

Berechne DFT

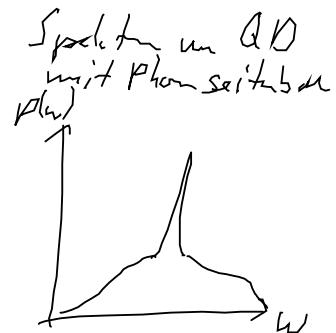
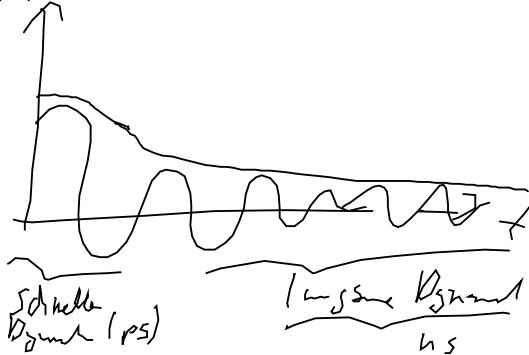
$$\text{DFT}[x] = \text{Omega}[x] \times f(H) \cdot \Delta t$$

Bemerkung: Typisch Verhalten



VI.15 Harmonic Inversion

$p(t)$ Nehmen wir an, wir wollen im Spektrum leben



Hier ist die Rate sehr gleichmäßig, so wie sehr niedrig die Dynamik bis in den Nanosekunden Bereich zu brechen

$$P(\omega) = \int_0^T p(t) e^{-i\omega t} dt \quad \text{für Four Transformation}$$

Mit der maximal Auflösung $\Delta E \approx \frac{1}{T}$

Ziel: Harmonic Inversion algorithm, der es erlaubt nur in kleinen Bereich der Zeitdynamik zu brechen und trotzdem das Spektrum zu erhalten.

Wir folgen Mandelstam und Taylor, J. Res. Phys. 107 (1957). Der Algorithmus ist die Basis der Library „harminv.“

Filter-Diagonalisierungs Methode für harmonische invariante

Idee: Wir tun so, als ob es ein fiktives Quantensystem gäbe, das keine Korrelation mit besteht.

Also betrachte $C(t) := \psi(t)$ und wir kritisieren $C_n := C(t_n) := C(nT)$, wir tun so als gäbe es \hat{U} und \hat{H} mit $|\phi_0\rangle$, so dass

$$C_n = \underbrace{\langle \phi_0 | \hat{U}^n | \phi_0 \rangle}_{\text{Zeitpropagator}} = \langle \phi_0 | e^{-i n T \hat{H}} | \phi_0 \rangle$$

Wobei die Zeitpropagator bzgl. Eigenbasis $|Y_k\rangle$

$$\hat{U} = \sum_{k=1}^N u_k |Y_k\rangle \langle Y_k| \quad \text{und der Eigenwert } u_k = e^{-i T \omega_k}$$

Wir können dann die beiden schreiben

$$C_n = \sum_{k=1}^N \underbrace{\langle \phi_0 | Y_k \rangle}_{d_k} u_k^n = \sum_{k=1}^N d_k e^{-i n T \omega_k}$$

Idee: Jedes Signal läßt sich durch abstellende Exp. Reihe beschreiben.

← Ziel von Harmonische Invarianz ist es d_k und ω_k zu extrahieren.

Für eindeutig Frequenz, kann man $-\frac{\pi}{T} < \text{Re} \omega_k < \frac{\pi}{T}$ wählen.

⇒ Nyquist Rate, sowie $\text{Im} \omega_k \leq 0(N \geq 0)$ (sollte innerhalb des Intervalls selbstständig zerfallen)

⇒ Wie können wir die Frequenz ω_k bekommen?

⇒ Diagonalisieren \hat{H} oder \hat{U} diagonalisieren.

Ausgabe: Das wäre möglich?! Wähle vollständige Basis $|q_j\rangle$
 $[j \in \{1, \dots, k\}]$

Das ist

$$u_{j-1} = \langle q_j | \hat{U} | q_{j-1} \rangle$$

$$s_{j-1} = \langle q_j | q_{j-1} \rangle$$

führt das zu reellwertigen Eigenwertproblem (Addy die Vektoren sind nicht orthogonal)

$$\underline{U} \cdot \underline{B}_k = u_k \underline{S} \cdot \underline{B}_k$$

$\hookrightarrow B_k$ sind die Eigenvektoren, Darstellung der Spaltenvektoren als

$$|Y_k\rangle = \sum_{j=1}^k B_{jk} |q_j\rangle$$

best. der Basis $|q_j\rangle$

$$\text{wobei } d_k = |\langle d_0 | Y_k \rangle|^2 = \left| \sum_{j=1}^k B_{jk} \langle d_0 | q_j \rangle \right|^2$$

Problem: Was ist die Basis die wir wollen?

Erste Idee \Rightarrow Krylov Basis zu U !

$$|\Phi_n\rangle = U^n |d_0\rangle \quad \text{mit } n = 0, 1, \dots, k.$$

Für $M = k-1$ ist die Basis vollständig, für $M > k-1$ übervollständig.

Die Matrixelemente haben dann die direkte Beziehung zu u_i

Wie sieht u_{j-1} in der Basis aus:

$$\langle \Phi_n | \hat{U} | \Phi_{n-1} \rangle = \langle \Phi_n | (U^n) (U^{n-1}) | d_0 \rangle = \langle \Phi_n | U^{2n-1} | d_0 \rangle$$

$$\text{Sinn Überlappung} = \langle \psi_{n+1} | \psi_n \rangle$$

$$\langle \psi_n | \psi_{n+1} \rangle = \langle \psi_n | \psi_n \rangle = 1 \quad \langle \psi_n | \psi_0 \rangle = \delta_{n0}$$

Das Eigenwertproblem zur Lösung hat die Form:

$$\sum_{n=0}^M c_{n+1} B_{nk} = \epsilon_k \sum_{n=0}^M c_{n+1} B_{nk} \quad \text{mit dem Eigenw.}$$

$$\epsilon_k = e^{-i\epsilon_k}$$

$$d_k = \left(\sum_{n=0}^M B_{nk} c_n \right)$$