

# Fouriertransformation-Notizen

13. November 2020

# Inhaltsverzeichnis

<b>1</b>	<b>Todo</b>	<b>1</b>
<b>2</b>	<b>Frequenz, Periode, Kreisfrequenz</b>	<b>1</b>
<b>3</b>	<b>Sinus plus Sinus</b>	<b>1</b>
<b>4</b>	<b>Nyquist-Frequenz</b>	<b>2</b>
<b>5</b>	<b>DFT-Auflösung</b>	<b>2</b>
5.1	Welche Frequenz gehört zu welchem $m$ ? . . . . .	3
<b>6</b>	<b>DFT in Python</b>	<b>3</b>
6.1	fftfreq . . . . .	3
<b>7</b>	<b>Zeropadding</b>	<b>3</b>
<b>8</b>	<b>Weitere Gedanken</b>	<b>5</b>
<b>9</b>	<b>Window functions</b>	<b>5</b>

## 1 Todo

- warum fftshift nutzen? Wieso ist das verschieben wichtig, wo kommt das her?
- Periodizität (va auch von DFT)

## 2 Frequenz, Periode, Kreisfrequenz

Die Kreisfrequenz gibt den überstrichenen Phasenwinkel der Schwingung pro Zeitspanne an. Da die Frequenz angibt, wie viele Schwingungen es pro Zeitspanne gibt und ein Phasenwinkel von  $2\pi$  einer Schwingungsperiode entspricht, ist  $\omega$  das  $2\pi$ -fache der Frequenz.

*Wenn  $f = 2\text{Hz}$ , dann wird pro Sekunde der Phasenwinkel  $2 \cdot 2\pi$  überstrichen.*

Wenn ich einfach nur den  $\sin(x)$  plote, erhalte ich eine harmonische Schwingung mit der Periode  $T = 2\pi$ , weil der Phasenwinkel bis  $2\pi$  einmal durchgegangen wird, bis eine Periode fertig ist. Ein Phasenwinkel von  $2\pi$  entspricht einer Schwingungsperiode. In diesem Fall hätte meine Schwingung eine Frequenz von  $f = 1/2\pi \sim 0.16\text{Hz}$ .

Wenn ich jetzt sage, dass ich eine harmonische Schwingung mit der Frequenz  $f$  haben möchte, muss man die Funktion so strecken, bzw stauchen, dass man auf die Periode  $T = 1/f$  kommt. Die Stauchung von  $T = 2\pi$  auf  $T = 1$  für die Funktion  $\sin(x)$  wird durch den Vorfaktor  $2\pi$  für  $x$  erhalten, da dadurch die Fkt um das  $2\pi$ -fache gestaucht wird. Dadurch habe ich jetzt eine Schwingung mit der Periode  $T = 1$  und damit  $f = 1$ . Wenn ich jetzt  $f$  erhöhen will, führt dies wieder zu einer Stauchung, da beispielsweise für  $f = 2$  jetzt zwei Perioden in eine Sekunde passen müssen, also wieder eine Stauchung um 2. Insgesamt:  $\sin(2\pi f x)$ , also  $\sin(\omega x)$ .

## 3 Sinus plus Sinus

Wenn man eine Sinusähnliche Welle mit einer anderen addiert, erhält man erneut eine Sinusförmige Welle. Das lässt sich über ein Zeigerdiagramm veranschaulichen.

Als Beispiel will ich die folgende Summe berechnen:

$$A \cos(\omega x) + B \sin(\omega x) \quad (1)$$

In diesem Bsp haben beide Wellen unterschiedliche Amplituden, aber die selbe Frequenz und die selbe Phase, nämlich Null. Zur Verdeutlichung die Abb. 1. In schwarz soll der Sinus dargestellt sein und er startet bei  $x = B$ . In dem Bild ist jetzt  $\omega x$  als  $\omega t$  geschrieben. Wenn jetzt ein beliebiger Punkt in der Zeit, (oder in  $x$ ) erreicht ist, dann steht der Sinus ja irgendwo und das soll hier oBdA zum Zeitpunkt des schwarzen Vektors sein. Der Cosinus ist in orange dargestellt und da er einfach nur ein um  $\frac{\pi}{2}$  verschobener Sinus ist ( $\cos(\omega t) = \sin(\omega t + \frac{\pi}{2})$ ), ist er hier auch so dargestellt (unter der Annahme, dass  $A$  kleiner als  $B$  ist).

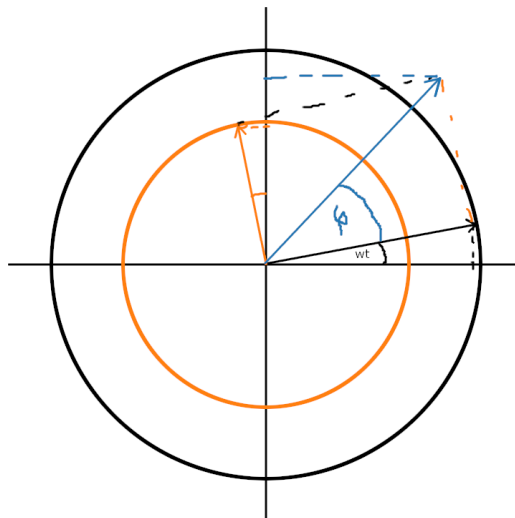


Abbildung 1: Veranschaulichung

Jetzt will man ja die Summe beider Sinusfunktionen haben. Das kann man machen, indem man die jeweiligen y-Komponenten der Vektoren zum gegebenen Zeitpunkt addiert. Es werden einfach direkt beide Vektoren addiert und es kommt der blaue Vektor raus, dessen y-Komponente der Summer der Sinusse entspricht. Wenn man das jetzt zu jeder Zeit macht, dreht sich der blaue Vektor mit der selben Frequenz wie die anderen beiden und seine y-Komponente entspricht immer der Summe der Sinusse.

Der entscheidende Gedanke ist jetzt: Das ist eine eigene Sinusschwingung! Die Amplitude davon ergibt sich über den Pythagoras. Man muss noch beachten, dass der neue Sinus mit einer Phase  $\phi$  beginnt zu schwingen und die ergibt sich über  $\tan^{-1}(\frac{A}{B})$ . Das Ergebnis sieht also so aus:

$$\sqrt{A^2 + B^2} \sin\left(\omega t + \tan^{-1}\left(\frac{A}{B}\right)\right) = C \sin(\omega t + \phi) \quad (2)$$

Wenn die Ausgangssinusse noch Phasen haben, dann ist die Herangehensweise die selbe, es kommen nur noch ein paar Ausdrücke dazu, weil man die eben mit berücksichtigen muss.

## 4 Nyquist-Frequenz

$$f_{\text{nyquist}} = \frac{1}{2} \cdot f_s \quad (3)$$

Sie ist definiert als die halbe Abtastfrequenz (Samplefrequenz). Nach dem dazugehörigen Abtasttheorem, müssen alle Anteile in einem Signal kleinere Frequenzen haben, als die Nyquist-Frequenz, damit das abgetastete Signal beliebig genau rekonstruiert werden kann.

$$f_{\text{signal}} < f_{\text{nyquist}} \quad (4)$$

Also muss die Abtastfrequenz größer als doppelt so groß wie die Frequenz des abzurasternden Signals sein.

## 5 DFT-Auflösung

*Dieses Kapitel ist sehr durcheinander*

Die diskrete FT ist erstmal nur eine Abbildung, die eine Liste von komplexen Zahlen in eine andere Liste komplexer Zahlen abbildet unter der Annahme, dass das "Eingangssignal" einer Periode eines periodischen Signals entspricht.

$$G(m) = \sum_{n=0}^{N-1} e^{-2\pi i \frac{kn}{N}} g(n) \quad (5)$$

Dabei ist  $G(m)$  genauso lang wie  $g(n)$ , nämlich  $N$  Einträge. Das ganze findet bisher ohne Rücksicht auf die Auflösung im Messraum statt. Gleichung 5 ausgeschrieben ist:

$$G(m) = \sum_{n=0}^{N-1} g(n) \left[ \cos\left(2\pi \frac{mn}{N}\right) - i \sin\left(2\pi \frac{mn}{N}\right) \right] \quad (6)$$

und man bezeichnet die cosinus und sinus Funktionen in nennt man diskrete Basisfunktionen  $C_m^N(n)$  und  $S_m^N(n)$ . Die inverse DFT sieht ja bis auf das Plus recht ähnlich aus. Jede der eindimensionalen Basisfunktionen

$$C_m^N(n) = C_n^N(m) = \cos\left(2\pi \frac{mn}{N}\right) \quad (7)$$

$$S_m^N(n) = S_n^N(m) = \sin\left(2\pi \frac{mn}{N}\right) \quad (8)$$

Damit sind sie "periodisch in  $N$ ": Wenn ich in einer Funktion das Argument durch  $N$  teile, dann strecke ich die Fkt um den Faktor  $N$ . Durch dieses  $N$  wird bewirkt, dass eine Periode, welche für  $m = 1$  genau den Wert 1 hätte, sich nun bis  $N$  streckt. Die Frequenz der Funktionen ist also  $m/N$  und damit die Kreisfrequenz

$\omega_m = 2\pi m/N$ . Die Zahl  $m$  wird in diesem Beispiel "Wellenzahl" genannt. Wenn jetzt zB  $N = 8$  und  $m = 4$ , dann benötige ich zur sinnvollen Abrasterung mindestens eine  $f_{\text{abstast}} > 2 \frac{m}{M} = 1\text{Hz}$ .

Da  $M$  für  $m = 1$  die Periode von 1 auf  $M$  streckt und  $m$  lediglich dafür sorgt, dass in  $M$  mehrere Perioden reinpassen, ist  $M$  selbst jedoch auch immer noch ein Punkt ab dem sich dann das Signal welches aus den  $m$  Perioden besteht wiederholt.

*M ist ja die Anzahl an Samplewerten die ich habe. Wenn ich 1000 Messpunkte habe in meinem Sample, dann ist  $M = 1000$  und ich weiß zwar nicht nicht genau, woher die Annahme kommt, dass dieses Sample eine Periode eines periodischen Signals darstellt, aber da diese Annahme da ist, ist das so. Das  $m$  geht dann seine Werte durch und für jedes  $m$  wird  $G(m)$  ausgerechnet. Wenn  $m = 1$ , bleibt in den Basisfunktionen nur  $2\pi \frac{n}{M}$  stehen, die haben dann die Frequenz  $1/M$ . Aber bisher wurden noch keine Einheiten betrachtet glaube ich, das heißt ich betrachte auch in welchem Abstand die Werte gesamplet wurden. Sei der Abstand zwischen zwei Samplepunkten  $\tau$ , dann decken meine  $M$  Punkte eine Distanz von  $M\tau$  ab (ich dachte erst  $(M - 1)\tau$ , aber die nächste Periode beginnt erst im Abstand  $\tau$  vom letzten Samplepunkt).*

## 5.1 Welche Frequenz gehört zu welchem $m$ ?

*Dieses Kapitel ist sehr durcheinander* Ich habe genauso viele Werte im Spektrum, wie ich Datenpunkte habe, nämlich  $M$  und decke damit eine Zeitspanne von  $M\tau$  ab. Für  $m = 1$  ergibt das für die Basisfkt eine Frequenz von  $1/M$ . Ist die Frequenz der Basisfunktionen gleichzusetzen mit der Frequenz, zu der die  $m$  gehören, also die die `fftfreq()` liefert? Ich denke nicht, weil die Frequenzen die zu den  $m$  gehören über folgendes definiert sind:

$$f_m = m \cdot \frac{1}{M\tau} \quad (9)$$

Die haben also einen Bezug zur Periode des gesamten Samples, welche ja  $M\tau$  ist.

Die Frequenzauflösung ist umgekehrt proportional zur Länge des Zeitfensters:

$$\Delta f = \frac{1}{M\tau} \quad (10)$$

und geht damit von 0 bis  $\frac{M-1}{M\tau}$ , weil  $m$  von 0 bis  $M - 1$  geht.

Das Spektrum eines reellwertigen Signals ist symmetrisch um den Ursprung.

## 6 DFT in Python

### 6.1 fftfreq

## 7 Zeropadding

Egal, wie viele zusätzliche Punkte man im Frequenzraum durch hinzufügen von Nullen erhält, erhält man dadurch keine bessere Frequenzauflösung. Damit meine ich, dass zwei Frequenzen die nahe beieinander liegen erst dann als separater Peaks im Spektrum auftauchen, wenn man mehr Daten der eigentlichen Funktion zum Transformieren nutzt.

Im Bsp mit 1MHz und 1.05MHz: Wenn ich dazu Punkte in einem Sample habe welches  $10\mu\text{s}$  abdeckt (also ein  $f_s = 1\text{e}8\text{Hz}$ ) und dazu noch 6000 Samples im selben Abstand per Zeropadding dazufüge, erhöht sich dadurch zwar meine Auflösung im Frequenzraum, jedoch habe ich immer noch nur 1000 Punkte von meiner eigentlichen Funktion und können nur Signale mit Frequenzunterschiede von mehr als 100Khz unterschieden werden (*Das ergibt sich über  $\frac{f_s}{N_{\text{fft}}} = \frac{1\text{e}8}{1\text{e}3} = 1\text{e}5 = 100\text{kHz}$* ). Da die beiden Sinuse sich in ihrer Frequenz jedoch nur um 50kHz unterscheiden, hat man gar nicht die Informationen, um die beiden Peaks getrennt sichtbar zu machen.

Wenn ich jetzt statt 1000 Datenpunkten 7000 nehme, dann kann ich Frequenzen bis  $\frac{1}{1\text{e}-8 \cdot 7000} \approx 14285.7$  auflösen.

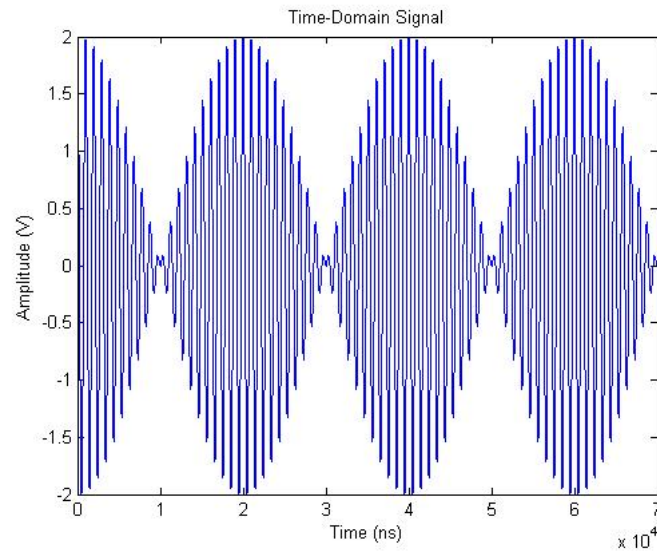


Abbildung 2: Daten

Wenn ich das in dem Beispiel aus Bild 2 mache, ergibt sich eine DFT mit dem Ausschnitt aus Abb. 3.

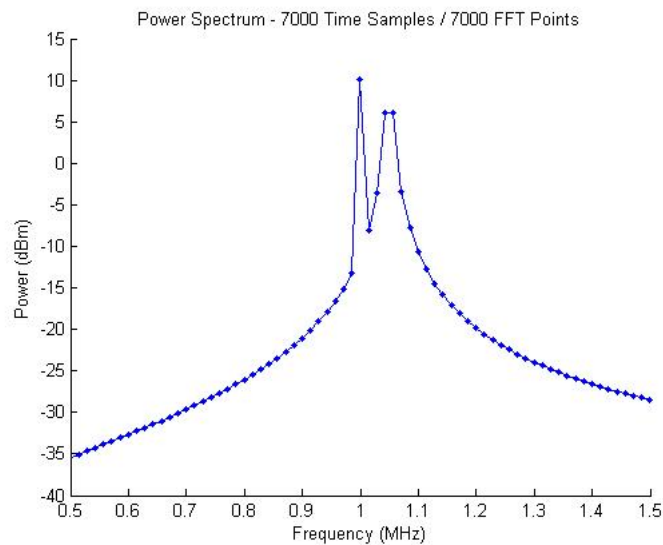


Abbildung 3: DFT

Man erkennt, dass der Peak bei 1MHz einzeln dargestellt wird, während der bei 1.05MHz auf zwei bins aufgeteilt wird. Das liegt daran, dass wir keinen Punkt bei exakt 1.05 haben. 1Mhz ist ein Vielfaches von 14285.7, da 1Mhz in diesem Bsp auch gerade der Samplingfrequenz entspricht. Das gilt nicht für die 1.05MHz: die nächsten Frequenzen sind 1.043 und 1.057 MHz und daher wird die "Energie" gleichermaßen zwischen diesen Bins aufgeteilt.

Um diesen Problem nun zu lösen, sollten beide Frequenzen bei einzelnen Punkten im Frequenzspektrum liegen. Da wir durch die genügend hohe Samplezahl die Informationen zur Auflösung bereits haben, kann hier nun zeropadding genutzt werden, um die Auflösung zu verfeinern.

## 8 Weitere Gedanken

$\Delta f = \frac{f_s}{N_{\text{fft}}} = \frac{1}{\tau N_{\text{fft}}}$ . Das würde bedeuten, dass ich meine Frequenzauflösung dadurch verbessere indem ich meine Samplingfrequenz verringere. Bei gleich bleibender Sampleanzahl würde das bedeuten, dass ich dadurch einen längeren Zeitabschnitt betrachte.

## 9 Window functions

Wenn ich eine Fouriertrafo mache, multipliziere ich durch die Auswahl des Samples meine zu beobachtende Funktion quasi mit einer Rechteckfunktion in dem Bereich des Samples. Diese Windowoperation verändert mein FT-Ergebnis, nämlich in dem Sinne, dass die FT die ich beobachte die Faltung der Frequenzantwort des Windows mit der der eigentlichen Funktion ist. Das Frequenzspektrum einer Rechteckfkt ist ein sinc mit Seitenbändern, welche prominenter sind, wenn das Fenster schmal ist.